

5-15-2019

DYNAMICS OF INTER-TEAM COORDINATION ROUTINES IN LARGE- SCALE AGILE SOFTWARE DEVELOPMENT

Tomas Gustavsson

Karlstad Business School, tomas.gustavsson@kau.se

Follow this and additional works at: https://aisel.aisnet.org/ecis2019_rp

Recommended Citation

Gustavsson, Tomas, (2019). "DYNAMICS OF INTER-TEAM COORDINATION ROUTINES IN LARGE-SCALE AGILE SOFTWARE DEVELOPMENT". In Proceedings of the 27th European Conference on Information Systems (ECIS), Stockholm & Uppsala, Sweden, June 8-14, 2019. ISBN 978-1-7336325-0-8 Research Papers.
https://aisel.aisnet.org/ecis2019_rp/40

This material is brought to you by the ECIS 2019 Proceedings at AIS Electronic Library (AISeL). It has been accepted for inclusion in Research Papers by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

DYNAMICS OF INTER-TEAM COORDINATION ROUTINES IN LARGE-SCALE AGILE SOFTWARE DEVELOPMENT

Gustavsson, Tomas, Karlstad University, Karlstad, Sweden, tomas.gustavsson@kau.se

Abstract

Software development organizations are adopting values, principles, and frameworks to implement agile ways of working today. But the agile methods were initially designed for use in small, single-team projects and routines for coordination between several teams have not been adopted in the same way as routines for coordination within the team. The Scaled Agile Framework has become the most common way to implement organizational routines for inter-team coordination, but critiques claim it to be too strict and formal, without leeway for adaption. This study investigates the dynamics of inter-team coordination routines at three organizations and provides thick descriptions of tailoring. Data collection was performed by 379 hours of observations and 28 interviews. The main findings highlight the variety in ostensive and performative aspects of coordination routines and how they change over time. Contrary to earlier findings, the ostensive and performative aspects in this study do not have opposing varieties. This indicates that the empirical relationship between ostensive and performative aspects might not be as atypical as previous results suggest. An important practical contribution is the described possible tailoring options when scaling up agile ways of working which contradict the view of the framework being too rigid.

Keywords: Routine Dynamics, Inter-team coordination, Project management, Large-scale, Agile Software Development.

1 Introduction

Today, agile ways of working have become the norm in software development as a majority of IT organizations are adopting the values, principles, and frameworks to become agile (VersionOne, 2018). However, the agile methods were initially designed for use in small, single-team projects (Boehm and Turner, 2005) and routines for coordination between several teams have not been adopted in the same way as routines within the team. As an example, the annual industry survey on agile ways of working conducted by VersionOne (2018) shows that 90 percent of all respondents conduct the “Daily standup” routine and 85 % perform the “Retrospectives” routine, but there is a diversity of implemented large-scale frameworks and, e.g., Scrum of Scrums is only adopted by 19 percent of the respondents. Since an increasing amount of larger software development organizations adopt agile ways of working (Xu, 2009; VersionOne, 2018) routines for inter-team routines will be increasingly important. Xu (2009) also highlights that larger software development organizations, using agile ways of working, risk a lack of interaction and difficulties in communication.

Hoegl, Weinkauf, and Gemuenden (2004) point out that coordination is more important to team performance in large projects, with several cooperating teams than in one-team projects. Vlietland and Vliet (2015) also propose that coordination routines within and between agile teams positively impact delivery predictability. It is important to study how routines for coordination are conducted in large-scale agile development since our understanding of large-scale coordination is limited (Dietrich, Kujala, and Artto, 2013).

Dietrich et al. (2013) studied the different types of coordination on individual versus group mode used in large-scale agile development, and Dingsøy, Moe and Seim (2018) studied different types of coordination mechanisms used in a large-scale agile program and how they evolved. Both studies present a list of routines for coordination in large-scale agile settings but do not investigate in detail how and why these routines are performed in the chosen manner. This paper focuses on the details of three ag-

ile inter-team coordination routines used in large agile software development projects. Previous studies on coordination between teams have put much attention to permanent constellations such as organizations, but less to temporal constellations such as projects (Dietrich et al., 2013).

By adopting an innovative research frame in the context of inter-team coordination research: the evolutionary theory of organizational routines (Pentland and Feldman, 2005), a detailed understanding of software development work might be gained. Investigating inter-team coordination through the lens of routines, using routines as the unit of analysis, enables us to examine the dynamics between routines for coordination and the tailoring of these routines. By understanding both the differences in *how* coordination routines are conducted, the performative aspect, as well as abstract patterns based on purpose and reasons for the way routines are enacted, the ostensive aspect (or the *why*), further insights into routine tailoring might be achieved. This study explores the dynamics within and across routines for inter-team coordination as they are enacted in practice. The paper offers rich descriptions of the use of concrete organizational routines for inter-team coordination in large-scale agile settings.

In the following, Section 2 discusses inter-team coordination in agile software development and presents the Scaled Agile Framework. In Section 3, routine dynamics as a concept is described and Section 4 provides a description of the data collection and method for data analysis. Section 5 describes results and Section 6 covers the discussion of the results.

2 Inter-team coordination in agile software development

Agile software development is often defined by the values and principles as described in the “Manifesto for Agile Software Development” (Beck et al., 2001) which mostly explain what is important for team members, working together in one team. The same goes for Scrum, the most commonly implemented agile framework (VersionOne, 2018). Nevertheless, Scrum and these values and principles are not enough to explain how to most efficiently organize multiple teams working together towards a common goal where teams need to coordinate their work. Therefore, several new frameworks designed for large-scale agile software development have emerged of which the Scaled Agile Framework (SAFe) is the most commonly adopted (VersionOne, 2018). SAFe was introduced at a conference in August of 2013 by the originator Dean Leffingwell and, together with other partners, it has been further developed and is now in version 4.6 (ScaledAgile, 2018). SAFe describes agile ways of working on different levels, starting from team level to the portfolio level, program level, and organizational level. On all levels, new roles and practices are prescribed in order to work in an agile and Lean way.

Regarding the dynamics of routines, it is an interesting contradiction that even though a fundamental value of agile ways of working is to adapt and improve continuously, SAFe routines are prescribed in a rather detailed manner, very seldom to encourage tailoring. Both researchers (Alqudah & Razali, 2016; Stojanov et al., 2015) and agile practitioners (Schwaber, 2013; Maximini, 2015) have criticized SAFe for being too strict and formal, based on detailed prescriptions in the framework. Below, the three routines intended for inter-team coordination in SAFe on team level are described.

2.1 PI planning

On the SAFe website, this practice is explained in the following way: “Program Increment (PI) Planning is a cadence-based, face-to-face event that serves as the heartbeat of the Agile Release Train (ART), aligning all the teams on the ART to a shared mission and Vision” (ScaledAgile, 2018). To further state the importance of this practice, it is clarified that “PI planning is essential to SAFe: If you are not doing it, you are not doing SAFe” (ibid.). The role of the person in charge of facilitating PI planning is called a Release Train Engineer (RTE) who coordinates and prepare the event.

In other words, PI planning means a common way for a number of teams (an ART) working towards a common goal to make a shared plan, aligning all the teams work for a specific time period into the future. Regarding the length of this set time period, the SAFe prescription is somewhat flexible in ex-

pressing “PIs are typically 8 – 12 weeks long. The most common pattern for a PI is four development Iterations, followed by one Innovation and Planning (IP) Iteration” (ScaledAgile, 2018). Development Iterations are also called *sprints*, defined in the Scrum framework (Sutherland & Schwaber, 2013) as a time-box of one month or less during which a "done", useable, and potentially releasable product increment is created. Each sprint starts with a sprint planning day where the final plan for the upcoming sprint is decided which means that flexibility is supported even within the PI to make changes in plans. But, time-length of a PI period is not intended to vary between different PI:s, as this quote explains: “PI planning occurs on a fixed cadence” (ibid.). Does this mean that the PI is the same as a release plan, i.e., should the final day of a PI be the day when all software during a PI is released? It can be but as the text on the website clarifies: "The cadence for the PI can be different from the release cadence" (ibid.). According to SAFe, a PI planning workshop should take place over two days according to a recommended standard agenda as shown in Fig. 1.



Figure 1. Standard agenda for PI planning workshop from the ScaledAgile (2018) website.

"During PI planning, the teams estimate what will be delivered and highlight their dependencies with other Agile teams and trains" (ScaledAgile, 2018) but by looking at the proposed schedule depicted in Fig. 1 a lot of time is also supposed to be spent on presentations and information sharing regarding business context, visions, practices, and planning context. From an ostensive perspective, the PI planning routine is not only to plan and highlight dependencies between teams and other trains, but also to inform and clarify the current context in terms of the business, product, and architecture. Also, with the final IP iteration in a PI intended for reflections and improvement, the PI planning routine is an area to propose new ideas and insights which could become planned work in the future.

2.2 Scrum of Scrums

In a case study report by Sutherland (2001), who introduced the Scrum framework at the company IDX Systems in 1996, Scrum of Scrums (SoS) was used to coordinate emerging dependency issues since hundreds of developers worked on dozens of products. The SoS routine is described as a weekly meeting between all Scrum Masters (SM) in a product line to discuss and solve dependency issues between teams. But the SoS routine adopted in this form has been widely criticized (e. g. Paasivaara et al., 2012). By allowing only SMs to the SoS, there is a risk for talking about problems rather than solving them. Practitioners have stressed that the SoS should be a place for resolving coordination issues, not a meeting for managers, and put forth the need for allowing any team member with a coordination issue to participate in the SoS (ibid.).

In SAFe, the routine is described as an occasion in which the "RTE, Scrum Masters, and others (where appropriate) meet to review their progress toward milestones, program PI objectives, and internal dependencies among the teams. The meeting is timeboxed for less than 30 minutes and is followed by a 'meet after' to solve any problems" (ScaledAgile, 2018). From an ostensive aspect, it seems as the purpose is both for status updates as well as for taking decisions on emerging inter-team coordination issues. Regarding cadence, the SoS routine is mentioned twice. First, the routine is described as "a

weekly (or more frequently, as needed) ... meeting" and, second, mentioned as "Twice a week recommended" (ScaledAgile, 2018). Although not contradicting, it is somewhat surprising that the first notion of the routine being a weekly meeting is after that pointed out as a best practice to be conducted twice a week.

2.3 Program board

The program board is one important outcome of the two-day PI planning workshop, and it is an artifact which "highlights the new feature delivery dates, feature dependencies among teams and with other ARTs, and relevant Milestones." The example of a program board provided by SAFe can be seen in Fig. 2.

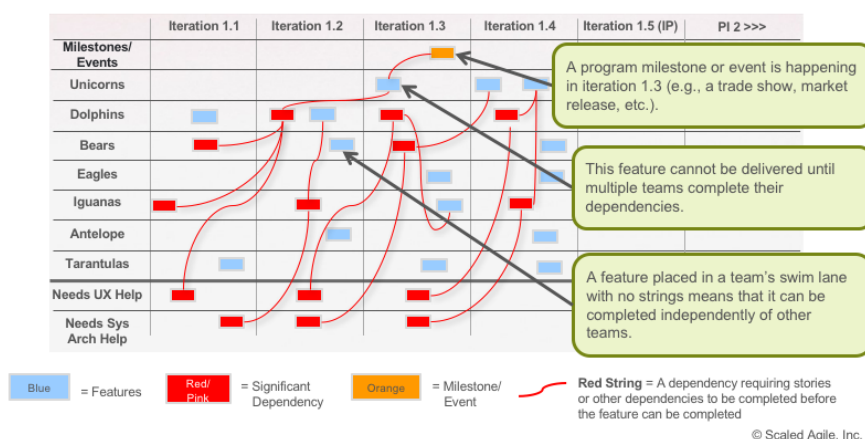


Figure 2. Example of a Program board from the ScaledAgile (2018) website.

As Fig 2. exemplifies, the program board should contain milestones, features and significant dependencies between features. There is no definition of what "significant" entails other than that a feature cannot be delivered until dependent features are completed.

Teams are supposed to update the board continuously during the PI planning workshop as soon as a new feature is planned for. At the end of the planning days, the program board should, therefore, be complete and show all known significant dependencies for the upcoming PI. Then, as work on the new PI begins, SAFe states that the "program board is often used during the Scrum of Scrums meetings to track dependencies, or it may not be maintained (manually) after that time. This depends upon the Agile project management tooling in place and the needs of the ART" (ScaledAgile, 2018). This description is a clarification from the previous version of SAFe, 4.0 (ScaledAgile, 2016), which only expressed that "the program board may or may not be maintained after that time," meaning after the PI planning workshop.

3 Routine dynamics as an analytical lens

In deciding on purposeful process theories of change (Van de Ven, 2013), selecting a routine perspective is relevant since many researchers (Feldman and Pentland, 2003; Becker, 2004) recognize that routines are dynamic processes that create both stability and change. Feldman and Pentland define organizational routines (hereafter, simply *routines*) as "repetitive, recognizable patterns of interdependent actions, carried out by multiple actors" (2003, p. 94). Routines are closely related to practices and processes, but from an analytical perspective, they differ. Compared to process analysis, analyzing routines is not only about connecting inputs with outputs but more about the internal structure. A process could also be accomplished without human intervention, such as a chemical or biological process. Compared to practices, a practice can be performed by an individual while routines involve multiple actors and also, routine analysis focus on different aspects or elements of the routine.

The three fundamental aspects of routines used for analysis are ostensive, performative and artefacts (Pentland and Feldman, 2005). The ostensive element or the abstract patterns (Pentland et al., 2012) of a routine is its intent, its rules, a purpose based on thoughts, emotions, and behavior. The ostensive aspect may vary in what actions are considered necessary and appropriate and can differ between individuals, communities, and organizations. Also, a method or a framework such as SAFe often prescribes what the intended, ostensive, aspect of a routine is supposed to be. Divergent understandings are probably more the norm than the exception (Pentland and Feldman, 2005).

The performative element of a routine or its concrete level (Pentland et al., 2012) are the actual, specific actions performed by organizational members. Finally, artefacts are the “physical manifestations of the organisational routines” (Pentland and Feldman, 2005, p. 797) like visual tools, documented procedures, and written rules. You might think that rules and tools determine the patterns of action that make up the performative aspect of a routine, but the practical effect of any particular rule or procedure is often quite remote from its original design or intention (Pentland and Feldman, 2005).

Feldman and Pentland (2003) explain that in many cases the overarching pattern of a routine may remain relatively stable while specific parts of the routine pattern may show considerable change. Miner et al. (2008) explain that routines contain smaller parts: the sub-routines. Although the overall pattern of a routine may remain stable, the different components (the sub-routines) may change. This also means that the actors of a routine sometimes can choose from several sub-routines for the specific implementation of a given routine, which may lead to variations. Also, Felin et al. (2012) highlight the role of micro-level phenomena, such as the individuals, the structures and the social processes, which also could be sources for variation in routines. All these examples explain the duality of routines as sources of stability and change. Routines are dynamic since they exist in the process of (re)production over time and space and through the on-going effort of actants which could be both people and things (Feldman et al., 2016). The term “routine dynamics” has come to stand for the study of the dynamics within and across routines as they are enacted in practice (ibid.). The purpose of creating artifacts may be the creation of a new routine (March et al., 1993). Artefacts are the “physical manifestations of the organisational routines” (Pentland and Feldman, 2005, p. 797) and, as explained by the authors, an example is items that deliberately attempt to capture or prescribe the routine, such as formal rules.

Visualizing work and progress are profound principles in agile ways of working (Beck et al., 2001) and new frameworks, such as SAFe, prescribes several visual tools. These artefacts contribute to the implementation of inter-team coordination routines since an artefact can support the implementation of new routines by formalizing procedures or rules to follow (March et al., 1993). Besides the actual visual, physical objects in the framework, the collection of coordination routines of SAFe itself can be conceptualized as a meta-artefact; an artefact containing a collection of artefacts, rules, and procedures, providing a general solution to a defined class of problems such as coordination between teams.

4 Method

A multiple-case studies design allows researchers to explore a phenomenon by using a replication strategy. Yin (1994) compares a replication strategy with conducting a number of separate experiments on related topics. A replication strategy is performed in two stages: first, a *literal* replication stage where previous studies of cases are selected to obtain similar results. Second, a *theoretical* replication stage, where cases are selected to explore and confirm or disprove patterns identified in the initial cases. In this study, the method descriptions of SAFe have been used to define what Yin (1994) would call a literal replication (the first stage). Earlier writings from the method originator Leffingwell (2007) together with the book describing SAFe version 4.5 (Leffingwell et al., 2017) as well as the website containing both version 4.0 (ScaledAgile, 2016) as well as the latest version 4.6 (ScaledAgile, 2018) have been the sources used for the literal replication. For the second stage, the theoretical replication, several case organizations have been studied. Criteria for the chosen organizations were to have prior experience of agile ways of working before implementing SAFe. To be able to collect meaningful data, it was important to study the organizations from the very first days of implementing the SAFe framework, and follow both the performative and ostensive aspects of the routines for coor-

dination for a longer period. Three of the contacted organizations fulfilled the criteria mentioned above for data richness. Fortunately, there are no rules about how many cases are required to satisfy the requirements of the replication strategy. The number of cases is sufficient if the results “provide compelling support for the initial set of propositions” (Yin, 1994, p. 46). Yin goes even further by explaining that since the multiple-case studies approach does not rely on the type of representative sampling logic used in survey research, “the typical criteria regarding sample size are irrelevant” (p. 50). It is more important that the chosen cases are in organizations in which the phenomena can be studied.

Feldman and Pentland (2008) recommend observations, interviews and document analysis as appropriate data collection methods to triangulate and capture routine elements and their relationships. Patton (2002) put forth the importance of using observations to examine routines, and Pentland (2003) specifically recommends observation to capture the performative element of routines. Observation should take place in situations where the consequences and the intent of the actions are clear for the observer.

4.1 Case descriptions and data collection

This study was conducted in three different organizations. The real names of the organizations have been anonymized, but the cases will be referred to as Auto, Gov, and Bank. The organizations have used agile ways of working for four to six years, with self-organized autonomous teams working side by side, before implementing SAFe. All three organizations decided to adopt routines for improved coordination between teams and started implementing SAFe during the beginning of 2017. Auto was first, starting in January while Gov and Bank began in April.

The Auto case is a product development department in an organization within the European automotive industry which mainly develops software but to some extent hardware as well. The observed department is organized in 20 to 25 cross-functional teams, divided into three different “value streams” or ARTs to use SAFe terminology (ScaledAgile, 2018). The department has grown, hence the different numbers of teams. The case Gov is a project where SAFe was implemented intended as a pilot project in a large Swedish Government Agency with the aim of finding best practices for implementing and tailoring SAFe in other parts of the organization. Gov consists of seven teams working in one ART. The Bank case is a department in one of the major business banks in Sweden consisting of seven teams that work together in one ART. Bank decided to implement SAFe because a new software platform was being developed which would increase the number of dependencies between all teams in the department.

In this study, attendance to coordination meetings and planning sessions, as well as visits at the offices to observe daily activities, provided observation opportunities. Data from observations contained field notes and photographs taken during meetings and planning sessions. Most photos were taken during meetings where MS Powerpoint presentations accompanied the oral presentations.

Semi-structured interview protocols were used in interviews conducted with informants from various hierarchical levels such as team members ($N=14$), scrum masters ($N=4$), product owners ($N=3$), RTEs ($N=4$), agile coaches ($N=2$), and a product manager ($N=1$). The data collection process ended when further interviews did not reveal any new data and theoretical saturation was perceived as being reached (Bryman and Bell, 2003). The protocols were divided into two sections to capture whether responses concerned either the performative or ostensive elements of routines. In particular, notes were kept to record variation in these elements.

Observations, using field notes and photographs, together with interview protocols and transcripts, were used for triangulation which altogether contributed to theoretical saturation. Reading archival records, mainly memoranda from meetings, also helped in capturing different elements of routines. Observations and interviews in the various cases are presented in Table 1. Data was collected from the starting point of implementing SAFe in the organizations (from January to April 2017) until November 2018, a period of almost two years.

Case organization	# of on-site visits	Hours of observation	# of interviews	Hours of interviews
Auto	6	196	14	1 h 48 min
Gov	5	113	6	6 h 12 min
Bank	6	70	8	7 h 32 min
Total:	17	379	28	25 h 32 min

Table 1. Observations and interviews.

After 379 hours of observations and have conducted 28 interviews, it was felt that theoretical saturation had been reached as very few new insights about the dynamics of coordination routines were found (Strauss and Corbin, 1994). At this point, only repetitive data was being collected, and it was possible to provide “some direction for operationalizing” (Bowen, 2008, p. 140).

4.2 Analysis

Although many routines and practices contribute to coordination between teams (Dietrich et al., 2013; Dingsøyr et al., 2018) in large-scale agile software environments, three routines were analyzed in this study: PI planning, SoS and the use of the program board. These three were chosen since they are intended explicitly for coordination according to SAFe (ScaledAgile, 2018). Other routines helpful for coordination, such as Communities of Practice, were excluded from analysis to allow for thick descriptions and to go more in-depth in the three chosen coordination routines.

First, the data were analyzed by coding field notes, together with photographs, based on the aspects of the routine elements involved (performative or ostensive), and the differences between the three cases. Anything describing *how* (performative) or *why* (ostensive) one of the three routines were conducted was coded accordingly. In this study, routines are analyzed from these two aspects, not from an artefact perspective per se. Instead, artefacts used are analyzed from the performative and ostensive perspective as well. Second, interviews and meeting records were analyzed and coded in the same manner. The observations and photographs of the daily activities were compared with the interpretations from the interview notes, transcripts and documents, and disagreements lead to further fact checking through complementary discussions and informal interviews during on-site visits.

Differences in variety, high or low, have been analyzed based on the classification defined in Pentland and Feldman (2005). For the ostensive aspect, the variety is considered as high if the explained purpose of a routine differs from the intended purpose, in this study as it is described in SAFe. Variety is also considered high if the purpose of a routine is understood differently by different roles in the case organization. For the performative aspect, variety is considered as high only if the routine was changed *deliberately* during the observed period, not by chance. In this study, either as a result of decisions in the team or because of decisions from stakeholders that caused changes in how the routine was performed.

5 Results

To present how inter-team coordination routines are performed at the three case organizations, a temporal aspect is sometimes needed since a routine “is only stable-for-now, at best” (Feldman et al. 2016, p. 510). Therefore, in this section, results displaying performative and ostensive aspects of PI planning, SoS and the use of the program board will be presented in narratives with some highlighted quotes from interviews and meetings.

The narratives present how routines were altered and tailored over time in the organizations. But, in an attempt to display a comparative overview of the results across the case organizations, a simplified summary of the differences between coordination routines from a performative aspect is presented in Table 2.

Routine	According to SAFe	Auto	Gov	Bank
PI planning	Set cadence, PI-plan every 8-12 weeks.	Set cadence, PI-plan every 10 weeks.	12-16 weeks.	Started with 5 weeks now set 9 weeks cadence.
	4*2 week sprints and IP sprint (2 weeks).	3*3 week sprints and IP sprint (1 week).	3-5 sprints of different length, no IP-sprint.	First: 2*12-days sprints. Now 4*12 and 2 days IP-sprint.
	Duration: 2 days.	Duration: 1,5 days.	Duration: 2 days.	Duration: From 0,5 to 1,5 days.
Scrum of Scrums	Twice weekly.	Varied from once a week to once per sprint.	Once a week.	Daily + "mid sprint review".
	Duration: 30 minutes and "meet after".	Duration: 20-30 min, no "meet after".	Duration: 30 min, no "meet after".	Duration: 15 min, sometimes "meet after".
	Attendees: RTE, SMs and possibly stakeholders.	RTE and SMs.	Managers, RTE and SMs.	RTE, SMs and some team members.
Program board	Contains: Feature delivery dates, feature dependencies and milestones.	Contains: Feature dependencies and milestones, not all features.	Contains: Feature delivery dates, feature dependencies and milestones.	Contains: Feature delivery dates, feature dependencies and milestones.
	May or may not be maintained during PI.	Not maintained during PI.	Not maintained during PI. Later abandoned.	Updated every sprint planning day and on "mid-sprint review."

Table 2. Differences in enacted routines from a performative perspective.

5.1 The PI planning routine

Auto decided from the very beginning to work in a strict ten-week cadence consisting of three sprints lasting for three weeks followed by an IP sprint during the tenth week where PI planning was performed. The reason for having a strict cadence was explained in the same way by two RTEs and three team members: by having a strict rhythm, team members will be better at predicting and estimating and that will help Auto to be more precise in predictions towards stakeholders. This reason is as prescribed by SAFe which means, from an ostensive aspect a *low variety*. The managers and RTEs at Auto thought that one and a half day would probably be enough time for the teams to finalize their plans and decided to conduct the workshop using a little less time for each item than prescribed in the standard PI planning agenda. During the final on-site visit, when Auto held their tenth PI planning workshop, the overall format was still the same, but the contents of the PI planning workshop had been tailored all along based on feedback from all people in the ART. At Auto, the SAFe standard agenda is an artifact that set up the pattern of actions from an ostensive aspect but not from a performative aspect. Notably, the amount of time spent on presentations compared to team breakout time had changed. In the beginning, only 30 percent on average of the time was spent on team breakout planning, but this changed after criticism of spending too much time in meetings. During the last studied PI planning workshop, 60 percent of the time was spent on team breakout planning, and the format of the introductory presentations had changed a lot.

At Gov, the length of the PI was tailored based on the number of available man-hours: during summer PIs, where most people in Sweden are on holiday leave for a couple of weeks, PIs have been longer to compensate for less available man-hours in the teams. The similar condition goes for Christmas and Easter. So, from an ostensive aspect, what is important for a PI cadence at Gov has been the number of man-hours to be planned for, not the calendar cadence (a *high variety* compared to SAFe). The department manager at Gov has not allowed an IP sprint for the ART, only a short (less than two hours)

retrospective meeting and the two days for PI planning, following the standard agenda from SAFe. The reason for not allowing a sprint for inspection and adaption is that it was seen as a waste of time and money to spend so much time on non-productive activities. This was the view from management, and it was not shared with RTE, POs, and SMs who argued that Gov would benefit from an IP sprint. The ostensive aspect regarding PI cadence and planning differs both from what SAFe prescribes and between managers and team members and could, therefore, be considered being of high variety. The performative aspect, on the other hand, was of low variety since the standard agenda from SAFe was followed and performed in the same manner during all of the observed planning occasions with only minor changes. Actually, team members complained that planning, in the same manner, felt “non-agile” since the workshop was conducted in the same way each time and that plans were not updated during sprints, making the whole PI seem like a commitment, instead of being a plan with the intended flexibility in the start of each new sprint.

For the first PI planning planned at Bank, the department manager only allowed half a day for a PI planning workshop of which the agile coach at place responded (in anger) that with so little time, they would only be able to plan for two sprints. But after the anger subsided, the agile coach changed the way he thought about the PI planning workshop regarding learning: instead of learning to plan according to a strict cadence, the organization will learn faster about how to conduct PI planning workshops since they will have the next workshop as soon as five weeks after the first attempt. Shorter learning loops will cause faster learning, was the new logic as explained by the RTE.

“When we wanted to do the first PI planning, the reason to why we only planned for two sprints was actually because we were not allowed to have a longer planning workshop, only four hours. Then I thought ‘well, that’s actually great, [it] means [that] we will learn how to PI plan faster.’ That’s why we have expanded the PI bit by bit, instead of having a long fixed cadence.” (Agile Coach at Bank.)

With this new view of the PI length as a way to increase learning, expanding PI length each PI caused the performance of PI planning to variate a lot from the first up until the fourth PI planning workshop. During the first two PIs, only two sprints were planned, during PI planning number three, three sprints were planned for and at the fourth PI planning workshop, Bank finally decided on a fixed cadence of four sprints per PI for the future PIs to come. For this studied period, the performative aspects of PI planning meant a high level of variety, due to the changed ostensive aspect of PI planning. But even from the ostensive aspect, the variety was also high: several team members and SMs reported that their understanding of the reason for the short PI period only had to do with managers not wanting to allow more time for planning.

5.2 The Scrum of Scrums (SoS) routine

Auto admits that SoS meetings have not been prioritized in their implementation of SAFe.

“We started with SoS [meetings] once a week, then every second week and now once a sprint. They got stuck in just talking about what they had done. Status reporting. [We] tried to foster a mindset of focusing on risks in the team instead of babbling too much about what they had done.” (RTE at Auto)

Apart from how often SoS meetings were conducted, they were performed in the same way (low variety), letting one SM at a time addressing what they had done since the last meeting and if they had problems or possible problems, i.e., risks. From an ostensive aspect, RTEs viewed SoS meetings as an arena for highlighting risks while SMs saw it mainly as a place to report progress. None regarded the meeting as a place specifically to resolve dependency issues. With this difference in perspectives from different roles, the variety from an ostensive aspect could be considered as high.

At Gov, SoS meetings were conducted once a week on a strict cadence: Wednesdays at 10.30 to 11.00. Apart from RTEs and SMs, several managers and relevant stakeholders attended. The meeting was held virtually (via Skype), and anyone at Gov could listen in during the meetings.

“So, the agenda is [the] same as always. First, we have information from RTEs, our architect, product management, our test manager and the [Business Development] BD-team. Then every team presents information that could be important for other teams and dependencies. Then, we address risks.”

(RTE at Gov)

According to SAFe, the SoS is a meeting to help out between teams, not an information meeting. But at Gov, managers, and stakeholders participated to be able to inform the SMs about important decisions and status on areas surrounding the project. From an ostensive aspect, all parties at Gov viewed the SoS primarily as an arena for information sharing, and secondly as a meeting for highlighting risks and dependencies (high variety). From a performative aspect, the meeting had the exact same format every time (low variety).

Bank chose to conduct SoS meetings in the same format as the Daily standup meeting (Sutherland and Schwaber, 2013), with a fixed timebox of 15 minutes per day, always between 12.45 and 13.00. The time was chosen based on that many other kinds of meetings at Bank started at 13.00. The focus in the SoS meetings was primarily to help out between teams, and every SM mainly talked about what their team needed help with. Based on the obstacles in the single team, additional team members sometimes joined the SoS meeting in order to explain the details of the problems further. Although the time frame was the same, this meant that there was a high variety of how the meeting was performed. For example, if some issues could not be solved during the SoS meeting, parties involved stayed to resolve issues in a “meet after,” whenever necessary. Also, Bank chose to add another routine to follow up progress: the “mid-sprint review” always conducted on the sixth or seventh day of the twelve-day sprint. In this meeting, every SM reported and discussed the progress of the teams PI objectives and the program board was updated. From an ostensive aspect, all parties at Bank viewed the SoS meeting as a place to solve issues between teams based on dependencies between teams just as prescribed in SAFe (low variety).

5.3 Program board

Auto decided from the very start only to use the program board to highlight features with dependencies between teams and other departments, not to show all features. Significant milestones were also displayed. This was performed in the same way during the whole observed period of time.

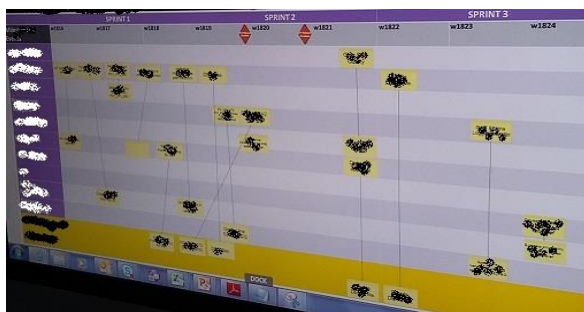


Figure 3. A program board showing feature dependencies using the online tool iObeya at Auto.

Instead of using a physical board, Auto used the online tool iObeya as a program board as can be seen in Fig. 3. At Auto, the program board was only updated during the PI planning and was not kept up to date during the PI. The variety, from both the ostensive and performative aspect, could be considered as low.

During the first PI planning workshop at Gov, all milestones, features, and dependencies were presented on the program board. The board showed that *all* of the features had dependencies either with other teams or other departments. The reason for this was due to the ostensive aspects of what dependency really means. SMs and team members presented a dependency on the board when the team only needed information from a person outside the team while RTEs and stakeholders defined a dependency as a need for a specific systems function to be fulfilled before the dependent function could be developed. After a brief discussion at the end of the PI planning workshop, all parties agreed on the latter definition which took away almost all dependencies, leaving only eleven dependencies to other departments. Suddenly, there were no dependencies towards another team within the same ART at Gov.

The program board was not updated during the PIs, and since so few dependencies between teams were recognized during PI planning, Gov abandoned using the program board during their fourth PI planning workshop. Instead, teams recorded dependencies in their own team plans. At Gov, both the ostensive and performative aspect could be considered as high.

At Bank, the program board contained all features, dependencies between features and milestones. Instead of updating the program board during SoS meetings, as suggested by SAFe, Bank updated the board during the first day of every new sprint and during the “mid-sprint review” where implemented features were color-coded. From a performative aspect, there was a low variety in how the program board was used, and all roles had a similar view of why the program board, and its content, should be used. From an ostensive aspect, this could be considered as low variety since the purpose of using it correlates with the intention put forth in SAFe.

5.4 Routine variety

The results based on a classification defined in Pentland and Feldman (2005) are presented in Table 3.

Routine	Ostensive	Performative
PI planning	Auto: Low variety Gov: High variety Bank: High variety	Auto: High variety Gov: Low variety Bank: High variety
Scrum of scrums (SoS)	Auto: High variety Gov: High variety Bank: Low variety	Auto: High variety Gov: Low variety Bank: High variety
Program board	Auto: Low variety Gov: High variety Bank: Low variety	Auto: Low variety Gov: High variety (and later abandoned) Bank: Low variety

Table 3. Differences in variety from ostensive and performative aspects.

6 Discussion

The agile manifesto (Beck et al., 2001) promotes responsiveness to change and to reflect, adjust and tune routines regularly. Dingsøyr et al. (2018) also report from a large-scale agile program that routines for coordination are not static, but dynamic and change over time. At the same time, the SAFe framework prescribes routines to be used both from an ostensive aspect (why) and a performative aspect (how) with very little advice on change or tailoring over time. One critique towards SAFe is that being too prescriptive will constrain the continuous development of the teams (Alqudah & Razali).

Researchers argue that routines are generative, dynamic systems, not static objects (Feldman and Pentland, 2003; Pentland and Feldman, 2005) and that routines are continuously emerging systems with internal structures and dynamics. The internal structure of a routine can vary between very stable and continuously changing, depending on the circumstances. Also, it is possible for one aspect of a routine to be relatively stable while the other aspect is relatively variable and, according to Pentland and Feldman (2005), routines that look ‘more variable’ from an ostensive perspective are ‘less variable’ from a performative perspective and vice versa.

In this study, routines for inter-team coordination have been investigated at three case organizations, and the variety has been studied from an ostensive and performative aspect. Contrary to the findings from the six cases presented in their study (Pentland and Feldman, 2005) cases with high performative variety do not have low ostensive variety and vice versa. In fact, in three of the investigated areas the ostensive aspects have high variety as well as the performative aspect, and in two of the cases, both the ostensive and performative aspects are low. This indicates that the empirical relationship between ostensive and performative aspects might not be as atypical as proposed in Pentland and Feldman

(2005). This is an important theoretical contribution and whether there is an atypical relationship in variety between ostensive and performative aspects could be an important area for further studies.

As the results show, although we might think the ostensive view of a routine is the same, such as PI planning being only intended for planning, there might be several underlying ostensive aspects that might differ. And even with a common, at least as presented, ostensive reason for the pattern of the routine, different underlying ostensive patterns are possible. The reasons might differ compared to how it is defined in SAFe, between organizations and between groups within the same organization.

The misalignment between corporate culture and the coordination routines might be one of the reasons for the different underlying ostensive aspects of the implemented routines (Bertels, Howard-Grenville, and Pek, 2016). As described in the Gov case, the IP iteration at the end of a PI, which is supposed to be a time for reflection and learning to improve work, was seen as waste. The PI was, from Gov's cultural perspective, only a period of planned work, not an area for idea generation and improvement of work routines. Bertels et al. (2016) conclude from their study that implementing a routine can lead to more and complex patterns of action than those originally intended, due to misalignments between the routine and the existing corporate culture. When an organization integrates a routine that is a poor fit, employees tend to continue to draw on familiar cultural principles of action which shapes how the routine is performed. At Gov, this was also evident in the expressed views from team members regarding the rigid view of the PI, more seen as a "non-agile" committed plan than a plan with several sprints with the intended flexibility to change during sprint planning at the beginning of each sprint.

This was also evident at both Auto and Gov where the SoS meetings were more performed as a traditional "information and status meeting" than as the intended purpose of being a routine for solving emerging dependency issues (ScaledAgile, 2018). Stojanov et al. (2015) also highlight this potential cultural problem of not having reached enough "agile maturity" in their proposed maturity model for implementing SAFe.

6.1 Practical contributions

Several agile practitioners (Schwaber, 2013; Maximini, 2015), as well as researchers (Alqudah & Razali, 2016; Stojanov et al., 2015), have criticized SAFe for being too strict and formal, based on detailed prescriptions in the framework (ScaledAgile, 2018). This paper presents several ways of tailoring the coordination routines, showing that managers do not need to implement SAFe by the book, thereby avoiding the challenges of formality or strictness.

The practical contribution from this study is to display and exemplify tailoring decisions when an agile organization needs to scale, forcing several teams to cooperate due to dependency issues. An example of tailoring the PI planning routine is, as identified at Bank, to start with shorter PIs to teach the organization to conduct PI planning faster and to set a fixed PI period after a number of PIs have been performed. For a manager, this is an alternative way of how to implement the PI concept in the organization. When looking at Auto and Gov, it shows an example of different logics regarding what cadence really is. At Auto, the number of weeks, in this case ten weeks, became the length of the PI, while at Gov, the available man-hours decided the length of the PI which, e.g., increased the number of weeks during Christmas holidays and the summer months.

For the actual PI planning days, the standard agenda from SAFe (ScaledAgile, 2018) prescribes much time spent in presentations and not as much in team break-out time. In all three cases, along the way more and more time was set aside for team break-outs, at Auto from 30 percent in the first PI planning workshops to 60 percent in the later. A suggestion to managers is, from these cases, to allow more time for team break-outs already from the start.

In a study by Paasivaara et al. (2012) the prescribed SoS meeting format was challenged and mainly seen as a waste in a large agile organization. The tailored solution in their study was to have feature-specific SoS meetings for 3-5 teams working on one large feature. This study presents alternative ways in tailoring the SoS routine, both from ostensive and performative aspects that a manager could benefit from. From a performative perspective, this study shows alternatives of cadence: Auto started out using the SoS only once a sprint but accelerated the use of the meetings as benefits became obvious and

later settled for a weekly rhythm. Gov chose a weekly cadence from the start but conducted the meeting virtually to let anyone listen in on these meetings which led to more stakeholders to participate. They viewed the meeting both as an essential information channel (since the people were not collocated and did not meet each other on a daily basis) as well as a meeting to solve risks and impediments. Bank chose a daily cadence, limiting the time to only 15 minutes. With such a short time frame for discussing and solving issues, there was no time for reviewing actual progress, only to highlight impediments. Therefore, Bank added a practice they named “mid-sprint review” practice, to be able to highlight progress on the program board. These three ways of implementing SoS meetings are useful tailoring alternatives to be used by a manager when scaling up an agile organization.

The Program board could also be set up in different ways, and a digital tool that could be used is the iObeya-board. Bank added the “mid-sprint review” where they not only showed dependencies but also used the Program board to highlight progress by color-coding already implemented features.

6.2 Limitations

The study described in this paper is to a large degree based on descriptions and responses from people working in the three case organizations. There is, of course, a risk in taking what is being said for granted. To mitigate the risk of making faulty conclusions by over-interpreting what respondents said, much time has been spent on observations, to see what they actually do, and informal discussions with people working in different roles during on-site visits.

The coding and classification decisions regarding the level of variety for the ostensive and performative aspect in the cases were double-checked for validation, and in cases of possible different interpretations, the aspects were discussed thoroughly with a research colleague. Such a pairwise activity adds to the reliability of the study.

References

- Alqudah, M. and R., Razali (2016). A review of scaling agile methods in large software development. *International Journal on Advanced Science, Engineering and Information Technology*, 6(6), 828-837.
- Beck, K., Beedle, M., Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jefferies, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., and D. Thomas (2001). *Manifesto for Agile Software Development*. URL: www.agilemanifesto.org (visited on 10/10/2018).
- Becker, M.C. (2004). “Organizational routines: a review of the literature.” *Industrial and Corporate Change* 13(4), 643–677.
- Bertels, S., Howard-Grenville, J. and S. Pek (2016). “Cultural Molding, Shielding, and Shoring at Oilco: The Role of Culture in the Integration of Routines”. *Organization Science* 27(3), 573-593.
- Boehm, B. and R. Turner (2005). “Management challenges to implementing agile processes in traditional development organizations.” *Software IEEE* 22(5), 30–39.
- Bowen, G. A. (2008). “Naturalistic inquiry and the saturation concept: A research note.” *Qualitative Research* 8(1), 137-152.
- Bryman, A. and E. Bell (2003). *Business Research Methods*. Oxford: Oxford University Press.
- Dietrich, P., Kujala, J. and K. Artto (2013). “Inter-Team Coordination Patterns and Outcomes in Multi-Team Projects.” *Project Management Journal* 44(6), 6-19.
- Dingsøyr, T., Moe, N. B. and E. A. Seim (2018). “Coordinating Knowledge Work in Multi-Team Programs: Findings from a Large-Scale Agile Development Program.” *Project Management Journal* 49(6), 64-77.
- Feldman, M.S. and B.T. Pentland (2003). “Reconceptualizing organizational routines as a source of flexibility and change.” *Administrative Science Quarterly* 48, 94–118.
- Feldman, M.S. and B.T. Pentland (2008). “Issues in empirical field studies of organizational routines.” In: *Becker, M.C. (Ed.), Handbook of Organizational Routines*. Edward Elgar Publishing.

- Feldman, M. S., Pentland, B. T., D'Adderio, L. and N. Lazaric (2016). "Beyond routines as things: Introduction to the special issue on routine dynamics." *Organization Science* 27(3), 505-800.
- Felin, T., Foss, N. J., Heimeriks, K. H. and T. L. Madsen (2012). "Microfoundations of routines and capabilities: individuals, processes, and structure." *Journal of Management Studies* 49(8), 1351–1374.
- Hoegl, M., Weinkauff, K. and H. G. Gemuenden (2004). "Interteam coordination, project commitment, and teamwork in multiteam R&D projects: A longitudinal study." *Organization Science* 15(1), 38–55.
- Leffingwell, D. (2007). *Scaling Software Agility: Best Practices for Large Enterprises*. Boston Massachusetts: Addison Wesley.
- Leffingwell, D., Yakyama, A., Knaster, R., Jemilo, D. and I. Oren (2017). *SAFe reference guide: Scaled Agile Framework for Lean Software and Systems Engineering*. USA: Addison Wesley.
- March, J. G., Simon, H. and H. S. Guetzkow (1993). *Organizations*. Cambridge, Massachusetts: Blackwell.
- Maximini, D. (2015). *A critical view on SAFe*. URL: scrumorakel.de/blog (visited on 10/9/2018).
- Miner, A.S., Ciuchta, M.P. and Y. Gong (2008). "Organizational routines and organizational learning." In: *Becker, M.C. (Ed.), Handbook of Organizational Routines*. Edward Elgar Publishing.
- Paasivaara, M., Lassenius, C. and V. T. Heikkilä (2012). "Inter-team coordination in large-scale globally distributed scrum: Do Scrum-of-Scrum really work?" In: *Proceedings of the ACM-IEEE international symposium on Empirical software engineering and measurement*, 235-238.
- Patton, M.Q. (2002). *Qualitative Research and Evaluation Methods*. Thousand Oaks, California: Sage.
- Pentland, B.T. (2003). "Conceptualizing and measuring variety in the execution of organizational work processes." *Management Science* 49(7), 857–870.
- Pentland, B.T. and M.S. Feldman (2005). "Organizational routines as unit of analysis." *Industrial and Corporate Change* 14, 793–815.
- Pentland, B.T., Feldman, M.S., Becker, M.C. and P. Liu (2012). "Dynamics of organizational routines: a generative model." *Journal of Management Studies* 49 (8), 1484–1508.
- ScaledAgile, Inc. (2016). *Scaled Agile Framework 4.0*. URL: v4.scaledagileframework.org (visited on 10/9/2018).
- ScaledAgile, Inc. (2018). *Scaled Agile Framework 4.6*. URL: www.scaledagileframework.org (visited on 10/10/2018).
- Schwaber, K. (2013). *UnSAFe at any speed*. URL: kenschwaber.wordpress.com/2013/08/06/unsafe-at-any-speed/ (visited on 11/10/2018).
- Stojanov, I., Turetken, O. and J. J. Trienekens (2015). "A maturity model for scaling agile development", In: *Proceedings of 41st Euromicro Conference on Software Engineering and Advanced Applications*, 446-453.
- Strauss, A. and J. Corbin (1994). "Grounded theory methodology: an overview." In: *Denzin, N.K., Lincoln, Y.S. (Eds.), Handbook of Qualitative Research*. Thousand Oaks, California: Sage.
- Sutherland, J. (2001). "Agile Can Scale: Inventing and Reinventing SCRUM in Five Companies." *Cutter IT Journal* 14(2), 5-11.
- Sutherland, J., and K. Schwaber (2013). *The scrum guide*. URL: www.scrumguide.org. (visited on 1/11/18).
- Van de Ven, A. (2013). "Process theories of change." In: *Kessler, E.H. (Ed.), Encyclopedia of Management Theory. Vol. 2*. SAGE Publications, Inc., Thousand Oaks, California, 612–616.
- VersionOne (2018). *11th Annual State of Agile™ Report*, URL: stateofagile.versionone.com (visited on 11/10/2018).
- Vlietland, J. and H. van Vliet (2015). "Towards a governance framework for chains of Scrum teams". *Information and Software Technology*, 57(1), 52-65.
- Xu, P. (2009). "Coordination in large agile projects." *The Review of Business Information Systems*, 13(4), 29-43.
- Yin, R. K. (1994). *Case Study Research: Design and Methods, 2nd ed.* Sage publications, Thousand Oaks.

