FACULTY OF TECHNOLOGY

BUSINESS INTELLIGENCE IN PERFORMANCE MANAGEMENT OF AGILE PROGRAMS

Jyri Kangasharju

INDUSTRIAL ENGINEERING AND MANAGEMENT

Master's thesis

December 2020

ABSTRACT

Business Intelligence in Performance Management of Agile Programs

Jyri Kangasharju

University of Oulu, Industrial Engineering and Management

Master's thesis 2020, 111 pp. + 2 appendixes

Supervisors at the university: Kirsi Aaltonen, Jaakko Kujala

This thesis was conducted for a case company with the primary objective of designing a business intelligence dashboard to be used in the management of agile programs. The aim is to increase the knowledge within the topic and to solve problems that the case company has with the lack of coherent and modern data-driven performance management practices. One of the paramount objectives of this thesis is to reduce the time consumed for manual performance reporting activities and therefore improve the productivity and efficiency within the program management processes and practices.

Design science is used as the primary research method in this study. However, the existing literature is also reviewed to explore what has already been studied within the topic and to identify the best practices to be exploited in designing the dashboard. The current state analysis of the case company is conducted and requirements for the dashboard are identified via a survey and interviews. Based on empirical research, two business intelligence dashboards are designed to cover the identified use cases for the performance management of agile programs.

This thesis contributes to program management and business intelligence research. Especially, agile program management is discussed at a rather practical level and a set of performance metrics to be used in it is proposed. The use of business intelligence is demonstrated through performance management methods. The results of this study can also be used for further research purposes. Therefore, this study makes clear scientific contributions in addition to the evident practical implications.

Keywords: agile software development, business intelligence, performance management, program management

TIIVISTELMÄ

Business Intelligence in Performance Management of Agile Programs

Jyri Kangasharju

Oulun yliopisto, tuotantotalous

Diplomityö 2020, 111 sivua + 2 liitettä

Ohjaajat yliopistolla: Kirsi Aaltonen, Jaakko Kujala

Tämän opinnäytetyön tavoitteena on suunnitella kohdeyritykselle business intelligence-raportointinäkymät hyödynnettäväksi ketterien ohjelmien johtamisessa. Työssä pyritään tarjoamaan ratkaisu kohdeyrityksen ongelmiin ja parantamaan johdonmukaisuutta nykypäiväisten datalähtöisten suorituskyvyn johtamiskäytäntöjen kautta. Yksi tämän opinnäytetyön keskeisimmistä tavoitteista on saada vähennettyä manuaalisiin raportointitoimenpiteisiin kuluvaa aikaa ja siten parantaa ohjelmajohtamisprosessien ja -käytäntöjen tehokkuutta ja tuottavuutta.

Pääasiallisena tutkimusmenetelmänä työssä käytetään Design Science -menetelmää. Aikaisempia tutkimuksia käydään läpi osana kirjallisuuskatsausta, jotta saadaan kartoitettua, mitä aiheen ympärillä on jo aikaisemmin tutkittu. Aikaisempia tutkimustuloksia hyödynnetään myös hyvien käytäntöjen selvittämisessä raportointinäkymän suunnitteluprosessille. Kohdeyrityksen nykytilaa ja sen osoittamia vaatimuksia analysoidaan kyselyn ja haastattelujen kautta. Perustuen empiiriseen tutkimukseen kaksi raportointinäkymää päätettiin suunnitella kattamaan tunnistetut käyttökohteet suorituskyvyn johtamiselle ketterissä ohjelmissa.

Tämä opinnäytetyö edistää aikaisempaa tutkimusta ohjelmajohtamisen ja business intelligencen alalla. Erityisesti tässä työssä käsitellään ketterää ohjelmajohtamista käytännön tasolla ja ehdotetaan suorituskyvyn mittareita siinä hyödynnettäväksi. Business intelligenceä havainnollistetaan erityisesti suorituskyvyn johtamismenetelmien kautta. Työn tuloksia voidaan myös hyödyntää jatkotutkimuksessa, joten työllä on myös selkeää tieteellistä kontribuutiota ilmeisten käytännön johtopäätösten lisäksi.

Avainsanat: ketterä ohjelmistokehitys, business intelligence, suorituskyvyn johtaminen, ohjelmajohtaminen

FOREWORD

This Master's thesis was written between September 2020 and December 2020. This thesis aimed to develop performance management practices within the product development unit of an agile software development organization. The primary objective was to identify requirements for a business intelligence dashboard to improve the current performance management practices of the case company. First and foremost, I would like to thank the case company for this interesting topic and for having trust in me. This year was not the brightest for any of us, but this opportunity meant a lot to me.

Especially, I would like to thank my instructor and supervisor Lazo for this opportunity and all the support during the process. Also, thank you Tanja and Toni for sparring and providing valuable insights into the topic. Not to forget everyone else from the case company who participated in the interviews and the research process. Also, I express my gratitude to my supervisors from the University of Oulu, Kirsi Aaltonen and Jaakko Kujala, for all the feedback and guidance. It was a pleasure to work with you and to learn how research should be conducted.

Finally, I want to thank my friends who have taken this journey with me and my family for all the support during my studies. These years have been full of moments that we can all look back with joy. I would not have made it to the finish line without you.

Oulu, 11.12.2020

Jyri Kangasharju
Jyri Kangasharju

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

ADO	Azure DevOps
ART	agile release train
BI	business intelligence
BIS	business intelligence system
BSC	balanced scorecard
DSRM	design science research methodology
ERP	enterprise resource planning
FTE	full-time equivalent
GQM	goal-question-metric
KPI	key performance indicator
SAFe	scaled agile framework
PI	product increment
PPM	project portfolio management
RTM	release train manager
STM	solution train manager

Agile

Approach referring to the use of lightweight methods for managing the changing project requirements. Introduces new concepts and practices to be used in the management of complex projects.

Business Intelligence

Managerial philosophy and a tool that utilizes data to help make business decisions. Typically contains the means to transform data into knowledge and to enable data-driven decision-making.

Performance Management

Process of creating the context for performance measurement. Covers the methods, metrics, processes, and tools that are used for managing performance.

Project

A temporary process to create a unique product, service, or result within a predefined start and end date.

Program

An entity with a determined purpose and predefined expectations to the benefits. Is typically used as a framework for organizing the effort of managing multiple interrelated projects.

Scaled Agile Framework

A set of organization and workflow patterns for implementing agile practices at enterprise scale. Organized around team, program, and portfolio levels.

1 INTRODUCTION

An increasing number of organizations arrange their business through projects and the trend is only increasing during the upcoming years (Scoper et al. 2018) However, on average just about a third of IT projects meet user needs and are completed based on the predefined requirements (Bragen 2018). This is a rather noteworthy issue that many organizations today are facing, and it requires new solutions to be used in the management of project-based organizations. That is also what this research contributes towards, for instance, one possible remedy for this problem is to take advantage of the organizations' data to improve the performance management practices.

Data is often considered a major trendsetter in today's business environment. However, data as such do not deliver any value, but once transformed into information and put into context, new know-how is created (Laursen & Thorlund 2016). This is where business intelligence (BI) proves beneficial as it can be used to automatize this knowledge creation process (Negash & Gray 2008). Those organizations that understand the importance of data utilization and have the abilities for it, possess great potential in their hands.

This thesis aims at giving guidelines and thoughts for project-based organizations to take advantage of their internal data. By doing so, organizations can generate valuable insights to improve their operations and to make their daily life a little bit more efficient and comfortable. Utilization of capabilities brought by analytics in project business can also lead to significant organizational benefits such as improving the accuracy of decisions and reducing the time used for decision making (Wang et al. 2018). Considering project-based organizations, this thesis aims to improve the success rate of development projects by better acknowledging the factors influencing the project deliveries.

BI is beneficial also in the program management context. Program management is used for the management of multiple interrelated projects and it brings value through operations (Thiry 2010). BI can be integrated with the operations to provide insights into the underlying trends (Negash & Gray 2008). The concept of agile calls for the acknowledgment of new factors within the traditional program management methodologies. Therefore, agile program management takes agile concepts to the traditional program management process and creates all-new entities with unique practices and metrics to be used within business intelligence.

1.1 Background of the study

Organizations increasingly use projects to achieve their business objectives most economically. Schoper et al. (2018) identified that roughly one-third of all economic activities are carried out in the form of projects and the trend is only growing in the future. This increases the number of projects in the organizations and creates the need for program and portfolio management to manage the multitude of simultaneous projects ongoing in an organization. Recently, agile methodologies have become more popular in project management and they are commonly used as an alternative to the traditional plan-based and linear way of executing projects (Dingsøyr et al. 2010)

The case company of this study is a large global software company that has arranged its product development with the Scaled Agile Framework (SAFe). Briefly mentioned, SAFe is a scaled-up model designed for incorporating agile methods in an enterprise context (Nuottila et al. 2016). The SAFe model contains team, program, and portfolio dimensions (Turetken et al. 2017). In this study, the program dimension is under the highest level of attention. However, to create reliable conclusions related to it, understanding the other levels is also required. Therefore, the program level can't be entirely isolated from the other levels since it is highly dependent on them.

The goal of this study is to design a business intelligence dashboard to be used for the management purposes of product development programs. Business intelligence can be defined as the use of data to help make business decisions and it can be used for the automatized reporting of the defined performance metrics and key performance indicators (Eckerson 2011). Dashboards provide value to the stakeholders by displaying critical information in real-time. In this study, a high level of attention is paid especially to the use cases in the field of agile program management. The aim is to identify generally recognized good practices and metrics to be utilized as the foundation for the business intelligence dashboard. The literature is reviewed to identify these practices and they are contrasted to the context of the case company. The need for this research became apparent as there is rather limited knowledge related to the topic and more scientific and practical knowledge within the theme was required.

1.2 Research problem and objectives

This research aims at increasing knowledge within the agile program management topic by discussing the performance management practices in the context. Especially, the aim is to identify performance metrics to be used in the performance measurement of agile programs and to improve the performance management practices of the case company. A business intelligence dashboard is designed for the case company to replace some of the existing manually compiled reports. The motivation for this is to improve the accuracy of the decision-making process and to reduce the time spent on manual work. Design of the dashboard requires the identification of the current practices of the case company but also of the best practices identified within the existing literature. For this purpose, design science was chosen as the main research method of this study.

The designed BI dashboard should solve the problems that the case company has with the lack of coherent and modern data-driven performance reporting practices. The biggest problem has been that it takes unnecessary time to create and update the reports manually for every reporting need. Also, the existing reporting solutions haven't been able to provide capabilities to create reports based on a long enough period of historical data. The dashboard should provide improvement to these problems by replacing some of the existing manually created reports and offer better capabilities of creating forecasts based on the data.

This research contributes to the existing literature by complementing project management research with business intelligence research. This research perspective has been rather limited in past studies. Especially, this research concentrates on the agile program management concept that has been a popular research topic from the software development point of view and at the project level (e.g. Anderson 2004, Chin 2004, Dingsøyr et al. 2010, Sheffield & Lémateyr 2013, Conforto et al. 2014), but there is very little research overall on the agile program management topic. Also, the existing research conducted from the data utilization perspective of project business is limited. This study contributes to this deficit by introducing the agile program management concept and discussing performance management practices such as performance metrics to be used within the area.

The research questions of this study have been defined to solve the research problems. They are to be answered through the literature review, empirical research, and the synergy between them. In particular, the literature review enables the design process used in empirical research. The research questions set for this study are the following:

RQ1: How can business intelligence be used in agile program management?

The first research question is answered solely based on the literature review and it creates a theoretical framework for this study. The literature is reviewed to identify program management practices and metrics to be used in the design process. Also, concepts related to program management especially in the agile context are being analyzed. One especially interesting point of view is how the performance reporting process should be managed and what are the contextual characteristics that the research environment puts on the table. Overall, the aim is to identify good practices for business intelligence utilization in the program management domain.

RQ2: Which metrics should be used to measure the performance of agile programs according to existing literature and empirical findings?

The second research question aims at discovering metrics to be utilized in performance management in an agile environment and especially within the program level. The findings from the literature review are then compared with the current practices of the case company. These practices are identified in chapter 4 and the comparison is made in chapter 5. In conclusion, a list of metrics to be used in the performance management of agile programs is formed. The performance metrics are then utilized as an input to the design process of the dashboard.

RQ3: How should business intelligence be utilized in developing the performance management practices of the case company?

The answer to the third question is provided based on empirical analysis. First, the current state of the case company is assessed through a survey to analyze the current practices, and especially challenges and improvement areas within the current working methods and processes. The empirical research is supplemented with interviews to identify requirements for the business intelligence dashboard. As the primary outcome of this thesis, the dashboard is designed and a general framework for business intelligence usage

in the agile programs is formulated. Also, the next steps for developing and implementing the dashboard are discussed.

1.3 Research scope

The scope of this study is within the systems development type of programs. According to Turner's (1999) classification, systems development typically has well-defined work methods, but the goals may be poorly defined. This means that it is often difficult to define the requirements for information system development since it is common that the needs of the customer change during the lifecycle. This is especially true when developing completely new solutions. Common for those types of deliveries is that the work methods are well defined, but the changing customer requirements may make it difficult to accurately define the requirements for the development.

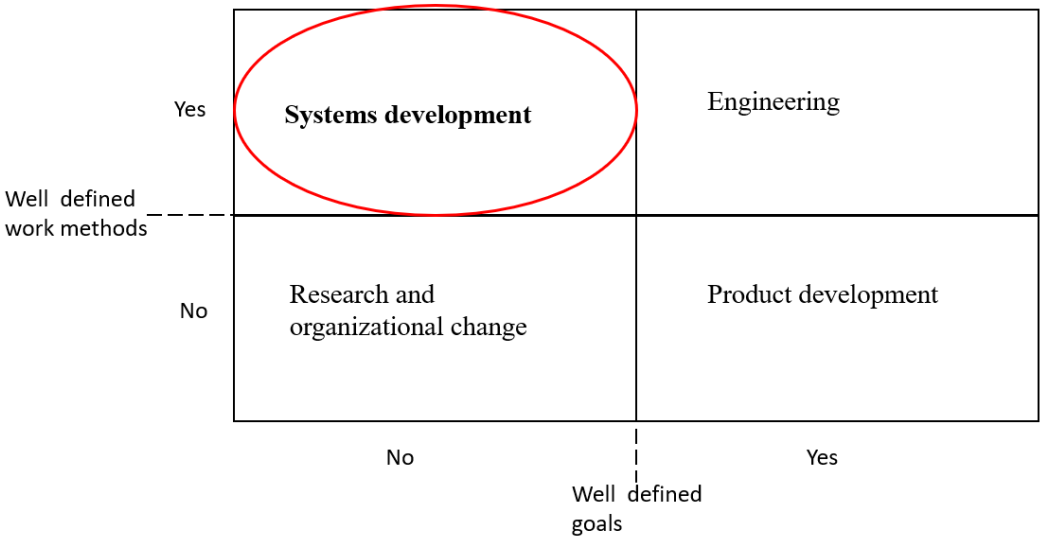

Figure 1: The goals and methods matrix (adapted from Turner 1999)

This classification is compatible with the scope of this study since the case company of this study operates in the software industry and the main products of the company are information systems that are developed based on the customer needs. These types of entities are often referred to as projects in literature, but since this research is done as a case study within an agile organization, programs are used as the primary construct. Also, program management is commonly identified as a research area within the project management research stream. Therefore, many theories related to project management are highly applicable also for the program entities and therefore within the scope of this study.

1.4 Structure of the thesis

This thesis is divided into chapters. In chapter 2, the theoretical concepts related to the most important themes are discussed. The motivation for this is to create a theoretical framework for this thesis that enables the reader to understand the central ideologies, concepts, and theories beneath the topic. The purpose of the literature review is to create an understanding of the research environment of this study as well as to identify practices and concepts that can be used in the empirical part of this research. The literature review also aims at deriving metrics and key performance indicators that can be utilized in the practical part of this study.

Chapter 3 gives an introduction to the research methods used in this study by introducing a model for conducting the design science research based on the literature. The model introduced in the research is adapted to this study. The fourth chapter introduces the case company of this study. The main idea in the chapter is to describe the background of the company including its' organizational structure, operating model, and current practices. Also, the results from the questionnaire and interviews are discussed.

In the fifth chapter, the dashboard is designed and the concepts related to it are discussed. Also, the recommendations are given for the case company. The last chapter wraps up the study by discussing the answers to the research questions, the limitations of this study, and gives suggestions for future research within the area.

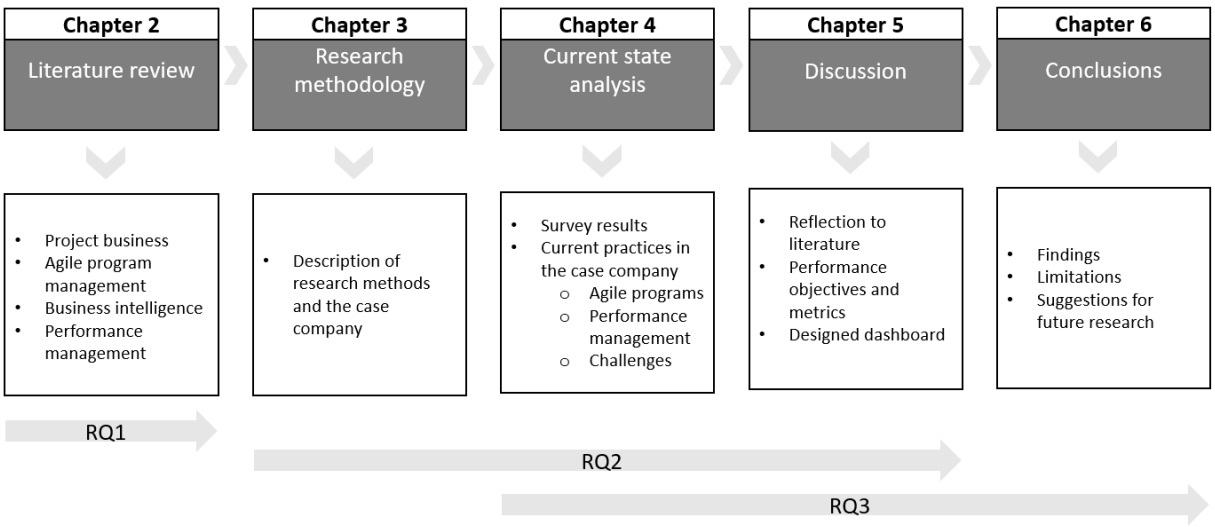

Figure 2: Execution of the study

2 LITERATURE REVIEW

In this chapter, the theories related to the topic of this thesis are discussed. The first subchapter discusses the traditional view on project business. The motivation for this is to give an overview of the environment where projects are the primary business driver. In this subchapter, the three levels of project business are discussed: project, program, and portfolio. Especially literature and concepts related to program management are given a high level of attention due to the scope of this study.

In the second subchapter, the concepts of agile and agile programs are discussed. The key elements and principles of the agile methods are scrutinized. Especially the common processes and practices to be used within an agile environment are identified. Considering the environment of this research, one important concept is the scaled agile framework. Its basic building blocks are discussed to give an introduction to the environment of this study.

Business intelligence (BI) is commonly used for performance management purposes. These two topics are discussed starting from performance management and then going deeper into the BI topic and the practical implementation of this study. Performance management is discussed primarily in the context of project business. Especially, metrics and KPIs are highly emphasized and the practices for selecting and using them are highlighted. Also, performance measurement in an agile environment is discussed and a list of agile program management metrics is created by contrasting traditional project metrics to the agile context. This is the most important part of this section and the metrics are brought as an input to the empirical part of this study.

The final subchapter demonstrates the business intelligence (BI) concept. The section aims to gain a basic understanding of what BI is and what are the requirements for the design and development of BI solutions. Theories related to this comprise a discussion about the BI systems in general and the common process of implementing and maintaining a BI solution. Also, the lifecycle of a BI solution is introduced and the BI dashboards are discussed.

2.1 Traditional view on project business

Projects were initially used in manufacturing for seeking improved coordination. However, in today's project management literature, they are brought to a more strategic context where multiple projects contribute to the fulfillment of strategic objectives in the entire business system. (Artto & Wikström 2005) Also, today an increasing number of organizations arrange their business through projects, and it is expected that the trend is only growing during upcoming years (Scoper et al. 2018). However, on average only about a third of IT projects meet user needs and are completed based on these predefined requirements concerning the cost, time, and quality (Bragen 2018). This is a rather alarming dilemma highlighting the importance of systematic project management.

Artto and Wikström (2005) defined project business as "the part of a business that relates directly or indirectly to projects, to achieve the objectives of a firm or several firms." Artto and Kujala (2008) expanded this idea to address single or several projects. From the perspective of this study, especially the context of one firm, with many projects is the most interesting (Figure 3). This stands for the management of a project-based firm, which is an area that addresses the managerial issues of a firm that conducts a specific part of its activities in a project form. These activities may involve projects of two types: external production or customer delivery type projects, and internal development or capital investment projects. (Artto & Kujala 2008)

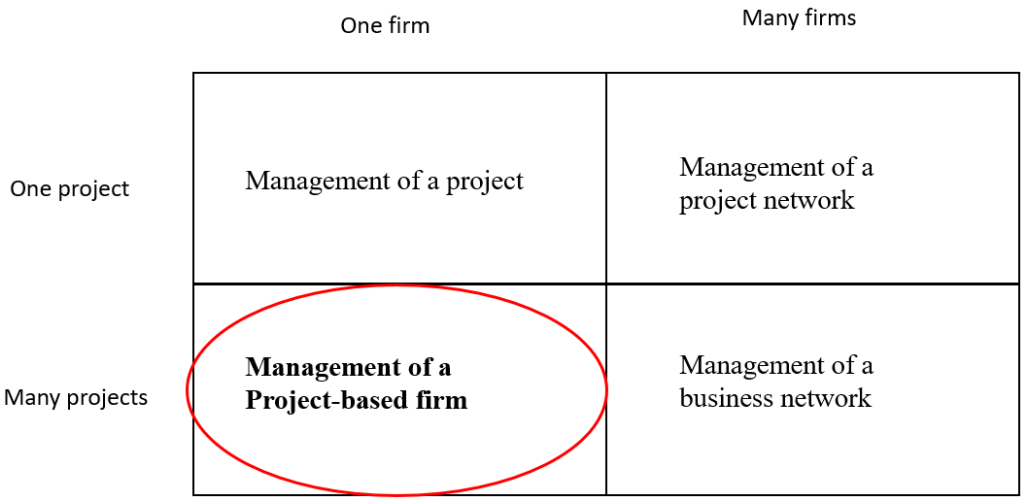

Figure 3: Framework of project business and the scope of this thesis (adapted from Artto & Kujala 2008)

2.1.1 Project management

A project is a group of activities and tasks with a pre-defined target, schedule, and resources (Todorović et al. 2015). Project Management Institute (2013) defined projects as temporary endeavors undertaken to create a unique product, service, or result. Temporariness relates to the fact that a project should have a defined starting and ending date. The project should end when its goals have been achieved or there is no longer a need for it.

The project lifecycle is the series of phases that a project passes through from its initiation to its closure (PMI 2013). Due to the uniqueness of each project, the lifecycles of individual projects are also different. However, all projects can be mapped to a generic lifecycle. There are various definitions for the stages of a project lifecycle, but they all share some similarities. Already in 1988 Pinto & Prescott identified that the project lifecycle can be divided into four stages: conceptualization, planning, execution, and termination. According to a more up-to-date definition of PMI (2013), these equivalent stages are starting the project, organizing and preparing, carrying out the project work, and closing the project.

The lifecycle of a project can be illustrated with a chart, as in figure 4, in which the x-axis presents the time factor and the y-axis the level of effort, meaning the costs and manhour associated with the project. (Pinto & Prescott 1988, PMI 2013)

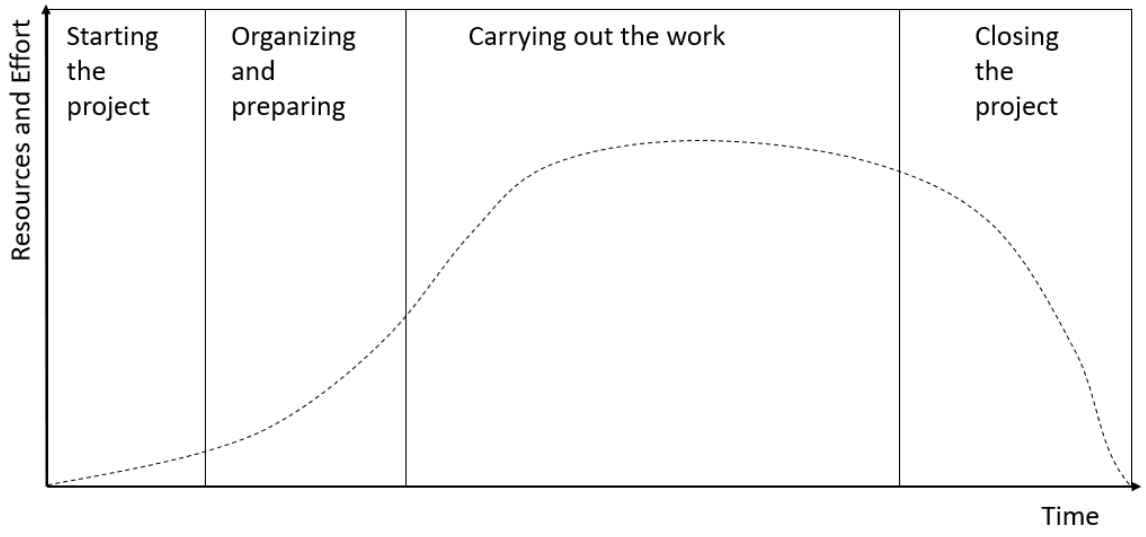

Figure 4: Project lifecycle (adapted from Pinto & Prescott 1988, PMI 2013)

Project management comprises the activities aiming at responding to the requirements of the project and it is performed at each stage of a project. Project management typically contains the identification of different stakeholder groups and communication with the stakeholders to develop the desired product. Also, balancing the limitations with project scope, quality, schedule, budget, resources, and risks is required within project management. (PMI 2013) Project management can be seen to be successful when it achieves the targets of the project which are typically related to schedule, costs, performance, and customer requirements. (Kerzner 2009)

2.1.2 Program management

Programs can be defined in many ways, but common for all the definitions is that they are identified as entities with a determined purpose and predefined expectations to the benefits (Artto & Dietrich 2007). For instance, Archibald (2003) stated that programs are long-term undertakings that include two or more projects that require close cooperation. The program consists of several associated projects that will contribute to the achievement of the strategic plan (PMI, 2013).

It is commonly recognized that projects are structured undertakings within a program and have a scheduled beginning and end (Kerzner 2009) while programs do not need to have a fixed timeframe or clearly defined deliverable (Pellegrineli 1997). According to Pellegrinelli (1997), a program is a framework for grouping existing projects or defining new projects, and for focusing on all the activities required to achieve a set of major benefits. This expression assumes that projects under the program are managed in a coordinated way, either to achieve a common goal or to gain benefits that would not be realized if they were managed independently.

Program management is increasingly used for the management of multiple interrelated projects (Archibald 2003, Pellegrinelli 1997, Thiry 2010). It can deal with complex issues when traditional project management is not adequate. Program management brings value by creating benefits that are implemented into the business through operations. In this case, the program can act as a link between the business strategy and the projects. Thiry (2010) expressed that the strategic objectives and expected benefits should be defined for the program to bring value.

Pellegrinelli (1997) highlighted that programs create benefits especially when the working environment contains multiple small projects. The value is created through the better organization of projects and programs taking a wider view to ensure business benefits from activities of the projects. The benefits contain the more efficient and appropriate use of resources, better planning and coordination, greater visibility, and better prioritization of projects. Blomquist and Müller (2006) recognized that it can also help in the development of business focus by defining the goals of individual projects and the entire program regarding the goals of the wider organization.

Programs should not be treated as big projects (Artto et al. 2009, Thiry 2010) or program management as multi-project management (Pellegrinelli 1997). Regarding this, Artto et al. (2009) pointed out that programs typically relate to a wide variety of management themes such as product development, organizational change, manufacturing, and quality. This means that in comparison to projects, that typically have a narrower scope of themes, programs have a broader emphasis on the number of activities. A typical view is that project focuses e.g. merely on product development whereas programs are considered as more long-term undertakings including two or more projects (Archibald 2003).

Table 1 below describes some key characteristics of programs and projects as well as the differences between them. Pellegrinelli (1997) identified the main difference between the entities to be that the program is typically defined as a framework while the project is an individual process within the program. Another difference is as Artto et al. (2009) recognized that the emphasis of different themes evolves in time with programs, whereas projects seem to evolve within the same product development theme. For instance, major changes in industry and society may introduce contemporary themes that programs are expected to address.

It can be concluded that programs and projects share some similarities, while still being entirely separate entities. There is something true in what Kerzner (2009) mentioned that the same policies and guidelines tend to regulate both project and program management and in certain circumstances, it is rather inconsequential whichever the undertaking is being called. However, like Artto et al. (2009) and Thiry (2010) expressed, programs should not be treated as scale-ups of projects or big projects.

Characteristic	Program	Project
Scope	A framework for organizing	A process to deliver a specific product, service, or result
Theme	Several themes of management science: product development, quality, work and organization change	One dominant theme
Level of analysis	Organization and its major parts	Single project
Time horizon	May have an undefined time horizon	Has a fixed duration
Objectives	Evolves in line with business needs	Has set objectives
Deliveries	May involve the management of multiple, related deliveries	Involves the management of a single delivery
Management responsibility	Program manager facilitates the interaction of numerous managers	Project manager has responsibility for the project's success
Types of innovation	E.g. process innovation, organizational innovation, infrastructure, and systems innovation	Product innovation
Outcome	Wide set of impacts	Concrete business results

Table 1: Characteristics of programs and projects (Adapted from Artto et al 2009, Pellegrinelli 1997)

2.1.3 Portfolio management

Portfolio management deals with the increasing number of projects ongoing in organizations. Ebert et al. (2005) defined portfolio management as the collection and evaluation of product and project information and the decision based on the totality of all projects to maximize their value. The projects may relate to various areas such as product development, changes in work processes, implementation of new IT systems, etc. However, a key task within portfolio management is to dedicate resources across all the various projects inside an organization. (Blichfeld & Eskerod 2007)

Project portfolio management (PPM) deals with the coordination of all the programs and projects of an organization. Archer and Ghasemzadeh (1999) defined a project portfolio as a group of projects that are carried out under the sponsorship and/or management of a particular organization. Portfolio management focuses on doing the right projects while project management on doing the project the right way (Ebert et al. 2005). Blichfeld and Eskerod (2007) specified PPM to comprise the managerial activities that relate to the initial screening, selection, and prioritization of project proposals, the simultaneous reprioritization of projects in the portfolio and the allocation and reallocation of resources to projects according to priority.

According to Cooper et al. (1997), portfolio management is a dynamic decision process in which the list of active new product projects is constantly updated and revised. During this process, new projects are evaluated, selected, and prioritized meaning that existing projects may be accelerated, killed, or de-prioritized. Resource allocation is a core activity of portfolio management, and resources are constantly reallocated to the active projects during the process. Ebert et al. (2005) stated that successful portfolio management aims at allocating resources most efficiently. This helps to reduce the number of projects to what is effective and to improve the communication between projects. Martinsuo (2012) formulated that portfolio management aims at efficient control of multiple projects pursuing the same strategic goals and competing for the same resources.

From the project business perspective, projects are the smallest entities and can be considered as the basic building blocks of project business whereas program management is the middle layer. Project portfolio management can be identified as the highest level of the hierarchy (see Figure 5). This level is composed of the management of all the programs and projects of the organization in a coordinated manner. It could be stated that

the project level is where most of the daily practical activities take place and program and portfolio levels are used for coordination of projects and to enable better synergies.

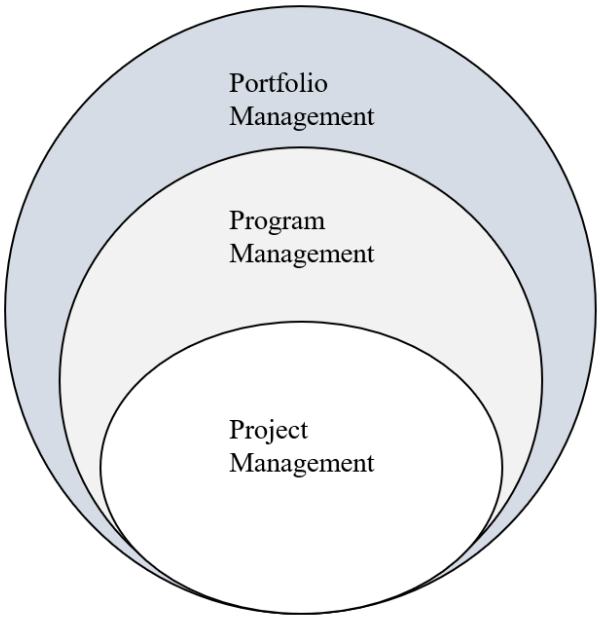

Figure 5*:* Portfolio, program, and project management

2.2 Agile program management

Traditionally projects are highly ordered but in complex situations and a turbulent environment, they may become unordered. This is what both program management and agile management aim to tackle and they are increasingly used for the management of multiple interrelated projects and stakeholders. (Thiry 2010) Agile management is discussed in this chapter. It originates from the agile concept that was first introduced in the software development context (Beck et al. 2001) but since then has been brought to a broader context (Cervone 2011). Also, SAFe is being scrutinized and the agile program management concept is being introduced.

Agile program management takes the traditional program concept and adapts it to the agile environment. It brings value through benefits that are created through operations. In this case, the program can act as a link between the business strategy and the projects. The strategic objectives and expected benefits should be defined for the program to bring value. (Thiry 2010) This seeks the continuous planning process that also Serrador and Pinto (2015) were referring to as they recognized that in agile management the early stage

planning is not skipped entirely, but it is rather done during the entire development cycle. According to Leffingwell (2007), agile entities are planned differently in comparison to "plan-driven" development, and the plan is revised and refined more often.

It should be noted that in the program context, agile management is primarily a development method (Thiry 2010). Within agile management, the focus is on adaptation to the changes, and in the production of high-quality products with simple processes (Dingsøyr et al. 2010). The focus of agile is on issues such as flexibility, iterative processes, team empowerment, and high levels of communication and hence can be used to reduce the need of using strategic and more structured practices on a project level. (Lappi et al. 2018)

2.2.1 The concept of agile

Agile management is an iterative approach typically used for managing software development projects (Drumond 2020). Agile management as a term has become known as a result of the dissemination of lightweight or agile software development methods. These methods include Scrum, Lean Software Development, Crystal, Feature-driven development (FDD), Adaptive Software Development, Dynamic System Development Method, and Extreme Programming. (Conforto et al. 2014b)

The agile concept was born in 2001 when the creators of these abovementioned methods joined forces to draw up a manifesto, called a manifesto for Agile Software Development. The manifest states that agile development should focus on four core values (Beck et al. 2001):

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

Agile methods were developed to thrive in situations that could not be dealt with using traditional project management methodology (Thiry 2010) and have grown in importance especially in the field of IT and software development. However, they are being increasingly used beyond the software development industry also e.g. in financial services, consulting, and even in machinery and manufacturing business (Conforto et al.

2014a). The agile concept has become an umbrella term for numerous new operations models of software planning, work coordination, communication to external stakeholders, and organizing software development both in large and small companies. (Dingsøyr et al. 2010)

Agile is a multidimensional concept that can be defined in many ways. Common for all these definitions is the focus on changing customer requirements and empowerment of the team with the proper balance of tasks and orientation. The most important thing about being agile is the readiness to execute changes according to customer needs while thinking of the change as a learning process. (Sheffield & Lemétayer 2013) According to Cockburn (2002) being agile requires effectiveness and easy steerability. The agile process, on the other hand, is light but adequate. Lightness is a way to easy steerability, and adequateness is a way to stay in the game.

Agile methods aim at minimalistic documentation through flexibility and maintaining the ability to react in changing environments. Agility also emphasizes early and continuous customer interaction. Customers should be involved in requirement setting, prototype development and to give feedback. The iterative nature of agile enables continuous interaction with the stakeholders, adaptation on the fly, and redefining targets according to the customer requirements. (Serrador & Pinto 2015) Agile methods also aim at identifying unpredictable challenges by recognizing influential persons and their influence on the project (Dingsøyr et al. 2010). Lindvall et al. (2002) defined agile characteristics to be iterative, incremental, self-organized, and that it allows requirements to change during the project. To conclude, these are the characteristics that separate agile management from traditional project management.

2.2.2 Agile methods in project management

Agile methods were initially developed to deal with projects that could not be dealt with using the methods of traditional project management (Thiry 2010). The methods revolutionized the execution and orchestration of software development (Stettina & Hörz 2015) and due to their success have spread to wider use (Serrador & Pinto 2015). According to a survey conducted by Conforto et al. (2014a), the agile methods are useful also in other circumstances than software development and they are being increasingly applied especially when a certain level of innovation is required. Still, the more traditional

management methods are more commonly used, but the agile approaches become more prevalent as the degree of innovation in the environment increases.

Agile management emphasizes a lower amount of planning work carried out in the beginning and prefers a more evolutionary type of process. Agile methods differ from the traditional management ones, such as the waterfall method, by stressing the continuum of planning, flexible scoping, and freezing the plan as late as possible. Being agile is described to be iterative and incremental because it aims to avoid the traditional model in which the planning is done in the beginning, the scope is fixed and the cooperation with customers is little. (Serrador & Pinto 2015)

According to a study carried out by Sheffield and Lémateyr (2013), one of the most defining properties of an agile project is its low criticality. This means that the impacts of late delivery or failure have been minimized. Another significant property is the experience and capability of the team. Due to the low number of personnel involved within the team, the importance of their expertise is being emphasized. Also, the uncertainty related to the requirements and used technology is often considered as a common characteristic for agile projects.

During the last years, the agile methodologies have been widely used to tackle challenges with traditional project management methods and they can be seen as a reaction to plan-based or traditional methods (Dingsøyr et al. 2010). Traditional project management is rather extensive, and it is proven to work in many situations. However, agile methods may end up being valuable in adding more ideas in some situations within creative and information-based industries. (Chin 2004) Especially in situations that require innovation. (Conforto et al. 2014a)

Within traditional project management, it is common to freeze the project's targets, budget, and schedule in the very early phase of the project. This means that also the quality of the outcome is defined by the fulfillment of these promises. The early definition of the project criteria, however, forces us to make accurate predictions of the required workload. To decrease the amount of this estimation work, new methods were developed to add clearance for defining the project criteria. (Anderson 2004) Therefore, agile methods may prove to be beneficial especially in an environment containing a high amount of uncertainty (Chin 2004).

Agile project management is not all or nothing method. Conversely, it is important to identify the means to combine the properties of traditional methods with the agile ones in a way that best suits the situation. (Chin 2004) Lappi et al. (2018) identified some traditional management practices that can be transferred to agile including clear organizational and ownership structure, stakeholder management, and risk management. In practice, the performance and the success of agile projects should be supported by practices giving project teams the freedom, authority, and capability to produce tangible value to and with the customer.

2.2.3 Agile processes and practices

Most of the agile processes and practices are based on Scrum and other commercialized approaches. It is common that under agile processes teams deliver product increments from sprints. Sprint, often referred to as an iteration, is a short and focused amount of delivery effort to deliver stories within sprint goal, that is normally between 2 and 4 weeks. The deliveries are done within releases and governed in a project. Once the product increment is delivered from the sprint, it is presented to the stakeholders. This allows for collecting feedback for the next planning period. (Measey 2015)

A product backlog is commonly used as the primary artifact for project management using agile practices. Sedano et al. (2019) defined it as a list of work items that teams use for coordinating the work and for planning the sprints. So basically, it is a container for all the requirements cut to pieces, most commonly stories and tasks. Other commonly identified work items are epics and features. In general, the product development efforts are directed by filling and prioritizing the backlog.

Agile teams typically use the backlog and its items to manage the scope and structure of the work. Initiatives, epics, and stories are commonly used concepts for that purpose. Rehnkopf (2020) identified stories as short requirements or requests that are written from the perspective of an end-user. Within the agile process, customer requirements are typically defined as features that are documented as user stories. Stories are commonly used as a form of defining requirements or features that can be delivered to the customer. They are used as a way of documenting what needs to be delivered. Stories reside on the backlog which is an ordered list of requirements that the customer wants. (Measey 2015; Rehkopf 2020)

Agile teams are responsible for deciding how to do the work. This contains refining stories with the customer and typically dividing them into tasks. A detailed breakdown is done for the tasks during sprint planning to create an effort estimate for each of the tasks. Typically, the tasks are broken down to be completed in less than 10 hours. The teams also organize themselves during the daily stand-up meetings to deliver maximum value to the customer. (Measey 2015)

Epics are large bodies of work that can be broken down into s features that are defined as user stories. According to Leffingwell (2009), epics represent the highest-level expression of a customer need. Initiatives compile epics from multiple teams to achieve a broader goal than any of the epics themselves. This structure is illustrated in the figure above (6).

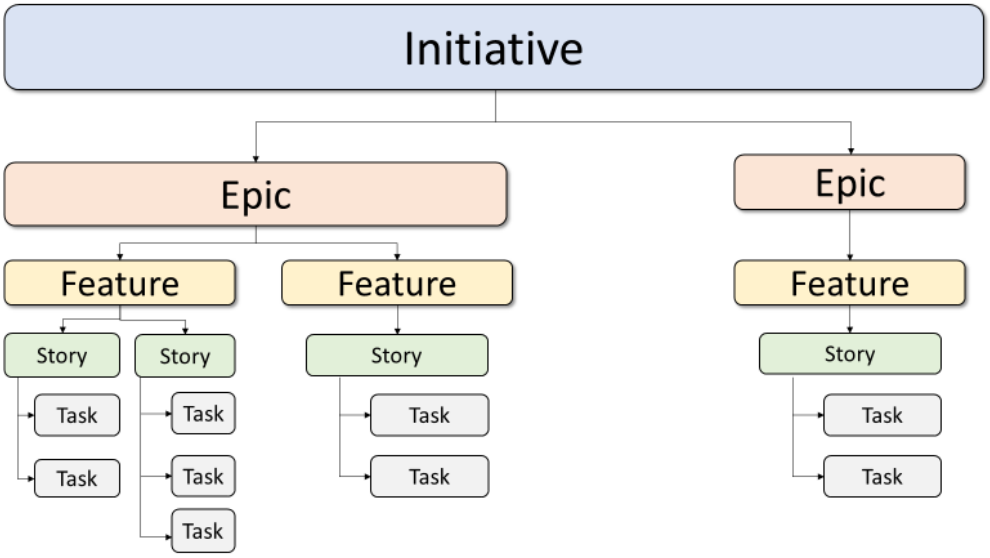

Figure 6: Initiative, epic, feature, story, task (hierarchy of the product backlog)

As illustrated in figure 5, initiatives may consist of multiple epics. Based on an epic, one or more features may be defined. User stories are used for documenting the features and one or more stories may be required to develop a feature. Tasks are typically the lowest level component of the hierarchy and usually, multiple tasks are derived out of a story.

2.2.4 Scaled Agile Framework

Agile methods were first used in small teams, projects, and companies. During the last few years, the usage has also been scaled up for use on a larger scale. (Nuottila et al. 2016) Scaled Agile Framework (SAFe) is a set of organization and workflow patterns for

implementing agile practices at enterprise scale. SAFe combines agile with lean product development and systems thinking (Leffingwell 2018) and gives guidance on roles and responsibilities, how to plan and manage the work, and values to uphold.

SAFe provides a process model covering the highest and the lowest level in the enterprise. The framework is organized around three layers: team, program, and portfolio. These are also illustrated in figure 7 below. The team level of the framework consists of agile teams, which are collectively responsible for defining, building, and testing software in fixed-length sprints and releases. SAFe also utilizes practices from other agile approaches such as Scrum, Kanban, Lean and agile manifesto. (Measey 2015, Turetken et al. 2017)

The agile program concept refers to the program level of the SAFe in which the Agile Release Train (ART) is the primary value delivery construct (Measey 2015). The main goal of the ART is to organize the teams in a way that optimizes the value delivery. (Turetken et al. 2017) ART aligns teams with a strategic vision and roadmap for each investment theme. Sheffield and Lémateyr (2013) emphasized the importance of the experience and capability of the team. Due to the low number of personnel involved within the team, their expertise is being emphasized.

The definition of Scaled Agile Inc. (2020) says that ARTs are organized around the enterprise's value streams and exist to create value by building solutions that deliver benefit to the end-user. According to Measey (2015), ART is typically formed with five to twelve teams operating within a program increment (PI) planning cycle covering 4-6 sprints, meaning a duration from 8 to 12 weeks. Program increments as a concept referring to these planning cycles covering the activities of the teams allocated to the program, which is typically from 5 to 12 teams. Business and architectural features are defined and prioritized in the program backlog during the increments.

The highest level of SAFe is the portfolio level in which programs are aligned to the enterprise business strategy. This is done along value streamlines which are a series of system definition, development, and deployment steps used to build and deploy systems that provide a continuous flow of value to the business or customer. This level is needed for organizations requiring governance and management models. Essential for the portfolio level is to achieve a balance between four goals: 1) Maximizing the financial value of the portfolio, 2) linking the portfolio to the strategy, 3) ensuring the feasibility

of scope by measuring appropriate metrics, and 4) balancing the portfolio on relevant dimensions. (Turetken et al. 2017)

In practice, the portfolio level is responsible for coordinating larger initiatives and epics that require implementation across multiple release trains. The coordination is done within the portfolio backlog that contains epics across the release trains. The epics can be customer-facing (business epics) or technology (architectural epics) initiatives. The portfolio level work is typically managed using a Kanban system. (Measey 2015)

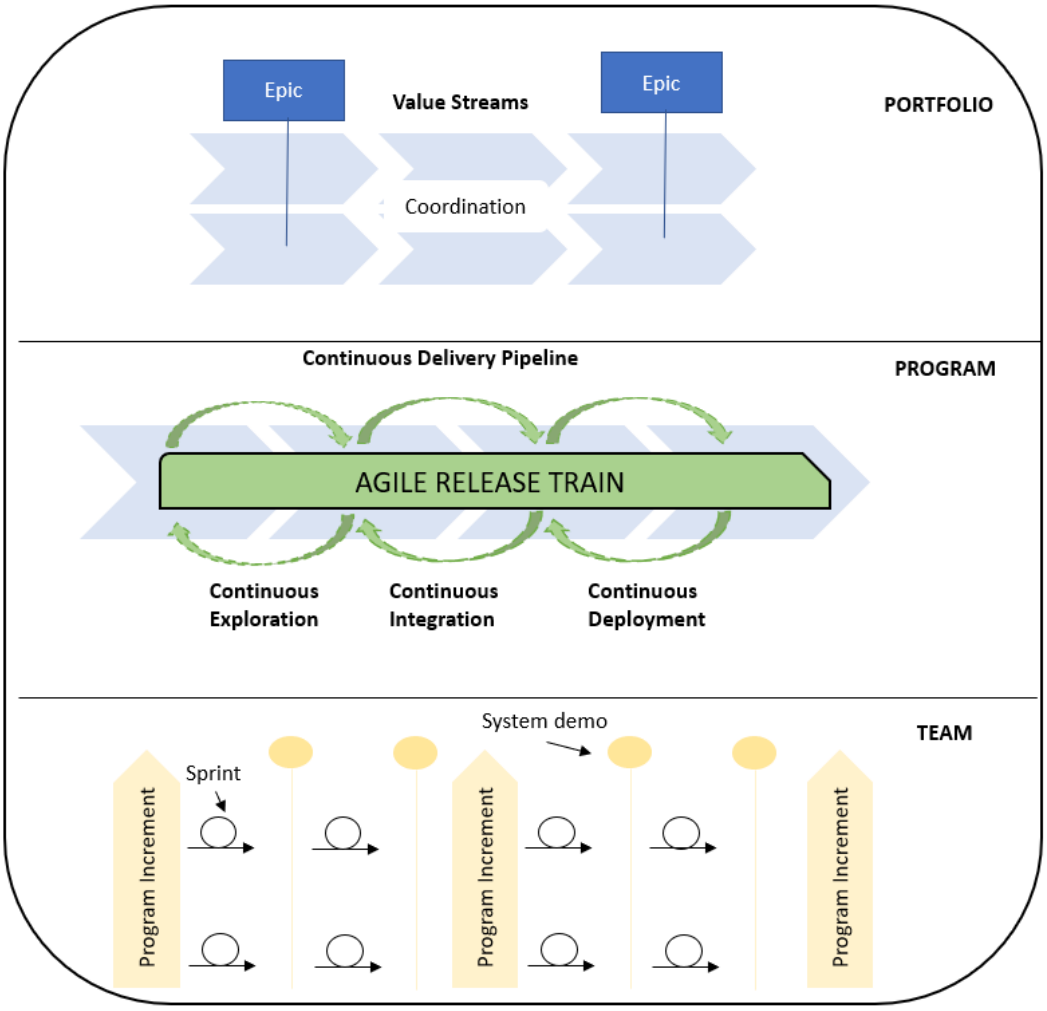

Figure 7: Main constructs in the portfolio, program, and team level of the SAFe (adapted from Scaled Agile, Inc. 2020)

2.3 Performance management

Performance management is a crucial practice for any organization that wants to learn and improve. Yadav et al. (2013) argued that performance measurement is the process to quantify the efficiency and effectiveness of actions and performance management is the preceding process of creating a context for measurement. Bonghez and Grigoroiu (2013) recognized that performance management covers the methodologies, metrics, processes, and tools that are used for managing performance. The use of dashboards and scorecards is typical for operational level performance management and it focuses on the objectives of departments or groups within the organization.

2.3.1 Performance criteria

Traditionally project performance has been defined in terms of meeting the cost, time, and quality criteria. (Bryde & Wright 2007, Mir & Binnington 2013) These dimensions also formulate the iron triangle criteria for managing project performance (e.g. Atkinson 1999, Caccamese & Bragantini 2012). Some definitions also include scope as a success criterion for the model (e.g. Caccamese & Bragantini 2012). The iron triangle is illustrated in figure 8.

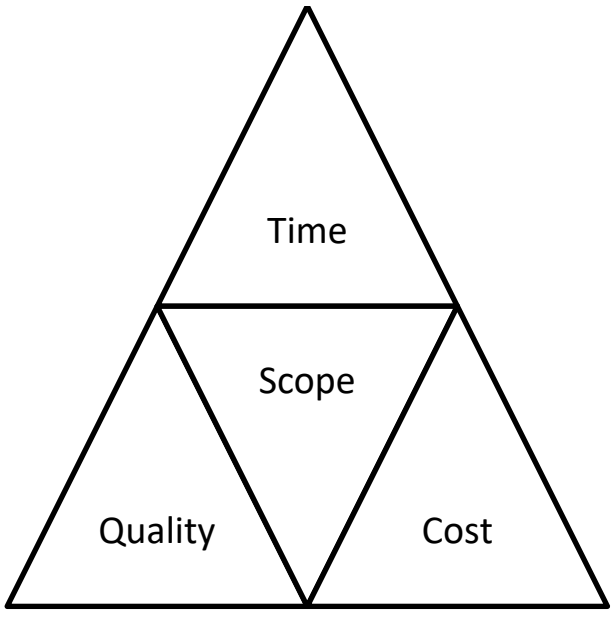

Figure 8: The iron triangle (Adapted from Atkinson 1999, Caccamese & Bragantini 2012)

The iron triangle criteria have received some criticism since it may encourage an emphasis on short-term measures and lead to performance optimization in the short term. That again can carry off the attention from other projects and psychosocial needs of project team members which may impair the performance in the long run. This gives motivation for managing the performance against a broader range of criteria. For instance, Mir and Binninton (2013) identified that a common approach is to focus on the expectations of the stakeholders. This, however, has led to difficulties since the stakeholders' needs are often difficult to manage and measure. Also, there may be resistance to go beyond the traditional criteria due to commercial pressures.

The fact that there are no two similar projects makes it difficult to create a common set of success criteria to be measured. Mir and Pinnington (2014) listed different ways of recognizing success such as assessing it according to short-term and long-term objectives or analyzing whether things were done right during the process. Bryde (2003) stated that a project can be viewed as a success despite poor project management and vice versa. The attainment of project management targets doesn't necessarily indicate that the targets the organization has set for the project have been accomplished. Therefore, performance management in project-oriented organizations should include the alignment of the objectives of the projects and programs to organizational goals, while also considering the individual and departmental goals. (Bonghez & Grigoroiu 2013)

2.3.2 Performance metrics

Cambridge Dictionary defines a metric as a system for measuring something. Metrics are used to track and assess the status of a specific process. In the project business, metrics keep stakeholders informed about the status of the project. They may determine if it is feasible to take on a certain project or if a certain course of action should be taken. The managers must agree with appropriate stakeholders on which metrics are to be used and how measurements will be made. Essential is also to understand what critical metrics need to be identified and managed for the project to be viewed as a success by all the stakeholders. (Kerzner 2013)

Measures typically refer to a numerical measure that represents the operating results concerning one or more dimensions. Todorović et al. (2013) categorized measures into traditional measures, performance indicators, and value measures. Traditional measures are used to measure the performance and the level of achievement of previously defined

results, including the cost and time variance. Performance indicators can be used to confirm that the critical success criteria defined at the beginning have been achieved. They have internal use but are also used for reporting on the status to the customer and other stakeholders. Third measures; value measures indicate whether the stakeholders' needs, such as the end date and value have been met.

One common method of deriving performance metrics based on business goals is the Balanced Scorecard (BSC). It was first developed by Kaplan and Norton (1992) and it is one of the best-recognized approaches to defining performance goals and measures. The scorecard encourages to first define the business goals and then consider the metrics based on internal and external perspectives. The scorecard provides a view of business value by considering the financial, internal business, customer, and innovation and learning perspectives.

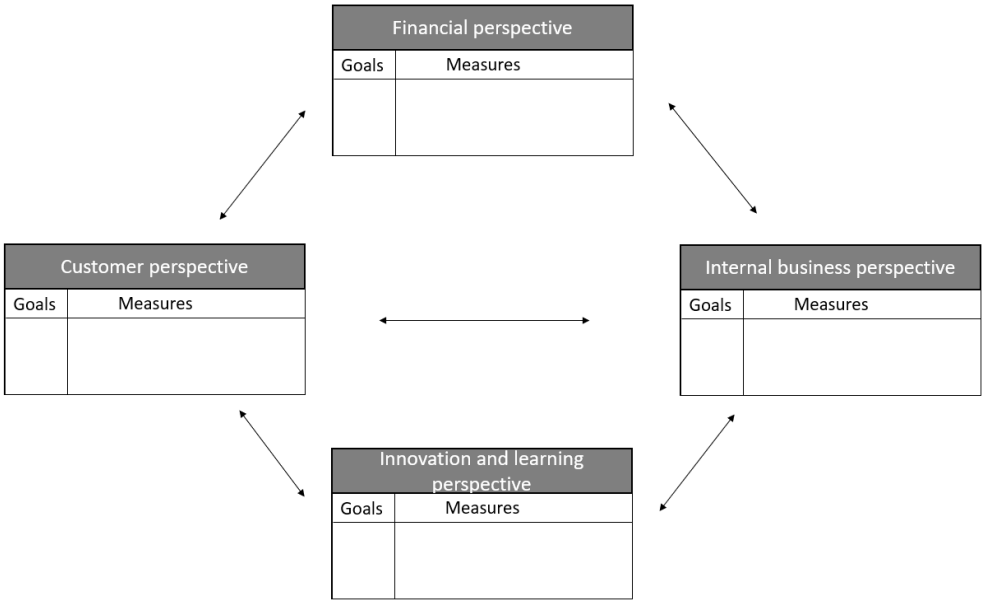

Figure 9: The Balanced Scorecard (adapted from Kaplan & Norton 1992)

A common phrase goes that "if it cannot be measured, then it cannot be managed". This indicates that it is impossible to define the metrics and to achieve the goals related to the projects without a clear definition of the measures. Lönnqvist and Pirttimäki (2006) underlined that an important issue in determining how and what to measure is to know the purpose of the measurement. Goal-Question-Metric is a commonly used concept for that purpose, and it can be used to supplement the BSC. According to Caldiera and Rombach (1994), it is based upon the assumption that measurement first requires

specifying the specific goals, and then data can be used to define those goals operationally. Questions should be defined based on the goals and then metrics can be identified to give answers to the defined questions (Ebert et al. 2005).

Once the metrics have been defined, it is highly beneficial to use a specific template for documenting the metrics to improve the common understanding around them. Nicoletta (2015) documented metrics regarding the questions answered, description, value, dependencies, and success factors. The questions answered should indicate what the metrics tell us while the description gives a brief description of the metric itself. Hartmann and Dymond (2006) also addressed e.g. the basis of measurement, assumptions, expected trend, and when to use the metric. The basis of measurement contains a clear statement of what is being measured including the units and the assumptions should be identified to ensure a clear understanding of the data represented.

2.3.3 Key performance indicators

A key performance indicator (KPI) is a metric that is often derived from organizational goals. Organizations today operate in a dynamic environment and therefore must deal with many KPIs in different areas. According to Eckerson 2006, (see Todorović et al. 2013) KPIs are measures that show how well the organization or individual is performing the operational, tactical, and strategic actions that are critical to the current and future success of the organization. The exact understanding of the KPI definition in the context of projects is in the explanation of its every word (Kerzner 2013):

- **Key** – a major contributor to the success or failure of the project
- **Performance** – a metric that can be measured, quantified, adjusted, and controlled
- **Indicator** – a reasonable representation of the present and future performances

Selecting metrics and KPIs is rather simple provided they can be measured. This, however, may be a major obstacle with some metrics. With traditional project management, metrics are established by the enterprise project management methodology and fixed for the duration of the project's lifecycle. However, with value-driven project management, metrics may change from project to project, during a lifecycle phase, and over time. (Kerzner 2013) This means that the metrics and KPIs must be separately chosen to fit each organization.

The first step in selecting the right KPIs is to define the general and specific characteristics that the KPIs should satisfy. Each KPI should be based on criteria that make it suitable for further analysis. The most often used set of criteria for constructing KPIs are the SMART-criteria. The SMART way to write goals and objectives was first introduced by Doran in 1981 and since then has been adapted to many formats. Today, the SMART-criteria are commonly used as a standard for developing effective and measurable goals and objectives.

The SMART-criteria are typically derived from Specific, Measurable, Achievable, Realistic, and Time-sensitive. (Shahin & Mahbod 2007) The criteria and their respective definitions are listed in the table 2 below:

Criteria	**Explanation**
Specific	Goals should be as detailed and as specific as possible, or "precise".
Measurable	Goals should be clear and concrete. Measures may be quantitative or qualitative, but measurement should be against a standard of performance.
Achievable/Accountable	Goals should not be out of reach. It should also be specified who will do it, meaning that they are in line with individual responsibilities.
Realistic/Relevant	The goals should be realistically achievable. They should also be directly related to the success or failure of the project.
Time-sensitive/Timely	The goals should have a time frame for completion, meaning when the results can be achieved. They should be suitable for their current needs.

Table 2: SMART criteria for defining KPIs (Adapted from Doran 1981, Ebert et al. 2005, Shahin & Mahbod 2007, Todorović et al. 2013)

In addition to the smart criteria discussed above, the KPIs should be based on real-time data to be timely. They should also be easy to understand so that employees know what is being measured, how it is calculated, and what they should do to affect the KPI positively. Also, effective KPIs sits at the intersection of multiple interrelated processes that drive the organization. (Kerzner 2013 see Todorović et al. 2013)

According to Todorović et al. (2013), good practices for the KPIs would be that they are automated, actionable, and predictive. In this case, the automatized reporting would minimize the chance of human error and could predict the future of the trend. Actionable KPIs would also trigger changes requiring corrective actions.

Kerzner (2013) pointed out that it is also important to consider the categorization of the KPIs in terms of what is being wanted them to show. One of the main KPI categorizations is as follows:

- Quantitative KPIs – numerical values.
- Practical KPIs – shows interfacing with company processes.
- Directional KPIs – getting better or worse.
- Actionable KPIs – effect change.
- Financial KPIs – performance measurements.

2.3.4 Performance measurement in agile programs

Dumke et al. (2008) argued that from a software measurement point of view not all metrics and methods from the traditional lifecycle models can be used without adaptation. Especially, agile techniques need new approaches for measurement. Hartmann and Dymond (2006) pointed out that under the agile process, the key metrics should allow the monitoring of the value produced by software development efforts. This means that metrics should help with making smart software investments and to deliver business value quickly. Misra and Omorodion (2011) stated that under the agile process, metrics are typically selected based on the firm and the kind of requirements. The team can design its metrics and use them based on the kind the needs.

Misra and Omorodion (2011) classified the agile metrics into four distinct categories: project metrics, resource metrics, process metrics, and product metrics. The project metrics are related to the traditional project success criteria of cost, time, and quality. In

the software development context, the resource metrics contain the measures related to the personnel, performance, software, and hardware. Especially one commonly used resource metric is the effort estimate (Kupiainen et al. 2015). The product and process metrics are primarily related to the products and their lifecycle. For instance, some more specific product metrics are the size of the product (e.g. count of code), architecture, and structure. The process metrics may be related to the management, maturity, or lifecycle of the product. (Misra & Omorodion, 2011) Each of these metrics discussed above also defines a range of metrics such as velocity, effort estimates, story points, and earned business value.

Kupiainen et al. (2015) identified that the common need for metrics is to support communication and decision making. The metrics are recognized to be useful especially in planning, estimation, and tracking as well as improving understanding of the quality and business objectives. Ebert et al. (2005) argued that metrics give timely and accurate information while enabling real-time and reliable decisions. Metrics can be used for instance to make reviews including the status and forecast e.g. for the quality, schedule, and budget. They are also commonly used for follow-up purposes of progress.

Kupiainen et al. (2015) identified some commonly used metrics in agile software development. The velocity was the most popularly used metric followed by effort estimate, customer satisfaction, and defect count. Cycle time and lead time were also among the popular metrics. Velocity is a commonly used metric in agile development. According to Nicoletta (2015), it is the "empirical measurement of the quantity of work the team delivers per unit of time." Hartmann and Dymond (2006) defined it as the amount of software that team can deliver per iteration. The measurement unit for velocity can be chosen by the team. The unit is typically defined as the backlog items developed during the iteration, but it may also be measured as the units of hours/days spent for the work.

The motivation for measuring velocity is to detect the efficiency of delivering the working software to the customers and if the team can deliver it at a stable rate (Hartmann & Dymond 2006). It is practical for short-term planning as it provides an empirical indicator of how much work a team can complete in a single time-boxed iteration. It is also useful for process improvement because it exposes irregularities in delivery performance. (Nicoletta 2015) Meaning that the team can compare its performance between the

iterations. It is especially important for planning how much work can be accomplished in the upcoming sprints (Rossberg 2019). Velocity metrics can also be used to improve the effort estimates for upcoming planning sessions and to help estimate the scope of the iterations (Kupiainen et al. 2015).

Software development effort typically includes the human effort required for the design, coding, and testing activities of the software product. It is often considered as the substitute for software development cost since personnel cost is the dominant cost in software development. (Agrawal & Chari 2007) Effort estimation is an estimation of the amount of effort that is needed to develop a software product. It is typically measured in terms of person-hours or money. (Kupiainen et al. 2015) Effort estimates can be used as a basis for resourcing and to prioritize the features for upcoming releases (Kupiainen et al. 2015). There are various models for effort estimation. Effort-estimation models primarily use the number of source lines of code as the basis for the effort estimation. (Agrawal & Chari 2007) This means that the amount of code is used as a basis to estimate the needed effort in terms of personnel cost.

Cycle time is an important metric because software projects are often carried out under strict delivery schedules (Agrawal & Chari 2007). Cycle time is measured as the time between the beginning and releasing time of an item. In the software development context, it is the elapsed time from the moment a work item is started until the work item is finished. Nicoletta (2015) described it as the “projection of the team’s likely future delivery performance based on empirical measurement”. It provides an indicator of the team’s delivery performance and can provide early warning of the potential risks associated with the delivery.

The common use of cycle time is to track the time for completion of a work item, and it is typically measured in hours. Cycle time can be used to identify the past performance of software development as well as to create a guideline for future performance. (Nicoletta 2015) The software development cycle time depends on two factors: planned value time and the difference between planned and actual development times. Agrawal & Chari (2007) stated that capacity-related issues such as employees participating in unplanned activities are the primary reason for actual development time being longer than planned.

A commonly used definition of software quality is the amount of post-release defects in a software product. Ebert et al. (2005) stated that defects are the reason why the software causes failure when executed. Software quality is commonly measured as the number of defects per thousand lines of code. (Agrawal & Chari 2007) The ISO 9126 standard defines software quality as "the totality of features and characteristics of a product or service that bears on its ability to satisfy given needs."

2.4 Business Intelligence

Before discussing the business intelligence concept more in-depth, it is relevant to go to the grass-roots level and elaborate on how the primary components; data, knowledge, and information are defined. Laursen and Thorlund (2016) defined data as the carrier of information. Data as such do not deliver any value to the user since it is too specific to be useful as decision support. Information is data aggregated to a level where it makes sense for decision support. Data become information when it is put in a specific context. Knowledge is generated when information has been analyzed and interpreted. This means that new know-how or understanding has been created.

Business intelligence can be simply defined as "the use of data to help make business decisions". It refers to a managerial philosophy and a tool used to help organizations manage and refine business information to make more effective business decisions (Lönnqvist & Pirttimäki 2006). BI became a popular term in the business and IT communities in the 1990s when BI techniques and tools were developed to turn data into information and to turn information into knowledge and plans that could be used to drive effective business activity (Eckerson 2011). BI can also be considered as a data-driven process that combines data storage and gathering with knowledge management to provide input into the business decision making process.

The common use of BI is to transform data into information. Reporting and analysis tools are used to examine the information and to convert it into knowledge. The software used in BI includes database management systems, data cleansing, data transformation, and analytical systems (Larson & Chang 2016). Users can discover trends and create rules based on them to power their operational plans. (Eckerson 2011) BI is also beneficial in understanding the capabilities available in the firm. This includes e.g. the state of the art, trends, and future directions. According to Božič & Dimovski (2019), BI can give insights

for different purposes such as to understand market behavior, optimize business processes, develop new products and services, manage risks, improve efficiency, identify faults, etc.

One of the main benefits of BI is that it can be used to assist strategic and operational decision making. A Gartner survey ranked the strategic use of BI in the following order (Willen, 2002 according to Negash & Gray 2008): 1) Corporate performance management; 2) Optimizing customer relations, monitoring business activity and traditional decision support; 3) Packaged standalone BI application for specific operations or strategies; 4) Management reporting of business intelligence.

2.4.1 Business intelligence system

BI systems provide actionable information delivered at the right time, at the right location, and in the right form to assist decision-makers (Negash & Gray 2008). Popovič et al. (2019) defined a BI system to provide "quality information in well-designed data-stores, coupled with software tools that provide users timely access, effective analysis, and intuitive presentation of the right information. "This definition enables the making of right actions by making the right decisions.

The simplified function of a BI system is shown in figure 10. The input data for the BI system may be structured or unstructured and the data is processed with the BI system to get decisions as output. (Negash & Gray 2008) Most BI systems include different technological components such as databases, visualization tools, and on-line analytic processing (OLAP) that allow decision-makers to view and work with subsets of data. (Richards et al. 2019)

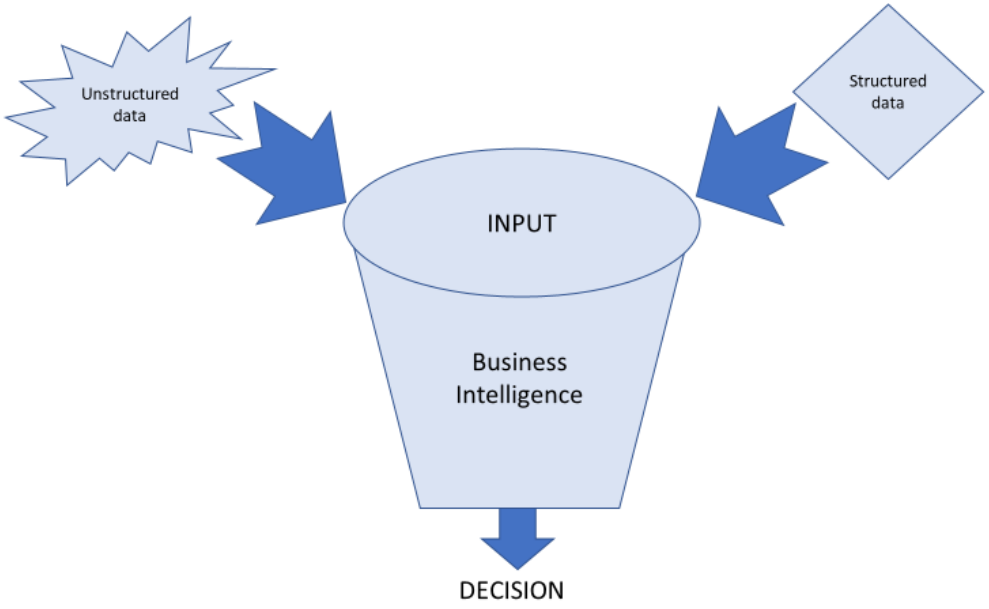

Figure 10: Business intelligence system (Adapted from Negash & Gray 2008)

BI as a term today heavily associated with large software vendors that offer reporting solutions for end users (Laursen & Thorlund 2016). According to Richards et al. (2019), BI systems are software applications delivering information to decision-makers to help maintain business performance. BI system delivers data to decision-makers by deriving insights of the data by using analytic techniques which can range from simple tables to more sophisticated analysis.

Laursen and Thorlund (2016) stated that "analytics is an advanced discipline within business intelligence". This means that analytics contains the means to analyze critical business data to help the enterprise make timely business decisions. Business analytics as a practice refers to the use of statistical techniques that leverage data delivered through a variety of sources including standard BI systems. (Richards et al. 2019) Data analytics comprises acquiring data, organizing it, discerning patterns in it, and determining how the insights can guide decision-making. In addition to the underlying data processing and analytical technologies, business intelligence and analytics includes business-centric practices and methodologies that can be applied to various applications. (Chen et al. 2012)

Larson & Chang (2016) pointed out that BI enables organizations to enhance the decision-making process but requires processes, skills, technology, and data to do so. The main objective of BI usage is to improve the timeliness and quality of inputs to the decision process. As discussed by Negash and Gray (2008) BI can be used for a wide range of

purposes such as creating forecasts based on historical data, past and current performance. It is also possible to create estimates of the future direction as well as to create a "what if" analysis of the impacts of changes and the alternative scenarios relative to it. It is common to aim at deriving strategic insights of the data and to utilize ad hoc access to the data to gain an answer to specific, non-routine questions.

Alpar and Schulz (2016) noted that BI systems have changed in recent years. One major change is the growth in potential data sources such as social media systems, machine sensors, and devices like smartphones. Another interesting change has been the transition in the scope of BI from strategic questions to operational tasks so that more employees need to apply BI. These developments have increased the demand for BI solutions.

2.4.2 Business intelligence process

Lönnqvist and Pirttimäki (2006) discussed the general phases for a BI process to include the identification and acquisition of information needs as well as the analysis and utilization of the information. The process is rather aligned with the information management cycle defined by Choo (2002). Choo's cycle is a continuous process with six phases (see figure 11): 1) identification of information needs, 2) information acquisition, 3) information organization, 4) development of information products and services, 5) information distribution, and 6) information use. Adaptive behavior starts a new cycle as the organization makes decisions and takes actions that result in effects and outcomes (Choo 2002).

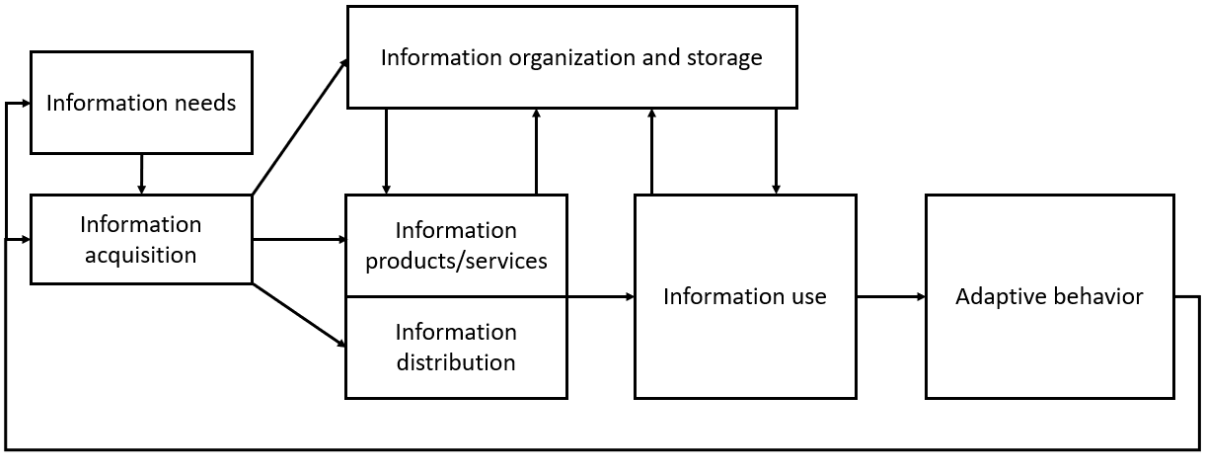

Figure 11: The information management cycle (adapted from Choo 2002)

The business intelligence process starts with the identification of information needs. This means that the needed business information is identified containing the kind of information required to resolve different problems and to make successful decisions. The information acquisition is driven by the business information needs and is considered a complex function since there are multiple different sources of information both inside and outside the organization. (Lönnqvist and Pirttimäki 2006) Božič & Dimovski (2019) stated that acquisition capability refers to identifying and obtaining valuable data from external sources. The data typically requires careful cleaning, conditioning, and integration of data sets to make them usable.

After the information has been acquired, the information is analyzed and transformed into solutions (Lönnqvist & Pirttimäki 2006) or information products and services (Choo 2002). According to Choo (2002), the information should be provided in a form that increases its usability. In the business intelligence context, the solution is most commonly a dashboard. The information used should enable decision-makers to find the required information as quickly as possible (Lönnqvist & Pirttimäki 2006) According to Božič & Dimovski (2019) also the transformation capability plays important role in facilitating the internalization of acquired knowledge within an existing knowledge basis. The insights need to be disseminated within the organization to reach the decision-makers.

Stakeholders of the BI process

Multiple roles within an organization can benefit from the use of the capabilities provided by modern data solutions. Bragen (2018) identified key roles within an IT organization to be IT managers, project managers, and project management office. IT managers may use analytics to catch potential risks before they even become problems. This helps to ensure that the projects meet the business needs. Project managers can use analytics to manage the human dimensions of projects. Automatized reporting may also be set up to give warnings of trouble when the KPIs trigger boundary values. Analytics may reduce the time spent in meetings and manual data-related work.

Figure 12 describes the people and tasks involved in the creation of business analytics as discussed by Laursen and Thorlund (2016). However, this model also applies well to the BI process. In the top layer of the model, in the business environment, the management specifies the information strategy based on the overall strategy of the organization. The strategy is used as a foundation to establish the operational business processes. In the

middle layer of the model, analysts, controllers, and report developers create the information and knowledge to be used by the company's operational decision-makers. In the technical environment, the database specialists or the ETL (extract, transform, load) developers merge and enrich the data to make it accessible to the business users. In the bottom layer, the IT professionals run and operate the date generating source systems. (Laursen & Thorlund 2016)

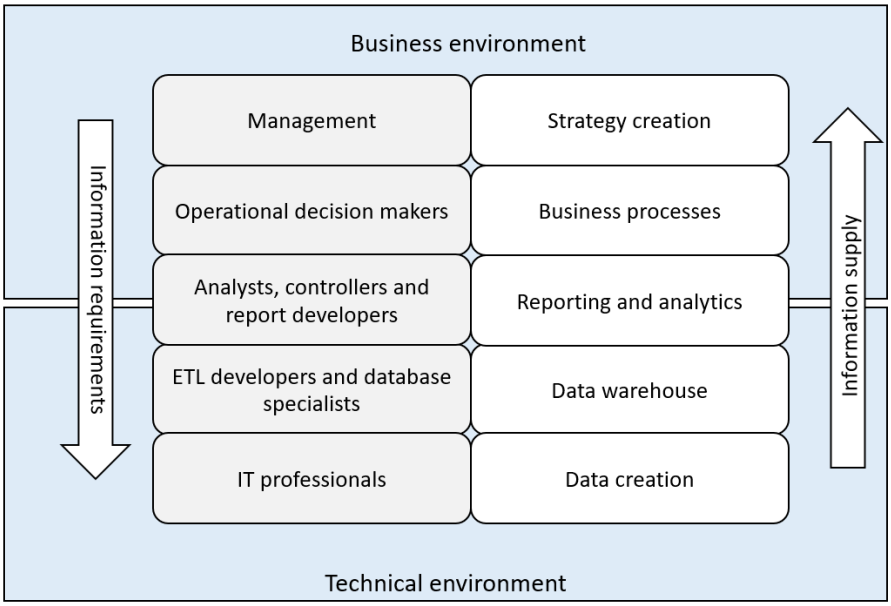

Figure 12: Stakeholders of the BI process (adapted from Laursen & Thorlund 2016)

2.4.3 Lifecycle of a business intelligence solution

Gangadharan and Swami (2004) underlined that BI is a strategic initiative in which organizations measure and drive the effectiveness of their competitive strategy. Larson & Chang (2016) pointed out the fact that the expectations of BI projects are not always clear to all the stakeholders. Commonly, end users know that they need information and analysis capabilities, and IT knows they need to deliver something. Therefore, BI delivery requires intense collaboration between business and IT stakeholders. Also, management commitment, support, and sponsorship have been widely accepted as the most important factor for BI system implementation. Without consistent support from business executives, it is difficult to secure the necessary resources for the project. (Yeoh & Koronios 2010)

Elbashir and Williams (2007) elaborated that BI should be integrated into an organization's business strategies and processes. Achieving BI impact requires the assimilation across the entire BI solution lifecycle. There are many opinions on how the lifecycle can be divided into phases. The lifecycle typically starts with the identification of the business requirements and analysis of the information requirements. Ending up to the value delivery phase through design, development, and implementation. The inference of the lifecycle phases is illustrated in figure 13. (Gangadharan & Swami 2004; Larson & Chang 2016)

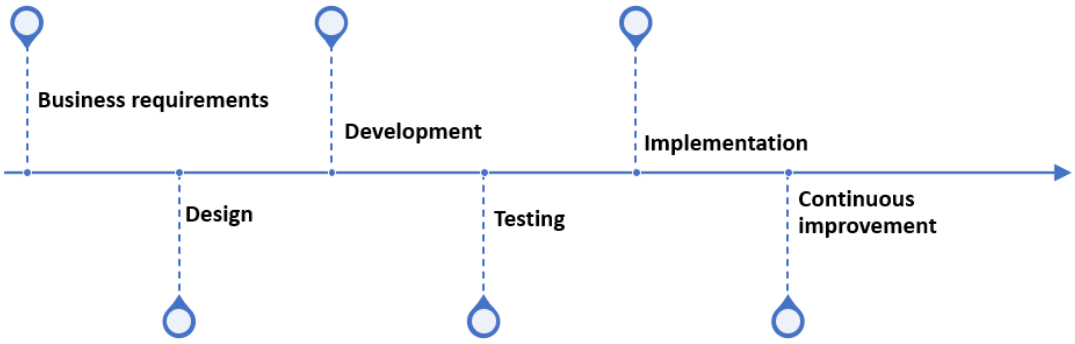

Figure 13: The Business Intelligence lifecycle (adapted from Gangadharan & Swami 2004, Elbashir & Williams 2007, Larson & Chang 2016)

The BI lifecycle starts with the identification of business opportunities with the aim of finding opportunities to improve the core business processes (Elbashir & Williams 2007). During the identification, the information requirements are defined according to business questions providing insight into data sources, dimensions, and facts needed (Larson & Chang 2016). Based on the requirements, business cases should be developed for the targeted business processes (Elbashir & Willliams 2007). Without the well-established business case the implementation may not satisfy the business needs and with a high level of probability will result in a failure (Yeoh & Koronios 2010).

According to Larson & Chang (2016), the design lifecycle phase focuses heavily on modeling and mapping but also requires establishing the architecture of the system. Also, the focus should be on the business processes that are to be improved (Elbashir & Williams 2007). The development and testing phases produce a working system that applies business context to the data and present information in a way that enables end-users to analyze actionable information.

According to Larson and Chang (2016), the primary goal of a BI project is to enable the use of information. Typically, BI development is less about creating a working product and more about the application of business context to data. According to Yeoh and Koronios (2010), the development approach should also be business-driven and iterative. An incremental delivery approach allows an organization to concentrate on crucial issues and seeks a high level of user involvement throughout the development process in order to deliver a system that meets the users' needs.

Through implementation, new components are introduced, and the stability of the BI system is maintained. Yeoh and Koronios (2010) also noted that the implementation is not a conventional application-based IT project. Instead, it shares similar characteristics with projects such as enterprise resource planning (ERP) systems implementation. This means that it is a complex undertaking requiring appropriate infrastructure and resources over a lengthy period. Implementation requires an interactive approach with extensive user training and adjustments.

The continuous improvement lifecycle phase contains the collection of BI usage and process-related data to be used in the value delivery including stabilization, maintenance, change management, and end-user feedback. Also, further process improvement opportunities are identified. (Gangadharan & Swami 2004, Elbashir & Williams 2007)

2.4.4 Business intelligence dashboard

The use of dashboards is common for operational level performance management (Bonghez & Grigoroiu 2013). Zdonek (2020) articulated that a dashboard is a tool used for managing all the business information from a single point of access. Dashboards help managers and employees to keep track of the company's KPIs and utilizes business intelligence to help companies to make data-driven decisions. Kerzner (2013) specified that dashboards aim at conveying the most critical information to the stakeholders the fastest way.

Kezner (2013) divided dashboards into three categories: operational, tactical, and strategic. Operational dashboards are the most commonly used dashboards in programs since they monitor the core processes and are used for the delivery or creation of organizational products and services. These are also the kind of dashboards used in this research. Tactical and strategic dashboards are primarily targeted for portfolio-level

reporting since they track departmental processes and projects that are of interest to a segment of the organization or a limited group of people and monitor the execution of strategic objectives.

One major ambition for data usage in project business is to gain visibility into data to improve decision making. Bragen (2018) categorized data collected by projects into three categories: past, present, and future. Present information is used for triggering status conditions that may need immediate action. It can be used to create real-time alerts about conditions meeting specific criteria. The historical data is useful from an analytics perspective when it is used as a foundation for improvement. The most effective use of past and present data is to use it for post-project reviews and to create checklists of possible problems for future projects.

Zdonek (2020) expressed that dashboards can display metrics for overall performance and progress. Their main object is to highlight problems requiring further attention within the project team. They are typically created for individual departments that want to monitor the progress and success of the projects. Therefore, the metrics used on them depend primarily on the department and the objectives of the projects. Data visualizations are used within a dashboard to provide an interactive way to explore data (Zheng 2017).

Dashboards commonly use data visualizations to represent data as pictures to support reasoning about the underlying data. Visualizations can be used for the representation of data of any size, type, or origin. Brady (2014) defined visual management as "a management strategy for organizational control, measurement, and improvement which uses visual aids to externalize information and improve communication by creating transparency." The goal of visualizing business data typically focuses on human information seeking and decision-making behaviors. Particularly in two broad goals. Firstly, to visualize the key metrics for easy and fast comprehension directly facilitating decision-making. Second, to provide a visual and interactive way to explore data. (Zheng 2017)

Visualizations are increasingly used to display data in decision making. Killen and Geraldi (2020) discovered improved success from the use of multiple types of visualizations by decision-makers. Visualizations may be used as a tool to reduce decision complexity and to enable better decisions to be made by a wider range of decision-makers. In a project decision environment, the requirement for simultaneous consideration of

several interrelated criteria may suit managers with a high need for cognition but alienate managers with a low need for cognition. Therefore, visualizations may improve decision making by effectively engaging managers with a low need for cognition by reducing complexity.

Visualizations can be used to display the performance information in a minimal amount of space. One commonly used indicator is the traffic light reporting concept. Red light indicated that a problem exists which may affect time, cost, quality, or scope. Yellow light is typically used for caution. It says that a potential problem may exist, perhaps in the future if the situation is not monitored. A green light is a state in which the work has progressed as planned. (Kerzner 2013)

Brady (2014) stated that the traditional approach to project management hinders effective communication and that visualizations can be used to support the deficiencies. However, this can be tackled with the use of agile methodologies such as Kanban, which aims at visualizing the workflow. Visualization eases management and makes it clearer for developers to understand the overall direction of work and helps to manage the flow. This also motivates and makes it easier to control the tasks. (Ahmad et al. 2013)

2.5 Synthesis of the literature review

Based on the literature review, more and more organizations arrange their business through projects (Schoper et al. 2018). The growing number of projects puts firms in a situation where the continuous identification of new tools and technologies as well as the development of the existing ones are found beneficial both in the management of a single project as well as in the management of a project-based firm. This has increased the importance of topics such as program and portfolio management. A project is typically defined as temporary endeavors undertaken to create a unique product or service (PMI 2013), while a program is a framework for grouping existing projects or to define new projects (Pellegrinelli 1997).

It is a commonly recognized fact that measurement is a prerequisite for management. Projects generate huge amounts of data to be utilized in various activities such as decision making. Project performance has been traditionally measured in terms of meeting the cost, time, and quality criteria (Bryde & Wright 2007, Mir & Binnington 2013). However,

the broader emphasis is highly advised to achieve benefits in the long run. Metrics are commonly identified to improve the understanding about the objectives of the projects as well as of the business in general (Kupiainen et al. 2015). This builds upon the requirement for systematic methods for data collection, processing, and utilization. This is where concepts such as business intelligence and analytics come in handy.

The program concept in this study refers to the program level of the scaled agile framework (SAFe). The primary construct at the program level is the agile release train (ART) typically consisting of 5-12 agile teams working together towards building solutions for the end-users. (Measey 2015) Agile teams typically use specific concepts to help manage the scope and structure of the work. Teams deliver product increments from sprints and during the process various process-specific metrics and measures are used. The product backlog is a container for the work items (e.g. epics, features, tasks) of the product development and it is commonly recognized as the primary artifact for agile project management (Sedano et al. 2019).

Business intelligence refers to a managerial philosophy and a tool used to help organizations manage and refine business information to make more effective business decisions (Lönnqvist & Pirttimäki 2006). In practice, BI is used to automatize the process of transforming data into knowledge. Users can use reporting and analysis tools for examining the information and view the data as visualizations representing the data as pictures supporting the reasoning about the underlying data. (Brady 2014)

The benefits of using business intelligence are taken into practice via a dashboard. A BI dashboard is a tool that can be used to manage all the business information from a single point of access (Zdonek 2020). It helps to keep track of the company's KPIs and to make data-driven decisions. KPIs are metrics that are derived from the organizational goals and they should be selected carefully, for instance using the SMART-criteria (Doran 1981). The implementation of a BI system is a complex function and requires resources over a lengthy period (Yeoh & Koronios 2010). The business intelligence process can be illustrated via Choo's (2002) information management cycle starting with the identification of information needs and acquiring the information to be utilized in the development of the business intelligence solution. Multiple roles within an organization can benefit from the use of data solutions such as project managers and IT managers (Bragen 2018).

The metrics to be used in the performance measurement of agile programs are defined by the characteristics of traditional program management as well as the definition for the agile program. Archibald (2003) defined traditional programs as long-term undertakings that include two or more projects that require close cooperation. This means that the same performance dimensions that are commonly used in project management are highly applicable also within the program level. These include issues such as budget, status, risks, stakeholders, and milestones.

However, the performance measurement of the agile program typically needs new approaches (Dumke 2008). Misra & Omorodion (2011) stated that under the agile process, metrics are typically selected based on the kind of project. Kupiainen et al. (2015) identified some commonly used metrics in agile to include e.g. velocity, effort estimate, and cycle time. The combination of these agile specific metrics with the more traditional project management metrics, such as budget, risks, and milestones, acts as a good foundation for a business intelligence dashboard development.

Table 3 below summarizes the agile program management metrics that were identified. The metrics are categorized based on the dimensions of the iron triangle

Category	**Metrics in agile program management**
Cost	Budget, Costs, Resources, Effort estimate
Time	Schedule, Milestones, Cycle time, Lead time, Velocity
Quality	Count of internal and external bugs, Number of test cases, Test coverage
Scope	Count of work items, Work in progress, Risks, Dependencies

Table 3: Agile program management metrics (Todorovic et al. 2013, Kupiainen et al. 2015, Misra & Omorodion 2011)

3 BACKGROUND OF EMPIRICAL RESEARCH

The empirical research of this study is conducted as a case study within a case company. As discussed in the previous chapter, the research was started out by carrying a rather extensive literature review. The literature review was accompanied by preliminary discussions with the key stakeholders of the project. The findings from these were used as a foundation for the empirical part of this research. This section discusses the research approach, research environment, and methodologies used in collecting the materials for this study as well as conducting the research itself.

3.1 Research method

The main ambition of this study is to design and implement a business intelligence dashboard for the case company. For this purpose, design science was chosen as the main research method due to its high level of applicability in this type of study. Design science is a commonly used research method when the goal is to build a solution to solve an identified problem. However, other research methods such as a survey and interviews were used to complement the general design science methodologies.

Design science is a commonly recognized method for research in the engineering and computer science disciplines. According to Hevner et al. (2004), design science is fundamentally a problem-solving process that involves designing artifacts, evaluating the design, and communicating the results to appropriate audiences (Peffers et al. 2007). In the process, the solution to the research problem is acquired while building artifacts that can broadly be defined as constructs, models, methods, and instantiations of the process (Matook & Brown 2017). Design science seeks to create innovations through which the analysis, design, implementation, management, and use of information systems can be accomplished.

Figure 14: DSRM model (Adapted from Peffer et al. 2007)

Peffer et al. (2007) defined a six-step model to execute the design science research methodology (DSRM). Next, these steps shown in figure 14 are applied to the context of this study.

1. Identify problem & motivate

The first step of conducting design science research is to identify the problem and to show the value of the solution. According to Hevner (2004), the problem should have relevance and the objective is to develop technology-based solutions to important and relevant business problems. It should also motivate to pursue the solution and to accept the results (Peffer et al. 2007).

In this study, the problem is that there is no commonly used tool for the data-driven performance measurement of the agile programs. The managers with responsibility for the programs need to manually collect the data and recreate the reports each time they want to give a presentation about the underlying issues. Also, the current tools for reporting aren't flexible enough to provide a long enough time horizon for reporting purposes.

The need for this study arose since it appeared that there was a need to develop the performance reporting practices related to the product development programs. Business intelligence had already been introduced as a solution to some extent, but it had not been taken to an adequate level. Therefore, a study related to this topic seemed rather advantageous. The primary motivation for this study is to give insights into the good practices of business intelligence usage in the program management area.

2. Define objectives of a solution

The second phase of the DSRM model is to define the objectives of a solution. These are typically inferred from the problem specification. The objectives are used as an input to create and design the artifacts and the artifacts should solve one or more instances of the identified problem. Hevner (2004) announced that it should be designed as an artifact, meaning that the research should produce a viable artifact in the form of a construct, model, method, or instantiation.

In this research, the survey was conducted as part of defining the objectives for the solution. Primarily, the aim was to gain an understanding of conducting in-depth interviews for the main stakeholders of the study. The findings from the interviews were then used to design and develop the dashboard. At this point, also the empirical part of the thesis was written containing e.g. the development proposals for the company as well as discussing the potential changes to be carried out inside the company.

3. Design & development

Figure 15 illustrates the design science research cycles defined by Hevner (2007) applied to the context of this study. The relevance cycle of the figure bridges the contextual environment of the research with the design science activities. The rigor cycle connects the design science activities with the knowledge base of scientific foundations, experience, and expertise that informs the research project. The design cycle in the center iterates between the core activities of building and evaluating the design artifacts and processes of the research.

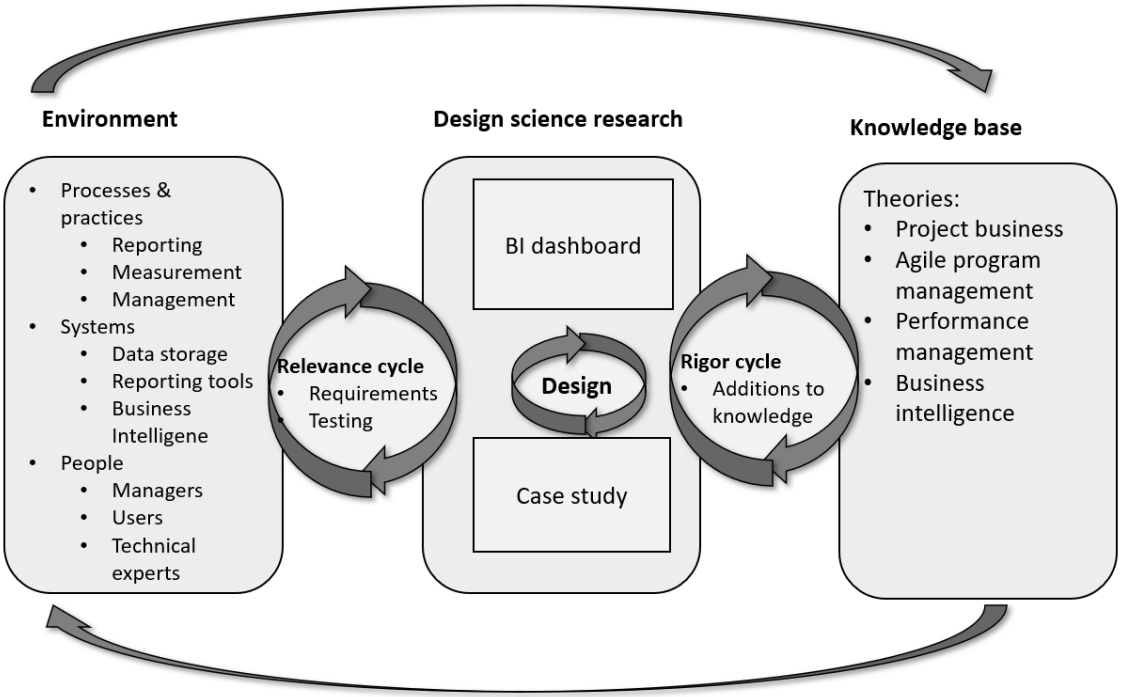

Figure 15: Design science in this study (adapted from Hevner 2007)

The above-discussed principles of the research cycles are also used in conducting this research. The literature review forms the knowledge base of this study and the

environment is being analyzed to define the requirements for the BI dashboard. Most of the knowledge base of this study is formed by the literature review discussed in chapters 2 and 3. The relevant scientific theories, methods, and concepts are utilized as the grounding to develop the main artifact of this research. The knowledge base is also accumulated by familiarizing with the environment of the study.

The environment of this study is formed by all the people involved in the process, especially the users of the dashboard, organizational systems, and technical systems as well as the processes and practices of the case company. The requirements for the dashboard and related processes are collected from the people, also considering the system requirements. The research environment was first analyzed through a survey that was carried out with a Microsoft forms questionnaire.

The actual design science research consists of development activities related to the dashboard. The development was done iteratively by first developing a prototype, meaning the first version of the dashboard. The dashboard was introduced to the user groups and feedback for further development was collected. Also, new reports and metrics were added to the dashboard according to the needs that emerged.

4. Demonstration, Evaluation & Communication

The utility, quality, and efficacy of a design artifact must be demonstrated via well-executed evaluation methods. It is also important to evaluate how the artifact supports the solution of the problem and to communicate the problem and its importance to researchers and other audiences. According to Hevner (2004), the research must be presented effectively both to technology-oriented as well as management-oriented audiences. (Hevner 2004, Peffer et al. 2007)

In this research, the dashboard should be presented to the target users and potentially specific training could be organized. Also, the need for additional documentation needs to be considered. The evaluation of the rate at which the artifact solves the problems will be done by surveying the state before the BI solution and the state after the target group has been using the solution for a specific duration of time.

3.2 Case company

The case company of this study is a large international software company offering information technology products and related services. The company comprises various separate units serving customers from different segments. This study takes place in one of the many business units of the case company. The business unit has arranged its operations according to the Scaled Agile Framework. This means that solution trains and agile release trains (ART) as concepts are used to give direction for product development. Solution train is an organizational construct that aligns the ARTs with a shared mission. Figure 16 illustrates the organizational overview and the units within it.

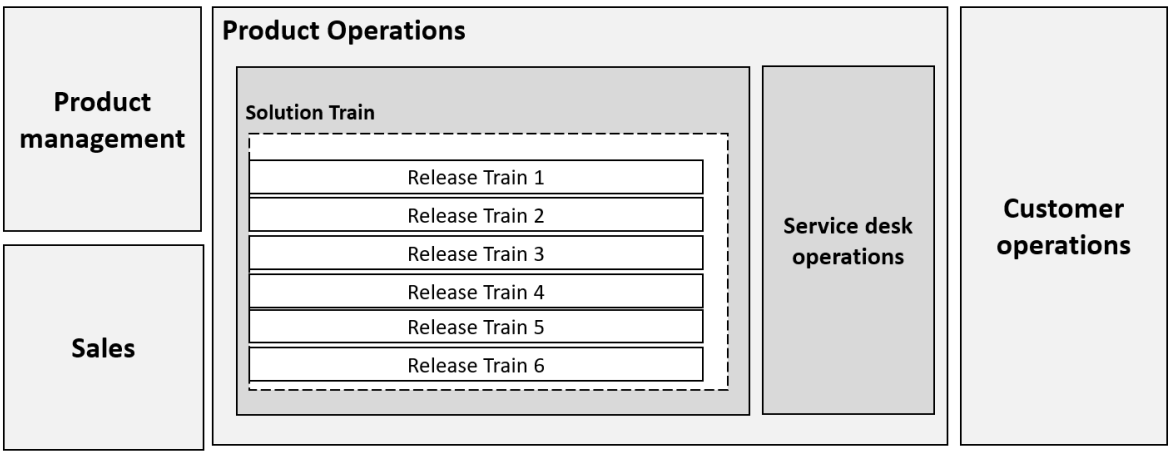

Figure 16: Simplified organizational structure

Product operations is the organizational function responsible for the development of the solutions. Within the function, there is one solution train responsible for the shared services and deployment work related to the ARTs. The ARTs align teams with a shared business and technology mission and are responsible for developing and delivering the solutions. The ARTs are constructed in relation to the product families of the BU. This provides them with the responsibility of the related product deliveries.

Other organizational functions within the BU are product management, sales, and customer operations. The focus of this study is primarily on product operations, but also to some extent in product management. The other functions are not within the primary scope of this study. However, all the functions work in dense cooperation so a single function can't be entirely separated out of the context.

Main stakeholders of the study

The main stakeholders of this study are managers with program management responsibility for product development. This study was ordered by managers working within the product operations organization. The primary target group of this study includes the users of the dashboard which are release train managers (RTM) and solution train managers (STM) within the organization. There are also other stakeholders indirectly participating in this study most importantly product management, but also customer operations and sales to some extent.

The main responsibility of RTM is to facilitate and enable end-to-end value delivery through orchestrating the release trains. Some of the more specific tasks include organizing and facilitating the release and program increment (PI) planning and execution. They also follow-up on the overall release train progress and ensure the release train maintains the schedule. This means that the milestones have been met within the predefined criteria. RTMs also have the responsibility of the risk management related to the release and they ensure the quality and maintaining of the budget. RTMs also work in close co-operation with other organizational stakeholders such as product management, area product owners, operative product owner, and other stakeholders regarding release planning, strategy, and execution alignment.

STM has the operative responsibility of product operations organization. This contains defining how to reach the requirements that product management sets for product operations. The more specific responsibilities include ensuring the delivery of releases with production quality and committed schedule and content. STM also ensures synchronization of development activities with key stakeholders by steering and guiding the operative work of solution train and release trains as well as together with process owners developing the ways of working. STM also participates in the strategic planning work of the organization as a member of the leadership team.

3.3 Data collection

The data used in this study were collected with a survey and interviews. These research methods were selected to gain an extensive amount of data for the design work. Without the survey, it wouldn't have been possible to get in-depth insights with the interviews. The survey contained open-ended questions and Likert-scaled questions (see appendix 1). The main motivation for the survey was to gain numerical data to be used in analyzing the starting point of the development process and to enable the measurement of the level of improvement in the future. Also, the open-ended questions used in the survey gave information to be utilized in formulating the actual interview questions for the target group.

The interviews were conducted as semi-structured. This means that a comprehensive list of questions was prepared beforehand but there was still room for improvisation. All the planned questions were discussed with all the interviewees, but some additional topics were discussed from outside the script. For instance, the interviewees were asked to present the reports they follow regularly. All the interviews were conducted individually and organized via Teams due to the current Coronavirus situation. The interviews were arranged between the 10th and 16th of November 2020. Four of the interviews were held in Finnish and one in English. The time reserved for each of the interviews was 60 minutes and the interviews were recorded to analyze the answers afterward.

The participants in the survey and the interviews were the same. They were selected to get full coverage of the target group of this study and were the ones with the direct program management responsibility in the case company. This means that all the primary users of the designed BI dashboard were involved. This resulted in a total of 5 persons involved in the process, 4 of them participating the survey and all in the interviews. Four of them were release train managers and one was a solution train manager. Each of the target group had been working from 1 to 5 years in their current position. So, they all had accumulated expertise to their roles and responsibilities. They also had a rather long history from various other positions within the industry.

In addition to the survey and interviews, various documents and company internal web pages were used as a data source in this study. For instance, information about internal processes and performance reports used by the target users were analyzed.

4 CURRENT STATE ANALYSIS

In this chapter, the current state of the case company is discussed based on the empirical research of this study. The main topics in this chapter are program management and performance management in the agile context. Performance management is scrutinized primarily concerning the objectives, metrics, and reporting practices associated with it. The chapter starts with an overview of the results of the survey that was conducted as a starting point of empirical research.

4.1 Overview to survey results

An online questionnaire was organized for the attending employees as a starting point of this study. The motivation for the survey was to gain preliminary insights into the current state of the program management practices, especially in the reporting dimension. The survey contained both Likert-scaled questions and open-ended questions. The Likert questions were used to collect numerical data of the defined areas and the open-ended questions were used to gain more practical information about the discussed topics.

First, the respondents weekly and monthly time used to create reports was evaluated. This data was planned to be used in assessing the results of this study in the form of time saved by using a BI dashboard instead of manually generating the reports. Three of the respondents answered to be spending more than one but less than five hours for report creation on weekly basis and one from five to ten hours. On monthly basis, there was more diversification in the answers as the time used varied from 1 to 5 hours to more than 10 hours. Hence, it could be stated that the time used for report creation is significant and all the respondents could benefit if the time used would be reduced. I.e. they could focus on more value-adding activities.

Within the survey questionnaire, the respondents were asked questions in four primary categories:

1. Program reporting process,
2. Data,
3. Metrics and KPIs,
4. Business intelligence usage

The respondent's answers related to the reporting process, data, and metrics and KPIs are illustrated in figure 17 below. The answers are scaled from 1 to 5 determining the extent to which the respondents agreed with the questions asked. In the figure, the data labels are colored green, yellow, or red based on the average of the answers. The answers below or equal to 3 are marked as red meaning that they are rather alarming and require further attention. The issues that achieved a score higher than 4 are marked as green and the issues that were scored between 3 and 4 are marked as yellow. Questions 7 and 10 are rather inconsistent due to the inverse question format and therefore assessed reversely as red.

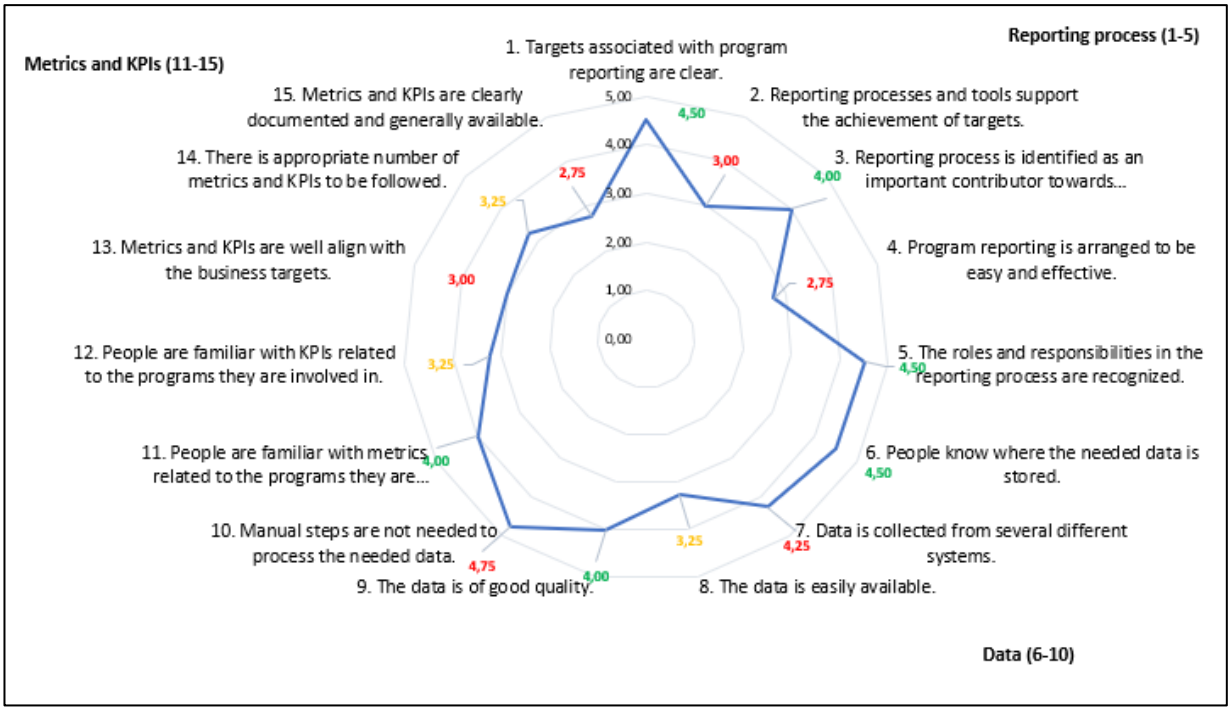

Figure 17: Averages of questionnaire answers

Reporting process

The flow of the program reporting process and the target group's sentiment with program reporting were surveyed through questions 1 to 5. The results indicate that on average the respondents are well familiar (4,5) with the targets associated with program reporting and they recognize the roles and responsibilities in the process (4,50). The reporting process was also recognized as an important contributor to successful product delivery (4,0). This sets the requirement for systematic and effective reporting practices.

However, the respondents do not think that the current reporting processes and tools support the achievement of the targets to the required extent (3,0). Also, the respondents feel like program reporting is not arranged to be as easy and effective as it should be (2,75). This supports the motivation for new solutions required for this purpose. One respondent also mentioned that the reporting is all handwork and hence a way to get data directly and easily from the tools is needed.

Data

The availability and quality of the data were surveyed through questions 6 to 10. It appears that the respondents are relatively well familiar with where the needed data is stored (4,5) but the issue is that the data is not always easily available (3,25). It is also more a rule than an exception that manual steps are needed to process the needed data (4,75). One respondent also emphasized the fact that manual data-related work takes a lot of effort. The fact that the data must be collected from several different systems (4,25) also increases the motivation for a system that does the collection work automatically or for a system where the data is stored collectively.

The respondents feel like the quality of data starts to be at a pretty good level already (4,0), however, one respondent mentioned that there is still struggle with having a coherent level of entering the data into systems. Also, the definitions are not followed well enough which results in a situation that the data is not always telling the whole truth. Therefore, manual polling work is still needed to get everyone to keep the data up to date to get the operative picture correctly out of the systems.

Metrics and KPIs

The metrics and KPIs related to program management were surveyed through questions 11 to 15. The results indicate that the respondents are better familiar with the metrics (4,0) of the programs than they are with the KPIs (3,25). However, this is defined by the fact that one respondent mentioned not to be sure what is meant with KPI. This raises the importance of communication and target setting at the company level.

The results show that respondents don’t agree nor disagree (3,0) that the metrics and KPIs are well aligned with the business targets associated with program reporting. Nor there is a very high level of agreement that the number of metrics and KPIs is exactly what it

should be (3,25). Regardless, the most alarming issue related to the metrics is the generally available documentation of the metric and KPI definitions (2,75).

Business intelligence usage

The respondents' sentiment towards business intelligence utilization was surveyed through five Likert-scaled questions and two open questions. The survey indicates that there is room for improvement within the understanding of the business intelligence concept (3,0). Also, only one of the respondents had previous experience using business intelligence in the current position. This shows that BI solutions are not regularly used for the daily reporting practices within the target group.

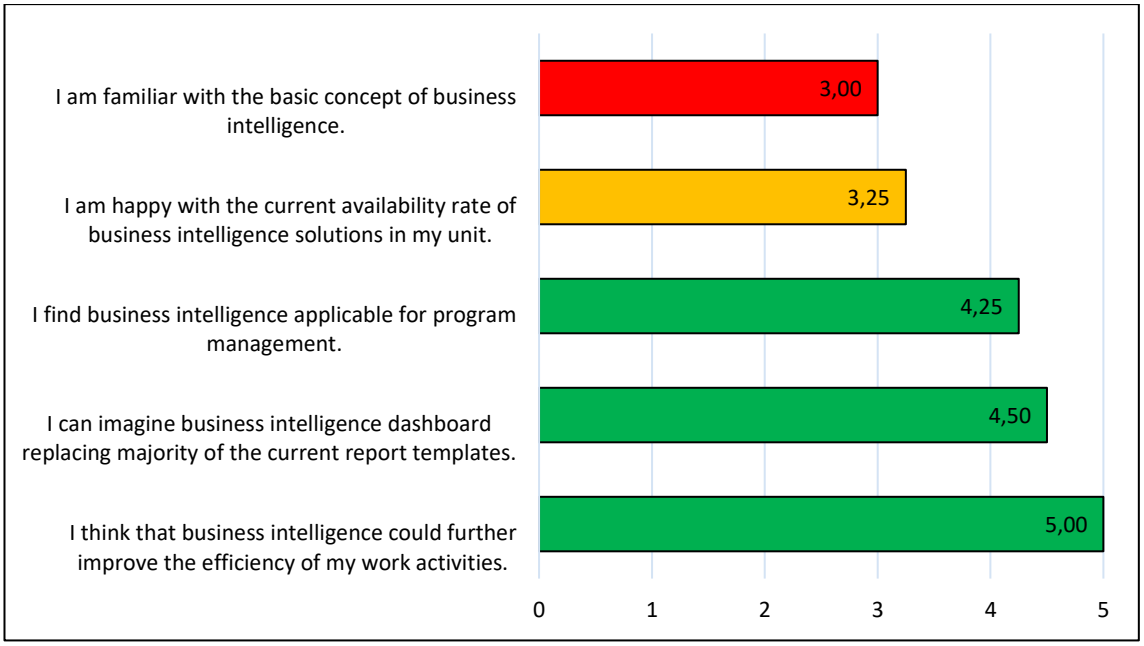

Figure 18: The business intelligence maturity rate

Important for this study is that majority of the respondents found BI highly applicable for program management (4,25). However, the availability rate of BI solutions within the unit is not at a required level since the respondents indicated that they are not highly satisfied (3,25) with the current availability rate of them. More importantly, all the respondents strongly agreed (5,0) that new business intelligence solutions could improve the efficiency of their work activities. This means that the interest in new BI solutions is very high among the target group and that there is room for improvement with the BI utilization.

The sentiment indicates (4,50) that the target group can imagine a new BI dashboard replacing the majority of the current report templates used within the unit. One of the respondents highlighted that they do get business data from the systems but that the current toolset to process it is quite limited and it requires manual work to get good, aggregated views of the data. This means that they are following more the different data sources as such but not combining too much the data to gain insights into the phenomena indicating how the business is going.

In general, the respondents found BI highly applicable for program management and noted that it could strongly improve the efficiency of their work activities. The respondents primarily expected BI to improve and give help with reporting and decision making. It is also expected that it could replace some of the existing reporting templates and reduce the manual work needed to create the reports. Also, they expected that the new solutions should be able to better utilize the historical behavior and time reporting data for building the operative picture and to improve the predictability of the operations. Also, there appears to be a need to better utilize customer satisfaction, the customer felt quality, and customer behavior in the product operations and management activities.

Conclusions of the survey answers

The survey was conducted to measure the current reporting practices of the case company. The motivation for this was to gain insights into the prerequisites for the BI solution implementation. The most important issue that arose from the survey was the inefficiency of the current reporting practices. There is room to improve the current reporting practices with the BI solution. The results also indicate that the current reporting process and especially the creation of the reports is a rather time-consuming activity among the target users. Each of the respondents stated to spend multiple hours weekly and monthly for the reporting activities. Especially the amount of manual work required in this process should be reduced.

The survey indicates that there is no high consensus that metrics are aligned with the business targets associated with program reporting, which raises the question of whether the metrics have been linked to the business targets. This should be confirmed to give business reasons and motivation for performance measurement. Also, specific target levels for each metric should be set to know what is expected to be achieved with the

measurements. The survey implied that the documentation of the metrics and KPIs should also be improved by defining the metrics and making them commonly available.

There is data available within the unit to make greater use of business intelligence in the operative dimension. Furthermore, the quality of the data is already at a pretty good level so the problem is not that where the data is stored but the fact that it takes a lot of manual effort to collect the data and to process it into the reports. It also appears that the target group has high expectations for business intelligence usage even though they are not very experienced and familiar with it. To solve this issue, in addition to implementing the BI solution, in-depth user training regarding its usage is needed.

4.2 Agile program management

In this section, the program management practices of the case company are discussed. Especially the program concept and the way programs are recognized within the case company are emphasized. Other essential topics are the objectives of the programs and the practices related to achieving the objectives.

4.2.1 Program concept

The program concept within the case organization is highly comparable to the program level of the SAFe. This means that agile release trains (ART) are the main constructs in it, which also align the product development with a direction. ART is identified as a continuous function that organizes the teams working towards one or more initiatives. The main responsibilities of the ART include the estimation work of defining requirements for the teams and tracking the extent to which the committed actions are achieved during an increment. Also, budgeting and ensuring the committed schedule and quality are key tasks of a release train.

Initiatives are created concerning the main product releases. The releases are targeted to be launched on a yearly basis which means that the duration of the initiatives is limited to the duration of the release. The interviewees also identified the initiatives as a type of program. The initiatives are the largest bodies of work that are first divided into epics by the product management organization.

Solution train as a concept was recently taken into use within the case company to orchestrate and coordinate the work performed by the release trains. Basically, the main product is defined in the solution level and the solution train coordinates the work of the release trains and brings together the releases. The interviewed STM identified the solution train as a program that puts together the smaller programs formed by the release trains. The ST also communicates the issues related to the RTs upwards while the RTs primarily work with the product management.

In practice, the interviewees recognized programs as continuous processes that align the product development work of one product family. Within a product family, multiple products are developed through the programs, which also means that there are products that may be in different lifecycle phases. Some of those are in new product development, meaning that they are all new investments and entirely new to the market while others are in maintenance mode or nearing the end of their lifecycle. One interviewed RTM recognized that around half of the products of the train are in maintenance mode, meaning that one important area of work is the bug fixes.

Largely the products of the case company are divided into two: independent products and products that are part of the main solution. The release trains are arranged to develop specific items for the main solution. However, the extent to which the major solution employs the release trains varies. Some of the ARTs focus more on independent products while the majority of them focus almost entirely on the main solution. The independent products can commonly be integrated into the main solution but also to the other solutions in the market.

4.2.2 Program objectives

The main objective of a program within the case company is to ensure that the outputs of the product development are of the right quality in order to bring value to the customers. This means that the releases are completed in time and in a form that satisfies the quality requirements associated with them. The requirements for the programs are provided by the product families that also have their own unique objectives, which are reflected in the ones of the release train. The release trains develop their own product families, which can be sold as individual releases but most commonly are part of a larger solution that is the main product of the unit. One interviewee recognized the releases as intermediate targets for the programs

Product management defines the targets and requirements for the contents of the releases and communicates them to the product operations. Product operations then gives estimation for the required work and gives commitment to feasible items and splits them into features. The implementation of the features is planned in increments. The teams are responsible for splitting the features into user stories that can be completed during a sprint. Hence, they are responsible for doing the hands-on development work of the release train. In this context, the work performed by the agile development teams can be analyzed in the form of projects. The programs supply teams with the features that are to be developed during the pre-defined duration of increments and sprints.

Many of the interviewees found the objectives of the main solution often to be rather unrealistic meaning that all the objectives can't be reached. Also, it was emphasized that the objectives are very strict and that the risks are not always considered to a required extent. The most decisive objective is the schedule, and, sometimes it is needed to compromise with the contents and quality of the product to achieve the schedule-related objectives. On the other hand, the objectives of the independent products were found more flexible than with the main product release. Commonly they are more customer-oriented meaning that specific features are sold to the customer and that there are no strict schedule requirements associated with them.

4.2.3 Product increment planning

The product development work of the case company is organized through product increments (PI). The purpose of the PI planning is to ensure that product development work is focusing on the right things according to the prioritization of product management. The goal of PI planning is to have a controlled working environment where teams can work without sudden surprises and work only with the backlog that brings business value.

The PI planning process starts four weeks before the new increment. First, the themes and targets for future releases are planned and the targets are split for the next increments. This contains defining and prioritizing epics and features based on the targets. During the process, product development works in close cooperation with product management. The epics are defined by product management and product development splits them into features that can be developed during an increment. Based on this, sprint backlogs, risks, and dependencies are created.

Only “content” work items, meaning initiatives, epics, and features are handled via the PI planning process. However, features will be split into user stories as part of the continuous planning process latest before the sprint they are taken into work. This means that a user story split of the features is recommended during the PI planning process. The continuous planning also contains the continuous scanning of markets, customer wishes, and product backlog prioritization done by product management.

The duration of the actual increment is six weeks, and it is divided into three sprints of two-week length development cycles. The features are supposed to be developed during one increment, but epics may last for a couple of increments. The schedule for the product increment planning and development is illustrated in figure 19.

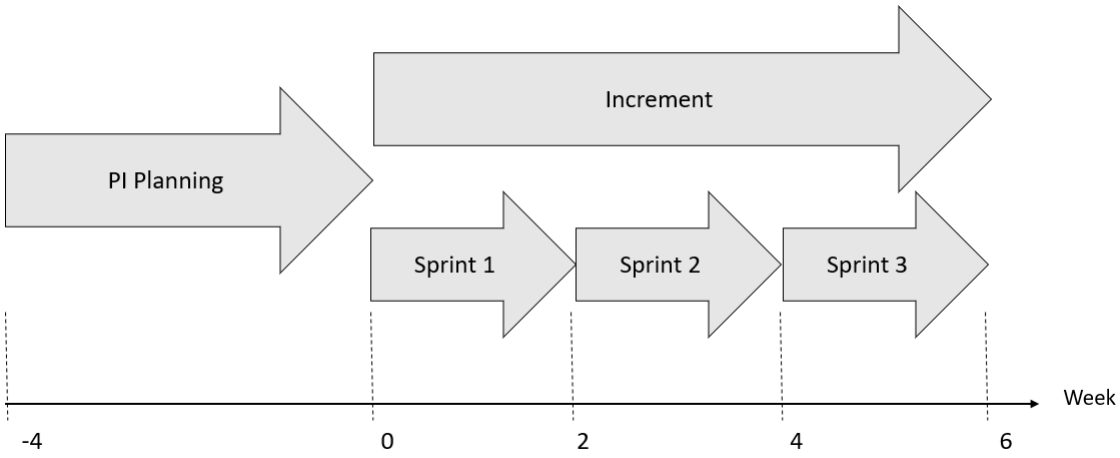

Figure 19: Increment and sprint schedule

One interviewee recognized that some of the product development carried out in the case company contains the development of products that are continuously produced through sprints and increments. Considering this type of product, product owners and development teams share the responsibility of creating work items that can be delivered during the sprints.

4.3 Performance management in agile programs

In this study, performance management is recognized as a process of creating a context for performance measurement. It covers the methods, metrics, processes, and tools that are used for managing performance. In this chapter, these elements are discussed in the

context of the case company and its programs. First, the reporting practices are reviewed since they are a prerequisite in ensuring the organization achieves the objectives set for the programs. Also, the tools and systems used in the activities related to managing the performance of the programs are demonstrated. However, the biggest weight is put on the performance metrics used in the agile programs due to their importance for the scope of this study.

4.3.1 Performance reporting

Performance reporting can be identified to create the context for performance measurement. Within the case company, the primary reason for performance reporting is to ensure that the products are developed based on the requirements and that the targets associated with them have been met. RTMs work in close cooperation with product management and execute reporting activities almost on a daily basis. Release trains also have a responsibility of keeping the solution level informed about what is going on within the release trains. However, this kind of reporting is rather informal and conducted primarily verbally three times a week.

The most formal reporting stream occurs between the trains and the head of the organization. This kind of reporting is conducted monthly and it contains reporting of the most important development related metrics. This is called business review or product development monthly report and it contains reporting of the overall performance of the trains, but especially the performance of the individual teams. The interviewed STM also participates in the leadership team's meeting approximately on a weekly basis to discuss the risks associated with the programs.

The program reporting is arranged at the release train level at least bimonthly. However, smaller monitoring is conducted for products and teams irregularly. The progress of the release trains is reported to product management and release managers in weekly key initiative follow-up calls and biweekly main release follow-up meetings. Another irregular reporting is needed e.g. for decision milestones or for the products that are under special monitoring. One of the interviewed RTMs concluded to participate in various reporting activities approximately twice a week.

There is also reporting tied with the increment and sprint schedule. Once every two weeks, the sprints outcome and the operative picture of the trains is analyzed internally

within the train. Once the increment ends, an increment review is organized in which the overall performance of the increment is scrutinized and especially the completed features and epics are reviewed. The outcome of the increments is also reported to product management. In this process, also the expectations for the next increment are planned.

4.3.2 Performance measurement

The research indicated that the main motivation for using performance measurement in program management includes the need from management, ensuring quality and schedule and to follow the progress and trend of the organizational development against the agreed goals. For instance, the lifecycle of work items and velocity of iterations is followed to understand the rate at which the "machine" works. The financial metrics and resources are followed against the operative planning including the budgets that have been made. Also, the timely organizational development goals like optimizing the customer deliveries, ticket handlings, and increase of test automation level are being followed. The basic work item quality-related indicators are also followed closely to be able to rely on the data.

The target group had varying opinions on which metrics have the highest level of business importance. Figure 20 illustrates the metrics that the interviewees mentioned to be regularly following in their programs. The stacked bar chart shows in grey the times the metrics were mentioned and in red the count the metrics were identified among the most important metrics. As a conclusion can be identified that cycle time and customer bugs were identified as the number one important metrics in the programs. The customer bugs were noted to be important since they are one of the few metrics showing customer satisfaction and due to their direct indication of how the product is performing. However, it seems that there is a rather high level of consensus of the metrics among the interviewees.

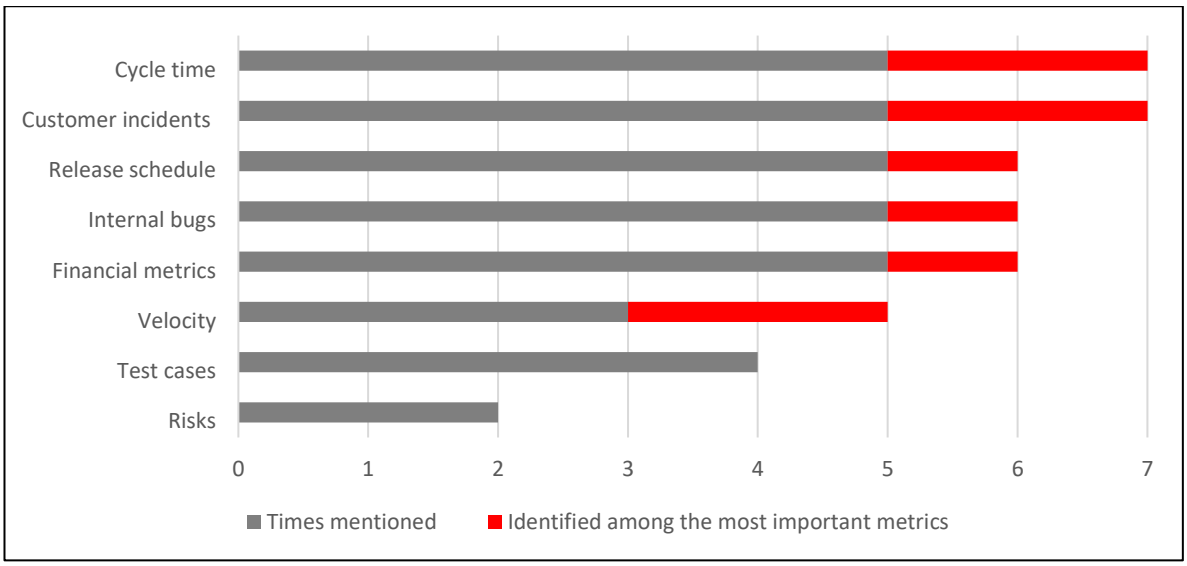

Figure 20: Performance metrics in the case company

Financial metrics, release schedule, and internal bugs are also followed by all of the interviewees and one of the interviewees identified them among the most important metrics. Surprisingly, only three of the interviewees mentioned to be regularly following the velocity metric, but two of those recognized it among the most important metrics. However, it appears that most of the interviewees are following it alongside the cycle time metric, so they possibly forgot to mention it. In general, the cycle time and velocity were recognized to have importance from a business perspective since they increase the predictability of work and gives an understanding of whether there is room for improvement.

Figure 20 also shows that majority of the interviewees are following the test cases as a metric. However, none of them identified it among the most important metrics. Another under-represented metric in the graph is risks. Only two of the interviewees mentioned to regularly follow risks and neither of those directly categorized them as a metric. In the following sections, the identified performance metrics are discussed more in-depth.

Cycle time

By definition, cycle time is defined as the amount of time a team spends working on producing an item. Within the case company, the objective for cycle time is to indicate whether the planned features and user stories can be completed during the sprints and increments. There is a target associated with the cycle time that the features should be

completed within 6 weeks and the stories within 2 weeks. One of the interviewees noted the trend to be that the sprints are commonly well predictable but that there is still to improve with the features. It was also suggested that after the targets with the features have been achieved the focus could be shifted towards the cycle times of epics. However, this would mainly indicate the planning capability of the product management unit.

Figure 21: Example of a cycle time graph

In the case company, cycle time is measured as the days it takes for the work items from when they are started to the moment they are marked as completed. The cycle time is calculated for each of the completed work items and the data is used to create a line graph of the data. Also, the visualization shows the average cycle time for all the work items during the defined timeframe. The cycle time visualization also indicates the number of specific work items completed during the selected time. There is a need to be able to filter the different work items to select which work items are to be included in the report. The alternatives for the work items are features, user stories, and bugs.

Release schedule

The major program objective is to ensure that the release schedule is being met. The main solution of the case unit is launched as a yearly release which also defines the primary schedule of one year. However, within the releases, the SAFe structure is identified as a key driver in ensuring the schedule since it sets the durations for increments and sprints. Within the case unit, the work is planned for increments and sprints and it is on release

trains' responsibility to ensure that the schedule is met for those. In cases of slippage, release trains aim at mitigating the damages caused by the slippage of the schedule. One interviewee mentioned that the work is organized based on defined priorities. The best estimates for the duration of the work are given and work is organized based on those.

One interviewee emphasized that there has been a will to move towards a partnership model that would better support agile practices. This would bring additional value to the customer organizations and help product development to optimize the activities. The current model doesn't support agile practices to the desired extent due to the customers' limited capability to accept new releases since the majority of the deliveries are made on-premises which causes interruption to their daily activities. This is forcing the case unit to create plans for a year off. Such long-term schedule related objectives increase the risk of slipping out of the schedule and hinders the incremental deployment. The product development is capable of incremental releases, but the customer base and business model don't support the agile release model to a required extent which causes contradictions.

Financial metrics

The financial objectives of the programs are highly associated with the planned budget and the actual costs caused by the development work. All of the interviewees recognized budget and costs matters among the most important objectives for a program. The budget for the products is planned as an estimation of the required workload for the product development effort which is then requested from the upper management. The budget is typically defined on a yearly basis and followed monthly in comparison to the actual costs.

In the last resort, upper management is responsible for deciding the yearly budget and allocating it to the programs. Release train managers follow closely the actual costs that have been realized during product development. The actual costs are compared to the planned costs by products in periods that can be defined on a monthly or year to date basis. Some of the interviewees mentioned that comparing the forecasted budget to the actual budget shows where the investment is going.

The common financial metrics reported at the program level are total budget and realized budget. Commonly, there is a need to analyze the difference between the estimated and realized budget. Some release trains also track the FTE count parallel to the budget data.

It appears that there isn't a standardized way for financial reporting since the data is manually retrieved from ERP for each reporting need. Also, some release trains do financial reporting as a yearly budget and some as a monthly budget. This refers to year-to-date (YTD) or period-to-date (PTD), which in the case company is tracked in monthly periods.

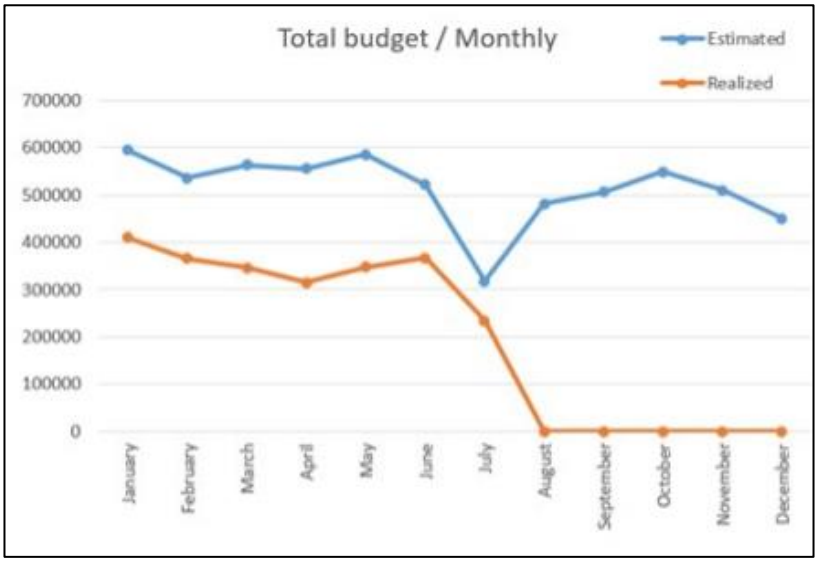

Figure 22: Example of a financial graph

Velocity

By definition, velocity is the measurement of the quantity of the work the team delivers per unit of time. It can be used for measuring the productivity of software development. In the case company, it is measured as the count of work items completed during a sprint and measured primarily on team level but also on release train level. Velocity is used to indicate the count of stories closed during a sprint and features during increments. In the case company, it is shown in four categories: planned, completed, completed late, and incomplete. The visualization also shows the average velocity for the work items.

The interviewees highlighted that the velocity is best used to track the extent to which the planned and committed items can be completed. Especially since the teams have the freedom to choose what they do in the sprints, so it indicates how well the team has achieved to complete the items it has planned to. This is important in order to be able to rely on the teams' planning capability in the long term.

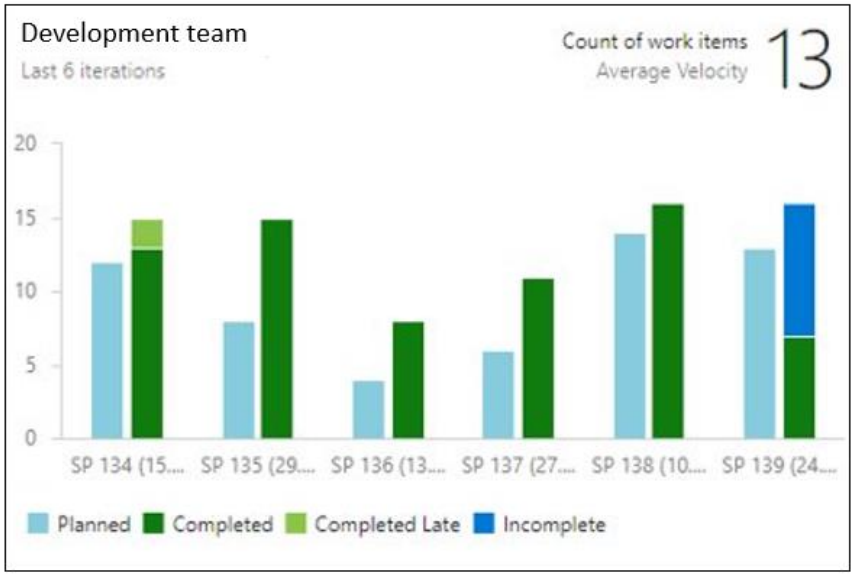

Figure 23: Example of a velocity graph

However, it was noted it is common that teams plan different kinds of stories, so it is difficult to generalize the velocity measures. Still, the trend of the velocity metric can be used to indicate show the development of the organizational performance if historical data is used.

Customer incidents and bugs

The quality of the software products is defined as the count of customer incidents and bugs. Customer bugs emerge from the incidents made by the customer which can also be recognized as post-release defects. The most important quality-related objective is to minimize the amount of customer reported incidents that arise when customers are having issues with the products and their functionality. However, the majority of the customer incidents can be resolved without a need to make changes to the software. Such incidents may be e.g. related to the configuration or infrastructure of the software. Therefore, incidents may be directly or indirectly related to the software.

The number of customer incidents is the most important measure of product quality. It indicates the overall quality of the product releases by showing e.g. the usability of the products. Commonly prioritization related to the product quality is made towards new products instead of the products that are closer to the end of the lifecycle. The majority of the bugs are identified internally via an automatic testing process, but it is also common

that there are some bugs identified after the system has been delivered to the customer. In these cases, customers create bug tickets via ServiceNow, from where they are communicated to the product development via the internal service desk.

The number of incidents is also analyzed by products and the products with the most incidents typically require actions and are taken into more intense monitoring. However, one interviewee highlighted that it is difficult to make a comparison between the products since the products may be much different from each other for instance considering the size of the product, customer segment, and the functionality of the product.

The bugs can indicate what has been tested and what requires further action. Also, the count of bugs can be analyzed to see how the trends progress over time. The internal bugs are identified via an internal test pipeline containing both manual and automatized testing.

Test cases

Test metrics are a commonly used metric, especially within the release trains. On the release train level, the count of completed test cases, automated test cases, and manual test cases are tracked. One metric commonly used within the case company is the count of test automation test cases, which is primarily measured and analyzed on a monthly basis. The interviewees highlighted that the count can be used to indicate how testing is being done and how it is improving. The analysis is made primarily on the testing performance of individual teams. This is done to identify the teams that are performing well or whether some team is having issues.

Both manual and automatized testing are performed within the case company and the major objective related to it is to increase the automatic testing and its coverage. However, there is currently no generally recognized way to measure it and the number of test cases is rather used to analyze the trend of the testing. The understanding of the test readiness and coverage is acquired by tracking the automated test cases are against a specific epic or initiative.

Testing related metrics were also identified to be rather difficult to measure since it requires a lot of manual digging from the systems. Especially it appears that there is a lot of manual work required to compile the overall status of the performance of the testing process. It was also pointed out that it is an important issue to be measured since it helps to stay on track with the quality of the work. Furthermore, it was mentioned that the

current test metrics don’t indicate the overall performance or security aspect of the release quality.

Risks

The main scope of risk assessment within the case company is related to the software products and services. It is the shared responsibility of the product development teams and product management to identify, analyze, and treat risks. The risks are commonly identified to affect the releases The case company uses a qualitative method for risk analysis for both severity and probability of the risks. The impacts are assessed on a scale from 1 to 3. Within severity assessment, the analysis is made based on the impacts on confidentiality, integrity, and availability. The probability of the risk occurrence is assessed based on how probable the risk is to realize. In this assessment, 1 means that the risk is low and improbable and 3 that the probability is high, and the risk is likely to happen.

The magnitude of the risks is calculated based on the product of the probability and severity calculation. If the associated value of severity and probability is over 5 or if the probability is high, the risks are not acceptable. For those risks that cannot be accepted there is a need to modify the risk that is made by applying controls that usually lower the risk. The selected control must be such that after the control is applied the residual risk is on an acceptable level. The decision to accept product-related risks shall be made after the evaluation of applied controls during the risk treatment.

One interviewee emphasized that there is a need to better acknowledge the risks with the main releases. Due to the long-term objectives of the releases, there is a rather high-risk level associated with achieving the defined objectives. There has been lacking resiliency with the risks and especially readiness to compromise within the schedule when it seems like the targeted schedule will not be achieved. It was mentioned that there is a high-risk level associated with the schedules and the risks have sometimes been realized due to insufficient risk management.

Probability \ Severity	Low (1)	Medium (2)	High(3)
Low (1)	1	2	3
Medium (2)	2	4	6
High (3)	3	6	9

Figure 24: Risk matrix used in the case company

4.3.3 Tools and systems

The tools and systems used in performance management contain the ones used for data storage and for processing the data. The main data storages are Enterprise Resource Planning (ERP) system, Azure DevOps (ADO), and increasingly also ServiceNow. ERP contains the data related to budget, cost, and resources and ADO is used for practically all the data related to product development. The ServiceNow is used for the customer incidents and its use is planned to be increased during the next year also in release train level. Also, internal websites are used for specific needs.

The primary tools used for data processing and presentation are Excel, PowerPoint and ADO. Especially, ADO queries are used. There are existing templates that are modified based on reporting needs. For instance, the data from ERP is acquired to excel and some processing activities are required to get the data into the desired format. PowerPoint is generally used for giving presentations of the data. Also, in some cases, direct reporting from ADO may be given and there are some dashboards integrated with the ADO tool that is commonly used for reporting.

The operative practices of the case company are experienced to be quite informal and not taking much additional effort. Lately, they have succeeded to keep the tools well up-to-date and that has helped a lot with building the operative picture and reporting from different stakeholders.

4.4 Challenges and improvement proposals

In this section, the challenges that were identified during the survey and interviews are discussed. These are categorized to the performance measurement related challenges and reporting related challenges. Also, improvement proposals are discussed based on the conclusions made out of the current state analysis.

4.4.1 Challenges with performance measurement

The existing performance measurement practices of the organization split the opinions of the interviewees. Some were rather satisfied with the metrics and measurement practices but some implied directly that the metrics are not at the level where they should be at. Also, one interviewee highlighted that the targets for the metrics and measurement have not been defined for this ongoing year. Therefore, there is no understanding of the target levels for some of the metrics. In general, the interviewees also found that there is room to improve with defining the objectives for the programs. For instance, it was stated that the defined objectives of the programs may differ based on the technical expertise of the product manager. Without a high enough understanding of the technical requirements, the expectations may not be very realistic.

The negative opinions on the performance metrics were defined by the factor that the measurement is made solely based on the data that can be easily accessed. One interviewee mentioned that the metrics don't always indicate what is being done and what could be improved. One major issue with the current measurement practices is the accuracy of the measurements. For instance, it was identified that it is possible to distort the measures e.g. for the cycle times by using faulty starting dates for the work related to the user stories and features. By doing so, the measures can be modified to show better performance than it is.

Several of the interviewees recognized testing as a challenging area for measurement. It was expressed that the current practices for testing related measurements are rather confusing and the capabilities for it insufficient. They are expected to measure the count of test automation cases with a target of continuously increasing the number of automated test cases. It was also mentioned that it is only possible to take snapshots of the data which means that it is not possible to analyze the trends related to the testing performance. Furthermore, it was criticized that this measure can't be compared between each of the

teams and release trains because of their different products and lifecycle phases. They have also been requested to manually document the automated test cases in the databases, but this has been seen as unnecessary among the interviewees.

Another thing that is causing challenges for measurement is the customer feedback side. For instance, the customer incidents are not linked to releases and they are sometimes communicated as issues that don't appear on backlogs meaning that they can't be prioritized. One interviewee suggested that the quality of the products could be measured based on the usage analytics. There has been a strong will to measure the usability of the products in this way, but the current products don't have the technical functionality for it. This is a thing that limits performance measurement to some extent. This could be enabled by including analytics within the products but until today this has been lacking.

4.4.2 Challenges with performance reporting

The primary performance reporting related challenges originate to the lack of capabilities within the current tools. For instance, the current tools don't support the need of analyzing the product development data in comparison to the time factor. One interviewee emphasized that there is a need to analyze the data in periods or from the beginning of the year. However, there is no reasonable way of doing this kind of analysis or it requires a ridiculous amount of manual effort. In general, the manual effort required for the report creation was commonly identified as a major issue. Especially, several interviewees emphasized that the process of acquiring data from the ERP takes a significant amount of time.

The ADO tool was also recognized as a major bottleneck within the reporting practices. The case company relies on the tool to a very high extent since it is used for storing practically all the product development related data of the organization. Furthermore, the tool is not solely used for data storage purposes since it is also used for analysis and reporting purposes which put a heavy load on it. This causes significant performance-related requirements for the tool. Also, the currently used visualizations, etc. related to the product development data are based on what can be gotten out of the tool.

One interviewee pointed out that one problem with the current reporting tools is that the information they show may already be outdated since things may change quickly and their refresh intervals may be rather long. Also, it was hoped to get rid of reporting as a

practice since with the reporting that is done e.g. on a monthly it may be too late to react to some critical things within the products. A better alternative would be to analyze the data in real-time with the tools and improve the timeliness of the decisions.

In general, there is plenty of data available to be utilized in reporting but there are no means of taking advantage of the it at the desired level. One interviewee deplored the fact that there is data within reach but much of the potential remains unused. Also, one challenge with the current reporting practices is to combine data from multiple sources. The current systems for reporting make it difficult to combine the data sources and compels to jump from system to system when collecting data and presenting it.

4.4.3 Improvement proposals

The interviewees emphasized that they are not using BI for any product development related reporting. They recognized that the use of BI would reduce the need of performing additional and manual reporting. One interviewee hoped that BI would enable better use of historical data in forecasting. The current dashboards used in the case company show the snapshot of the data meaning that the analysis of the historical data and creation of estimations of future behavior is not possible. BI was also expected to be highly beneficial in creating trendlines of the data since the current reporting solutions don't have that capability.

One interviewee set a target state for BI usage to be that there would be a reporting system that is so configurable that every stakeholder group could use it to gain a real-time view of all the key metrics relevant to one's role. This kind of solution would then eliminate the need of sitting all day long in meetings showing the metrics from PowerPoint and Excel. This would also end up in significant savings yearly when there is no need to manually compile these reports and spend time giving the presentations.

In table 4 below, the key takeaways from the current state analysis are summarized. Proposals are made based on the conclusions made out of the survey but also based on the challenges discussed in this section. They are also categorized to direct actions and indirect opportunities. Direct actions are something that can be rather easily fixed but the indirect opportunities can be achieved only after the direct actions have been made.

Proposal	Description	
Documentation of the metrics	The metrics and measurement practices should be documented and made generally available to increase the understanding around them.	Direct actions
Align metrics with business objectives	The metrics should be linked to the business targets to give a business reason why specific metrics are used.	
Define targets for performance measurement and KPIs	The target values for performance measurements should be defined to increase the understanding of the purposes of the metrics.	
Improve the data utilization rate	Plenty of data remains unused. New solutions are needed to improve the utilization rate to gain new insights into the business.	
Increase the availability of BI solutions	The current availability rate of BI solutions was discovered to be insufficient. There is a need for new solutions to be used in the management of the programs.	
Improve the efficiency of the reporting practices	The current reporting practices and tools were not identified to be as easy and effective as they should be.	Indirect opportunities
Reduce the time used for manual reporting practices	The time used for reporting practices at a weekly and monthly level should be reduced. Currently, the amount of required handwork for reporting practices is notable.	
Increase understanding of business intelligence	The understanding of the business intelligence concept and practices related to it should be increased among the target group.	

Table 4: Improvement proposals based on the current state analysis

5 DISCUSSION

This chapter combines empirical evidence with insights from the literature review. This is done to describe the good practices for developing the BI dashboard and for selecting the metrics to be utilized in the performance management of agile programs in the case company. Basically, this chapter seeks answers to the research question: "How should business intelligence be utilized in developing the performance management practices of the case company?" The dashboard is designed as the primary artifact used for the improvement purposes.

Performance management is the process of creating a context for performance measurement. It covers the methods, metrics, processes, and tools that are used for managing performance. (Yadav et al. 2013, Bonghez & Grigoroiu 2013) This means that the improvement of the performance management practices focuses first of all on the methods, metrics, tools, and processes used for it. In this study, the focus is primarily on the metrics and tools since the existing performance metrics of the company are being analyzed and the BI solution is being used as the primary artifact for the improvement. However, the designed BI solution also has an indirect impact on the methods and processes of the company.

5.1 Performance objectives

In this section, performance objectives for the agile programs are discussed. These are later utilized in defining the requirements for the business intelligence dashboard. Referring to the dimensions of the balanced scorecard (BSC) (Kaplan & Norton 1992), the objectives and metrics can be categorized according to the internal business perspective, financial perspective, customer perspective, and innovation and learning. The performance metrics and objectives of the case company were already discussed in chapter 4.3. In this section, these are categorized according to the BSC. Figure 25 illustrates this categorization based on the performance objectives that were identified within the empirical research.

There are also some objectives and metrics that affect multiple dimensions but can't be directly categorized to either of the financial, customer, or internal business dimensions.

The most important ones are the risks that occur in the internal processes and may have an influence on the customer dimension and therefore also on the finances.

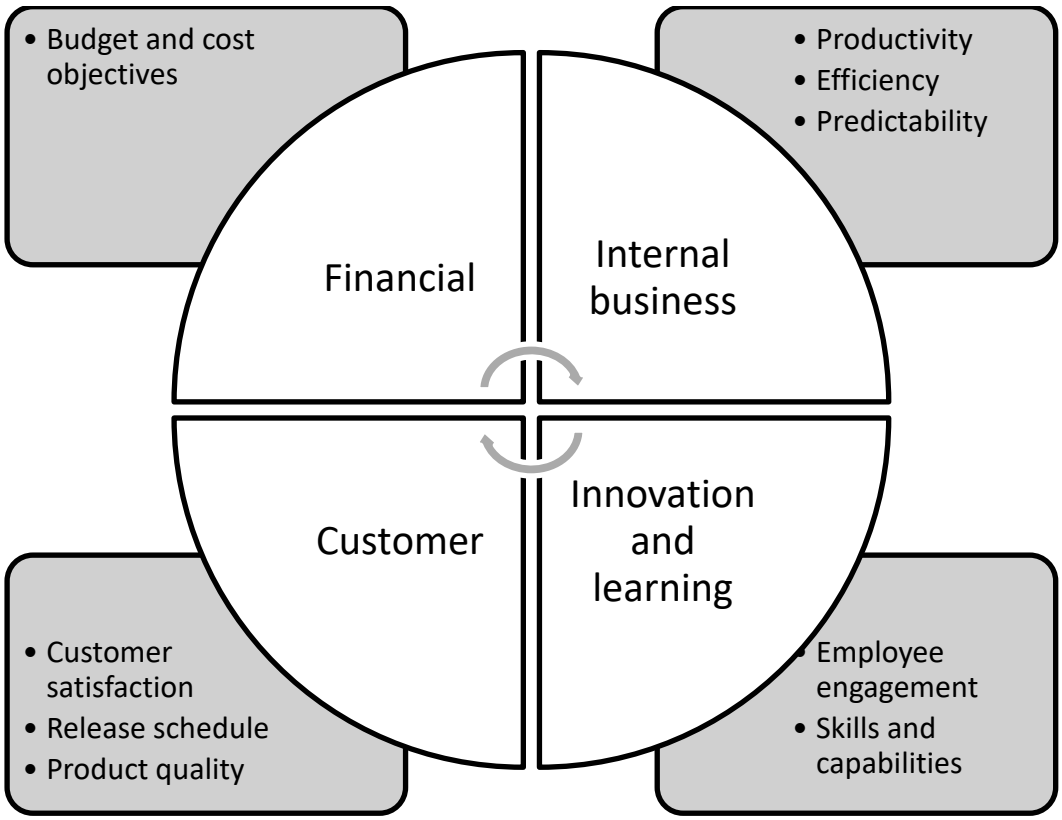

Figure 25: Balanced scorecard categorization of program-level performance objectives in this study

Based on the empirical research, budget and cost-related objectives are the main elements from the financial perspective. The financial objectives of the programs are highly associated with the planned budget and the actual costs caused by the development work. All the interviewees recognized budget and cost matters among the most important metrics for a program. The primary objectives of the programs can be identified to contain reaching the budget and therefore creating value financially. The importance of the cost objectives is highlighted to analyze the profits. Also, it should be noted that the risks may have financial consequences when realized.

The majority of the program level objectives and performance metrics are related to the internal business dimension. The primary metrics used for tracking the organizational performance within the case company are cycle time and velocity. These are followed on the team, release train, and solution train levels. The most common use of these metrics is to track the performance of the teams and trains by sprints and increments. They are

followed within iteration review points, meaning bi-weekly for sprints, every six weeks for increments, and monthly for business steering calls. However, these metrics were emphasized to be important also in enabling predictability of the sprints and increments.

Interviewees recognized the objective of achieving the sprint and increment schedules and outcomes as an important objective for the programs. Especially the rate at which the planned items are completed was recognized as an important measure. Velocity is one commonly used metric for analyzing this factor. There are also targets associated with velocity and cycle time. Especially for cycle time, it is targeted to complete stories within a sprint, meaning 14 days and a feature during one increment or 42 days. Internal bugs are also an important metric considering the internal business and especially the quality of the product development.

Customer satisfaction is the main objective of the customer dimension. Some sub-objectives related to this can be identified to be the release schedule and the responsiveness to the customer requirements. These are major factors leading to customer satisfaction. If the schedule is not met or the product doesn't match the customer's requirements, the customer may change the supplier. Product quality is a major contributor to customer satisfaction and common metrics for it are customer incidents and bugs. The count of customer incidents should be minimized by identifying the bugs internally before releasing the products.

Innovation and learning dimensions are not within the central scope of this study, but some objectives related to it came up during the interviews. Most importantly, one program level goal is to ensure that the teams perform at the required level. This requires employees' skills and capabilities to be at the required level to reach the technical objectives associated with developing the products. Therefore, it is required for the program manager to ensure that the teams are working as needed. Also, employee engagement and satisfaction are important to achieve success in the long-term.

5.2 Defining requirements for the dashboard

In this section, business requirements for the BI dashboard are identified. In this study, these requirements originate from the conclusions made out of the survey indicating that there is a need to improve the efficiency of the current performance reporting practices and to reduce the time used for manual reporting activities. Larson and Chang (2016) stated that a good practice during the identification is to use business questions to provide insights into needed data sources, dimensions, and facts. Also, there should be goals associated with the selected metrics (Lönnqvist & Pirttimäki 2006, Caldiera & Rombach 1994, Ebert et al. 2005) and the goals should be aligned with organizational objectives (Bonghez & Grigoroiu 2013).

During the literature review of this study, the best practices for defining the metrics were discussed. One way of ensuring that the metrics are aligned with goals is to use the Goal-Question-Metric (GQM) approach to supplement the balanced scorecard. Within the GQM approach, questions are defined based on the goals and then metrics are identified to give answers to the questions. The goals can be considered as requirements that the dashboard should satisfy. In this study, they were derived from the objectives of the programs and the challenges that were identified with the current performance measurement and reporting practices.

In table 5 below, the performance management requirements are used as foundations for the GQM and based on those the metrics for the BI dashboard are defined. The selected goals originate from the performance objectives discussed in the previous subchapter (5.1). The more specific goals were defined based on the current state analysis and especially based on the challenges that were identified with the current performance metrics and measurement practices. The questions were formed based on the defined goals and metrics were selected to gain answers to the questions.

Goal	Question	Metric
#1 Achieve the release schedule and contents as planned	What is the planned schedule for the release?	Planned release date
	What is the readiness for the release?	Percentage of initiatives closed of total initiatives
	What is the overall risk of not achieving the schedule and contents as planned?	Count of open risks and magnitudes
#2 Improve the predictability of the increments and sprints	How well can we reach the target of completing features during an increment and stories during a sprint?	Feature cycle time
		Story cycle time
	How many stories are completed during one sprint?	Sprint velocity
	How reliable and accurate is the PI planning process?	Count of epics created after the F3 milestone
#3 Deliver the products within high-quality standards	Are we able to fix the bugs that have been identified?	Internal bugs
	How high is the quality of our products?	Customer bugs
#4 Improve the performance of the testing process	How is our testing pipeline performing?	Test coverage
		Count of test cases

Table 5: Selected metrics derived from organizational goals and related questions

The defined goals and metrics associated with achieving those goals should also be SMART checked. Meaning that they are specific, measurable, achievable, realistic, and time-sensitive (Shahin & Mahbod 2007). Specificity calls for the goals to be as precise as possible. In this case, this builds upon the requirement of giving more accurate descriptions for each of the goals e.g. by defining target levels for each of the goals and discussing how the goals are planned to be reached. It can be stated that each of the goals is measurable since all the metrics derived out of those are quantitative. However, the standard or assumed performance for the measures should be defined.

The achievability says that the goals should not be out of reach and there should be someone responsible for them. This can be recognized to contain the overall responsibility of the goals and the responsibility of the performance measurement. The interviews indicated that the responsibilities within the case company are rather well-acknowledged in general. The teams are responsible for measuring their performance and RTMs and STMs have their entities to be managed. Realistic or relevant goals should be realistically achievable and directly related to the success or failure of the programs. In this case, the lack of specificity disturbs the evaluation of whether the goals can realistically be achieved or not. However, since the goals were derived out of empirical evidence, they can be recognized to be highly relevant.

The time-sensitivity of the goals requires defining a specific deadline for achieving those. The majority of the interviewees recognized that there is room to improve with analyzing the performance in comparison to time. Currently, the possibilities to analyze the historical performance or estimating future direction have been limited. However, there is a lot of historical data available and the deadlines for the goals can easily be drawn.

5.3 Designing the business intelligence dashboard

The designed dashboard should satisfy the needs of the primary stakeholders of this study. This contains the RTMs and STMs working densely with the agile programs. According to Larson & Chang (2016), the main activities in the design phase of the BI lifecycle are modeling and establishing the architecture of the system. Moreover, it should focus on business processes (Elbashir & Williams 2007). In this project, the business processes were considered when selecting the metrics for the dashboard. The identified

organizational objectives and the metrics derived out of those are used as a foundation for the design work.

It is evident that the solution train and release trains bring different requirements for the dashboard. Solution train mainly operates in the portfolio level of the SAFe in which the coordination of initiatives and epics is done across the release trains (Measey 2015) whereas the main goal of the release trains is to organize teams in a way that optimizes the value delivery (Turetken et al. 2017). This means that the release trains operate at the program level and are more interested in organizing the daily work of the teams while the solution train orchestrates the release trains. The teams are primarily working with the features and stories.

Based on the empirical study, the designed dashboard should enable an analysis of the readiness of the products and especially of the release schedule. This was prioritized as the first requirement for the dashboard. The second objective of improving the predictability of the increments and sprints is a major contributor to the release schedule. Furthermore, it can logically be included in the same dashboard. The third goal of delivering the products with high-quality standards can be incorporated in the solution through the bugs that have been linked to the releases in ADO. One good way of analyzing the bugs is to analyze the overall bug count per the releases to assess the quality of the products. Another perspective is to assess the count of bugs closed during the sprints.

The fourth objective of improving the performance of the testing process is a rather independent entity in comparison to the other objectives. However, since the test metrics are primarily analyzed at a team level they should be included in the same dashboard as the other team-level metrics. This builds the motivation for designing two dashboards, one for overall release performance management and another for assessing the more in-depth performance of the teams and release trains.

The financial objectives were decided not to include within the scope of the dashboard since the financial metrics and resources are stored in separate storage and there is currently no means to map them with the product development data. Also, they are a rather independent entity and there is already a BI solution available for them. Therefore, they were not prioritized in this dashboard.

5.3.1 Principles of the dashboards

Two separate dashboards were decided to be designed to ensure their suitability for the identified use cases. Also, the required contents for the dashboards appeared to be too diverse to be included in one collective solution. Therefore, one dashboard was designed for managing the performance of the product releases and another assessing the performance of the release trains and teams.

Release dashboard

The release dashboard concentrates on the paramount issues influencing the delivery of the releases. The empirical research indicated that there is no proper way to measure how well the release schedule and contents are being met. Currently, the release schedule can be analyzed by looking at the planned date and comparing it to the actual completion date. However, there is no way to measure whether all the requirements associated with the release have been fulfilled. The most desired objective for the BI dashboard is to enable measurement of the release schedule and the extent to which the planned work items are completed. This would enable the analysis of the completion rate of the release. This primarily contains the measurement of the rate at which the releases, initiatives, and epics are completed during the predefined timeframe or the schedule. Especially this would fill the deficit that the existing tools have with analyzing the data in contrast to time.

The defined objectives for the release dashboard contain the measurement of the readiness of the releases and the rate at which the contents have been reached in comparison to the schedule. The dashboard should also enable assessing the quality of the release meaning the count of bugs detected related to it. This would enable an analysis of the product quality, which was identified among the most crucial objectives for the programs. The empirical research proved risk management practices to require further emphasis. Hence, the dashboard should also emphasize the risks affecting the release to estimate whether it is likely that the release will be completed as planned.

The releases primarily comprise the initiatives and a prerequisite for assessing the release readiness is that the scope of the initiatives has been frozen. The interviewees recognized that it is problematic when new epics are created after the design process has ended, meaning that the F3 milestone has been reached. Therefore, it appears that there is a need to analyze the number of new items created after the solution design process has ended.

Especially, it appears that there is a need to analyze the timing when new epics are created. This means that in an ideal situation once the product management has split the initiative into epics and given the F3 approval new epics should not appear to the initiative. Contrasting the epics to the milestone schedule builds upon the requirement of analyzing the work items in comparison to time. This was also identified as a major deficit within the current performance measurement possibilities.

In general, there is a strong will to measure the readiness of the work items. For instance, the count of completed epics within an initiative. This metric could be used in various levels of the backlog hierarchy to measure the completion rate of the items. Especially there is a need to track the date when the items have been created, when they are planned to be finished, and have the targets been achieved. This would also demonstrate the planning capability meaning the rate at which the planned items are completed.

Release train and team dashboard

A second dashboard was designed to enable a better organizational aspect to performance management. The dashboard focuses on assessing the release trains' and teams' performance during the increments and sprints. Reflecting on the balanced scorecard, the primary dimensions are internal business and innovation and learning. This means that the objectives associated with the dashboard are productivity, efficiency, predictability, and teams' skills and capabilities. The performance metrics included in the dashboard contain cycle time, velocity, and testing-related metrics. The scope of the dashboard can be roughly categorized into sprint and increment outcomes and testing performance.

At the release train level, the analysis of the readiness level requires identification of the rate at which the committed items have been completed during the increment and sprints. The interviewees recognized organizational performance including productivity and predictability as important objectives for programs. Cycle time and velocity are commonly used metrics for that purpose, and they are regularly followed and utilized for reporting purposes to give information about how the different levels: solution train, release train, and teams are performing. Especially the use of cycle time was highlighted to achieve predictability of the increments and sprints. One interviewee highlighted cycle time to be the most important metric from the perspective of product operations since it indicates the success of the planning process.

There is an objective associated with the cycle time that the user stories should be completed during one sprint. This means that they are planned to be of manageable size and that the right amount of resources is allocated for them. One interviewee recognized that this is objective is achieved to a pretty good extent already. However, the objective of features going from start to complete during one increment is not very well achieved yet.

The number of bugs or defects is an important measure both from the internal process point of view but also from the customer point of view. The majority of the interviewees follow the number of internal bugs as well as the number of customer bugs. Especially the trend of the bugs is being followed and the rate at which the bugs can be closed and how many bugs remain open. A couple of the interviewees mentioned that the number of incidents reported by the customers is the most important metric to measure. It directly implies the quality of products and customer satisfaction.

The dashboard should enable an analysis of the performance of the testing process. Testing related issues were mentioned by many of the interviewees to require further improvement. Both manual and automatized testing are performed within the case company and the major objective related to it is to increase the automatic testing and its coverage. However, there is currently no means to measure it and the number of test cases is used instead to analyze the trend of the testing. The understanding of the test readiness and coverage is acquired by tracking the automated test cases against a specific epic or initiative.

Overall, the count of completed test cases, automated test cases, and manual test cases are tracked. Especially the number of automated test cases of all test cases is being analyzed. One interviewee highlighted that a trend view related to testing is required per team, per release, and release train. This could be used to see how things are progressing and what the forecast would be based on the trend of the past sprints. The forecast could be used to see when work is expected to be completed. One suggestion was to measure the velocity of the testing performed.

5.3.2 Practical implementation of the dashboards

In this section, the practical implementations and functionalities of the designed dashboards are discussed. As a starting point, it was decided that to ensure the simplicity of the development, only the Azure DevOps is used as the data source for the solution. It was identified as the primary data storage for the program-related data while other data sources such as ERP can be incorporated into the dashboard in the future.

Release dashboard

The release dashboard has its primary scope on the release and its contents. The primary metrics in it are release schedule and readiness. The first page of the dashboard gives an overview of the releases, their states, and schedules. Also, details of the important metrics such as bug count and risks related to the release are given. A release list will be drawn with an opportunity to drill through to the more specific release related reports. These reports contain initiatives, risks, and bugs associated with the release.

The initiative report of the dashboard gives an overview of the initiatives concerning the selected release. A list of initiatives is drawn showing the readiness rate of the initiative calculated based on the rate of epics closed within the initiative. Also, more specific details showing the total number of epics, the number of closed epics, and the number of open epics will be shown. A pie graph is drawn showing the count of initiatives by their states. The initiative list also provides an opportunity to drill through to the details page illustrating the epics beneath the initiative. The details page shows the same metrics as for the initiative report but the epics.

The risk report shows the risks affecting the release. The report presents a list of risks that have been identified to have impacts on the delivery of the release. The risk types, states, impacts, severities, and magnitudes of the risks will be illustrated, and a matrix will be drawn showing the count of risks in the respective categories. Especially the risks with the highest magnitude, meaning the risks that cannot be accepted will be highlighted.

The dashboard also enables measurement of the release quality through the defect count. The count of defects related to the release will be calculated. The defect report gives an in-depth presentation of the bugs, their states, and severities. Also, the defects will be

classified according to their sources since the defects are classified as customer tests, internal tests, and production bugs.

Release train and team dashboard

The first page of the dashboard gives an overview of the most important release train level KPIs. First, it allows the user to select a specific release train(s) to be analyzed. Once the release train has been selected, a matrix will show a list of all the individual teams that work under the train. The matrix will show the highlights of the individual team level reports. Also, drill through to the separate report pages is provided.

The second page of the dashboard shows the increment and sprint results by release trains and teams. This report can be used for sprint and increment reviews. On the page, there is a possibility to select the increment and sprint(s) that are to be assessed. Based on the selection, a cycle time graph will be drawn for the release train and the selected team. There is also an opportunity to select which work items are to be included in the cycle time graph, meaning features, user stories, or bugs. Also, velocity will be measured, and a graph will be drawn showing the count of completed work items during the sprints. The page will also provide insights into the count of bugs, their statuses, and severities.

The third page of the dashboard provides an overview of the testing performance of the release trains and teams. First, the release train or team is to be selected. The report will create a filtered list of the test cases showing the title, test type, test usage, quality area, and the automation status for them. Based on the outcomes of the test cases, a pass rate and progress rate will be calculated for the testing performance of the release trains and teams. Also, a pie graph will be drawn showing the test cases by the automation status indicating the coverage of test automation cases of all test cases.

Table 6 below summarizes the designed dashboards that were discussed. In the table, the primary scope, objectives, metrics and work items for the dashboards are discussed. Also, the identified primary use cases for the dashboards are given.

	Release dashboard	Release train & team dashboard
Scope	Releases (products)	Release train, team performance
Objectives	Measure the schedule and readiness of the releases, Track the quality of products, Assess the risks affecting the releases	Evaluate the performance of the increments and sprints, Assess the quality of product development, Analyze the performance of the testing process
Metrics	Release schedule, work items closed (count and percentage), bugs, risks	Cycle time, velocity, test cases, bugs
Work items	Release, initiative, epic	Feature, user story, bug
Usage	Release review, initiative follow-up, risk management	Sprint review, increment review, monthly review (train and team performance)

Table 6: Summary of the designed dashboards

5.4 Next steps

In this section, the next steps with the BI dashboard are demonstrated. As discussed within the literature review, the lifecycle of a BI solution is a continuous process starting with the identification of business requirements and proceeding towards continuous improvement (Gangadharan & Swami 2004, Elbashir & Williams 2007, Larson & Chang 2016). This research process comprised defining the business requirements and designing the dashboard (see figure 26). The following phases of the process are development, testing, and implementation. Writing this thesis, the development of the BI solution has already begun, and a testing and implementation plan has been conducted.

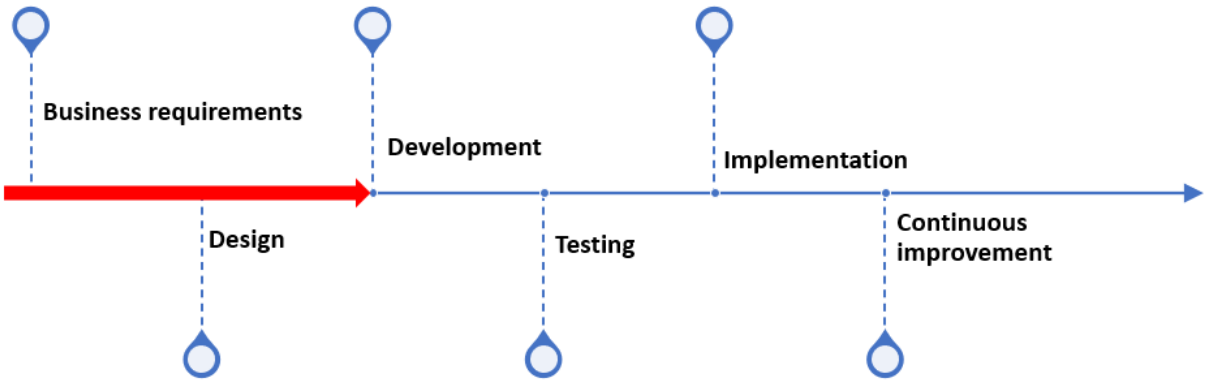

Figure 26: Research timeline contrasted to the BI lifecycle

As discovered during the literature review, the development approach should be incremental, and business-driven (Yeoh & Koronios 2010). This means that a high level of user involvement is required for the system to meet the users' needs. The users were highly involved during the identification of business requirements and design phases. The users were interviewed to identify the requirements and the designed dashboard was also discussed with them before proceeding to the development phase. During the development, a prototype of the dashboard was introduced to the users, and feedback for further development was collected.

According to the literature, the implementation of a BI system is a complex undertaking that requires appropriate infrastructure and resources over a lengthy period (Yeoh & Koronios 2010). Especially, it requires user training and readiness to conduct adjustments

to the system based on new needs. Also, in this case, there needs to be a readiness to carry out user training for the user groups. This means that once the dashboard has been developed and tested, the users should be invited to a session in which the functionality of the dashboard is instructed by going through some common use cases.

The continuous improvement phase contains the identification of further process improvement opportunities (Gangadharan & Swami 2004, Elbashir & Williams 2007). In this case, the natural way to continue is to scale up the BI solution to the entire unit level. For instance, during the interviews, one interviewee mentioned that there has been a will to create dashboards for all the stakeholder groups containing the information they need and hence reduce the need for the reporting streams. The natural next step for scaling up the BI solution is to expand to the product management organization.

The empirical research also indicated that there is a clear need of incorporating financial metrics within the program dashboard. As a starting point of development, these were decided to be descoped to implement the first version more effortlessly. This is because currently there isn't a feasible technical possibility of combining them within the same report since it would require modifications to the current systems to enable the mapping of this data. However, it was decided that e.g. during the continuous improvement phase, the possibility of including them within the dashboards is assessed and reconsidered.

6 CONCLUSIONS

In this chapter, the conclusions are presented including the key findings of the study. Especially the answers to the research questions are presented and the managerial and scientific contribution of this study is elaborated. Also, the limitations of this study are discussed and some topical areas for future research are presented.

6.1 Findings and practical implications

This study aimed at improving the performance management practices of the agile programs in the case company. Especially, this study examined the programs and their objectives and derived performance metrics out of them to be incorporated in the BI solution.

The following research questions were used:

1. How can business intelligence be used in agile program management?
2. Which metrics should be used to measure the performance of agile programs according to existing literature and empirical findings?
3. How should business intelligence be utilized in developing the performance management practices of the case company?

6.1.1 Business intelligence in agile programs

The literature review was conducted to reach an answer to the first research question. First of all, there was a need to define the concept of agile program management since the existing literature on the topic was rather limited. In this study, agile program management was identified to bring together the traditional program concept and the concept of agile. The literature revealed that programs are traditionally seen as entities with a determined purpose and predefined expectations. They are rather long-term undertakings that don’t necessarily have a fixed timeframe or a clearly defined deliverable. They are also increasingly used for the management of multiple interrelated projects

Agile program management takes the traditional program concept and adapts it to the agile environment. The existing literature revealed that agile program management brings value through operations by acting as a link between business strategy and the projects of the organization. This was also confirmed in this study since the program concept of the case company was identified as a continuous function that organizes the product development operations. There are various opinions on which organizational elements are identified as programs but the most obvious and important one is the agile release train that is used for the management and coordination of multiple agile teams that are working in a project form during the increments and sprints.

Business intelligence commonly refers to a managerial philosophy and a tool used to help organizations to manage and refine business information to make more effective business decisions In practice, BI is used to automatize the process of transforming data into knowledge and the most common use of BI is through dashboards. According to the literature review, the implementation of a BI system is a complex function and requires resources over a lengthy period. The business intelligence process can be illustrated via Choo's (2002) information management cycle starting with the identification of information needs and acquiring the information to be utilized in the development of the business intelligence solution.

Business intelligence is most commonly used for performance management. This is also how its usage intertwines in the agile program management context. The literature review revealed that performance management covers the methods, metrics, processes, and tools that are used for managing the performance of the programs. It was identified that BI has potential in improving the performance management practices of the programs by creating solutions that automatically retrieve the data from the data storages and illustrates the metrics and KPIs to the users through dashboards. Also, BI is beneficial in understanding the trends and future directions of the agile programs. This process may lead to increased efficiency within the daily working methods but also to give insights into the business that might not be had otherwise. Especially it appeared to be useful in tracking the performance of the agile programs and the teams operating within them.

6.1.2 Performance metrics in agile programs

The second research question was answered through the literature review and accompanied by the empirical findings. It appears that agile metrics have been a rather popular research topic concerning agile software development in general. However, the knowledge of the metrics to be used especially at the program level is missing. That is what this research contributes towards. However, the metrics identified in this study are primarily directed for the case company, but they may have practical value in other circumstances as well.

As discussed above within the first research question, agile programs take characteristics of traditional programs and the agile principles. Therefore, also the performance metrics in them are defined by the characteristics of those. Traditionally the performance of the programs is defined in terms of meeting the cost, time, and quality criteria. However, this definition is rather narrow, and hence in this study, it was expanded to better acknowledge the complexity of the programs. Some commonly identified metrics for traditional programs were discussed during the literature review to contain budget, costs, resources, schedule, risks, and milestones.

The contemporary literature indicated that agile techniques increase the need for new measurement approaches. Literature suggested some commonly used metrics in agile to include e.g. velocity, effort estimate, and cycle time. Combination of the agile specific metrics with the more traditional project management metrics, such as budget, risks, and milestones, acts as a good foundation for the performance metrics of agile programs. However, in the agile process, metrics are often selected based on the company and its requirements so the generalizability of the metrics may be difficult.

The empirical findings of this study revealed the metrics used in the agile programs to be heavily related to the internal processes of the case company. It turned out that cycle time and velocity are the most important metrics used in measuring the performance of the agile programs. They are commonly used both at the program and team level to track how the release trains and teams are performing. The performance of the agile teams is primarily analyzed in terms of sprint velocity and cycle time, containing the measures of user stories and bugs. Also, the outcome of the increments is important and for that purpose, the cycle time and velocity are measured on the feature level.

Other performance metrics based on empirical research contain the count of internal and external bugs, which are a major indicator of the quality perspective. From the financial perspective, the metrics contain actual costs, meaning the cost accumulated by the production, budget, full-time-equivalent, and effort. Budget and costs are direct measures while resources and effort can be identified to be major contributors to the costs. The effort estimate is generally identified as the main cost of software development since it includes the human effort required for the design, coding, and testing activities of the software product.

The research indicated that testing performance is a common subject to be measured in the agile environment. The literature review suggested test coverage as the most important metric for it. However, in companies that aim at the efficiency of the testing by performing the majority of the test cases automatically, the count of automated test cases in comparison to the manual test cases is another important measure showing the trend of it. In this case, the coverage of automated test cases is an important metric.

In this research, also new metrics were suggested for measuring the performance of the agile programs. The metrics were derived out of organizational goals to ensure their relevance. Some of the most notable program objectives identified in this study contain productivity, predictability, quality, and schedule. The metrics were defined concerning these objectives. Some of the suggested metrics were all new such as the metric to measure the readiness as the percentage of work items closed. Especially considering the initiatives closed of total initiatives were suggested as a metric for assessing the readiness of the releases. The same metric applies also for lower hierarchy items such as initiatives by measuring the count of completed epics.

6.1.3 Development of the performance management practices

The answer to the third research question was acquired through the empirical part of this study. With empirical research, a business intelligence dashboard was designed using the design science research method (DSRM). The dashboard was targeted to tackle the problems with the existing performance management practices of the case company especially related to performance measurement. The problems to be solved were preliminarily identified through a survey that indicated several issues within the current practices of the case company. Some of the most notable being improving the efficiency

of the performance reporting process and reducing the time used for manual reporting practices.

The empirical research indicated that there is room for improvement what comes to aligning the performance metrics with business objectives. This was targeted within this study by first identifying the objectives and categorizing them according to the dimensions of the balanced scorecard (BSC). And then using a structured way of deriving the metrics from the organizational goals with the goal-question-metric (GQM) approach. This was done to increase the understanding of the motivation for the usage of the metrics. The selected metrics were then used in the design process of the BI dashboard.

A lifecycle model was used for designing the BI solution starting with the identification of business requirements. According to the identified requirements, the BI dashboard was designed to satisfy the needs of the primary stakeholders of this study. The dashboard was designed based on the identified performance reporting responsibilities of the stakeholders. It was designed primarily to improve the efficiency of the current reporting practices with an ultimate goal to reduce the need for regular reporting activities and to provide new capabilities to gain insights into the data. For instance, it should enable better analysis of the historical data and give new insights to the underlying trends by contrasting the data to the time function.

The overall use of the dashboard was classified into two dimensions: release dimension and release train and team dimension. It was identified that there is a different kind of need for performance reporting on these dimensions. The most important release-related objective is to measure the schedule and readiness of the releases. Also, it was identified that quality and risks are significant factors in achieving the release schedule and contents as well. The primary work items included in this dimension are the top-three of the backlog hierarchy, i.e. releases, initiatives, and epics.

Release trains and teams are the primary scope in the organizational dimension. The empirical research indicated that there is a need to evaluate the release trains' and teams' performance during the increments and sprints. It appeared that this kind of reporting is arranged primarily during the sprint, increment, and monthly reviews. Therefore, it proved out to be beneficial to design a separate dashboard with the primary scope of assessing the performance during the predefined durations of sprints and increments.

The research also indicated that the targets for the performance metrics are not comprehensively clear. For some of the metrics such as the cycle time and velocity the purpose and target of measurement are commonly recognized but e.g. the targets associated with quality and testing are not clear to all the stakeholders. Furthermore, it turned out that the case company doesn't have any documentation of the metrics. This is one thing that might also increase the understanding of the metrics and their targets.

It was also evident that the common understanding of business intelligence needs to be increased within the case company. The participants of this study have very high expectations for the BI utilization even though they don't know the realities of it. It was planned that while putting the dashboard into operation, user training should be arranged to demonstrate the dashboard in practice but also to increase the understanding of BI.

6.2 Scientific implications and suggestions for further research

While conducting this research it turned out that the existing research on the topic of agile program management is rather limited. In this study, a holistic definition for the concept was given by combining the existing program management literature with the agile perspective on programs and projects. The agile software development and traditional program management as such have been very popular research topics but within the combination of those not much previous research has been conducted. Overall, this research is one of the first ones conducted about the agile program management topic.

However, this research didn't solely introduce the agile program management concept but also introduced performance metrics to be used in it. Existing research on the program management topic (e.g. Pellegrinelli 1997, Archibald 2003, Blomquist & Müller 2006, Artto & Dietrich 2007, Artto et al. 2009), and on the topic of agile management (e.g. Anderson 2004, Chin 2004, Dingsøyr et al. 2010, Sheffield & Lémateyr 2013, Conforto et al. 2014, Lappi et al. 2018) were reviewed to discuss the foundations for performance management in agile programs. Especially in this study, the common agile metrics (e.g. Misra & Omorodion 2011, Todorovic et al. 2013, Kupiainen et al. 2015) were assessed in the agile program context. As an outcome, a comprehensive list of performance metrics to be used in the management of agile programs was composed. These were also cross-checked to the case company and as a conclusion, a list of confirmed agile program metrics was created. This list could be also used as a foundation for further research.

Also, there is a lot of previous research on the business intelligence topic from the technical perspective, but rather little research has been conducted about its use cases in the project-oriented business. In this study, the selected research articles suitable for this study (e.g. Gangadharan & Swami 2004, Elbashir & Williams 2007, Negash & Gray 2008, Larson & Chang 2016, Laursen & Thorlund 2016, Popovič et al. 2019 , etc.) were reviewed in order to identify the use cases of BI in project-based organizations. This research also contributes to the deficit by describing one extensive use case and presents a timeline for the BI solution in the context. In practice, this research described the way business intelligence can be used in program management through performance management practices.

There would be potential further research opportunities considering the deficits of this study. This research gives an overview of the program and team levels and of the agile software development organizations especially considering the performance metrics and measurement practices. One potential topic for further research would be to expand the research to the unit level. It would be interesting to gain insights into the underlying reasons why the program level activities are done as they are. This includes the identification of the factors on how the objectives and targets have been set for agile programs. All in all, this research considered the direct program managers of the case company. Nevertheless, to achieve a more holistic perspective on the topic it would require the involvement of the diverse stakeholders contributing to the programs indirectly. For instance, some evident stakeholders would be product management, operative product owner, unit head, and members of the agile team.

6.3 Limitations of the study

This research has some limitations, which should be considered when interpreting the results. First of all, the scope of the research was intentionally limited to solely cover the program level of the organization. This allows thorough analysis of the program level practices but blurs the visibility to the other levels of the organization which may distort the big picture. For instance, this research did not directly consider the individual teams when analyzing their way of working. Therefore, this research is primarily applicable to agile trains.

Due to the potentially rather large topic area of this study, it was decided to limit the number of interviewees accurately. The survey and interviews were arranged only for the direct target users of the designed solution. This leads to a situation in which the designed solution is useful for the selected target group, but the conclusions made out of the research may be poorly generalized outside the research environment. Survey and interview as research methods also rely heavily on the formatting of the questions and on how the questions are interpreted. Especially within the Likert-scaled survey, the respondents' contemporary state of mind influences significantly on the answers.

The research was conducted as a case study and the author was working inside the case company. This means that the research may be affected by the subjectivity of the author. Also, since the study was conducted in the specific environment of the case company it can be questioned whether the results of this study can be generalized to other circumstances. This research only considers the characteristics of the software industry which limits the generalizability of the results heavily. Furthermore, the research was conducted for one unit of the case company meaning that the results can’t be directly generalized even at the entire organization level not to mention the entire software industry.

Also, the research relied heavily on existing research conducted in the topical areas. The majority of the existing literature was published in scientifically recognized journals and books but some of the topics required relying on facts that were available on web pages. Therefore, it is possible that all the facts used in this paper are not the absolute scientifically recognized truth.

REFERENCES

Ahmad, M. O., Markkula, J., & Oivo, M. 2013. Kanban in software development: A systematic literature review. Proceedings of the 39th EUROMICRO conference on Software Engineering and Advanced Applications (SEAA), Washington: IEEE, p. 9-16. ISBN 978-0-7695-5091-6

Anderson D.J., 2004. Agile management for software engineering: Applying the theory of constraints for business results. 9th edition. United States at Hamilton in Castleton, New York: Prentice Hall Professional Technical Reference, 313 p. ISBN 0-13-142460-2

Agrawal, M., & Chari, K. 2007. Software effort, quality, and cycle time: A study of CMM level 5 projects. IEEE Transactions on software engineering, 33 (3), 145-156.

Archer, N. P., & Ghasemzadeh, F. 1999. An integrated framework for project portfolio selection. International Journal of Project Management, 17 (4), 207-216.

Archibald, R., 2003. Managing high technology programs and projects. 3rd edition. Hoboken, New Jersey USA: John Wiley and Sons, 396p. ISBN 978-0-471-26557-3

Atkinson R., 1999. Project management: "Cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria". International Journal of Project Management, 17 (6), p. 337-342.

Artto, K., & Dietrich, P. 2007. Strategic business management through multiple projects. The Wiley guide to project, program & portfolio management, p. 1-33. ISBN 978-0-470-22685-8

Artto, K., & Kujala, J. 2008. Project business as a research field. International Journal of Managing Projects in Business, 1 (4), p. 496-497.

Artto, K., Martinsuo, M., Gemünden, H. G., & Murtoaro, J. 2009. Foundations of program management: A bibliometric view. International journal of project management, 27 (1), p. 1-18.

Artto, K. A., & Wikström, K. 2005. What is project business?. International Journal of Project Management, 23 (5), p. 343-353.

Beck K., Beedle M., Van Bennekum A., Cockburn A., Cunningham W., Fowler M., . . . Jeffries R., 2001. Manifesto for agile software development [online document]. Available from: http://agilemanifesto.org, [Accessed 18.9.2020]

Blichfeldt, B. S., & Eskerod, P. 2008. Project portfolio management–There's more to it than what management enacts. International Journal of Project Management, 26 (4), p. 357-365.

Blomquist, T., & Müller, R. 2006. Practices, roles, and responsibilities of middle managers in program and portfolio management. *Project Management Journal, 37* (1), p. 52-66.

Bonghez, S. & Grigoroiu, A. 2013. If it can't be expressed in figures—project performance management. Paper presented at PMI® Global Congress 2013—EMEA, Istanbul, Turkey. Newtown Square, PA: Project Management Institute

Božič, K., & Dimovski, V. 2019. Business intelligence and analytics for value creation: The role of absorptive capacity. International journal of information management, 46 (2019), p. 93-103.

Brady, D. 2014. Using visual management to improve transparency in planning and control in construction (Doctoral dissertation, University of Salford).

Bragen M. 2018. IT solutions of Data Analytics as Applied to Project Management. In Spalek, S. (Ed.) Data analytics in project management. New York: CRC Press, p. 133-150. ISBN 978-1-138-30728-5

Bryde, D. J. 2003. Modelling project management performance. International Journal of Quality & Reliability Management, 20 (2), 229-254.

Bryde, D. J., & Wright, G. H. 2007. Project management priorities and the link with performance management systems. Project Management Journal, 38 (4), p. 5-11.

Caccamese, A. & Bragantini, D. 2012. Beyond the iron triangle: year zero. Paper presented at PMI® Global Congress 2012—EMEA, Marsailles, France. Newtown Square, PA: Project Management Institute.

Caldiera, V. R. B. G., & Rombach, H. D. 1994. The goal question metric approach. Encyclopedia of software engineering, p. 528-532.

Cambridge Dictionary. Available from: http://dictionary.cambridge.org [Accessed 13.10.2020].

Cervone, H. 2011. Understanding agile project management methods using scrum. OCLC Systems & Services: International Digital Library Perspectives, 27 (6) p. 18-22.

Chen, H., Chiang, R., & Storey, V. 2012. Business Intelligence and Analytics: From Big Data to Big Impact. MIS Quarterly, 36 (4), p. 1165-1188.

Chin G., 2004. Agile project management: How to succeed in the face of changing project requirements. New York: AMACOM, a division of American Management Association, 229 p. ISBN 0-8144-7176-5

Choo, C.W. 2002. Information Management for the Intelligent Organization: The Art of Scanning the Environment. 3rd edition. Medford, NJ: Information Today, Inc., 325 p. ISBN 1-57387-125-7

Cockburn A., 2002. Agile software development. 2nd edition. Boston: Pearson Education, Inc., 278 p. ISBN 0-201-69969-9

Conforto, E. C., Rebentisch, E., & Amaral, D. C. 2014a. Project management agility global survey (Massachusetts Institute of Technology, Consortium for Engineering Program Excellence—CEPE). Cambridge, MA: Massachusetts Institute of Technology

Conforto, E. C., Salum, F., Amaral, D. C., Da Silva, S. L., & De Almeida, L. F. M. (2014b). Can agile project management be adopted by industries other than software development?. Project Management Journal, 45(3), p. 21-34.

Cooper, R., Edgett, S., Kleinschmidt, E., 1997. Portfolio management in new product development: lessons from the leaders I. Research Technology Management 40 (5), p. 16–28.

Dingsøyr T., Dybå T. & Moe N. B., 2010. Agile software development: Current research and future directions. Berlin: Springer Science & Business Media, 238 p. ISBN: 978-3-642-12574-4

Doran, G. T. (1981). There'sa SMART way to write management's goals and objectives. Management review, 70 (11), p. 35-36.

Drumond, C., 2020. Agile project management. Atlassian [online document]. Available from: https://www.atlassian.com/agile/project-management [Accessed 7.10.2020].

Ebert, C., & Dumke, r., Bundschuh, M. & Schmietendorf, A., 2005. Best Practices in Software Measurement: How to use Metrics to Improve Project and Process Performance, 295 p. ISBN 3-540-20867-4

Eckerson, W. 2006. Performance Dashboards: Measuring, Monitoring and Managing Your Business. New Jersey, NJ: John Wiley and Sons, 336 p. ISBN 978-0-470-58983-0

Elbashir, M., & Williams, S. 2007. BI impact: The assimilation of business intelligence into core business process. Business Intelligence Journal, 12 (4), p. 45-54.

Gangadharan, G., & Swami, S., 2004. Business intelligence systems: design and implementation strategies. Proceedings of 26th International Conference on Information Technology Interfaces, Cavtat: IEEE p. 139-144. ISBN 953-96769-9-1

Hartmann, D., Dymond, R., 2006. Appropriate Agile Measurement: Using Metrics and Diagnostics to Deliver Business Value. Proceedings of AGILE 2006 Conference, Washington: IEEE Computer Society, p. 423. ISBN 9780769525624

Hevner, A. R., March, S. T., Park, J., & Ram, S. 2004. Design science in information systems research. MIS quarterly, 28 (1), p. 75–105.

Hevner, A. R. 2007. A three cycle view of design science research. Scandinavian journal of information systems, 19 (2), p. 87-92.

Kaplan RS and Norton DP. 1992. The Balanced Scorecard—Measures that Drive Performance. Harvard Business Review. p. 71-79

Kerzner H., 2009. Project management: A systematic approach to planning, scheduling, and controlling. 10th edition. New Jersey: John Wiley & Sons, Inc., 1094 p. ISBN 978-0-470-27870-3

Kerzner, H. 2013. Project Management Metrics, KPIs and Dashboards. 2nd edition. New Jersey, NJ: John Waley and Sons, 448 p. ISBN 978-1119427285

Killen, C. P., Geraldi, J., & Kock, A. 2020. The role of decision makers' use of visualizations in project portfolio decision making. International Journal of Project Management, 38 (5), p. 267–277.

Kunz, M., Dumke, R. R., & Zenker, N. 2008. Software metrics for agile software development. Proceedings of the 19th Australian Conference on Software Engineering, Washington: IEEE, p. 673-678. ISBN 978-0-7695-3100-7

Kupiainen, E., Mäntylä, M. V., & Itkonen, J. 2015. Using metrics in Agile and Lean Software Development–A systematic literature review of industrial studies. Information and Software Technology, 62 (2015), p. 143-163.

Lappi, T., Karvonen, T., Lwakatare, L. E., Aaltonen, K., & Kuvaja, P. 2018. Toward an improved understanding of agile project governance: a systematic literature review. Project Management Journal, 49 (6), p. 39-63.

Larson, D., & Chang, V. 2016. A review and future direction of agile, business intelligence, analytics and data science. International Journal of Information Management, 36 (5), p. 700–710.

Laursen, G. H., & Thorlund, J. 2016. Business analytics for managers: Taking business intelligence beyond reporting. John Wiley & Sons, 295 p. ISBN 9781119302520

Leffingwell, D. 2007. Scaling software agility: best practices for large enterprises. Pearson Education, 349 p. ISBN 0321458192

Leffingwell, D., 2009. Agile Enterprise Requirements Information Model (6b) – Subset for Portfolio Management: Epics [online document]. Available from: https://scalingsoftwareagility.wordpress.com/tag/agile-big-picture/ [Accessed 15.10.2020].

Leffingwell, D. 2018. SAFe 4.5 Reference Guide: Scaled Agile Framework for Lean Enterprises. Addison-Wesley Professional, 816 p. ISBN 978-0134892863

Lindvall M., Basili v., Boehm B., Costa P., Dangle K., Shull F., Tesoriero R., Williams L. & Zelkowitz M. 2002. "Empirical Findings in Agile Methods", Proceedings of the Conference on extreme programming and agile methods. Berlin, Heidelberg: Springer-Verlag, p. 197-207. ISBN 978-3-540-44024-6

Lönnqvist, A., & Pirttimäki, V. 2006. The measurement of business intelligence. Information systems management, *23* (1), p. 32–40.

Martinsuo, M. 2013. Project portfolio management in practice and in context. International journal of project management, 31 (6), p. 794-803.

Matook, S., & Brown, S. A. 2017. Characteristics of IT artifacts: a systems thinking-based framework for delineating and theorizing IT artifacts. Information Systems Journal, 27 (3), p. 309–346.

Measey, P. 2015. Agile Foundations: Principles, Practices and Frameworks. BCS, The Chartered Institute for IT, 196 p. ISBN 9781780172545

Mir, F. A., & Pinnington, A. H. 2014. Exploring the value of project management: linking project management performance and project success. International journal of project management, 32 (2), p. 202-217.

Misra, S., & Omorodion, M. 2011. Survey on agile metrics and their inter-relationship with other traditional development metrics. ACM SIGSOFT Software Engineering Notes, 36 (6), p. 1-3.

Negash, S., & Gray, P. 2008. Business intelligence. In: Burstein, F., & Holsappe, C. (eds) Handbook on decision support systems 2. Berlin, Heidelberg. Springer, p. 175-193. ISBN 978-3662501016

Nicolette, D., 2015. Software Development Metrics. Manning Publications, 192 p. ISBN 9781617291357

Nuottila, J., Aaltonen, K., & Kujala, J. 2016. Challenges of adopting agile methods in a public organization. International Journal of Information Systems and Project Management, 4 (3), p. 65–85.

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. 2007. A design science research methodology for information systems research. Journal of management information systems, 24 (3), p. 45–77.

Pellegrinelli, S. 1997. Programme management: organising project-based change. International Journal of Project Management, *15* (3), p. 141-149.

Pinto, J. K., & Prescott, J. E. 1988. Variations in critical success factors over the stages in the project life cycle. Journal of management, 14 (1), p. 5-18.

Popovič, A., Puklavec, B., & Oliveira, T. 2019. Justifying business intelligence systems adoption in SMEs. Industrial Management & Data Systems, 119 (1), p. 210-228

Project Management Institute, 2013. A Guide to the Project Management Body of Knowledge (PMBOK® Guide), 5th edition., 616 p. ISBN: 9781935589679

Rehnkopf, M., 2020. Atlassian. Agile project management. Epics, Stories, themes, and Initiatives. Available from: https://www.atlassian.com/agile/project-management/epics-stories-themes [Accessed 8 October 2020].

Richards, G., Yeoh, W., Chong, A. Y. L., & Popovič, A. 2019. Business intelligence effectiveness and corporate performance management: an empirical analysis. Journal of Computer Information Systems, 59 (2), p. 188-196.

Rossberg, J. 2019. Agile Project Management with Azure DevOps: Concepts, Templates, and Metrics. Apress, 226 p. ISBN 978-1-4842-4482-1

Scaled Agile, Inc., 2020. Scaled Agile Framework. Available from: https://www.scaledagileframework.com/ [Accessed 7 October 2020].

Sedano, T., Ralph, P., & Péraire, C. 2019. The product backlog. Proceedings of the 41st International Conference on Software Engineering (ICSE), Montreal: IEEE, p. 200-211. ISBN 978-1-7281-1764-5

Serrador P. & Pinto J. K., 2015. Does agile work? —A quantitative analysis of agile project success. International Journal of Project Management, 33 (5), p. 1040-1051.

Sheffield J. & Lemétayer J. 2013. Factors associated with the software development agility of successful projects, International Journal of Project Management, 31 (3). p. 459-472.

Schoper, Y. G., Wald, A., Ingason, H. T., & Fridgeirsson, T. V. 2018. Projectification in Western economies: A comparative study of Germany, Norway and Iceland. International Journal of Project Management, 36 (1), p. 71-82.

Stettina C. J. & Hörz J. 2015. ”Agile portfolio management: An empirical perspective on the practice in use”, International Journal of Project Management, 33 (1), p. 140- 152.

Thiry, M. 2010. Agile program management. Paper presented at PMI® Global Congress 2010—North America, Washington, DC. Newtown Square, PA: Project Management Institute.

Todorović, M., Mitrović, Z., & Bjelica, D. 2013. Measuring project success in project-oriented organizations. *Management*, *68* (2013), p. 41-48.

Todorović M. L., Petrović D. Č, Mihić M. M., Obradović V. L. & Bushuyev S. D. 2015. “Project success analysis framework: A knowledge-based approach in project management”, International Journal of Project Management, 33 (4), p. 772-783.

Turetken, O., Stojanov, I., & Trienekens, J. J. 2017. Assessing the adoption level of scaled agile development: a maturity model for Scaled Agile Framework. Journal of Software: Evolution and process, 29 (6), p. 1–18.

Turner, R. 1999. Handbook of Project-Based Management. 2nd edition. London: McGraw-Hill Companies, 529 p. ISBN: 0-07-709161-2

Yeoh, W., & Koronios, A. 2010. Critical success factors for business intelligence systems. Journal of computer information systems, *50* (3), p. 23–32.

Wang, Y., Kung, L., & Byrd, T. A. 2018. Big data analytics: Understanding its capabilities and potential benefits for healthcare organizations. Technological Forecasting and Social Change, 126 (2018), p. 3–13.

Willen, C. 2002. Airborne Opportunities, Intelligent Enterprise, 5 (2), p. 11–12.

Yadav, N., & Sagar, M. 2013. Performance measurement and management frameworks. Business Process Management Journal, 19 (6), p. 947-971.

Zdonek, I. 2020. Project Indicators Visualization Using an Interactive Dashboard. Scientific papers of Silesian university of technology, number 143. p.329–339.

Zheng, J. G. 2017. Data visualization for business intelligence. In: Munoz, M. (ed.) Global Business Intelligence. New York: Taylor & Francis, p. 67–82. ISBN 9781315471136

APPENDICES

APPENDIX 1: Survey Questionnaire

Primarily Likert-scaled questions (1-5) and open-ended questions.

1. **Background**
   - Current job position
     a. Release Train Manager
     b. Solution Train Manager
   - Experience in current position
     a. less than 1 year
     b. 1 to 5 years
     c. more than 5 years
   - What are your main responsibilities?
   - Time used weekly to create reports
     a. Less than one hour
     b. 1-5 hours
     c. 5-10 hours
     d. more than 10 hours
   - Time used monthly to create reports
     a. Less than one hour
     b. 1-5 hours
     c. 5-10 hours
     d. more than 10 hours
2. **Program reporting process**
   - I am familiar with the targets associated with program reporting. (1-5)
   - I find the current reporting processes and tools to support the achievement of targets. (1-5)
   - I identify the reporting process as an important contributor towards successful product delivery. (1-5)
   - In my opinion program reporting is arranged to be easy and effective. (1-5)
   - I recognize the roles and responsibilities in the reporting process. (1-5)

- Who are the main stakeholders involved in the reporting process? What do you identify as their main responsibilities during the process?
- Additional comments related to reporting practices.

3. **Data**
   - I know where the data I need is stored. (1-5)
   - I need to collect data from several different systems. (1-5)
   - The data I need is easily available. (1-5)
   - The data I need is of good quality. (1-5)
   - It is common that manual steps are needed to process the data I need. (1-5)
   - What data is needed for your reporting purposes? What are the systems in which the data is stored?
   - Additional comments related to data.
4. **Metrics and KPIs**
   - I am familiar with metrics related to the programs I'm involved in. (1-5)
   - I am familiar with KPIs related to the programs I'm involved in. (1-5)
   - I think the metrics and KPIs are well align with the business targets. (1-5)
   - I think there is appropriate number of metrics and KPIs to be followed. (1-5)
   - I think that the definitions of metrics and KPIs are clearly documented and generally available. (1-5)
   - What is the main motivation of using metrics and KPIs in your program?
   - What metrics and KPIs are regularly followed in your program? How often are these followed? (e.g. weekly/monthly)
   - Which metrics and KPIs have the highest level of business importance and why?
   - Additional comments related to metrics and KPIs.
5. **Business intelligence**
   - I am familiar with the basic concept of business intelligence. (1-5)
   - I find business intelligence applicable for program management. (1-5)
   - I think that business intelligence could further improve the efficiency of my work activities. (1-5)
   - I am happy with the current availability rate of business intelligence solutions in my unit. (1-5)
   - I can imagine business intelligence dashboard replacing majority of the current report templates. (1-5)

- I have previous experience using business intelligence in my current position. (y/n)
- In which use cases have you used business intelligence? (if you answered yes to the question above)
- What purposes do you think business intelligence is the most applicable for? What are your expectations for the business intelligence usage in the program management domain?
- Free word about the topic.

APPENDIX 2: Interview Questionnaire

A: Background

- How long have you been working in the company and in what positions?
- Are you involved in one or more programs? What is your role and main responsibilities in those?

B: Programs and their objectives

- How do you define a program?
- Could you tell briefly of the programs you are involved in. When did they start and when are they planned to end? What regular activities do the programs have?
- What are the main targets of the program you are involved in? (how does it create value)
- How have these targets been defined and who has set the targets?
- Are there specific target levels associated with achieving the objectives? How is the rate of achieving the targets measured?
- Do you find the targets reasonable and achievable?
- What have you found most surprising about the practices of the programs? Would you like to change something in the practices?

C: Metrics and measurement

- What metrics and KPIs do you use and follow regularly?
- Why were these metrics selected? Which factors affected the choice of these metrics? (e.g. tradition, command, development method, customers)

- Were you involved in the selection of the metrics? Are you following some voluntary/additional metrics?
- Have the metrics and measurement methods been clear since the beginning of the program? How do you get information about these metrics?
- How do you measure and get data for following?
    - Quality
    - Budget
    - Costs
    - Risks
    - Milestones
    - Velocity
    - Cycle Time
    - Bugs
    - Testing
- For what purposes do you primarily use metrics? (e.g. planning, monitoring, decision making, development)
- Do you find some metrics more important than others? Why?
- Which metric do you find the hardest to measure?
- Do you have some improvement ideas concerning the metrics and measurement? Have the metrics and measurement methods been clear since the beginning of the program? How do you get information about these metrics?

D: Reporting practices

- To whom and how often do you report forward the program progress?
- What are the main activities related to the reporting? Which of these are the most time consuming? (data collection, report creation)
- What tools are you using for reporting? Has there been any issues?
- What systems do you use for the data acquisition? How do you process the data?
- What kind of visualizations do you find best for reporting?
- Has there been any difficulties with reporting?
- Do you hope some kind of improvement concerning the current reporting practices?

Can you show or provide me any specific reporting material that you are using?