Software Analytics in Continuous Delivery: A Case Study on Success Factors

Huijgens, Hennie; Spadini, Davide; Stevens, Dick; Visser, Niels; van Deursen, Arie

DOI
10.1145/3239235.3240505

Publication date
2018

Document Version
Accepted author manuscript

Published in
ESEM' 18

Citation (APA)
Huijgens, H., Spadini, D., Stevens, D., Visser, N., & van Deursen, A. (2018). Software Analytics in Continuous Delivery: A Case Study on Success Factors. In *ESEM' 18: Proceedings of the 12th International Symposium on Empirical Software Engineering and Measurement* (pp. 1-10). [25] Association for Computing Machinery (ACM). https://doi.org/10.1145/3239235.3240505

Delft University of Technology
Software Engineering Research Group
Technical Report Series

Software Analytics in Continuous Delivery: A Case Study on Success Factors

Hennie Huijgens, Davide Spadini, Dick Stevens
Niels Visser, Arie van Deursen

Report TUD-SERG-2018-02

TUDelft

SERG

TUD-SERG-2018-02

Published, produced and distributed by:

Software Engineering Research Group
Department of Software Technology
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology
Mekelweg 4
2628 CD Delft
The Netherlands

ISSN 1872-5392

Software Engineering Research Group Technical Reports:
`http://www.se.ewi.tudelft.nl/techreports/`

For more information about the Software Engineering Research Group:
`http://www.se.ewi.tudelft.nl/`

Note: Accepted for publication in the Proceedings of the 12th International Symposium on Empirical Software Engineering and Measurement (ESEM 2018).
`https://doi.org/10.1145/3239235.3240505`

Software Analytics in Continuous Delivery: A Case Study on Success Factors

Hennie Huijgens
ING Tech Research
Amsterdam, The Netherlands
hennie.huijgens@ing.com

Davide Spadini
Delft University of Technology
Delft, The Netherlands
d.spadini@tudelft.nl

Dick Stevens
ING
Amsterdam, The Netherlands
dick.stevens@ing.com

Niels Visser
ING
Amsterdam, The Netherlands
niels.visser@ing.com

Arie van Deursen
Delft University of Technology
Delft, The Netherlands
arie.vandeursen@tudelft.nl

ABSTRACT

Background: During the period of one year, ING developed an approach for software analytics within an environment of a large number of software engineering teams working in a Continuous Delivery as a Service setting. *Goal:* Our objective is to examine what factors helped and hindered the implementation of software analytics in such an environment, in order to improve future software analytics activities. *Method:* We analyzed artifacts delivered by the software analytics project, and performed semi-structured interviews with 15 stakeholders. *Results:* We identified 16 factors that helped the implementation of software analytics, and 20 factors that hindered the project. *Conclusions:* Upfront defining and communicating the aims, standardization of data at an early stage, build efficient visualizations, and an empirical approach help companies to improve software analytics projects.

CCS CONCEPTS

• **Software and its engineering** → *Empirical software validation*;

KEYWORDS

Software Economics, Software Analytics, DevOps, Continuous Delivery, Experience Report, ING

ACM Reference Format:
Hennie Huijgens, Davide Spadini, Dick Stevens, Niels Visser, and Arie van Deursen. 2018. Software Analytics in Continuous Delivery: A Case Study on Success Factors. In *ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM) (ESEM '18), October 11–12, 2018, Oulu, Finland.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3239235.3240505

1 INTRODUCTION

Software analytics is a well-known practice that uses analysis, data, and systematic reasoning for decision making on software data for managers and software engineers. It aims to empower software development individuals and teams to gain and share insight from their data, to make better decisions [21] [3]. Software engineering lends itself well to benefit from analytics: it is data rich, labor intensive, time dependent, dependent on consistency and control, dependent on distributed decision-making, and has a low average success rate [3]. Although much research has been conducted on software analytics, little work has covered its implementation in practice, and even less in a continuous delivery setting [25] [28].

This paper reports on the experience of deploying software analytics into a continuous delivery process at a bank. We conducted structured interviews with 15 project stakeholders in many roles, on five topics: 1) goals of the analytics project; 2) getting data; 3) analyzing data; 4) visualization; 5) and collaboration with researchers. For each topic, the interviews gathered Likert-scale data about what was perceived as positive or negative, with open-ended questions to discover why. By coding transcriptions of the open-ended responses, we report on factors that help and hinder each of the topics.

1.1 Background and Terminology

ING is a large Netherlands-based bank, operating worldwide, with a strong focus on technology and software engineering. The bank is in the midst of a technology shift from a pure finance-oriented to an engineering-driven company. In recent years, ING has implemented a fully automated release engineering pipeline for its software engineering activities in more than 400 teams, which perform more than 2500 deployments to production each month on more than 750 different applications.

This release engineering pipeline - based on the model described by Humble and Farley [7] - is within ING known as *CDaaS*, an abbreviation of *Continuous Delivery as a Service*. Its main goal is to support teams in maximizing the benefits of shared use of tools. The pipeline fully automates the software release process for software applications. It contains several specialized tools which are connected to each other, such as ServiceNow (backlog management), GitLab (collaboration), Jenkins (code), SonarQube (code inspection), OWasp (security), Artifactory (build), and Nolio (deploy).

The mindset behind CDaaS is to move to production as fast as possible, while maintaining or improving quality. CDaaS is at the core of a transition that is ongoing within ING towards BizDevOps, a model were software developers, business staff, and operations

ESEM '18, October 11–12, 2018, Oulu, Finland

ACM ISBN 978-1-4503-5823-1/18/10…$15.00
https://doi.org/10.1145/3239235.3240505

staff work together in one team that works in an agile (Scrum or Kanban) way. The idea behind this is that teams - or *squads*, as they are called within ING in line with terminology used at Spotify [17] - develop software more quickly, be more responsive to user demand, and ultimately maximize revenue.

1.2 The GAMe-project

In the midst of implementing these CD-pipelines and the new way of working in a large number of its global teams, ING started a software analytics project, called *GAMe*; an abbreviation of *Global Agile Metrics*. At the time of conducting this study, the GAMe-project has run for approximately one year. A team of software engineers with a data warehouse, software analytics, or business intelligence background implemented an infrastructure that was based on continuous automated mining of log-files of tools within the software engineering pipeline. Log data was analyzed based on multiple regression analysis, resulting in so-called *strong metrics*: metrics that are highly correlated with (and hence have strong predictive power for) future software deliveries.

Three of the five authors of the present study were actively involved in the GAMe-project, the first author as research advisor, the third as project manager, and the fourth as business consultant. In this study we built on earlier work on the GAMe-project [13], in which we focused specifically on the analysis of strong metrics, whereas this paper provides insights in the implementation aspects of such a project itself.

1.3 Problem Statement

Where financial organizations, such as banks, traditionally focused their research activities on financial-, risk- and economic-oriented aspects, nowadays a new focus on technological issues is gaining more and more attention [7][10]. With this technology shift, new horizons towards analytics open up. We assume that - e.g. due to a trend towards automation - also research topics such as software analytics are strongly influenced by contemporary technological developments such as pipeline-automation, continuous delivery, and shorter iterations. With this in mind, we address the following research question:

RQ1: What factors affected the experience of implementing the GAMe-project's software analytics solution within ING's continuous delivery squads?

RQ2: What can be done to improve future implementations of software analytics solutions in such environments?

Better understanding of the implementation of a software analytics solution - where specific research capabilities are closely linked with the daily practice of software engineering - will help both researchers and practitioners to optimize their collaboration in future software analytics projects. The GAMe-project also serves as a starting point for a longer term collaboration between ING and the Delft University of Technology. For that reason we use the context of this project to address a third research question:

RQ3: To what extent can collaboration of practitioners and researchers help to improve future software analytics projects?

The remainder of this paper is structured as follows. In Section 2 related work is described. Section 3 outlines the research design. The results of the study are described in Section 4. We discuss the results in Section 5, and finally, in Section 6 we draw conclusions and outline future work.

2 RELATED WORK

Three challenges apply regarding software analytics in practice [28] [21]: 1) How to know that analysis output is insightful and actionable? 2) How to know that you use the right data? 3) How to evaluate analysis techniques in real-world settings?

Buse and Zimmermann [3] point out a model for analytical questions, adapted from [5]. The model distinguishes between information – which can be directly measured – and insight – which arises from analysis and provides a basis for action.

A large scale survey with 793 professional data scientists at Microsoft [16] reveals 9 distinct clusters of data scientists, and their corresponding characteristics. A small number of recent, other studies on how data scientists work in practice has been performed, such as Fisher et al. [9] and Kandel et al. [15]. Although both studies where relatively small and with a more general focus, they revealed equal challenges as found in the Microsoft study. Related to earlier mentioned challenges regarding data quality, Agrawal and Menzies [2] found that for software analytic tasks like defect prediction, data pre-processing can be more important than classifier choice, while ranking studies are incomplete without such pre-processing.

Collaboration of researchers and practitioners in software analytics is a slowly emerging topic in software engineering research. As a result, most studies are relatively small and exploratory in nature (e.g. [11] [8] [19] [23] [27]).

A vast number of studies can be found about continuous delivery - sometimes described as release engineering. Although software analytics is mentioned in many of these, a focus dedicated to the implementation aspects of software analytics in such an environment is lacking in most of them. However, one aspect that is mentioned in every study, is the power of automation for analytics purposes.

Waller et al. [26] examine for example how automated performance benchmarks may be included into continuous integration. Adams et al.[1] emphasize the fact that every product release must meet an expected level of quality, and release processes undergo continual fine tuning. They argue that release engineering is not taught in class at universities; the approaches are quite diverse in nature and scope - an indication that the same goes for software analytics in such an environment?

Others, such as Matilla et al. [20] inventory analytics to visualize the continuous delivery process maturity itself. Chen [4] describes six strategies to help overcome the adoption challenges of continuous delivery.

Misirli et al. [22] mention three common themes for software analytics projects they examined: 1) increase the model output's information content with, for example, defect-severity or defect-type prediction, defect location, and phase- or requirement-level effort estimation; 2) provide tool support to collect accurate and complete data; and, 3) integrate prediction models into existing systems, for example, by combining the results of defect prediction with test interfaces to decide which interfaces to test first, or creating a plug-in that seamlessly works in development and testing environments.

To emphasize the fact that not much has been written about software analytics in a continuous delivery context; Laukkanen et al. [18] identified a total of 40 problems, 28 causal relationships and 29 solutions related to adoption of continuous delivery, yet, software analytics is not one of them; the phrase is not even mentioned in the study. Fitzgerald and Stol [10] describe a roadmap and an agenda for continuous software engineering. Again, also in this study software analytics is not mentioned once.

Therefore, the innovation of our study is, that we perform a case study with quantitative and qualitative data in a large continuous delivery setting, with a focus specific on the implementation aspects of software analytics.

3 RESEARCH DESIGN

Our study methodology involved inventory of artifacts from the GAMe-project, and a series of semi-structured interviews, that included a number of quantitative survey questions. All study materials can be found in our technical report [14].

3.1 Artifacts from the GAMe-project

To get an overview of the results that were delivered within the scope of the GAMe-project, we collected and analyzed artifacts such as overview of collected data, structured metrics, results of data analysis, dashboards, and other visualizations.

3.2 Interviews with Stakeholders

To examine underlying reasons and causes of aspects that helped or hindered the GAMe-project, we opted for semi-structured interviews with its stakeholders [6] [24]. The interviews were performed in an open way, allowing new ideas to be brought up during the interview as a result of what the interviewee says.

Discussion upfront among the group of authors who participated in the project, taught us that the stakeholders of the GAMe-project - based on their main task in the project - could roughly be mapped on the three technology pillars in the Microsoft Research Model for Software Analytics: 1) large-scale computing, 2) analysis algorithms, and 3) information visualization [28]. On top of that we added two topics to the interviews. The first one covers the aims of the project as experienced by its stakeholders. The second topic addresses collaboration between scientists and practitioners from industry in the context of the GAMe-project. Based on these assumptions, we grouped interview questions into five topics, that we used as a framework of themes to be explored:

(1) Purpose and aim of the GAMe-project.
(2) Large-scale Computing: Getting the Data.
(3) Analysis Algorithms: Analyzing the Data.
(4) Information Visualization and action-ability of dashboards.
(5) Research collaboration.

To reduce the risk of missing important topics we ended each interview with an open question about remaining issues to address.

Survey-questions as part of the interviews. Each of the five interview topics was preceded by two or three survey-questions that ask to what extent the interviewee agrees with a statement, followed by a statement, and a 1 to 5 point Likert-scale (strongly disagree - disagree - neutral - agree - strongly agree - don't know).

Table 1: Overview of Interviewees

ID	Role	Organization
P01	Product Owner	CDaaS Squad
P02	Pipeline / Software Engineer	CDaaS Squad
P03	Team Lead Engineering	CDaaS Squad
P04	Data Warehouse Engineer	Data Warehouse Squad
P05	Product Owner	Infrastructure Squad
P06	GAMe Project Manager	ING Tech
P07	Data Analyst / R-programmer	ING Tech
P08	Information Manager	Dashboard Squad
P09	Business Consultant	Dashboard Squad
P10	Dashboard Engineer	Dashboard Squad
P11	Business Sponsor	ING Tech
P12	Business User	Development Squad
P13	Data Warehouse Engineer	Data Warehouse Squad
P14	Agile Coach	ING Tech
P15	Junior Researcher	ING Tech

Each survey-question was followed by a question 'Can you please explain the choice made to us?' Subsequently, two questions were asked on top-3 aspects that helped with regard to the topic, and top-3 barriers that hindered with regard to the topic. See the technical report [14] for a detailed overview of the interview questions.

Selection of participants. We identified interviewees from the target population of people that collaborated in the past year in the GAMe-project, either directly in the project itself, or as a business customer of the project within the ING organization. Table 1 gives an overview of interviewed stakeholders, their role, and organizational unit. Each interview lasted 30 to 60 minutes. The interviews were semi-structured and contained the same set of questions for each interviewee. In total, 15 interviews were conducted.

The list of interviewees was narrowed down to 15 by selecting only stakeholders that were personally involved in the GAMe-project, and still working within ING. Interviewees did work in different teams, some were working in the same team. Interviews were conducted orally in person. Both the first and the second author participated in the interviews, alternately one of them fulfilling the role of main interviewer. The first author did know many of the interviewees because he also was involved in the GAMe-project. The second author did not know any of the interviewees. See Table 1 for information on the part of the organization interviewees came from. An overview of all interview questions - including the survey questions - can be found in the technical report [14].

Analysis of the interview results. We computed the standard deviation for each question, based on a 1-5 Likert scale. Subsequently we calculated indicators in order to interpret the results of the survey (see Figure 4):

(1) *Percent Agree* or *Top-2-Box*; the percentage respondents that agreed or strongly agreed.
(2) *Top-Box*: the percentage respondents that strongly agreed.
(3) *Net-Top-2-Box*; the percentage respondents that chose the bottom 2 responses subtracted from percentage respondents that chose the top 2 responses.

(4) *Coefficient of Variation* (CV); also known as relative standard deviation; the standard deviation divided by the mean. Higher CV-values indicate higher variability.

Where the first three are measures of the central tendency, CV is a measure of variability; we used it in addition to the other approaches. In order to examine whether the free format text resulting from the survey confirmed observations from the survey analysis, we coded the free text from the interviews. We used a transcription service to transcribe the audio, then coded the interviews using the R-package RQDA [12]. Coding was set up by the first author, who also was involved in the interviews, and subsequently checked by the second author who performed as interviewer too.

Limitations regarding the interview and survey design. We realize that because three of the authors - the ones working at ING - were involved in the GAMe-project itself, some bias might be introduced. We have tried to prevent this as much as possible by performing the interviews with two interviewers, in this case the first and the second author of this paper. Contrary to the first author, the second author was not involved in the GAMe-project in any way, and therefore did not have any prior knowledge of the project itself.

To overcome weakness or intrinsic biases due to problems that come from single method, single-observer and single-theory studies, we applied triangulation by combining multiple interviewers and performing the coding and analysis process with multiple authors. Also, we tried to avoid affecting the design of the interview and the coding of responses by including independent authors in this process too.

A remark is in place regarding the fact that not every interviewee was knowledgeable about every topic. As a consequence, some of the interview data may be based on partial or less relevant knowledge. We tried to mitigate this by emphasizing towards interviewees that not all topics needed to be answered in the interview. Answering "don't know" was a valid option in the survey questions.

4 RESEARCH RESULTS

In this section, we report results on the interview topics and the artifacts delivered in the GAMe-project.

4.1 Inventory of the GAMe-projects' Artifacts

In the following subsections, we provide an overview of the artifacts delivered in the GAMe-project.

Data collection and data cleaning. From the start of the GAMe-project data from the source systems was recorded as structured metrics in a repository. The GAMe-project used a pragmatic approach to determine what metrics were in scope. In line with the Principles for Software Analytics as mentioned by Menzies and Zimmermann[21] we opted for a practical approach, and to 'live with the data we have'.

Based on the availability of data a series of queries was built, to transform data emerging in the continuous delivery pipeline into structured metrics. Metrics were defined and collected from two data sources: ServiceNow, the backlog management tool that was used by the squads, and Nolio. A complete overview of the built queries and associated metrics is included in our previous paper on the GAMe-project [13].

Three metrics were assessed as lagging: (1) *planned stories completion ratio*; the number of planned stories that were completed in a sprint divided by the number of planned stories, (2) *planned points completion ratio*; the number of completed planned story points divided by the number of planned story points, and (3) *cycle-time*; the mean time from first test deployment after last production has been done until the next production deployment for all applications of a squad. The choice for these three lagging metrics was driven by the assumption that they are typically output related and cannot easily be planned upfront.

Beside that a number of so-called leading metrics - usually input oriented and easy to influence metrics; they give a signal before the trend or reversal occurs - was assessed; see our previous paper on the GAMe-project [13] for an overview of all metrics in scope. All metrics were structured in a dimensional model - the so-called *I4C data warehouse* - and related to conformed dimensions so metrics are comparable within date/time and organization structure.

Leading Lagging Matrix to identify strong metrics. As described in detail in our previous paper [13], descriptive statistics were examined based on the subset of GAMe project metrics. To understand relationships between metrics, and to identify strong metrics - metrics with strong predictive power - both linear regression and pairwise correlation were performed. For visualization purposes a correlation matrix - the so-called *Leading Lagging Matrix* - was prepared that plots positive and negative correlations between all individual metrics. Besides the set of lagging metrics a reference set of five other metrics was plotted on the x-axis of the matrix, although these were not assumed to be lagging. A figure depicting the Leading Lagging Matrix is not included in this paper, but can be found in our previous paper [13] and in the technical report [14]. The analysis led to three implications for squads:

(1) Squads can improve *planned stories completion ratio* and reduce *cycle-time* by slicing deliverables into smaller user stories.
(2) Squads can reduce *cycle-time* by keeping open space in their sprint planning (e.g. increasing remaining time ratio).
(3) Squads can increase *planned stories completion ratio* (delivery predictability) by reducing unexpected unplanned work, for example by improving the quality of systems to reduce incidents that lead to last-minute bug fixes.

Information Visualization. Based on these implications a dashboard was developed. The GAMe-dashboard consists of a series of visualizations that focus on a specific squad; see for example the *Squad Onepager* in Figure 1. The *Squad Onepager* gives squads a summary-view on the status of the most important squad metrics. In the example *sprint completion ratio* of the last sprints, *average cycle time* of the last sprints, and *average cycle time kanban* of the last weeks are depicted.

Another example of a visualization that has been developed within the scope of the GAMe-dashboard is a graph of the *number of squad members in sprint points breakdown*, as depicted in Figure 2. In the graph, which splits the number of completed points into three groups, the number of squad members has also been added

Figure 1: Example of a Squad Onepager on the GAMe-dashboard.

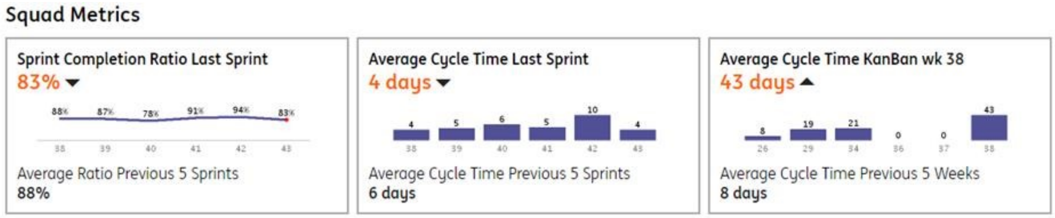


Figure 2: Example of the Number of Squad Members in Sprint Points Breakdown.

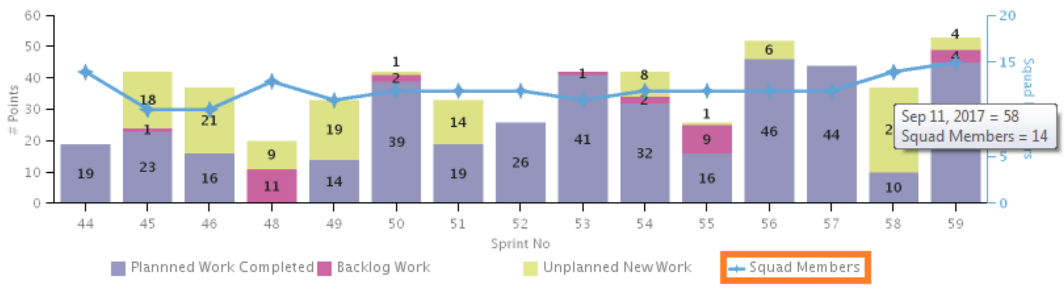
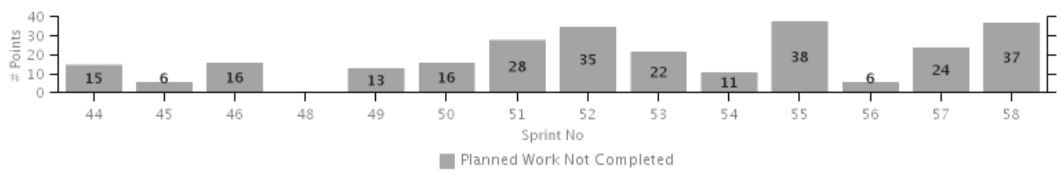


Figure 3: Example of the *planned points not completed.*



during the start of the sprint. The *planned points not completed* are shown in a separate chart, see an example in Figure 3.

IBM Cognos Analytics as BI tool. The GAMe-dashboard has been built in the business intelligence (BI) tool IBM Cognos Analytics. Users need to login with a dedicated userid and password to make use of the dashboard, even when they are already logged in the ING work environment. To monitor the use of the GAMe-dashboard a very limited set of usage metrics were recorded. Some aggregated numbers are that during a period of four months following the implementation of the dashboard 237 unique users logged in at it at least once, and in total 660 times. However, these figures do not provide a clear insight into the extent to which these users actually made use of the information on the dashboard.

4.2 Interview Results

In the following subsections we provide an inventory of results from the interviews with GAMe stakeholders, grouped by the five topics from the interview questions framework. We have anonymized parts of quotes to maintain interviewees' privacy. When quoting survey respondents, we refer to the individual contributor using a *[Pxx]* notation, where xx is the stakeholder's ID (see Table 1).

No consensus on purpose and aims of the GAMe-project. Each interview started with four questions regarding the purpose and the aims of the GAMe-project. A first observation is that there is no real consensus among the interviewees about the aims and goals of the GAMe-project. The first interview question deals with the purpose and aims of the project, and whether these were completely clear to its stakeholders. The respondents were split: as many interviewees agreed that the aims were clear as there were who disagreed (see Figure 4). A similar outcome - with even larger differences between the interviewees - is found in the answers to the interview question whether the project achieved its aims and goals. The outcomes of the two interview questions on top-3 aspects that helped and hindered the project in achieving its goals are summarized in Table 2, where the number of interviewees mentioning an aspect is included between brackets.

Management attention is important. Management attention is mentioned by many interviewees as an important help for achieving the aims of the GAMe-project; *"Management attention, for sure. We now have a weekly meeting with the executives, where we focus [...] on improve the lead time of epics" [P03].* This statement did refer to management commitment as such, and also to an increasing interest in the outcomes of the project: *"I think that what also helped, and this was a little bit later in the project to the end, that there were*

Figure 4: Overview of the outcomes of the survey questions within the interviews.

Interview Question	Likert Distribution	Number of Respondents	Percent Agree	Top-Box	Net-Top-2-Box	Coefficient of Variance
Q1.1: The purpose and aims of the GAMe-project were completely clear to its stakeholders.		12	50%	0%	8%	31%
Q1.2: The GAMe-project did achieve its aims and goals.		12	42%	8%	-8%	41%
Q2.1: The data that we used within the GAMe-project was of good quality.		12	50%	17%	42%	30%
Q2.2: Getting the data that we needed for the GAMe-project was easy.		13	15%	8%	-31%	45%
Q2.3: Preparing the data for further use within the GAMe-project (e.g. combining data from different sources, shaping of the data) was easy.		10	30%	0%	-10%	41%
Q3.1: When analyzing the data within the GAMe-project, scale (e.g. size of the data) did not cause problems.		7	100%	43%	100%	11%
Q3.2: When analyzing the data within the GAMe-project, machine learning (e.g. building predictive models) did not cause problems.		4	75%	0%	50%	40%
Q4.1: The GAMe-dashboard contains the right metrics for the squads to steer on.		7	86%	14%	86%	13%
Q4.2: The GAMe-dashboard is useful for our squads.		11	73%	9%	55%	30%
Q5.1: Performing research on the analytics behind the software delivery processes will help solution delivery teams.		14	100%	57%	100%	11%
Q5.2: Collaboration with (technical) universities will help to improve research activities.		13	100%	38%	100%	11%

Column 'Likert Distribution' shows a graph of the distribution on a 1-5 point Likert scale for each question with from left to right the values 'Strongly Agree', 'Agree', 'Neutral', 'Disagree' and 'Strongly Disagree'.

Table 2: Codes related to Aims and Goals

Helped to achieve goals
Management attention (8)
Different stakeholders involved (multi-disciplinary) (5)
Focus on cycle time reduction (5)
Frequent (weekly) team meetings (3)
Hindered to achieve goals
Lack of time and priorities (5)
Customers do not see added value (5)
Project was not a joint effort (3)
Focus on performance instead on innovation (3)
Squads work in different ways (2)
No trust in the project approach (2)

some senior managers quite interested" [P08]. Again, also on this topic interviewees do not all agree. Many interviewees praise the management commitment as an important help. A minority of the interviewees, however, call management attention as an obstacle.

Regarding management attention the interviews revealed that managers sometimes think differently about the approach toward performance improvement and innovation. Where some believe in a disruptive approach, others opt for small steps in the change process, although this usually takes a long duration: *"We agree on many things, but not all. One of these things is how you do change. I like to create a slice and work with a squad that is interested in solving this small problem" [P11].*

Lack of time and priorities. This latter statement matches an often mentioned obstacle that hindered achieving the aims of the GAMe-project; a lack of time and priorities. As P02 states it: *"What I noticed, at least with the product owners, was that they were busy to perform, and had no room to innovate" [P02].* An aspect that relates to this is a focus on performance instead on innovation, which was mentioned by a number of interviewees.

Many different stakeholders involved. Another mentioned aspect that helped to achieve the goals of the GAMe-project, was the fact that different stakeholders were involved. The multi-disciplinary character of the project was praised by many: *"To be confronted with other ideas, which can help with a better result. That is important and should be applied at every squad" [P02].* Furthermore, a number of interviewees mentioned the clear focus of the GAMe-project on cycle time reduction as helpful.

Yet, at the same time not all stakeholders were convinced that the project really did change things in the squads: *"I did not see the impact. It did not feel as the set of metrics that we should use. We maybe have done this too much in a waterfall kind of way" [P11].* The project was not seen as a joint effort by some: *"It was a group of consultants looking from a distance, saying how it works" [P01].* Some emphasized the fact that squads do not all work in the same way, and even that some stakeholders did not have too much trust in the approach that was chosen for the GAMe-project.

No consensus on getting and preparing the data. The second topic in each interview focused on the aspects of getting the data. The first question dealt with whether the data that was used within the project was of good quality. A vast majority of stakeholders agreed with this or was at least neutral. A second question was about how easy it was to get the data for the project. Consensus among the interviewees on this statement was very low, with an emphasis on stakeholders not agreeing.

The third question dealt with how easy it was to prepare the data for further use within the GAMe-project (e.g. combining data from different sources, shaping of the data). Again, consensus between interviewees is low - indicated by a *Coefficient of Variance* score of 41%, with an emphasis on stakeholders not agreeing. The outcomes of the two interview questions on aspects that helped or hindered with regard to getting and preparing the data are summarized in Table 3.

Table 3: Codes related to Getting the Data

Helped getting and preparing the data
I4C Data warehouse as a solution (9)
ServiceNow data was of good quality (7)
Hindered getting and preparing the data
Difficulties with availability of data (11)
Lack of standardization (10)

I4C data warehouse was appreciated. Many stakeholders mentioned the availability of the I4C data warehouse solution as a big pro for the GAMe-project. However, the data warehouse had two sides; it both helped structuring the data, but at the same time it created a backlog of queries to be developed that sometimes slowed down the project a bit. Overall as the strengths of the I4C data warehouse was mentioned that it is a future-proof solution in which the data is recorded with structured metrics that are ready for further use, with a frequent automated feed.

Good data quality, but not for all parts of the project. Asked about the quality of the data, many interviewees replied that the ServiceNow data was of good quality. Regarding the other data - in particular the data from CDaaS - many interviewees mentioned difficulties in getting the data, sometimes even a kind of silo-behavior in the teams to cooperate with the GAMe-project: *"Getting the data out of the systems... this is really a barrier [...] It is an organizational and technical thing. Sometimes it is technical, because people say 'you cannot have my data, because if you do your queries then the systems will break'. Or otherwise 'no, it is my data, you cannot have it', we have seen that also" [P05].*

Lack of standardization in the CD-tools. Lack of standardization of the data in the different tools within the continuous delivery pipeline is mentioned by many interviewees as a cause for problems in the squads: *"Standardized tooling is a big thing. They want to standardize tooling, but that is a long way. That is why we have a problem at this thing. Everybody is testing in its own way. With their own tooling and that kind of things" [P06].* It did slow down the GAMe-project, and even was mentioned as a cause for not achieving its goals in the end: *"For some parts of the goals of the Game Project the data was very good. Mainly the backlog management side, the Incident Management side. But for the software delivery, following the code, it is very bad quality of data" [P09].*

Scale was not an issue. The first interview question within this topic dealt with whether analyzing the data within the project scale (e.g. size of the data) caused problems. Although the number of interviewees that answered this question was quite low, all agreed that scale did not cause problems.

R was a help, but also a small obstacle. The second question, about whether analyzing the data and machine learning (e.g. building predictive models) caused problems, was answered by even less interviewees. Most of them agreed on the statement by referring to the open source tool R that was used for statistics. However, one disagreed: *"The difficulty may be that there were sometimes unexpected results out of the data analysis. And then we miss the statistical knowledge to understand why" [P06].* Apparently R was experienced as a good tool for statistics, but at the same time experienced as a slight bottleneck due to its somewhat steep learning curve and the fact that not enough statistical knowledge was available in the team.

The results of the two interview questions on aspects that helped and hindered the project with regard to analyzing the data are summarized in Table 4.

Table 4: Codes related to Analyzing the Data

Helped analyzing the data
Use of R for analyzing (5)
Collaboration with academia (1)
Hindered analyzing the data
Use of R for analyzing (2)
Lack of statistical knowledge (2)

For data that was not available, some stakeholders assume that analyzing unstructured data could help: *"I think for example that in all sorts of unstructured descriptions of user stories, it must be possible to find a structure in this with machine learning" [P12].* Stakeholders do believe in prediction techniques: *"With machine learning we can get very far I think" [P14].*

The dashboard contained the right metrics. The fourth topic in each interview focused on the more or less specific aspects of information visualization. The first question within this topic was about whether the GAMe-dashboard contains the right metrics for the software delivery team(s) to steer on. Although not all interviewees did answer this question, most of the stakeholders agreed with this or at least were neutral. The second question is about action-ability and usefulness of the dashboard for software delivery team(s). Despite a relatively large variance, a majority of stakeholders agrees with this statement.

The outcomes of the two interview questions on aspects that helped or hindered the project with regard to building dashboards are summarized in Table 5.

Users like the simple set-up of the dashboard. Interviewees mentioned as a help regarding the visualizations, that the GAMe-dashboard was set-up quite simple, including a very limited set of metrics: *"It is basic and that is actually what I like. The simple, basic set that is there now, yes, it can help teams a lot" [P08].* They also emphasize the setup of the dashboard as a subset with different goals: *"There is not one GAMe-dashboard. We have a dashboard on squad level, on tribe level and on domain level. But the dashboard on*

Table 5: Codes related to Building Dashboards

Helped building dashboards
Dashboard contains only a limited number of metrics (5)
Infrastructure for building dashboards (3)
Dashboard helps, but users need to be convinced (3)
Agile Way of Working (1)
Hindered building dashboards
Dashboard is not used by the squads (7)
Dashboard is not user friendly (6)
Unclear goals of the dashboard (4)
People's opinions are in the way (3)
Accessibility of dashboard is too low (2)

squad level, yes it is widely used. And we are getting feedback on it" [P09].

The GAMe-dashboard is not used a lot. Yet, at the same time stakeholders - especially those from the squads themselves - think that the dashboard is not really used by squads: *"Nobody is using it" [P03].* As a reason for this they mention the fact that the dashboard is not user friendly: *"Developers want to develop. And if you create reports in a specialized application where you have to log-in and you have to find a report that runs once in a period, they are not going to do that" [P01].* Some argue whether Cognos BI is the right tool for visualizations: *"The tool itself is quite limited in the visualization. So, if you want to make it more advanced, you better make your graphs in Excel, because then they become better. That is a bit of a shame for the tooling" [P08].*

Some interviewees propose the idea to include visualizations in the delivery pipeline itself. For example, visualizations incorporated in ServiceNow: *"Pop-ups are put-off by developers as quickly as possible, that's no solution. But looking at your email once in the hour for five minutes, or including visualizations in ServiceNow as part of the daily stand-up would be a good solution" [P12].*

People's opinions were sometimes blocking. Some interviewees mentioned that people's opinions were sometimes blocking progress regarding the use of dashboards: *"There are also squads, and even agile coaches, who really do not believe in generic metrics because every squad has to define their own" [P09].* Or they just don't believe in steering on metrics: *"People saying "I do not believe in steering on metrics, because we, as a team, need to evolve and cooperate and discuss, more the softer approach to coaching a team" [P06].*

Empirical approach was appreciated. The final topic in each interview focused on the aspects of research collaboration. A first question within this topic dealt with the perception whether performing research on the analytics behind the software delivery processes of ING helps solution delivery teams. Not as a surprise, all interviewees agreed or strongly agreed with this. Yet, it is interesting to see why they did so.

Many interviewees argued that due to research the way of working of squads can be better understood. As P08 stated: *"The scientific part was really one of the real gains" [P08].* More specifically, by using real data the performance of squads could be explained: *"I really believe in looking at data in this way" [P06].*

Collaboration with academia is liked by many. Also the second question in this topic whether collaboration with (technical) universities helped ING to improve its research activities was agreed upon by all interviewees, although more stakeholders than the former question agreed instead of strongly agreed. As one underlying reason a fresh look at innovation was mentioned: *"Fresh insights from someone who looks at it with a new, fresh look" [P02].* Another reason - mentioned by many interviewees - was that a scientific approach can help ING: *"A lot of help came from Delft University of Technology, on the statistical analysis, R Studio, what packages to use" [P06].* However, a warning was mentioned too for over-complicated analysis approaches: *"I think that a challenge at the same time is to make it really down to earth as well" [P08].*

The results of the two interview questions about aspects helping or hindering ING to improve research on its software delivery processes are summarized in Table 6.

Table 6: Codes related to Research Collaboration

Helped to improve research
Understand the way of working of squads (8)
A scientific (evidence-based) approach (7)
Expectation that universities are ahead (6)
Real data to explain performance (6)
Hindered to improve research
Research did not solve the problem (4)
Outcomes were not discussed with the squads (3)
Focus on risk and security of a bank (2)
Adoption of scientific approach might be difficult (2)
Too early drawing conclusions (1)

Sharing outcomes of research is important. Apart from the aforementioned positive aspects, a number of issues were also mentioned that hindered the research, as done within the GAMe-project. Some interviewees - all from software delivery squads - mentioned the fact that the outcomes of the GAMe-project were not shared with the squads at the end of the project: *"I am wondering if they ever asked feedback on the reports within teams" [P01].* Others mentioned - regardless of their agreement on the first two questions in this topic - that research in the end did not help: *"It is nice that an article was written about it. But for the short term, it did not solve a problem" [P03].*

Compliance, security and risk are challenges. Furthermore, the focus of a bank on risks and security might have caused challenges for the project: *"A CDaaS pipeline full of security findings and risk related things, a CDaaS pipeline that is not built for reporting" [P01].* A stakeholder mentioned an risk related to research versus the delivery squads: *"You now see a difference occurring between research and development within ING. For example, I see little from the research department going to production. Actually, there is a gap there, and that is not desirable either" [P12].*

5 DISCUSSION

In this section we discuss the outcomes of our study, and we examine implications for industry, and threats to validity.

5.1 How to improve software analytics projects

Cherish management attention. The inventory of aspects that helped and hindered the GAMe-project, as addressed in the second research question - *RQ2: What can be done to improve future implementations of software analytics solutions in such environments?* - indicates that a thorough preparation of a software analytics project is an important precondition for similar projects performed in future. We argue that, although management attention was praised by many during the interviews - one thing to cherish for future projects -, a lack of steering and unanimity on the topic of software analytics might cause barriers during the course of a project.

Software analytics is typically research oriented, and thus goals maybe should be better defined in a different - hypothesis - kind of way. Zhang et al. [28] come up with some advice for software analytics activities - "create feedback loops early, with many iterations" - that might be very suitable for ING purposes too. Operations-driven goals such as 'cycle-time shortening' or 'quality improvement' might fit better in such an approach, leaving enough space for discovering new horizons with great impact.

Consider the approach on data collection and storage. We assume that a question that was raised by some of the interviewees - whether it would be a better approach to collect and store unstructured data and perform machine learning on these to look for structure and information - offers challenging yet very interesting horizons for future research.

Aim for Standardization of CD-tools and data solutions. The availability of a standardized solution within the company to set up the I4C data warehouse, in combination with good quality of data derived from the backlog management tool, was experienced by many stakeholders as a great help. The choice that was made early in the project to build queries to create metrics in a dedicated repository can easily be explained. On the other hand, data from other sources - especially CD-tools - were difficult or impossible to obtain, and were overall of a low quality. Linking data across the boundaries of different tools was therefore difficult and sometimes even impossible. Make decisions on standardization of both CD-tools and data collection and storage upfront to prevent additional work afterwards.

Use R for analysis purposes. The practice of the GAMe project showed, that as the project progressed the knowledge of R within the team had increased enormously. We therefore expect that in subsequent projects the backlog in this area will soon be eliminated, and that the benefits of R will outweigh the disadvantages.

Optimize for actionable information. One of the interviewees said it clearly: *"Visualizations need to be in your face" [P08].* Apparently this was not the case in the GAMe-dashboard yet. Especially the fact that users need to log-in into a specific business intelligence tool seems one step too far for squad members. We assume that future research regarding dashboards should be focused on how to include visualizations in the daily practice of squad members, executives, and other stakeholders, and on optimization of insightful and actionable information. Potentially promising ideas came up in the interviews, such as include visualizations in the backlog management tool, that is used by the squads in their daily stand-ups. Furthermore, future research should be focused on how to measure the real impact of visualizations, instead of only looking at number of log-ins in the business intelligence tool.

Comparison of studies. We built our study on earlier work [13] in which we focused specifically on the analysis of a subset metrics. The goal of that study was to identify strong metrics. For this purpose a project - the so-called GAMe-project - was performed. In our initial study [13] we analyzed a subset of 16 metrics from 59 squads at ING. We identified two lagging metrics and assessed four leading metrics to be strong. The results of the initial study were used by ING to built a series of dashboards for squads to steer on.

In the follow-up study that we describe in *this* paper we evaluated the process of implementing this GAMe-project, mainly looking from the perspective of 'how did stakeholders of the project experience the implementation process?' We did not evaluate the artifacts delivered by the GAMe-project as such, but instead we asked stakeholders for their experiences and strategies for improvement of future software analytics projects.

5.2 Implications

The outcomes of our study might not simply be generalized to other environments, within or outside ING. Yet, we identify some take-away-messages that apply to implementing a software analytics solution in a CD-setting:

(1) Companies should think ahead about the aims they want to achieve with software analytics, and then continuously communicate about this to all involved. In view of the investigative nature of software analytics projects - and with them the often vague objectives - we argue that it is preferable to appoint research as an objective in itself.
(2) Companies that set up CD-pipelines, should give attention at an early stage to standardization of data, especially across the boundaries of different systems and tools in the pipeline. This aspect should be high on the agenda of enterprise data architects involved in such activities.
(3) Visualizations should - when applicable - be incorporated in the daily work environment of delivery teams.

A fourth implication that we identified was that companies should use an empirical approach when starting a software analytics project. Collaboration with academia helps. However, continuous attention must be paid to presentation of results and explaining the scientific approach. As a note to this implication we realize that respondents might be somewhat biased by the academic partners being involved and maybe are not entirely neutral about their own role.

5.3 Threats to Validity

We see the following key limitations.

First, our study is conducted in single company. Nevertheless, we argue that the results can be used to draw many lessons, since the company in question, ING, is at the forefront of applying analytics and continuous delivery at scale in an industrial context that has a tradition of being conservative and risk averse from a technological point of view. This makes our results applicable to many other organizations in a similar situation.

Second, our study is based on subjective analysis, and some of the authors were involved in the project under study. We mitigated this by being open about our background, and by involving co-authors who were not involved in the project.

Third, the analytics context and the specific metrics used are specific to the company in question, and their prominence may not be widespread yet. Nevertheless, we think the factors identified in Tables 3-7 are largely independent of the actual setting, and apply to many software analytics contexts.

6 CONCLUSIONS

We studied the outcomes of a software analytics project that was performed during one year within the continuous delivery teams of ING. Within the scope of the project a dataset built from backlog management and continuous delivery data from ING was analyzed, in order to identify strong metrics: metrics with high predictive power towards a subset of lagging variables. Based on this analysis three implications for improvement strategies for squads were identified, and a dashboard was build based on these to help squads to improve their performance. To understand any causes behind the implementation of the project, we interviewed 15 stakeholders about five project related topics. Based on the interviews we identified 16 factors that helped the implementation of software analytics, and 20 factors that hindered the project.

ACKNOWLEDGMENTS

The authors would like to thank ING and all interviewees for giving the confidence to share their experiences with us. This project has received funding from the European Unionś Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 642954.

REFERENCES

[1] B. Adams, S. Bellomo, C. Bird, T. Marshall-Keim, F. Khomh, and K. Moir. 2015. The Practice and Future of Release Engineering: A Roundtable with Three Release Engineers. *IEEE Software* 32, 2 (Mar 2015), 42–49. DOI: http://dx.doi.org/10.1109/MS.2015.52

[2] Amritanshu Agrawal and Tim Menzies. 2017. "Better Data" is Better than "Better Data Miners" (Benefits of Tuning SMOTE for Defect Prediction). *CoRR* abs/1705.03697 (2017). http://arxiv.org/abs/1705.03697

[3] Raymond P.L. Buse and Thomas Zimmermann. 2010. Analytics for software development. In *FoSER '10 Proceedings of the FSE/SDP workshop on Future of software engineering research*. IEEE, 77–80.

[4] Lianping Chen. 2017. Continuous Delivery: Overcoming adoption challenges. *Journal of Systems and Software* 128 (2017), 72 – 86. DOI: http://dx.doi.org/https://doi.org/10.1016/j.jss.2017.02.013

[5] T. Davenport, J. Harris, and R. Morison. 2010. Analytics at Work. *Harvard Business School Publishing Corporation* (2010).

[6] Thomas Diefenbach. 2008. Are case studies more than sophisticated storytelling?: Methodological problems of qualitative empirical research mainly based on semi-structured interviews. *Quality & Quantity* 43, 6 (02 Apr 2008), 875. DOI: http://dx.doi.org/10.1007/s11135-008-9164-0

[7] J. Humble en D. Farley. 2010. *Continuous Delivery, reliable software releases through build, test and deployment automation.* Addison-Wesley.

[8] Daniel Méndez Fernández and Stefan Wagner. 2016. Case Studies in Industry: What We Have Learnt. In *Proceedings of the 4th International Workshop on Conducting Empirical Studies in Industry (CESI)*. IEEE, 25–31.

[9] Danyel Fisher, Rob DeLine, Mary Czerwinski, and Steven Drucker. 2012. Interactions with Big Data Analytics. *interactions* 19, 3 (May 2012), 50–59. DOI: http://dx.doi.org/10.1145/2168931.2168943

[10] Brian Fitzgerald and Klaas-Jan Stol. 2017. Continuous software engineering: A roadmap and agenda. *Journal of Systems and Software* 123 (2017), 176 – 189. DOI: http://dx.doi.org/https://doi.org/10.1016/j.jss.2015.06.063

[11] Tihana Galinac Grbac and Per Runeson. 2016. Plug-in Software Engineering Case Studies. In *Proceedings of the 4th International Workshop on Conducting Empirical Studies in Industry (CESI)*. IEEE, 21–24.

[12] Ronggui HUANG. 2017. *RQDA: R-based Qualitative Data Analysis*. http://rqda.r-forge.r-project.org R package version 0.3-0.

[13] Hennie Huijgens, Robert Lamping, Dick Stevens, Hartger Rothengatter, Georgios Gousios, and Daniele Romano. 2017. Strong agile metrics: mining log data to determine predictive power of software metrics for continuous delivery teams. In *Proceedings of the 11th Joint Meeting on Foundations of Software Engineering (ESEC/FSE)*. IEEE, 866–871.

[14] Hennie Huijgens, Davide Spadini, Niels Visser, Dick Stevens, and Arie van Deursen. 2018. TUD-SERG-2018-002 - Software Analytics in a Continuous Delivery Setting. (2018). http://swerl.tudelft.nl/twiki/pub/Main/TechnicalReports/TUD-SERG-2018-002.pdf

[15] S. Kandel, A. Paepcke, J. M. Hellerstein, and J. Heer. 2012. Enterprise Data Analysis and Visualization: An Interview Study. *IEEE Transactions on Visualization and Computer Graphics* 18, 12 (Dec 2012), 2917–2926. DOI: http://dx.doi.org/10.1109/TVCG.2012.219

[16] M. Kim, T. Zimmermann, R. DeLine, and A. Begel. 2017. Data Scientists in Software Teams: State of the Art and Challenges. *IEEE Transactions on Software Engineering* PP, 99 (2017), 1–1. DOI: http://dx.doi.org/10.1109/TSE.2017.2754374

[17] Henrik Kniberg and Anders Ivarsson. 2012. Scaling Agile @ Spotify with Tribes, Squads, Chapters and Guilds. (2012). https://creativeheldstab.com/wp-content/uploads/2014/09/scaling-agile-spotify-11.pdf

[18] Eero Laukkanen, Juha Itkonen, and Casper Lassenius. 2017. Problems, causes and solutions when adopting continuous delivery—A systematic literature review. *Information and Software Technology* 82 (2017), 55 – 79. DOI: http://dx.doi.org/https://doi.org/10.1016/j.infsof.2016.10.001

[19] Mathieu Lavallée and Pierre N. Robillard. 2015. Planning for the unknown: lessons learned from ten months of non-participant exploratory observations in the industry. In *Proceedings of the 3rd International Workshop on Conducting Empirical Studies in Industry (CESI)*. IEEE, 12–18.

[20] Anna-Liisa Mattila, Timo Lehtonen, Henri Terho, Tommi Mikkonen, and Kari Systä. 2015. Mashing Up Software Issue Management, Development, and Usage Data. In *Proceedings of the Second International Workshop on Rapid Continuous Software Engineering (RCoSE '15)*. IEEE Press, Piscataway, NJ, USA, 26–29. http://dl.acm.org/citation.cfm?id=2820678.2820685

[21] T. Menzies and T. Zimmermann. 2013. Software Analytics: So What? *IEEE Software* 30, 4 (July 2013), 31–37. DOI: http://dx.doi.org/10.1109/MS.2013.86

[22] A. T. Misirli, B. Caglayan, A. Bener, and B. Turhan. 2013. A Retrospective Study of Software Analytics Projects: In-Depth Interviews with Practitioners. *IEEE Software* 30, 5 (Sept 2013), 54–61. DOI: http://dx.doi.org/10.1109/MS.2013.93

[23] Lutz Prechelt, Franz Zieris, and Holger Schmeisky. 2015. Difficulty factors of obtaining access for empirical studies in industry. In *Proceedings of the 3rd International Workshop on Conducting Empirical Studies in Industry (CESI)*. IEEE, 19–25.

[24] Per Runeson and Martin Höst. 2008. Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering* 14, 2 (19 Dec 2008), 131. DOI: http://dx.doi.org/10.1007/s10664-008-9102-8

[25] Emad Shihab, Ahmed E. Hassan, Bram Adams, and Zhen Ming Jiang. 2012. An Industrial Study on the Risk of Software Changes. In *Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering (FSE '12)*. 1–11.

[26] Jan Waller, Nils C. Ehmke, and Wilhelm Hasselbring. 2015. Including Performance Benchmarks into Continuous Integration to Enable DevOps. *SIGSOFT Softw. Eng. Notes* 40, 2 (April 2015), 1–4. DOI: http://dx.doi.org/10.1145/2735399.2735416

[27] Aiko Yamashita. 2015. Integration of SE research and industry: reflections, theories and illustrative example. In *Proceedings of the 2nd International Workshop on Software Engineering Research and Industrial Practice (SER&IP)*. IEEE, 11–17.

[28] D. Zhang, S. Han, Y. Dang, J. G. Lou, H. Zhang, and T. Xie. 2013. Software Analytics in Practice. *IEEE Software* 30, 5 (Sept 2013), 30–37. DOI: http://dx.doi.org/10.1109/MS.2013.94

TECHNICAL REPORT

Interview and Survey Set-up

(1) **Purpose and aims of the GAMe-project**
  (a) To what extent do you agree with the following statement? "The purpose and aims of the GAMe-project were completely clear to its stakeholders"
  strongly disagree - disagree - neutral - agree - strongly agree - don't know
    (i) Follow-up question: Can you please explain the choice made to us?
  (b) To what extent do you agree with the following statement? "The GAMe-project did achieve its aims and goals"
  strongly disagree - disagree - neutral - agree - strongly agree - don't know
    (i) Follow-up question: Can you please explain the choice made to us?
  (c) What top-3 aspects did help the GAMe-project in achieving its goals?
  (d) What top-3 barriers did hinder the GAMe-project in achieving its goals?

(2) **Large-scale Computing: Getting the Data**
  (a) To what extent do you agree with the following statement? "The data that we used within the GAMe-project was of good quality"
  strongly disagree - disagree - neutral - agree - strongly agree - don't know
    (i) Follow-up question: Can you please explain the choice made to us?
  (b) To what extent do you agree with the following statement? "Getting the data that we needed for the GAMe-project was easy" strongly disagree - disagree - neutral - agree - strongly agree - don't know
    (i) Follow-up question: Can you please explain the choice made to us?
  (c) To what extent do you agree with the following statement? "Preparing the data for further use within the GAMe-project (e.g. combining data from different sources, shaping of the data) was easy"
  strongly disagree - disagree - neutral - agree - strongly agree - don't know
    (i) Follow-up question: Can you please explain the choice made to us?
  (d) What top-3 aspects did help the GAMe-project with regard to getting and preparing the data?
  (e) What top-3 barriers did hinder the GAMe-project with regard to getting and preparing the data?

(3) **Analysis Algorithms: Analyzing the Data**
  (a) To what extent do you agree with the following statement? "When analyzing the data within the GAMe-project scale (e.g. size of the data) caused problems"
  strongly disagree - disagree - neutral - agree - strongly agree - don't know
    (i) Follow-up question: Can you please explain the choice made to us?
  (b) To what extent do you agree with the following statement? "When analyzing the data within the GAMe-project machine learning (e.g. building predictive models) caused problems"
  strongly disagree - disagree - neutral - agree - strongly agree - don't know
    (i) Follow-up question: Can you please explain the choice made to us?
  (c) What top-3 aspects did help the GAMe-project with regard to analyzing the data?
  (d) What top-3 barriers did hinder the GAMe-project with regard to analyzing the data?

(4) **Information Visualization**
  (a) To what extent do you agree with the following statement? "The GAMe-dashboard contains the right metrics for the software delivery team(s) to steer on"
  strongly disagree - disagree - neutral - agree - strongly agree - don't know
    (i) Follow-up question: Can you please explain the choice made to us?
  (b) To what extent do you agree with the following statement? "The GAMe-dashboard is useful for our software delivery team(s)"
  strongly disagree - disagree - neutral - agree - strongly agree - don't know
    (i) Follow-up question: Can you please explain the choice made to us?
  (c) What top-3 aspects did help the GAMe-project with regard to building and using dashboards for decision-making?
  (d) What top-3 barriers did hinder the GAMe-project with regard to building and using dashboards for decision-making?

(5) **Research Collaboration**
  (a) To what extent do you agree with the following statement?
  "Performing research on the analytics behind the software delivery processes of ING will help solution delivery teams"
  strongly disagree - disagree - neutral - agree - strongly agree - don't know
    (i) Follow-up question: Can you please explain the choice made to us?
  (b) To what extent do you agree with the following statement? "Collaboration with (technical) universities will help ING to improve its' research activities"
  strongly disagree - disagree - neutral - agree - strongly agree - don't know
    (i) Follow-up question: Can you please explain the choice made to us?
  (c) What top-3 aspects do you expect to help ING to improve research on its' software delivery processes?
  (d) What top-3 barriers do you expect to hinder ING to improve research on its' software delivery processes?

Aggregated Survey Results

Table 7: Aggregated Survey Results

	Q1.1	Q1.2	Q2.1	Q2.2	Q2.3	Q3.1	Q3.2	Q4.1	Q4.2	Q5.1	Q5.2
Count	12	12	12	13	10	7	4	7	11	14	13
Sum	37	35	42	33	27	31	13	28	39	64	57
Mean	3.08	2.92	3.50	2.54	2.70	4.43	3.25	4.00	3.55	4.57	4.38
Median	3.50	2.50	3.50	3.00	3.00	4.00	4.00	4.00	4.00	5.00	4.00
Standard Deviation	0.95	1.19	1.04	1.15	1.10	0.49	1.30	0.53	1.08	0.49	0.49
Percent Agree	50%	42%	50%	15%	30%	100%	75%	86%	73%	100%	100%
Top-2-Box	50%	42%	50%	15%	30%	100%	75%	86%	73%	100%	100%
Top-Box	0%	8%	17%	8%	0%	43%	0%	14%	9%	57%	38%
Net Top Box	0%	0%	8%	-15%	-20%	43%	-25%	14%	0%	57%	38%
Net Top-2-Box	8%	-8%	42%	-31%	-10%	100%	50%	86%	55%	100%	100%
Coefficient of Variation (CV)	31%	41%	30%	45%	41%	11%	40%	13%	30%	11%	11%

TUD-SERG-2018-02
ISSN 1872-5392