

## Method<sup>1</sup>

To investigate our research question—*how is the inter-team coordination strategy impacted by a change from the first- to second-generation large-scale agile development methods*—we have designed a longitudinal embedded explanatory case study (Runeson and Höst 2009; Yin 2018). The systematic literature review of large-scale agile methods shows that ‘purposefully designed longitudinal studies on the adoption and application of large-scale agile methods are rarely seen in the existing literature’ (Edison et al. 2021). We draw on previously established theories on coordination, mainly from management science, and from prior studies of inter-team coordination in large-scale agile development. We position the study as a positivist case study seeking to explain the impacts of a change by drawing on prior theory to define a set of novel propositions. In the following, we describe the research design, the procedures for data collection and the data analysis. The main limitations are discussed in Section 5.5.

### 1.1 Case study design

The objective of the present study was to increase the understanding of coordination in large-scale agile development, particularly to empirically examine strategies for inter-team coordination. This means that we have not focused on coordination at the team level. Prior studies have identified changes in coordination mechanisms over time, but as Edison et al. (2021) found, few longitudinal studies have been conducted.

The case is a very large-scale agile development programme. A programme involves a temporal organisation, which differs from a permanent software development organisation in that many participants will work for a shorter period. The case was selected as one of several large-scale software development projects followed in a research project. The criterion for selecting the case was that it should be an extreme case for coordination in that it had a high number of development teams (what we describe as a very large-scale agile development programme) (Dingsøyr et al. 2014). The programme had 200 participants at the most, with about 130 working in 10 development teams and in programme organisation. The programme was co-located, which meant that we did not need to focus on topics related to sociocultural distance (Ågerfalk and Fitzgerald 2006) or distributed agile development (Šmite et al. 2010).

---

<sup>1</sup> Excerpt from Dingsøyr, T., Bjørnson, F. O., Schrof, J., and Sporse, T., "Transitioning from a first- to a second-generation large-scale agile development method: A longitudinal explanatory case study of coordination in a very large development programme," *Work in progress*, 2022.

We describe the case as extreme for the following reasons. The first is its size. Second, the programme is also an extreme case of a large-scale agile development method in the initial choice of a first-generation large-scale agile development method which is more oriented towards plan-based development than, for example, the Perform programme (Dingsøyr et al. 2018a). Our case was more oriented towards plan-based development in that it had two projects, *business* and *development*, with formal handovers between them. When reorganising, the programme chose to work with continuous deployment and autonomous teams, which we argue are more in line with agile principles (Baham and Hirschheim 2021) than some of the second-generation large-scale agile development methods that, for example, prescribe a number of roles.

The unit of analysis is inter-team coordination strategies in the central part of the programme. The original plan was to focus on how the programme adjusted its coordination strategies over time. The programme was planned with three releases, and the plan for data collection focused on documenting practices and perceptions of practices amongst different groups for each release. However, the programme did reorganise, which gave us a unique opportunity to study changes in coordination after reorganisation. As a consequence, we revised the data collection procedures, as described below. We focused on two phases of the programme in which 10 development teams worked in parallel: one phase using a first-generation large-scale agile development method and another phase using a second-generation large-scale agile development method.

We asked to follow the programme from early 2017 and were granted access to interview its participants, read relevant documents and observe meetings. We were also given a series of briefings about the organisation and the progress of the programme.

The study was part of a more extensive work in which we already obtained approval from the Norwegian Centre for Research Data (reference 848084). We secured informed consent from the interview participants and ensured that the data used in the reports are not traceable to individuals and that we regularly gave feedback about the findings to the case participants.

## **1.2 Data collection**

We had to carefully consider our strategy for data collection. The programme was located in Oslo, but most of our research team members were located 500 kilometres away in

Trondheim. We therefore chose to organise regular visits to the case, in which three to four researchers would participate in the data collection and subsequent discussion. A PhD candidate partly contributed to the data collection and gave us much insight into the context by studying changes in the central IT department of the case organisation (Vestues 2021). The discussions after data collection were crucial in developing a collective understanding of the programme organisation and coordination challenges amongst the research team.

Data collection was conducted through individual interviews, group interviews, observation and collection of documents. We also held meetings with programme management to obtain an understanding of the organisation of the programme. Field notes were written after the meetings and observations.

We interviewed individuals in a variety of roles to understand coordination challenges and practices, as shown in Table 3. Our primary focus was on software development practices, and most of our informants had roles related to development; however, we also interviewed several individuals in other roles to understand programme organisation. The interview guides were revised from a previous study (Dingsøyr et al. 2018a; Dingsøyr et al. 2018b) (see Appendix 1). These guides focused on coordination challenges and practices, as well as on contrasting between work on releases. The questions were mainly open and phrased in a language familiar to the respondents, such as ‘What dependencies do you have on other teams? Examples?’ and ‘How do you manage dependencies?’ We made minor changes in the last round of interviews to focus on the effects of work reorganisation, which we call a transition from the first- to second-generation large-scale agile development methods.

We visited the case three times over two days. We were three to four researchers conducting semi-structured interviews in parallel, which were followed by a feedback session with our interpretation of what was said. During the visits, the first interviews were conducted by a pair of researchers to ensure consistency in the use of the interview guide. Later interviews were conducted by a single researcher. The interviews lasted from 24 to 120 minutes, typically around 30 minutes. These were recorded and transcribed for analysis. In total, we interviewed 39 informants—13 in December 2017, 12 in January 2019 and 13 in November 2019. We conducted another interview in January 2020 (see the participants’ roles in Table 3). As described in the limitations section, we could not interview participants from all teams during all visits, but we always interviewed people involved in development or test,

requirements engineering, architecture and project or programme management. In total, the interview material contained 456 pages of text.

We also invited key people from the programme to a workshop in October 2020, in which we established a timeline and brainstormed on what worked well and what could be improved. This workshop led to a separate article on key learning from the transformation process, written with practitioners from the case (Dingsøyr et al. 2021). We further conducted group interviews to discuss coordination and the requirements engineering process. The group interview on coordination included a project manager and a product owner from NAV and a project manager, an assisting project manager and the construction responsible for the development project. This two-hour interview was recorded and transcribed into a 42-page document.

*Table 3: Roles interviewed after the interview round and the phases in the programme*

Phase	Roles interviewed
First phase – round 1	Application architects (2), construction responsible, developer (2), functional architects (2), functionality responsible, scrum master (2), senior solution architect, solution manager, tester
End of first / start of second – round 2	Customer manager, central IT, developer, functional architect, product owners (2), programme owner, scrum masters (3), testers (2)
Second phase – round 3	Application architect, central IT (2), developer (2), programme manager, project manager deployment, product owner (3), project manager development, solution architect, test automation, tester

When negotiating access to the case, we avoided data collection in periods close to a release. Consequently, the first round of interviews was conducted during a relatively calm period and could be characterised by a neutral mood amongst the subjects. The second round was done after the initial shock of the reorganisation had settled, which was characterised by a mix of frustration and optimism. The third round was completed after the programme ended. One of the researchers wrote, *‘I’ve never interviewed people who are uniformly so happy with their situation!’* (Field notes, interview round 3)

We observed arenas for inter-team coordination, such as daily meetings and planning meetings, when visiting the case. To obtain further insight, we also facilitated retrospectives

on team coordination in November 2017 and one on the delivery model in January 2018. Apart from facilitating these two retrospectives, we did not intervene in how the programme organised inter-team coordination.

The documents included an initial overall plan (39 pages), the proposal to reorganise the programme (23 slides) and a document describing the new release pipeline (209 pages). We also obtained access to minutes from team retrospectives, which provided insight into what the teams perceived to work well and what was perceived as challenges.

### **1.3 Data analysis**

The data material was imported into a tool for qualitative analysis (Nvivo 12). All data material was anonymised, and files were given attributes that described the programme phase, role (where relevant) and which interview round the file belonged to (if relevant). The dominant data source used in the analysis was the qualitative interviews.

We used interview guides that gave us much context on the case. We first conducted descriptive and holistic coding on material related to coordination. Three researchers first coded the interviews independently and then compared the coding. This happened in a series of workshop meetings, and the goal was to align our understanding of the codes. The three researchers who participated in the coding all took part in the data collection and discussions of the case over time, and all had prior experience in coding similar material.

We further independently coded the material in more detail by using codes on coordination mechanisms, such as scrum of scrums meetings, issue trackers and artefacts, such as dependency maps. 22 codes were taken from previous studies on coordination in large-scale agile development (Dingsøyr et al. 2018a; Dingsøyr et al. 2018b). Coordination mechanisms were coded in broad groups using the coordination modes proposed by Van de Ven et al. (1976): the group, personal and impersonal modes. A sample text coded as ‘scrum of scrums’ and was related to the first phase of the programme was ‘... *we had scrum-of-scrums in which team leaders on each team met, and then we could raise issues with the other teams; we often identified if a team was waiting for another team, or if there were other causes for delay*’. We found 30 coordination mechanisms, as described in the results section.

We added coding about context, such as the descriptions of phases and product releases. The context information also included the codes used to describe ‘programme complexity and uncertainty’, ‘perceived project success’ and ‘coordination effectiveness’ (Figure 1).

After coding, we engaged in several activities for within-case analysis (Eisenhardt 1989). We first generated reports for the coordination mechanisms for each phase, which were tabulated. Langley (1999) described this as a temporal bracketing strategy to theorise from process data in which we see fairly stable processes within each phase. We can then examine how the context affects each phase and determine the consequences of the processes in the form of coordination efficiency. We had several discussions within the research team regarding the findings, and we compared our initial results with those of another study (Carroll et al. 2020). Furthermore, the initial findings were presented, first, to the informants in the case and, second, in an online open meeting at the IT department. We also wrote a report in Norwegian, in which we presented the context and organisation of the programme to obtain feedback on our understanding, and we developed a description for a narrative strategy (Langley 1999). Finally, in parallel with the analysis of the material for this article, the first author wrote a magazine article with the key participants from the case; the article summarised key learning from the transition (Dingsøyr et al. 2021). Overall, these activities helped us increase our understanding of the organisation and the challenges in the case.

Through this iterative process (Eisenhardt 1989), we built an explanation of coordination in the case. Following the steps described by Sjøberg et al. (2008), first, we drew on existing constructs from coordination theory from Van de Ven et al. (1976) and Strode et al. (2012), together with constructs from software engineering and agile software development. We also used our novel definitions of first- and second-generation large-scale agile methods. Second, by contrasting the two phases in the case study, we developed five novel propositions on coordination in large-scale agile development, which we suggest describe the impact of coordination in the transition from the first- to second-generation agile development methods. Third, the discussion shows our logical justification for the proposition, building on both our interpretation of the case study and our synthesis of related work presented in the background section. Fourth, we discuss the scope of the suggested propositions in Section 5.4. Finally, we discuss how the propositions might be tested in Section 5.5.

## References

- Ågerfalk, P. J., & Fitzgerald, B. (2006). Flexible and distributed software processes: Old petunias in new bowls? *Communications of the Acm*, 49(10), 26-34. doi:10.1145/1164394.1164416
- Baham, C., & Hirschheim, R. (2021). Issues, challenges, and a proposed theoretical core of agile software development research. *Information Systems Journal*, n/a(n/a). doi:<https://doi.org/10.1111/isj.12336>
- Carroll, N., Bjørnson, F. O., Dingsøy, T., Rolland, K.-H., & Conboy, K. (2020). *Operationalizing agile methods: examining coherence in large-scale agile transformations*. Paper presented at the International Conference on Agile Software Development, Eighth International Workshop on Large-Scale Agile Development.
- Dingsøy, T., Fægri, T., & Itkonen, J. (2014). What Is Large in Large-Scale? A Taxonomy of Scale for Agile Software Development. In A. Jedlitschka, P. Kuvaja, M. Kuhrmann, T. Männistö, J. Münch, & M. Raatikainen (Eds.), *Product-Focused Software Process Improvement* (Vol. 8892, pp. 273-276): Springer International Publishing.
- Dingsøy, T., Jørgensen, M., Carlsen, F., Carlström, L., Engelsrud, J., Hansvold, K., . . . Sørensen, K. O. (2021). From a First- to a Second-Generation Large-Scale Agile Development Framework “In-flight”: Key Lessons from the Transition Process. *Work in progress*.
- Dingsøy, T., Moe, N. B., Fægri, T. E., & Seim, E. A. (2018a). Exploring Software Development at the Very Large-Scale: A Revelatory Case Study and Research Agenda for Agile Method Adaptation. *Empirical Software Engineering*, 23(1), 490-520. doi:<https://doi.org/10.1007/s10664-017-9524-2>
- Dingsøy, T., Moe, N. B., & Seim, E. A. (2018b). Coordinating Knowledge Work in Multi-Team Programs: Findings from a Large-Scale Agile Development Program. *Project Management Journal*, 49(6), 64-77. doi:DOI: 10.1177/8756972818798980
- Edison, H., Wang, X., & K., C. (2021). Comparing Methods for Large-Scale Agile Software Development: A Systematic Literature Review. *IEEE Transactions on Software Engineering*, 1-1. doi:10.1109/TSE.2021.3069039
- Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review*, 14(4), 532-550.
- Langley, A. (1999). Strategies for Theorizing from Process Data. *The Academy of Management Review*, 24(4), 691-710.
- Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 14, 131-164.
- Sjøberg, D. I., Dybå, T., Anda, B. C., & Hannay, J. E. (2008). Building theories in software engineering *Guide to advanced empirical software engineering* (pp. 312-336): Springer.
- Šmite, D., Moe, N. B., & Ågerfalk, P. J. (2010). *Agility across time and space: Implementing agile methods in global software projects*: Springer Science & Business Media.
- Strode, D. E., Huff, S. L., Hope, B. G., & Link, S. (2012). Coordination in co-located agile software development projects. *Journal of Systems and Software*, 85(6), 1222-1238.
- Van de Ven, A. H., Delbecq, A. L., & Koenig Jr, R. (1976). Determinants of coordination modes within organizations. *American sociological review*, 322-338.
- Vestues, K. (2021). *Using digital platforms to promote value co-creation: A case study of a public sector organization*. (PhD), Norwegian University of Science and Technology.
- Yin, R. K. (2018). *Case study research and applications: Design and methods* (6th ed.). Thousand Oaks, California, USA: Sage.