From synchronizing to sustaining:

Coordinating across divergent temporal structures in collaborations of agile and non-agile units

Inauguraldissertation zur Erlangung des Doktorgrades

der Wirtschafts- und Sozialwissenschaftlichen Fakultät
der Universität zu Köln

2025

vorgelegt von

Carolin Müller, M.Sc

aus Gerolstein

Referent: Prof. Dr. Mark Ebers
Korreferent: Prof. Dr. Christoph Rosenkranz
Tag der Promotion: 09.12.2025

Table of contents

List of tables

List of figures

List of abbreviations

BaFin	Federal Financial Supervisory Authority
BAIT	Supervisory requirements for IT in financial institutions
CoRep	Common Reporting Framework
CEO	Chief Executive Officer
EBA	European Banking Authority
ECB	European Central Bank
ITE	Integrated test environment
IVR	Interactive voice response
MaRisk	Minimum requirements for risk management
MVP	Minimum viable product
PO	Product owner
TIP	Time, interaction, and performance theory
UAT	User acceptance testing
UX	User experience

Abstract

Coordination among interdependent actors is facilitated by shared temporal structures, yet actors often operate with divergent temporal structures. Such divergences create coordination challenges, conflicts, and inefficiencies, necessitating the need to understand how actors can coordinate across temporal differences. Research has identified two approaches: synchronizing practices, which aim to (re-)create shared temporal structures, and sustaining practices, which preserve differences in temporal structures while enabling coordination. Prior research has largely focused on synchronizing but overlooked its costs and negative consequences as well as the agentic role of non-time-setting actors in decision-making. Research on sustaining practices is still in its infancy, with only a few sustaining practices uncovered that prove effective only in the specific contexts in which they have been identified. Moreover, the complementarity and the dynamics between these approaches remain largely unexplored.

This dissertation addresses these gaps by investigating how actors with divergent temporal structures coordinate their interdependent activities. The empirical context comprises eight case studies of collaborations between agile and non-agile units in banking and insurance organizations, a context that provides both substantial temporal differences and strong interdependencies that necessitate coordination. It draws on data from 91 interviews, 256 participatory observations, and 642 pages of archival documents.

The findings advance understanding in three ways. First, they reveal previously underexplored drawbacks of synchronizing practices, highlighting how actors deliberately choose to engage in or resist synchronization. Second, they extend sustaining practices by uncovering buffering and harmonizing practices. Third, they highlight the dynamic nature of temporal structuring, showing how actors continuously integrate synchronizing and sustaining practices to effectively coordinate over time.

Overall, this dissertation contributes to temporality research by advancing and integrating the so far fragmented bodies of research on synchronizing and sustaining practices. It also contributes to research on agile practices. The study demonstrates that combining synchronizing and sustaining practices over time allows actors to achieve durable coordination across temporal boundaries.

1. Introduction

Research has widely acknowledged that shared temporal structures facilitate coordination between interdependent actors (Blagoev et al., 2024; Orlikowski & Yates, 2002). Temporal structures refer to the socially embedded ways of conceiving and ordering time that guide social life (Zerubavel, 1981). They manifest in surface-level temporal patterns that describe how actors structure their activities, including their pacing, timing, and sequencing. Moreover, they are constituted in deep-level temporal conceptions and orientations that more fundamentally describe how actors understand and relate to time (Ancona, Okhuysen, et al., 2001; Rowell et al., 2016). Actors share temporal structures when they use the same temporal patterns and/or hold the same temporal conceptions and orientations. Shared temporal patterns facilitate coordination (Ancona & Chong, 1996; Pérez-Nordtvedt et al., 2008) by providing temporal reference points that align actors' expectations of the timing, pacing, and sequencing of their interdependent activities. However, deep-level temporal structures should not be overlooked as well, as they strongly influence how actors shape and respond to temporal patterns (Ancona, Okhuysen, et al., 2001) and thus underpin the effectiveness of common coordination instruments like plans (Ballard & Seibold, 2000), deadlines (Yakura, 2002), or routines (Turner & Rindova, 2018).

Despite the coordinative benefits, the temporal structures of interdependent actors are often not shared but inherently different. Differences arise in various dimensions of temporal structures and across multiple levels of social and organizational life. For example, actors may have to collaborate across different time zones (Massey et al., 2003; Mell et al., 2021; Montoya-Weiss et al., 2001) or proceed in their interdependent tasks at different paces (Dille et al., 2023; Stjerne et al., 2019; van Berkel et al., 2016). They may consider different time horizons when thinking about future activities (Slawinski & Bansal, 2012; Stjerne et al., 2019) or differently structure their work with the regularity and predictability of the clock or based on the less predictable occurrence of specific events (Dougherty et al., 2013; Reinecke & Ansari, 2015). Temporal structures are rooted in diverse cultural backgrounds (Bluedorn & Ferris, 2004; M. J. Chen & Miller, 2011), in affiliation with different functional or occupational groups (Klein et al., 2007; Landy

et al., 1991), and in the heterogeneous cognitive traits of individuals (Bluedorn, 2002; Shipp et al., 2009). They manifest across different levels of collaboration and occur among individual actors in teams (Mell et al., 2021; Mohammed et al., 2017), between teams and organizational units (Dougherty et al., 2013; Oborn & Barrett, 2021; Otto et al., 2024; Reinecke & Ansari, 2015), and between organizations (Dille et al., 2023; Hilbolling et al., 2022; Slawinski & Bansal, 2012).

Actors who do not share the same temporal structures often encounter coordination challenges and conflicts that impede effective collaboration. When temporal structures diverge, actors hold different expectations regarding the timing, pacing, and sequencing of their interdependent activities (Bartel & Milliken, 2003; Jansen & Kristof-Brown, 2005; Mohammed et al., 2017; Mohammed & Nadkarni, 2011; Waller et al., 2001), which increases the need for explicit and active coordination among them. In addition, these divergent expectations can give rise to temporal conflicts (Dougherty et al., 2013; Reinecke & Ansari, 2015; Stjerne et al., 2019). If these conflicts are left unaddressed, they may undermine actors' willingness to collaborate (Jansen & Kristof-Brown, 2005; Mohammed et al., 2017) and sometimes result in the dissolution of collaborations (Dille et al., 2023; van Berkel et al., 2016). Given that divergent temporal structures can pose considerable challenges to effective coordination and collaboration, yet are pervasive in organizational and social reality, it is important to understand how actors can bridge these temporal differences to coordinate and collaborate successfully.

1.1. Research gaps

Temporality research reveals two approaches by which actors may coordinate across divergent temporal structures. Building on the notion that temporal structures can be actively shaped by human action (Orlikowski & Yates, 2002), actors can either intentionally alter their temporal structures to synchronize them with those of collaborating actors (synchronizing practices), or sustain their divergent temporal structures through targeted practices that navigate differing expectations and emerging conflicts (sustaining practices) (Bansal et al., 2022; Blagoev & Schreyögg, 2025; Hilbolling et al., 2022; Kaplan & Orlikowski, 2013; Otto et al., 2024; Pentland et al., 2025). This chapter introduces both synchronizing and sustaining practices. It outlines

the valuable contributions of prior research to understanding coordination across temporal structures. However, it also highlights the research gaps that leave our current understanding of how this coordination unfolds incomplete and fragmented.

Synchronizing practices dominate research on temporality. They intend to establish shared temporal patterns, thus taking advantage of the implicit coordination benefits they afford (McGrath, 1991; McGrath & Rotchford, 1983). Among synchronizing practices, entraining has emerged as the most prominent (Ancona & Chong, 1996; McGrath & Kelly, 1986; Pérez-Nordtvedt et al., 2008). Entraining involves actors aligning with the established temporal patterns of a dominant Zeitgeber, thereby relinquishing their own unique temporal patterns to achieve synchronization. Past research on entraining has focused on two distinct areas of interest. First, a range of quantitative empirical studies has demonstrated that entraining has a positive impact on organizational and team performance (Dibrell et al., 2015; Hopp & Greene, 2018; Khavul et al., 2010; Shi & Prescott, 2012). Second, qualitative studies have explored how time-setting actors motivate collaborating actors to entrain by providing information about temporal patterns (Gersick, 1989; Tukiainen & Granqvist, 2016; Yakura, 2002), by creating desirable goals and future imaginaries (Crawford et al., 2022; Hilkamo et al., 2021), or by directing actors' behavior through formal authority (Mohammed & Nadkarni, 2011; Otto et al., 2024).

By focusing on positive outcomes and unidirectional entraining processes, the current literature presents a relatively limited and one-dimensional perspective on entraining. Prior studies have largely overlooked two issues: the negative consequences of entraining and the decision-making processes of actors who adjust their temporal patterns to achieve entrainment. While the positive effects of entrainment have garnered considerable scholarly attention, its negative consequences have been largely neglected, fostering the misleading perception that entrainment is a "purely cost-free state of harmony" (Blagoev & Schreyögg, 2019, p. 1843). Recent research has begun to discuss limitations of entraining, especially its restricted applicability in the temporally complex settings of multi-party collaborations (Blagoev & Schreyögg, 2019, 2025; Hilbolling et al., 2022). Nevertheless, to date, the potential drawbacks for actors relinquishing their own temporal structures during entrainment remain unexplored. In the absence of such knowledge, we cannot fully understand the decision-making of actors who entrain to the temporal

patterns of a dominant Zeitgeber. Most studies have focused on entraining from the perspective of time-setting actors (Dille et al., 2023; Kunzl & Messner, 2023) and largely overlooked the agentic decision-making of actors who may abandon their temporal patterns. The agency of non-time-setting actors in entraining decisions is evidenced by studies showing how they resist entrainment (McGivern et al., 2018; Shipp & Richardson, 2021). Still, the drivers behind actors' decisions to engage in or resist entraining remain largely unexplored and warrant further attention (Shipp & Richardson, 2021; Zhang et al., 2023). Investigating these drivers is important as they lie at the intersection of synchronizing and sustaining practices, thereby contributing to an understanding of why actors choose either to synchronize or to preserve their divergent temporal patterns.

Synchronizing practices encompass not only entraining but also developing new temporal patterns. Like entraining, developing new temporal patterns serves to establish shared temporal patterns, enabling actors to coordinate more effectively (Orlikowski & Yates, 2002). Unlike entraining, however, this practice does not presume actors' adaptation to fixed and immutable temporal patterns. Instead, the newly developed temporal patterns constitute temporal innovations for all collaborating actors (Granqvist & Gustafsson, 2016; Lifshitz-Assaf et al., 2021). Research to date has focused on why actors develop new temporal patterns rather than entraining to established ones (Feldman et al., 2022), how new temporal patterns influence organizational outcomes (Lifshitz-Assaf et al., 2021; Otto et al., 2024), and how they constitute institutional change (Granqvist & Gustafsson, 2016).

A notable limitation of this body of research concerns its lack of attention to multilateral development processes. Prior studies consistently describe how new temporal patterns are developed and imposed unilaterally by a single actor (Granqvist & Gustafsson, 2016; Jarvenpaa & Välikangas, 2022; Lifshitz-Assaf et al., 2021). This emphasis on one time-setting actor renders new temporal patterns virtually indistinguishable from entrainment from the perspective of non-time-setting actors, since in both instances, they adjust to externally imposed temporal patterns (Omidvar et al., 2025). Because the role of non-time-setting actors in developing new temporal patterns has been largely overlooked, our understanding of this synchronizing practice, similar to entrainment, remains relatively one-dimensional. Given that actors do not always accept new temporal patterns (McGivern et al., 2018), greater attention to their agency in developing them is warranted.

Sustaining practices aim to preserve the coexistence of divergent temporal structures. Recently, they have received growing attention (Blagoev et al., 2024; Hilbolling et al., 2022; Kunisch et al., 2021). The rationale is twofold. First, synchronizing practices have limited applicability in temporally complex contexts (Blagoev & Schreyögg, 2025; Hilbolling et al., 2022). Second, preserving divergent temporal structures can itself provide benefits, such as reducing cognitive biases (Bakker & Knoben, 2015; Slawinski & Bansal, 2012) and supporting effective task performance under temporal uncertainty (Geiger et al., 2021). From the comparatively few studies, three key sustaining practices have emerged. Isolating involves intentionally separating actors with divergent temporal structures to avoid the occurrence of temporal tensions (Dille et al., 2023; Hilbolling et al., 2022; Oborn & Barrett, 2021). Brokering establishes bridging ties between actors with divergent temporal structures (Mell et al., 2021). These ties permit actors to remain separate across organizational or functional boundaries; thus, brokering practices can serve as complements to isolating practices. Finally, negotiating practices can foster temporal reflexivity, enabling actors to recognize and understand diverse temporal conceptions and orientations (Magrelli et al., 2022; Reinecke & Ansari, 2015). They serve as a precondition for creating ambitemporal patterns, which allow the preservation of diverse deep-level temporal structures.

Research on sustaining practices is still in its early stages, with a key objective of uncovering additional practices to explore how actors can maintain divergent temporal structures across different contexts. The applicability of isolating, brokering, and negotiating practices is limited to the specific conditions of the case studies in which they were identified. Isolating practices prove effective when actors face relatively weak interdependencies or little time pressure, allowing collaboration to be suspended for indefinite periods without endangering goal attainment (Dille et al., 2023; Hilbolling et al., 2022). Brokering practices have been studied thus far only in the context of time zone differences (Mell et al., 2021), raising questions about their applicability to other temporal differences. Negotiating practices are helpful when actors seek to preserve differences in their deep-level temporal structures (Reinecke & Ansari, 2015); however, they offer limited applicability in cases where sectors seek to preserve their divergent surface-level temporal patterns. Where interdependencies are high, time is pressing, or coordination must span multiple temporal dimensions, current research provides little guidance on which practices can be used to sustain their divergent temporal structures. Hence,

expanding the portfolio of sustaining practices is crucial for understanding how temporal structures can be maintained across the various facets of organizational reality.

Finally, current research emphasizes the significance of viewing temporal structuring as a dynamic process. Temporal structures were long assumed to be static and immutable (Ancona & Chong, 1996; McGrath & Kelly, 1986). While case studies have reinforced this perception by illustrating temporal structures in lock-in situations (Blagoev & Schreyögg, 2019; Perlow et al., 2002), research has begun to recognize that they can change dynamically, implying that temporal structuring practices are recurrent rather than singular interventions (Ancona & Waller, 2007). Likewise, research highlights the need for dynamic temporal structuring processing, since episodic temporal structuring only provides temporary solutions to temporal coordination problems, with temporal tensions resurfacing over time (Feldman et al., 2022; Hilbolling et al., 2022; Otto et al., 2024; Shipp & Richardson, 2021), while persistent and rigid temporal structures can entail substantial drawbacks (Blagoev & Schreyögg, 2019; Perlow et al., 2002).

However, the dynamics of temporal structuring have so far attracted little scholarly attention. Research to date has concentrated on individual temporal structuring episodes (Otto et al., 2024; Shipp & Richardson, 2021), leaving the question of whether, and if so, how, these episodes coalesce into ongoing and dynamic temporal structuring processes largely unexplored. Consequently, we lack understanding of how the dynamics of temporal structuring unfold and what drives change in temporal structures. Advancing our understanding of the dynamic processes that may underlie temporal structuring is crucial, given the current shortcomings in describing how temporal coordination can be sustained over time. It is thus important to broaden the current focus on single temporal structuring episodes to a continuous, potentially repeating process, and to theorize the mechanisms that underlie both the dynamics and the persistence of temporal structures (Blagoev et al., 2024).

Overall, the temporality literature offers only a partial and fragmented understanding of how actors with divergent temporal structures coordinate through synchronizing and sustaining practices. Synchronizing practices have predominantly been studied for their positive influence on coordination and organizational outcomes, with little consideration given to potential adverse effects. Moreover, the focus on synchronizing efforts of time-setting actors leads to a one-dimensional view of synchronizing practices that overlooks

the agency and decision-making of non-time-setting actors. This has left our understanding of synchronizing practices partial and incomplete. Moreover, research has identified sustaining practices that facilitate coordination across temporal differences, but only within particular contextual settings. Thus, there is a need to discover further practices that are effective in other contexts to elucidate how actors maintain divergent temporal structures in various scenarios of organizational collaboration. Focusing on single temporal structuring episodes yields a fragmented view of coordination across divergent temporal structures, where synchronizing and sustaining practices appear as dichotomous, either-or choices. Shifting this focus toward a dynamic temporal structuring process, in which synchronizing and sustaining practices are combined over time, is crucial for understanding how actors with divergent temporal structures achieve sustained coordination and for theorizing the ongoing and recurring changes in temporal structures.

1.2. Goal and structure of this dissertation

Seeking to advance understanding of coordination across divergent temporal structures against the background of the research gaps outlined in the previous chapter, this dissertation focuses on the following research question: *How do actors with divergent temporal structures coordinate their interdependent activities?*

Addressing this question necessitates an empirical context that satisfies two essential conditions. First, it must involve actors with clearly divergent temporal structures. Second, interdependencies between these actors should be sufficiently strong to necessitate collaboration. Collaborations between agile and non-agile units in banking and insurance organizations provide a revealing context, as agile units exhibit temporal structures that diverge substantially from those historically entrenched in these organizations. Thus, I analyzed eight case studies that focus on agile software development or restructuring teams in banking and insurance organizations, all of which exhibit strong interdependencies with various non-agile units that necessitate ongoing collaboration across their work processes. To identify and describe the practices and processes through which agile and non-agile units coordinate their interdependent activities, this dissertation draws on qualitative data. These include 91 interviews with representatives of agile and non-agile units, 256 participant observations of meetings

where collaboration between agile and non-agile units occurred, was planned, or was reflected upon, and 642 pages of archival documents that provide a deeper understanding of the respective organizational and team contexts.

The findings illustrate how agile and non-agile units coordinate throughout their work processes. They are organized according to three key temporal differences between agile and non-agile units that give rise to tensions: different planning cycles shaped by different near- and distant-future time orientations, different uses of deadlines rooted in divergent clock and event-time conceptions, and different rhythms of delivering outcomes influenced by varying levels of time urgency. Across these temporal differences, the findings show how agile and non-agile units respond to emerging temporal tensions, rooted in their divergent temporal structures, by enacting and reenacting various synchronizing and sustaining practices that ensure effective coordination and collaboration.

The findings make several contributions to temporality research. First, they advance current knowledge on synchronizing practices by shedding light on the previously underexplored drawbacks of synchronization. Specifically, they demonstrate how agile units perceive entraining as detrimental, as it compels them to abandon their unique temporal patterns, thereby diminishing the effectiveness of agile practices by restricting core agile capabilities. The findings also shed light on the underexplored agentic role of non-time-setting actors in both entraining and developing new temporal patterns. They show how agile units consider the potential loss of their agile capabilities alongside relational and resource-specific factors in their-decision-making, thereby deliberately choosing to engage in or resist synchronization. They also show how actors can mitigate the harmful effects of entraining by applying complementary practices. Moreover, the findings reveal the emergence of new temporal patterns as a result of multilateral negotiation processes, involving both agile and non-agile units. Thus, they show how the agency of non-time-setting actors contributes to temporal patterns that more closely align with their temporal orientations, thereby mitigating resistance. Second, the findings advance knowledge on sustaining practices by uncovering buffering and harmonizing practices. By coordinating ex-ante during episodes in which their temporal patterns are shared, buffering practices allow agile and non-agile units to sustain their divergent temporal patterns in later stages of their work processes, when they are beneficial.

Harmonizing practices intentionally create synchronous moments in which agile and non-agile units coordinate, even as their temporal patterns differ. Third, the findings broaden the focus on single temporal structuring episodes to ongoing temporal structuring processes. Specifically, they show how agile units engage in continuous temporal structuring and re-structuring, modifying their temporal structures to respond to recurring temporal tensions. By considering these dynamics of temporal structuring processes, the findings enrich our understanding of how actors with divergent temporal structures can coordinate effectively for sustained periods of time. Finally, the dynamic perspective on temporal structuring contributes to theorizing change in temporal structures by discussing the drivers behind the shifts between shared and divergent temporal structures.

Apart from its contributions to temporality research, this dissertation also advances research on agile practices. First, it introduces a temporal lens for describing agile practices that contributes to conceptually delineating agile from non-agile work practices. Second, the findings address the previously overlooked coexistence of agile and non-agile practices in organizations and reveal how temporal differences impair their interactions and collaboration. They indicate how addressing these tensions leads to agile method tailoring and the development of hybrid work practices. Third, temporal differences between agile and non-agile units explain why adopting agile practices does not necessarily translate into enhanced agile capabilities.

The dissertation is structured as follows. Chapters 2 and 3 present the definitional and theoretical foundations. *Chapter 2* introduces the time-as-structure lens (Blagoev et al., 2024), elaborates on surface- and deep-level temporal structures, and illustrates how different temporal structures can hinder collaboration between interdependent actors. It also conceptualizes temporal differences between agile and non-agile work practices. *Chapter 3* reviews the current state of research on coordination across divergent temporal structures. It elaborates on synchronizing and sustaining practices and highlights limitations and gaps in both domains. *Chapter 4* details the methodological approach. It explains the choice of research design, provides a rationale for the empirical context, outlines the case selection process, and introduces the eight selected case studies. It also describes the data collection instruments and techniques applied for data analysis. *Chapter 5* presents the findings. They are organized around three key differences in the temporal structures of agile and non-agile units (different planning cycles, different clock

and event-time conceptions, different release rhythms), describing the tensions that arise from these differences and the (combinations of) synchronizing and sustaining practices through which they are resolved. These findings are discussed in *Chapter 6*, highlighting the theoretical implications for research on temporality and agile practices, as well as the dissertation's main limitations and potential directions for future research. *Chapter 7* concludes with a concise summary, reflecting on the dissertation's core contributions.

2. Introducing the time-as-structure lens: Implications of shared and divergent temporal structures

The temporal lens has elevated time from a peripheral concept to an explicit and central dimension of management research (Ancona, Goodman, et al., 2001; Blagoev et al., 2024; Bluedorn & Denhardt, 1988; J. M. George & Jones, 2000; Huy, 2001). Blagoev and colleagues (2024, p. 2155) define the temporal lens as a "meta-theoretical perspective that (i) assumes time as a central dimension of management, (ii) highlights specific domains in which time matters for management, and (iii) encompasses time-based concepts and approaches that explain how time matters for management." Across various management disciplines, including strategy (Huy, 2001; Kunisch et al., 2017; Mosakowski & Earley, 2000; Schultz, 2022), human resources (Aguinis & Bakker, 2021), entrepreneurship (Wood et al., 2021), or organizational studies (Shipp & Cole, 2015; Sonnentag, 2012), the temporal lens has emerged as "a powerful way to view organizational phenomena" (Ancona, Goodman, et al., 2001, p. 660) and has contributed to a deeper understanding of diverse organizational phenomena and behaviors, as well as to the advancement of theory.

The temporal lens comprises multiple facets (Ancona, Okhuysen, et al., 2001; Blagoev et al., 2024; Shipp & Fried, 2014). Time can be conceptualized as a resource, a process, or a structure (Blagoev et al., 2024). Viewing time as a resource suggests that time is a valuable asset that managers must actively leverage to generate economic benefits (Blagoev et al., 2024; Bluedorn & Denhardt, 1988). Accordingly, the speed, timing, and time horizons of activities and decisions are managed to enhance organizational performance. For example, strategy research has explored how speed can function as both a source of (Hawk et al., 2013) and a threat to competitive advantage (Hawk & Pacheco-de-Almeida, 2018). It has been demonstrated that both early action (Barbero et al., 2017; Elfenbein & Knott, 2015; Merida & Rocha, 2021) and delayed action (Boyd & Bresser, 2008; Raaijmakers et al., 2015) shape organizational success. Finally, studies that conceptualize time as a resource have examined the benefits and drawbacks of long-term

versus short-term time horizons for economic outcomes (DesJardine & Bansal, 2019; Flammer & Bansal, 2017; Nadkarni et al., 2016; Opper & Burt, 2021; Reilly et al., 2016).

Additionally, time can be understood from a process perspective. A process view considers time as "an indivisible flow of interconnected events" (Blagoev et al., 2024, p. 2169). Because it assumes that time is indivisible, the time-as-process perspective treats conventional temporal distinctions (e.g., past and future, before and after) as artificial and malleable. Accordingly, actors are seen as actively performing time by interrelating past and future events and negotiating and enacting them in the ongoing present (Hernes, 2022). To date, research has advanced our understanding of linking past, present, and future to sustain continuity, which is particularly helpful during times of crisis (Bednar et al., 2020; Cunliffe et al., 2004; Danner-Schröder, 2020; Feldman et al., 2022; Hernes & Schultz, 2020), and how disconnecting them can create the discontinuity necessary to enable change (Berends et al., 2021; Geiger et al., 2021; Ybema, 2010). In addition, studies on how actors imagine the future in various ways have shed light on the consequences of divergent future imaginaries (Augustine et al., 2019; Rauch, 2025; Wenzel et al., 2020, 2025). Finally, research grounded in the time-as-process lens has examined the conditions under which individuals and groups evoke the past in the present (Hampel & Dalpiaz, 2023; Ravasi et al., 2019; Schultz & Hernes, 2013).

The primary temporal perspective adopted in this thesis is the time-as-structure lens. Temporal structures are conceptualized as "sociotemporal order, which regulates the structure and dynamics of social life" (Zerubavel, 1981, p. 2), and through which actors experience and make sense of time (Orlikowski & Yates, 2002). Prior research has identified several dimensions along which temporal structures manifest (Ancona, Okhuysen, et al., 2001; Bansal et al., 2022; Blagoev et al., 2024; Orlikowski & Yates, 2002). At the surface level, temporal structures are evident in observable temporal patterns (Ancona, Okhuysen, et al., 2001; Omidvar et al., 2025; Rowell et al., 2016). At a deeper level, they are embedded in temporal orientations and, more fundamentally, in underlying temporal conceptions (Ancona, Okhuysen, et al., 2001; Rowell et al., 2016). These deeper manifestations of temporal structures are typically invisible, intangible, and largely informal (Blagoev et al., 2024). They shape how actors construct and respond to the temporal patterns that constitute surface-level temporal structures (Janicik & Bartel, 2003).

In contrast to the time-as-resource lens, the time-as-structure lens posits that time is socially constructed and that temporal structures are negotiated through human interaction (Durkheim, 1965; Oborn & Barrett, 2021; Perlow, 1999; Sorokin & Merton, 1937). Through frequent and routine enactments, temporal structures may acquire high legitimacy and come to be perceived as natural, taken for granted, and unchangeable (Orlikowski & Yates, 2002). Nevertheless, they remain potentially malleable (Reinecke & Ansari, 2015) and can be actively shaped by actors and their interactions. In this vein, Orlikowski and Yates (2002, p. 684) understand temporal structures "as both shaping and being shaped by ongoing human interaction; and thus, as neither independent of human action (because shaped in action) nor fully determined by human action (because shaping that action)." Consequently, scholarly attention has increasingly focused on how temporal structures evolve and how they are actively shaped by actors (Cuganesan, 2022; Gersick, 1989; Granqvist & Gustafsson, 2016; Kremser & Blagoev, 2021; Staudenmayer et al., 2002). At the same time, research has examined why, in some instances, temporal structures remain persistent, even when they have potentially detrimental consequences for the actors embedded within them (Blagoev & Schreyögg, 2019; Perlow, 1999).

Actors can influence a range of broader organizational outcomes by deliberately creating and enacting temporal structures (Crossan et al., 2005; Fayard, 2024; Lifshitz-Assaf et al., 2021; Reinecke & Ansari, 2015). For instance, temporal structures based on cyclical temporal patterns are more conducive to addressing long-term challenges such as climate change than those based on linear temporal patterns (Slawinski & Bansal, 2012). Furthermore, combining event-time-based and clock-time-based temporal structures can facilitate innovation outcomes within organizations (Fayard, 2024).

Most important for this study is how temporal structures affect coordination among actors (Bechky & Okhuysen, 2011; Blagoev et al., 2024; Oborn & Barrett, 2021; Orlikowski & Yates, 2002). Coordination refers to how actors work collectively to integrate interdependent and specialized activities to accomplish shared goals, tasks, or pieces of work (Faraj & Xiao, 2006; Okhuysen & Bechky, 2009). It is facilitated through various mechanisms that enable actors to anticipate one another's actions (Okhuysen & Bechky, 2009; van de Ven et al., 1976). Research has identified a variety of coordination mechanisms. For example, Thompson (1967) distinguishes between standardization (rules, procedures, and policies that promote consistency across interdependent actors), planning (task scheduling and timeline setting), and mutual adjustment (ongoing formal

and informal communication as well as real-time negotiation). Building on this foundation, Okhuysen and Bechky (2009) provide an integrative review that expands Thompson's (1967) set of coordination mechanisms to include objects, representations, roles, and routines. Collectively, coordination mechanisms serve as shared reference points that convey information and establish expectations, thereby fostering accountability, predictability, and a shared understanding among actors (Okhuysen & Bechky, 2009).

The effectiveness of various coordination mechanisms depends on the extent to which actors share the same temporal structures (Geiger et al., 2021). For example, the success of planning relies on shared temporal reference frames (Ballard & Seibold, 2003). This is illustrated by early coordination research on American railroads in the nineteenth century (Chandler, 1962; Stover, 1970). The use of plans, in the form of printed timetables, enabled railroad corporations to coordinate on a large scale (Stover, 1970). However, timetables only functioned effectively due to the introduction of Standard Railroad Time, which replaced more than 500 local times and prevented timetables from being interpreted differently by various actors. Similarly, routines, defined as sequential patterns of task accomplishment (Pentland & Feldman, 2007), depend on shared interpretations of the order, timing, or pace at which activities are performed. For example, Turner and Rindova (2018) demonstrated how shared timing norms influenced the performance of garbage collection routines. Timing norms shared by both the garbage collection organization and the city's residents ensured that garbage bins were in the right place at the right time, thereby facilitating the garbage collection routines. Thus, by enabling coordination mechanisms, shared temporal structures may facilitate coordination among interdependent actors (Bansal et al., 2022; Orlikowski & Yates, 2002; Zerubavel, 1981).

However, actors who need to coordinate do not necessarily share the same temporal structures. Several studies have documented instances in which the temporal structures of interdependent actors diverge (Cuganesan, 2022; Dille et al., 2023; Hilbolling et al., 2022; Kaplan & Orlikowski, 2013; Oborn & Barrett, 2021; Reinecke & Ansari, 2015; Stjerne et al., 2019; Waller et al., 2001). Differences in temporal structures can generate contradictory or conflicting expectations regarding when and how activities and events should be performed. Such discrepancies may give rise to various tensions among actors, ultimately hindering effective coordination (Orlikowski & Yates, 2002). For example,

Waller and colleagues (2001) argue that disparate temporal orientations among individuals within teams influence how they perceive and respond to deadlines. When perceptions of and behaviors toward deadlines differ, teams often struggle to meet them, as they must negotiate their understanding and establish a unified approach. Oborn and Barrett (2021) demonstrate how differing temporal orientations among medical professionals, specifically the extent to which they focus on past, present, and future dimensions of time, impede collaboration in hospitals. For example, they illustrate how surgeons declined to consult oncologists, aiming to circumvent potential tensions between their present focus and the oncologists' future focus. Although this behavior aligned with the surgeons' fast-paced temporal patterns, it ultimately undermined the quality of patient care.

These examples illustrate that temporal structures are complex and multidimensional. They have contrasted clock time and event time (Cuganesan, 2022; Reinecke & Ansari, 2015), cyclical and linear time (Slawinski & Bansal, 2012), and different temporal orientations (Oborn & Barrett, 2021; Waller et al., 2001). While each of the aforementioned studies addresses temporal structures, they focus exclusively on specific dimensions. Consequently, temporal structures do not give rise to a single type of temporal difference but can generate multiple, distinct temporal differences. This underscores the importance of clearly delineating the dimensions of temporal structures in which such differences may emerge. The following chapter outlines the core dimensions of temporal structures, describes the temporal differences within these dimensions, and, drawing on extant research, illustrates how tensions may arise from discrepant temporal structures.

2.1. The multidimensionality of temporal structures and their differences

To comprehend the multidimensional nature of differences in temporal structures across and within organizations, teams, and individuals, it is paramount to categorize and clearly delineate the various dimensions of temporal structures. This dissertation draws on Ancona, Okhuysen, and colleagues' (2001) temporal framework, which comprises three

overarching dimensions: (1) temporal conceptions, (2) temporal orientations, and (3) temporal patterns. This framework provides a comprehensive lens for understanding temporal structures from multiple perspectives and has been widely adopted in temporality research (Ballard, 2009; Hernes & Schultz, 2020; Kunisch et al., 2017; Rowell et al., 2016). The first dimension, temporal conceptions, concerns the fundamental properties actors attribute to time. It addresses questions such as what time is and explores the social and cultural constructions of time developed by individuals, organizations, and societies. Chapter 2.1.1 elaborates on the two most prominent temporal conceptions in management research: clock time and event time, each reflecting distinct beliefs and assumptions about the nature of time. The second dimension, temporal orientations, focuses on how actors relate to time. Temporal orientations reflect deep-level temporal structures that influence how actors value and attend to time (Rowell et al., 2016). While research has identified a wide range of temporal orientations, Chapter 2.1.2 focuses on the two particularly salient in the empirical context of this dissertation: temporal depth and time urgency. The third and final dimension, temporal patterns, pertains to how actors map activities and events onto time, that is, the visible temporal structuring of activities and events (Rowell et al., 2016). Chapter 2.1.3 delineates and defines the various temporal mapping choices through which actors create temporal patterns. Importantly, Ancona, Okhuysen, and colleagues (2001) use the term 'actor' to refer to multiple levels of analysis, ranging from individuals and teams to organizations and broader societies.

It is important to recognize that the three dimensions of temporal structures should not be regarded as discrete or isolated, but as interconnected and mutually influential (Ancona, Okhuysen, et al., 2001). While Rowell and colleagues (2016) contend that deep-level temporal conceptions and temporal orientations exert a unidirectional influence on surface-level temporal patterns, Ancona, Okhuysen, and colleagues (2001) propose that these dimensions are bidirectionally interrelated. Temporal conceptions, actors' fundamental beliefs about time, shape how they construct temporal patterns (Rowell et al., 2016). However, if actors are exposed to specific temporal patterns for a prolonged period of time, their underlying temporal conceptions may be reshaped. Similarly, actors create different temporal patterns if they hold different temporal orientations (Waller et al., 2001; Zhang et al., 2023). For example, actors with high time urgency tend to pace activities at a faster rate than actors with low time urgency (Waller et al., 2001). However, actors may also, at least temporarily, adopt a temporal orientation suited to navigating a

specific temporal pattern (Gibson et al., 2007). Finally, temporal orientations may also be influenced by temporal conceptions. For instance, research shows that actors who hold a clock-time conception, viewing time as a finite resource, tend to exhibit higher levels of time urgency than those who regard time as cyclical and repetitive (Huy, 2001; Mosakowski & Earley, 2000). Figure 1 illustrates the interrelations among the three temporal dimensions, along with the specific temporal constructs that I will elaborate on within each dimension.

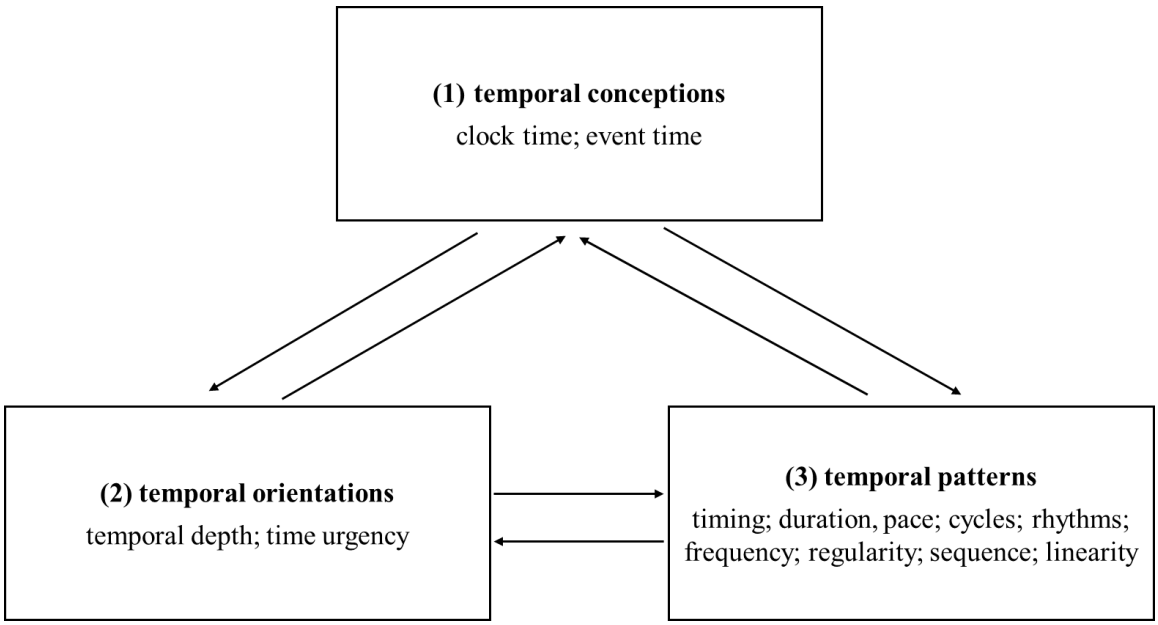

Figure 1: Dimensions of temporal structures and their interrelations

2.1.1. Temporal conceptions

Conceptions of time refer to the fundamental assumptions that actors hold about the nature of time. These assumptions are frequently expressed as dichotomies (Blagoev et al., 2024; Holt & Johnsen, 2019; Rowell et al., 2016), such as clock time versus event time (Bluedorn & Denhardt, 1988; Cuganesan, 2022; Jaques, 1982; Orlikowski & Yates, 2002; Reinecke & Ansari, 2015; Slawinski & Bansal, 2012), objective versus subjective time (Clark, 1990; Durkheim, 1965; Mosakowski & Earley, 2000; Orlikowski & Yates, 2002; Shipp & Fried, 2014; Shipp & Jansen, 2021), quantitative versus qualitative time (Huy, 2001; Sorokin & Merton, 1937; Sztompka, 1993), real versus epiphenomenal time (Mosakowski & Earley, 2000; Tuma & Hannan, 1984), linear versus cyclical time

(Ancona, Okhuysen, et al., 2001; M. J. Chen & Miller, 2011; Crossan et al., 2005; Gell, 1992; Mosakowski & Earley, 2000; Slawinski & Bansal, 2012), or natural versus social time (Adams, 1995; Hassard, 1991; B. S. Lawrence, 1984).

In organizational research, temporal conceptions are commonly framed in terms of clock time and event time (Bluedorn & Denhardt, 1988; Cuganesan, 2022; Fayard, 2024; Jaques, 1982; Orlikowski & Yates, 2002; Reinecke & Ansari, 2015). These two concepts represent superordinate temporal conceptions under which many of the previously mentioned dichotomies can be subsumed (Reinecke & Ansari, 2015). For example, Huy (2001) explicitly equates quantitative time with clock time and qualitative time with event time. Similarly, Orlikowski and Yates (2002, p. 685) state that "the objective-subjective dichotomy is often presented in terms of the contrast between clock time and event time."

Clock time refers to a conception of time as invariant, universal, quantitative, measurable, linear, and finite (Bluedorn & Denhardt, 1988; Huy, 2001; Jaques, 1982; Orlikowski & Yates, 2002; Reinecke & Ansari, 2015). Its invariant nature aligns with a realist view on time. For example, Tuma and Hannan (1984, p. 231) assert that time "is a fundamental category and exists independently of events, space, and motion." Similarly, Clark (1990, p. 142) emphasizes invariance, stating that time exists "independent of man." Accordingly, a clock-time ontology assumes time to be objective and universal (Huy, 2001; Orlikowski & Yates, 2002; Sztompka, 1993), grounded in external metrics that are uniformly interpreted and applied across contexts (Mosakowski & Earley, 2000). External metrics include clocks and calendars, such as the Gregorian calendar, which has gained almost universal recognition and is used globally for the temporal structuring of events and activities (Zerubavel, 1981). They enable standardization and quantification of time by segmenting it into fixed, measurable units, such as years, months, weeks, days, hours, minutes, and seconds (Ancona, Okhuysen, et al., 2001; Lauer, 1981; McGrath & Rotchford, 1983). These units are consistent in duration and interpreted uniformly by actors (Bluedorn & Denhardt, 1988). As a result, the passage of time is perceived as steady and continuous, irrespective of individual subjective experiences. As Shipp and Fried (2014, p. 2) note: "The clock keeps ticking regardless of whether we perceive it." Moreover, clock time presupposes a linear understanding of time. This linearity is rooted in Judeo-Christian traditions, in which time is seen as "beginning with creation and ending with the apocalypse" (Reinecke & Ansari, 2015, p. 620). Time is thus understood

as forward-moving, with minimal or no recurrence of events (Gell, 1992; Mosakowski & Earley, 2000). It unfolds as a chronological sequence in which the present follows the past and precedes the future (Jaques, 1982; Shipp & Fried, 2014). Actors with a linear clock-time conception tend to perceive each day as relatively new and largely unconnected to past or future events. Finally, the notion of time as finite reflects the belief that time is a scarce and non-renewable resource (Huy, 2001; Mosakowski & Earley, 2000). Because time is viewed as linear and non-repetitive, once it has passed, it cannot be recovered. This understanding is especially salient in strategic management research (Kunisch et al., 2017; Mosakowski & Earley, 2000). For example, Tyre and Orlikowski (1994) highlight the finite nature of time by showing that time-limited windows of opportunity constrain technological adaptation. Once such a window closes, the opportunity may be lost.

Event time is typically presented as the antithesis of clock time (H. Lee & Liebenau, 1999). It is sometimes also referred to as process time (Reinecke & Ansari, 2015) or psychological time (Shipp & Fried, 2014). This conception of time emphasizes subjectivity, contextuality, qualitative assessment, and cyclicality (Bluedorn & Denhardt, 1988; Granqvist & Gustafsson, 2016; Orlikowski & Yates, 2002). In contrast to clock time, event time is not considered universal. Instead, it is always situated within specific contexts (Bluedorn & Denhardt, 1988). Unlike clock time, event time cannot be measured quantitatively (Jaques, 1982). It is not divided into standardized units but is instead experienced and structured in relation to events or opportunities. These events may arise irregularly and unpredictably (Lauer, 1981), resulting in indeterminate and non-linear temporal trajectories (Huy, 2001). As a result, event time is perceived as flowing unevenly and is often regarded as more dynamic and adaptive than clock time (Bluedorn & Denhardt, 1988). Finally, the event-time conception assumes that time is cyclical. From this perspective, time unfolds in recurring cycles and repetitive patterns, with the past, present, and future being deeply interconnected (Gell, 1992; Mosakowski & Earley, 2000; Slawinski & Bansal, 2012). Thus, actors who hold a cyclical event-time conception are likely to experience the recurrence of past experiences or events in the present or the future (Bartunek & Necochea, 2000).

Despite their frequent portrayal as dichotomous, clock time and event time are not mutually exclusive (Crossan et al., 2005; Cuganesan, 2022; Cunha, 2004; Orlikowski &

Yates, 2002; Reinecke & Ansari, 2015). Societies, organizations, groups, and individuals can simultaneously hold both temporal conceptions (Bluedorn & Denhardt, 1988). Orlikowski and Yates (2002), for instance, illustrate how the same individuals may draw on clock and event time in different contexts. People typically adopt a clock-time conception, structured around the fiscal calendar, when filing tax returns to meet strict deadlines. In contrast, the decision to plow roads during a snowstorm reflects an event-time conception, where unfolding events prompt action. Crossan and colleagues (2005) and Cunha (2004) show that the dichotomy between clock time and event time can be bridged through improvisation. Similarly, Antoni and colleagues (2023) show that childcare workers interweave both temporal conceptions. While childcare work is inherently event-driven, workers in their study relied on clock-time-based schedules to trade working hours, thereby circumventing fixed quotas per child. This hybrid approach allowed them to remain responsive to individual children's needs, preserving an event-time conception within clock-time-based temporal patterns. Fayard (2024) likewise shows that clock and event time can be integrated to support idea generation in social innovation hackathons. For example, increasing the number of feedback interactions (event-time conception) during a fixed hackathon period (clock-time conception) helped facilitate creative outcomes. Similarly, Garud and colleagues (2011), in their study of innovation at 3M, highlight how the company manages the temporal complexity of innovation by combining both temporal conceptions. Employees were allotted 15% of their work time for exploration (clock-time conception), but they did not use this time uniformly each week. Instead, they reserved it for moments when promising opportunities arose (event-time conception).

Although research has shown that actors can draw on both clock and event-time conceptions, it has also demonstrated that diverging temporal conceptions often create tensions within and between collectives because one temporal conception typically dominates (Cuganesan, 2022; Reinecke & Ansari, 2015). Since the onset of industrialization, clock time has become the dominant temporal conception in the Western world (Bluedorn, 2002; Bluedorn & Denhardt, 1988; Jaques, 1982; McGrath & Kelly, 1986). Actors routinely rely on clock time, frequently marginalizing or disregarding event time (Bluedorn & Jaussi, 2008; Cuganesan, 2022; McGivern et al., 2018). The dominance of clock time is so pervasive that some scholars have referred to it as clock-time hegemony (Reinecke & Ansari, 2015). In contrast, event time is more

prominent in traditional Asian cultures (M. J. Chen & Miller, 2011). For example, the Chinese calendar organizes time in recurring 60-year cycles, a logic that differs markedly from the linear progression of the Gregorian calendar. These culturally embedded differences in temporal conceptions can produce tensions in globalized organizations, causing individuals and groups with divergent temporal conceptions to struggle to coordinate and collaborate effectively.

Differences between clock-time and event-time conceptions are not limited to the societal level; they also emerge between and within organizations. Slawinski and Bansal (2012) examined how Canadian oil companies responded to climate change and found that some firms operated based on clock-time conceptions, while others exhibited event-time conceptions. However, because their study did not focus on interorganizational collaboration, it only demonstrates that temporal differences can exist between organizations in the same industry, without addressing the potential tensions these differences might cause in collaborative settings. Gersick (1994) investigated the collaboration between a venture capitalist and a company in the medical products industry. She observed temporal differences: the venture capitalist drew on an event-time conception, while the company's chief executive officer (CEO) was guided by a clock-time conception. Although Gersick (1994) confirms that differing temporal conceptions can coexist in interorganizational relationships, she does not examine the potential tensions that may arise from these differences. Nevertheless, she emphasizes that both temporal conceptions have distinct advantages and drawbacks, suggesting that each may be beneficial in particular contexts and situations.

Tensions arising from conflicts between clock time and event time are well documented in research on collaboration between different occupational groups within organizations. For example, Dougherty and colleagues (2013) examined interactions between managers and scientists in a pharmaceutical company. They found that managers are typically oriented toward clock time, while scientists tend to follow an event-time logic. These different temporal conceptions reflect the distinct tasks and pressures each group faces. Managers, on the one hand, must respond to external demands, such as meeting shareholder expectations and paying annual dividends. Because the stakeholders they engage with usually follow a clock-time conception, managers align accordingly. The work of scientists, on the other hand, revolves around key learning events that signify

progress in innovation processes (Garud et al., 2011). Consequently, scientists assess progress through the occurrence of significant events, whereas managers rely on clock-time-based milestones. Dougherty and colleagues (2013) show that these differing temporal conceptions can create substantial tension in coordinating the two occupational groups.

Similarly, Reinecke and Ansari (2015) illustrate how conflicting conceptions of clock time and event time generate tensions between different groups within a single organization. They examine a fairtrade organization dedicated to supporting the development of marginalized agricultural producers in the Southern hemisphere. This objective is fundamentally grounded in event time. Producers' developmental progress is evaluated based on the achievement of specific events, which cannot be precisely measured using clock-time-based milestones due to considerable variability in development levels across producers. However, the organization was compelled to respond to the demands of Northern market customers who required certifications to verify the developmental progress of Southern producers. These certification procedures reflect the temporal expectations of Northern customers, which are inherently clock-time-based and assess progress in standardized temporal units. The incompatibility between event-time-oriented developmental activities and clock-time-driven certification processes gave rise to two temporal tensions. First, it led to persistent disagreements and conflicts between the teams responsible for operational development and the team overseeing certification. Second, the divergence between clock time and event time posed a significant risk to the success of the developmental efforts, as the two processes were deeply interdependent. Without clock-time-based certifications, the organization risked losing credibility, and Northern customers might cease purchasing its fairtrade products, potentially bringing developmental efforts to a halt. Conversely, reliance on clock-time-based certifications impeded accurate assessment of producers' developmental progress. As a result, the most vulnerable and targeted producers were often excluded for failing to meet standardized milestones, contradicting the core principles of the organization's missions. Thus, Reinecke and Ansari (2015) not only elucidate the incompatibility between clock-time and event-time conceptions but also highlight the detrimental consequences of such temporal tensions.

This chapter has introduced clock time and event time as two distinct temporal conceptions that can give rise to temporal differences. Conflicts between these conceptions often lead to temporal tensions that hinder collaboration and coordination both between and within collectives. However, even when actors share the same temporal conception, temporal differences may still emerge. For example, actors who share an event-time conception may structure their activities according to divergent event trajectories. Similarly, those who share clock-time-based temporal structures may differ in their paces, cycles, or time horizons. As Reinecke and Ansari (2015, p. 402) note, although the "clock depicts objective clock time, we rarely experience time uniformly." To fully capture the complexity of temporal differences, it is essential to explore how actors relate to time in varied ways and how they differ in mapping activities onto time (Ancona, Okhuysen, et al., 2001). These additional dimensions of temporality, namely temporal orientations and temporal patterns, will be introduced in the following chapters.

2.1.2. Temporal orientations

Temporal orientations represent the second dimension of Ancona, Okhuysen, and colleagues' (2001) temporal framework. This dimension emphasizes how actors relate to time, thereby shifting the emphasis toward the actors themselves (Ancona, Okhuysen, et al., 2001). Temporal orientations are defined as the "shared manners of valuing and attending to time that are enacted in practice" (Rowell et al., 2016, p. 12). Although they are not as broadly shared as temporal conceptions, temporal orientations nonetheless constitute deep-level temporal structures (Rowell et al., 2016). In this sense, they reflect the subjective nature of time, recognizing that each actor may value, experience, and engage with time in ways that differ from others (Shipp & Jansen, 2021).

Prior research has identified various temporal orientations. The most prominent include temporal depth, time urgency, temporal focus, polychronicity, and pacing styles (Tang et al., 2020). Table 1 provides an overview of these temporal orientations, including definitions and illustrative references. As Chapter 2.2 will demonstrate, temporal depth and time urgency are particularly relevant for the empirical context of this dissertation, as they differ markedly between agile and non-agile organizational units. Accordingly, this chapter will focus on defining and explaining temporal depth and time urgency,

examining how these most salient temporal orientations for the empirical context differ across collectives, and discussing both the positive and negative implications of these temporal differences.

Table 1: Summary of temporal orientations

Temporal orientations:	Definition:	Exemplary references:
temporal depth	the "distance into the past and future that actors typically consider when contemplating over events that have happened, may have happened, or may happen".	Bluedorn (2002, p. 114); Bluedorn & Standifer (2006)
time urgency	the "consistent concern with the passage of time, with those higher in time urgency feeling generally hurried across situations"	Shipp & Cole (2015, p. 243); Landy et al. (1991); Conte et al. (1995)
temporal focus	the importance that actors devote to the past, present, and future	Bluedorn & Standifer (2006); Bluedorn (2002); Shipp & Cole (2015); Shipp et al. (2009)
polychronicity	"the extent to which people (1) prefer to be engaged in two or more tasks or events simultaneously […] and (2) believe their preference is the best way to do things."	Bluedorn (2002, p. 51); Bluedorn & Martin (2008); Hall (1983);
pacing style	the "behavioral tendencies regarding the distribution of effort over time in working toward deadlines"	Gevers et al. (2015, p. 502)

Temporal depth:

This chapter begins by introducing the construct of temporal depth, which has received considerable attention in the literature. Temporal depth refers to the distance "into the past and future that individuals and collectives typically consider when contemplating events that have happened, may have happened, or may happen" (Bluedorn, 2002, p. 114; Bluedorn & Standifer, 2006). Scholars typically distinguish between future temporal depth (how actors think ahead into the future) and past temporal depth (how far actors think back into the past) (Bluedorn & Standifer, 2006; Nadkarni et al., 2016). Together, future and past temporal depth constitute an actor's total temporal depth, defined as the span of time from the most distant past to the most distant future that is typically taken into account (Bluedorn & Standifer, 2006).

Temporal depth is sometimes described using the term time horizon. In most research, time horizon refers specifically to the future-oriented part of temporal depth (Blagoev et al., 2024; Bluedorn, 2002; J. Chen et al., 2021; Das, 1987; Shipp et al., 2009; Strathman et al., 1994). In this sense, Bluedorn and Martin (2008) argue that temporal depth is a broader construct than time horizon, as it encompasses both future- and past-oriented considerations. The common association of time horizons with the future may stem from the disproportionate scholarly attention given to future-oriented inquiry. For example, strategic management literature has heavily emphasized future time horizons in relation to strategic planning and investment decisions (Das, 1987; Laverty, 1996; Marginson & McAulay, 2008; Reilly et al., 2016; Souder & Bromiley, 2012), whereas the past has only recently garnered sustained academic interest (Ravasi et al., 2019; Sasaki et al., 2020). Accordingly, the view that time horizons pertain solely to the future is not universally accepted. For instance, DesJardine and Bansal (2019) conceptualize time horizons as comprising both prospective and retrospective dimensions. While the empirical findings of this dissertation primarily reveal prospective time horizons, I adopt DesJardine and Bansal's (2019) broader definition and use the term time horizon analogously with temporal depth to denote the span of time over which actors think forward into the future or backward into the past.

Temporal depth can manifest in different forms. Das (1987) distinguishes between near- and distant-future time orientations, which denote the extent to which actors consider the future. Although Das (1987) does not explicitly address the past, one may analogously speak of near- and distant-past time orientations. A distant-future (or -past) time orientation reflects actors' tendency to focus on a temporally remote future (or past) (Das, 1987). Other scholars describe this tendency as a long-term orientation (Lin et al., 2019; Lumpkin & Brigham, 2011; Reilly et al., 2016). Actors with a distant-future time orientation can identify patterns that remain obscured in the short term (Bluedorn, 2002; Bluedorn & Martin, 2008; Strathman et al., 1994), thereby enabling them to anticipate and capitalize on long-term opportunities (Das, 1987; T. Wang & Bansal, 2012). Consequently, Eisenhardt and colleagues (1997) characterize executives exhibiting a distant-future time orientation as visionaries. Similarly, a distant-past time orientation allows actors to address present challenges efficiently by implementing strategies that have proven successful historically (Nadkarni et al., 2016). However, an orientation toward the distant future or past carries the risk of overcommitment to long-term visions

or entrenched strategies, which may diminish flexibility and adaptability in response to short-term developments (Levinthal & March, 1993; Nadkarni et al., 2016). In contrast, a near-future (or -past) time orientation reflects a preference to consider more immediate time horizons (Das, 1987). A near-future (or past) time orientation facilitates flexibility and rapid adaptation to change (Crossan et al., 2005; Nadkarni et al., 2016; Shipp et al., 2009). However, actors with a near-future (or -past) time orientation may fail to recognize longer-term patterns and opportunities, thereby potentially jeopardizing sustained success. Consequently, this orientation is sometimes negatively characterized as short-termism (M. J. Chen & Miller, 2011; Laverty, 1996; Marginson & McAulay, 2008) or temporal myopia (Levinthal & March, 1993; Opper & Burt, 2021; Souder & Bromiley, 2012).

Temporal depth and, by analogy, time horizons, determine the extent to which actors consider past temporal patterns and plan for future ones. An actor with a distant-future time orientation is likely to schedule activities and events over more extended periods than an actor with a near-future time orientation. Similarly, an actor with a distant-past time orientation exhibits heightened awareness of the rhythmic patterns of past activities and events, which may increase their propensity to perpetuate these rhythms compared to an actor with a near past orientation. Notably, actors can determine the length of their time horizons, and thus their temporal depth, both in clock time and event time (Bluedorn & Jaussi, 2008), depending on whether they measure time in standard clock units (e.g., years, days, or hours) or in terms of the occurrence of events.

In collectives, the temporal depth of individual members may vary. The literature suggests that temporal depth is a relatively stable cognitive characteristic (Bluedorn & Denhardt, 1988; Strathman et al., 1994), or even a temporal personality trait (Ancona, Okhuysen, et al., 2001; Das & Teng, 2001). Accordingly, individuals may have a natural tendency to adopt either a near- or distant-future (or -past) time orientation, which shapes their capacity to reflect on past experiences or anticipate future outcomes. Das (1987) argues that these intrinsic preferences are so pervasive that an individual with a near-future time orientation, when attempting to plan for the distant future, may simply extrapolate their short-term perspective.

In addition, perceptions of what constitutes the near or distant future (or past) are neither uniform nor objective. Bluedorn and Ferris (2004) highlight a significant difference

between American and Japanese time horizons used to define the distant future. While Americans tend to view the distant future as spanning hundreds of years, the Japanese consider it to encompass thousands. Beyond cultural differences, individuals vary considerably in their understanding of the long term. For example, in a case study of an interorganizational project in Norway, Stjerne and colleagues (2019) found that project partners' interpretations of the long term ranged from a few months to several decades. Similarly, Gutierrez and colleagues (2024) observed that entrepreneurs tend to have a nearer future time orientation compared to salaried employees. P. R. Lawrence and Lorsch (1967) identified functional-level variations in organizational time horizons. Employees in marketing, production, and research and development units exhibited notable differences in their temporal depth. However, this study does not clarify whether individuals' preferences are shaped by their organizational function or whether certain functions attract individuals with particular temporal orientations. Thus, within any collective, especially one composed of individuals from diverse cultural or functional backgrounds, members may hold varying near- and distant-future (and -past) time orientations.

Prior research has indicated that group-level variations in temporal depths can be beneficial because they mitigate the cognitive biases associated with actors who exhibit either near- or distant-future (or -past) time orientations (Bakker & Knoben, 2015; Blagoev et al., 2024; J. Chen et al., 2021; Reilly et al., 2016; Slawinski & Bansal, 2012; Stjerne et al., 2019). An exclusive focus on either the near or distant future (or past) can result in an imbalance between the consideration of long-term and short-term organizational needs (Smith & Lewis, 2011). J. Chen and colleagues (2021) found that organizations with a near-future (or -past) time orientation tend to engage in exploitation activities to a greater extent than exploration. In contrast, organizations with a distant-future (or -past) time orientation prioritize exploration over exploitation. Stjerne and colleagues (2019) provided evidence for the detrimental impact of homogeneous temporal depth on organizational collaboration. If project partners focus solely on the distant future, they become inflexible regarding initially established objectives and experience frustration when these objectives are altered. Conversely, if collaboration partners focus exclusively on the near future, they lack a basis for developing common goals. This uncertainty about the goals of the collaboration prevents project partners from

investing resources in it. A group consisting of individuals with heterogeneous temporal depths can help balance short-term and long-term concerns.

Still, such groups may also experience heightened tensions and conflicts. J. Chen and colleagues (2021) show that the positive relationship between time horizon diversity and a balanced pursuit of exploration and exploitation activities is moderated by temporal leadership. This observation suggests that groups with diverse temporal depths may experience tensions and conflicts that require effective coordination to be resolved. Such tensions may concern the prioritization of long-term versus short-term goals, the allocation of resources to corresponding tasks (J. Chen et al., 2021; McGrath, 1991), or may stem from a lack of understanding of other group members' shorter or longer time horizons (Stjerne & Svejenova, 2016). More attention should be paid to the tensions and conflicts that may arise as a consequence of differences in individual- or group-level temporal depth, and on how to overcome them.

Time urgency:

Time urgency is defined as "a consistent concern with the passage of time, with those higher in time urgency feeling generally hurried across situations" (Shipp & Cole, 2015, p. 243). This temporal orientation originates from psychological research on the Type A behavior pattern[1], in which time urgency constitutes a central component (Conte et al., 1995; Friedman & Rosenman, 1974; Landy et al., 1991). It is important to distinguish time urgency from related constructs such as time pressure and time compression. While time pressure and time compression refer to external conditions that compel actors to accelerate their pace, time urgency is an internal disposition (Rastegary & Landy, 1993), meaning that individuals with high time urgency tend to act hastily regardless of situational demands.

Time-urgent actors exhibit two primary characteristics. First, they display high levels of general hurry, defined as "the extent to which [they] rush when performing activities" (Conte et al., 2001, p. 1729). Second, they demonstrate a high level of deadline control,

[1] "A characteristic action-emotion complex which is exhibited by those individuals who are engaged in a relatively chronic struggle to obtain an unlimited number of poorly defined things from their environment in the shortest period of time and, if necessary, against the opposing effects of other things or persons in this same environment." (Friedman & Rosenman, 1974, p. 67).

referring to "the extent to which [they] create or appear to be controlled by deadlines" (Conte et al., 2001, p. 1729). In contrast, time-patient actors (Mohammed & Harrison, 2013) do not exhibit these traits to the same extent. Time-urgent individuals experience a persistent sense of urgency, driven by the perception of time as a scarce resource (Ballard & Seibold, 2003; Burnam et al., 1975). Unlike time-patient actors, they are acutely aware of the passage of time, frequently monitoring the remaining time available to complete tasks (Price, 1982) and often underestimating how much time is available (Burnam et al., 1975; Conte et al., 2001). In contrast, time-patient actors tend to overestimate the time available for task completion (Burnam et al., 1975). Time-urgent actors feel significant pressure to use their time as efficiently as possible. As a result, they often overschedule, committing to more activities than can reasonably be completed within a given timeframe (Friedman & Rosenman, 1974; Waller et al., 2001). They frequently respond with impatience or irritation to delays caused by external circumstances or the slower pace of others (Conte et al., 2001; Glass et al., 1974). During periods of inactivity, they tend to experience restlessness and perceive time as being wasted (Frankenhaeuser et al., 1980). Their behavior is often dominated by deadlines (Gastorf, 1980; Glass et al., 1974), which function as coping mechanisms to retain a sense of control over potentially uncontrollable events (Burnam et al., 1975; Glass et al., 1974). In contrast, time-patient actors attach less importance to deadlines and may even disregard them altogether (Landy et al., 1991). The combination of general hurry and deadline control results in greater working speed among time-urgent actors compared to their time-patient counterparts, who generally struggle to maintain a fast pace (Conte et al., 2001; Frankenhaeuser et al., 1980). Consequently, time-urgent actors are more likely to exhibit monochronic behavior (Waller et al., 1999) and to minimize time spent coordinating with others (Leroy et al., 2015), as both can disrupt their tempo. In trade-offs between speed and quality (Beersma et al., 2003; Karau & Kelly, 2004; Perlow et al., 2002; Tang et al., 2020), time-urgent actors typically prioritize speed, whereas time-patient actors favor quality (Mohammed & Harrison, 2013). Ultimately, an actor's level of time urgency shapes their temporal preferences and their propensity to establish fast-paced temporal patterns (Jansen & Kristof-Brown, 2005).

Like temporal depth, time urgency is regarded as a relatively stable cognitive trait, as demonstrated by the findings of Conte and colleagues (1995) and Landy and colleagues (1991). Although it has occasionally been suggested that situational factors may influence time urgency (Conte et al., 1995), the majority of research indicates that it is difficult to

modify an individual's level of time urgency, for example, through time management training (Bartel & Milliken, 2003; Conte et al., 1998; Macan, 1994). Given its classification as a stable, cognitive trait, most studies have investigated time urgency at the individual level. However, some scholars have examined the construct at the group level (Jansen & Kristof-Brown, 2005; Mohammed et al., 2017; Mohammed & Harrison, 2013; Mohammed & Nadkarni, 2011; Tang et al., 2020). For example, Mohammed and colleagues (2017) have extensively studied temporal diversity in teams, focusing on variation in time urgency among members (Mohammed et al., 2017; Mohammed & Harrison, 2013; Mohammed & Nadkarni, 2011). Their research examines how misalignment between an individual's time urgency and that of the team affects both willingness to support others and overall satisfaction (Mohammed et al., 2017; Jansen & Kristof-Brown, 2005). The notion of collective time urgency was also introduced by Landy and colleagues (1991), who found that research departments tend to exhibit lower collective levels of time urgency than marketing departments. They also identified certain occupations as inherently high in time urgency. Subsequent studies have supported this notion, by demonstrating elevated levels of time urgency among emergency room personnel (Bartel & Milliken, 2003; Klein et al., 2007; Mohammed & Harrison, 2013; Mohammed & Nadkarni, 2011) or among surgeons, who are described as "swift" and "beating the clock" (Oborn & Barrett, 2021, p. 394).

Discrepancies in time urgency within groups can have both beneficial and detrimental effects. Mohammed and colleagues (2017), in a field experiment involving chef teams at a culinary institute, found that variation in individual time urgency enhanced team performance. The authors attribute this effect to the task context, in which the need to balance speed and quality is particularly pronounced. In the restaurant setting, food must be delivered quickly while also meeting quality standards of taste and presentation. Teams comprising individuals with differing levels of time urgency were better able to balance these competing demands than teams consisting solely of either highly time-urgent or time-patient members (Bartel & Milliken, 2003; Mohammed & Harrison, 2013; Mohammed & Nadkarni, 2011).

Conversely, temporal diversity in time urgency can create a context in which tensions arise that undermine team performance. When actors differ in their inherent levels of time urgency, tensions may emerge from the trade-off between speed and quality (Mohammed et al., 2017; Mohammed & Harrison, 2013; Schriber & Gutek, 1987). These differences

can lead to misalignment in prioritizing tasks, scheduling activities, and adhering to deadlines (Bartel & Milliken, 2003; Jansen & Kristof-Brown, 2005; McGrath, 1991; Mohammed et al., 2017; Mohammed & Nadkarni, 2011; Waller et al., 2001), thereby hindering coordination (Mohammed & Harrison, 2013). Time-urgent individuals tend to schedule and complete as much work as possible within a limited timeframe, often adhering strictly to self-imposed deadlines. When time-patient team members complete fewer tasks within the same period or ignore deadlines, time-urgent actors may respond with frustration (Mohammed & Harrison, 2013). Conversely, time-patient actors may become frustrated when time-urgent team members sacrifice quality for speed. Research also shows that time-patient actors often experience anxiety in response to time-urgent demands, which may lead them to avoid collaboration (Jansen & Kristof-Brown, 2005; Waller et al., 2001). This explains the observation that temporally diverse teams tend to become fragmented by splitting into subgroups with similar levels of time urgency (Bartel & Milliken, 2003; Mohammed & Harrison, 2013). These subgroups may exhibit in-group favoritism (Tajfel & Turner, 1978), viewing the other subgroup with distrust and suspicion. Consequently, failures are frequently attributed to the other subgroup, further undermining collaboration between time-urgent and time-patient members (Blount & Janicik, 2002; Mohammed & Harrison, 2013).

Bluedorn and Standifer (2004) argue that discrepancies in actors' levels of time urgency tend to generate more tensions than performance gains. However, the extent to which such tensions arise depends on the context in which actors operate. In high-urgency occupations, where speed is paramount, heterogeneity in time urgency has been shown to impair performance (Klein et al., 2007; Oborn & Barrett, 2021). In such settings, time-urgent individuals are typically better equipped to meet task demands. Consequently, teams with mixed levels of time urgency may be less aligned with the performance requirements of high-urgency occupations and underperform compared to more homogeneous teams composed solely of time-urgent members (Mohammed & Harrison, 2013; Schriber & Gutek, 1987).

This chapter has focused on temporal depth and time urgency as two distinct temporal orientations that have been shown to vary across individuals and groups. While research acknowledges the benefits of heterogeneous temporal orientations, it also highlights that actors with differing temporal depth and time urgency encounter coordination challenges and conflicts that must be reconciled to enable effective collaboration. Together with the

temporal conceptions described in the previous chapter, temporal orientations form the deep-level temporal structures (Rowell et al., 2016) that strongly influence the visible temporal patterns actors prefer to use. These visible temporal patterns, the third dimension of temporal structures as defined by Ancona, Okhyusen, and colleagues (2001), are introduced in the following chapter.

2.1.3. Temporal patterns

Temporal patterns are the visible, surface-level expressions of deeper temporal structures (Rowell et al., 2016) or, put differently, "the visible anchoring of activities in relation to time" (Blagoev et al., 2024, appendix). Temporal patterns emerge through the placement of activities and events along a temporal continuum, a process Ancona, Okhuysen, and colleagues (2001) describe as mapping activities to time. Temporal patterns reflect temporal mapping choices about single activities (e.g., start and end points, duration), multiple activities (e.g., sequences), and recurring activities (e.g., cycles, rhythms) (Ancona, Okhuysen, et al., 2001; Blagoev et al., 2024). This chapter outlines the various temporal mapping choices that shape temporal patterns and defines key constructs commonly used to describe them (summarized in Table 2). It also highlights research that explicitly investigates tensions arising from misaligned temporal patterns of actors engaged in collaborative work.

Ancona, Okhuysen, and colleagues (2001) begin by outlining the temporal mapping of single activities. This involves decisions about start points, end points, durations, and the pace at which activities or events unfold. A start point marks when actors initiate a specific activity or event, while the end point indicates when that activity or event concludes. The placement of end points, which are often referred to as deadlines (Blagoev et al., 2024; Labianca et al., 2005), is a single mapping choice that plays a central role for creating and interpreting temporal patterns. They function as temporal boundary objects (Blagoev et al., 2024; Yakura, 2002), providing a shared temporal reference that facilitates the coordination of activities across various actors. Duration refers to the amount of time that elapses between an activity's or event's start and endpoints (Ancona, Okhuysen, et al., 2001; Zerubavel, 1976). The choice of start and end points, and thus the duration of activities or events, is closely tied to pace, defined as the tempo, speed, or rate

at which an activity is performed, or an event unfolds (Ancona, Okhuysen, et al., 2001; Ballard & Seibold, 2003; Lauer, 1981; Levine, 1988; Moore, 1963). Pace can shape start points, end points, and durations as temporal mapping choices, while also being shaped by them in return.

Table 2: Summary of temporal mapping choices

Temporal mapping choices of single activities		
start point	the determined beginning of an activity, set of multiple activities, or cycle	Ancona, Okhuysen et al. (2001)
end point	the determined end of an activity, set of multiple activities, or cycle	Ancona, Okhuysen et al. (2001)
duration	the elapsed time between the start and the end of an activity, set of multiple activities, or cycle	Ancona, Okhuysen et al. (2001); Zerubavel (1976)
pace	the speed, tempo or rate with which an activity, set of multiple activities, or cycle is carried out	Ancona, Okhuysen et al. (2001); Ballard & Seibold (2003); Lauer (1981); Levine (1988); Moore (1963), Zerubavel (1976)
Temporal mapping choices of repeated activities		
cycle	"a single completed execution of a periodically repeated phenomenon", which can be an activity, a set of multiple activities, or an event	Ancona & Chong (1996); Ancona & Chong (1999, p. 6), McGrath & Rotchford, 1983; Sandra et al. (2023)
rhythm	consistent patterns of repetitions that are defined by their frequency, regularity, and sequence (only in the case of repetitions of multiple activities)	Ancona, Okhuysen et al. (2001); Katila et al. (2020); Cunha (2009); Zhang et al. (2023)
frequency	the rate of occurrence of a repetition during a specific time period	Ancona, Okhuysen et al. (2001); Eddleston et al. (2020); Zhang et al. (2023)
regularity	the variability in the frequency of activities, sets of activities, phases, or cycles with activities being either distributed relatively equally or unequally over a given time period	Klarner & Raisch (2013); Zhang et al. (2023);
Temporal mapping choices of multiple activities		
sequence	the order of activities within a set of activities that determines which activity is to follow or precede another activity	Ancona, Okhuysen et al. (2001); Langley (1999); Moore (1963); Schriber & Gutek (1987); Zerubavel (1976)
linearity	the sequential or concurrent execution of activities, set of multiple activities, or cycles	Ballard & Seibold (2003)

In addition, temporal patterns encompass temporal mapping choices related to the repetition of activities. Any activity that recurs over time can be described as a cycle, that is, a "single complete execution of a periodically repeated phenomenon" (Ancona & Chong, 1996, 1999; McGrath & Rotchford, 1983; Sandra et al., 2023). Such a

phenomenon may involve a single activity, a set of activities, a single event, or multiple events (Zhang et al., 2023). The temporal mapping choices that apply to single activities, such as start and end points, duration, and pace, can likewise be applied to cycles (Ancona, Okhuysen, et al., 2001). Actors may define the start and end points of a cycle, determine its duration, and thereby shape its pace. In the context of repeated activities, mapping choices particularly focus on rhythms, characterized by their frequency and regularity (Zhang et al., 2023; Klarner & Raisch, 2013).

Research occasionally employs the terms cycle and rhythm interchangeably (McGrath & Kelly, 1986). However, it is important to distinguish between these concepts. A cycle denotes the repetition of an activity or event (or a set of activities or events), whereas a rhythm characterizes the manner in which this repetition occurs. In other words, a rhythm describes the nature of the repetition (Ancona & Chong, 1999; Sandra et al., 2023).

The literature provides various definitions of rhythms. Some scholars define rhythms as consistent patterns of repetition (Ancona, Okhuysen, et al., 2001; Ancona & Chong, 1999; Zhang et al., 2023), while others emphasize the intrinsic attributes of a rhythm. For instance, Katila and colleagues (2020, p. 1311) describe rhythms as "repetitive temporal patterns that define the order and tempo of practice." This definition highlights two critical dimensions: the existence of a consistent sequence of activities within repeating cycles and the frequency with which these cycles recur. The term frequency refers to the number of times an activity or set of activities is repeated within a specific timeframe (Eddleston et al., 2020; Zhang et al., 2023). Notably, Cunha (2009) stresses that rhythms inherently involve regular repetitions of activities or events. Nevertheless, some researchers conceptualize rhythms in terms of their degree of regularity, suggesting that rhythms may be either regular or irregular (Klarner & Raisch, 2013). Regularity pertains to the variability in the frequency of activities or cycles, with repetitions being distributed either evenly or unevenly over a given period. This apparent contradiction regarding whether the degree of regularity is a characteristic of a rhythm can be reconciled through the two temporal conceptions: clock time and event time. Some rhythms are anchored in clock time, with repetitions evenly spaced across defined intervals (Eisenhardt & Brown, 1998; Klarner & Raisch, 2013; Sastry, 1997). These clock-time-based rhythms are regular and thus predictable. In contrast, other rhythms are grounded in event time, wherein repetitions are triggered by specific events (Brown & Eisenhardt, 1997; Eisenhardt &

Brown, 1998; Klarner & Raisch, 2013). From the perspective of conventional Western clock time, event-time-based rhythms appear irregular and often unpredictable.

Finally, the temporal framework proposed by Ancona and colleagues (2001) encompasses the mapping of multiple activities onto time. This generally involves the same mapping choices required for single or repeated activities, namely defining start and end points, durations, and paces for either the entire sequence or for each activity within it. When sequences of multiple activities are repeated, actors must also determine the rhythm at which these repetitions occur. However, the mapping of multiple activities introduces two additional temporal mapping choices. First, actors must determine the sequence of activities, which refers to the prescribed order in which activities occur within a set of multiple activities (Langley, 1999; Moore, 1963; Schriber & Gutek, 1987; Zhang et al., 2023). Thus, the sequence determines which activity precedes or follows another. Second, actors must decide on the linearity of activity execution, that is, whether multiple activities are performed sequentially or concurrently (Ballard & Seibold, 2003).

The terminology used to describe the temporal mapping process varies across the literature. Most commonly, scholars refer to scheduling (Ballard & Seibold, 2003; Lauer, 1981; McGrath & Kelly, 1986; Schriber & Gutek, 1987; Zerubavel, 1976), timing (Desai & Madsen, 2022; Dille & Söderlund, 2011; Merida & Rocha, 2021; Turner & Rindova, 2018; Zerubavel, 1976), or pacing (Ancona & Waller, 2007; Brown & Eisenhardt, 1997; Cunha, 2004; Sastry, 1997; Waller et al., 2002). Notably, these terms are used so interchangeably across studies that no clear conceptual demarcation emerges from the existing literature. McGrath and Kelly (1986, p. 109) define scheduling as the determination of "when some event will occur […] in relation to an external calendar and clock." From this perspective, scheduling involves decisions regarding start and end points as well as the sequence of activities (Lauer, 1981; McGrath & Kelly, 1986). Schriber and Gutek (1987) adopt a broader definition, emphasizing that scheduling encompasses the delineation of temporal boundaries, including start and end points, durations, sequences, and pace of activities. Timing and scheduling are often used interchangeably, for example, by Kunisch and colleagues (2021). Zerubavel (1976) defines timing as the selection of start and end points and the sequence of activities, a definition closely aligned with the conception of scheduling found in McGrath and Kelly (1986) and Lauer (1981). In contrast, Merida and Rocha (2021) and Desai and Madsen

(2022) use the term timing more narrowly, referring specifically to choices about start and end points. Conversely, Dille and Söderlund (2011) employ a broader understanding of timing, one that aligns with Schriber and Gutek's (1987) inclusive definition of scheduling, and which incorporates the pace of activity execution. Ancona and Waller (2007, p. 117) define pacing as "the timing of when change occurs", suggesting an overlap with the concept of timing. However, Waller and colleagues (2002, p. 1047) use the term task pacing to refer more narrowly to "the rate of task performance activity." Similarly, Gersick (1994) employs pacing to describe the speed at which work groups complete their tasks. These conceptual overlaps are to be expected, given the inherently interconnected nature of the temporal mapping choices to which the terms refer. For instance, the specification of start and end points is inextricably linked to the pace at which activities are conducted. When actors establish tight deadlines, they are implicitly required to accelerate their pace (Gersick, 1994; Labianca et al., 2005).

To ensure that the terms timing, scheduling, and pacing are used clearly and consistently, this dissertation adopts the following definitions. In line with Merida and Rocha (2021) and Desai and Madsen (2022), timing refers to "the positioning of activities and events in chronological time (the 'when' of performing a particular action), especially in terms of choosing the right time to perform a specific activity." (Blagoev et al., 2024, appendix). This encompasses temporal mapping choices related to the start and end points of activities, as well as closely connected aspects such as duration, linearity, and the regularity of rhythms. The term scheduling is used to denote temporal mapping choices primarily concerned with the sequence in which activities are arranged. Finally, pacing refers to temporal mapping choices focused on determining or adjusting the speed at which activities unfold (Amis et al., 2004; Blagoev et al., 2024; Waller et al., 2002), including decisions regarding the frequency of rhythms.

Disparate temporal patterns arise when actors make different temporal mapping choices regarding the timing, scheduling, and pacing of activities and events. The reasons for such variation are multifaceted. As Ancona, Okhuysen, and colleagues (2001) suggest, these differences may stem from actors' distinct temporal conceptions and orientations. Dougherty and colleagues (2013) confirm that differing temporal conceptions among managers and scientists lead to different timing choices. Managers, who are oriented toward clock time, define temporal milestones using standardized time units, resulting in

regular and predictable work rhythms. In contrast, scientists operate with an event-time conception, establishing temporal milestones based on learning events. As a result, their timing choices generate more irregular and less predictable rhythms than those of managers. In their review of strategy rhythms, Zhang and colleagues (2023) conclude that a variety of environmental, organizational, and managerial antecedents shape the frequencies and regularities of rhythms. A firm's external environment can significantly influence its temporal patterns, as managers often synchronize with the temporal rhythms of a dominant Zeitgeber (Ancona & Chong, 1996; Pérez-Nordtvedt et al., 2008). If the dominant Zeitgeber, such as a major competitor, has established fast and regular rhythms, the focal firm is likely to adapt accordingly. Thus, when two collaborating firms align with different dominant pacers, their temporal patterns may diverge. At the organizational level, actors' choices regarding rhythm design may depend on governance structures (Connelly et al., 2010), the availability of (slack) resources (Ferrier, 2001; Schinoff et al., 2020), or the organization's experience in a given activity (Nadolska & Barkema, 2007). Nadolska and Barkema (2007) demonstrate that organizations with greater acquisition experience exhibit faster acquisition rhythms compared to less experienced firms. Managerial antecedents of temporal patterns primarily include the temporal orientations of managers, such as their temporal depth (Nadkarni et al., 2016). Accordingly, depending on specific environmental, organizational, and managerial characteristics, actors are likely to make different temporal mapping choices.

Any discrepancy in the timing, scheduling, or pacing of activities has the potential to create asynchrony between individuals and within collectives, which is widely regarded as detrimental (Ancona & Chong, 1996; Dille & Söderlund, 2011; Pérez-Nordtvedt et al., 2008). Research shows that organizations whose temporal patterns are misaligned with those of a dominant Zeitgeber often experience inefficiencies and performance losses (Ancona & Chong, 1996; Gersick, 1994; Khavul et al., 2010). These negative consequences are typically attributed to two primary mechanisms. First, Pérez-Nordtvedt and colleagues (2008) offer an institutionalist explanation. Synchrony with a dominant Zeitgeber confers legitimacy, which in turn facilitates access to critical resources. In contrast, temporal misalignment may be perceived as illegitimate, thereby hampering resource acquisition. Second, consistent with the concept of shared temporal structures (Orlikowski & Yates, 2002), asynchronous temporal patterns are thought to obstruct

effective coordination among actors (Ancona & Chong, 1999; Dille & Söderlund, 2011; Jarvenpaa & Välikangas, 2022).

The detrimental impact of divergent, asynchronous temporal patterns on coordination is particularly evident in studies of temporary organizing and interorganizational collaboration (Dahlgren & Söderlund, 2001; Dille et al., 2018, 2023; Dille & Söderlund, 2013; Stjerne et al., 2019; van Berkel et al., 2016). For example, van Berkel and colleagues (2016) examine the coordination between a temporary project organization and a permanent public organization within a Dutch infrastructure project. Their findings reveal pacing asynchronies. The temporary organization operated at a considerably faster pace than its permanent partner organization. This disparity frequently led to delays, as the temporary organization had to await input and approvals from the slower permanent organization. Eventually, due to mounting frustration and an unwillingness to accept further delays, the temporary project organization began making decisions unilaterally, foregoing coordination with the partner organization. From the perspective of the permanent public organization, this lack of coordination compromised project outcomes, privileging rapid completion over long-term functionality. Similarly, a series of studies on a Norwegian emergency call system project illustrate how asynchrony impeded coordination between three project partners (Dille et al., 2018, 2023; Dille & Söderlund, 2013). A key challenge lay in the uneven pace at which partners progressed with the implementation of a new emergency call system. One partner aimed to initiate a nationwide rollout significantly earlier than the other was ready to do. Given the need for a synchronized launch, this phase misfit posed a serious threat to the project's success. As a result, project managers had to devote substantial time and effort to accelerating the slower partners, attempting to recreate a temporal fit and avoid asynchrony during deployment. In a different context, Stjerne and colleagues (2019) identify a pacing misfit between collaborating project partners as a source of frustration and escalating conflict in joint meetings. The pacing misfit disrupted the partners' ability to engage in what was perceived as essential to effective collaboration – an "honest and open dialogue" (p. 355).

The aforementioned studies demonstrate that disparate temporal patterns can significantly impede coordination and collaboration among actors. They also suggest that such disparities may be rooted in actors' divergent temporal orientations. As van Berkel and colleagues (2016) argue, the accelerated pace of temporary project units can be traced to

a heightened sense of urgency perceived by actors operating within these temporary structures. Similarly, Dille and colleagues (2018; 2023) and Dille and Söderlund (2013) link the observed pacing misfits to conflicting temporal orientations among project partners. While one partner exhibited a strong preference for rapid implementation, another partner prioritized functionality and precision over speed. These findings highlight the pivotal role of actors' temporal orientations not only in shaping the temporal patterns they enact but also in influencing how they respond to divergent temporal patterns.

In sum, Chapter 2.1 pursued two principal objectives. First, it introduced the framework proposed by Ancona and colleagues (2001) to establish a foundation for understanding the multidimensionality of temporal structures, namely, temporal conceptions, temporal orientations, and temporal patterns, as well as the key terms and constructs used to describe them. Second, it demonstrated that differences can arise across all three dimensions of temporal structures. In the subsequent chapter 2.2, I will build on this foundation by conceptualizing and illustrating the temporal differences that emerge between agile and non-agile organizational units, which constitute the focal research setting of this dissertation.

2.2. An illustration of divergent temporal structures: Agile and non-agile organizational units

Chapter 2 has thus far established the conceptual foundations necessary for understanding temporal structures. It has demonstrated how actors' temporal structures differ in terms of temporal conceptions, orientations, and patterns (Ancona, Okhuysen, et al., 2001; Rowell et al., 2016). The chapter concludes with an illustrative example of divergent temporal structures, contrasting agile and non-agile organizational units. In this context, an agile unit is defined as one that employs agile methods to carry out its tasks. Conversely, a non-agile unit is characterized by the absence of such methods. Because this definition of non-agility is based solely on the non-use of agile practices, the

underlying work practices in non-agile units can vary considerably. These may include plan-driven practices as well as highly spontaneous ad-hoc practices (Harris et al., 2009), which are particularly common in organizational units with direct customer interaction (Mena & Parker, 2024). The choice of agile and non-agile units to illustrate divergent temporal structures provides a conceptual bridge to the empirical setting of this dissertation and establishes a foundation for its understanding.

This chapter is structured as follows: Section 2.2.1 provides the conceptual foundations for understanding the work practices that underlie agile and non-agile organizational units. To this end, I juxtapose the agile Scrum method (Fitzgerald et al., 2006; Schwaber & Sutherland, 2020), which is consistently applied across all agile units examined in this dissertation, with the core principles of plan-based methods, which are employed by most of the non-agile units. Once these conceptual foundations are established, Section 2.2.2 turns to the divergent temporal structures of agile and non-agile units. To the best of my knowledge, these temporal differences have not yet been systematically investigated. I therefore derive the respective temporal structures from the foundational values, principles, and documented practices of both plan-based and agile (Scrum) methodologies. As such, the temporal structures presented in this chapter are conceptual in nature and will be subject to empirical validation in the findings.

2.2.1. Agile and plan-based methods

This chapter provides a conceptual introduction to the work practices and methods underlying agile and non-agile organizational units. It begins by outlining the key characteristics of plan-based methods, which serve as a representative form of non-agile work practices. I selected plan-based methods because they are employed by the majority of non-agile units examined in this dissertation and are commonly portrayed as the conceptual counterpart to agile methods.

Plan-based methods rest on the assumptions that task requirements are sufficiently well understood at the outset of a work process and remain largely stable over time (Boehm & Turner, 2004). These assumptions enable an upfront planning approach, which is characteristic of plan-based methods (Boehm & Turner, 2004; Karlstrom & Runeson,

2005). Through upfront planning, actors identify all activities and resources needed to fulfill the known tasks at the beginning of the work process and for its entire duration (Petersen & Wohlin, 2010). Project managers with decision-making authority are responsible for defining tasks and conducting the upfront planning (Pettersen, 1991). However, in fulfilling this role, they may consult with the team responsible for operational task execution (Tessem, 2014).

Upfront planning typically results in a work process as illustrated in Figure 2. Similar activities, or sets of activities, are grouped into discrete phases and meticulously scheduled throughout the entire work process (Boehm & Turner, 2004; Fernandez & Fernandez, 2008; Oellgaard, 2013; Petersen & Wohlin, 2010). Following the initial planning phase, the plan-based work process usually proceeds with the operational implementation of task requirements, which can be further subdivided into multiple phases (Project Management Institute, 2021). Sequential dependencies between activities across phases, where activity B depends on the outcomes of a preceding activity A, necessitate their consecutive execution (Fernandez & Fernandez, 2008; Gemino et al., 2021). Because task requirements are expected to change minimally, if at all, these phases are performed linearly (Boehm & Turner, 2004). Nevertheless, returning to earlier phases is possible, albeit costly due to the rework involved in revisiting completed activities (Repenning, 2001). In the final phase, external or internal customers receive an outcome that fulfills the requirements specified in the upfront plan. Due to the linear and sequential

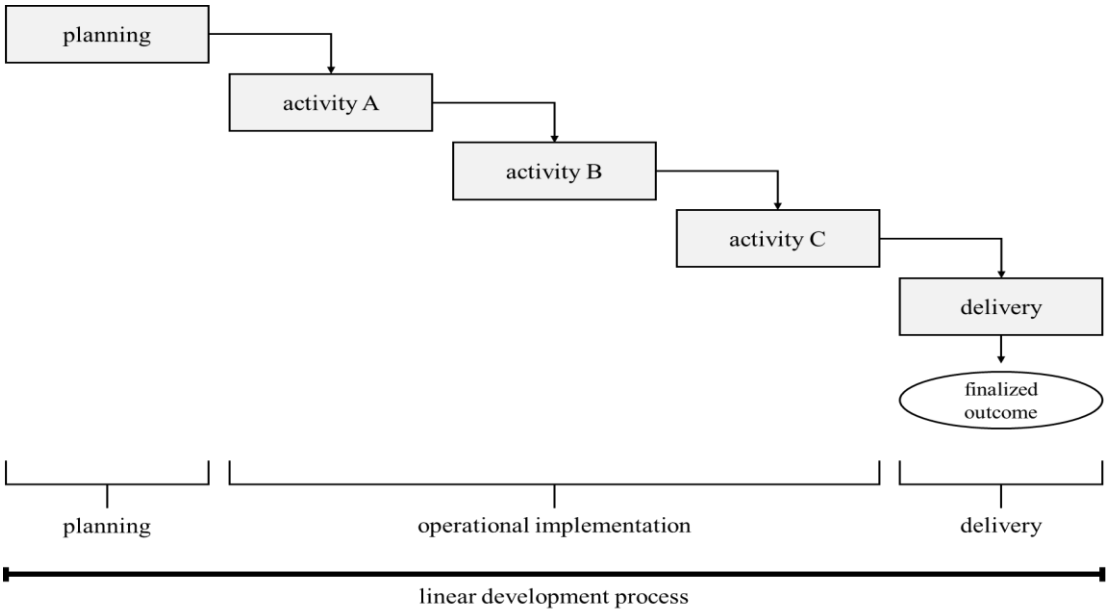

Figure 2: Representation of a plan-based work process

nature of plan-based work processes, customers typically do not receive usable intermediate deliverables, leading to the final delivery often being described as a "big bang" (Karlstrom & Runeson, 2005, p. 48).

Throughout the entire work process, the plans developed in the initial phase serve as the primary coordination instruments (Lappi et al., 2018; Project Management Institute, 2021). They specify requirements regarding the quality, costs, and delivery dates of the final outcome, as well as delineate the activities to be executed, their schedules and deadlines, and the resources required to perform them (Karlstrom & Runeson, 2005). These plans function as artefacts continuously used to ensure the accuracy of the final outcome. To this end, plans also define milestones, understood as "significant points or events" within the work process (Project Management Institute, 2021, p. 243), which are employed to monitor task progress during operational implementation (Lappi et al., 2018). While other coordination mechanisms may be used, they play a minor role compared to upfront plans. For instance, mutual adjustment through customer interaction is minimal and largely confined to the planning and delivery phases (Bianchi et al., 2020; Boehm & Turner, 2004; Nerur et al., 2005), as it becomes largely redundant given the detailed specifications of the plans, which are expected to remain valid throughout the work process.

By specifying tasks, activities, and resources for the entire duration of a work process, plan-based methods seek to control uncertainty and avoid change (Bianchi et al., 2020; Cooper & Sommer, 2018). In doing so, they promote predictability, stability, and assurance (Boehm & Turner, 2004; Vinekar et al., 2006) in contexts where tasks are sufficiently well understood at the outset and expected to remain largely stable over time (Boehm & Turner, 2004; Gemino et al., 2021; Port & Bui, 2009).

Agile methods are often regarded as the conceptual opposite of plan-based methods (Boehm & Turner, 2004). They differ fundamentally in their core assumptions, positing that tasks are neither sufficiently known nor well understood at the outset of a work process, and that these requirements may change substantially over time (Boehm & Turner, 2004; Port & Bui, 2009). Given these assumptions, the upfront planning characteristic of plan-based methods proves ineffective, as frequent changes undermine the validity of plans throughout the work process. Consequently, rather than relying on upfront planning, agile methods employ iterative planning, regularly determining the

activities and resources needed to implement them over relatively short time horizons (Ghosh & Wu, 2023; Junker et al., 2023).

The iterative planning approach characteristic of agile methods leads to a work process that contrasts sharply with the linear and sequential nature of plan-based work processes. As illustrated in Figure 3, the agile process is structured around recurring cycles known as sprints, rather than discrete phases containing different sets of activities (Schwaber & Sutherland, 2020; van Oorschot et al., 2018). The Scrum framework recommends that each sprint should last between one and four weeks (Schwaber & Sutherland, 2020). Each sprint constitutes a self-contained work cycle that encompasses the same range of activities as an entire plan-based process, albeit in a compressed and scaled-down form due to its shorter duration (Bianchi et al., 2020; Lindkvist et al., 1998). For example, each sprint begins with a condensed sprint planning event during which task requirements, activities, and resources are determined for the upcoming sprint, typically within a few hours (Beck et al., 2001). Unlike plan-based methods, the operational Scrum team is responsible for planning. A product owner, who represents customer interests and is consistently available to the team (Hobbs & Petit, 2017; Schwaber & Sutherland, 2020), proposes a set of requirements, called user stories in Scrum terminology (Cohn, 2004). However, the Scrum team autonomously decides the number of user stories to include in the forthcoming sprint, thereby determining the task requirements as well as the associated activities and resources (Schwaber & Sutherland, 2020).

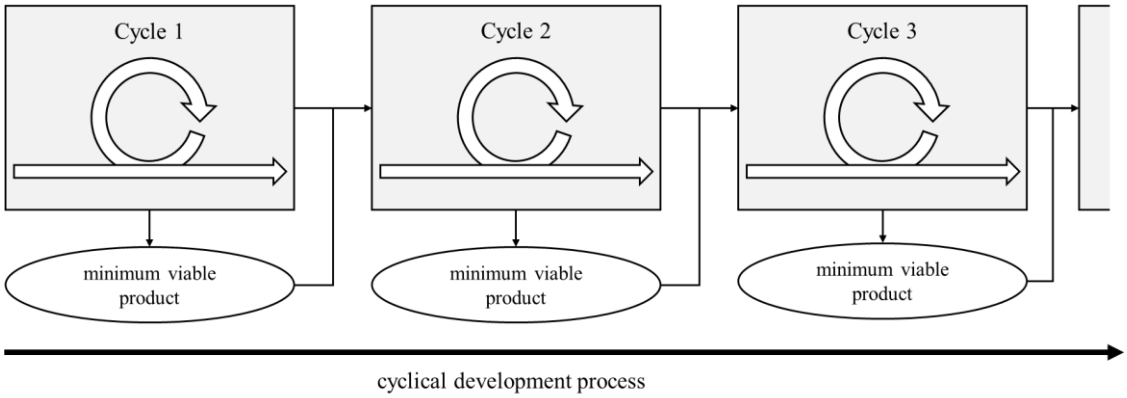

Figure 3: Representation of an agile work process

The activities involved in executing these requirements are similarly compressed in time. Given this temporal compression, sprints do not produce fully finalized outcomes; instead, each sprint delivers a minimum viable product (S. Lee & Geum, 2021). This

minimum viable product is an intermediate deliverable featuring basic functionalities that customers can potentially use and that will be iteratively refined and adapted in subsequent sprints (Ceschi et al., 2005; Ghosh & Wu, 2023; Junker et al., 2023). The total number of sprints required to complete the entire agile work process is intentionally undefined (Schwaber & Sutherland, 2020), allowing for potentially continuous repetition (Baskerville & Pries-Heje, 2004; Cooper & Sommer, 2018).

Within agile work processes, coordination is primarily achieved through mutual adjustment (Thompson, 1967). The Scrum method prescribes a variety of meeting formats that promote regular interaction both within the Scrum team and between the team and its customers (Schwaber & Sutherland, 2020). First, daily standup meetings are mandated, during which Scrum team members convene for 15 minutes to share what tasks they completed the previous day, what they plan to accomplish that day, and to identify any obstacles impeding progress (Ghosh & Wu, 2023; Junker et al., 2023; Schwaber & Sutherland, 2020), Second, a sprint review is held at the end of each sprint, where the team presents the minimum viable product to customers (Schwaber & Sutherland, 2020). This meeting ensures continuous customer involvement (Nerur et al., 2005), enabling customers to evaluate the deliverables and provide feedback for improvement (Boehm & Turner, 2004; Nerur et al., 2005). Based on this feedback, the product owner updates the product backlog, a prioritized list of customer requirements that serves as "the single source of work undertaken by the Scrum team" (Schwaber & Sutherland, 2020, p. 10), which forms the foundation for planning subsequent sprints (Ghosh & Wu, 2023; Junker et al., 2023). Both daily standups and sprint reviews serve not only to support coordination but also to monitor task progress continuously (Lappi et al., 2018). Third, the Scrum method includes retrospective meetings at the end of each sprint (Schwaber & Sutherland, 2020). Retrospectives provide structured time and space to reflect on collaborative experiences, with the aim of improving team dynamics and internal processes (Ghosh & Wu, 2023; Junker et al., 2023). All of these meetings are facilitated by the Scrum master, who is responsible for ensuring adherence to Scrum rules, fostering efficient communication and decision-making, and removing any impediments to progress (Schwaber & Sutherland, 2020; Shastri et al., 2021).

In addition to coordination through mutual adjustment, sprint planning also plays a vital role. During this event, the requirements, activities, and resources for the upcoming sprint

are documented in the sprint backlog (Schwaber & Sutherland, 2020). Serving as a dynamic alternative to traditional project plans, the sprint backlog provides orientation for the Scrum team and helps to visualize and track task progress throughout the sprint.

Unlike plan-based methods, agile methods do not seek to control uncertainty; instead, they aim to adapt to it and respond as effectively as possible to change (Bianchi et al., 2020). The increasing popularity of agile methods in recent years (Annosi et al., 2020) can be attributed, on the one hand, to this responsiveness and, on the other hand, to their perceived velocity that results from the delivery of usable intermediate outcomes throughout the entire work process. However, it is important to emphasize that agile methods are not universally superior to plan-based methods. Their suitability depends on several contextual factors, such as the ambiguity of requirements at the outset of the work process (Gemino et al., 2021) and the willingness of customers to engage in intensive and ongoing dialogue that agile approaches demand (Nerur et al., 2005). To conclude this section on the conceptual foundations of plan-based and agile methods, Table 3 presents a comparative overview of their key characteristics.

Table 3: Key characteristics of plan-based and agile methods

	Plan-based methods	**Agile methods**
core assumptions:	requirements are sufficiently understood at the outset of a work process and expected to change little over time	requirements are not sufficiently understood at the outset of a work process and expected to change a lot over time
task planning & resource allocation:	one-time upfront planning and resource allocation	recurrent iteration planning and resource allocation
work process:	divided into sequential phases, each containing a different set of activities	divided into recurrent cycles, each containing the whole spectrum of activities
work outcomes:	finalized solutions once at the end of the work process that meet the specifications documented in the upfront plans	minimum viable products after each sprint that are continuously improved over the course of the entire work process
primary coordination mechanisms:	plans	mutual adjustment, plans
performance indicators:	predictability, stability, and high assurance	responsiveness to change; velocity in value creation

2.2.2. Temporal structures in agile and non-agile organizational units

Chapter 2.2.1 established the conceptual foundations for understanding plan-based and agile methods, which are commonly employed in non-agile and agile organizational units, respectively. Chapter 2.2.2 builds on these foundations to describe the temporal structures that may characterize agile and non-agile units. This section pursues two main objectives. First, it provides a concrete illustration of divergent temporal structures as introduced in Chapter 2.1, and second, it lays the conceptual groundwork for selecting collaborations between agile and non-agile units as the empirical setting for this dissertation.

Since the non-agile units examined in this dissertation predominantly use plan-based methods, while the agile units exclusively follow the Scrum method, I will continue to treat these approaches as representative of non-agile and agile work practices, respectively. Although some research has addressed selected aspects of the temporal structures of plan-based and agile methods (Kalff, 2022; Langley & McGivern, 2024; Nachbagauer, 2022; O'Riordan et al., 2013; Ritter et al., 2024), to the best of my knowledge, none has provided a conceptual or empirical comparison of their temporal structures. However, the foundational descriptions provided in key sources of agile and plan-based methods (Beck et al., 2001; Project Management Institute, 2021; Schwaber & Sutherland, 2020) allow for such a conceptual comparison. The results of this comparison are summarized in Table 4 and will be presented in the following sections.

Temporal conceptions:

Chapter 2.1.1 introduced two fundamental conceptions of time. The clock-time conception views time as objective, quantitative, measurable, linear, and finite (Bluedorn & Denhardt, 1988; Jaques, 1982; Orlikowski & Yates, 2002). In contrast, the event-time conception considers time to be subjective, context-dependent, qualitative, and cyclical (Bluedorn & Denhardt, 1988; Jaques, 1982; Orlikowski & Yates, 2002).

Table 4: Temporal structures of plan-based and agile methods

	Plan-based methods	**Agile Scrum method**
temporal conceptions	▪ clock-time conceptions at the level of the single phase ▪ clock-time conceptions at the level of the entire work process	▪ clock-time conception at the level of the single sprint ▪ event-time conception at the level of the entire work process
temporal patterns	▪ phases as central temporal units ▪ phases have clearly defined start and end points, and their duration can be freely determined ▪ linear work processes with sequential execution of phases	▪ sprints as central temporal units ▪ sprints have clearly defined start and end points, and their duration is fixed between one and four weeks ▪ cyclical work processes with sprints recurring in fast and regular rhythms
temporal orientations:	▪ distant-future and -past time orientations ▪ facilitation of time patience	▪ near-future and -past time orientations ▪ facilitation of time urgency

Plan-based methods are deeply rooted in the conception of clock-time. This is reflected in the linear structure of plan-based work processes (Boehm & Turner, 2004; Fernandez & Fernandez, 2008; Kalff, 2022), which are characterized by a sequential progression through distinct phases, each of which must be completed before the next begins (Lindkvist et al., 1998). Plan-based methods rely on an objective view of time, in which time is perceived as standardized, measurable, and quantifiable. Clock time facilitates effective coordination through detailed upfront planning (Kalff, 2022; Project Management Institute, 2021), providing actors with a shared understanding of the temporal elements of plans, such as task sequences, activity durations, and deadlines for solution delivery (Drury-Grogan, 2014; Langley & McGivern, 2024). Moreover, the temporal elements embedded in upfront plans enact what Langley and McGivern (2024) term temporal closure, as plan-based work processes are typically defined by a clear end point (Batra et al., 2011; Špundak, 2014). Consequently, the time available for solution development is treated as finite and bounded.

In contrast, the agile Scrum method incorporates both clock-time and event-time conceptions. At the level of the individual sprint, a clock-time conception predominates. Time within a sprint is conceived as objective, quantitative, and measurable. The duration of the sprint itself, as well as that of the various events within it, is defined by precise, standardized units. For example, the Scrum Guide stipulates that a sprint lasts between one and four weeks, while daily standup meetings are fixed at precisely 15 minutes (Schwaber & Sutherland, 2020). Moreover, sprints unfold along a linear sequence of

events: they begin with sprint planning, continue with daily standup meetings, and end with a sprint review and a retrospective (Schwaber & Sutherland, 2020). Sprints are also strictly timeboxed, meaning they are bound by fixed deadlines that must not be extended (Cooper, 2014; Cooper & Sommer, 2016; Schwaber & Sutherland, 2020; van Oorschot et al., 2018). If a Scrum team risks missing a deadline, the deadline is maintained while the scope is reduced, for instance, by dropping lower-priority requirements (Bianchi et al., 2020; Drury-Grogan, 2014; Karlstrom & Runeson, 2005). Thus, in addition to being seen as objective, quantitative, and linear, time within an individual sprint is also treated as finite and non-negotiable.

Nevertheless, when moving from the level of the sprint to that of the entire work process, the underlying temporal conception shifts toward event time. The overall Scrum process is characterized by a recurring sequence of sprints (Schwaber & Sutherland, 2020), reflecting a cyclical pattern that contrasts with the linear progression of plan-based methods (Boehm & Turner, 2004). Although the conclusion of an individual sprint can be precisely defined using standardized clock-time units, the overall work process lacks a similarly fixed endpoint. Instead, it advances iteratively from one sprint to the next (Baskerville & Pries-Heje, 2004). In some cases, the endpoint of the entire process is tied to specific events, such as achieving a certain level of customer satisfaction, that are inherently ambiguous or subjective (Langley & McGivern, 2024). In other cases, the temporal scope of a work process may not be constrained at all, especially when actors perceive a need for continuous improvement (Schwaber & Sutherland, 2020). In such cases, time may be regarded as open-ended or even infinite.

In conclusion, plan-based methods predominantly embody a clock-time conception of time, viewing it as linear, objective, quantitatively measurable, and finite. The Scrum method also reflects a clock-time conception at the level of the individual sprint. However, its temporal structure diverges at the level of the overall work process, which is cyclical rather than linear. The endpoint of this process is not determined by objective and measurable time units, but by subjective events, or, in some cases, may not be defined at all, with time treated as an open-ended or even infinite resource. Thus, while clock-time conceptions clearly dominate in plan-based methods, the Scrum method integrates both clock-time and event-time conceptions.

Temporal orientations:

Both plan-based methods and the agile Scrum method appear to support specific temporal orientations more strongly than others. In Chapter 2.1.2, I introduced two temporal orientations in greater detail: temporal depth and time urgency. I expect that plan-based and agile methods differ in the extent to which they support these temporal orientations. Accordingly, the following section elaborates on how plan-based and agile methods accommodate varying levels of temporal depth and time urgency.

Temporal depth refers to the extent to which actors project their thinking into the future or the past (Bluedorn, 2002; Bluedorn & Standifer, 2006). Plan-based methods typically support a more distant temporal depth compared to the agile Scrum method. Plan-based methods involve extensive upfront planning for the entire work process (Boehm & Turner, 2004; Fernandez & Fernandez, 2008; Tereso et al., 2019), often spanning several months (Lindkvist et al., 1998; Petersen & Wohlin, 2010). The capacity to plan effectively over such extended periods necessitates a distant-future time orientation (Das, 1987). Additionally, optional lessons learned meetings, which are usually conducted once at the end of the project, require a distant-past time orientation, enabling actors to delve deeper into the past and thus reflect on extended parts of the entire process.

In contrast, the Scrum method tends to promote a near-future time orientation. In Scrum, planning is typically limited to the upcoming sprint (Boehm & Turner, 2004; Junker et al., 2023; Schwaber & Sutherland, 2020). This short-term planning horizon is aligned with the values of the Agile Manifesto, which emphasizes responsiveness (Beck et al., 2001), a quality widely associated with a near-future time orientation (Crossan et al., 2005; Nadkarni et al., 2016; Shipp et al., 2009). Scrum relies on deadlines that are "never distant in time" (van Oorschot et al., 2018, p. 80) and prioritizes short-term performance metrics (Lappi et al., 2018). Decision-making within Scrum teams often focuses on relatively brief timeframes and tends to reflect long-term strategic considerations less than in plan-based processes (Cooper & Sommer, 2018; Lappi et al., 2018). Both sprint reviews and retrospectives are confined to reflection on the most recent sprint cycle (Ritter et al., 2024; Schwaber & Sutherland, 2020), indicating a near-past time orientation.

Time urgency is defined as "a consistent concern with the passage of time, with those higher in time urgency feeling generally hurried across situations" (Shipp & Cole, 2015, p. 243). Plan-based methods appear to accommodate time patience rather than time urgency. As Boehm and Turner (2004) note, quality assurance is a primary goal within plan-based approaches, which clearly takes precedence over rapid delivery. This orientation aligns with the preferences of time-patient actors, who prioritize quality over speed, and contrasts with time-urgent actors, who emphasize speed over quality (Mohammed & Harrison, 2013). Moreover, the treatment of deadlines in plan-based methods tends to support time patience. Although deadlines are crucial in demarcating the end of project phases and the entire work process (Langley & McGivern, 2024), they are often postponed if customer requirements remain unmet (Boehm & Turner, 2004). This conflicts with the preferences of time-urgent actors, who seek strict deadline control and aim to meet deadlines even at the expense of specific requirements (Conte et al., 2001). Conversely, it aligns with the preferences of time-patient actors, who are less constrained by deadlines in structuring their work. Nevertheless, it is important to recognize that deadlines can also be treated as binding within plan-based work processes. In such cases, especially after midpoint transitions when deadlines become more immediate and pressing (Gersick, 1989), plan-based methods may stimulate deadline control and thus support time urgency to a greater extent.

In contrast, the Scrum method is unequivocally associated with time urgency. Its primary objective is to deliver customer value as rapidly as possible (Batra et al., 2011; Boehm & Turner, 2004). This goal is pursued even at the expense of the quality of the minimum viable product (Baskerville & Pries-Heje, 2004), reflecting the priorities of time-urgent actors who favor speed over quality (Mohammed & Harrison, 2013). Research indicates that Scrum generates continuous time pressure due to a series of short-term deadlines (Baskerville & Pries-Heje, 2004; Fitzgerald et al., 2006; O'Riordan et al., 2013; van Oorschot et al., 2018). Since Scrum explicitly disallows deadline postponement (Bianchi et al., 2020; Drury-Grogan, 2014; Karlstrom & Runeson, 2005), it aligns with the preferences of time-urgent actors who derive intrinsic satisfaction from working under time pressure and adhering to deadlines (Conte et al., 2001; Rastegary & Landy, 1993).

To conclude, this section demonstrated that plan-based methods and the agile Scrum method support different temporal depths and levels of time urgency. Due to the emphasis

on upfront planning, plan-based methods facilitate a more distant-future and -past time orientation. In contrast, the agile Scrum method promotes a near-future and -past time orientation, as its planning typically spans only a few weeks. Given the high priority placed on quality, plan-based methods generally support time patience rather than time urgency. However, they may also foster time urgency when deadlines are treated as fixed and non-negotiable. Conversely, the agile Scrum method clearly supports time urgency due to its binding and tightly scheduled deadlines.

Temporal patterns:

In accordance with the temporal framework of Ancona and colleagues (2001), temporal patterns can be described through several temporal mapping choices. These choices relate to the start and end points, durations, and pace of temporal units, such as activities, events, phases, or cycles, as well as to their sequence and linearity, and to their rhythms.

The central temporal units in plan-based methods are the sequential phases, each comprising a distinct set of activities (Boehm & Turner, 2004; Fernandez & Fernandez, 2008; Oellgaard, 2013; Petersen & Wohlin, 2010). Actors have the discretion to define the start and end points of these phases, and thus their duration (Project Management Institute, 2021). In many cases, a single phase may span several weeks or even months (Lindkvist et al., 1998). The chosen duration of each phase influences the overall pace or velocity of the work process. While the type and number of phases can vary across projects (Project Management Institute, 2021), plan-based methods typically require phases to be executed sequentially rather than concurrently. When determining the types and number of phases, actors must also decide on their sequence, taking into account dependencies between them (Oellgaard, 2013; Petersen & Wohlin, 2010). These processes are generally linear in structure, and the repetition of phases is not intended; however, returning to earlier phases is possible, but it is generally avoided due to the high costs involved (Karlstrom & Runeson, 2005; Repenning, 2001). As such, inherently recurring activities are rare, except for routine events like team or milestone meetings, which may occur once or multiple times within each phase of the plan-based work process (Hobbs & Petit, 2017; Ly et al., 2025).

In the Scrum method, the central temporal units are the recurring sprints. Like the phases in plan-based methods, sprints must have clearly defined start and end points. However, unlike plan-based phases, the Scrum method explicitly constrains sprint duration, specifying that each sprint must last between one and four weeks (Schwaber & Sutherland, 2020). This timeframe is widely regarded as short, particularly when compared to the extended duration of plan-based phases, which may span several months (Baskerville & Pries-Heje, 2004; Cooper, 2014; Lindkvist et al., 1998). Because sprints follow one another seamlessly (Schwaber & Sutherland, 2020), the temporal structure of Scrum is characterized by a fast and regular rhythm (Bianchi et al., 2020; Cooper & Sommer, 2016), which "defines the heartbeat of the [work process]" (Cooper, 2017, p. 50). Within each sprint, the Scrum team performs all necessary activities to deliver a minimum viable product. The team retains autonomy in deciding which activities to carry out and whether they should be performed sequentially or concurrently (Schwaber & Sutherland, 2020). The only prescribed sequence applies to the core Scrum events, which begin with sprint planning, include daily standup meetings, and conclude with sprint reviews and retrospectives (Schwaber & Sutherland, 2020).

Due to its clearly defined, rapid rhythms and the possibility of executing multiple activities concurrently (Cooper, 2014; Schwaber & Sutherland, 2020), Scrum is often perceived as enabling faster workflows than traditional plan-driven practices (Lindkvist et al., 1998). This perception is further reinforced by the absence of prolonged upfront planning phases in Scrum (Hobbs & Petit, 2017; Leybourne, 2009) and the early delivery of potentially usable outcomes to customers (Ceschi et al., 2005; Petersen & Wohlin, 2010). However, the short duration of individual sprints does not necessarily imply that an entire Scrum process is completed more quickly than its plan-based counterpart (Petersen & Wohlin, 2010). Research indicates that agile processes may be prolonged by the ongoing opportunity for customers to request changes at any stage (Bianchi et al., 2020; G. Lee & Xia, 2010). Additionally, the overall timeline can extend if the Scrum team intentionally limits the number of tasks undertaken per sprint (Karlstrom & Runeson, 2005). Such practices align with the Agile Manifesto, which states that developers "should be able to maintain a constant pace indefinitely" (Beck et al., 2001), without prescribing specific expectations regarding speed. Therefore, although Scrum is frequently perceived as faster than plan-based methods, it does not explicitly claim to accelerate the overall work process.

In summary, the temporal patterns of plan-based methods and the agile Scrum method differ fundamentally. Plan-based methods are organized into distinct phases, each comprising a specific set of activities with durations that can be freely defined. These phases are executed sequentially, typically without the intention of repeating the activities they encompass. In contrast, Scrum is structured around recurring sprint cycles of a maximum duration of four weeks, characterized by a rapid and regular rhythm. Within each sprint, team members may carry out the full range of activities necessary to develop a minimum viable product, either in sequence or concurrently. Owing to these distinctive temporal patterns, Scrum is often perceived as faster than plan-based methods; however, this perception does not apply universally to every work process.

This chapter illustrated divergent temporal structures, following the model of Ancona, Okhuysen, and colleagues (2001), using plan-based and Scrum methods as concrete examples. I selected plan-based methods to represent the work practices of non-agile units and the Scrum method to represent those of agile units. A comparison of their temporal structures reveals differences across temporal conceptions, patterns, and orientations. Notably, the use of plan-based methods does not fully align with my initial definition of non-agile units. I defined non-agile units as those that do not employ agile management frameworks, such as Scrum. While this clearly includes plan-based methods, non-agile units may also rely on other types of work practices. These may include ad-hoc practices (Harris et al., 2009), which are commonly observed in customer service and sales units (Mena & Parker, 2024). I have sporadically identified such practices in some of the non-agile units examined in this dissertation. Due to their comparatively lower relevance relative to plan-based practices, I chose not to conceptualize their temporal structures in detail. Nevertheless, I expect that the temporal structures of ad-hoc practices also differ from those of the Scrum method. For example, work processes in customer service units may be more firmly grounded in event time than those of Scrum. In their efforts to respond to customer requests as rapidly as possible, these units may operate in even shorter cycles, accommodating even higher levels of time urgency.

In this chapter, I developed a conceptual account of the temporal structures of agile and non-agile organizational units and of the differences between them, based on the characteristics of Scrum and plan-based methods. In the findings, I will assess the validity of this conceptualization using the empirical data I collected.

3. Coordinating in light of divergent temporal structures

Chapter 2 demonstrated that actors' temporal structures can differ along three dimensions: temporal conceptions, temporal patterns, and temporal orientations (Ancona, Okhuysen & Perlow, 2001). It also showed that collaboration between actors with divergent temporal structures is often marked by tensions. When the coordinating function of shared temporal structures is lost (Orlikowski & Yates, 2002; Zerubavel, 1981), actors with differing temporal structures often experience temporal tensions, which tend to manifest as coordination challenges and conflicts (Gibson et al., 2007; Mohammed et al., 2017; Mohammed & Harrison, 2013; Reinecke & Ansari, 2015; Stjerne et al., 2019). Such tensions can reduce actors' willingness to collaborate (Oborn & Barrett, 2021; Stjerne et al., 2019; van Berkel et al., 2016; Waller et al., 2001). This raises the question of how actors with divergent temporal structures navigate the tensions they experience to ensure effective collaboration. This chapter offers a synopsis of the extant research addressing this important question, while also highlighting its current limitations and research gaps.

Research on temporal structuring (Blagoev & Schreyögg, 2019; Geiger et al., 2021; Orlikowski & Yates, 2002; Pentland et al., 2025) and temporal work (Bansal et al., 2022; Kaplan & Orlikowski, 2013; Otto et al., 2024) provides valuable insights into how actors cope with divergent temporal structures. The notion of temporal structuring is grounded in Orlikowski and Yates' (2002) practice-based perspective on time, which posits that human action both shapes and is shaped by temporal structures. Research in this area explores how actors produce and reproduce temporal structures (Geiger et al., 2021; Pentland et al., 2025). Most of these studies focus on temporal coordination, namely, how actors draw upon specific temporal structures to effectively align their ongoing activities with those of others (Blagoev & Schreyögg, 2019; Orlikowski & Yates, 2002).

Temporal work is closely related to temporal structuring and is often used interchangeably. Kaplan and Orlikowski (2013) were the first to introduce the term temporal work, defining it as the process by which actors reconcile differing

interpretations of the past, present, and future in order to develop viable strategic projects. Building on this foundational definition, subsequent research has employed this term to examine how actors navigate differences in temporal conceptions, temporal orientations, and temporal patterns (Kent & Granqvist, 2025; Otto et al., 2024; Reinecke & Ansari, 2015). Accordingly, I adopt the broader definition proposed by Bansal and colleagues (2022, p. 7), who define temporal work as "any individual, collective, or organizational effort to influence, sustain, or redirect the temporal [conceptions, orientations] [2], or patterns that shape strategic action." In contrast to temporal structuring, temporal work places less emphasis on the coordination of interdependent actors. Instead, it encompasses a broader spectrum of temporal tensions, some of which involve coordination among multiple actors, while others pertain solely to a single focal actor. For instance, Otto and colleagues (2024) illustrate how researchers in a pharmaceutical company deliberately disengaged from managerially imposed schedules to reconsider their fundamental hypotheses, thereby addressing the temporal tension of not achieving desired results within a predefined time frame. Although this disengagement resolved a tension experienced primarily by the researchers, it nevertheless gave rise to coordination challenges with managerial actors.

Research on both temporal structuring and temporal work points to two fundamental approaches that actors can choose to cope with divergent temporal structures. First, actors may synchronize their disparate temporal structures (Otto et al., 2024; Shipp & Richardson, 2021). Synchronization helps establish shared temporal structures, thereby facilitating coordination among actors (McGrath, 1991; McGrath & Rotchford, 1983). This includes aligning actors' temporal structures with those of a dominant Zeitgeber, a process commonly referred to as entrainment (Ancona & Chong, 1996; McGrath & Kelly, 1986; Pérez-Nordtvedt et al., 2008). In the language of temporal structuring, actors reproduce the temporal structure of a dominant Zeitgeber; in terms of temporal work, they influence or redirect their own temporal structures to align with it. However, synchronization also encompasses situations in which actors develop novel temporal patterns to which they subsequently align in order to enable effective coordination

[2] Instead of temporal conceptions and temporal orientations, Bansal and colleagues (2022, p. 7) originally refer to "temporal assumptions." As chapter 2, in particular with reference to Rowell and colleagues (2016), has shown, temporal assumptions are the temporal conceptions and orientations that underlie temporal patterns. For ease of understanding, I decided to use consistent wording.

(Granqvist & Gustafsson, 2016; Lifshitz-Assaf et al., 2021; Otto et al., 2024). From the perspective of temporal structuring, this reflects the production of new temporal patterns enacted by interdependent actors. In terms of temporal work, it involves influencing or redirecting one's temporal structures toward emergent, co-created temporal patterns.

Second, actors may maintain their distinct temporal structures, resulting in the coexistence of multiple divergent temporal structures (Blagoev & Schreyögg, 2019; Otto et al., 2024; Shipp & Richardson, 2021). From a temporal structuring perspective, actors reproduce their unique yet distinct temporal structures. Temporal work captures this coexistence by accounting for how actors sustain their own temporal structures over time (Bansal et al., 2022). Both temporal structuring and temporal work thus acknowledge that divergent temporal structures can coexist and offer a foundation for analyzing the practices that support such coexistence. This chapter focuses specifically on practices through which actors actively seek to resolve or mitigate the coordination tensions arising from their divergent temporal structures. These include the practices of isolating, brokering, and negotiating. Isolating entails severing interdependencies with actors who operate under different temporal structures, thereby minimizing the need for coordination and avoiding associated tensions (Dille et al., 2023; Hilbolling et al., 2022; Oborn & Barrett, 2021). Brokering involves establishing bridging mechanisms that facilitate coordination between actors with divergent temporal structures (Mell et al., 2021). Negotiating fosters temporal reflexivity through dialogue and contestation, enabling actors to surface and reconcile their temporal differences, thus creating the conditions for ambitemporality (Magrelli et al., 2022; Reinecke & Ansari, 2015). While acknowledging that temporal tensions may also be endured (Otto et al., 2024) or coordination can occur in serendipitous moments (Hilbolling et al., 2022; Pentland et al., 2025), this chapter does not include such passive practices, as they do not constitute intentional strategies undertaken by actors to address the tensions arising from temporal divergence.

In this chapter, I elaborate on entraining and developing new temporal patterns as two practices aimed at synchronizing divergent temporal structures, and on isolating, brokering, and negotiating as three practices to sustain them. The chapter focuses on the extent to which each of these practices facilitates effective coordination and collaboration among actors whose temporal structures differ. In addition, it highlights the key limitations and research gaps identified with regard to these practices. Table 5 provides

an overview of these practices, their respective effects on coordination and collaboration, and the key limitations and gaps in the existing literature.

3.1. Synchronizing practices

Through the process of synchronization, actors establish shared temporal patterns that serve as a common foundation for effective coordination and collaboration (Orlikowski & Yates, 2002; Zerubavel, 1981). Massey and colleagues (2003, p. 91) define synchronization as "aligning the pace of effort within and between [actors]", while Im and colleagues (2005, p. 91) describe it as a state in which "all [actors] engage in the same activity at the same time." Blagoev and Schreyögg (2019, p. 1820) further define synchronization as "the convergence of two principally independent rhythms in terms of phase or pace." Despite referring to different aspects of temporal patterns, these definitions share a common emphasis on alignment. Building on this commonality, I define synchronization as the alignment of all or selected aspects of temporal patterns, including the start and end points of activities or cycles, their durations, the sequence in which they unfold, the pace of progression, and the rhythms at which activities or cycles recur. Importantly, while divergent temporal patterns can be aligned through synchronization, it is considerably more difficult, if not impossible, to synchronize divergent temporal conceptions and orientations as these dimensions represent deep-level assumptions and cognitive traits that can hardly be changed (Bluedorn, 2002; Bluedorn & Denhardt, 1988; Conte et al., 1995; Shipp et al., 2009).

By enacting synchronized temporal patterns, actors can rely on shared timing, sequencing, and pacing of their interdependent activities, thereby reducing perceived uncertainty (Moore, 1963). When actors know and agree on when to act, how quickly to proceed, and in what sequence to perform their tasks, coordination with interdependent actors is facilitated. Notably, this coordinating effect can manifest even when actors hold different temporal conceptions and orientations. For example, Mohammed and Nadkarni (2011) demonstrate that leaders can facilitate coordination within teams exhibiting varying levels of time urgency by synchronizing their temporal patterns.

Table 5: Summary of practices to cope with divergent temporal structures

Practice:	Description:	Coordinative effects:	Limitations and gaps of current research:
		Synchronizing practices	
entraining	synchronizing with the prevailing temporal structures of a shared Zeitgeber (Ancona & Chong, 1996; Pérez-Nordtvedt et al., 2008)	(re-)establishing shared temporal patterns that facilitate coordination	▪ What are the costs of entrainment for actors who give up their temporal patterns? ▪ Which agency do non-time-setting actors have in entraining processes? ▪ How do the dynamics of entrainment unfold, and what factors influence transitions into, out of, and between states of entrainment?
developing new temporal patterns	synchronizing with newly developed temporal structures that reflect the disparate temporal structures of all actors (Granqvist & Gustafsson, 2016; Lifshitz-Assaf et al., 2021)	(re-)establishing shared temporal patterns that facilitate coordination	▪ How do actors develop new temporal patterns that address the tensions of a collaborative situation? ▪ How can resistance to new temporal patterns be overcome or prevented? ▪ Which contingencies shape the dynamic adaptation of new temporal patterns?
		Sustaining practices	
isolating	cutting interdependencies between actors with divergent temporal structures (Oborn & Barrett, 2021; Dille et al., 2023; Hilbolling et al., 2022)	reducing the need for coordination	▪ Which conditions explain the different outcomes of isolating practices? ▪ How can actors shape, navigate, and transit among the dynamic episodes of isolating and collaborating?
brokering	establishing bridging links between otherwise isolated actors with divergent temporal structures (Mell et al., 2021)	increasing coordination capacities by providing brokering roles	▪ Under which conditions are brokering practices effective in mitigating temporal tensions? ▪ Are these practices applicable to other dimensions of temporal structures? ▪ Which different forms do they take to coordinate across various dimensions of temporal differences?
negotiating	engaging in dialogue and contestation fosters temporal reflexivity among actors with divergent temporal structures and motivates to develop ambitemporal patterns (Reinecke & Ansari, 2015; Magrelli et al., 2022)	(re-)establishing shared temporal patterns that accommodate different temporal conceptions and orientations and that facilitate coordination	▪ How does temporal reflexivity arise from dialogue and contestation between actors? ▪ How do ambitemporal patterns arise from temporal reflexivity? ▪ Which factors support or hinder these processes?

Synchronization represents the most common and well-established approach to achieving coordination amid divergent temporal structures. Its significance is emphasized in various temporal theories and concepts. For example, the time, interaction, and performance (TIP) theory (McGrath, 1991) identifies synchronization as a primary solution to temporal coordination challenges, particularly those arising from conflicting temporal interests and requirements. Likewise, Zerubavel (1981) presents synchronization as one of the two fundamental patterns of temporal coordination. More recent studies continue to affirm its role in facilitating coordination among actors with divergent temporal structures (Dille et al., 2023; Geiger et al., 2021; Karaduman, 2021). Synchronization underpins two key practices that actors employ to cope with divergent temporal structures: (1) entraining, and (2) developing new temporal patterns. The following chapter elaborates on these two practices in greater detail.

3.1.1. Entraining

Entrainment refers to "the adjustment of the pace or cycle of one activity to that of another" (Ancona & Chong, 1996, p. 253). Through entrainment, actors align their temporal patterns with those of a dominant Zeitgeber, often relinquishing their own unique temporal patterns (Ancona & Chong, 1996; Bluedorn, 2002; Khavul et al., 2010; Pérez-Nordtvedt et al., 2008). This process results in the creation of shared temporal patterns between the focal actor and the dominant Zeitgeber, thereby facilitating coordination and collaboration (Orlikowski & Yates, 2002). Examples for dominant Zeitgebers include seasonal demand cycles of key customers, product innovation rates of major competitors, or the governmental fiscal year (Dibrell et al., 2015; Khavul et al., 2010; Pérez-Nordtvedt et al., 2008). However, Zeitgebers can also be internal to an organization and include the rhythms of internal planning cycles, work schedules, or dominant organizational routines (Hopp & Greene, 2018; Pérez-Nordtvedt et al., 2008).

Entrainment primarily involves aligning actors' temporal patterns. In this sense, Ancona and Chong (1996) distinguish between two forms of entrainment: tempo entrainment and phase entrainment. Tempo entrainment occurs when actors adjust their pace to match that of a dominant Zeitgeber. Depending on the dominant Zeitgeber's pace, actors either accelerate or decelerate their own activities, thereby reducing or preventing tempo misfits

(Pérez-Nordtvedt et al., 2008). Phase entrainment involves adjusting the timing or sequencing of activities (Ancona & Chong, 1996). Actors modify the start, end, or sequence of their activities to align with the dominant Zeitgeber's temporal patterns, thereby reducing or preventing phase misfits (Pérez-Nordtvedt et al., 2008).

The positive effects of entrainment on performance have received considerable scholarly attention. At the organizational level, prior research has demonstrated the beneficial impact of temporal fit on organizational performance (Dibrell et al., 2015; Hopp & Greene, 2018; Khavul et al., 2010; Pérez-Nordtvedt et al., 2008; Shi & Prescott, 2012), which can be attributed to improved temporal coordination between collaborating actors (Ancona & Chong, 1996; McGrath & Kelly, 1986; Shi & Prescott, 2012). Shared temporal patterns foster predictability concerning the timing and pacing of actors' activities (Khavul et al., 2010). For example, when actors entrain to the temporal patterns of their customers, they can more accurately anticipate optimal delivery times, thereby better meeting customer demand (Pérez-Nordtvedt et al., 2008). Synchronizing supply with demand is likely to enhance sales, reflecting improved organizational performance. Similarly, entraining to internal Zeitgebers, such as planning schedules, can increase predictability regarding the timing of both resource allocation and demand, thereby helping to avoid costly inefficiencies, including delays and idle time (Shi & Prescott, 2012).

Research emphasizes the benefits of entrainment, yet it largely neglects that it may also entail drawbacks. To date, only a few studies have acknowledged that entrainment may incur costs (Blagoev & Schreyögg, 2019; Zhang et al., 2023). These studies have conceptualized the costs of entrainment exclusively in terms of its restriction to coordination with a single Zeitgeber, as entraining with one Zeitgeber necessarily requires decoupling from others (Blagoev & Schreyögg, 2019). Thus, entrainment provides limited solutions for coordination in temporally complex situations that are characterized by the presence of multiple, potentially conflicting, Zeitgebers (Blagoev & Schreyögg, 2025; Hilbolling et al., 2022). Thus, existing research provides only a narrow perspective on the potential drawbacks of entrainment, limiting our current understanding of why actors in entrainment processes may resist adopting the temporal patterns of a Zeitgeber.

Alongside its (positive) effects, a substantial body of research on entrainment focuses on how entrainment is achieved. In the natural sciences, where the concept originates,

entrainment is typically viewed as an automatic process (Ancona & Chong, 1996; McGrath & Kelly, 1986). For example, two pendulum clocks placed side by side eventually synchronize after one is disturbed, and human circadian rhythms tend to adjust automatically when crossing time zones. However, such automaticity is rarely observed in organizational contexts. Instead, organizational research shows that entrainment is a deliberate and effortful endeavor (Dille et al., 2023; Kunzl & Messner, 2023). Scholars have identified three key mechanisms through which time-setting actors induce entrainment: (1) providing information, (2) setting goals, and (3) directing behavior.

First, numerous studies underscore that time-setting actors provide information to motivate other actors to entrain to their temporal patterns. These studies often refer to the role of (temporal) objects in conveying information and thereby shaping actors' temporal patterns (Bansal et al., 2022; Blagoev et al., 2018; Lupu & Rokka, 2022; Schultz & Hernes, 2013; Yakura, 2002). In particular, temporal boundary objects, such as deadlines (Gersick, 1989; Labianca et al., 2005), schedules (Chang et al., 2013; Yakura, 2002), or project plans (Tukiainen & Granqvist, 2016), play a crucial role. These are defined as "physical artefacts or abstract concepts that serve as a common reference point across disparate temporal orders, structures, or orientations" (Blagoev et al., 2024 Appendix I). Like conventional boundary objects (Carlile, 2002; Star, 2010; Star & Griesemer, 1989), temporal boundary objects facilitate interpretive processes that enable actors to align their temporal understandings (Star, 2010). By visualizing the temporal patterns of the dominant Zeitgeber (Yakura, 2002), such objects support actors in self-regulating their temporal behavior to achieve entrainment (Kunzl & Messner, 2023). However, for this mechanism to be effective, the information provided must be correctly interpreted and understood. Dille and colleagues (2023), for instance, found that project plans led to entrainment only after they were "translated" into work practices and languages familiar to the actors involved.

Second, time-setting actors have been shown to promote entrainment by linking their temporal structures to desirable goals that actors are motivated to achieve. What constitutes a desirable goal varies depending on the actors and context. Examples in the literature range from sustaining organizational competitiveness (Hilkamo et al., 2021) to preserving a natural wonder (Crawford et al., 2022). Such goals may be set by actors with or without formal authority and are often articulated through narratives (Alimadadi et al.,

2022; Bartel & Garud, 2009; Crawford et al., 2022; Hilkamo et al., 2021; Rindova & Martins, 2022) or imaginations of the future (Hilbolling et al., 2022). For example, Hilbolling and colleagues (2022) describe how a protovision of an interorganizational project's future outcome helped synchronize the timing, sequencing, and pacing of actors' interdependent activities. The importance of goal alignment is further illustrated by McGivern and colleagues (2018), who show that management consultants failed to secure sustainable entrainment when they did not connect their desired temporal patterns to goals that were meaningful for their clients. As a result, clients eventually reverted to their original temporal patterns.

Third, time-setting actors often exercise directive authority to induce entrainment. Otto and colleagues (2024, p. 1808) use the term "imposed entrainment" to illustrate how formal authority can enforce specific temporal patterns. Formal authority is associated with the right to reward and sanction, which facilitates actors' entrainment to the temporal patterns of a dominant Zeitgeber (Klein et al., 2007; Mohammed & Nadkarni, 2011; Okhuysen & Bechky, 2009). In addition, actors who possess formal authority can impose specific temporal patterns by exerting control over others' resources. For example, by skillfully allocating temporal resources, actors with formal authority can influence the timing, sequencing, and pacing of activities, thereby mandating other actors' compliance with their desired temporal patterns (Maruping et al., 2015; Schriber & Gutek, 1987).

This review of the existing literature on entrainment processes reveals that, in their efforts to explain the mechanisms through which time-setting actors induce entrainment, studies have largely overlooked the role of non-time-setting actors. Shipp and Richardson (2021) offer an initial contribution to addressing this gap by proposing four contingencies that account for why actors entrain to a Zeitgeber. Actors are likely to entrain to a Zeitgeber with whom it has high outcome interdependencies, who holds greater power in their dyadic relationship, when information uncertainty is low, and when future interactions are anticipated. However, with only these four relational contingencies considered, the agentic decision-making of non-time-setting actors remains largely unexplored. Research has yet to pay more attention to the various contingencies and drivers that shape the decision-making of non-time-setting actors in entrainment processes.

A final remark concerns the potential dynamics of entrainment. Early research on entrainment portrayed entrained temporal patterns as largely static and persistent (Ancona

& Chong, 1996; McGrath & Kelly, 1986). McGrath and Kelly (1986) noted that once actors have aligned with the temporal patterns of a dominant Zeitgeber, these temporal patterns tend to persist even after the Zeitgeber is removed. Several case studies support this notion of persistence. For example, Perlow and colleagues (2002) show how an internet start-up, having adapted to accelerated temporal rhythms in pursuit of competitiveness, found itself in a speed trap from which it could no longer escape, even as the accelerated rhythms grew increasingly detrimental. Similarly, Blagoev and Schreyögg (2019) demonstrate that management consultancies, once entrained to extreme working hours imposed by clients, struggled to detrain even when these extreme working hours revealed severe negative consequences. These examples suggest that entrainment can give rise to rigid temporal patterns that actors are unable to adapt in response to temporal complexity (Blagoev & Schreyögg, 2024).

However, research has also begun to highlight the dynamic potential of entrainment. Ancona and Waller (2007), for example, describe how actors engage in a dance of entrainment, dynamically entraining, detraining, and re-entraining over the course of their work. Otto and colleagues (2024) show how actors deliberately detrain following initial episodes of entrainment. Shipp and Richardson (2021, p. 313) conceptualize entrainment as "a matter of degree", emphasizing that entrainment can be both sustained and unsustained. While the former aligns with the view of entrainment as persistent, the latter suggests that it may represent a temporary coordination mechanism, ultimately replaced by alternative strategies for managing divergent temporal structures (Otto et al., 2024; Shipp & Richardson, 2021). In light of these contrasting perspectives, we still lack understanding of how the dynamics of entrainment unfold, including the events, processes, tensions, practices, or conditions that prompt transitions to and from entrainment.

In summary, by entraining, actors synchronize their divergent temporal patterns with a dominant Zeitgeber, thereby establishing shared temporal patterns that enable effective coordination and collaboration. Despite its established role in temporality research, entrainment continues to raise important unanswered questions. This chapter has sought to highlight several of these gaps: What are the costs associated with entrainment, for example, for non-time-setting actors who relinquish their unique temporal patterns? Which agency do non-time-setting actors have in entraining processes? Which are the

drivers that motivate them to entrain or resist entrainment? How do the dynamics of entrainment unfold, and what factors influence transitions into, out of, and between states of entrainment?

3.1.2. Developing new temporal patterns

A second synchronizing practice involves the creation of and alignment with new temporal patterns. While extant literature predominantly refers to new temporal structures, it is important to emphasize that this practice concerns surface-level temporal patterns rather than deep-level temporal orientations and temporal conceptions[3]. Unlike entrainment, this approach does not assume that Zeitgebers are immutable (Granqvist & Gustafsson, 2016). Instead, in line with an agentic perspective on time (Emirbayer & Mische, 1998; Orlikowski & Yates, 2002), it suggests that actors can actively reshape Zeitgebers, thereby developing new temporal patterns with which they synchronize (Granqvist & Gustafsson, 2016; Jarvenpaa & Välikangas, 2022; Lifshitz-Assaf et al., 2021; Otto et al., 2024). Because new temporal patterns similarly depend on synchronization, their implications for coordination and collaboration mirror those of entrainment. They foster shared temporal patterns that facilitate interactions among actors with initially divergent temporal structures (Orlikowski & Yates, 2002; Zerubavel, 1981).

New temporal patterns refer to any temporal patterns that are temporal innovations that have not been previously used by all actors involved (Feldman et al., 2022; Lifshitz-Assaf et al., 2021; Otto et al., 2024). However, this also means that they can be imported from other organizational contexts (Granqvist & Gustafsson, 2016; Jarvenpaa & Välikangas, 2022; Lifshitz-Assaf et al., 2021). For example, Granqvist and Gustafsson (2016) illustrate how actors collaborating within a university context integrated the accelerated rhythms of a governmental reform into their own transformation initiative. Similarly, in Jarvenpaa and Välikangas' (2022) study, members of an interorganizational collaboration

[3] This is in line with the fundamental idea behind the synchronization approach. Any synchronization, either with an immutable Zeitgeber or with newly created temporal structures, aligns the surface-level temporal patterns of actors while their deep-level temporal orientations may continue to differ.

decided to adopt the sprint-based work rhythms they had encountered during a workshop on agile methods.

Actors may choose to develop new temporal patterns rather than entraining to existing ones for a variety of reasons. In some instances, creating new temporal patterns becomes necessary when previously relied-upon temporal patterns are disrupted (Feldman et al., 2022). In other cases, actors may assess that new temporal patterns are better suited to task performance. In doing so, they respond to tensions that either originate from established temporal patterns (Jarvenpaa & Välikangas, 2022; Otto et al., 2024) or are insufficiently addressed by them (Granqvist & Gustafsson, 2016; Lifshitz-Assaf et al., 2021). For example, Jarvenpaa and Välikangas (2022) demonstrate how actors in an interorganizational collaboration recognized that their prevailing temporal patterns resulted in disproportionately long planning phases at the expense of operational work. In response to this tension, they introduced sprint-based temporal patterns that allowed for a greater focus on execution. Granqvist and Gustafsson (2016) describe a case in which existing temporal patterns were perceived as too slow to support timely transformation. Facing the need for rapid change, actors devised new temporal patterns characterized by an accelerated pace. Finally, actors may opt to create new temporal patterns when operating in environments lacking a clear Zeitgeber, or when the Zeitgeber fails to provide sufficiently specific temporal guidance (Shipp & Richardson, 2021).

Research has started to inquire into the performance outcomes of new temporal structures. For example, Otto and colleagues (2024) show how new temporal patterns enabled research teams in a pharmaceutical company to capitalize on innovation opportunities that would have remained untapped under formerly imposed managerial temporal patterns. In relation to this dissertation's central question, Lifshitz-Assaf and colleagues (2021) demonstrate that, consistent with the notion of shared temporal patterns (Orlikowski & Yates, 2002; Zerubavel, 1981), newly developed temporal patterns can enhance coordination. They distinguish between temporal patterns imported from other contexts and those developed from scratch. In their study of hackathon-based product development, imported temporal patterns offered a full basis for coordination from the beginning, whereas those developed from scratch initially offered only a minimal foundation for coordination. Interestingly, their findings show that the full basis for coordination did not necessarily lead to superior outcomes. Teams relying on imported

temporal patterns failed to complete a prototype within the hackathon timeframe, while those that developed new temporal patterns from scratch at least delivered prototypes with basic functionality. Lifshitz-Assaf and colleagues (2021) attributed the inferior performance of the imported temporal patterns to their limited suitability for addressing the high temporal ambiguity of the hackathon, while temporal patterns that teams jointly developed during the hackathon better reflected the situational needs. Their findings imply that the processes by which new temporal patterns are created shape their capacity to address the complexities and tensions of collaborative situations.

Granqvist and Gustafsson (2016) offer a seminal contribution to the question of how actors develop new temporal patterns. They conceptualize their development as an institutional change process in which actors actively construct a sense of urgency for adopting new temporal patterns, entrain others to them, and enact momentum until they become irreversibly embedded. Notably, Granqvist and Gustafsson (2016), along with other authors (Feldman et al., 2022; Jarvenpaa & Välikangas, 2022; McGivern et al., 2018), describe the development of new temporal patterns as a unilateral process in which temporal patterns are enforced by a single actor. Similar to the entrainment literature, research on the development of new temporal patterns has adopted a narrow focus on the actions of time-setting actors, thereby neglecting the agency of those who are expected to adopt new temporal patterns. Yet, new temporal patterns often fail precisely because of the resistance of such actors (Ancona & Waller, 2007; McGivern et al., 2018; Perlow, 1999). Thus, it is important to place greater emphasis on the exploration of their agentic role and developing and synchronizing with new temporal patterns. This connects to the broader gap in understanding how and through which (collaborative) processes temporal patterns are developed that reflect and address the specific tensions of collaborative situations.

Finally, as with entrainment, new temporal patterns do not offer permanent solutions for avoiding temporal tensions or ensuring sustained coordination and collaboration among actors with divergent temporal structures (Otto et al., 2024). Recent studies recognize that new temporal patterns are provisional arrangements that require ongoing development and revision (Lifshitz-Assaf et al., 2021; Otto et al., 2024). This prompts the need to explore the specific conditions that trigger and shape the dynamic adaptation of newly developed temporal patterns.

To conclude, developing new temporal patterns constitutes a second synchronizing practice that both addresses temporal tensions and facilitates coordination and collaboration by establishing shared temporal patterns. In contrast to entraining, it does not presuppose the existence of an immutable Zeitgeber but instead highlights actors' agency in shaping and influencing their own temporal patterns. This chapter has sought to identify gaps in the current literature, including the following: How and through which processes do actors develop new temporal patterns that adequately address the tensions of a collaborative situation? How can resistance to new temporal patterns be overcome or prevented? Which contingencies shape the dynamic adaptation of new temporal patterns?

3.2. Sustaining divergent temporal structures

The synchronization approach outlined in Chapter 3.1 is not universally viable for managing disparate temporal structures across all collaborative settings. First, synchronization is often only partial. In multi-party collaborations, synchronizing with the temporal patterns of one actor frequently entails a detachment from those of others (Hilbolling et al., 2022; Kunisch et al., 2021; Nowotny, 1992; Ofori-Dankwa & Julian, 2001; Zhang et al., 2023). This is illustrated by Blagoev and Schreyögg's (2019) study on the persistence of extreme working hours. They show that management consultancies' synchronization with the extended working hours preferred by their clients inevitably led to a decoupling from broader societal rhythms of work and leisure. As a result, because focal actors are only able to synchronize their temporal patterns with a limited number of other parties, they must rely on alternative practices to coordinate and collaborate with those actors whose temporal patterns remain misaligned.

Second, synchronizing practices are limited to specific dimensions and elements of temporal structures. As outlined in Chapter 3.1, synchronization facilitates the alignment of temporal patterns, while underlying temporal conceptions and orientations often remain unaffected and continue to diverge (McGivern et al., 2018; Orlikowski & Yates, 2002). Moreover, even within temporal patterns, certain elements are not easily amenable to synchronization or are rarely addressed through such practices. A salient example is the collaboration of actors across different time zones, as commonly observed in globally dispersed organizations (Massey et al., 2003; Mell et al., 2021; Montoya-Weiss et al.,

2001). In these settings, asynchrony tends to dominate due to the logistical challenges and costs of synchronization. Consequently, actors often begin and end their workdays at different times (Espinosa & Carmel, 2003), with little or no temporal overlap (Mell et al., 2021), and must therefore rely on practices other than synchronization to ensure effective coordination and collaboration.

Third, synchronizing practices are not always optimal in every situation. In some contexts, it may be more valuable for actors to retain their distinct temporal structures rather than abandon them in favor of synchronization (Jarvenpaa & Välikangas, 2022). Such contexts include the work of firefighters (Geiger et al., 2021), newsmaking organizations (Patriotta & Gruber, 2015), SWAT teams (Bechky & Okhuysen, 2011), and emergency response units (Berthod et al., 2017). What these cases have in common is that they operate under conditions of high temporal uncertainty. Under these conditions, it becomes critical for actors to preserve their own temporal structures to maintain control over the timing, sequencing, and pacing of their routines and activities. Temporal autonomy allows for reliable task execution without becoming overwhelmed by the temporal demands of the surrounding context (Blagoev & Schreyögg, 2019; Geiger et al., 2021; Rowell et al., 2016). These observations raise important questions about the practices that enable actors to sustain their unique temporal structures and effectively "march to the beat of their own drums" (Shipp & Richardson, 2021, p. 299), while also navigating the tensions that emerge in collaborative situations.

Acknowledging that synchronizing practices are not always applicable or effective, recent research has begun to explore how divergent temporal structures can coexist, employing a variety of conceptual labels. Otto and colleagues (2024) have introduced the notion of detrainment to describe the deliberate decision to maintain distinct temporal patterns, while Shipp and Richardson (2021) refer to this phenomenon as resistance to entrainment. More recently, Blagoev and Schreyögg (2025, p. 6) have conceptualized Eigenzeit as "the specific temporal mode through which a given organization selects the rhythms of its activities." Although Eigenzeit may include entrained modes, it also captures how actors consciously adopt temporal structures that diverge from those of their environment. Relatedly, the concepts of ambitemporality (Reinecke & Ansari, 2015) and temporal

ambidexterity[4] refer to actors' ability to accommodate contradictory temporal structures, with a particular emphasis on temporal conceptions and temporal orientations (Cuganesan, 2022; Fayard, 2024; Reinecke & Ansari, 2015; Slawinski & Bansal, 2012; S. L. Wang et al., 2019). Together, these concepts shed light on the practices actors may adopt to navigate the tensions that arise from sustaining divergent temporal structures. Drawing on this literature, I identify three overarching practices: (1) isolating, (2) brokering, and (3) negotiating. In the following sections, I elaborate on each of these practices, with a particular focus on their capacity to support coordination and collaboration across divergent temporal structures.

3.2.1. Isolating

Sustaining divergent temporal structures while mitigating potential tensions can be accomplished through isolating practices, which involve deliberately disconnecting actors with divergent temporal structures from one another. These practices build on the notion of temporal differentiation, which posits that different temporal structures are enacted in distinct organizations, organizational units, or practices (Ancona, Okhuysen, et al., 2001; Blagoev & Schreyögg, 2025; Luhmann, 1995). The logic of isolating divergent temporal structures parallels that of structural ambidexterity, which holds that exploration and exploitation can be pursued simultaneously when assigned to separate organizational units (Tushman & O'Reilly, 1996). In both cases, the primary benefit of isolation lies in its capacity to prevent direct clashes between incompatible logics, be they strategic or temporal, by reducing interdependence (March & Simon, 1958). I intentionally use the term isolating rather than separating to emphasize that actors with

[4] Temporal ambidexterity as defined by S. L. Wang and colleagues (2019) must not be confused with the classic notion of temporal ambidexterity as proposed by Tushman and O'Reilly (1996). Tushman and O'Reilly (1996) conceptualize temporal ambidexterity as a phased process of alternating between exploration and exploitation activities. In contrast, S. L. Wang and colleagues (2019, p. 789) define temporal ambidexterity as "the organizational capability to demonstrate simultaneous and strong commitment to actions with short-term and long-term organizational implications." S. L. Wang and colleagues (2019) contend that both concepts are fundamentally different since temporal ambidexterity, according to their understanding, cannot be achieved by temporally separating different activities, given that organizations are continuously subject to long-term and short-term demands.

divergent temporal structures do not merely operate in different teams, units, or organizations, but actively suspend or minimize their mutual dependencies.

Extant research shows that isolating practices occur across multiple levels of analysis. Actors with divergent temporal structures have been found to isolate from team members (Mohammed & Nadkarni, 2011), from other organizational units (Oborn & Barrett, 2021), from other organizations (Dille et al., 2023; Hilbolling et al., 2022), and even from their broader environment (Geiger et al., 2021). These practices are particularly prevalent in contexts marked by both divergent temporal patterns and orientations. At the team level, Mohammed and Nadkarni (2011) illustrate how managers isolate team members with incompatible temporal orientations by allocating them to different tasks. In an organizational context, Oborn and Barrett (2021) examine collaboration between medical professions and show that surgeons, who adopt a fast-paced activity rhythm due to a strong present focus, deliberately isolate from oncologists, whose future orientation prompts a slower work pace. To maintain their tempo, surgeons reduced interdependencies by foregoing scheduled consultations with oncologists. At the interorganizational level, Dille and colleagues (2023) and Hilbolling and colleagues (2022) illustrate isolating practices in interorganizational collaborations, involving organizations with distinct temporal structures. Dille and colleagues (2023), in their study of emergency call system implementation across Norway's police, fire, and health sectors, observe that the health sector, operating at a significantly slower pace than the police and fire sectors, isolated itself from its project partners. Similarly, Hilbolling and colleagues (2022) describe how organizations whose temporal structures conflicted with an interorganizational project's objectives were temporarily excluded from collaboration.

The consequences of isolating practices are mixed. On the one hand, they preserve disparate temporal structures while avoiding tensions between actors (March & Simon, 1958). For example, members of the health sector in Dille and colleagues' (2023) study reflected that they were only able to complete the project on time because, by isolating themselves from their project partners, they preserved their own temporal structures and avoided the conflicts and debates that arose from temporal differences. In Mohammed and Nadkarni's (2011) study, isolating team members with divergent temporal orientations (i.e., different levels of time urgency) not only avoided tensions but also leveraged the benefits of temporal diversity. Team leaders were able to enhance task

performance by assigning different tasks to team members based on which temporal orientation was better suited to the task at hand. On the other hand, the limitations of isolating practices are evident. They may avoid conflict by masking temporal differences (Reinecke & Ansari, 2015), but they can still significantly hinder coordination and collaboration between actors. This is illustrated in Oborn and Barrett's (2021) study, in which surgeons deliberately abstained from collaboration with oncologists to preserve their fast-paced rhythms. However, this decision to isolate, and thus limit collaboration, had adverse consequences. Oborn and Barrett (2021) observed that isolating practices ultimately reduced the information available to patients, making it more difficult for them to make informed decisions regarding medical treatment. Similarly, Hilbolling and colleagues (2022) reported the detrimental effects of isolating practices. In the interorganizational project they examined, isolation led to prolonged periods in which collaboration was impossible, and as a result, the project stagnated.

Given these mixed effects, research has yet to explore the conditions under which isolating practices are effective. Findings from Mohammed and Nadkarni's (2011) study on temporal leadership suggest that isolating practices can be effective when a temporal leader bridges between actors with divergent temporal structures. This implies that the effectiveness of isolating practices may depend on the supplementary use of brokering practices. In addition, current research does not explain why isolating practices sometimes accelerate actors' activities and enable them to complete work on time (Dille et al., 2023), while in other cases they result in stagnation (Hilbolling et al., 2022). To elucidate these inconsistencies, future studies could investigate varying degrees of interdependence among actors, both at a given point in time and as they evolve over the course of collaboration. Moreover, similar to Shipp and Richardson's (2021) conceptualization of entrainment, researchers could begin to frame isolation as a matter of degree, rather than a binary choice between complete isolation and full collaboration. Such a perspective could help explain the diverse consequences of isolating practices.

A final remark pertains to the dynamics of isolating practices. Hilbolling and colleagues (2022) show that decisions to isolate from actors with divergent temporal structures are episodic rather than permanent. They observe that such actors isolate from one another until serendipitous moments of alignment emerge that allow collaboration to resume. The notion of dynamic decoupling and recoupling is also reflected in Blagoev and

Schreyögg's (2025) conceptualization of a pluritemporal mode of Eigenzeit, which is regarded as a particularly effective approach for navigating the multifaceted temporal demands of various Zeitgebers. Thus, existing research suggests that dynamic shifts between isolation and collaboration can be advantageous, enabling actors to preserve distinct temporal structures while resolving tensions around coordination and collaboration. However, we still lack insight into the agentic roles of actors within these dynamics, particularly during transitions between periods of isolation and collaboration. Hilbolling and colleagues (2022) attribute the timing of these episodes to serendipity. Yet, if actors wish to foster more effective collaboration, it is crucial to understand how they identify or deliberately create opportune moments for isolation and collaboration, and how they can design and manage transitions between the two modes without introducing new tensions.

In sum, isolating practices reduce tensions by separating actors with disparate temporal structures. While this makes them useful in contexts where coordination needs can be minimized, it limits their applicability in situations where actors have strong interdependencies. Nevertheless, a dynamic interplay between episodes of isolation and collaboration may still support effective coordination among actors with divergent temporal structures. This chapter also highlighted several unanswered questions regarding isolating practices: What currently unidentified conditions account for their varied outcomes? How can actors intentionally shape and navigate the dynamic transitions between isolating and collaborating?

3.2.2. Brokering

Brokering practices can be employed in contexts where actors operate within divergent temporal structures and remain organizationally or functionally separated. The concept of brokering involves creating bridging ties between such actors, for instance, through the assignment of intermediary roles (Mell et al., 2021). Grounded in social network theory (Halevy et al., 2019; Obstfeld et al., 2014), brokering refers to any "behavior by which an actor influences, manages, or facilitates interaction between other actors" (Obstfeld et al., 2014, p. 141). Accordingly, brokering practices are particularly valuable in contexts where actors with divergent temporal structures are disconnected. They facilitate

coordination through a limited set of bridging ties while preserving the structural separation between actors with disparate temporal structures.

To date, the application of brokering practices to manage divergent temporal structures has received limited attention within temporality research. Mell and colleagues (2021) introduced the concept of temporal brokerage in the context of globally distributed teams whose members operate across multiple time zones. These individuals exhibit distinct temporal patterns, as their working hours vary depending on their geographic location (Espinosa & Carmel, 2003). In extreme cases, two team members may have no temporal overlap, rendering their collaboration entirely asynchronous. Even partial asynchrony can hinder effective coordination and communication within such teams (Cummings et al., 2009). Mell and colleagues (2021) suggest that temporal brokers are positioned in intermediary time zones, having temporal overlap with both non-overlapping team members. This positioning allows them to mediate interaction by relaying information between temporally disconnected individuals.

Consequently, brokering practices can be employed to alleviate coordinative tensions while preserving actors' divergent temporal structures (Mell et al., 2021). Similar to brokerage as defined in social network theory, such practices facilitate collaboration between otherwise disconnected subgroups (Obstfeld, 2005). However, unlike conventional forms of brokerage, temporal brokering entails a lower risk of exploitative use of structural holes (Burt, 1992), as actors are separated only by temporal but not by relational or structural boundaries (Mell et al., 2021). In other words, communication remains possible, either asynchronously (Mell et al., 2021; Rhymer, 2023) or even synchronously when actors adjust their working hours (Chauvin et al., 2024). A key limitation of brokering practices, however, lies in the risk of overburdening individuals who occupy brokering positions (Mell et al., 2021). This underscores the boundaries of their effectiveness: brokering alone may not suffice to resolve coordinative tensions, particularly when actors are highly interdependent. However, the specific conditions under which brokering practices can effectively mitigate tensions arising from divergent temporal structures remain insufficiently understood.

With regard to the conditions under which brokering practices are applicable, the question arises whether, and how, they function in relation to temporal differences beyond time zones. Can brokering practices be employed to bridge actors who, for instance, exhibit

divergent temporal orientations or prefer to work at different rhythms? Although Mohammed and Nadkarni's (2011) study on temporal leadership does not explicitly propose a brokering role for leaders, it is conceivable that managers mediate between actors with divergent temporal orientations to whom they have assigned distinct tasks. However, in contexts where other dimensions of temporal structures differ, the mere transmission of information may be insufficient to resolve temporal tensions. In such cases, brokering practices may be designed and function differently, potentially assuming a synchronizing role (Henry & Möllering, 2023). Rather than simply relaying information, temporal brokers may engage in temporal boundary-spanning practices aimed at motivating actors with divergent temporal structures to, for example, extend or shorten their planning horizons, or adjust the pace at which they carry out their activities (Stjerne et al., 2019). Thus, our understanding remains limited as to whether brokering practices can assume different forms in coordinating across temporal differences. Thus, our understanding remains limited as to whether brokering practices can assume different forms in coordinating across temporal differences.

In summary, brokering practices offer a means to sustain divergent temporal structures while addressing the tensions that arise from them. They enable actors with differing temporal structures to remain separated, thereby avoiding the tensions that may emerge when different temporal patterns collide. At the same time, by establishing bridges between otherwise isolated temporal actors, brokering practices facilitate coordination. Given the limited attention they have received in the literature, this chapter highlights several open questions that warrant further investigation: Under which conditions are brokering practices effective in mitigating temporal tensions? Are these practices applicable to other dimensions of temporal structures? Which different forms do they take to coordinate effectively across various dimensions of temporal differences?

3.2.3. Negotiating

The objectives of both isolating and brokering practices are to sustain divergent temporal structures in a state of separation, with connections between them established solely through temporal brokers. In contrast, negotiating practices aim to integrate disparate temporal structures by encouraging actors to develop temporal reflexivity through

dialogue and contestation (Orlikowski & Yates, 2002; Reinecke & Ansari, 2015). Temporal reflexivity enables actors to recognize and understand diverse temporal conceptions and orientations (Cuganesan, 2022; Magrelli et al., 2022; Reinecke & Ansari, 2015). On this basis, they can establish ambitemporal patterns to coordinate their activities, thereby accommodating different temporal conceptions and orientations (Reinecke & Ansari, 2015).

The studies by Reinecke and Ansari (2015) and Magrelli and colleagues (2022) provide illustrative examples of negotiating practices. Reinecke and Ansari (2015) investigate conflicting temporal conceptions and orientations in the collaboration between a fairtrade organization and its certification unit. The fairtrade organization adopts an event-time conception characterized by a cyclical perception of time and considers long-term horizons spanning generations. In contrast, the certification unit embraces a clock-time conception with a strictly linear perception of time and operates within significantly shorter time horizons. These conflicting temporal conceptions and orientations led to disagreements over the specific criteria and timing of producer certifications in developing countries. Magrelli and colleagues (2022) observe temporal conflicts within family businesses, manifested as differing past, present, and future foci among generations. These varying temporal foci gave rise to prioritization and decision-making conflicts. In both cases, conflicting actors did not isolate themselves but engaged in ongoing dialogue through which they contested and negotiated their divergent temporal conceptions and orientations. Magrelli and colleagues (2022) further demonstrate that external mediators can facilitate such dialogue. This dialogue surfaced divergent temporal conceptions and orientations that are typically subconscious (Mohammed & Harrison, 2013), offering actors the opportunity to articulate the advantages of their own temporal conceptions and orientations while reflecting on their drawbacks for others. Both Reinecke and Ansari (2015) and Magrelli and colleagues (2022) thus show how negotiating and contesting divergent temporal conceptions and orientations through interaction fosters temporal reflexivity, defined as “the extent to which actors question, articulate, and rethink the temporal assumptions anchoring [their] organizational practices (Gevers et al., 2009; Orlikowski & Yates, 2002; Reinecke & Ansari, 2015; Tang et al., 2020, p. 220).

Although both studies (Magrelli et al., 2022; Reinecke & Ansari, 2015) demonstrate that actors can develop temporal reflexivity through dialogue and negotiation, it should not be regarded as an automatic outcome. It is possible that, despite engaging in dialogue and contestation, actors may continue to suppress temporal conceptions and orientations that differ from their own (Cuganesan, 2022). While existing research acknowledges temporal reflexivity as a potential outcome of negotiating practices, it has yet to specify the conditions under which this desirable outcome emerges. To consciously foster temporal reflexivity, it is therefore crucial to investigate the factors that facilitate or impede this development.

Temporal reflexivity alone may not be sufficient to resolve all tensions arising between actors with divergent temporal conceptions and orientations. While understanding and recognizing these differences can improve collaboration by alleviating tensions, such as prejudices (Waller et al., 2001) or distrust (Magrelli et al., 2022) toward actors with divergent temporal orientations, coordinative tensions may not be addressed through reflexivity alone. Nonetheless, temporal reflexivity can facilitate interpretative shifts among actors (Staudenmayer et al., 2002) and motivate the development of ambitemporal patterns that accommodate seemingly conflicting temporal conceptions and orientations (Reinecke & Ansari, 2015). For example, Fayard (2024) illustrates how actors interweave clock-time and event-time conceptions within their temporal patterns by combining clock-time-based timelines with specific events that guide product innovation processes. Similarly, Reinecke and Ansari (2015) demonstrate how actors transitioned from a single, unitary timeline for certifying local producers within a fairtrade organization to multiple, plural timelines that accounted for producers' varying developmental levels.

Negotiating practices resolve coordination challenges by enabling the development of new ambitemporal patterns. This suggests a complementary relationship between synchronizing and sustaining practices, as the coordinative effect of ambitemporal patterns relies on the creation of shared temporal patterns (Orlikowski & Yates, 2002). Although this relationship may appear paradoxical at first glance, it can be explained by the multifaceted nature of temporal structures (Ancona, Okhuysen, et al., 2001; Rowell et al., 2016). Negotiating practices thus show that synchronizing actors' temporal patterns can support the sustained coexistence of divergent temporal conceptions and orientations while simultaneously reducing potential tensions. However, as negotiating practices serve

as enablers rather than a definitive solution for resolving the coordinative tensions that arise between actors with divergent temporal structures, they raise important questions about when, and in response to which types of tensions, actors are inclined to pursue ambitemporal patterns, and what factors may constrain their willingness to do so. Reinecke and Ansari (2015) offer a valuable starting point for addressing these questions. Their study shows that actors did not achieve ambitemporality solely through temporal reflexivity but also because they recognized their mutual interdependencies. This suggests that strong interdependencies may be a critical condition that facilitates the transition from temporal reflexivity to the development of ambitemporal patterns. However, the enabling and constraining factors that influence when and why temporally reflective actors move from reflexivity to action and engage in the creation of ambitemporal patterns have yet to be fully understood.

In summary, negotiating practices enable actors to sustain their divergent temporal conceptions and orientations by fostering temporal reflexivity. This reflexivity, in turn, allows for the development of ambitemporal patterns that accommodate differing, even conflicting, temporal logics simultaneously. In contrast to isolating and brokering practices, which maintain separation between disparate temporal structures, negotiating practices aim to integrate them. While existing research has examined temporal reflexivity and ambitemporal patterns as outcomes of negotiation, it has offered limited insight into the underlying processes through which these outcomes emerge. To make the use of negotiation practices more targeted and effective, it is essential to deepen our understanding of how temporal reflexivity and ambitemporal patterns unfold through interactions among actors with divergent temporal structures. This entails addressing questions such as: How does temporal reflexivity emerge from dialogue and contestation? How do ambitemporal structures evolve from temporal reflexivity? Which factors facilitate or impede these developmental processes?

3.3. Interim Conclusion

The objective of Chapter 3 was to illustrate the various practices that actors can use to address tensions arising from their divergent temporal structures. The practices of entraining and developing new temporal patterns aim to synchronize divergent temporal

patterns, thereby creating shared temporal patterns that facilitate coordination. While entraining assumes an immutable Zeitgeber, developing new temporal patterns rests on the assumptions that a Zeitgeber can be actively shaped and reshaped through human agency. In contrast, the practices of isolating, brokering, and negotiating focus on sustaining temporal differences while managing the tensions they produce. Each practice impacts coordination and collaboration in distinct ways. Isolating prevents tension by severing interdependencies between actors, thus reducing the need for coordination. Brokering maintains separation between divergent temporal structures but enables coordination through intermediaries who mediate and translate across temporal boundaries. Negotiating, by contrast, seeks to integrate divergent temporal conceptions and orientations. It fosters the development of ambitemporal patterns that actors can draw upon to coordinate more effectively.

Throughout Chapter 3, I described each practice in terms of its functioning and its effects on temporal tensions, particularly those related to coordination and collaboration among actors with divergent temporal structures. For each practice, I also identified limitations and gaps in the existing literature. When viewed collectively, these gaps reveal three overarching themes that warrant greater scholarly attention and to which this dissertation seeks to contribute.

First, previous research has predominantly highlighted the positive outcomes of synchronizing practices, thereby neglecting their potential drawbacks and lacking a critical perspective on entraining and developing new temporal patterns as coordination practices amid temporal differences (Blagoev & Schreyögg, 2019; Luciano et al., 2018). By failing to acknowledge the disadvantages of synchronization, our understanding of why actors engage in or resist such practices remains incomplete (Shipp & Richardson, 2021), as we do not fully understand the agentic decision-making of non-time-setting actors. Thus, research needs to broaden its focus to incorporate both the disadvantages of synchronizing as well as the neglected role of non-time-setting actors to cultivate a more nuanced understanding of the factors that determine whether actors engage in or resist synchronization.

Second, extant research shows that the applicability of each of the three practices aimed at sustaining actors' unique temporal structures is confined to the specific conditions of the contexts in which they have been identified. Isolating practices have substantial

drawbacks in contexts marked by strong interdependencies that demand continuous collaboration (Oborn & Barrett, 2021). Similarly, brokering practices, so far, have only been found to coordinate effectively across time zone differences (Mell et al., 2021), leaving it unclear whether they can be effectively applied across other temporal differences. While negotiating practices can facilitate coordination even amid strong interdependencies between actors with divergent temporal structures, they do not automatically lead to the emergence of ambitemporal patterns. Additionally, negotiating practices primarily address divergent temporal conceptions and orientations but do not seem to support the preservation of distinct temporal patterns (Magrelli et al., 2022; Reinecke & Ansari, 2015). Consequently, our understanding of how divergent temporal structures can be preserved remains limited to specific conditions and excludes, for example, collaborative situations that are characterized by strong interdependencies.

Third, recent literature underscores the growing importance of the dynamics of temporal structuring. To date, research has focused on single episodes of temporal structuring, without considering whether and how these episodes coalesce into dynamic patterns of temporal structuring that trigger recurrent changes in actors' temporal structures. Given that neither synchronizing nor sustaining practices provide permanent solutions for coordination across divergent temporal structures (Hilbolling et al., 2022; Otto et al., 2024; Reinecke & Ansari, 2015), it is important to understand how effective coordination can be sustained. Shifting the focus from single episodes to the broader temporal structuring process might help to advance knowledge on how actors with divergent temporal structures effectively coordinate over time, by leveraging ongoing changes in their temporal structures to address different tensions at different times.

4. Methods

Building on the conceptual foundations established in the preceding chapters, this chapter outlines the research methods employed to investigate the central question of this dissertation: how do actors with divergent temporal structures coordinate their interdependent activities. Chapter 4.1 describes the research design. Chapter 4.2 introduces the research setting, collaborations between agile and non-agile units in the German financial services industry, which I expect to exhibit substantial temporal differences, making it a suitable context for examining this dissertation's central research question. Chapter 4.3 focuses on the case studies, detailing the case selection process and introducing the eight cases selected for this study. Chapter 4.4 elaborates on the data collection process and presents the three data sources: interview data, observational data, and archival data. Finally, Chapter 4.5 outlines the specific approaches used to analyze these data.

4.1. Research design

To explore how actors with divergent temporal structures coordinate their interdependent activities, I employ a multiple case study approach (Eisenhardt, 1989, 1991; Yin, 2009). A case study is defined as "an empirical inquiry that investigates a contemporary phenomenon within its real-life context" (Yin, 2009, p. 13). Several reasons justify the use of a case study approach in this dissertation. Most importantly, it addresses an exploratory research question, for which case studies are particularly well-suited (Eisenhardt, 1989; Yin, 2009). The overarching research question of this dissertation contributes to three areas that, as outlined in Chapter 3, have received limited attention in temporality research. First, by investigating the negative consequences of entraining and the neglected role of non-time-setting actors in synchronization processes, this dissertation aims to illuminate the decision-making processes through which actors engage in or resist synchronization. Second, it aims to identify practices that enable effective coordination while preserving differences in temporal structures. Third, it seeks

to shed light on the dynamic nature of temporal structuring by analyzing how temporal structuring episodes coalesce into broader coordination processes.

These research questions emphasize practices and processes, making a case study design particularly appropriate. Case studies have long been an established strategy for research adopting practice and process lenses, as demonstrated, for example, by Davis and Eisenhardt (2011), Feldman et al. (2022), Jarzabkowski and Lê (2012), Kremser and Blagoev (2021), and Oborn and Barrett (2021). Practice theory posits that situated action is inherently embedded in the social context in which it occurs (Feldman & Orlikowski, 2011). Accordingly, because practices cannot be meaningfully examined in isolation from their context, case studies are well-suited to practice-theoretical research, as they do not require separating a phenomenon from its setting (Yin, 2009). Instead, they generate the context-dependent knowledge that captures the nuanced realities of organizational life (Flyvbjerg, 2006; Gibbert et al., 2008). This allows researchers to identify practices and trace how they unfold in specific settings (Flyvbjerg, 2006). Eisenhardt (2021) emphasizes that case studies often rely on longitudinal data, which is essential for addressing process-theoretical questions. Process theory focuses on "the way in which phenomena emerge, evolve, or terminate over time through activities and events" (Cloutier & Langley, 2020, p. 3; Langley, 1999). Leonard-Barton (1990, p. 249) defines the case study "as a history of a past or current phenomenon," underscoring that observing phenomena over time is central to the case study method (Gephart, 2004). Case studies also accommodate multiple perspectives and levels of analysis (Yin, 2009), enabling researchers to uncover both the patterns of processes and the conditions shaping them (Leonard-Barton, 1990). Consequently, case studies are widely recognized as a powerful method for investigating complex, context-dependent, and temporally unfolding processes (Flyvbjerg, 2006; Graebner et al., 2012; Langley, 1999; Leonard-Barton, 1990). Given the practice- and process-oriented nature of this dissertation's research question, I selected a case study design.

In contrast to a single case study, the multiple case study design adopted in this dissertation involves several cases. A multiple case study approach employs a replication logic, whereby each additional case serves to confirm or refute the inferences drawn from previous ones (Eisenhardt, 1989; Yin, 2009). Consequently, multiple case studies are

often credited with producing more parsimonious, robust, and generalizable[5] theory that is less susceptible to biases such as misjudgments of representativeness based on single events (Eisenhardt & Graebner, 2007; Herriott & Firestone, 1983; Leonard-Barton, 1990; Yin, 2009). However, the benefits of a multiple case study design in this dissertation extend beyond generating more robust insights into synchronization processes, coordination practices, or temporal structuring patterns. A particularly valuable aspect is the capacity to compare cases systematically (Eisenhardt, 1989; Herriott & Firestone, 1983). Such comparisons can reveal differences among cases, as well as conditions that explain how synchronization processes and temporal structuring patterns unfold, and why actors select specific coordination practices.

Eisenhardt (1989) notes that case studies may incorporate both quantitative and qualitative data. For this dissertation, I have chosen to rely on a qualitative data set. Qualitative data are particularly well suited for exploring practices and processes because they provide rich and detailed descriptions (Weick, 2007). They enable the capture of nuances, effects, and mechanisms that are often inaccessible through the precise and narrowly defined constructs and measures typical of quantitative data (Graebner et al., 2012). This is especially important in the context of this dissertation, which focuses on the multifaceted and dynamic nature of temporal structuring, including synchronizing practices, sustaining practices, and evolving temporal patterns (Bansal & Corley, 2012).

4.2. Research setting

A case study benefits from a research setting in which the phenomenon of interest, namely the collaboration and coordination of actors with divergent temporal structures, is especially evident (Pettigrew, 1990; Yin, 2009). To address the central research question of this dissertation, I employed a purposeful sampling technique (Patton, 2015) to identify and select a meaningful research setting. Unlike randomized sampling methods commonly used in deductive research, purposeful sampling does not select a research

[5]The generalizability associated with multiple case studies is not statistical (Yin, 2009), but rather analytical generalizability, which refers "to the generalization from empirical observations to theory, rather than a population." (Gibbert & Ruigrok, 2010, p. 714)

setting at random to represent a broader population. Instead, it deliberately selects a context based on theoretical considerations that illuminate the phenomenon of interest (Eisenhardt, 1989; Eisenhardt & Graebner, 2007; Yin, 2009). Thus, purposeful sampling facilitates the identification of a revelatory research setting that offers rich and detailed insights into the phenomenon under study (Flyvbjerg, 2006; Merriam, 2009).

A revelatory research setting that enables investigation into how actors with divergent temporal structures collaborate and coordinate their interdependent activities, and that addresses the key questions of interest (How do actors make decisions regarding engaging in or resisting synchronization? Which coordination practices sustain divergent temporal structures? How do dynamic patterns of temporal structuring unfold?) requires clearly observable temporal differences. I expect such differences to emerge when agile and non-agile units within an organization are compelled to collaborate. In Chapter 2.2, drawing on foundational descriptions of the methods and practices underlying agile and non-agile units, I developed a conceptualization of their temporal structures and highlighted their key differences. First, plan-based practices typical of non-agile units promote a temporal conception grounded in clock time, whereas agile practices intertwine clock-time with event-time conceptions. Second, plan-based practices follow linear temporal patterns characterized by relatively lengthy activity phases executed sequentially, while agile practices exhibit cyclical temporal patterns with short cycles that recur rapidly and regularly. Third, with respect to their temporal orientations, plan-based practices cultivate distant future-time (and past) perspectives and time patience, whereas agile practices accommodate near future-time (and past) orientations and time urgency.

Collaborations between agile and non-agile units occur in a wide variety of organizations spanning different sizes and industries. The growing popularity of agile methods has prompted many organizations to adopt them (Annosi et al., 2020; Cooper & Sommer, 2016; Dybå & Dingsøyr, 2008; Rigby et al., 2018). When agile methods are introduced in specific units while the rest of the organization continues to operate with established, non-agile practices, a promising context emerges for studying collaborations between agile and non-agile units. This also applies to organizations undergoing long-term agile transformations (Dikert et al., 2016), provided they are still in the early stages and have not yet introduced agile practices organization-wide.

While many organizations may provide fertile ground for collaborations between agile and non-agile units, the extent of temporal differences between these units is likely to depend on contextual factors, such as the organization's size and industry. Context may influence the temporal structures of non-agile units more strongly than those of agile units. As outlined in Chapter 2.2, I define agile units by their use of agile management frameworks, such as the Scrum method. These frameworks are not specific to any industry or organizational size[6]. Therefore, as long as agile units adhere reasonably closely to the principles and practices of the chosen agile method, they tend to exhibit relatively consistent temporal structures across different organizational contexts. In contrast, non-agile units, defined by the absence of agile frameworks, are likely to display greater variation in their temporal structures. Their temporal structures depend on work practices ranging from plan-based methods to ad-hoc approaches (as described by Harris and colleagues, 2009) and may be shaped by the organization's size or industry. For instance, in small start-ups, ad-hoc work practices are often prevalent. Because agile practices share some characteristics with ad-hoc approaches, temporal differences between agile and non-agile units may be less pronounced in such settings. To observe more substantial temporal differences, I therefore selected a context in which non-agile units predominantly rely on plan-based methods.

Established banks and insurance companies may provide a fruitful organizational context for observing temporal differences between agile and non-agile units. I expect these organizations to rely heavily on plan-based work practices for several reasons. First, for many years, banks and insurance companies have operated in a highly stable market environment (Haveman, 1992), largely protected from competition by favorable market regulation (Kerfoot & Knights, 1993; Knights & Morgan, 1995; Webb & Pettigrew, 1999). This historical stability has shaped dominant organizational cultures characterized by conservatism and risk aversion, with a strong emphasis on stability and predictability

[6] Although the Scrum Guide provides clear recommendations regarding the size of the Scrum team (not exceeding eleven people, including the product owner and Scrum master), it does not limit the size of the organization in which the Scrum method is applied (Schwaber & Sutherland, 2020). Notably other agile management frameworks, in particular the Scaled Agile Framework (SAFe) and the Large-Scale Scrum (LeSS), claim to offer better tools for coordinating between multiple agile teams (Larman & Vodde, 2008; Leffingwell et al., 2016), and thus are often considered more suitable for use in large organizations. However, these frameworks are largely employed in organization-wide agile transformation initiatives (Dikert et al., 2016). Since the focus of this dissertation is on dyadic collaborations between selected agile and non-agile units, I do not warrant a detailed summary of this discussion.

(Stjerne et al., 2024; Vermeulen, 2005; Vermeulen et al., 2007). Such orientations align closely with plan-based practices, which aim to create stability, predictability, and assurance (Boehm & Turner, 2004). Second, these organizations respond to uncertainty through formalization and planning. With the deregulation of European financial markets in the 1980s and 1990s, their previously stable environment became more competitive and dynamic (Knights & Morgan, 1995; Muehlberger, 2007; Sturdy, 1997). In response, banks and insurers sought to control uncertainty by establishing formal rules and engaging in long-term strategic planning (Kerfoot & Knights, 1993; Knights & Morgan, 1995; Mena & Parker, 2024; Stjerne et al., 2024; Vermeulen et al., 2007). Controlling uncertainty through detailed planning is a key characteristic of plan-based practices (Bianchi et al., 2020; Boehm & Turner, 2004). Third, strict regulation of financial markets further promotes plan-based work practices. Regulators prioritize the stability of banks and insurance companies (Chick & Dow, 1996). For example, in Germany, the Minimum Requirements for Risk Management (MaRisk) for banks and the Solvency II Directive for insurers mandate detailed formal rules and process specifications to ensure stability, requiring organizations to adopt long planning horizons (BaFin, 2016, 2024). These regulations reinforce conservative, risk-averse cultures and efforts to control uncertainty through formalization and planning. For these reasons, I expect banks and insurance companies to make extensive use of plan-based methods. Several studies support this expectation (Stjerne et al., 2024; Vermeulen, 2005; Vermeulen et al., 2007). For example, Vermeulen (2005) finds that innovation processes in large Dutch and British banks and insurance companies follow a plan-based logic, characterized by extensive upfront planning of the innovation outcome, sequential execution of activities, and low customer involvement throughout the process.

Despite the expected dominance of plan-based methods, banks and insurance companies have increasingly adopted agile methods within selected units in recent years (Brühl, 2022; Rigby et al., 2018). This shift may be a response to their increasingly volatile market environment, characterized by numerous exogenous shocks such as the bursting of the tech bubble, the global financial crisis, the European debt crisis, and more recent disruptions from FinTech and InsureTech companies (Flier et al., 2003; Kunisch et al., 2021; Luger et al., 2018; Volberda et al., 2001). Because established banks and insurance companies generally operate with plan-based practices but have introduced agile methods in specific units, I consider them a revelatory context (Eisenhardt, 1989; Siggelkow,

2007) that provides substantial and observable temporal differences and enables the investigation of how actors with divergent temporal structures collaborate and coordinate their interdependent activities.

4.3. Case studies

Collaborations between agile and non-agile units in established banks and insurance companies provide a fruitful research setting in which I expect to observe how actors with divergent temporal structures coordinate their interdependent activities. The next step was to select case studies that exemplify such collaborations within this setting. This chapter presents the process of acquiring and selecting case studies (Chapter 4.3.1) and introduces the cases ultimately chosen for this dissertation (Chapter 4.3.2).

4.3.1. Acquisition and selection of case studies

To acquire participants for this study, I collaborated closely with another doctoral researcher. We began by searching for organizations that could provide compelling cases of collaboration between agile and non-agile units, reaching out to banks and insurance companies through social networks, the university's partnership network, and trade fairs. When presenting the study to representatives of interested organizations, we did not emphasize the topic of coordination across divergent temporal structures. Instead, we focused on the broader research context: collaborations between agile and non-agile units. Preliminary findings from twelve pilot interviews revealed that many established, predominantly non-agile organizations faced challenges when introducing agile methods in selected units. Consequently, the question of how agile and non-agile units can effectively collaborate and coordinate addressed a relevant, practical problem experienced by many of the organizations we contacted. Furthermore, by focusing on the research context without highlighting the role of divergent temporal structures, we avoided biasing participants' individual experiences and perceptions with our prior assumptions. Thus, this approach allowed us to critically evaluate both the suitability of

the chosen research context and the relevance of divergent temporal structures for successful collaboration between agile and non-agile units.

Eighteen agile units[7] from a total of ten banking and insurance organizations agreed to participate in the study. To determine whether these agile cases were suitable for this dissertation, I evaluated them against four essential conditions. Two of these conditions ensure that a case can reveal the coordination of actors with divergent temporal structures: (1) the agile unit must apply an agile management framework, and (2) the agile unit must directly collaborate with at least one non-agile unit. The other two conditions guarantee that the cases are supported by sufficiently rich data: (3) interviews must be conducted with actors from both the agile unit and at least one interdependent non-agile unit, and (4) collaboration between agile and non-agile units must be directly observable through participatory observations. For most cases, I was able to confirm or rule out these conditions based on information gathering during an initial conversation or interview, typically held with the product owner or Scrum master of the agile unit. Table 6 summarizes the four criteria and the results of case selection.

The decision to focus on collaborations between agile and non-agile units rests on the assumption that agile management frameworks generate distinct temporal structures that differ from those associated with non-agile work practices. Accordingly, the first condition that all cases included in this dissertation had to meet is that the focal agile unit applied an agile management framework as faithfully as possible. Except for case [F1], all selected cases satisfied this condition by implementing the Scrum method. In case [F1], the agile unit adopted only selected Scrum elements while largely maintaining established, non-agile work practices. As a result, the temporal structures of the agile unit in [F1] did not differ significantly from those of its interdependent non-agile units, and I therefore excluded it from this dissertation.

An equally important condition is that agile units collaborate directly and regularly with at least one non-agile unit. This criterion is intended to ensure that both the tensions

[7] This dissertation does not focus on agile organizational units, but on collaborations between agile and non-agile organizational units. However, in the search for such collaborations, it proved pragmatic to approach agile units first, since in the largely non-agile banks and insurance companies, agile units are likely to collaborate with at least one non-agile unit, whereas many non-agile units do not have any interdependence with the few agile units.

Table 6: Summary of the selection process for case studies

Case:	**Condition 1:** Application of an agile management framework in an agile unit	**Condition 2:** Direct collaboration between agile and non-agile units	**Condition 3:** Interviews with members of agile and non-agile units	**Condition 4:** Possibility to collect data in participatory observations	Status:
[A1]	yes	yes	yes	yes	included
[A2]	yes	yes	yes	yes	included
[A3]	yes	yes	yes	no, project terminated	excluded
[A4]	yes	weak interdependencies between agile and non-agile units	yes	no direct collaboration between agile and non-agile units	excluded
[B1]	yes	yes	yes	yes	included
[B2]	yes	yes	yes	yes	included
[B3]	yes	yes	yes	yes	included
[C]	yes	yes	yes	yes	included
[D]	yes	yes	yes	yes	included
[E]	yes	yes	yes	yes	included
[F1]	limited, application of selected elements of the Scrum method	yes	yes	yes	excluded
[F2]	yes	yes	yes	project terminated	excluded
[F3]	yes	yes	yes	project terminated	excluded
[G]	yes	collaboration between agile units only	collaboration between agile units only	yes	excluded
[H1]	yes	yes	provision of one informant only	no permission for participatory observations	excluded
[H2]	yes	yes	provision of one informant only	no permission for participatory observations	excluded
[I]	yes	yes	no provision of one informant only	no permission for participatory observations	excluded
[J]	yes	yes	provision of one informant only	project terminated	excluded

arising from divergent temporal structures, as well as the practices and processes through which agile and non-agile units manage to collaborate, can be observed. Two cases did not meet this requirement. In case [A4], although actors from both agile and non-agile units reported mutual interdependencies, these proved to be relatively weak. The agile unit was able to complete its tasks independently, without involving non-agile units. Although it held regular sprint reviews, non-agile units were not invited to participate, as their input was not (yet) required. In case [G], the focal agile unit operated within an organization that had already adopted agile methods across a significant portion of its units. As a result, the focal unit collaborated extensively with other agile units, but I did not observe collaboration with non-agile units.

Conditions three and four were designed to ensure that the selected cases provide rich data and capture multiple perspectives on collaboration between agile and non-agile units. To avoid an exclusive emphasis on the agile perspective, it was essential to conduct interviews with actors from non-agile units as well. In addition to case [G], where the agile unit had no interdependencies with non-agile units, we were unable to conduct interviews with members from non-agile units in cases [H1], [H2], [I], and [J]. Moreover, to minimize informant bias (Golden, 1992) and enable triangulation (Eisenhardt, 1989; Yin, 2009), it was necessary to conduct participatory observations of collaborative interactions between agile and non-agile units. This was not feasible in cases [A4] and [G] due to the absence of such collaboration. In cases [A3], [F2], and [F3], the agile projects under examination had already concluded, preventing the collection of real-time observational data. In cases [H1], [H2], [I], and [J], we were not granted access to participatory observations. While cases [A4] and [G] had already been excluded based on the second condition, I also excluded cases [A3], [F2], [F3], [H1], [H2], [I], and [J] for not meeting the data richness criteria.

During the selection process, I identified eight case studies ([A1], [A2], [B1], [B2], [B3], [C], [D], and [E]) that met all four conditions. This number falls within the range of four to ten cases that Eisenhardt (1989) recommends as a suitable basis for analytical generalization. The following chapter introduces these cases.

4.3.2. Introduction to the selected cases

This chapter introduces the eight cases that meet the criteria established in the selection process and that form the empirical foundation of this dissertation. Each case involves direct collaboration between agile and non-agile units. The primary objective of the chapter is to characterize these collaborations in terms of their coordination needs. In addition, it aims to highlight the contextual differences across cases. In multiple case study designs, such differences are not considered problematic. Yin (2009) argues that replications should not only be carried out with cases that duplicate the characteristics of an initial case, but also with those that differ along selected dimensions. This approach strengthens the robustness of findings by testing their reliability and identifying the conditions under which they apply (Leonard-Barton, 1990; Yin, 2009). Table 7 presents an overview of the key characteristics of the eight cases examined in this dissertation.

I begin by outlining the contextual characteristics of the cases. The collaborations between agile and non-agile units examined in this dissertation are situated either in banks ([C] and [E]) or in insurance companies ([A1], [A2], [B1], [B2], [B3], and [D]). This enables an analysis of the potential influence of industry context, banking or insurance, on the manifestation of different temporal structures and tensions, as well as on the selection and effectiveness of the practices used to address them. Cases [A1] and [A2], as well as [B1], [B2], and [B3], are each situated within the same insurance company, respectively. This case selection thus supports comparisons both across and within organizations, allowing for the identification of potential organization-level influences.

Across all cases, the agile units share the common feature of having implemented the Scrum method with a high degree of fidelity. Consequently, I expect that these units display similar work processes and temporal structures, shaped substantially by the principles outlined in the Scrum Guide (Schwaber & Sutherland, 2020). However, the cases differ in the duration for which Scrum has been applied. In most cases, the method is relatively new. In cases [A1], [B2], [B3], and [C], Scrum was introduced in 2020, approximately one year prior to the start of the data collection. In case [A1], it was introduced into an existing software development team that had previously relied on plan-based methods. In contrast, the agile units in cases [B2], [B3], and [C] were established

Table 7: Summary of selected case studies

Case information:		**Characteristics of agile units:**			**Coordination with interdependent non-agile units**		
Case:	**Industrial context:**	**Agile method**	**Unit type:**	**Purpose and objectives:**	**Non-agile interdependent units:**	**Coordination needs:**	**Type of inter-dependence:**
[A1]	insurance	Scrum (since 2020)	permanent line unit	*software development:* development and maintenance of sales software for life insurance products, used by sales agents	regulatory compliance unit	*input provider & governance:* the agile unit depends on the information about legal and regulatory requirements provided by the regulatory compliance unit; the regulatory compliance unit depends on the agile unit for implementing regulatory changes;	reciprocal
					IT units	*input provider & end user:* the agile unit must synchronize its software development activities with non-agile backend development units in order to ensure its functioning; and vice versa;	reciprocal
					sales agents	*end user:* the agile unit depends on the sales agents for reporting software bugs and suggestions for improvement; sales agents depend on the agile unit for developing and maintaining the sales software they use;	reciprocal
[A2]	insurance	Scrum (since 2017)	permanent line unit	*software development:* development and maintenance of administration software for life insurance products, used by the customer service unit	regulatory compliance unit	*input provider & governance:* the agile unit depends on the information about legal and regulatory requirements provided by the regulatory compliance unit; the regulatory compliance unit depends on the agile unit for implementing regulatory changes;	reciprocal
					product management unit	*input provider:* the agile unit depends on new and changed product specifications provided by the product management unit;	sequential
					IT units	*input provider:* the agile unit depends on the development of specialized software functionalities provided by several IT units;	sequential
					customer service unit	*end user:* the agile unit depends on the customer service unit for reporting software bugs and suggestions for improvement and automation; the customer service unit	reciprocal

Case information:		Characteristics of agile units:			Coordination with interdependent non-agile units		
Case:	**Industrial context:**	**Agile method**	**Unit type:**	**Purpose and objectives:**	**Non-agile interdependent units:**	**Coordination needs:**	**Type of inter-dependence:**
						depends on the agile unit for developing and maintaining the administration software they use;	
[B1]	insurance	Scrum (since 2018)	temporary project	*software development:* development and maintenance of sales software for different insurance products, used by sales agents	data warehouse unit	*input provider*: the agile unit depends on the data warehouse unit for establishing the technical interfaces between the sales software and the data warehouse that enable the sales software to access customer data;	sequential
					management	*input provider & governance:* the agile unit depends on the company's management for the allocation of project resources and critical decision-making; management depends on the agile unit for developing a sales software that contributes to achieving the company's efficiency goals;	reciprocal
					sales agents	*end user & input provider:* the agile unit depends on the sales agents for providing usability feedback on the sales software; the sales agents depend on the agile unit for developing an improved sales software for their daily activities;	reciprocal
[B2]	insurance	Scrum (since 2020)	permanent line unit	*software development:* development and maintenance of tariff calculators for different insurance products, used by external customers	insurance line units	*input provider*: the agile unit depends on the insurance line units for seconding staff and for providing specific information on insurance products and contracts that are required to program tariff calculators;	sequential
					management	*governance:* the agile unit depends on the company's management for critical decision-making and task prioritization; management depends on the agile unit for developing tariff calculators that contribute to achieving sales goals;	reciprocal
					IT units	*input provider*: the agile unit depends on the company's IT units for providing the hardware and IT infrastructure required to run the tariff calculators;	sequential

Case information:		Characteristics of agile units:			Coordination with interdependent non-agile units		
Case:	**Industrial context:**	**Agile method**	**Unit type:**	**Purpose and objectives:**	**Non-agile interdependent units:**	**Coordination needs:**	**Type of inter-dependence:**
[B3]	insurance	Scrum (since 2020)	permanent line unit	*software development:* development and maintenance of a customer self-service portal, used by external customers	insurance line units	*input provider*: the agile unit depends on the insurance line units for seconding staff and for providing specific information on insurance products and contracts that are required to develop a self-service customer portal;	sequential
					management	*governance:* the agile unit depends on the company's management for critical decision-making and task prioritization; management depends on the agile unit for developing a customer portal that contributes to achieving customer satisfaction goals;	reciprocal
					IT units	*input provider*: the agile unit depends on the company's IT units for providing specialized technical deliveries, such as web design elements;	sequential
[C]	banking	Scrum (since 2020)	temporary project	*software development:* development of a regulatory reporting software, used by the regulatory reporting unit	regulatory reporting unit	*end user & input provider:* the agile unit depends on the regulatory reporting unit for providing information on reporting processes and the usability of the software; the regulatory reporting unit depends on the agile unit for obtaining improved regulatory reporting software;	reciprocal
					data warehouse units	*input provider*: the agile unit depends on two data warehouse units for establishing the technical interfaces between the data warehouses and the regulatory reporting software, which enables the latter to access financial data;	sequential
					release management unit	*release management:* the agile unit depends on the release management unit for making software increments available to end users;	sequential
					management	*governance:* the agile unit depends on the company's management for allocating project resources and critical decision-making; management depends on the agile unit to develop a regulatory reporting software that ensures efficient and correct regulatory reporting;	reciprocal

Case information:		Characteristics of agile units:			Coordination with interdependent non-agile units		
Case:	**Industrial context:**	**Agile method**	**Unit type:**	**Purpose and objectives:**	**Non-agile interdependent units:**	**Coordination needs:**	**Type of inter-dependence:**
[D]	insurance	Scrum (since 2019)	temporary project	*software development:* development of a quotation and administration software for composite insurance products, used by insurance line units	insurance line units	*end user & input provider*: the agile unit depends on the insurance line units for seconding staff and providing specific information on insurance products and work processes required to develop an administration software for liability and property insurances; insurance line units depend on the agile unit for obtaining a software that allows them to administer liability and property insurance contracts efficiently;	reciprocal
					IT units	*input provider:* the agile unit depends on the IT units for seconding staff and providing knowledge about the company's IT work processes and software standards;	sequential
					release management unit	*release management:* the agile unit depends on the release management unit to make software increments available to end users;	sequential
					management	*governance:* the agile unit depends on the company's management for critical decision-making and allocating project resources; management depends on the agile unit to develop a customer portal that contributes to achieving efficiency goals;	reciprocal
[E]	banking	Scrum (since 2021)	temporary project	*restructuring:* concept development for optimizing business process and organization structures in a customer advisory unit	corporate client advisory unit	*end user & input provider*: the agile unit depends on the customer advisory unit for seconding staff and for providing specific information about customer advisory work processes; the customer advisory unit depends on the agile unit to develop concepts for optimized work processes and team structure that fulfill its needs and interests;	reciprocal
					works council	*governance:* the agile unit depends on the works council for obtaining the approvals required under German co-determination rights for the implementation of the developed concepts; the works council depends on the	reciprocal

Case information:		Characteristics of agile units:			Coordination with interdependent non-agile units		
Case:	**Industrial context:**	**Agile method**	**Unit type:**	**Purpose and objectives:**	**Non-agile interdependent units:**	**Coordination needs:**	**Type of inter-dependence:**
						agile unit to develop concepts that adequately represent employee interests;	
					IT units	*input provider:* the agile unit depends on the IT units for providing estimates of the technical feasibility of the developed concepts for structure and process optimizations as well as reviewing compliance with software and security standards;	sequential
					management	*governance:* the agile unit depends on the company's management for critical decision-making; management depends on the agile unit to develop new team structures and work processes for the customer advisory unit that contribute to achieving sales and efficiency goals;	reciprocal

2020 and adopted agile practices from the beginning. Similarly, the agile unit in case [E] applied Scrum from the beginning of its operations. However, the unit itself was only established three months before data collection began in 2021. In cases [A2], [B1], and [D], Scrum had already been in use for several years, most notably in case [A2], where it was introduced in 2017. These variations in the length of Scrum implementation enable conclusions regarding whether certain practices represent emerging responses to temporal tensions or have already been established as routine solutions.

The case selection also facilitates an analysis of how different organizational integrations of agile units, as well as varying project durations, influence collaboration. In half of the cases, the agile units were established as permanent line units ([A1], [A2], [B2], and [B3]). In cases [A1] and [A2], the Scrum method was introduced into pre-existing software development teams within the organization's IT function, which had previously relied on plan-based approaches. In contrast, in cases [B2] and [B3], management established new agile line units in 2020 to deliver digital service solutions directly to customers. At the time of data collection, these units had only recently transitioned from their initial start-up phases to operational work. In the remaining cases ([B1], [C], [D], and [E]), the agile units were organized as temporary projects. The (anticipated) durations of these projects varied: several years in cases [B1], [C], and [D], and several months in case [E]. The agile unit in case [E] adhered to its planned four-month project timeline, followed by a three-month implementation phase. In case [C], the agile project lasted for three years but discontinued its use of the Scrum method toward the end of our data collection period, transitioning to a plan-based project. The total durations of the agile projects in cases [B1] and [D] remain unknown. Both had already been active for several years at the time of data collection and had not concluded by its end. In case [D], the agile project team was preparing an application for a two-year extension. In case [B1], the project lacked a fixed end date but was required to reapply for funding annually to continue operations. Although the team could not predict the precise project end date, there was a shared understanding that the project would eventually conclude.

In seven of the eight cases ([A1], [A2], [B1], [B2], [B3], [C], and [D]), the agile units were responsible for developing software solutions for internal or external customers and were embedded within the IT functions of the respective banking or insurance companies. Table 7 provides further details on the specific types of software developed in each case.

In case [E], by contrast, the agile unit was tasked with developing a restructuring concept for an internal corporate client advisory unit. The inclusion of case [E] in the sample allows for a comparison of temporal structures, temporal tensions, and the practices used to address them across different types of tasks.

Finally, I will now turn to the coordination needs between agile and non-agile units. Table 7 shows that each agile unit collaborates with several non-agile units. These non-agile units can be clustered into four distinct types of actors: (1) governance, (2) input providers, (3) end users, and (4) release management. Coordination needs between agile units and these distinct types of actors differ. In the following, I will describe the unique coordination needs of agile units with each of these actors, and based on this, derive their type of interdependence. For this purpose, I draw on Thompson's (1967) definitions of sequential and reciprocal interdependencies. Sequential interdependencies describe that a focal unit depends on the input of an upstream unit to produce its output, while a downstream unit is dependent on the focal unit's output. In reciprocal interdependencies, "the outputs of each become inputs for the others" (Thompson, 1967, p. 55). Thus, unlike sequential interdependencies, reciprocally interdependent units have bidirectional instead of unidirectional orders in their dependency relationships.

Across all cases, agile units coordinated with governance actors. Governance actors included management, executive functions acting on behalf of management (e.g., the regulatory compliance unit in cases [A1] and [A2], and works councils. Coordination needs between agile units and governance actors centered on resource allocation, critical decision-making, and monitoring. Agile units were dependent on management for resource allocation. In most cases, management was represented by a steering committee composed of senior executives, often the heads of interdependent non-agile units. Only in case [E] did the agile unit report directly to the board of directors. In order to secure resources for carrying out their activities, agile units participated in managerial budgeting and planning processes. Because resource allocation was closely tied to task planning, coordination between agile units and management primarily focused on defining tasks and the resources needed to execute them. While agile units were responsible for proposing tasks, the final decisions regarding their prioritization rested with management. In addition, governance actors were responsible for making critical decisions. Examples of managerial decisions included short-term changes to task priorities and conflict

resolutions. Decisions of work councils concerned approvals of work outcomes or approaches subject to co-determination. With regard to decision-making, coordination between agile units and governance actors involved sharing problems and obstacles and providing information to facilitate informed decisions. In their monitoring role, governance actors also coordinated with agile units to receive regular reports and status updates on task progress. To meet this need, they organized and convened regular coordination meetings. Ultimately, management, holding responsibility for organizational success, was indirectly reliant on agile outcomes, given their expected impact on revenues and expenses. Accordingly, since governance actors and agile units exhibited mutual coordination needs, their interdependence can be characterized as reciprocal.

Moreover, agile units depended on input-providing units during task execution. These units provided the information, resources, or deliverables required for agile units to execute their tasks. For instance, in cases [A1] and [A2], the regulatory compliance unit supplied information on regulatory changes that had to be integrated into the software. In other cases, such as [D], [B2], and [B3], non-agile units provided resources by seconding staff into the agile units. Often, agile units needed specific deliverables that were developed by non-agile units. For example, in cases [B1] and [C], agile units required technical interfaces to connect new software to the organizational data warehouses. Agile units coordinated closely with input-providing units to define their required inputs and agree on their delivery schedule. As input-providing actors did not rely on outcomes of the agile units, they represented sequential interdependencies.

In contrast, the interdependence between agile units and end-user units was reciprocal. End-user units provided important inputs while simultaneously being the intended beneficiaries of agile units' outputs. Agile units coordinated frequently with them to identify opportunities for optimization and to gather feedback through usability tests or sprint reviews. End users in software development cases were typically the first to detect software bugs and report them to agile units. In addition, end users' operational expertise enabled them to identify problems and inefficiencies in existing work processes. Finally, coordination needs between agile units and end users often encompass the timelines for releasing or handing over outputs.

Finally, agile units had to coordinate with release management units, which provided the infrastructure for software deployment. Release management units were primarily responsible for ensuring that new software did not introduce errors or compromise the functionality of the organization's software landscape. To this end, release management units conducted regression tests, which had to be completed before the software could be employed. Coordination between agile units and release management primarily revolved around the execution of these tests. Release management monitored whether software was submitted on time for testing activities. If regression tests uncovered software defects, agile units and release management units coordinated to correct and retest them. Ultimately, release management units held the authority to approve or reject software for release. While agile units were highly dependent on release management, release management itself did not rely on the outputs of agile units. Thus, the interdependence between release management and agile units is sequential.

This chapter introduced the eight cases selected for this dissertation. Although the cases differ in various contextual characteristics, they all represent collaborations between agile and non-agile units. The chapter demonstrated that, across all eight cases, agile and non-agile units have similar coordination needs. This provides a solid foundation for comparing the tensions arising from divergent temporal structures, as well as the practices employed to mitigate them. Nonetheless, contextual differences may offer valuable insights into the variation in tensions in mitigation practices observed across cases. The following Chapter 4.4 presents the data collection.

4.4. Data collection

This chapter outlines the data underpinning this dissertation and details the procedures used for data collection. Following the recommendations of Yin (2009) and Leonard-Barton (1990), who emphasize the importance of drawing on multiple sources of evidence in case study research, I collected three types of qualitative data: (1) interview data, (2) data from participatory observations, and (3) archival data. This qualitative data facilitates rich and nuanced descriptions of practices and processes (Bansal & Corley, 2012; Graebner et al., 2012; Weick, 2007). The use of multiple data sources also enables

triangulation, thereby enhancing the accuracy and validity of the findings (Eisenhardt, 1989).

Data collection was conducted between June 2021 and September 2023 in collaboration with another doctoral researcher. Employing a multiple-investigator approach offers two key advantages (Eisenhardt, 1989). First, different investigators bring diverse perspectives to the data, helping to mitigate the potential biases or limitations of a single researcher and enabling the generation of new insights. Second, the convergence of interpretations across investigators strengthens the credibility, validity, and robustness of the findings. Table 8 provides a summary of the results of the data collection process, both in aggregate and disaggregated by case. The following sections provide a detailed account of the data collection procedures for each data source.

Table 8: Summary of data collection

	Interviews		**Participatory observations**			**Documents**
Case	*Number:*	*Hours:*	*Number:*	*Hours:*	*Field note pages:*	*Pages:*
[A1]	4	4	2	1	15	21
[A2]	5	5	14	17	186	57
[B1]	11	9	4	3	28	13
[B2]	9	8	21	10	100	18
[B3]	9	7	8	6	99	50
[C]	30	16	122	69	863	190
[D]	12	9	9	7	108	35
[E]	18	10	90	60	729	258
Total	**91**[8]	**62**	**256**	**173**	**2,128**	**642**

4.4.1. Interviews

We conducted a total of 91 semi-structured interviews with informants from both agile and non-agile units across all cases. Our informants include 50 individuals working in

[8] In cases [A1], [A2], [B1], [B2], and [B3], a total of five informants provided insights across multiple cases. While the number of interviews per case indicates how many informants were interviewed in each individual case, the total was calculated without counting duplicate contributions. Thus, compared with the sum of all cases, the total displays a delta of seven interviews (91 instead of 98), with a duration of six hours (62 instead of 68).

agile units, 38 individuals in non-agile units, and three individuals with dual roles affiliated with both. Of the total, 42 informants were regular team members, while 49 held leadership or leadership-related roles. In non-agile units, leadership roles were typically formal and included positions such as project managers or unit heads. In agile units, I classified product owners and Scrum masters as leadership-related roles due to their responsibilities in guiding agile teams and facilitating processes. A detailed overview of the informants per case is provided in Table 16 in the appendix.

We did not select informants randomly, as our objective was not to obtain representative samples of all employees in agile and non-agile units. Instead, we specifically sought informants with deep insights into the collaboration between these two types of units (G. P. Huber & Power, 1985; Kumar et al., 1993). Such insights are not evenly distributed. In both agile and non-agile units, employees are often shielded from external interactions and thus have limited direct contact with stakeholders during their daily work. However, due to regular sprint reviews, Scrum team members are more likely to engage directly with representatives of non-agile units. This explains the slightly higher number of informants from agile units. In each case, data collection began with an interviewee from an agile unit who held a leadership-related role, typically a product owner or Scrum master, responsible for communication and coordination with interdependent (non-agile) units. These initial interviews helped us understand how the agile units were embedded within the broader organizational (non-agile) context, including their interdependencies and coordination needs. In addition, these preliminary informants recommended further knowledgeable informants from both agile and non-agile units, enabling us to identify additional interviewees through snowball sampling (Merriam, 2009). In many cases, they also facilitated the initial contact, which helped increase informants' willingness to take part in the study.

Three overarching questions guided the semi-structured interviews. First, we sought to understand the interdependencies between agile and non-agile units, with particular focus on what necessitated coordination between them. This question was discussed intensively during the preliminary interview in each case. It was revisited in further interviews to validate perceptions regarding the nature and extent of interdependencies and coordination needs. Second, we inquired about the tensions informants had experienced in the collaboration between agile and non-agile units. We began with open-ended

questions (e.g., “How do you experience the collaboration with the agile unit?”), designed to encourage informants to share detailed narratives of both positive and negative experiences. When these narratives revealed situations particularly relevant to our research question, we employed more directive and suggestive questioning techniques to explore finer details. Directive questions also proved helpful in prompting informants to recount specific events and situations when their responses tended to be generalized or abstract in describing tensions. The third guiding question focused on the practices that agile and non-agile units applied to address the tensions experienced in their collaboration. For this question, we used directive questioning techniques from the outset, following up on tensions reported by informants. Our queries centered on how these practices functioned and informants’ evaluations of their effectiveness. The guiding questions were not posed in a strictly sequential order; rather, conversations iterated between tensions and the practices used to resolve them. Within our team of two doctoral researchers, we aimed to conduct each interview jointly. Occasionally, scheduling conflicts necessitated splitting up when interviews overlapped. In interviews where both researchers were present, one moderated the conversation along the guiding questions, while both posed in-depth follow-up queries.

Interviews ranged in duration from eleven to 93 minutes, with an average length of 42 minutes. As the number of interviews per case increased, the duration of individual interviews tended to decrease, reflecting our growing understanding of the contextual details of the cases, especially of interdependencies and coordination needs. All interviews were conducted via virtual communication platforms (e.g., Microsoft Teams, WebEx, or Zoom), based on the preferences of the informants. With informants’ consent, interviews were audio-recorded and subsequently transcribed, resulting in a total of 1,355 pages of transcripts.

Interview data are susceptible to informant bias, which refers to the distortion of subjective perceptions and experiences due to factors such as individuals’ specific roles or difficulties in accurately recalling past events (Graebner et al., 2012; G. P. Huber & Power, 1985). Several aspects of our data collection process were designed to mitigate informant bias. First, informants were assured of confidentiality, which can encourage more accurate responses and reduce the likelihood of distorted statements due to fear of sanctions (Miller et al., 1997). Second, we interviewed multiple informants per case. By

compensating for the biased perceptions of individual informants, multiple perspectives can enhance the accuracy and objectivity of researchers' understanding of each case (Hannah & Eisenhardt, 2018). Third, we relied on multiple data sources, particularly triangulating retrospective interview data, which may be prone to informant bias, with real-time data from participatory observations to increase the accuracy and validity of our findings (Golden, 1992; Miller et al., 1997). The next chapter details the procedures for collecting data from participatory observations.

4.4.2. Participatory observations

We collected data from 256 participatory observations, amounting to a total of 173 hours. While interviews provided retrospective insights into the tensions experienced by agile and non-agile units in their collaboration, as well as the practices through which these tensions were addressed, participatory observations complemented our dataset by offering real-time data. Such real-time data not only mitigates informant bias (Golden, 1992; Miller et al., 1997) but also enables us to trace the evolution of tensions and coordination practices over time, observe their unfolding dynamics, and thereby capture the interplay between them (Leonard-Barton, 1990; Pettigrew, 1992).

We observed meetings in which agile and non-agile units collaborated directly (e.g., sprint reviews, jour fixes, steering committee meetings), meetings that served to prepare such collaboration (e.g., daily standups, sprint plannings), as well as meetings that reflected upon it (e.g., sprint retrospectives). The eight cases varied in terms of the meeting formats to which we were granted access and the time periods during which collaboration between agile and non-agile units was observed. These differences account for the variation in the number and duration of observations across cases, as presented in Table 8. While we were granted access to selected meetings in cases [A1], [A2], [B1], [B2], [B3], [C], and [D], I was allowed to join the agile unit in case [E] as an intern for a period covering three sprints. This role allowed for deeper immersion into the collaboration between the agile unit and its interdependent non-agile units, including interactions occurring outside the agile routine events. For example, I observed both one-off meetings and ad-hoc conversations with selected representatives of non-agile units,

during which Scrum team members sought specific information or feedback on the development of their concepts. Accordingly, case [E] includes a relatively high number of (virtual) meeting observations. The total of 90 observations with a duration of 60 hours, as indicated in Table 8, includes all meetings I attended in case [E], but excludes the time spent working on agile unit tasks outside of these meetings. A comprehensive list of all meetings attended per case is provided in Table 17 in the appendix.

Except for case [E], the majority of meetings were attended jointly by our team of two doctoral researchers. The use of multiple observers is recommended in prior research, as it enhances the reliability and accuracy of observational data (Leonard-Barton, 1990). At the outset of the first meeting in each case, we briefly introduced ourselves and explained the purpose of our presence, namely, to gain insights into the collaboration between agile and non-agile units as part of our doctoral research. We ensured that all participants consented to our attendance. In cases [A1], [A2], [B1], [B2], [B3], [C], and [D], we also clarified that our role was strictly observational. In case [E], however, I was not only an observer but also a temporary member of the Scrum team. In this role, I actively participated in tasks and contributed to the collaboration between agile and non-agile units. Nevertheless, at the beginning of my internship in case [E], I informed all involved parties of my primary objective: collecting data on agile-non-agile collaboration. During all meetings, we took extensive field notes and engaged in post-meeting reflection to compare and synthesize our observations. These reflective discussions enabled us to identify converging impressions, examine discrepancies, and aggregate our individual notes into shared observation protocols. The resulting documentation includes “thick descriptions” of collaborative situations (Geertz, 1973, p. 6), amounting to a total of 2,128 pages.

Except for a few meetings in case [E], all observed meetings took place virtually, as organizations supported remote work during the COVID-19 pandemic. Consequently, our observations focused primarily on verbal interactions between representatives of agile and non-agile units. The virtual setting limited our ability to observe non-verbal cues, as many participants either kept their cameras off or only a small number of participants were visible on screen at any given time. Whenever non-verbal signals were observable, we documented them in our field notes and discussed them during our post-meeting reflections.

A limitation of our observational data collection is that, in most cases, it was confined to selected meetings. This approach, observing a phenomenon of interest only at specific points in time, is not uncommon (see, for example, Leonard-Barton, 1990) and constitutes a relatively efficient form of observation (Herriott & Firestone, 1983). However, it does not meet the expectations of conventional ethnography, which typically requires researchers to develop close relationships with participants (van Maanen, 1988) and gain deep insights into both formal and informal aspects of collaboration (Leonard-Barton, 1990). Collaboration outside of agile routine events and dedicated jour fixes was observed only in case [E], and even there, it was limited due to Covid-19 restrictions, which strongly encouraged remote work. During on-site visits in case [E], I observed minimal informal communication within the agile unit and between the agile unit and its non-agile stakeholders. Employees working remotely did not encounter one another by chance in hallways, at coffee machines, or during lunch breaks. Consequently, even in case [E], communication was largely confined to scheduled meetings. In the other cases, organizations similarly supported remote work arrangements: employees were rarely on-site, and all meetings were held virtually. I therefore assume that a conventional ethnographic approach involving continuous on-site presence would have yielded few additional insights into informal collaboration, simply because such interactions were rare at the time of data collection. While our focus on selected meetings where agile and non-agile units formally collaborated limits the depth of our observational data, I argue that, given the prevailing working conditions, these meetings nonetheless provided sufficiently rich insights into the nature of agile-non-agile collaboration.

4.4.3. Archival data

Finally, we collected archival data comprising a total of 642 pages of documents across all cases. Table 8 presents the breakdown by case. These materials include excerpts and screenshots from digital collaboration tools (e.g., Jira), management presentations, and sections of organizational manuals provided by our informants. All documents were anonymized by removing any information that could reveal the identities of individuals or organizations.

Most of the documents consisted of excerpts, displaying only selected information from digital collaboration tools, management presentations, or organizational manuals. Each document type contributed distinct insights. In particular, excerpts and screenshots from digital collaboration tools enhanced our understanding of the interdependencies between agile and non-agile units. For instance, the collaboration tool Jira, commonly used by agile units, documented the description, progress, and status of tasks planned for each sprint, often updated daily. Task descriptions frequently indicated whether coordination with non-agile units was required. Occasionally, these tools also revealed tensions; for example, employees sometimes recorded delays in Jira, noting that progress on specific tasks was hindered by pending input from interdependent non-agile units. In addition, Jira often displayed the product backlog, offering visibility into upcoming tasks. Management presentations provided valuable contextual information. In several cases, they explained the rationale for adopting agile methods or illustrated corporate planning processes. They frequently included timelines for software releases, as well as the milestones and deadlines to which both agile and non-agile units were expected to adhere. Excerpts from organizational manuals were particularly helpful for understanding structural and role-related aspects of collaboration. Manuals often contained organizational charts that clarified the composition of team structures of agile and non-agile units. They also described newly established roles, particularly those designed to bridge the interfaces between agile and non-agile units.

We used three types of archival data to enrich our contextual understanding of the cases, whereas interviews and observational data served as the primary sources for identifying differences in temporal structures, tensions, and practices for resolving those tensions. Following this presentation of our data sources and data collection procedures, the next chapter will outline the data analysis techniques employed.

4.5. Data analysis

The data analysis is structured into four steps, which are outlined in this chapter. In the first step, temporal tensions were identified across all cases and compiled into a tension inventory (Chapter 4.5.1). In the second step, resolution practices applied to resolve each

tension were identified and examined (Chapter 4.5.2). These two steps enabled the consistent labeling of similar tensions and practices across different cases. In the third step, the previously identified tensions and practices, initially considered in relative isolation, were integrated into path models for each case (Chapter 4.5.3). In a final step, a cross-case comparison was conducted to identify similarities and differences between the path models (Chapter 4.5.4)

4.5.1. Step 1: Identifying temporal tensions

As a first step in the data analysis, I openly coded the collected empirical data to identify the tensions shaping the collaboration between agile and non-agile organizational units. Following the methodological approach of Gioia and colleagues (2013), I developed first-order concepts for the tensions identified during this open coding stage, whether reported in interviews with members of agile or non-agile units, observed in participatory observations, or derived from archival material.

As this dissertation aims to investigate how actors with divergent temporal structures coordinate effectively, it focuses on tensions arising from differences in the temporal structures of agile and non-agile units (hereinafter referred to as temporal tensions). However, not every tension identified in the initial open coding constitutes a temporal tension. Many tensions arise from other causes, such as resource scarcity or power differences. To distinguish temporal from non-temporal tensions, I further analyzed the first-order concepts to assess whether the tensions they depict originate from differences in temporal structures. For this purpose, I drew on the temporal dimensions and concepts established in the temporality literature, which are commonly used to describe temporal structures (Ancona, Okhuysen, et al., 2001; Ancona & Chong, 1996; Ballard & Seibold, 2003; Bluedorn, 2002; Rowell et al., 2016) and were introduced in Chapter 2. By iteratively engaging with this literature and the empirical data, I identified three aspects of temporal structures that differed between agile and non-agile units and were found to underlie various tensions experienced in their collaboration. These three aspects are (1) differing temporal depths which translate into divergent planning cycles in agile and non-agile units (Ancona & Chong, 1996; Bluedorn, 2002; Das, 1987), (2) differing levels of time urgency, leading to different rhythms at which agile and non-agile units prefer to

deliver their work outputs (Ancona, Okhuysen, et al., 2001; Ancona & Chong, 1996; Conte et al., 1995), and (3) differing conceptions of clock time and event time, which influence how agile and non-agile units respond to deadlines (Bluedorn, 2002; Dougherty et al., 2013; Reinecke & Ansari, 2015).

I analyzed the first-order concepts for similarities and differences to identify overarching clusters of tensions. Through iterative comparisons of the first-order concepts, considering both temporal and non-temporal origins, I ensured that similar tensions were consistently mapped to the same second-order themes, thereby enabling a cross-case consistent understanding of the tensions experienced in collaborations between agile and non-agile units. Table 18 in the appendix provides an overview of this initial coding step, presenting all first-order concepts and illustrating how they were aggregated into second-order themes, with temporal tensions highlighted in bold letters.

After establishing the core temporal tensions, I focused on gaining a deeper understanding of them. For each temporal tension, I first coded which actors were affected by its deleterious impact. I distinguished between agile and non-agile units and, following the classification of non-agile units introduced in Chapter 4.3.1, further subdivided the non-agile units with which agile units needed to coordinate to complete their tasks into (1) governance, (2) non-agile units providing input to the agile units (input providers), (3) non-agile units providing the necessary infrastructure for deploying work outcomes (release management), and (4) non-agile units using the agile units' work outcomes for their daily activities (end users).

Moreover, for each temporal tension, I coded when it occurred in the collaboration between agile and non-agile units, drawing on the concept of "temporal bracketing" (Langley, 1999, p. 703). Temporal bracketing is commonly used in process research to divide a continuous timeline into distinct segments, which do not necessarily need to be theoretically significant but imply discontinuities or changes in activities or dynamics. Defining temporal brackets posed a challenge because agile and non-agile units structured their work processes differently. Agile units followed the cyclical processes characteristic of agile practices, in which minimum viable products are developed incrementally in short, recurring cycles and delivered in rapid rhythms (Schwaber & Sutherland, 2020). In contrast, most non-agile units' work processes (except for non-agile end-user units with customer service, client advisory, or sales responsibilities) were linear, consisting of

several sequential phases, each lasting several weeks, with solutions delivered only at the end of the process. Regardless of the divergent temporal structuring of their work processes, the data revealed that activity stages contained within agile cycles and non-agile phases were similar. Both the cyclical and linear work processes began with a planning stage, in which agile and non-agile units planned their activities and secured the resources needed to accomplish them. This planning stage was followed by a task execution stage, in which they carried out the operational activities they had previously planned. For agile units, this primarily involved conceptualizing and developing software and restructuring solutions. For non-agile units, activities during the task execution stage were more diverse. They ranged from software development in input-providing IT units, to monitoring software development and preparing stability tests in release management units, or compiling regulatory reports in the end user unit in case [C]. The task execution stage was followed by a delivery stage, in which agile and non-agile units made their work outcomes available to their end users. This stage encompasses both testing and approval procedures, designed to ensure the functionality and stability of outcomes from agile or non-agile task execution stages, and the actual release of software solutions, either developed by non-agile input providing units for agile units or by agile units for non-agile end-user units, as well as the handover of restructuring solutions. While these processes were observable in both agile and non-agile units, they differed in their temporal structures, particularly in duration and timing. Non-agile units typically carried out the three process stages sequentially over a time horizon of several months. In contrast, agile units iterated multiple times between planning, task execution, and, in some cases, delivery within the same period. Regardless of this temporal misalignment, I used planning, task execution, and delivery as temporal brackets and coded each tension according to the stage in which it emerged.

The result of the first analytical step is an inventory of all temporal tensions experienced by agile and non-agile units due to their divergent temporal structures (see Table 9). The temporal tension inventory provides information on which aspects of temporal differences the various tensions can be attributed to, which actors experience them, and at what points in time. Moreover, while temporal tensions were initially systematized independently of the case in which they were observed, the temporal tension inventory now presents a disaggregated view, indicating in which of the eight cases the respective temporal tensions occurred.

Table 9: Temporal tension inventory

Temporal tensions:	**Underlying differences in temporal structures:**	**Perceiving actor:**	**Process stages:**	**Observed in cases:**							
				[A1]	**[A2]**	**[B1]**	**[B2]**	**[B3]**	**[C]**	**[D]**	**[E]**
communication effort to negotiate new temporal patterns	▪ time urgency & different release rhythms	▪ agile units ▪ release management	▪ delivery						●	●	
coordination gaps due to asynchronous planning	▪ temporal depth & different planning cycles	▪ agile units ▪ input providers	▪ planning		●						
deadline-induced **quality concerns**	▪ clock-time- and event-time-based temporal patterns	▪ agile units ▪ release management	▪ task execution	●					●	●	
delayed task completion due to planning tensions	▪ temporal depth & different planning cycles	▪ agile units	▪ task execution	●	●	●	●	●	●	●	●
delayed delivery due to differences in temporal structures	▪ temporal depth & different planning cycles ▪ clock- time and event-time-based temporal patterns	▪ agile units ▪ end users ▪ governance	▪ delivery	●		●	●	●		●	
high costs for accelerating release rhythms	▪ time-urgency & different release rhythms	▪ agile units ▪ release management	▪ delivery	●	●				●	●	
high efforts for creating long-term plans	▪ temporal depth & different planning cycles	▪ agile units	▪ planning						●	●	
high workload following from planning differences	▪ temporal depth & different planning cycles	▪ agile units	▪ task execution		●						
high workload due to temporal constraints	▪ clock- time and event-time-based temporal patterns	▪ agile units ▪ input providers	▪ task execution	●	●		●	●	●	●	●
high workload from quality versus speed tensions	▪ time urgency & different release rhythms	▪ agile units ▪ release management units	▪ task execution ▪ delivery						●	●	

Temporal tensions:	**Underlying differences in temporal structures:**	**Perceiving actor:**	**Process stages:**	**Observed in cases:**							
				[A1]	**[A2]**	**[B1]**	**[B2]**	**[B3]**	**[C]**	**[D]**	**[E]**
limited flexibility due to deadline rigidities	▪ clock- time and event-time-based temporal patterns	▪ agile units	▪ task execution	●	●		●	●	●	●	●
limited flexibility due to rigid planning	▪ temporal depth & limited planning cycles	▪ agile units	▪ task execution		●	●	●	●	●	●	
limited planning ability due to misaligned planning practices	▪ temporal depth & limited planning cycles ▪ clock- time and event-time-based temporal patterns	▪ agile units ▪ end users ▪ input providers	▪ planning ▪ task execution	●	●	●	●	●	●	●	●
limited speed due to low-frequency releases	▪ time urgency & different release rhythms	▪ agile units ▪ governance	▪ delivery		●				●	●	●
non-value adding effort from temporal misalignment	▪ temporal depth & different planning cycles ▪ clock- time and event-time-based temporal patterns	▪ agile units ▪ input providers	▪ planning ▪ task execution	●	●	●	●		●	●	
schedule disruptions from temporal misalignments	▪ temporal depth & different planning cycles	▪ agile units ▪ input providers	▪ task execution	●	●	●	●	●	●	●	
speed-induced quality concerns	▪ time urgency & different release rhythms	▪ release management ▪ governance	▪ delivery	●	●				●	●	●
temporal **coordination effort**	▪ temporal depth & different planning cycles ▪ clock- time and event-time-based temporal patterns .	▪ agile units ▪ input providers ▪ release management ▪ governance	▪ planning ▪ task execution ▪	●		●			●		
temporal **limitations of responsiveness**	▪ temporal depth & different planning cycles ▪ clock- time and event-time-based temporal patterns	▪ agile units ▪ end users ▪ input providers	▪ task execution		●	●		●	●	●	●
time-related **dissatisfaction**	▪ temporal depth & different planning cycles	▪ end users ▪ input providers	▪ planning ▪ delivery	●	●				●		

Temporal tensions:	**Underlying differences in temporal structures:**	**Perceiving actor:**	**Process stages:**	**Observed in cases:**							
				[A1]	**[A2]**	**[B1]**	**[B2]**	**[B3]**	**[C]**	**[D]**	**[E]**
uncertainty about resource allocation	▪ temporal depth & different planning cycles	▪ agile units	▪ planning	●	●	●	●	●	●	●	●
uncertainty about timely delivery	▪ clock- time and event-time-based temporal patterns	▪ agile units ▪ governance	▪ task execution	●	●		●	●	●	●	●
uncertainty from stakeholder communication dynamics	▪ time urgency & different release rhythms	▪ agile units ▪ end users	▪ delivery								●
uneven capacity utilization over time	▪ temporal depth & different planning cycles	▪ agile units ▪ input providers	▪ task execution			●					
waiting times due to planning misalignments	▪ temporal depth & different planning cycles	▪ agile units	▪ task execution	●	●	●	●	●	●	●	●

4.5.2. Step 2: Coding and systemizing temporal practices

Following the development of a cross-case inventory of temporal tensions, detailing when and for whom temporal tensions emerged in each case, I examined the temporal practices through which agile and non-agile units sought to address these tensions. To this end, I analyzed each temporal tension in isolation and traced, in the empirical data, how agile and non-agile units across all eight cases responded to them. As in the previous stage, I drew on the analytical framework of Gioia and colleagues (2013), conducting open coding to translate all documented temporal practices into first-order concepts. I then grouped these first-order concepts into second-order themes, which served to highlight similarities and differences among the temporal practices. Table 19 in the appendix presents the results of this coding process and lists all temporal tensions that agile or non-agile units sought to resolve through temporal practices.

The first coding step revealed that different tensions were sometimes addressed through similar temporal practices. For example, agile units used harmonizing practices both to minimize waiting times and to reduce uncertainty about timely delivery. To establish a consistent and systematic labeling of temporal practices across the different temporal tensions, I moved from examining each temporal tension in isolation to comparing the second-order themes across tensions. In line with the idea of axial coding (Strauss & Corbin, 1998), I compared the second-order themes across temporal tensions and further consolidated them into aggregate dimensions (Gioia et al., 2013).

To consolidate second-order themes into distinct aggregate dimensions, I first coded whether a practice aimed to address coordination challenges between agile and non-agile units. Accordingly, I began analyzing temporal practices with respect to their coordinative function. Drawing on the information processing view (Galbraith, 1973), I coded each practice according to whether it reduced coordination requirements between agile and non-agile units or enhanced the coordination capacity of at least one of them. In doing so, I occasionally revisited the initial mapping of first-order concepts onto second-order themes, particularly when a second-order theme contained practices that both reduced coordination requirements and enhanced coordination capacity. In such

cases, I reassigned first-order concepts to other, or in some instances, newly defined, second-order themes to ensure a clear differentiation by coordinative function. Once each second-order theme was unambiguously categorized, I then investigated how reductions in coordination requirements and increases in coordination capacity were actually achieved.

I identified two mechanisms that reduced coordination requirements. The first, rooted in the notion of standardization (Thompson, 1967; van de Ven et al., 1976), is synchronization, which reduces the need for active coordination by allowing actors to implicitly rely on shared timing, pacing, and sequencing (Orlikowski & Yates, 2002). In line with the synchronizing practices outlined in Chapter 3.1, I derived two aggregate dimensions from the standardization mechanism inherent in synchronization: *entraining* and *developing new temporal patterns*. The second mechanism, according to Galbraith (1973), reduces coordination requirements by minimizing interdependencies between actors, for example, by creating self-contained tasks or building slack resources. From this mechanism, I derived two further aggregate dimensions. When interdependencies between agile and non-agile units were reduced only during certain stages of a work process, combined with ex-ante coordination in earlier stages, this constituted *buffering*. When the reduction of interdependence was not confined to a specific process stage but persisted in a way that precluded ex-ante coordination, it was categorized as *isolating*.

Capacity-increasing mechanisms were exemplified by intensified communication between agile and non-agile units, rooted in the notion of mutual adjustment (Thompson, 1967; van de Ven et al., 1976). Building on the capacity-increasing mechanism of mutual adjustment, and with reference to the temporal coordination practices presented in Chapter 3, I derived two additional aggregate dimensions. First, *harmonizing* practices encompassed the intensified coordination and communication efforts between agile and non-agile units to enable coordination based on synchronous moments without altering their temporal structures. Second, *brokering* practices increased coordination capacity by specifically assigning responsibility for coordination between agile and non-agile units to designated individuals (Halevy et al., 2019; Mell et al., 2021).

From the empirical data, three practices emerged that neither reduced coordination requirements nor enhanced coordination capacity. Agile units applied these practices to restore losses of their agile capabilities without tackling underlying coordination

problems. First, and in contrast to the previously described dimensions of temporal practices, agile units occasionally dissolved the standardization mechanism inherent in synchronization. These temporal practices, which effectively increased the active coordination needs between agile and non-agile units, constitute the aggregate dimension of *detraining*. Second, agile units sought to restore their agile capabilities by flexibilizing time, workload, and team size. Unlike detraining, *flexibilizing* did not increase active coordination needs between agile and non-agile units, which is why I derived a separate aggregate dimension from it. Third, one agile unit substituted the required resources with those available at short notice. While this practice may appear similar to isolating, it is distinct. Isolating practices eliminate interdependencies, whereas the agile unit that substituted resources continue to actively coordinate them. I therefore derived a separate aggregate dimension, which I labeled *substituting*. The final, cross-tension data structure derived from analyzing and coding the temporal practices that actors used to respond to temporal tensions is illustrated in Figure 4.

4.5.3. Step 3: Developing patterns of tensions and practices

The third analytical step focused on examining the interplay between temporal tensions and temporal practices, which produced different patterns of coordination across divergent temporal structures in each case. To understand how temporal tensions and practices are related, I revisited the data to gain further insights into the consequences of temporal practices. From the preceding second step, I already knew the intended consequences of the practices, that is, the specific tensions that agile and non-agile units aimed to resolve by employing them. During the renewed evaluation of practices, I focused on the unintended consequences of each practice, namely the positive and negative effects that went beyond the intended tensions. This evaluation revealed that many practices had the potential to recreate temporal tensions. Table 10 summarizes the intended and unintended consequences of temporal practices in each case.

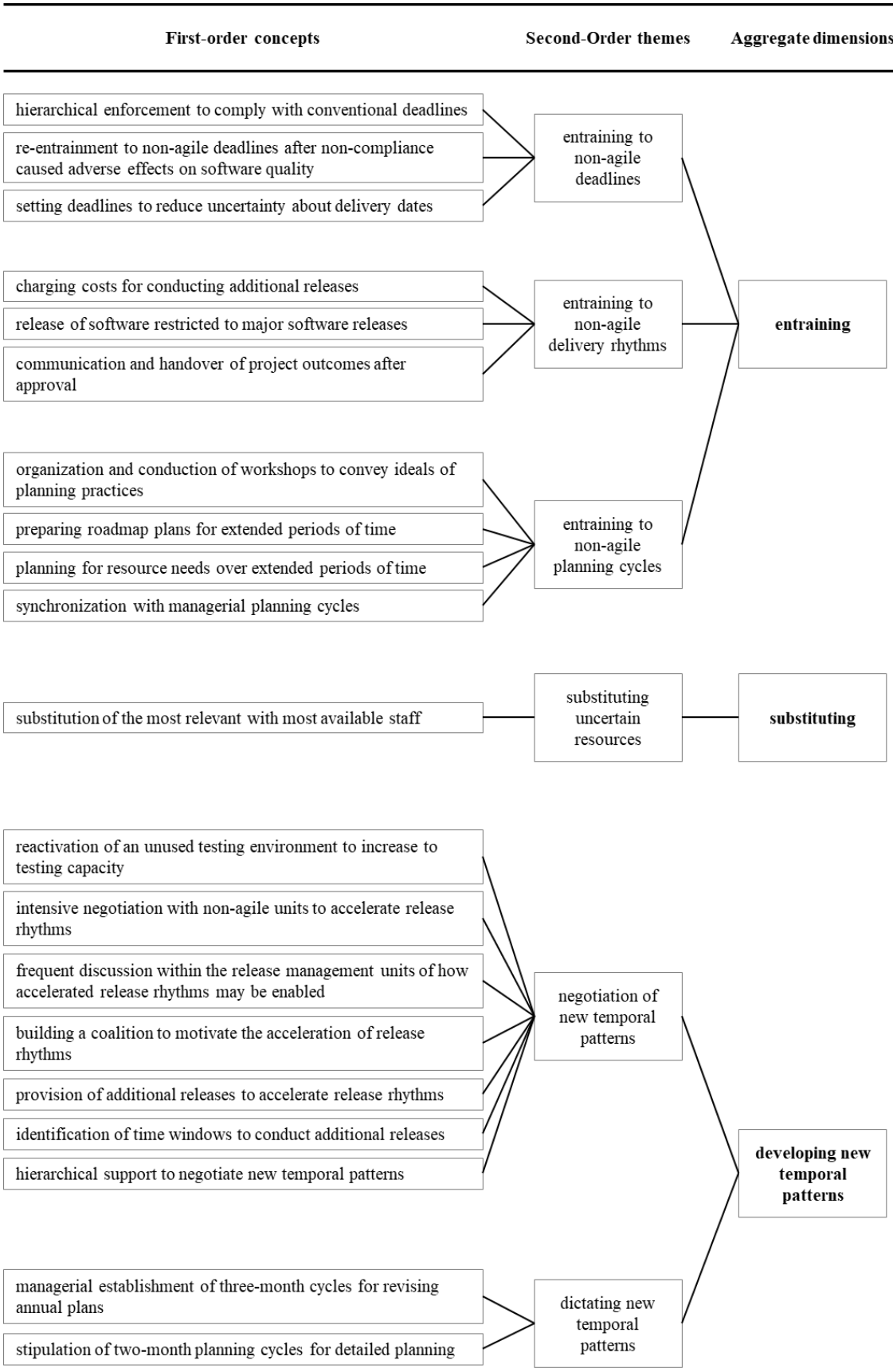

Figure 4: Data structure of temporal practices across cases and tensions

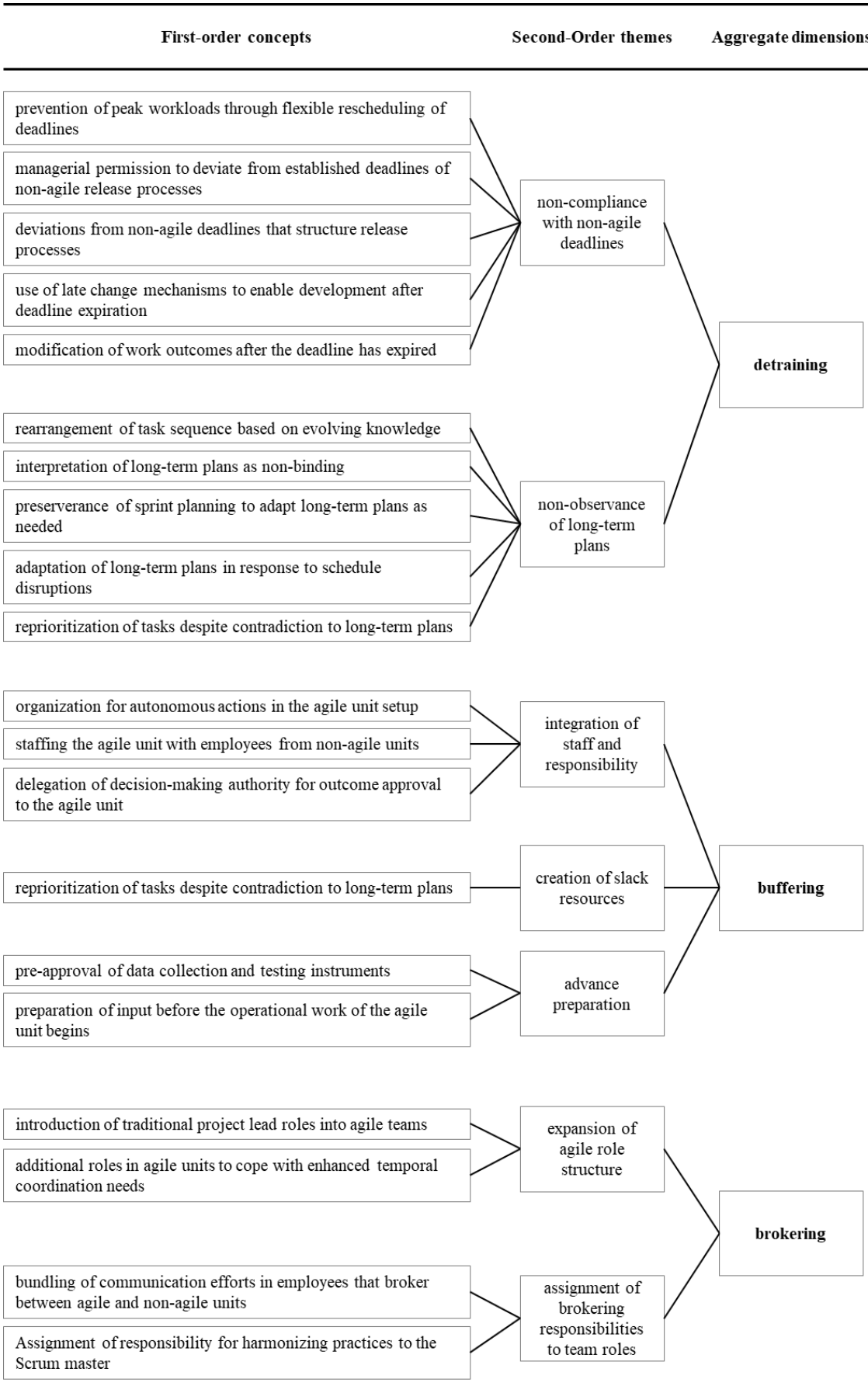

Figure 4: Data structure of temporal practices across cases and tensions (continuation)

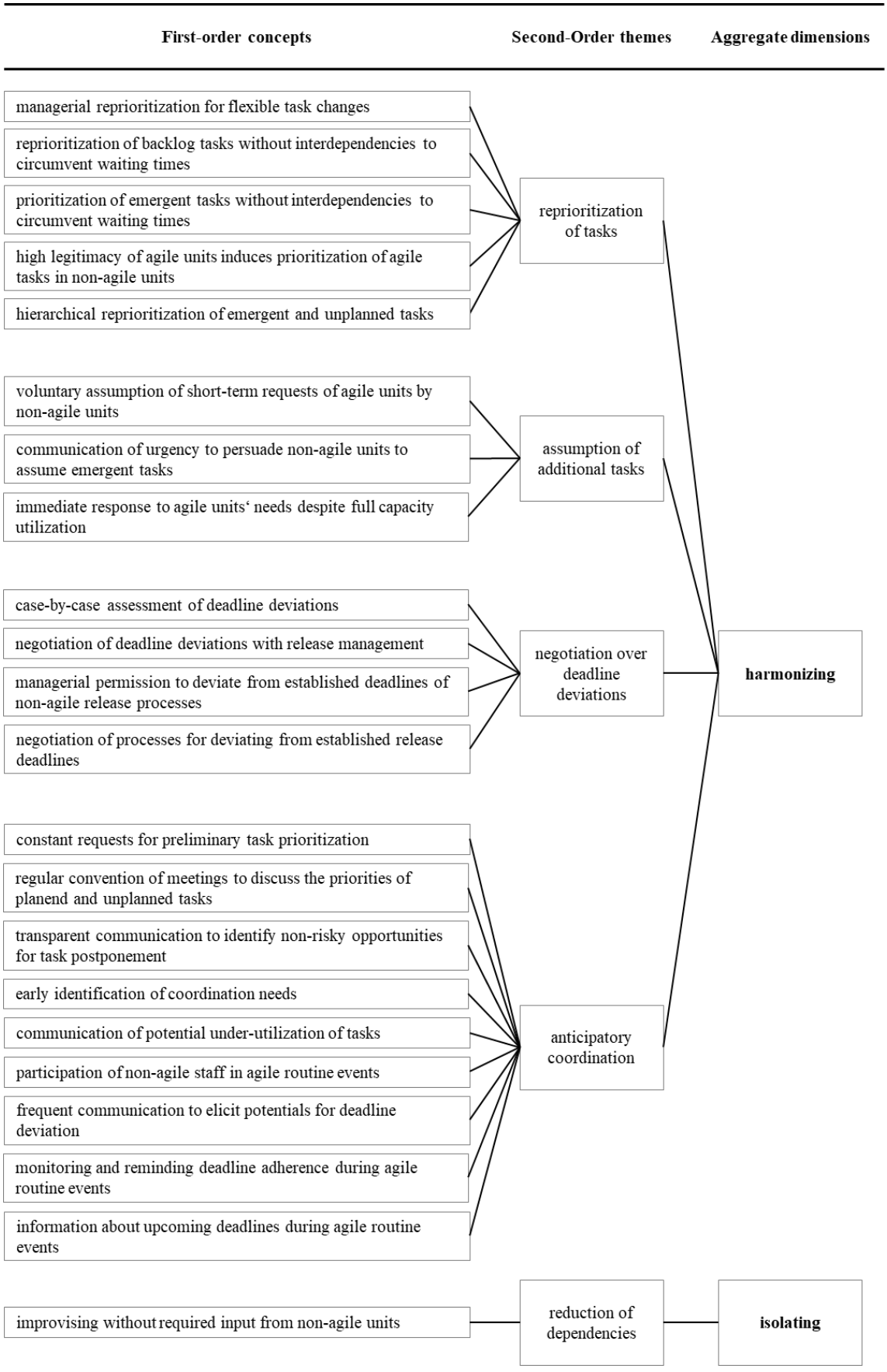

Figure 4: Data structure of temporal practices across cases and tensions (continuation)

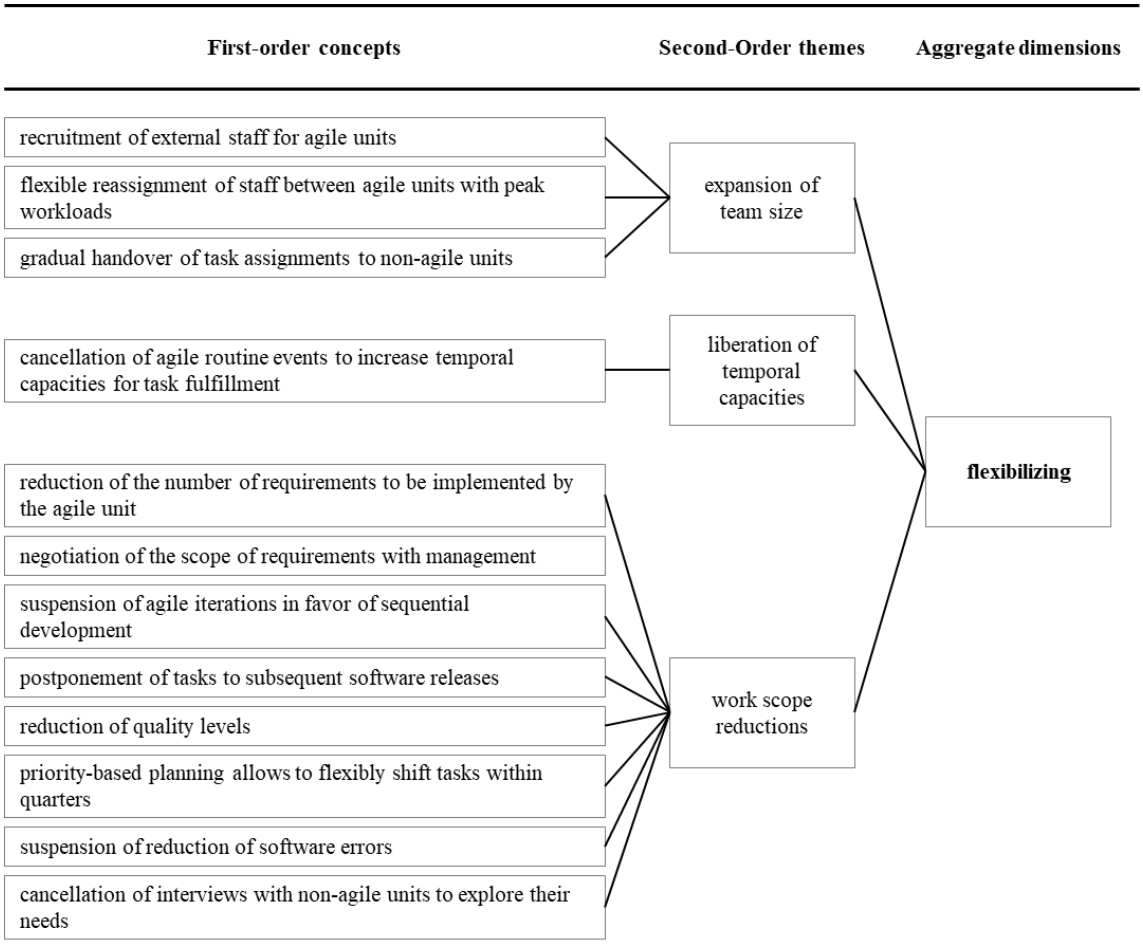

Figure 4: Data structure of temporal practices across cases and tensions (continuation)

Drawing on the intended and unintended consequences, as well as the information documented in the temporal tension inventory regarding when temporal tensions arose during the work process and which temporal differences caused them, I was able to reconceptualize the patterns of interplay between temporal tensions and temporal practices across all eight cases. I began by identifying the tensions that resulted directly from the three temporal differences. Different planning cycles resulted in uncertainties regarding resource allocation in agile units. Differences in clock- and event-time patterns, which shaped actors' relation to deadlines, generated uncertainty over the timely completion of work outcomes, affecting non-agile units. Similarly, differing release rhythms led to quality concerns in non-agile units. I then mapped practices to the tensions they were intended to resolve. In doing so, I also recorded how their enactment influenced the temporal structures of agile units. The unintended consequences of temporal practices allowed me to identify the new temporal tensions they triggered. Engaging in a recursive process, I assigned the new temporal tensions to the respective stage of the work process,

Table 10: Intended and unintended consequences of temporal practices

Intended consequences	**Unintended consequences**	**Cases**							
Resolves …	***Creates …***	**[A1]**	**[A2]**	**[B1]**	**[B2]**	**[B3]**	**[C]**	**[D]**	**[E]**
	Entraining								
uncertainty of resource allocation		●	●	●	●	●	●	●	●
limited planning ability		●	●	●	●	●	●	●	●
high workload from planning differences			●						
	high effort of long-term planning						●	●	
	limited flexibility due to planning rigidities		●	●	●	●	●	●	
	temporal limitations of responsiveness		●			●	●	●	●
	high planning-induced workloads		●						
	time-related dissatisfaction		●						
uncertainty about timely delivery of outcomes		●	●		●	●		●	●
deadline-induced quality concerns								●	
	limited flexibility due to deadline rigidities	●	●		●	●			●
	high workload due to temporal constraints	●	●		●	●			●
speed-induced quality concerns		●	●				●	●	●
uncertainty from stakeholder communication dynamics									●
	limited speed		●				●	●	●
	Detraining								
handling schedule disruptions		●		●	●	●	●	●	
limited flexibility due to planning rigidities		●		●	●	●	●	●	
	uncertainty in resource allocation								●
	limited planning ability			●			●	●	
	waiting times	●	●	●	●	●	●	●	
temporal limitations of responsiveness							●	●	
	uncertainty about timely delivery of outcomes						●	●	

Intended consequences *Resolves …*	**Unintended consequences** *Creates …*	**Cases** [A1]	[A2]	[B1]	[B2]	[B3]	[C]	[D]	[E]
	Substituting								
uncertainty of resource allocation									●
	competence- and solution-related quality concerns								●
	limited operational independence								●
	waiting times								●
	Developing new temporal patterns								
	limited flexibility due to planning rigidities			●	●	●			
	handling schedule disruptions			●	●	●			
limited speed							●	●	
high workload from quality versus speed tensions							●	●	
speed-induced quality concerns							●	●	
	high communication effort to negotiate new temporal patterns						●	●	
	high costs for accelerating release rhythms	●	●				●	●	
	Harmonizing								
waiting times		●	●	●	●	●	●	●	●
uncertainty about timely delivery of outcomes							●	●	
	temporal coordination effort	●		●			●		
	delayed delivery	●							
	time-related dissatisfaction	●							
	Brokering								
temporal coordination effort		●					●		

Intended consequences	**Unintended consequences**	**Cases**							
Resolves …	***Creates …***	**[A1]**	**[A2]**	**[B1]**	**[B2]**	**[B3]**	**[C]**	**[D]**	**[E]**
	Buffering								
waiting times			●		●	●		●	●
	high workload from resource and capacity constraints							●	●
	emotional strain							●	
	coordination gaps		●						
	non-value adding effort from temporal misalignment		●						
	Flexibilizing								
limited flexibility due to deadline rigidities		●	●		●	●			●
high workload due to temporal constraints		●	●		●	●			●
	systematic deprioritization		●						●
	temporal limitations of responsiveness		●			●			●
	intra-team collaboration inefficiencies				●				
	meeting-induced communication overload	●			●				

as documented in the temporal tension inventory, examined the practices intended to resolve them, noting the potential changes they made to agile temporal structures as well as the new temporal tensions they triggered. This approach enabled me to derive patterns that captured both the dynamics of coordination processes and the resulting changes in the temporal structures of agile units. For easier comparison in the subsequent step, I illustrated these patterns graphically.

4.5.4. Step 4: Comparing patterns across cases

In the final analytical step, I compared the previously reconceptualized patterns of temporal coordination processes and the resulting changes in temporal structures across the eight cases. I examined the patterns arising from each of the three key temporal differences (different planning cycles, different use of deadlines rooted in divergent clock- and event-time conception, different release rhythms), each in isolation. A systematic comparison of the eight cases, guided by the replication logic of the multiple case study design (Eisenhardt & Graebner, 2007), allowed for identifying differences and similarities between the patterns of the individual cases.

Based on their similarities and differences, I consolidated the within-case process patterns into several overarching patterns. With respect to different planning cycles, I identified four distinct patterns. Two of these patterns illustrate coordination processes in which various temporal structuring practices (entraining, detraining, and developing new temporal patterns) are combined to coordinate across divergent temporal structures. In one pattern, coordination relies on entraining and detraining practices, whereas in the other, these practices are supplemented by developing new temporal patterns. The dynamic transitions between temporal structuring practices result in recurring changes in the temporal structures of agile units. The two remaining patterns each employ only a single temporal structuring practice (either entraining or detraining), resulting in stable temporal structures. Regarding the differences in clock-time and event-time conceptions that influence how deadlines are perceived and used, I identified two patterns. One of them is based on persistent entraining, whose unintended negative consequences are compensated for by flexibilizing patterns. The other involves persistent detraining, whose

drawbacks are offset by harmonizing practices. Lastly, with respect to differing release rhythms, two patterns emerge. The first pattern shows how agile units entrain to the release rhythms of non-agile units. The second pattern extends the first, showing that some agile units translate the tensions arising from this entraining into new temporal patterns.

Chapter 4 provided insights into the methodological approaches underlying this dissertation. It aimed to justify both the research design and the empirical research context, introduced the case studies selected for analysis, illuminated the data collection process, and described the techniques used for data analysis. The next chapter presents the findings, focusing on how agile and non-agile units coordinate their interdependent activities across three key differences in their temporal structures.

5. Coordinating across divergent temporal structures in agile and non-agile units

How do agile and non-agile organizational units coordinate their interdependent activities despite divergent temporal structures? This question lies at the heart of Chapter 5, which presents the empirical findings of this dissertation. Diverging temporal structures create tensions both within and between agile and non-agile units, tensions that must be actively managed to enable effective collaboration and coordination. Building on the conceptualization of temporal structures outlined in Chapter 2.2, the empirical data reveal three key temporal differences between agile and non-agile units. Each of these differences gives rise to specific temporal tensions, which in turn require targeted coordination practices to address.

The findings are structured around three key temporal differences. Chapter 5.1 examines the divergent temporal depths of agile and non-agile units, which lead to differing cycles for planning activities and allocating resources. These misaligned planning cycles create resource allocation tensions that can disrupt agile workflows. Such tensions are mitigated through combinations of entraining, detraining, establishing new temporal patterns, and deploying coordination practices that enable both agile and non-agile units to maintain their distinct temporal structures. Chapter 5.2 analyzes how divergent conceptions of clock time and event time in agile and non-agile units give rise to contestations over deadlines. In particular, deadlines imposed on agile units by non-agile actors may constrain their ability to respond flexibly to customer needs and change, undermining core agile capabilities. The chapter identifies three response patterns through which agile units manage these tensions, each reflecting different degrees of acceptance of resistance toward externally imposed deadlines and the corresponding loss of agile capabilities. Chapter 5.3 explores the differing levels of time urgency across agile and non-agile units, which translate into divergent rhythms for delivering outcomes to end users. These differences give rise to conflicting imperatives: the need to ensure quality and stability versus the demand for speed and responsiveness. While many agile units entrain to the slower release rhythms of non-agile units, thereby accepting limitations of their speed, others navigate the speed-stability tradeoff by negotiating new temporal patterns.

5.1. Divergent planning cycles: Navigating tensions of resource allocation

Collaboration and coordination between agile and non-agile units are shaped by their divergent planning cycles, which define the timeframes over which each unit plans tasks and allocates resources. As visible temporal patterns, planning cycles are rooted in the differing temporal depths, or planning horizons, embedded in agile and non-agile work practices.

As outlined in Chapter 2.3, agile work practices are characterized by a near future-time orientation, reflected in low temporal depth and short planning horizons, which manifest in brief planning cycles. The eight cases empirically support this conceptualization. In all cases, agile units operate within short sprint cycles, a defining feature of the Scrum method, typically lasting two to three weeks. Table 11 summarizes the sprint cycle duration across all agile units, supported by interview data. These sprint cycles delineate the concrete planning horizons of agile units. Although the product backlog usually contains tasks (referred to as user stories) that extend beyond a single sprint, agile units only decide which tasks to implement in the upcoming sprint. Thus, across all cases, agile units consistently apply two- to three-week planning cycles to determine task priorities and resource needs, in line with the principles of the Scrum method. By contrast, non-agile units operate with significantly longer planning cycles. Table 11 also presents the durations of the dominant non-agile planning cycles to facilitate direct comparison. It is important to note that planning cycles among non-agile units are not necessarily homogeneous. Different non-agile units may operate on different planning cycles. The planning cycles highlighted in Table 11 correspond to those non-agile units that provide critical resources for agile task accomplishment.

In cases [A1] and [A2], the biannual planning cycles of the regulatory compliance unit, responsible for setting regulatory changes that the agile units had to implement every six months, were dominant. In cases [B1], [B2], [B3], and [E], annual planning cycles prevailed, through which management allocated resources to both agile and non-agile units across the organization. In case [C], the agile unit primarily relied on developer capacity provided by a non-agile data warehouse unit that planned its activities and

resources over a six-month time horizon. In case [D], corporate planning cycles used for resource allocation also operated on a six-month schedule.

Table 11: Agile and non-agile planning cycles

Case	Duration of agile sprint cycles		Duration of (dominant) non-agile planning cycles	
[A1]	2 weeks	We work in two-week sprints, and sprint planning takes place every two weeks as well. (I.A1.3)	6 months	At the moment, we still think in release cycles that are very waterfall-driven and typically last six months. (I.A.1)
[A2]	3 weeks	We have adopted Scrum: three-week sprint cycles (I.A2.1)	6 months	The [non-agile units] need the request before we even know what we'll require. (I.A2.1)
[B1]	2 weeks	In our unit, we work in two-week [sprint] cycles. (I.B1.5)	12 months	In our company [B], we conduct annual planning. It essentially involves asking: What's on the agenda? Then I have to say, 'I want to implement this and that', and those become our tasks. (I.B1.1)
[B2]	3 weeks	So, we have three-week sprints. They work quite well for us. (I.B2.3)	12 months	The rest of [B] isn't agile at all. They plan the entire next year in advance – in August! (I.B2.1)
[B3]	2 weeks	We work in two-week sprints, so our cycles are relatively short (I.B3.3)	12 months	We still follow annual cycles, using traditional project planning methods. (I.B3.1)
[C]	2 weeks	Every two weeks, we hold planning sessions, retrospectives, reviews, and other Scrum ceremonies. During the planning sessions, tasks for the upcoming sprint are defined. (I.C.10)	6 months	[Because] we're planning six months in advance, it would be helpful to know what the [agile unit] aims to achieve over the next six months. (I.C.9)
[D]	3 weeks	We plan our tasks within our three-week sprints. (I.D.3)	6 months	A planning period always covers six months. Typically, the last couple of months I planned in the previous cycle overlap with the first two months of the next planning cycle. As a result, I end up with about three and a half planning cycles per year. (I.D.2)
[E]	2 weeks	We have these two-week sprints. (I.E.3)	12 months	We have an annual plan for projects. (I.E.9)

Differences in planning cycles between agile and non-agile units can disrupt their respective work processes. Tensions arise as early as the planning stage, where conflicting expectations emerge regarding both the timing and horizon of planning activities. The misalignment between agile units' short, iterative planning cycles and non-agile units' longer planning cycles hampers joint task and resource planning, both vital for coordinated action and effective collaboration. These tensions are not limited to the

planning stage but often persist through task execution and delivery, where agile and non-agile units carry out operational tasks and hand over outcomes to internal or external customers.

Across the eight cases, four distinct response patterns emerge from the differing temporal depths and planning cycles between agile and non-agile units. The following chapters provide detailed accounts of each. Chapter 5.1.1 describes a dynamic pattern of entraining and detraining practices observed in cases [A1], [C], and [D]. Chapter 5.1.2 examines how these dynamics are complemented by the establishment of new temporal patterns in cases [B1], [B2], and [B3]. Chapter 5.1.3 focuses on case [A2], which exhibits a pattern of persistent entraining. Chapter 5.1.4 explores case [E], which demonstrates a persistent pattern of sustaining divergent planning cycles. Chapter 5.1.5 compares and contrasts these four response patterns to synthesize the findings on divergent temporal depths and planning cycles.

5.1.1. Pattern 1: Entraining-detraining dynamics

The first pattern reveals a dynamic interplay between entraining and detraining practices of agile units, adopted to address tensions that originated from agile and non-agile units' divergent planning cycles. As illustrated in Figure 5, agile units initially responded to the ***uncertainty about resource allocation*** by ***entraining*** to the longer planning cycles of non-agile units. This synchronization of their planning cycles facilitated resource allocation for agile units, thereby reducing uncertainty. Over time, however, the extended planning cycles ***limited agile units' flexibility***, prompting them to ***detrain*** and revert to their shorter, sprint-based planning cycles. This shift recreated temporal differences, giving rise to ***waiting times*** as non-agile units continued to follow longer-cycled plans. To mitigate waiting times, agile units applied two sustaining practices, ***buffering and harmonizing***, that helped reduce uncertainty in resource provisioning that underlay waiting times. While buffering practices were fully capable of mitigating waiting times, harmonizing practices often only partially reduced them, resulting in sporadic ***delays*** in solution delivery.

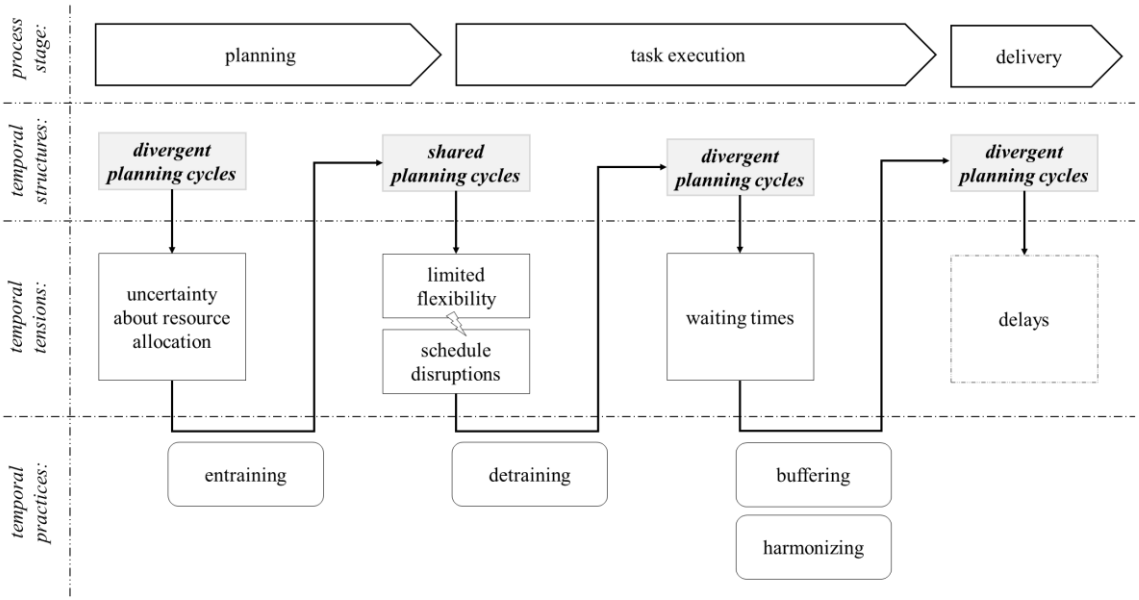

Figure 5: Entraining-detraining dynamics

Cases [A1], [C], and [D], which constitute the first pattern, all revolve around agile software development teams working with diverse non-agile actors, both internal and external to their organizations. Across all cases, agile units depended on resources supplied by non-agile actors. Case [A1] describes an agile IT team, tasked with creating sales software for sales agents, that relied on software developers contracted through an external IT service provider. In case [C], an agile project team developing software for regulatory reporting at a banking company relied heavily on the developer capacity of a non-agile data warehouse unit, which provided the technical interfaces required to access data and figures for reporting activities. Case [D] involved an agile project team that developed liability insurance administration software for non-agile insurance line units. To execute software development activities, the agile project team drew on staff seconded from the insurance line and IT units.

Across all three cases, divergences of agile and non-agile planning cycles created uncertainty about resource allocation for agile units that depended on the resources provided by non-agile actors. Whereas non-agile actors scheduled tasks and resources over six-month horizons, agile teams operated in two- to three-week sprint cycles, specifying their tasks and resource needs with far shorter lead times. With their resources locked into long-term task commitments, non-agile actors were often unable to accommodate agile teams' short-term resource requests. As a result, agile teams faced significant uncertainty over the availability and provision of required resources.

In case [A1], the agile team consisted largely of external software developers contracted through an IT service provider. Although theoretically feasible, engaging external software developers for two-week planning cycles introduced substantial uncertainty for the agile team. Short-term contracting would not only have required substantial effort to (re-)negotiate contracts but also carried the risk that developers with the necessary expertise might not be available on short notice, as they could already have accepted assignments with other companies to reduce their own uncertainty.

> **Case [A1]:** All our developers are external contractors, and so are all our testers. The internal team consists of the product owners, me as the Scrum master, and the business analysts. These roles are filled by permanent employees. Yeah … And because we rely on external contractors for specialist skills, the team composition changes from one release to the next. Depending on the focus of each release, we decide which external people to bring in. (I.A1.3)

> **Case [A1]:** We work with quite a few external contractors, and they want to know what their role will be and how many people they should plan to assign. (I.A1.2)

In case [C], the non-agile data warehouse unit scheduled tasks and allocated resources according to six-month planning cycles, which conflicted with the sprint-based planning of the agile units. This misalignment often left the agile team uncertain about whether the data warehouse unit was able or willing to deviate from its long-term commitments to deliver required interfaces on short notice.

> **Case [C]:** So, in theory, the specification and scope need to be communicated to the [data warehouse unit] six months in advance, which is often unrealistic. (…) Then there's the agile world, where I say, 'I can't tell you what I'll need in six months, but I need something concrete from you right now.' Our colleagues aren't on board with that. And to be honest, we're not really set up for fast-paced rhythms either. That's where things usually get stuck. (I.C.1)

Finally, in case [D], the agile team also experienced uncertainty about resource allocation if they communicated their resource needs within the rhythms of their three-week sprint cycles, which diverged from non-agile six-month horizons. The agile team recognized that deviating from non-agile planning horizons could result in the non-provision of resources and yet cause tremendous problems for the agile team to accomplish its tasks and goals.

> **Case [D]:** And of course, the [non-agile] units have to plan their resources. They run into real problems if they don't get an early heads-up, like 'Oh! Here comes [D].' (…) Let's say they have ten employees, all scheduled for the next six months. (…) But if no one's heard anything from [D] and then we show up at the last minute, saying, 'Hey! We've got something new for you. We think it'll take about twenty person days starting tomorrow,' they come back with their own estimate: 'What you're asking for will take three months.' That's when the problem starts. And that's not what we signed up for. (I.D.7)

The agile teams across all three cases responded to the uncertainty about resource allocation by ***entraining*** to the extended planning cycles of non-agile actors. In doing so, the agile teams extended their planning horizons to six months until they were synchronized with those of the interdependent non-agile actors. Once they shared the same planning cycles, agile and non-agile units were able to coordinate resource allocation more effectively, allowing agile teams to secure the resources they needed to carry out their activities.

> **Case [A1]:** Then we also need to create a forecast. We'll definitely do that early next week, so we're prepared for the second half of the year. (I.A1.2)

> **Case [C]:** The agile project also has to plan further ahead. In fact, it has to forecast the scope of its tasks about six months in advance, even though agile methods, strictly speaking, don't normally demand that. (I.C.9)

> **Case [D]:** In the sprint review, the product owner presents the activities relating to migrating the software to a new, modern interface design. He shows a PowerPoint slide outlining the timeline of the migration. The product owner has planned the migration activities through to the first release of 2022, six months in advance. He explains that having the tasks preplanned makes it possible to determine resource needs, such as the number of additional developers required. He assures that the extended planning reduces uncertainty for the agile project team: "We've already scheduled the migration into future sprints. That means we have security!" (Vignette from observation O.D.2)

It is noteworthy that, while the agile team in case [A1] viewed entraining as largely beneficial, the agile teams in cases [C] and [D] perceived it as a compromise or necessary evil required to secure resources. Unlike in cases [C] and [D], the agile team entrained with relative ease. A large portion of its tasks involved implementing regulatory directives as well as new or amended product features into the sales software. These tasks were defined by the regulatory compliance and product management units for six-month periods, respectively, with changes only occurring in rare and exceptional circumstances. Thus, they provided the agile team with a solid and stable foundation for forecasting its resource needs over a six-month planning period.

> **Case [A1]:** The January 2022 release hasn't started yet, but the roadmap and sprint planning are already finalized because everything needs to proceed in a staggered sequence. The assignments for the January release have been confirmed, and we have already received the requirements, which actually happens six months in advance. This is a good thing because, without it, proper planning would be impossible. It means we know which requirements must be completed by January. (…) Resource planning is also already finished, and based on the release schedule, I can already plan my sprints. So, everything has to follow this staggered approach. Otherwise, it simply wouldn't work. (I.A1.3)

For the agile teams in cases [C] and [D], entraining was accompanied by a noticeably higher effort for creating long-term plans. In case [C], the agile project team highlighted the difficulty that estimating the capacity needed from the data warehouse unit

presupposed flawless interface specifications at the outset of the planning cycle. Thus, the agile team had to dedicate substantial time and effort to estimating its resource needs over the planning horizon requested by the data warehouse unit, a task they perceived as burdensome.

> **Case [C]:** What I find exhausting is that you always have to consider all possible scenarios right from the start and have perfect mappings and specifications. Otherwise, you already have a problem. (I.C.21)

In case [D], entraining to the non-agile units' longer planning cycles compelled the agile project team to participate in complex planning processes. The insurance company in which the agile project team operated divided the year into four planning cycles, each extending over six months which inevitably led to overlaps. Each planning process, from the initial resource request and negotiation to the final allocation, lasted two months and was perceived as highly time-consuming, burdensome, and inefficient. The agile project team's perception of inefficiency largely reflected their experience that six-month plans were rarely realized as initially envisioned.

> **Case [D]:** In January, I'm asked to do planning for the period from May to September. Then it works. Basically, I have to specify everyone I need, including which month and how many person-days. (…) And by March, it's time to plan again, maybe for July to December or so. For me, it's just a lot of work. (…) Then there are overlapping months, and I think, 'Really? Already again?' It's a process that probably takes around two months to complete: filling out all the Excel sheets, consolidating them, sending them back to everyone, and then redistributing them for confirmations. These are just company processes that, frankly, I find exhausting. For us, it's pure overhead. They put so much emphasis on planning (…), but honestly, a lot of it is just like looking into a crystal ball. (…) Ultimately, of course, it's just a plan. Of course, things always turn out differently anyway. (…) But really, all the effort you put into thinking it through, planning it out and so on, it's kind of pointless. Because in the end, it never goes the way you planned. (I.D.2)

In addition to the high efforts for creating long-term plans, the agile teams regarded entraining to six-month planning cycles as problematic, as it could limit their flexibility. Flexibility proved to be an essential capability for the agile teams, who regularly encountered schedule disruptions that demanded immediate response. The six-month planning cycles tied the resources allocated to the agile teams to specific tasks, creating tensions whenever schedule disruptions caused these tasks to change. This was particularly pronounced for the agile teams in cases [C] and [D].

In case [C], the agile project team experienced two types of schedule disruptions, each demanding considerable flexibility. First, drawing on past experience, the agile project team recognized that regulatory authorities frequently issued new requirements on short notice, which then had to be implemented into the software within tight timeframes. These

externally triggered schedule disruptions typically required changes to the software functionalities, often requiring different interfaces with the data warehouse.

> **Case [C]:** Past experience has shown that there's always some last-minute change. Either the ECB [European Central Bank], the EBA [European Banking Authority], or someone else comes up with something new just three weeks before the deadline. (…) That tends to happen. They drop something on us and say, 'Alright banks, jump or die.' And you don't really think twice. You just jump. (I.C.2)

Second, a noticeable share of schedule disruptions emerged from the agile project team's inherent work practices. Adopting an iterative approach, the agile project team steadily gained insights and ideas for refining the regulatory reporting software as development progressed. Consequently, the interfaces that the agile project team required from the data warehouse unit evolved as new insights and ideas emerged. However, referring to the plans agreed upon at the beginning of the six-month planning cycles, the data warehouse unit showed little willingness to accept changes to the requested interfaces or their underlying specifications, as this could compromise their internal capacity planning. The data warehouse unit's rigid enforcement of the six-month plan limited the agile team's flexibility, which they relied on to handle schedule disruptions successfully. The observational vignette below illustrates this recurring tension in the collaboration between the agile project team and the data warehouse unit.

> **Case [C]:** During the sprint retrospective, a member of the agile project team criticizes that staff from the non-agile data warehouse unit are unwilling to revisit decisions that have already been made, even when changing them could lead to better outcomes. Specifically, it had become clear during the iterative development process that significantly more derivatives had to be processed than initially anticipated, which required adjustments to the planned software solutions and required changes to the interfaces required from the data warehouse unit. The project team member explains: "From our perspective, we've reached a crossroads where we should reasonably consider taking a different path to position ourselves better for the future." The Scrum master asks which colleagues they had discussed this with. The project team member points to two members of the data warehouse unit. The Scrum master then asks whether these colleagues might still be convinced or if they are completely opposed to the change. The project team member clarifies his point: "What matters to me is that we recognize that, in an agile setting, we sometimes need to revisit decisions made six months ago. That shouldn't be a problem … I just don't feel that we need to put up a wall here." (Vignette from observation O.C.40)

Similarly, in case [D], the agile project team, drawing on experience, expected that the tasks and resource needs defined in six-month plans were likely to change over time. As in case [C], many of the non-agile units, though not all, strictly adhered to the six-month plans. When changes in tasks or effort estimates led to short-term changes in resource needs in the agile project, non-agile units often showed little willingness to accommodate them.

> **Case [D]:** For some units, it doesn't really matter what's in the Excel sheet. They'll help us anyway. But others insist on the details, 'No, it says here you only want two person-days in August.' 'Yeah, well, I said that back in December last year. At that time, I didn't know I'd need seven.' (I.D.2)

In case [A1], the agile team encountered regular schedule disruptions but did not view its flexibility as being constrained. In contrast to cases [C] and [D], most schedule disruptions in case [A1] were caused by non-agile units. Non-agile units often fell behind on the development of interdependent software components, preventing them from meeting the delivery dates specified in the six-month plans. As a result, schedule disruptions occurred due to modifications of existing plans by non-agile units, which the agile team accommodated by adapting its own plans without encountering resistance, as in cases [C] and [D].

> **Case [A1]:** In waterfall, things tend to get delayed. And then everything gets pushed back as a result. Naturally, we have to adjust our plans accordingly. (I.A1.2)

To handle schedule disruptions effectively, the three agile teams sought to regain flexibility by detraining from the six-month planning cycles of non-agile units. By continuing to plan tasks and resource needs each sprint, the agile teams effectively operated with planning cycles of only two to three weeks, depending on the duration of their sprints. Sprint planning outcomes were binding for the agile teams, even when they contradicted the plans defined for six-month horizons.

In case [A1], although the sprints had been pre-planned six months ahead, the agile team continued to hold sprint planning sessions every two weeks. The plans developed during sprint planning were the only ones actually implemented, even when they diverged from the six-month plans agreed upon with non-agile units. This detraining allowed the agile team to handle schedule disruptions flexibly. While the six-month plans still reflected the projected delivery dates of interdependent software components, the two-week sprint planning often revealed that non-agile software development teams would not release them on time, enabling the agile team to adjust its tasks accordingly.

> **Case [A1]:** If something doesn't fit the schedule, we have to reorganize our sprints. We work in two-week sprints. So, if [printing] says, 'No, we can't deliver now, only two weeks later,' I say, 'Alright, then we'll postpone the topics that depend on [printing] until later. There's no point in developing them now if we can't test them yet because [printing] isn't ready. (I.A1.3)

In case [D], the agile project team estimated tasks and resource needs over a six-month planning horizon to secure resources. Yet neither these estimates nor the resulting six-month plan specifying tasks and resources were considered binding. Unlike the agile team

in case [A1], the agile project team in case [D] did not pre-plan its sprint cycles, anticipating that both tasks and resource needs would change substantially over the course of six months. Instead, it conducted sprint planning at the beginning of each sprint cycle to define the tasks and resource needs for an upcoming three-week interval. In doing so, the agile project team accepted deviations from the longer-term plans agreed upon with non-agile units.

> **Case [D]:** We don't plan the sprints in advance. (I.D.2)

> **Case [D]:** Ultimately, of course, it's just a plan. Things almost always turn out differently anyway. (…) In the end, it just ends up happening differently. (I.D.2)

While restoring flexibility, the agile project team's detraining severely limited planning abilities in non-agile insurance line and IT units. Short-term requests from the agile project team required capacities that these non-agile units had already allocated to other tasks in accordance with their six-month plans. Thus, the agile project team's short-term capacity demands represented schedule disruptions for the non-agile units, which either undermined their capacity planning when they complied with the requests or created waiting times for the agile project team if they did not.

> **Case [D]:** The problem is simply that we often end up having to allocate staff. That's my issue. Like with the risk manager … There are quite a few things coming in that need to be reviewed. And of course … If I just had a longer planning horizon, I could say: 'I'll schedule these tasks.' But [the agile team] comes to us spontaneously, saying, 'We've already done this and that, and it all needs to be professionally reviewed.' I just think, 'Oh! Great you did that. But now I need to find someone to review it.' I don't have the people just sitting here twiddling their thumbs, you know? (I.D.8)

In case [C], the agile project team also detrained from a six-month planning cycle in the task execution stage. Although it considered the capacities it had estimated over a six-month horizon to be binding, these estimates were not derived from a valid and reliable task plan. Anticipating that the requirements for the reporting software were constantly changing due to externally or internally triggered schedule disruptions, the agile project team refrained from defining specific tasks for six-months. Instead, it decided iteratively in two-week sprint planning cycles which software functionalities to develop and which interfaces were needed.

> **Case [C]:** You don't know exactly what's coming, but you just start anyway. It's not like it's completely without a plan; it's just different. Step by step. We go through it carefully, step by step. And we don't make a big, overly detailed plan upfront, which in the past often just gave an illusion of precision and security, or still does. I firmly believe that plans are just as fallible as any other plans [laughs]. And that's really the difference. But when people have always worked in a highly controlled environment, or under the illusion of control, it's obviously tough to say, 'Look, I don't

> know what we'll need from you in six months. We just know we'll need you, and it'll probably be moderately challenging, and we'll need you roughly X days. So, assuming that, please reserve this capacity for us.' And usually, that ends the discussion. But then they ask, 'What if you need more, or that person is on vacation? Then we say, 'Well, that's our project risk. We'll note it on our risk list. Thanks!' That's fine. I mean, we manage just fine with that. (I.C.2)

Like in case [D], detraining provided flexibility to the agile project team, but limited the planning abilities of the non-agile data warehouse unit in case [C]. First, it could not predict when the agile unit would request interfaces. Second, and even more critically, it could not anticipate which interfaces the agile project team would request. In light of the specialized expertise of its developers, the data warehouse unit was concerned that short-notice interface requests could strain its resources and create bottlenecks that hindered its ability to accommodate the agile team's requests. As the previous quote illustrates, the agile project team itself did not consider these potential bottlenecks problematic, but instead regarded them as an inherent part of project risk. Thus, negative repercussions of agile detraining were primarily articulated by the non-agile data warehouse unit.

> **Case [C]:** What we struggle with when collaborating with [the agile project team] is the following: we're actually a finance department in a bank. (…) So, two worlds collide. It's completely clear that we provide the foundation for their technical solutions. I think it's also clear that we would reasonably need some lead time to be able to work together over the course of the project. But with this agile approach, it's not possible. Because the colleagues can't tell us what topics will come up in the next six months! (…) That means we never really have sufficient lead time to get an idea of what might come our way. (I…) For me, it's not crucial whether we're talking about ten or fifteen [interfaces]. The quantity is interesting, of course, but what matters more is knowing what topics we'll likely be dealing with. Quite simply, it could be something like: 'Oh! We'll be discussing this type of information, and I know very little about it, but my colleague is the expert.' That would be useful to know because the colleague might be on parental leave during a specific period. These are the very basic things where planning really makes sense. Planning means doing something ex ante rather than realizing ex post, 'If I only had known before.' Next year, I might actually have parental leave. It would be good to plan around it. Otherwise, we'd have a resource bottleneck. (I.C.19)

Detraining from six-month planning cycles increased agile teams' flexibility in handling schedule disruptions, yet revived uncertainty in resource allocation, manifesting in waiting times during the task execution stage. Detraining often led to temporal misalignment between the resource needs of agile teams and the availability of resources in non-agile units. When agile teams requested resources at times no longer consistent with previously agreed plans, non-agile units that continued to follow those plans were often unable to meet these requests. Resources became available to agile teams only after being released from prior planning commitments. If agile teams sought to access planned resources at a time that diverged from the formal schedule, they were typically required to wait until the formally designated allocation time. Requests for additional or different resources typically had to be formally incorporated into the next planning cycle of the

non-agile units. Thus, the resulting waiting times varied from several days to several months, occasionally leading to substantial delays for the agile teams.

> **Case [A1]:** There have been occasional frustrations because we had to postpone things to make everything work. But it has never happened that we simply went into production without coordinating with our subsystems. Of course, if there's trouble, it's usually the product unit or sales unit that blames us. If we make a big fuss and say, 'We're going to launch this by that date', only to find out it's not possible, we have to apologize to the sales agents and explain that certain topics won't be ready in time. (I.A1.1)

> **Case [C]:** About four weeks ago, we received a requirement from the agile project. Ideally, we should have gotten it months earlier. Now, as I have always feared, this isn't something that can be explained or handled in ten minutes or an hour. (…) We probably won't be able to approve developing this requirement for the current release. Instead, we'll have to say, 'We can only start this next year.' That's what I mean by delay. (…) I think it's quite likely that we won't finish it in time and will have to postpone it. So, based on the requesters' wishes, we'll probably deliver about six months later. That's what I mean by delay. (I.C.19)

> **Case [D]:** We can't just say, 'We have this requirement. Please include it in your next sprint or the one after that.' Instead, the [non-agile units] basically told us, 'It's May now. Our planning horizon is closed until September. The next release is in December, and that's when you can submit your requirements.' (I.D.2)

Two practices emerged across the three cases to avoid or reduce waiting times: buffering and harmonizing. In case [D], the agile project team aimed to reduce waiting times by employing two distinct buffering practices. It drew on the creation of slack resources and on the integration of decision-making authority.

Waiting times in case [D] arose because the agile project team had detrained from the six-month planning cycles to respond to evolving task and resource needs, while the non-agile insurance line and IT units enforced adherence to these plans. The agile project team sought to prevent such waiting times by creating slack resources. During the planning stage, it intentionally requested significantly more resources than it realistically needed to carry out its activities. It even requested capacities in non-agile units, when it was not yet certain whether any interdependencies would occur. As a result, the agile project team was able to respond to evolving resource needs without the need to renegotiate previously agreed plans with non-agile units during the task execution stage.

> **Case [D]:** We come from an annual planning approach. We allocate skills over a very long timeframe. That means I basically have to say about six months in advance which skill I'll need. And that skill will be reserved for me. Now … We often need certain skills on shorter notice. So, we've found solutions for that. One approach is actually thinking ahead about when we'd need whom. Sometimes, we even ended up 'stockpiling' people. (I.D.3)

The extent to which the agile project team built slack resources depended on the specific non-agile unit with which it collaborated. Non-agile units differed in how firmly they

insisted on adherence to previously established plans. Especially in collaboration with non-agile units that had previously followed these plans very rigidly, the agile project team requested considerably more capacity, which was not required in collaboration with non-agile units that were generally willing to deviate from existing plans.

> **Case [D]:** Or I stockpile capacity upfront, which is, of course, kind of nonsense, rather than taking a step back and asking: what are the organization's … what are our current priorities? What do we want to work on this quarter? (…) It really depends on the department. If it's a department where I know how people operate, and they prefer scheduling rather than an unwanted change to their plan later, then I do it. Of course I do. (I.D.2)

While creating slack resources helped the agile project team prevent waiting times, it also entailed disadvantages. Slack resources could only be built as long as the agile project team could persuade non-agile units of the genuine necessity of the capacities it requested. If the agile project team did not call upon the capacities granted, it risked losing credibility. To mitigate this risk, the agile project team pressured itself to create enough tasks to use the excess staff capacities, which led to high workloads that it occasionally perceived as burdensome.

> **Case [D]:** By actually thinking ahead about when we would need whom, we did in fact reserve certain people in advance. That puts us under some pressure to actually give them work to do. Because if we keep reserving people and then don't make use of them, we lose credibility. (I.D.3)

A second buffering practice involved integrating decision-making into the agile project team. Once developed, the software functionalities had to be reviewed and approved by the non-agile insurance line units, the end users of the administrative software. Within this sign-off process, waiting times emerged because the agile project team had detrained from the six-month planning cycles, leaving the non-agile insurance line units unable to plan whether, when, or how many staff should be allocated to review and approve software functionalities. As a result, staff members were often engaged in scheduled tasks, which prevented them from reviewing software functionalities immediately. To avoid these waiting times, the agile project team sought to integrate decision-making authority for the approval of software functionalities. Ultimately, it succeeded in shifting the decision-making authority for approving software functionalities from the non-agile insurance line units to the insurance specialists within the project team or to the product owner. Consequently, the agile project approved software functionalities independently, without experiencing waiting times caused by divergent planning cycles during the task execution stage.

> **Case [D]:** The sign-off used to be done by [the insurance line units]. That's the old world. And we said, 'That slows us down. We want sign-off to happen within the project team because we have experts from various departments, and the people assigned to the team need the authority to make decisions.' So, they can say, 'Yes, that works for us,' or 'No, it doesn't.' If we constantly have to go through another loop or only find out three months later during integration testing whether a story is acceptable, that just doesn't work in an agile approach. (…) For example, we had to have conversations with the [insurance line units] to make sure we could work the way we wanted: that sign-offs would be done either by representatives within the agile team or by the product owner, to make sure we wouldn't be blocked or slowed down. (I.D.2)

Alongside the buffering practices in case [D], agile teams across all three cases employed harmonizing practices. Harmonizing practices aimed to prevent waiting times through anticipatory coordination, the assumption of additional tasks, or the reprioritization of tasks, without, however, synchronizing the planning cycles of agile teams and interdependent non-agile units.

In case [D], the agile project team could not coordinate all interdependencies during the task execution stage ex-ante by creating slack resources or integrating decision-making authority. Sporadically, capacity needs emerged that slack resources could not cover, or decision-making fell outside the authority delegated to the agile project team. In such instances, the agile project team relied on non-agile units to harmonize their divergent timelines by assuming additional tasks. Thus, the agile project team trusted that non-agile units, especially insurance line units, which were the end users of the administrative software, were sufficiently interested in the timely completion of new software functionalities to provide short-term support when needed.

> **Case [D]:** Otherwise, I have to say, when it comes to working with colleagues from the [insurance line units] – if we need something from them like, 'Could you quickly check this?' or 'Could you quickly do that?' – it usually works very well and pretty quickly. (I.D.4)

> **Case [D]:** Of course, it's fair enough that these tasks eventually come our way. But if I'd known five weeks ago, I could have organized it better and planned for it, like, 'Hey, something is coming up.' But we'll manage. We'll find people. And yeah … It just has to get done. It's important. (I.D.7)

Similarly, in case [C], the non-agile data warehouse unit commonly harmonized divergent timelines by assuming unplanned tasks that the agile project team requested at short notice. Notably, only the data warehouse unit raised concerns about potential waiting times within the agile project. Since the agile project team considered waiting times to be a general project risk, it did not itself initiate buffering or harmonizing practices to reduce waiting times. By contrast, the non-agile data warehouse unit actively sought to minimize waiting times. Anticipating that waiting times in the agile project could delay software delivery, data warehouse unit developers assumed short-notice requests for interfaces in

addition to their planned tasks. However, without holding temporal buffers for unplanned tasks, they could harmonize divergent planning cycles only to a limited extent by assuming additional workload. When short-notice requests from the agile project team were too large or complex, the data warehouse unit was unable to assume them and postponed them to a subsequent planning cycle. Thus, waiting times in the agile project team were not always reliably resolved through harmonizing.

> **Case [C]:** So far, we've actually managed to get through a lot [of additional short-time requirements] without much extra time. Meaning, up until now, we've been doing it without really building in any time buffers. (I.C.19)

In case [A1], the agile team combined anticipatory coordination with reprioritizations of its tasks to create harmonious moments for collaboration with non-agile software development teams. Non-agile software development teams supplied interdependent software components that had to be released concurrently with the agile team's software functionalities in order to render the sales software fully functional for end users. Frequent changes to the schedules of these non-agile software development teams undermined the reliability of their planned release timelines. While the initial plan presented synchronized release timelines, ensuing changes amplified the risk that the agile and non-agile software development teams would operate on divergent schedules. This necessitated careful coordination to realign release timelines and to ensure the concurrent deployment of interdependent software components. To raise awareness of modifications of the release timelines within these non-agile software development teams, the agile team established weekly jour fixes to discuss and collaboratively adjust release timelines.

> **Case [A1]:** I have to maintain continuous communication between the [non-agile backend teams]. I also hold a weekly meeting where I bring together all the [non-agile backend team] leads and say, 'Here's the roadmap I've created for each release.' We have different environments where we develop, and I update them like this: 'We are here now, and in three weeks, we plan to be here. How is it going with [printing]? Where are you? Does it fit? Or not?' There have definitely been moments when things didn't line up. In those cases, we work together to find a solution. (I.A1.3)

Participatory observations offered valuable insights into the process through which divergent release timelines were harmonized during the weekly jour fixes. Each session began with the agile team sharing its current plan of tasks and timelines to ensure transparency and guide the discussion. This plan typically outlined the period from the focal jour fixe to the agile team's upcoming software release. It was structured in two-week intervals that aligned with agile sprint cycles, providing an overview of the components planned for development. Serving as a common point of reference, the agile

team's plan facilitated discussions in which all participants, including non-agile team representatives, collaboratively aligned their divergent release timelines. Discussions centered on the agile team's planned release timeline and exposed interdependencies among the various software components. Once identified, both the agile team and the non-agile software development teams collaboratively negotiated adjustments to their timelines, frequently requiring modifications to individual release timelines by either advancing or postponing the deployment of specific software components to enable concurrent releases. Thus, the jour fixe served as a key mechanism for harmonizing divergent temporal patterns by facilitating both the identification of interdependencies and the reprioritization of tasks to create synchronous moments for concurrent software releases. Notably, while aligning release timelines through harmonizing practices, both the agile and non-agile software development teams sustained their underlying temporal structures, continuing to plan and modify tasks and release dates within their respective yet distinct planning cycles.

> **Case [A1]:** The Scrum master facilitates the weekly status meeting. At the start, she presents the plan for the upcoming August release, showing an Excel spreadsheet that lists the topics for the next four sprints leading up to that release. She asks the interdependent IT backend teams if the plan works for them. The product owner of the agile team inquires whether a specific software functionality can be delivered in the next release. An employee from one backend IT team points out that this functionality requires simultaneous implementation in two sales software systems; otherwise, the software will produce errors. He adds that the team responsible for the other system does not plan to implement it before January. The product owner looks puzzled and mentions she had heard that the functionality was supposed to be implemented by October. A participant responsible for the other sales software replies that he is not aware of the functionality or any October implementation. The product owner seems relieved and says, 'Then I'll do it in January as well.' At this point, an employee responsible for developing the initially mentioned functionality speaks up, confirming that it will be implemented and asks when it would be best. The first employee insists that the functionality must appear simultaneously in both sales software applications. The product owner stresses that January is the only feasible option, saying, 'So January is the only option since that's when we both have a release scheduled. I can live with that. We just need to agree on it here.' (Vignette from observation O.A1.1)

Harmonizing was not confined to the weekly jour fixes. Effective coordination necessitated ongoing communication, extending beyond the formally scheduled jour fixe sessions. As a voluntary arrangement between agile and non-agile software development teams, participation in the weekly jour fixe was optional, and not all non-agile software development teams attended every session. Personal or operational constraints, such as vacations or increased workloads due to impending software releases, could hinder attendance. Nevertheless, coordination remained essential and was maintained through bilateral communication between the Scrum master and representatives of the respective non-agile software development teams. Moreover, coordination needs occasionally arose

between jour fixe meetings due to their urgency, making it impossible to defer them until the next session. In such cases, the agile team maintained continuous communication with non-agile software development teams.

> **Case [A1]:** The product owner asks whether a representative from the software development team [printing] is attending the meeting. The Scrum master responds that no one from that team is currently present. The product owner notes that [printing] has a release scheduled just a few days after the agile team's own release. The Scrum master confirms this. The product owner then asks the Scrum master to find out what kind of joint testing [printing] would like to carry out. She expresses her support for coordinated testing. The Scrum master agrees and promises to follow up with the colleagues from [printing] after the weekly jour fixe. (Vignette from observation O.A1.1)

Harmonizing planned release timelines involved extensive temporal coordination efforts, compelling the agile team to devote a substantial portion of its members' capacity to communication. Agile project team members reported spending multiple hours per week on coordinative tasks, primarily communicating with non-agile software development teams and reconfiguring their sprint plans.

> **Case [A1]:** It is actually a process that never ends. It's recurring. Every week, I have to check the roadmap again, bring all the component owners together, and keep asking, 'Is this still okay, or has anything changed?' And if something has changed, we have to replan our sprints. Then I need to communicate with the [product owner] again and say, 'Look, the way you pre-planned the sprints no longer works because … Let's reschedule them. (…) So, it's a lot of coordination. A whole lot of coordination. There are days when I literally spend ten hours on the phone. (I.A1.3)

In response to the substantial temporal coordination effort arising from harmonizing practices and to define clear responsibility, the agile team extended the conventional role structure of the Scrum framework by introducing an additional role. Drawing on prior plan-driven work practices, it reinstated the role of release manager, explicitly responsible for aligning and synchronizing divergent release timelines. This responsibility was eventually assigned to the Scrum master, who thereby assumed a dual role, ensuring effectiveness both within the agile team and across boundaries with interdependent non-agile units. The Scrum master, initially a role focused within the agile team, has thus evolved into a brokering role bridging the divergent temporal structures of agile and non-agile software development teams.

> **Case [A1]:** So, the most important thing here is good release management. That's it. What's interesting is that this actually goes back to earlier times. In agile, there's no longer a dedicated release manager. At least, I haven't heard of that role. And I don't know who's supposed to take it on nowadays. For us, it's the Scrum master. It's actually the same person who takes care of release management. (I.A1.1)

In summary, the first pattern illustrates how three agile teams navigate the tensions arising from divergent planning cycles by using entraining and detraining practices, which enable

dynamic shifts between shared and divergent temporal patterns. Entraining, applied during the planning stage, creates a synchronous episode in which not only initial uncertainty in resource allocation is resolved, but also allows for buffering practices that coordinate ex-ante to allow for maintaining divergent temporal patterns during the task execution stage. However, entraining to longer planning cycles limits the flexibility of agile teams and, consequently, their ability to respond to schedule disruptions. They mitigate these tensions by detraining from longer-term planning cycles once resources required for task execution have been secured. During the task execution stage, detraining causes waiting times, which are addressed through buffering and harmonizing, two practices that allow agile teams to sustain their divergent temporal patterns.

5.1.2. Pattern 2: Entraining-detraining dynamics with new temporal patterns

Figure 6 depicts the structure of the second pattern, which closely mirrors that of the first. It reveals temporal dynamics arising from the interplay of entraining and detraining practices. Agile units navigated ***uncertainty about resource allocation*** by ***entraining*** to the extended planning cycles of non-agile actors. Unlike in the first pattern, non-agile units attempted to mitigate the ***limited flexibility*** of agile units by developing and ***dictating new temporal patterns***, aiming to maintain shared temporal structures. Since this intervention did not fully resolve agile flexibility constraints, agile units ***detrained*** from both preexisting and newly established planning cycles, thereby reproducing divergent temporal structures. Consistent with the first pattern, detraining gave rise to ***waiting times***, which agile units addressed through ***buffering*** and ***harmonizing*** practices, without reinstating shared temporal structures.

I observed the second pattern in cases [B1], [B2], and [B3], all situated in the same insurance company, which involved collaborations between three agile software development teams and various interdependent non-agile units. Case [B1] involved a long-term agile project undertaken with the objective of developing new software for sales agents. To develop functional software, the agile project team depended on the capacity of developers in a non-agile data warehouse unit, which provided the technical

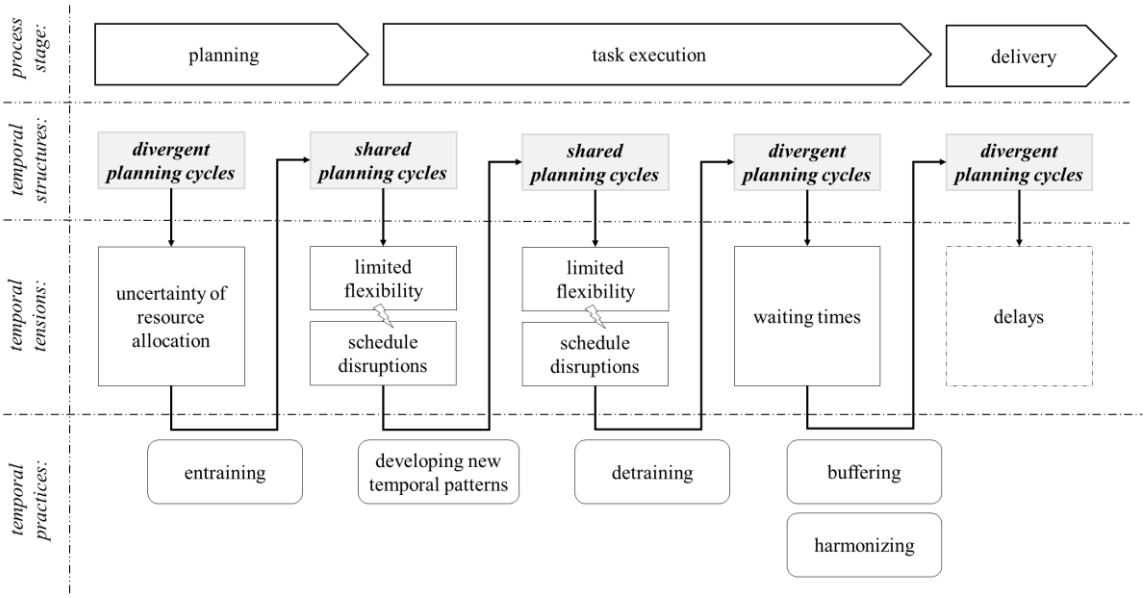

Figure 6: Entraining-detraining dynamics with new temporal patterns

interfaces that connected the sales software to the company's data warehouse. The agile teams in cases [B2] and [B3] were embedded within a newly founded line unit, established one year before data collection, with the mission of delivering digital services to end customers in a flexible, fast, and customer-centric manner. In case [B2], the agile team was responsible for developing premium calculators that permitted customers to calculate individual insurance premiums autonomously via an online platform. The agile team in case [B3] was tasked with creating a customer portal that allowed customers to manage personal information, such as addresses or bank account details, and perform minor administrative tasks independently, including adjusting the annual mileage of their car insurance policy without assistance from a customer advisor. In both cases, the agile teams relied on the expertise of insurance specialists seconded from various insurance line units for one-year periods to support the agile teams' activities. In addition, although each agile team had a core of permanently employed software developers, they occasionally depended on input from IT specialists who, for efficiency reasons, had not been integrated into the agile teams.

Differences in the planning cycles of agile teams and interdependent non-agile units created uncertainty about resource allocation for the agile teams. During the early stages of agile adoption, the agile teams planned exclusively on a sprint-by-sprint basis. The resulting short-term resource demands, however, conflicted with the non-agile units' annual planning horizons. In the organization, tasks and resources were routinely

scheduled within annual planning cycles. Within these corporate planning processes, the resources of non-agile units were typically fully committed to assigned tasks. Consequently, non-agile units were often unable to accommodate the agile teams' short-term resource demands that had not been anticipated in their annual plans.

> **Case [B1]:** We [the agile unit] definitely plan. But … The requirements just come – I don't want to say more spontaneously – but more agile. At some point, you just realize, 'Okay, in this sprint, we want to focus on this topic. Alright, we still need data from them.' Then, the back-and-forth coordination begins. And then, [name of data warehouse unit developer] doesn't say, 'Great! I have been waiting for you.' Instead, it's more like, 'You'll have to wait your turn because I've got other tasks to complete for this team and that team.' (I.B1.4)

> **Case [B2]:** Insurance specialists are in every team, except for [the new agile team], which was founded mid-year, and where the insurance line units no longer had the capacity to second insurance specialists. (I.B2.2)

> **Case [B3]:** It kind of feels like we're on an island right now. We work agile, but most people around us still work differently. And whenever you need something from someone else, whether it's interfaces or maybe something related to the business side, there's always friction to coordinate it. And of course, it's about resources. Sometimes colleagues just aren't available on short notice to get it done. (I.B3.4)

Across all three cases, the agile teams responded to uncertainty about resource allocation by entraining to the established annual planning cycles. Since both the non-provision of developer capacities in the data warehouse in case [B1] and the non-secondment of insurance specialists could compromise task execution and goal attainment, the agile teams began to project their expected tasks over annual planning horizons. In doing so, they detrained from sprint-based planning and synchronized with the organization's long-established annual planning cycles. By adopting shared planning cycles, the agile teams could effectively participate in the corporate planning process and secure the resources required to carry out their activities. Specifically, in case [B1], the agile team secured a defined number of person-days that developers in the non-agile data warehouse unit were committed to dedicate to the agile team's tasks. In cases [B2] and [B3], the agile teams ensured that insurance line units and IT units seconded the required specialists to support the agile teams' activities.

> **Case [B1]:** At [the insurance company], we have an annual corporate planning process for the upcoming year. It's basically to determine: What's on the horizon? Then I have to say, 'As a project, I want to implement this and that. These will be the tasks. And in these tasks, you, dear group leader, will be involved with your activities.' Of course, they ask, 'What exactly do you need?' And they have to report back on the required effort. Once that's done, there are usually many, many follow-up questions. Naturally … This year is no different. Once the project is underway at the beginning of the year, they ask, 'So, when is it roughly coming? Do you have a schedule? When can I block those resources for myself, so they're not assigned elsewhere?' (I.B1.1)

> **Case [B2]:** So, this [annual] planning process is basically the corporation's planning tool, and it applies to everything happening at [B], including both traditional and agile projects. (…) Everyone starts by submitting a wish list. Each unit can tell another, 'For my 100 projects, I need this many

> person-days from various units.' By the end, all units are usually heavily overbooked. For example, at [B2], we were almost overbooked by a factor of four. Then negotiations begin about what to cut. Accordingly, if more projects come from the car insurance division, the car insurance division has to second more experts. If more come from the liability division, then liability must second more. So, it's always a topic-related secondment that lasts for one year (I.B2.2)
>
> **Case [B3]:** We're still working on annual cycles, following this classic project planning process. (…) Throughout the year, up to specific deadlines, projects for the following year have to be specified. These projects are submitted from January to June, so during the first half of the year, they are submitted for the next year. These represent the major initiatives for the upcoming year. (…) Those that match the capacity available [in the agile project] for the next year will then be approved sometime in August or September and planned accordingly. (I.B3.1)

The agile teams problematized the entraining practice, perceiving it as a limitation to their flexibility and thus undermining their ability to address schedule disruptions. Drawing on prior experience, the agile teams had learned that tasks and resource needs often changed over the course of the year for which they had been planned. Tasks could prove more complex than expected, requiring additional time and resources for completion. Unexpected tasks could emerge, creating additional resource needs. This not only made tasks and resource needs difficult to predict over the course of a year but also rendered plans spanning annual horizons invalid and obsolete. However, agile teams were largely tied to the tentatively estimated resources documented in annual plans, making it difficult to adjust or supplement resources as needed. This limitation on flexibility represented a significant source of tensions for agile teams, who viewed flexibility as a core advantage of agile practices.

> **Case [B1]:** I just think the whole idea of planning a year or even one and a half years ahead feels a bit … Since we're now working in an agile way (…), taking smaller steps, I don't think that's really fair. Because it doesn't always hold true, I'd prefer the cycle could somehow be shortened and we could check in more frequently – maybe quarterly – to ask: Are we still on track? Or what capacity should I actually plan for in the next quarter? That seems more reasonable than (…) planning a whole year ahead and knowing exactly what we're going to do. Especially since, at least in our project (…), what's happening and how we're changing feels too fast for me to predict a whole year in advance what's going to happen and to what extent. (I.B1.1)
>
> **Case [B2]:** [B] has an annual corporate planning process. That's always in the summer. (…) Except for [B2], everything else is traditional. That means all application development, insurance operations – basically everything except what we do. About a hundred people are involved in the corporate planning process. It's all traditional, not agile. The rest of the company is not agile at all. (…) In August, they plan the entire next year in advance. I really have to say: What do I want? Then it gets estimated. (…) And controlling monitors all of that. Everyone knows I can plan today, and by tomorrow it might already be obsolete (…) So we also have to plan a year ahead. That's … Well, it goes against what we do. (I.B2.1)
>
> **Case [B3]:** Our company's organizational unit, which is originally responsible for the project planning, doesn't understand it because they plan in annual cycles. They say, ‚But why? All the topics are planned. All the capacities have been allocated. What's the problem?' But we say, 'If we now have to squeeze all the topics into one quarter (…) just because they have to be done by then, it simply doesn't work.' (I.B.3)

The second pattern differs from the first pattern described in the previous chapter in that non-agile units attempt to restore the flexibility of agile teams by developing new temporal patterns. Across all three cases, I observed that non-agile units shortened planning cycles as a way to bridge the differences between annual managerial planning and agile teams' sprint-based planning. In cases [B2] and [B3], new temporal patterns were developed and introduced top-down by management. In case [B1], new temporal patterns emerged bottom-up, developed and introduced by the non-agile data warehouse unit to resolve its own tensions with the annual planning process. Notably, across the three cases, new temporal patterns were developed solely by non-agile units rather than collaboratively, and were then dictated to the agile teams. In none of these cases did the new temporal patterns replace the established annual planning cycles. Instead, they were implemented in parallel and functioned as supplements. Agile teams were still required to entrain to annual planning cycles to address uncertainties in resource allocation during the planning stage. However, the shortened planning cycles enabled annual plans to be reviewed and adjusted at regular intervals.

In cases [B2] and [B3], management developed and introduced new temporal patterns. When setting up the organizational unit in which the two agile teams were embedded, management intentionally adopted agile practices to enable rapid, flexible, and customer-focused delivery of digital services. Limitations in agile teams' flexibility undermined the broader objectives management sought to achieve, resulting in flexibility tensions also recognized by management.

To enhance agile teams' flexibility, management established quarterly opportunities for reviewing and adjusting annual plans. They introduced quarterly meetings to review, discuss, and adjust the tasks and resource allocations documented in the annual plans of the agile teams. Adjustments could be initiated by management or agile teams themselves. If management identified a need to reprioritize tasks, it used the quarterly review meetings to revise task plans and resource allocation as necessary. Similarly, agile teams, regularly receiving unplanned tasks from the insurance line units, used the quarterly review meetings to propose reprioritizations and request adjustments to resource allocation.

> **Case [B2]:** And there's a quarterly review meeting where we go in and say, 'Okay. What have we accomplished? What are our goals for the next quarter? What are our priorities?' Then, of course, just like in the steering committee … The board members and department heads can make decisions and say, 'Thanks for pointing that out. I've thought it over, and my priorities have now changed.' (I.B2.4)

> **Case [B2]:** And ultimately, communication (…) towards stakeholders (…) happens once a quarter, for example, through the (…) quarterly review meeting, to check: What's the status? Maybe there are new requirements, and so on. Of course, additional requirements can come up during the year, outside of this planning process. (I.B2.3)

Quarterly meetings allowed annual plans to be decomposed into quarterly intervals. By allowing agile teams to review and adjust annual planning outcomes quarterly, they effectively reduced planning horizons from twelve to three months. Although agile teams continued to forecast tasks for the entire year during the initial planning cycle, they were relieved of the pressure to provide exact resource estimates, as these could be revised quarterly.

> **Case [B2]:** So, we really only have a quarterly perspective now, where we create standard roadmaps. (…) How does [name of the product owner] handle it now? (…) In a very broad way. She has extended the timelines. There are only release dates for [car insurance tariff launches]. You have to understand this to make sense of it. Some topics must be planned firmly because they're tied to a tariff launch. (…) That's why we have to work with releases for the tariff calculators here. For [liability, accident, and property insurances], we mostly operate only on a quarterly level. We want to develop at that pace, and going forward, we plan to prioritize only quarterly. (I.B.3)

> **Case [B3]:** During the first sprint review in April, the product owner of the agile team in case [B3] informs the team about the last quarterly meeting with management. He shows the team the PowerPoint slides he presented to management the previous day. These slides display not only the tasks the agile team will address in the second quarter (May to July) but also an annual roadmap with task schedules for the third and fourth quarters. (Vignette from observation O.B3.7)

Management widely perceived the introduction of quarterly planning cycles as a significant achievement. This favorable perception arose from comparing the new quarterly planning cycles with the more rigid annual planning cycles, which remained prevalent in the organization.

> **Cases [B1], [B2], and [B3]:** [The insurance company] comes from a completely plan-driven approach. (…) Now we're planning in quarterly cycles. That's already a huge, huge step for [the insurance company]. (I.B.3)

The agile teams, however, assessed the quarterly planning cycles less favorably. Rather than comparing them with the established annual planning cycles, the agile teams contrasted them with the shorter sprint-based planning cycles inherent to agile practices. From this viewpoint, the quarterly planning cycles were still considered too long. Accurately predicting in which quarter a task would be accomplished remained a challenge for the agile teams. Moreover, adjustments of tasks and resource needs within an ongoing quarter remained infeasible, which continued to limit agile teams' flexibility. Consequently, agile teams continued to perceive a misalignment between quarterly planning cycles and their preferred sprint-based planning cycles.

> **Case [B2]:** For us, [just prioritizing tasks] would be ideal. We could simply rank the 27 epics from 1 to 27. But our management board, which is supposedly agile, insists on quarterly planning. So now we have to go through it, quarter by quarter. (I.B2.1)

> **Case [B3]:** Quarterly planning is our holy grail now. It's what we align everything we do around. (…) Essentially, everything hinges on this quarterly planning, which – let's be honest – is anything but agile. (I.B3.4)

In case [B1], new temporal patterns were developed and introduced by the non-agile data warehouse unit to resolve a tension that arose within this unit itself regarding the annual planning cycles. Unlike the agile teams, the data warehouse unit did not problematize the duration of annual planning cycles but rather criticized the lack of precision in timing resource allocations. Annual planning processes determined the total capacity that the non-agile data warehouse unit had to allocate to both interdependent agile and non-agile units, assuming that capacity needs were evenly distributed over the year. However, the data warehouse unit found that capacity needs peaked in the second and third quarters, precisely when most software development teams transitioned from conceptualization to development activities. Consequently, developer capacity in the data warehouse unit was underutilized during the first and fourth quarters and overutilized during the second and third. As a result, the data warehouse unit was unable to meet all capacity demands of the interdependent agile and non-agile units, although they had been formally committed during the annual planning process. Thus, the core tension for the data warehouse unit was balancing capacity availability against the uneven timing of capacity demands throughout the annual planning period.

> **Case [B1]:** Normally, it's only around this time of the year that we actually start planning for the next year: what we're actually going to do. And the outcome is usually … Well, I have to say, the planning itself is crap. It gets done, and then we're already booked for a bunch of projects. (…) The problem is this: everything is based on resources and capacities. And these projects come along and say, 'Okay, we'll need X person-days or person-months from you next year. You currently have X employees, so that's X person-months. And that means we end up with 110% capacity utilization, and everyone's happy. But the planning never accounts for when those capacities will actually be needed. They just say, 'We need them,' but not when. And every project is more or less the same: in Q1, they want to develop a concept, in Q2 and Q3, they want to test and deploy. Hallelujah! So that means we hit 110% utilization in Q2 and Q3. It's completely messed up! (I.B1.8)

Since the plans formulated within the context of the annual planning cycles specified only the volume of resources, but not the timing of their use, the data warehouse unit relied on interdependent agile and non-agile units to indicate precisely when capacity would be required. However, agile units in particular were unable to provide precise information, offering only tentative timelines that typically changed over time.

> **Case [B1]:** We try to get some kind of dates from [the agile units] about when they'll need interfaces from us. They always try their best to provide them, but they're not sure either. They can only roughly estimate in which sprint it might be. (…) So, when they finally have a more precise date, it's usually quite soon that they actually need it. They only think in terms of two or three sprints, and they only plan ahead that far. (I.B1.8)

The data warehouse unit expected that shorter planning cycles would yield more precise and dependable information on the timing of capacity needs. Thus, it divided the annual planning cycle into six two-month intervals, each governed by a dedicated planning process. Although the corporate annual planning process continued to specify the total capacity the data warehouse unit had to allocate to various agile and non-agile units, the timing of capacity needs, based on the specific tasks assigned to the data warehouse developers, needed to be forecasted accurately only for the next two-month interval. Thus, by introducing two-month planning cycles, the non-agile data warehouse unit established new temporal patterns, which were enacted in parallel with the established annual corporate planning cycles.

> **Case [B1]:** The idea was basically to say: We're trying to be an agile project among agile projects, for the things that need to be handled by us. And we approached it the same way as the agile projects, but still had certain planning horizons in place. For example, I would contact all the projects every two months and ask, 'Will you need anything from us in the next two (…) months?' The goal was to get a general sense of direction: What topics do we need to tackle, and which ones don't we? Then we would prioritize them accordingly and work through them in sprints. (I.B1.2)

To become "an agile project among agile projects", the non-agile data warehouse unit also adopted two-week sprints. Although the two-month planning cycles were divided into four sprints, the data warehouse unit did not perform sprint planning at the beginning of each sprint. Instead, tasks were planned once at the beginning of each two-month interval and then allocated to the four sprints. Thus, these sprints were implemented neither to improve flexibility nor to support iterative development; their primary purpose was to enhance transparency. It allowed the data warehouse unit to indicate transparently to agile and non-agile units when interfaces would be available, facilitating harmonization with agile units' sprint cycles.

> **Case [B1]:** We structured [our activities] around two-week sprints. We said, 'We'll try to release all features in calendar weeks 2, 4, 6, 8, 10', and we communicated that externally to ask, 'Would that work for you? Does that roughly align with your planning?' Then we had a round of coordination, 'Yes, that works' or 'No, that doesn't work.' After that, we had a plan of 200 tasks for the next two months, where everyone could check, based on their best knowledge and judgment, when the [data warehouse unit] would deliver. (I.B1.2)

Similar to cases [B2] and [B3], the agile team in case [B1] continued to perceive its flexibility as constrained. Because tasks were rigidly and tightly scheduled across the four sprints within a two-month planning interval, both the scope and timing of interface

delivery were fixed, leaving the agile project team unable to flexibly change interfaces within the two-month planning intervals. This largely precluded iterative refinements of interfaces, for example, when the agile project team's expectations were only partially met, within the agile project team's sprint cycles.

> **Case [B1]:** Regarding the colleagues from [the data warehouse unit], whom we rely on … I'd say they mostly work in a traditional way. They just tell us when the [interfaces] will be ready, and that's basically it. Take it or leave it. We might try another round, saying 'we really need to adjust this or that,' but in the end, they generally don't respond. (I.B1.5)

Consequently, the data warehouse unit observed that the agile project team in case [B1] did not fully entrain to the shortened planning cycles. Although the agile project team generally communicated capacity needs at the beginning of the two-month planning cycles, it did not refrain from requesting refinements or new interface developments during an ongoing interval, in line with its sprint-based work rhythms. Thus, shortening the planning cycles did not fully alleviate the data warehouse unit's planning tensions, which continued to result from misaligned planning practices.

> **Case [B1]:** [Question]: Outside of the requests you made, did agile teams ever come with short-term demands like, 'We need something on short notice?' And if so, within what time frame? (…) [Answer]: Yeah, of course that happens. [laughs]. (I.B1.2)

Across all three cases, new temporal patterns failed to fully resolve the flexibility limitations experienced by the agile teams. Although two- and three-month planning cycles marked a substantial departure from annual planning cycles, the agile teams still perceived them as too long to fit their sprint-based work cycles. Although the agile teams engaged with these new temporal patterns, participating in quarterly meetings to revise tasks and resource needs as well as in the data warehouse unit's bimonthly planning sessions, they did not regard the outcomes of these planning processes as binding.

Detraining from both established and shortened non-agile planning cycles, the agile teams continued to perform sprint planning on two- or three-week cycles, determining tasks and resource needs independently from longer-term plans. In case [B3], the agile team rejected the idea of committing to plans across three months. It submitted task and resource estimates in the quarterly planning process, yet did not treat them as binding and routinely revised them in biweekly sprint planning. In case [B1], the agile project team participated in the data warehouse unit's bimonthly task and capacity planning, yet ultimately rescheduled tasks and capacity needs within its own two-week sprint planning cycles. The agile team in case [B2] showed the closest alignment with quarterly planning

cycles imposed by management, opting to plan one to three sprints ahead. Given that each sprint lasted three weeks, planning over three sprints aligned naturally with quarterly planning cycles. However, sprints scheduled further in advance were increasingly tentative, allowing the agile team to redefine tasks and capacity needs on short notice and preserve flexibility within the quarterly planning cycles. Thus, by detraining from longer-term planning cycles, the agile teams regained flexibility that had previously been limited by both the annual corporate planning cycles and the newly developed two- and three-month planning cycles. Retaining two- to three-week sprint cycles enabled the agile teams to respond more flexibly to unexpected tasks and adapt plans when tasks required more resources or more time than initially expected.

> **Case [B1]:** Theoretically, once the [capacities] are approved, once they've been cleared by the other side, I can plan how I actually need to use them. (I.B1.1)
>
> **Case [B2]:** I've now introduced a practice of planning at least one and up to three sprints in advance. (…) So, the next sprint is already very detailed and refined, and you know exactly: Okay, how many story points are in the next sprint? But the further ahead you look, the more open things still are. (I.B2.3)
>
> **Case [B3]:** We plan from sprint to sprint and define sprint goals. So, we don't plan four or five sprints ahead right now. We try to estimate the overall implementation timeframe. (…) But it hasn't really proven to be that important to predict it as precisely as possible – well, for us at least. Instead, we just start and then respond, seeing where we end up. Because whether we plan or not … If you say 'three sprints' and it ends up being six sprints … Well, then it's six sprints. (I.B3.1)

However, detraining also revived resource allocation tensions, which manifested as waiting times in the task execution stage. Deviations from plans jointly established during shared planning cycles resulted in misalignments between the agile teams' capacity needs and the availability of those capacities in non-agile units. Although agile units had secured capacity in non-agile units by entraining to annual planning cycles, they were often unable to access it at the times needed during the task execution stage.

> **Case [B1]:** But when you look at the team from a product owner's perspective, you realize, 'I really want to push my progress.' Or the Scrum master, who is responsible for performance, repeatedly says, 'This is frustrating. We have to wait again. We really need this issue resolved by the [data warehouse unit]. This can't keep happening, because we need to keep moving forward.' You can definitely sense that frustration. It also affects the developers, creating … Well, maybe 'overall' is too strong … But it does contribute to a somewhat negative atmosphere within the product team. (…) 'We can't be waiting again. We've already asked five times. Now it's your turn. Do something!' (I.B1.1)
>
> **Case [B2]:** During sprint planning, the developers schedule their tasks for the upcoming sprint. An insurance specialist suggests a task involving replacing the icons on the website. One developer points out a dependency on another team: 'We first need an update of the StyLib. That's planned for their upcoming sprint. So, it will definitely take at least another week.' She emphasizes several times that the update won't be expected any sooner than a week after that date. Nonetheless, during the daily standup meeting the next day, the same issue comes up again. The developer once again reminds the team that she has to wait for the update of the StyLib unit. (Vignette from observation O.B2.9)

> **Case [B3]:** It's not like you get what you need or want right away. Sometimes you wait for months. And that really takes a toll, especially when you have interdependencies with other units. (I.B3.3)

In case [B2], I observe waiting times during participatory observations, yet the agile team members do not frame them as a problem. This could indicate that they occurred less frequently, were shorter, or were overall less consequential, possibly for two reasons. First, as noted earlier, the agile team in case [B2] aligned more strongly with the quarterly planning cycles than those in cases [B1] and [B3], as it tentatively planned up to three sprints in advance. Second, as will be outlined in Chapter 5.2, the daily activities of both the agile team and interdependent non-agile units were strongly guided by deadlines, which unambiguously defined when tasks needed to be completed. Given that they were working toward the same deadlines, the scope for detraining was more limited than in cases [B1] or [B3].

As in the first pattern, the agile teams used buffering and harmonizing practices to minimize waiting times during the task execution stage. In cases [B2] and [B3], the agile teams implemented buffering practices to coordinate interdependencies with non-agile units ex ante. Both agile teams were strongly dependent on insurance line units, where highly specialized knowledge about insurance products, tariffs, and contracts resided. This knowledge was essential for the agile teams to develop premium calculators and a customer portal that met customer needs while ensuring the accuracy of product and contract information. The agile teams employed buffering practices by integrating insurance specialists into their teams, which required the non-agile insurance line units to second staff.

> **Case [B2]:** The insurance specialists still belong to their [insurance line] unit, but they work for us [in the agile team] and are physically located here. They are also managed by the [agile] project manager, even though they remain affiliated with their original organizational unit. (I.B2.1)

> **Case [B3]:** I am an insurance specialist. So, I'm not an employee of [the agile team], but rather an employee of the car insurance division. Basically, I'm seconded to the [agile team] as an insurance specialist. (I.B3.3)

Integrating insurance specialists into the agile teams constituted a buffering practice, as it substantially reduced cross-boundary coordination needs during the task execution stage and thus prevented waiting times. After being seconded, the insurance specialists worked exclusively for the agile teams and were no longer required to juggle agile team tasks with their line responsibilities. Minor disruptions to this exclusivity arose because insurance specialists were still expected to attend relevant meetings in their line units, which accounted for approximately one day. This ensured that they stayed connected to

their line units and retained ongoing access to current domain-specific knowledge, enabling them to contribute effectively to the activities of the agile team. Through their secondment, the insurance specialists synchronized with the temporal patterns of the agile teams. By organizing and planning work in two- to three-week sprint cycles and attending the daily standup meetings, they aligned both the timing and pacing of their activities with those of the agile teams.

> **Case [B2]:** They are 100% seconded to [the agile team]. 100% means 4+1: four days working within [the agile team] and one day in their original unit. Because, of course, they are still expected to attend group meetings there to maintain contact. (I.B2.1)

> **Case [B3]:** So, I am fully seconded to the [agile team]. That means, in my regular workday, except for one group meeting per week where I still see my former colleagues, whom I no longer really work with, I am, more or less, a full member (…) within the Scrum environment. I attend all the regular meetings you're supposed to join, whether it's the daily, planning, refinement, or whatever. And yes … By now, it has definitely become second nature. (I.B3.3)

The scope of specialist secondments to the agile teams was determined during the annual planning processes, in which resources were formally allocated to agile and non-agile units. Thus, the success of this buffering practice relied on the agile teams entraining to non-agile planning cycles during the planning stage, where it created shared temporal patterns that facilitated coordination. Each secondment initially lasted one year, at the end of which the insurance specialists either returned to their non-agile insurance line units or were seconded for an additional year.

> **Case [B2]:** So, this [annual] planning process is basically the corporation's planning tool, and it applies to everything happening at [B], including both traditional and agile projects. (…) Then negotiations begin about what to cut. Accordingly, if more projects come from the car insurance division, the car insurance division has to second more experts. If more come from the liability division, then liability must second more. So, it's always a topic-related secondment that lasts for one year (I.B2.2)

In case [B1], the agile team also considered the integration of specialists from the non-agile data warehouse, representing the most critical dependency, as a preferred approach to minimize waiting times during the task execution stage. However, in case [B1], this buffering practice was not implemented. Operating with a small, highly specialized workforce, the data warehouse unit was responsible for providing support to numerous agile and non-agile units across the organization and therefore declined to second staff to the agile project team.

> **Case [B1]:** My wish was that someone from [the data warehouse unit] would work with us directly on the team, because we have such extensive interdependencies. Unfortunately, that didn't work out from a staffing perspective. (…) The colleagues managing this central data warehouse not only collaborate with the sales unit. They have interdependencies across the entire organization. And they already have an annual project planning process where they‘re completely overloaded. Then

> you come along with an agile project and say, 'Actually, we need four people from you working with us full-time.' That just doesn't work. (I.B1.6)

Consequently, the agile project team and the data warehouse unit relied on harmonizing practices to jointly identify and exploit ephemeral moments of synchrony, enabling coordinated work without departing from their inherently different planning cycles. To achieve this, they combined anticipatory coordination with the assumption of additional tasks and the reprioritization of tasks. As a first attempt at harmonization, I observed anticipatory coordination initiated by the non-agile data warehouse unit. At the beginning of its two-month planning intervals, the data warehouse unit sent an email to all interdependent agile and non-agile units, requesting them to provide their tasks and capacity needs for the forthcoming two months. In doing so, it used anticipatory coordination to raise the agile project team's awareness of its two-month planning cycles.

> **Case [B1]:** If I sent out my email on January 1st, asking, 'What does it look like for the next two months?' All the respective project managers get contacted, 'I know you need something from us. When exactly do you need it?' (I.B1.2)

This inquiry enabled the data warehouse unit to harmonize the timing of its capacity provisioning with the capacity demands of both agile and non-agile units. Using the submitted tasks, the data warehouse unit was able to forecast its capacity utilization. When these forecasts indicated over- or underutilization, the data warehouse unit took proactive steps to mitigate these imbalances. Overutilization prompted it to organize tasks by priority, postponing those of lower importance to subsequent planning intervals. When it anticipated underutilization, it contacted agile and non-agile units that had submitted few or no tasks, warning that leaving capacity unused in the current planning interval could lead to overutilization and unmet capacity demand in future planning intervals. This problem-focused communication often prompted interdependent units to reconsider their plans and reprioritize tasks that had been scheduled for later periods.

> **Case [B1]:** If no one has responded [capacity needs] for the first two months, I would at least follow up with something like, 'Are you sure? Or are there some other possibilities to explore?' So, it's about trying to steer things a bit, so to speak. […] And quite often, solutions come up where someone says, 'Okay, I hadn't planned to take care of this yet, but maybe for this reason, I'll go ahead and do it after all.' (I.B1.2)

In addition to the proactive communication initiated by the data warehouse unit at the beginning of each two-month planning interval, the agile project team and the data warehouse unit jointly held weekly jour fixes to coordinate emergent changes in the timing of capacity demand and provisioning. Deliberately kept brief so as not to be

perceived as an undue burden, jour fixes were regarded as a valuable opportunity for anticipatory coordination.

> **Case [B1]:** For projects where we knew we (…) were more involved, we started setting up weekly jour fixes. At some point, that just became the norm. (I.B.2)

> **Case [B1]:** For us, it's simply faster, or rather … What do I mean by faster? I think it just involves less bureaucratic effort. Because otherwise … I mean, this half hour doesn't cost much time. If I think back … No idea, on July 19th, we had eight topics. Often, they're just questions. People aren't even sure if there's really anything to do. In the end, they say, 'Yeah, let's take it on. It's something completely different. You just misunderstood it.' 'Ah. Okay. That works!' So, these are often just questions that need to be asked, not bugs we have to put on hold. And that's why I would say that the jour fixes have proven to be a pragmatic approach for us. (I.B1.7)

> **Case [B1]:** And at least I don't think [the jour fixes] are a bad thing. Because even though we're not involved in the sprint planning of [the agile team], it's nice to at least get a bit of insight when bigger topics are coming up. (I.B1.8)

Emergent changes in capacity demands had two main causes. First, they arose when developed interfaces were defective or found to be suboptimal for use in the agile project team's sales software. In such instances, the agile project team required ad-hoc capacity from the data warehouse unit to correct defects or improve the interfaces. Because these capacity needs emerged from ongoing development work, they could not be anticipated or planned in advance. Still, they were particularly pressing for the agile project team, which could not proceed with sales software development without the interfaces, and therefore aimed to avoid the waiting times that would inevitably arise if emergent capacity needs were addressed only in the subsequent two-month planning interval.

> **Case [B1]:** Both developers and insurance specialists are part of these weekly meetings. That's where you coordinate things like: My return code looks like this, and here's how you can respond to it. Or from [the agile team's] side, someone might say, ‚I'd like the interface to behave a little differently.' Right. And because this is a bigger change and something new, there is simply more to discuss. So, meeting weekly makes sense. (I.B.2)

Second, emergent changes in capacity demands also arose when the agile project team requested new interfaces. Prior experience had shown that the agile project team often made ad-hoc requests for new interfaces, disregarding the two-month planning cycles of the data warehouse unit. The weekly jour fixes helped to coordinate ad-hoc requests and harmonize them with ongoing planned development work in the data warehouse unit in the ongoing two-month planning interval. They served as a space for collaborative discussion, during which the agile team would present new interface requests along with their desired timelines for delivery. The data warehouse unit would then assess the feasibility of developing the interfaces within these timelines and provide immediate feedback on whether and when they could be delivered.

> **Case [B1]:** So, we've now set up a weekly meeting with [the data warehouse unit], where we basically build an agenda: What issues have come up? Whether it's interface errors or anything else that might occur. Or we use it as a kind of forward look. To stick with [name of data warehouse developer] as an example: 'In five weeks, we'll need communication data from you, like email addresses, phone numbers, and that kind of stuff.' That way, he can prepare and say, 'Okay, five weeks that's doable,' or 'No, I've still got these other topics I need to tackle first.' (I.B1.4)

> **Case [B1]:** We discuss new requirements exclusively in this jour fixe meeting. (…) I think that's a good solution because otherwise … Yeah … It gives everything a framework, I'd say. Without it, we'd still be passing along our requirements independently. Just because we don't have a jour fixe doesn't mean the requirements disappear. But without it, the whole process would be far less coordinated. (I.B1.7)

During the jour fixes, discussions primarily focused on whether the emergent capacity needs of the agile project team could be addressed by the data warehouse unit assuming additional tasks or by the agile project team reprioritizing planned tasks. The data warehouse unit assessed whether the additional workload associated with emergent capacity demands was acceptable. Typically, this applied to tasks requiring no more than four hours of coding but that were critical for the agile project team's progress. Moreover, the data warehouse unit and the agile project team jointly assessed whether the emergent capacity needs could be addressed by the agile project team rescheduling lower-priority tasks into subsequent planning intervals. When reprioritizations were possible, the data warehouse was able to accommodate higher capacity demands within ongoing planning intervals.

> **Case [B1]:** If [the data warehouse unit] just has to code for two, three, or four hours and that would block someone else, of course, we would still try to squeeze it in. (…) If the project has other tasks in the queue, they can reschedule immediately. That isn't a problem. And if it is the same insurance line unit, it is also possible to talk to the person responsible and say: 'If you've got two products, as long as we haven't started yet, you can still change your priorities. We don't mind.' (I.B1.2)

Occasionally, interdependent agile or non-agile units withdrew capacity needs. In such cases, the data warehouse unit reprioritized its tasks and allocated the freed capacity to address emergent demands from the agile project team. Given the agile project's high priority and organizational importance, the data warehouse unit was encouraged to give precedence to its tasks whenever capacity became available.

> **Case [B1]:** We only have [temporal] buffers if somebody didn't show up after all. That happens just as often as when someone comes and says, 'We just thought of something. We've just detailed the story and realize we need something from you.' Of course, the same can happen the other way around, like when they say, 'We've found a different solution. We don't need you after all.' We don't always have a choice in those situations. We just have to react more quickly. (I.B1.2)

> **Case [B1]:** These days, especially since [the agile team] is quite a prestigious project, our supervisors have made it clear that we should give them fairly high priority whenever they need something. (I.B1.8)

Still, when neither taking on additional tasks nor reprioritizing planned tasks was possible, the agile project team's emergent capacity needs remained unaddressed, and waiting times persisted. Thus, harmonizing practices could reduce, but not entirely eliminate, waiting times.

> **Case [B1]:** And if there were no [available capacities or options to swap tasks], then the [agile team] really had to get in line and accept: 'Alright, so it won't happen for another two months.' (I.B1.2)

In cases [B2] and [B3], harmonizing practices complemented buffering practices. The need for harmonizing practices arose because not all potential interdependencies could be coordinated ex-ante through buffering practices. Waiting times emerged when the agile project team did not anticipate interdependencies with non-agile units during the planning stage, and thus did not request (slack) resources in these units. In such cases, when its capacity needs were not accounted for in non-agile units' plans, the agile project team tried to reduce waiting times for urgent capacity needs by leveraging close relationships with employees in non-agile units. It aimed to have non-agile units quickly accommodate their unplanned capacity needs, either by assuming unplanned tasks in addition to planned workloads or by reprioritizing planned tasks in order to prioritize emergent capacity needs.

> **Case [B2]:** Sometimes it works like this because people know each other personally, and you can negotiate a deal with one group leader or another. You say, 'Okay, maybe I didn't plan for the capacity, but I might have other tasks where I need to do less than planned.' Then you juggle things around. But this isn't a formal process or workaround. It's about commitment and goodwill. (I.B2.2)

> **Case [B3]:** In the Scrum environment – the agile environment – it works like this: If something is urgent and important (…), I create a ticket and assign it the highest priority. Then it's done within two weeks. We can't and don't expect the same from colleagues in the [non-agile] line organization. But, of course, by communicating, you can emphasize the importance, and usually, things then work out very well and quickly. (I.B3.3)

In summary, the second pattern, like the first, shows that three agile teams resolve tensions arising from temporal differences with non-agile units by dynamically shifting between synchronized and divergent planning cycles. The second pattern differs from the first in that, after initially entraining during the planning stage, new temporal patterns emerge, which, however, do not adequately address the flexibility tensions of the agile teams during the task execution stage. Consequently, the agile teams detrain from both the initially entrained and newly developed temporal patterns, leading to waiting times, which they resolve, as in the first pattern, through a combination of buffering and

harmonizing practices. This allows them to preserve divergent temporal structures during the task execution stage.

5.1.3. Pattern 3: Persistent entraining

Distinct from the previous two patterns, which were characterized by a dynamic interplay of entraining and detraining, the third pattern depicts a case in which an agile unit persistently entrains to the planning cycles of non-agile units. Similar to the first two patterns, the agile unit in case [A2] resolved its initial ***uncertainty in resource allocation*** by ***entraining***. However, unlike the six cases of the first two patterns, where ***limited flexibility*** due to entraining was addressed through detraining, the agile unit in case [A2] sought to ***entrain*** non-agile units with shorter planning cycles to the longer planning cycles of its dominant Zeitgeber. This led to the emergence of a persistent entraining pattern that impaired not only the flexibility but also ***limited the responsiveness*** of the agile unit. Despite these synchronization efforts, this pattern also involved ***waiting times*** caused by schedule disruptions in both agile and non-agile units, which were addressed through ***buffering***, ***harmonizing***, and ***isolating*** practices. Figure 7 illustrates the third pattern.

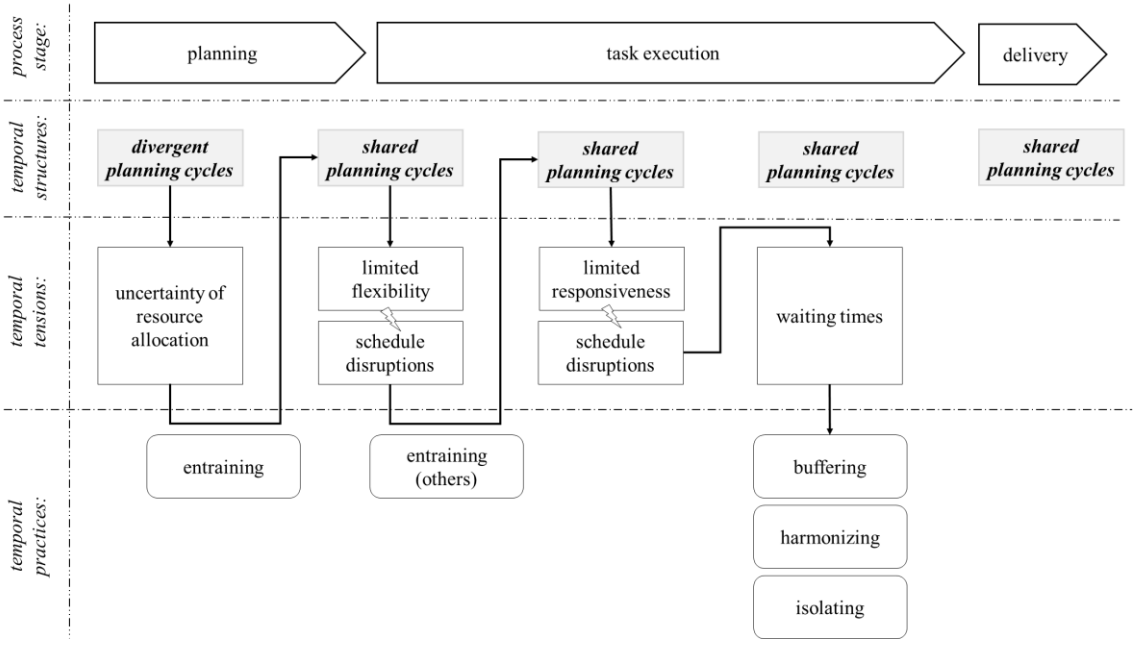

Figure 7: Persistent entraining

Case [A2] describes an agile software development team tasked with the continuous improvement of administrative software employed by the customer service unit of an insurance company. As a line unit within the company's IT division, the agile team retained its own dedicated personnel. Nevertheless, the agile unit was unable to develop all the required software functionalities independently. Several components were developed exclusively by specialized non-agile software development teams that employed plan-based work practices. These components encompassed the calculation and disbursement of commissions, as well as the management of customer correspondence, the latter being referred to as the electronic mailbox. To achieve full functionality for end users, these specialized components needed to be integrated into the administrative software. Personnel capacity in these non-agile software development teams was therefore a critical resource, essential for the agile team to fulfill its objectives. Given that these teams followed six-month planning cycles, the agile team was compelled to project its tasks and resource needs across extended planning horizons to secure the required capacities, thereby mitigating uncertainty in resource allocation.

> **Case [A2]:** When we work with [the commissions unit], or with [the data warehouse unit], or with [the electronic inbox unit] (…), that's when we step outside the agile world. These [non-agile IT units] require a formal request at a point when we often don't even know what exactly we need. That includes [the data warehouse unit] and [the electronic inbox unit]. The electronic inbox is where we manage resubmissions, both incoming and outgoing. Incoming emails can trigger automated business transactions, and outgoing emails (…) need to be properly archived there. These are two examples of dependencies that must work smoothly. Especially when they say, 'No, we don't have the time or resources for that,' we can only respond, 'That's unfortunate.' These dependencies can actually become a showstopper for a given initiative. (I.A2.1)

By entraining to the longer planning cycles of the specialized software development teams, the agile team ensured access to the required capacities. This entrainment proved unproblematic, since the agile team was largely able to forecast its tasks reliably over a six-month horizon. A substantial portion of its tasks derived from regulatory requirements or product development initiatives and was assigned to the agile team by the regulatory compliance and product management units at the beginning of their respective six-month planning cycles. These tasks offered a solid basis for planning resource needs and enabled the agile team to entrain to the planning cycles of non-agile software development teams with relative ease.

> **Case [A2]:** For the second half of the year, we've made a provisional plan in which mandatory topics, such as legal requirements, have been assigned. It's now up to the agile teams to decide when to address each item and who will take responsibility for it. (I.A.1)

Whereas agile teams in most cases explicitly criticized limitations of flexibility resulting from entrainment, the agile team in case [A2] did not. Nevertheless, I observed how the agile team's flexibility was constrained by the six-month planning cycles, a limitation that became particularly evident during collaboration with the non-agile customer service unit, the primary end user of the administrative software. The temporal structures of the customer service unit were shaped by its interactions with external customers. Due to the unpredictable nature of customer inquiries, long-term planning proved largely ineffective, prompting the customer service to implement ad-hoc work practices characterized by short, day-to-day planning cycles. These ad-hoc practices shaped how the customer service interacted with the agile team. In contrast to the regulatory compliance and product management units, which submitted requirements solely at the beginning of the six-month planning cycle, the customer service unit introduced new requirements more spontaneously, often during sprint reviews in which it evaluated the functionalities developed by the agile team.

Although this behavior embodied an incremental approach to software development, a hallmark of agile practices, it generated considerable tension for the agile team. The short-term planning cycles of the customer service unit conflicted with the six-month cycles to which the agile team had entrained to secure resources from interdependent non-agile software development teams. The agile team regarded these evolving requirements as unwelcome schedule disruptions, mainly because its entrainment had constrained it to a detailed long-term task plan with limited flexibility. Consequently, the agile team often declined to accommodate short-term requests, limiting its responsiveness to the customer service unit's needs.

> **Case [A2]:** We have a dialogue system that mainly handles payouts and generates letters (…). And I'm talking about actual letters here, not statements or invoices, but real customer correspondence, which can trigger resubmissions and follow-up actions. So, we said, 'That makes no sense. The system is far too complex to maintain, so we'll move those functionalities into our administrative system, where they belong.' The idea was solid and was well-received. We then aimed to replicate the letters (…). That part went fairly quickly. But then [the customer service unit] realized, 'If that's the case, then all the letters are incorrect because …' There were various reasons. So, we needed to make adjustments. Fine, we made adjustments! But with each one, they identified a new requirement. And that always happened once we showed them the updated letter, as part of a test! It's really these colleagues, although experts in their field, keep identifying new requirements that they could have raised earlier, if they had thought about it. But they only notice once they've seen and tested something. And so, these issues always come up at the very end. (…) If we had known six or eight weeks ago, we would have included those topics from the start. (I.A2.1)

Notably, the agile team attributed the tensions it experienced exclusively to the customer service unit's lack of foresight. Thus, the agile team did not attempt to restore flexibility

by detraining from the six-month planning cycles. Instead, it sought to entrain the customer service unit to these extended planning cycles. By doing so, the agile team aimed to suppress short-term demands, thereby attempting to reduce the necessity to operate flexibly. Consequently, despite agile temporal structures being inherently more closely aligned with those of the customer service unit, the agile team ultimately advocated for the extended planning cycles of the dominant Zeitgeber.

To entrain the customer service unit to extended planning cycles, the agile team engaged in different preventive and reactive activities. Seeking to prevent disruptions to the six-month planning cycles through short-term requirements, the agile team initiated a meeting with the customer service unit to define and schedule its requirements for the next planning cycle. In preparation and to ensure effective planning, it requested a comprehensive list of all foreseeable requirements from the customer service unit. In an effort to provide sufficient input for the entire planning cycle, the customer service unit extrapolated its requirements, generating a list that substantially surpassed the agile team's implementation capacity. Upon reviewing the list, the agile team identified substantial redundancies, with many items targeting the same software functionalities or test configurations. This situation provoked frustration and dissatisfaction within the agile team, which interpreted the list as further evidence of the customer service unit's unrealistic and disruptive planning practices, which ultimately generated superfluous workloads.

> **Case [A2]:** For this release, I tried to schedule a meeting early on to ask: What exactly do we want to test? And … well … in hindsight, that question kind of backfired. They actually created a list of everything they wanted to check. It ended up being longer than the time we had available for implementation. (…) The testing requirements were so extensive that around 50% of the results would have been identical, just with slightly different test configurations, making it hard to trace the underlying logic. We could have achieved the same coverage with a third of the test cases using cross-testing. But instead, they took a very rigid approach. If this case, then that … then that … and then I need this. There was no use of combinatorial logic. (I.A2.1)

The preventive practice eventually failed, as the customer service unit was unable to plan its requirements over a six-month horizon in accordance with the agile team's expectations. To avoid the excessive planning-induced workloads resulting from the customer service unit's extrapolation of requirements, the agile team decided to reject a portion of them. This, in turn, provoked dissatisfaction and irritation within the customer service unit, which had regarded the submitted list as a binding commitment for implementation. As a result, the customer service unit once more experienced the agile

team as unresponsive to its needs, notwithstanding its attempts to entrain to the six-month planning cycle advocated by the agile team.

> **Case [A2]:** We were able to get it back under control. Maybe having a list like that isn't such a bad thing – if you then work together to turn it into a set of cross-combinations. In this release, though, it created some tension: 'You were the one who wanted this. Now you've got it, and now we're not getting what we need.' (I.A2.1)

The reactive practice proved more effective. Whenever the customer service unit submitted requirements during the ongoing six-month planning cycle, the agile team's product owner leveraged these opportunities to initiate reflective discussions with customer service unit employees. He addressed short-notice submissions by emphasizing that they could have been planned in advance, consistently illustrating his argument with prior examples of short-term requests that had created tensions for both the agile team and the customer service unit. Although this reactive practice had not yet produced observable behavioral changes during the data collection period, the product owner considered it successful. Customer service unit employees began to demonstrate greater awareness of and sensitivity to the agile team's planning preferences, as indicated by their rising propensity to apologize for, rather than justify, short-term submissions of requirements.

> **Case [A2]:** At the same time, I've been trying to have more in-depth conversations with colleagues. I keep bringing up examples like: 'Come on … this is such a classic case. We really could've thought of this from the beginning.' And their responses are becoming more and more remorseful. At first, it was the usual resistance and so on. But now it's more like, 'Yeah, you're actually right, but …' So, something has shifted in their mindset. And that's already a meaningful step. (I.A2.1)

By applying preventive and reflective practices to entrain the customer service unit to six-month planning cycles, the agile team aimed to gradually resolve the tensions it faced, namely the schedule disruptions caused by the customer service unit's ad-hoc requirements. Unlike agile units in previous cases, the agile team in case [A2] did not seek to restore its own flexibility and responsiveness. Instead, it aimed to obviate the need for these capabilities by entraining the customer service to extended planning cycles that minimized the occurrence of unplanned, ad-hoc requirements.

Notably, despite persistently entraining to the six-month planning cycles of non-agile units and adhering to pre-established schedules to avoid misalignments between resource demand and availability, the agile team in case [A2] still encountered waiting times during the task execution stage. For example, the agile team occasionally had to wait for non-

agile IT units to deliver specialized software components. At times, it also had to wait for the product management, regulatory compliance, and customer service units to provide requirements for software functionalities or feedback on whether the developed software functionalities met their expectations. Waiting times routinely surfaced during sprint planning sessions, which always began with a review of tasks left unfinished in the previous sprint. Agile team members frequently attributed delayed task completion to waiting times that occurred in their collaboration with non-agile units.

> **Case [A2]:** During sprint planning, the agile team discusses which tasks from the previous sprint remain incomplete. One team member explains that her task is still unfinished because she's waiting for information from the actuarial unit. Without that input, she can't move forward. She says she'll follow up again with her contact in that unit to get the missing information. (Vignette from observation O.A2.3)

Waiting times emerged because, despite all efforts to synchronize the planning cycles of interdependent units carefully, schedule disruptions could not be prevented. They continued to occur and eroded the validity of plans established during the planning stage, which in turn reduced their effectiveness for coordination. A particularly salient example was the postponement of a company-wide software migration. While the postponement itself constituted a schedule disruption, the migration additionally created unforeseen problems in various non-agile units, leading them to deviate from planned timelines, which substantially affected the agile team's work. The agile team no longer received the required inputs as initially scheduled and had to wait for non-agile units before it could proceed with its tasks.

> **Case [A2]:** At the beginning, we were still working to a different schedule. Then, in the summer (...), the major system migration was delayed by three weeks, even though we were already in the middle of it. (…) Apparently, we didn't raise any red flags at that point. That's probably on us, because the entire plan had relied on having those three weeks. (…) We had no insight into the backlogs and workloads that the migration had created. Eventually, a crisis meeting was held about [the accounting unit], because it was only six or eight weeks ago that we learned none of our topics had been implemented yet – they were still finalizing the annual financial statements. (…) That's when it became clear that we were running three, four, even five weeks behind schedule. (I.A2.1)

However, schedule disruptions were not only caused by major events. They also arose when non-agile units, which could not focus exclusively on the agile team's tasks but instead juggled multiple responsibilities and assignments simultaneously, did not prioritize the agile team's requests, resulting in the delayed delivery of urgently required inputs.

> **Case [A2]:** I'm involved in many other projects, tasks, working groups – all the usual things that come up during the day. This is an agile team, of course, and with agile ways of working comes

> the expectation that someone is always available. Naturally, the assumption is: 'I'll get a quick response from [I.A2.3].' But when I'm juggling so many different things, I can't always keep up. Sometimes I have to say, 'Sorry, I just can't do it right now.' And in agile teams, the reaction is: 'We're working in an agile way now, we need quick feedback! We can't move forward without it!' That's a recurring challenge. (I.A2.3)

In their collaborations, the agile team and various interdependent non-agile units applied different practices to reduce waiting times. I observed the use of buffering practices in collaboration with the product management and regulatory reporting units, harmonizing practices in collaboration with the customer service unit, and, uniquely across all cases, isolating practices in collaboration with selected IT units.

The agile team was primarily dependent on the product management and regulatory compliance units to provide requirements for software functionalities and to review and approve them after development. To implement their requirements, the agile team relied on detailed specifications that clearly described each requirement. Without such specifications, it was unable to develop software that fulfilled underlying requirements. To avoid disruptions in the agile team's task execution caused by waiting for these requirements, both the regulatory compliance and product management units employed buffering practices. Specifically, they engaged in advance preparation with the goal of completing specifications for all potential requirements before these had to be communicated to the agile team at the beginning of the six-month planning cycle. The asynchronous structuring of work prevented the agile team from waiting for regulatory compliance and product management units to prepare specifications, thereby enabling software development without the need for coordination during task execution.

> **Case [A2]:** The second category includes all topics and requirements for which preparations need to be made. That means functional specification, effort estimations, and resource planning. At the same time, there's no guarantee yet that any of it will actually be implemented. The point is to have the preparations in place so that if the agile team eventually pulls a task from the backlog into implementation, there's no need to start debating: 'Wait a minute, we can't manage this now because we're still missing input.' (I.A.1)

Advance preparation of specifications shifted the agile team's dependency on the regulatory compliance and product management units from the task execution to the planning stage. This buffering practice proved effective, as the agile team did not report waiting times in its collaboration with these units, nor did I observe such waiting times during sprint planning sessions or sprint reviews. However, it had adverse effects on the regulatory compliance and product management units. They were required to complete all specifications without knowing whether the agile team was able to process all

requirements. Thus, the asynchronous nature of advance preparation created coordination gaps. Consequently, the regulatory compliance and product management units faced the challenge of deciding, under uncertainty, how many specifications to prepare in advance.

> **Case [A2]:** We don't have just one agile team; we have several. (…) And of course, each team has different amounts of time and resources. At that point, you really have to take a top-down view to ensure everything can still be coordinated (…) Right from the start, we need to be clear: 'Are we doing all three priority categories or not?' (…) And that's where the discussions always start: 'Well … can we really handle all 40 topics? Or do we need to say we stop at 20? Then the next 20 won't happen, because if there are no requirements in place, no one can implement them.' (I.A2.2)

Preparing too few specifications, the regulatory compliance and product management units made the buffering practice ineffective and reinstated waiting times in the agile team. Conversely, preparing too many specifications led the regulatory compliance and product management units to utilize their own capacities inefficiently, as some of the specifications would not be processed within the upcoming six months. Often, this ultimately led to wasted capacity, particularly when requirements were deprioritized or substantially altered in the subsequent planning cycle, rendering the specifications unusable. Thus, minimizing waiting times for the agile team required the regulatory compliance and product management units to tolerate some inefficiency in their own capacity use.

> **Case [A2]:** The colleagues have fully covered the second priority category – or are capable of fully covering it. The resources are available. But there are so many individual components (…) and burdens involved that we can't actually implement many of the topics from the second category. That means they've prepared something, and now it's just sitting on the shelf until we eventually use it. Or they have to roll it back, which is a problem (…), and in the worst case, they've just produced junk data. (I.A2.1)

In collaboration with the customer service unit, I observed harmonizing practices. Despite synchronization efforts, the temporal patterns of the customer service unit and the agile team were not always aligned, as the responsibilities of the customer service unit toward external customers frequently required ad-hoc task changes in response to customer requests. The customer service unit's unpredictable workloads sometimes interfered with the timelines for providing input to the agile team. Input from the customer service unit typically involved information or clarification needed to process the requirements they had submitted to the agile team. As the end users of the software developed by the agile team, employees in the customer service unit had a vested interest in providing timely input, anticipating that the delayed delivery of software functionalities would significantly impair their work effectiveness in the future. Thus, employees in the customer service unit actively sought to harmonize their temporal patterns with those of

the agile team, particularly by reprioritizing tasks. For example, when customer service responsibilities prevented immediate response to the agile team's request, customer service unit employees frequently interrupted their current tasks and prioritized the agile team's request. In doing so, they prevented waiting times in the agile team, yet shifted them into their collaboration with external customers.

> **Case [A2]:** So, communication is mainly via email. Very much so … You can tell right away. I mean, Outlook is basically my main work tool. About 80% of everything runs through Outlook. I can immediately see who's writing. Then, of course, if I know that's the agile team, I at least take a quick look, because I have to. It can't just be left hanging. That's not an option. But if I realize I can't help right now, I'll write back: 'I cannot support you at the moment. At the latest by the end of the week. If [the product owner] then says, 'Yes, sorry. We need it today or tomorrow at the latest,' I have to weigh things up and say: 'Okay, sorry.' Because these implementations have a really high impact. (…) If I don't answer the question, it might prevent implementation. In other words, if this doesn't come through now, we'll have to keep processing everything manually. I know that if that really happened, by the start of the year at the latest, we'd have problems. The number of cases we'd have to handle manually would increase enormously, seriously jeopardizing operational stability in customer service. That means afterwards we wouldn't be able to focus on anything else. That's something I obviously keep in mind and have to consider. So, I say, 'Stop!' and provide the information. (…) It's hard for me to say: 'No, I don't have time right now.' (I.A2.3)

Unique among all cases examined in this dissertation, the agile team in case [A2] employed isolating practices when collaborating with selected non-agile software development teams, such as the electronic mailbox team, which was responsible for developing software to manage customer correspondence. This non-agile software development team ensured that all customer letters generated by the agile team's administrative software were systematically stored in an electronic archive. For this purpose, each letter had to be assigned a specific category label. While the electronic mailbox team provided these labels upon request, they did not always do so within the timeframe required by the agile team. The agile team circumvented waiting times by creating its own labels, thereby effectively cutting its dependency on the electronic mailbox team. Although this improvised solution enabled the agile team to continue work immediately, it compromised the long-term consistency of the electronic archive, potentially increasing future effort for the customer service staff, who might face difficulties locating documents due to the improvised labeling.

> **Case [A2]:** With [the electronic inbox unit], sometimes we end up with a solution that's a bit messy – but at least, it's a solution. (…) Normally, a new letter would be categorized for the inbox. I'll just say: information with invoice, tax confirmation, and so on. Letters need to be stored with a related synonym in this inbox. Then we assign it once we receive it. If we don't get it, we start by sorting into broad categories like 'information with invoice' and general information' as a first step. At least that way, the documents can be properly achieved later. The tagging is pretty poor. Customer service doesn't really know where to look and just ends up searching manually. That's not a big problem right now, because with the release information we send out, they're informed that this will be the tagging system for the next six months. But if a customer calls four years from

> now asking, 'Where's the confirmation?', no one will remember the release info from four years ago. (I.A2.1)

To forestall future burdens on the customer service unit, the agile team treated its improvised labeling as a temporary solution. Thus, the agile team typically requested a correct and system-compliant label from the electronic mailbox team by which it replaced the improvised label in the subsequent planning cycle. However, the update of the new label entailed additional effort for the agile team, as it required prior technical implementation and testing. Thus, although improvised workarounds allowed the agile team to bypass dependencies and avoid waiting times in the short term, they increased future workload.

> **Case [A2]:** These are the kinds of (…) things where we're not exactly precise. But we manage by using whatever is available – something that roughly matches what's needed. Then, with the next possible update, we reclassify the letters again. Of course, that means another adjustment, another necessary test, so additional costs that weren't planned. (I.A2.1)

It shall be noted that isolating practices could not be employed in collaboration with any non-agile software development teams. Evidence shows that in many other cases, the agile team had to endure waiting times.

> **Case [A2]:** We're usually stuck when [the data warehouse unit] doesn't deliver. For example, if they need to implement a new interface that we depend on, we can only continue development once that interface starts providing data. Until then, everything is on hold. (I.A2.1)

In sum, pattern 3 illustrates how an agile team persistently entrained to the temporal structures of a dominant non-agile Zeitgeber. Similar to patterns 1 and 2, the agile team initially entrained to reduce the uncertainty of resource allocation. However, it responded differently to the flexibility limitations arising from the initial entrainment. Instead of regaining flexibility through detraining, the agile team attempted to reduce the necessity for flexible responses by entraining non-agile units with shorter planning cycles to the six-month planning cycles of its dominant Zeitgeber. Thereby, it limited its responsiveness to the evolving needs of its end users. Despite its efforts to minimize schedule disruptions by maintaining shared planning cycles, schedule disruptions continued to occur and caused waiting times for the agile team. These waiting times were mitigated by using buffering, harmonizing, and isolating practices.

5.1.4. Pattern 4: Persistent detraining

The fourth pattern, as shown in Figure 8, represents a case of persistent detraining. Distinct from the first three patterns, the agile unit in the fourth pattern did not rely on entrainment to manage the initial ***uncertainty about resource allocation***. Instead, it exhibited persistent ***detraining***, maintaining its characteristic sprint-based planning cycles throughout the work process. The agile unit heavily depended on the goodwill of non-agile units to provide resources. It applied ***substituting*** practices when resources were permanently unavailable, and managed ***waiting times*** primarily through ***harmonizing*** practices.

Case [E] focuses on an agile project team tasked with the reorganization of business processes and team structures in the corporate client advisory unit of a banking organization. As in the preceding cases, the agile team operated in an organizational context characterized by rigid annual planning cycles, established and enforced by management as the dominant Zeitgeber.

> **Case [E]:** Ideally, [in the bank], people aim to plan everything well in advance. (…) To me, it's almost a mantra: projects are planned meticulously and long before they begin. (I.E.8)

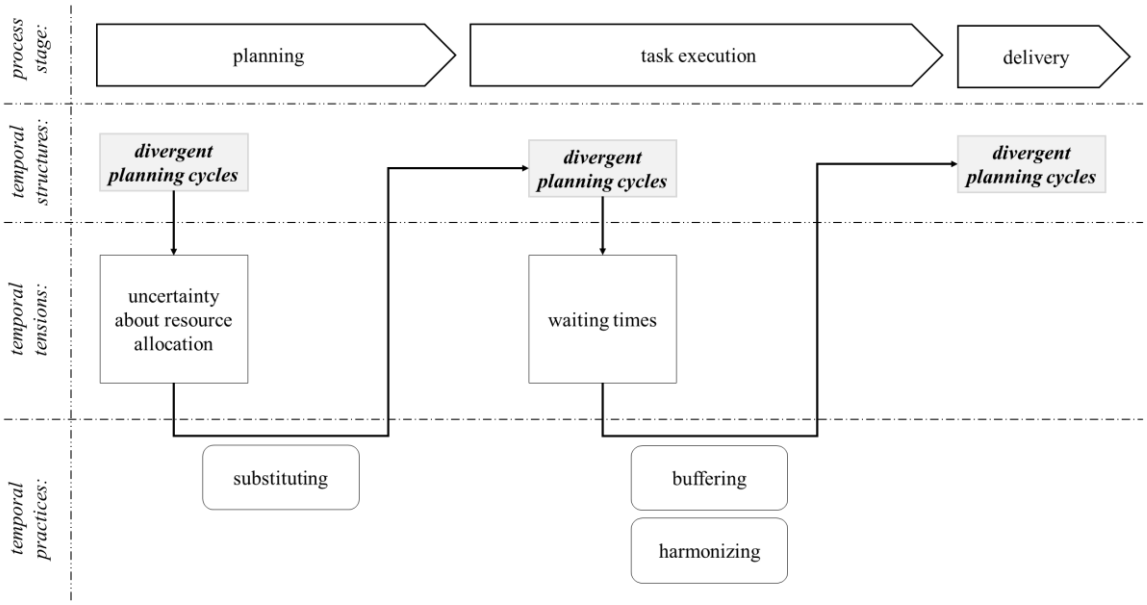

Figure 8: Persistent detraining

Project planning for a given year was routinely completed during the summer of the preceding year. Guided by a strategic roadmap, key initiatives were identified, and financial and personnel resources were allocated accordingly. Agile projects were

typically scheduled for durations of four months. During this period, staff from non-agile units with the required knowledge, skills, and experience were temporarily assigned to the agile project team to carry out project activities. However, the agile project in case [E] was initiated outside this standard planning process. Departing from the established timeline for project planning, management decided in March of the planning year to launch an agile project to restructure the corporate client advisory unit, scheduled to begin two months later in May. As a result, the agile project was, albeit unintentionally, not synchronized with the organization's standard planning cycles from the outset.

This initial detraining had notable implications. Operating outside the formal planning timeline severely limited access to personnel from non-agile units, particularly those required to staff the agile project team. By the time the agile project began planning its activities and resource needs, non-agile units had already allocated their staff to tasks established in the prior corporate planning process. Due to the delayed resource requests, non-agile units declined to release staff, precluding the secondment of employees with the necessary skills and expertise to the agile project. Thus, as a result of enforced detraining from the organization's established planning cycles, the agile project team in case [E] was unable to reduce the uncertainty in resource allocation arising from its divergent planning cycles.

> **Case [E]:** The planning for [the agile project] was rushed through almost like a covert operation. There wasn't any proper planning. (…) It basically came together in March, and it was supposed to launch by the end of May. So, everything had to be checked, from the product owner to the Scrum master. None of these resources was available. Everyone was already tied up with other responsibilities. We have an annual roadmap plan for projects, but [the agile project] wasn't part of it. It was thrown together at short notice in February or March. (…) The [non-agile units] you would normally expect to be involved, such as the technical organization or the credit office, didn't have the capacity. They could simply say, 'That's not in my plan, so I can't take it on.' And again, I don't want to blame anyone, but it was clearly reflected in how the project was staffed. These weren't the people you'd normally assign to a project like that. And that's because it was never properly planned from the start. (I.E.9)

> **Case [E]:** In the final retrospective meeting, the product owner reflects on how the project team was staffed. She explains that before the project began, she had asked the technical organization unit to second a specialist in telecommunication systems, someone the team ended up consulting repeatedly during the sprints. Her request, however, was declined because the employee had no available capacity: "[Name of employee from the technical organization unit] had no capacity. Of course, I would have liked to have had her on the project team." A similar request to assign a full-time employee from the credit office was also rejected. The product owner expresses understanding for the decisions made by these non-agile units: "They had, quite rightly, already laid out a roadmap at the beginning of the year. They'd considered which topics would be relevant for them. And then at some point, a project group comes along." In response, the Scrum master acknowledges her reflections and concludes that agile projects simply need to be planned earlier. At the same time, he points out that the late planning wasn't the team's fault: "But to be fair, we didn't have a chance." (Vignette from observation O.E.90)

Given that most non-agile units were unable to release staff to support the project, which nonetheless had to progress, the agile team was forced to improvise. To assemble a team capable of executing the project activities, the product owner began substituting employees with the most relevant skills, experiences, and competencies with personnel who could be mobilized quickly. She drew on employees who were relatively new to their roles and not yet fully integrated into routine responsibilities, in addition to recent graduates, students, and trainees available during their practical training periods.

> **Case [E]:** The product owner asked, 'Who has availability? Who can temporarily assign someone?' And that's how [I.E.13] and I ended up being assigned, because we happened to have some availability. Sure, they probably also look at who's well-suited for this, who's worked on something similar before. But that didn't apply to [I.E.13] or me, because we're both quite new. (I.E.11)

By substituting employees with the highest expertise for those who were more readily available, the agile project team raised concerns within the corporate client advisory unit, the intended end user of the restructuring concept being developed. Having a vested interest in the quality of the restructuring concepts, the corporate client advisory unit feared that the agile project team lacked sufficient expertise to devise effective solutions. In response to these concerns, the corporate client advisory unit decided to second additional staff to the agile project, notwithstanding the strain placed on remaining staff. Ultimately, this decision benefited the agile project team by allowing it to compensate for its limited experience by integrating additional specialists from the corporate client advisory unit.

> **Case [E]:** When I realized that the whole project hadn't really been planned, that the team would be very small and lacked domain-specific expertise … Well, a lot of people showed up – and I don't mean this disrespectfully – but they were young and had no prior experience with our topic. So, I said, 'It would probably be better if we sent one more person.' And I offered that from my side, fully aware that it would be a big hit for [the corporate client advisory unit]. (I.E.9)

By combining substituting practices with the secondment of additional staff from the corporate client advisory unit, the agile project team reduced the uncertainty in resource allocation during the planning stage and reached operational readiness to execute project activities. In doing so, it followed a methodologically rigorous interpretation of the Scrum framework. At the beginning of each sprint, the product owner proposed user stories from the product backlog, which the team had compiled during a preliminary two-week design phase. The team then discussed the proposed user stories, broke them down into smaller work packages, and estimated the effort required for each. Using these estimates, the agile team autonomously decided how many user stories it could realistically complete within

the sprint. The agile project team could select user stories without being restricted by the long-term planning of non-agile units, thereby maintaining flexibility throughout the work process.

Nevertheless, the limitations of substituting the originally requested personnel with the required expertise with those available on short notice became apparent during the task execution stage. The lack of critical knowledge within the agile project team limited the agile project team's operational independence and constrained its capacity to accomplish its goals autonomously. Thus, the agile project team was compelled to engage in extensive coordination with non-agile units that had previously refused to second staff. Reviving the previous uncertainty in resource allocation, coordinating with these units proved challenging, as their staff remained committed to the tasks defined by the annual plans, limiting their availability for interaction with the agile project team. Consequently, the agile project team encountered considerable difficulties in scheduling timely coordination meetings within the temporal constraints of its two-week sprint cycles. The agile project team often had to await the availability of non-agile unit employees to schedule coordination meetings.

> **Case [E]:** It's only natural that we don't know everything. We come from different disciplines, so of course, we're looking at a subject that has been worked on for years with fresh eyes. That means there's always a need for coordination. And those meetings have to be scheduled. And depending on who you're talking to or want to talk to, their calendars are already pretty full. (I.E.3)

Waiting times were exacerbated by the relatively longer planning and work cycles of non-agile units. Even when non-agile units agreed to support the agile project team despite existing capacity constraints, they seldom provided this support within the two-week sprint intervals of the agile project team. Most employees from non-agile units had little or no exposure to agile practices and were therefore unaware that the agile team's workflows were organized around two-week sprints. Requests from the agile project team were routinely incorporated into the non-agile units' longer planning cycles, which intensified waiting times and amplified tensions within the agile project team. As waiting times jeopardized the completion of sprint objectives, the agile project in case [E], as in the preceding cases, regarded them as a major source of tension.

> **Case [E]:** We work in an agile way. But everyone around us, all the departments we need, don't work agile. So, if we need something from them, their timeline is 'in five weeks'. They just can't estimate it properly. Because we say, 'We're working on this now,' but no one realizes that we only have two weeks. And it can't wait five weeks, or we won't get anywhere. Yeah … Within the team, we're agile. But the rest [of the company] is still operating like before. (I.E.12)

Waiting times were most evident when the agile project team collaborated with the technical organization unit. This non-agile unit was responsible for assessing whether the concepts developed by the agile project team could be technically implemented within the organization's IT systems. In order to prevent work on potentially unfeasible ideas, the agile team had to coordinate regularly with the technical organization unit during the task execution stage. This coordination primarily involved clarifying the capabilities and limitations of existing IT systems and presenting proposed concepts to obtain immediate feedback on feasibility. Typically, the agile project team's user stories included an acceptance criterion requiring the technical organization unit to confirm technical feasibility before the story could be considered complete (illustrated in Figure 9).

To mitigate waiting times occurring in collaboration with the technical organization unit, the agile project team employed harmonizing practices. The agile project team regarded these practices as a second-best solution. During the planning stage, the product owner made considerable efforts to integrate employees of the technical organization unit into the project team (buffering). However, the technical organization unit declined this request because its staff were already fully committed to tasks scheduled at the beginning of the annual planning cycle (see the vignette from observation O.E.90 on page 170). Thus, the agile project team addressed its coordination needs through harmonizing practices, which took different forms depending on the collaborative situation.

User story title: Performance targets

As an employee of the corporate client advisory unit, I want my performance to be evaluated in a way that is comparable to my colleagues, based on the requirements of my role, so that workload is distributed as evenly as possible and additional efforts are recognized.

Acceptance criteria:

- *The type of performance targets (team vs. individual) is defined and, if necessary, differentiated by functional group.*
- *Measurement criteria for each functional group are defined (specific target levels do not need to be set).*
- *The sales staff is involved.*
- ***The technical organization unit has confirmed the technical feasibility.***
- *The works council is involved.*
- *Next steps for implementation are defined.*

Figure 9: Exemplary user story presented in observation O.E.14

In many cases, coordination meetings served to address the agile team's coordination needs effectively. A key difficulty was that the technical organization unit's staff were not always available for coordination meetings on short notice, as required by the agile

project team. As concepts developed during the sprints, feasibility assessment often could not be conducted until the end of a sprint, prompting the agile project team to schedule meetings quickly, ideally on the same day. Due to the considerably longer planning and work cycles in the technical organization unit, meetings on the same day were rarely feasible. By engaging in anticipatory coordination, the agile project team therefore sought to identify coordination needs with the technical organization unit as early as possible, so that meetings could be scheduled with longer lead times. Coordination needs were typically identified and discussed during sprint planning, after which the agile project team promptly contacted the technical organization unit to schedule a meeting. In doing so, the agile project team harmonized the longer planning horizons of the technical organization unit with its sprint-based cycles, while maintaining the distinct temporal structures.

> **Case [E]:** When it comes to these coordination needs, we just have to recognize early on who we need to talk to and schedule these meetings as early as possible. We have these two-week sprints. I know from other companies that they work with four-week sprints, so they have a bit more time. But most of us here are working full-time, so the two-week sprints make sense. The key is just to plan coordination early. (I.E.3)

> **Case [E]:** During the sprint retrospective, the agile project team discusses how to identify interdependencies with non-agile units earlier and more reliably. A team member suggests that the team should not only exchange about interdependencies during sprint planning, but also use the first daily standup meeting after the sprint planning to revisit and refine their ideas: "We could use the first daily to present this to the whole team and get feedback. Like, have we maybe forgotten a department? Because … when you think about it more thoroughly, sleep on it overnight, you come up with different ideas than right after planning, when your thoughts are spread across many stories." The agile project team consents on this suggestion and follows it in the next sprint. (Vignette from observations O.E81, O.E.82 and O.E.83)

Not all coordination needs, however, could be resolved in a single meeting. Some tasks extended beyond collecting information or providing ad-hoc feedback, requiring a greater workload for feasibility assessments by the technical organization unit. In such instances, the agile project had to wait not only for available slots for coordination meetings, but for the technical organization unit to perform the required assessment.

> **Case [E]:** We're dealing with existing IT and whether it needs to be expanded or not. And because of that, our topics always take time. They come with special requirements, too, (…) like the demands placed on software rollouts, protection requirements, IT security, and so on. That's why these topics can't be decided within a few days, even though that's what we'd ideally like. For example, in the current sprint, we'd really like to get a quick thumbs up or down on technical feasibility. But that's difficult. (I.E.3)

For the staff of the technical organization unit, these tasks constituted unplanned work that conflicted with their scheduled tasks. Staff members of the technical organization unit were often willing to assume these unplanned tasks in addition to their planned

activities as long as the resulting workload remained manageable. Thus, they occasionally harmonized their longer planning cycles with the sprint-based planning cycles of the agile project team through the assumption of additional tasks.

> **Case [E]**: The agile project team discusses with an employee from the technical organization unit about implementing individual and team performance targets in the company's IT systems. The employee from the technical organization unit who is responsible for implementation points out that her capacities are largely utilized, but that she is willing to take on additional workload if the task is not too labor-intensive: "I'm fully booked for the next year, but if it's quick, I will do it." (Vignette from observation O.E.55)

Given that their capacities were typically fully utilized with planned tasks, staff members of the technical organization unit often required more time to complete voluntarily assumed tasks than the agile project team had envisioned. In such instances, the agile project team resorted to an additional harmonizing practice. It intervened with management and requested a reprioritization of the technical organization unit's planned tasks. This harmonizing practice was similarly employed when the technical organization unit declined to handle a task at short notice because it exceeded manageable workloads. Upon a successful intervention, management informed the technical organization unit of the elevated priority of the agile project team's task, allowing it to be treated as a planned task and given precedence. While the planning cycles of the technical organization unit and the agile project team remained divergent, task reprioritizations created ephemeral moments of synchrony.

> **Case [E]:** At the end of a sprint, the sprint retrospective takes place. The Scrum master asks for an update on the conceptualization of new performance targets. A member of the Scrum team provides the current status and suggests moving the review of technical feasibility to the waiting column, since its completion heavily depends on [name of employee from the technical organization unit]. The waiting column is a separate lane on the Kanban board. When a task is moved to this column, its progress is no longer regularly discussed during the daily standup meetings. The Scrum master asks when the task can be regularly discussed again. The team member notes that the responsible employee in the technical organization unit will be absent the next day and is very busy with other tasks. A second team member confirms this. Therefore, it is unclear when they expect feedback. The product owner steps in: "[Name of employee from technical organization unit] is always fully booked because management orders so much from him. We need to ask him what's overwhelming him and then speak with management to reprioritize his tasks." (Vignette from observation O.E.59)

Because hierarchical interventions were a standard coordination practice within the organization, the technical organization unit did not interpret them as a critique of their performance or responsiveness. Instead, they regarded them as valuable coordination mechanisms for addressing short-term and unplanned tasks.

> **Case [E]:** Actually, [the colleagues] didn't react negatively at all. Their argument was basically: 'Yeah, some tasks have higher priority from management, so we're handling those first.' The

> prioritization was just rearranged. But then he said, 'I don't care what's at the top of the pile, I'm doing it.' So, we didn't have any problems or issues. (I.E.2)

Although buffering practices were difficult to implement due to the persistent detraining, they were used in collaborations with the corporate client advisory unit and the works council. As the primary recipient of the restructuring concepts, the corporate client advisory unit had a strong interest in co-creating high-quality solutions, in contrast to the technical organization unit, which merely contributed input without being directly impacted by the agile project's solutions. Thus, despite the asynchrony between corporate and project planning cycles, the corporate client unit was willing to second staff to the agile project team (as evidenced in its response to the agile project team's substituting practice on page 171), thus allowing corporate client advisors to be integrated into the agile project team.

A second buffering practice became evident in the collaboration between the agile project team and the works council. The interdependence between the agile project team and the works council arose from the works council's statutory co-determination rights. On the one hand, many of the restructuring concepts developed within the agile project were subject to co-determination. On the other hand, many of the agile project team's activities, such as employee interviews, had to be co-determined and required the approval of the works council. The organization had established a standard process to comply with co-determination rights. This process stipulated that all matters subject to co-determination were to be presented at the works council's biweekly meetings in order to obtain the required approvals. Although these meetings were held regularly, they inherently posed a risk of waiting times. If the agile project team members decided during an ongoing sprint that interviews were necessary, they had to wait until the next works council meeting before conducting them. In the best case, the next meeting took place within a few days. At worst, the agile project team had to wait two weeks, which prevented the timely completion of the task planned for the current sprint.

> **Case [E]:** Actually, every [interview] is technically subject to co-determination. That means if you come up with something in a sprint and say, 'Let's just test this,' whether for two weeks or four, the usual process requires running it by the works council first, waiting for their approval, and only then starting the test. The problem is that we only meet every two weeks. And that two-week rhythm really clashes with the pace of an agile project. (I.E.4)

The agile project team and the works council had already anticipated these waiting times during project setup. Together, they decided to address coordination needs during the task

execution stage ex ante, by pre-approving the instruments for data collection, analysis, and testing in the project setup. This relieved the agile project team from the obligation to present interviews, surveys, or tests in works council meetings for approval before conducting them. Instead, it was sufficient to notify the works council by email about the use of these instruments. This practice buffered the agile project team from intensive coordination with the works council during the task execution stage and allowed it to carry out its tasks independently of the works council's divergent temporal patterns.

> **Case [E]:** At the start of a project, we receive a briefing document that outlines what the project involves, including a note that there will be things like testing or interviews. Normally, every interview would need to go through us for co-determination. Depending on the questions, it's not always just for information. But even those that are purely informational technically have to be brought to a meeting so that every works council member (…) has seen them. That's why we introduced the briefing document. In it, we agree up front that all testing or similar activities planned during the project are pre-approved. We just need to be kept in the loop by email, so we're informed. That way, there's no need to wait for formal approval each time. (…) We handle it through this initial agreement by saying: 'We approve this process as proposed.' (I.E.4)

In conclusion, the fourth pattern shows an agile team that is persistently detrained from the temporal patterns of interdependent non-agile units. It addressed uncertainties in resource allocation by substituting its resource needs with resources that were available on short notice. This created significant dependencies from non-agile units during the task execution, which led to waiting times. In the absence of synchronized planning cycles in the planning stage, buffering practices could only be implemented when they either did not draw on non-agile units' capacities or when non-agile units had a strong interest in participation and co-creation. With non-agile units, to which this did not apply, the agile project team coordinated through harmonizing. Despite persistent detraining, the concluding project observations showed that frequent waiting times did not translate into delayed delivery of project outcomes. Even when tasks were not completed on time within a designated sprint, waiting times did not extend beyond a single additional sprint. The agile project was successfully completed on schedule within four months. Only a few user stories, none deemed critical to the restructuring success of the corporate client advisory unit, were eventually left unaddressed and excluded from the project scope, consistent with a minimum viable product approach. This underscores the effectiveness of the practices used to address waiting times.

5.1.5. Interim conclusion

I conclude the first findings chapter with an interim summary that synthesizes the similarities and differences between the four patterns. Across all four patterns and eight cases, agile and non-agile units exhibited divergent planning cycles, reflecting the different temporal depths embodied in their underlying work practices. Different planning cycles led to uncertainties of resource allocation in agile units during the planning stage. When adhering to their short, sprint-based planning cycles, agile units struggled to secure resources controlled by interdependent non-agile units, which defined tasks and allocated resources over significantly longer planning horizons.

The four patterns illustrate different ways in which agile units addressed these uncertainties in resource allocation. Two of the patterns (patterns 1 and 2) reveal a dynamic interplay of entraining and detraining practices. Agile units entrained to the longer planning cycles of non-agile units in order to secure access to required resources. Becoming tied to long-term task and resource plans, they lost their capacity to respond flexibly to evolving needs or schedule disruptions, prompting them to detrain from non-agile planning cycles and revert to their sprint-based planning cycles once the resources required for task execution had been obtained. Pattern 2 additionally illustrates how attempts by non-agile units to impose two- to three-month planning cycles, intended as a compromise between annual and sprint-based cycles, failed to adequately resolve the flexibility tensions of the agile units. In both patterns, agile and non-agile units sustained their divergent planning cycles during the task execution stage by employing buffering and harmonizing practices, and eventually re-entrained to non-agile planning cycles during the subsequent planning stage. Thus, both patterns illustrate how temporal structures dynamically shift between synchrony and divergence, driven by resource allocation and flexibility tensions.

In contrast, patterns 3 and 4 show how temporal structures are locked into persistent states of synchrony or divergence. In pattern 3, the agile unit did not detrain from the planning cycles of its dominant non-agile Zeitgeber after the initial entraining episode. It attributed the tensions it experienced during the task execution to the limited foresight of non-agile units that continued to operate in shorter planning cycles. Thus, it engaged in entraining these non-agile units to the longer planning cycles it had adopted for itself, leading to the

emergence of a persistent entraining pattern. Despite synchronization efforts, waiting times arose. In pattern 4, the agile unit, albeit involuntarily, was persistently detrained from non-agile planning cycles. It could only resolve uncertainties in resource allocation by substituting required resources with available resources, which gave rise to frequent waiting times during task execution.

Regardless of whether temporal structures shifted dynamically or persisted, agile units encountered waiting times during the task execution stage. Thus, navigating the divergent planning cycles successfully meant mitigating waiting time tensions during the task execution stage. Across all cases, this was accomplished by employing two key coordination practices: buffering and harmonizing. Table 12 summarizes their coordinative effects and refers to illustrations in the eight cases.

Table 12: Buffering and harmonizing practices

Practices:	**Coordinative effect:**	**Examples from cases:**
buffering	coordination needs between two units with divergent temporal structures are shifted from the task execution stage to the planning stage; interdependencies are coordinated ex ante	▪ integration of staff and decision-making authority into the agile units in cases [B2], [B3], [D], and [E] ▪ advance preparation of inputs and approvals in cases [A2] and [E] ▪ creation of slack resources in case [D]
harmonizing	coordination needs between two units with divergent temporal structures are addressed through intensive mutual adjustment and alignment	▪ anticipatory coordination in cases [A1], [B1], and [E] ▪ assumption of additional tasks in cases [B1], [B2], [B3], [C], [D] and [E] ▪ reprioritization of tasks in cases [A1], [A2], [B1], and [E]

Buffering practices shifted coordination between agile and non-agile units from the task execution stage to the planning stage. By coordinating ex ante, agile units were relieved from coordination during task execution, where they found it beneficial to preserve their sprint-based planning cycles. Thus, synchronized planning cycles facilitated the use of buffering practices. Buffering practices included the integration of staff or decision-making authority from non-agile units with divergent planning cycles into the agile units, the advance preparation of inputs that agile units inevitably required during their task execution, and the creation of slack resources that allowed them to respond to evolving resource needs without renegotiating resources with non-agile units. While buffering proved particularly effective in preventing waiting times in agile units, it entailed drawbacks for non-agile units, such as capacity shortages through secondments or

inefficient capacity usage when comprehensive input prepared in advance remained unused. Thus, non-agile units occasionally declined to coordinate through buffering, as illustrated in case [B1], where the non-agile data warehouse unit refused to second staff into the agile project team in order to avoid severe capacity shortages. As buffering practices were not feasible with many non-agile units, none of the observed cases relied solely on buffering for coordinating interdependencies during the task execution stage. Thus, buffering practices were invariably supplemented by harmonizing practices.

Harmonizing practices were intended to intentionally create ephemeral moments of synchrony without altering underlying temporal structures. Agile and non-agile units could exploit these synchronous moments for collaboration, yet maintain their divergent planning cycles. Harmonizing practices included anticipatory coordination, the assumption of additional tasks, and the reprioritization of tasks. Agile and non-agile units engaged in anticipatory coordination to identify both interdependencies and asynchronies due to their divergent planning cycles, for example, by holding regular jour fixes. They could also harmonize divergent timelines by taking on unplanned tasks that emerged on short notice in addition to their planned responsibilities, or reprioritize their tasks. I observed harmonizing practices in all cases. While they were able to reduce waiting times, they did not fully prevent them. Waiting times persisted and had to be endured when agile or non-agile units could neither assume additional work nor reprioritize their tasks. Because these practices intensified coordination efforts in both agile and non-agile units, they were occasionally supplemented with brokering practices.

Chapter 5.1 illustrated how divergent planning cycles generated resource and flexibility tensions in the collaboration between agile and non-agile units. It showed four distinct patterns of how agile units navigated temporal differences that display (1) entraining-detraining dynamics, (2) entraining-detraining dynamics with new temporal patterns, (3) persistent entraining, and (4) persistent detraining. It also identified buffering and harmonizing as key coordination practices to bridge temporal differences by mitigating waiting time tensions. Chapter 5.2 now turns to a second distinction in the temporal structures of agile and non-agile units, whereby variations in clock-time and event-time perceptions translate into differing uses of deadlines to temporally structure development processes.

5.2. Divergent clock- and event-time conceptions: The contested role of deadlines

A second temporal difference surfaces in that agile units perceive the deadlines set by non-agile units as a source of tension. The exemplary quotes presented in Table 13 demonstrate that agile units problematize the deadlines imposed by non-agile units, emphasizing their incompatibility with agile work practices. The underlying reason is that non-agile deadlines constrain both flexibility and responsiveness in agile units, thereby undermining their core agile capabilities.

Why do non-agile deadlines create these tensions for agile units? Deadlines are not inherently a source of tension for agile units, as they rely on them to structure their work themselves. According to the Scrum Guide (Schwaber & Sutherland, 2020), agile sprint cycles are strictly time-boxed. Thus, the end of each sprint represents a binding and non-negotiable deadline by which all tasks planned for that sprint must be completed. Across all cases, I observe that agile units treat the ends of their sprint cycles as key temporal reference points, orienting their activities toward them and monitoring adherence with rigor. Consistent with the Scrum Guide, agile units refrain from setting further deadlines for the temporal structuring of their work processes.

The deadlines established by non-agile units differ from agile sprint deadlines. Non-agile units employ two distinct types of deadlines. First, similar to agile units, they use deadlines to temporally delimit discrete phases in their work processes. These phases, however, differ from agile sprint cycles. They are structured according to the linear-sequential flow typical of plan-based work processes, often span several months, and require all activities within a phase to be fully and finally completed. Consequently, these deadlines impose greater restrictions on flexibility than agile sprint deadlines, which, due to the cyclical nature of agile work processes, permit activities completed in one sprint to be revisited in subsequent sprints. Second, in sharp contrast to agile units, non-agile units use deadlines to define calendrically fixed endpoints for their entire work processes.

Table 13: Tensions evolving around non-agile deadlines in agile units

Case	Exemplary quotes
[A1]	Strictly speaking, the context my team operates in isn't really agile. But I still try to present it as agile to the team. In practice, though, I almost always have a fixed scope and a fixed deadline. For example, with legal requirements that come into effect at the turn of the year, I can only adjust the scope. (I.A1.2)
[A2]	In my opinion, the fixed milestones we still have kind of undermine agile practices. They just go against everything agile stands for. So, even if two or three teams are working as agile as possible, once I set fixed milestones, the whole thing kind of breaks down. Yeah … I have to live with it, but it actually slows things down. (I.A2.2)
[B2]	If, for example, we want to launch a new tariff, like we do for car insurance during the year-end sales period, then it simply has to be ready by a fixed date. And in my experience, that just doesn't fit with an agile approach. Because a product like that isn't an MVP. I can't … I mean, I have a defined scope and a fixed completion date. And for legal reasons, it's just not possible to release something unfinished, because the insurance terms and conditions always have to stay the same. And that's where I see a real tension. (I.B2.2)
[B3]	We have a somewhat unusual situation this year. (…) Specifically, there are parts of our application that we have to replace by a fixed deadline. So, we know: this is the scope, this is the deadline, and we have to be done by then. That means the timeline actually includes both elements. And that's different from, say, being five people in an agile team, like a start-up, just working on a feature, developing it sprint by sprint, and then releasing it. We're not set up that way right now, because we have very strict deadlines we have to meet. (I.B3.4)
[C]	[The company] is structured so that it has two major releases per year. They don't follow an agile process. They're planned well in advance, with deadlines set quite early. And I'd say that it doesn't really fit with the agile approach we're using in the project. I think that's where the biggest challenges lie: the deadlines and the mindset of the broader release management don't align with the project's agile approach. (I.C.11)
[D]	When people ask us if we're working agile, we always say, 'Yes, we're trying to. But there are conditions that just don't really allow us to work in an agile way.' We have these deadlines, and they're really challenging for us (…) because the mechanisms are just so slow. We have to finish development six to eight weeks [before a release], and then we have to test. We're stuck in these traditional waterfall processes. (I.D.1)
[E]	We ultimately implemented this 16-week plan from [name of a consultancy] (…), and I'd say that's how the bank understands agile projects. I think we followed the Scrum framework quite closely. And yeah … how agile the project really was overall? (…) Well, we used Scrum, sure. But apart from that, I'd say the processes and structures are more traditional, more of a classic, plan-driven project with a milestone plan and fairly strict deadlines, where we applied agile methods. That's why I think that we are semi-agile. (I.E.7)

By their function, deadlines limit time. Thus, they embody the finite nature of time and represent temporal patterns that are grounded in a clock-time conception. The different ways in which agile and non-agile units use and set deadlines point to differences in their temporal conceptions. Agile units use deadlines to temporally delimit sprints, but not to constrain the timeframes in which they develop final outcomes. The endpoints of overall work processes in agile teams are inherently subjective and context-specific, with an indefinite number of time-boxed sprints repeated until a satisfactory outcome is attained. The specific use of deadlines in agile units thus reveals that agile units exhibit a clock-

time conception at the sprint level, whereas their broader work processes are guided by an event-time conception. In contrast, by using deadlines to structure both the distinct phases and the entire work processes, non-agile units exhibit a consistent clock-time conception at both levels. This is consistent with the conceptualization of temporal structures in agile and non-agile units presented in Chapter 2.2. Thus, non-agile deadlines, which limit the time for completing sequential activity phases and overall work processes, conflict with the event-time conception inherent in agile practices. The different temporal conceptions embodied in agile and non-agile units' work processes explain why agile teams perceive deadlines established by non-agile units as a source of tension.

In the following chapters, I describe how agile units respond to the tensions that arise from the deadlines established by non-agile units. They focus on seven cases where such tensions were observed[9]. The first response pattern, outlined in Chapter 5.2.1, shows how agile units entrain to the deadlines established by non-agile units, resulting in limitations of their agile capabilities that they seek to offset through various flexibilizing practices. Chapter 5.2.2 presents a second response pattern, where agile units do not entrain to deadlines imposed by non-agile units, while counteracting the resulting coordination deficits through harmonizing practices.

[9] In case [B1], the agile unit was able to operate independently of non-agile deadlines. Management did not impose a fixed project duration but required the agile unit to reapply for resources each year. Likewise, no date had yet been scheduled by which end users were expected to use the software. The software was intended to undergo continuous development until it reached a satisfactory level of maturity.

> **Case [B1]:** I made the decision years ago: [I don't want] a project (…) that drags on for six years, only to check at the end what still needs doing. Instead, I've chosen to work in one-year project cycles. I constantly have to [renegotiate whether the project continues]. Even though I know there's still a lot left to do, and it'll take years overall, I have to keep going back to [the corporate budgeting and planning process]. (I.B1.1)

> **Case [B1]:** We don't have a fixed launch date at the moment because we first need to reach a certain level of readiness before we can deliver the software to our customers and users. Otherwise, it wouldn't make sense to release it. That's actually our advantage right now: we still have the flexibility to change plans easily. (I.B.7)

5.2.1. Pattern 1: Complementary use of entraining and flexibilizing

The first pattern reveals that, at the beginning of the task execution stage, management and management-related functions experience ***uncertainty about the timely completion*** of deliverables in agile units. To reduce this uncertainty, they impose deadlines, which results in agile units ***entraining*** to non-agile deadlines. This entrainment ***limits the flexibility*** of agile units, because the deadlines confine the time for developing final outcomes and thus reduce their capacity to respond to emergent and unplanned tasks. To counteract these limitations, agile units apply ***flexibilizing*** practices aimed at restoring their flexibility. These include work scope reductions, expansions of team size, or the liberation of temporal capacity. While in some cases, these flexibilizing practices effectively safeguard agile flexibility, in other cases, the very act of flexibilizing limits their responsiveness, leaving another dimension of their agile capabilities constrained. Figure 10 illustrates the first pattern.

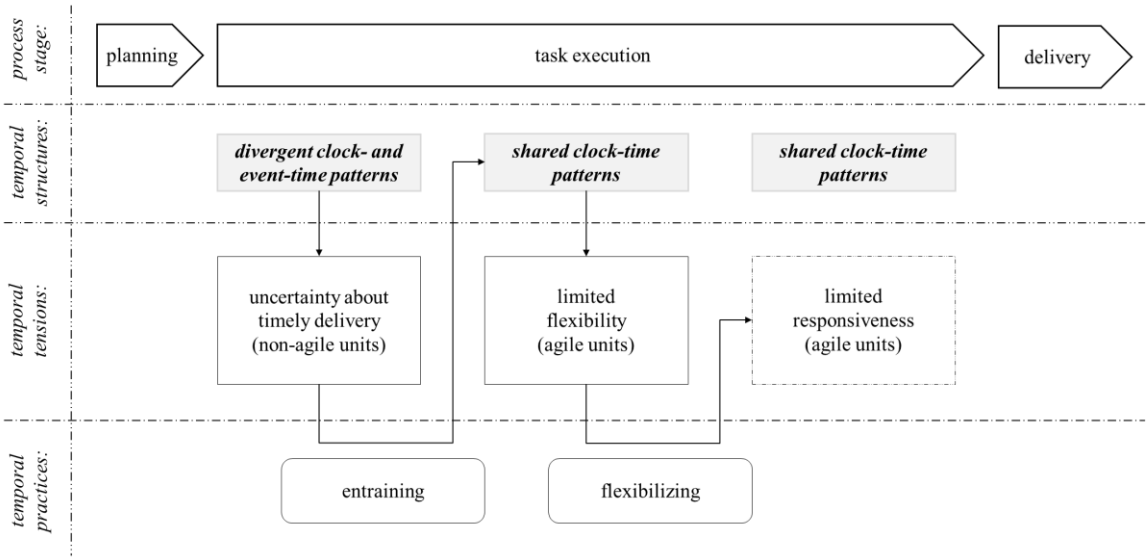

Figure 10: Complementary use of entraining and flexibilizing practices

In cases [A1], [A2], [B2], [B3], and [E], management and management-related functions experienced fundamental uncertainty that outcomes might not be completed on time. This uncertainty was rooted in the serious consequences that delayed outcome delivery could have for organizations. Although it was not inherently linked to the choice of agile and non-agile practices, it was amplified by the cyclical and open-ended character of agile

temporal patterns, which diverged from the clock-time-based temporal patterns characteristic of non-agile units. Management thus used deadlines as control mechanisms to reduce uncertainty about timely completion and to ensure that outcomes were completed by designated dates. Deadlines both constrained the overall work process and often also served to delineate intermediate process steps that had to be completed within it.

In some cases, the deadlines set by management or management-related functions were determined by external actors. In cases [A1] and [A2], these external actors were the regulatory authorities. In both cases, the regulatory compliance unit, acting as the executive arm of management, determined deadlines to ensure the timely completion of regulatory requirements in accordance with the deadlines mandated by regulatory authorities. Failure to meet these deadlines would have jeopardized the certification or recertification of insurance products, thereby prohibiting their sale. Consequently, deadlines set by the regulatory compliance unit were accorded high priority across the organization and were treated as binding by all units involved in implementing regulatory changes, including the agile software development teams in cases [A1] and [A2].

> **Case [A1] and [A2]:** A major factor for us is the lawmakers and the regulatory agency. (…), Specifically, it's about Riester[10]? If you make any changes to a Riester product, you have to submit it to the agency for approval. And that process takes a few months. So, if I submit it now, I can more or less count on being able to launch the product in January. But if I'm four to six weeks later, then that's it. I won't be able to launch in January. And if that happens, the product is basically dead. I can't just say, 'Let's see how far I get, and if it makes it through approval, then we'll go with it.' No. That's not how it works. Either I've submitted the change notification or recertification, or I haven't. That's the reality. We don't have a choice. No matter how agile we try to be, it just doesn't change. That's why I'll always have to work with certain fixed milestones. It's the same for the sales agents. They need the software to be fully updated with the latest legal, tax, and informational requirements by January. Otherwise, they'd be selling in a legally risky situation, and that's simply not an option (I.A.1).

The regulatory compliance unit established two distinct types of deadlines. First, it set deadlines that specified the final completion dates for implementing regulatory requirements. These deadlines were aligned with the organization's software release schedules, as regulatory requirements were implemented through changes to the software used to sell and administer insurance contracts. Consequently, these software changes had to be released before the formal deadline mandated by the regulatory authorities.

[10] The Riester pension (German: Riester-Rente) is a state-subsidized, private retirement insurance introduced in 2002 to supplement the statutory pension system in Germany (see https://riester.deutsche-rentenversicherung.de/)

> **Cases A1 and A2:** But the two big [software releases] are actually the sticking point for us. And accordingly … That's the deadline (I.A2.1)

Second, the regulatory compliance unit also established deadlines for the completion of intermediate process steps. These "milestones" aligned with the sequential stages of plan-driven software development processes, setting target dates for both the conceptualization of technical specifications and the development of the software.

> **Cases A1 and A2:** And for software like this [that maps regulatory changes], I need proper lead time for developing and testing. In cases like that, it has to remain a waterfall process. It's not like a typical agile project where you can say, 'Okay, I didn't finish it in this sprint, so I'll just handle it in the next one. And if it gets delayed a bit, like Berlin Airport, then it gets delayed.' That might work sometimes, but not in cases like this. (I.A.1)

Likewise, in case [B2], deadlines were driven by an external Zeitgeber. Here, management set deadlines based on the launch dates for new insurance tariffs, which had become established market expectations. All software used to sell and administer insurance products had to be updated with the new tariffs by the time each product was launched. As the company offered a wide range of insurance products, the launch dates for new tariffs were spread throughout the year. The most prominent example was the launch of new car insurance tariffs, which, across the industry, occurred in November when customers began renewing their car insurance policies for the upcoming year. Because delays in tariff launches could reduce company revenue, management treated these deadlines as non-negotiable. Consequently, all units, including the agile software development team responsible for premium calculators in case [B2], were required to strictly adhere to them.

> **Case [B2]:** There are currently some fixed launch dates. (…) A typical example is the launch of the new car insurance tariff. The typical changeover period for car insurance is in November. (I.B2.3)

> **Case [B2]:** But we work in an insurance company, which means we have deadlines we have to meet. We have to release the software on the exact day, no matter what. October 7 was one of those days, for example. October 4 was another. And now, December 20 is coming up. We can't miss those deadlines. If we do, all hell breaks loose here. (I.B2.5)

In the remaining two cases, [B3] and [E], deadlines were not determined by external actors but set by management based on economic considerations regarding costs and revenues. In case [B3], an agile team developed software for a new customer portal that allowed customers to access and update personal and contractual information and perform minor administrative tasks autonomously. During development of the new software, end users continued to use the legacy system, rendering the agile team relatively independent from immediate user demands. However, because the new software was expected to

enhance customer satisfaction, and maintaining both systems simultaneously was costly, management established a deadline for replacing the legacy system to reduce uncertainty regarding the time of delivery.

> **Case [B3]:** The project lead reports from the steering committee meeting. He describes the fixed deadline for phasing out the legacy software as a major pain point for the agile team. By September 30, the legacy system is to be shut down; any components not migrated to the new software by then will no longer be usable (Vignette from observation O.B3.2)

Similarly, in case [E], which focused on an agile team that developed new business processes and team structures for restructuring a corporate client advisory unit in a bank, management had established a deadline for the full implementation of new business processes and team structures in the corporate client advisory unit. Management placed great importance on meeting this deadline, as the restructuring was expected to generate substantial cost savings and revenue gains starting from the beginning of the 2022 fiscal year. At the start of data collection in case [E], this deadline was five months away.

> **Case [E]:** During a meeting with the executive board member who is sponsoring the project, the product owner raises concerns that the deadline for implementing the new business processes and team structures in the corporate client advisory unit might not be met. The board member offers his support. The product owner assures him that the team will give it their all. The board member emphasizes: "We are genuinely committed to having the corporate client advisory unit fully restructured by January 1, 2022, with clear communication and direction for employees, so we can really deliver the full impact in 2022." (Vignette from observation O.E.49)

To further reduce uncertainty concerning the adherence to this deadline, management established additional deadlines, each delineating the completion of specific, sequential process steps. These deadlines defined a two-week exploration phase during which the agile project team conducted interviews and surveys with both internal and external stakeholders to identify their needs and expectations. This was followed by a two-week phase in which the team designed user stories based on the findings from the exploratory activities. Subsequently, the operational phase commenced, comprising six two-week sprint cycles during which the agile project team developed concepts for new business processes and team structures.

> **Case [E]:** In 2017, [Name of a business consultancy] said that [agile] projects should take about 16 weeks. Since then, that timeline has been followed to the letter here. (I.E.18)

> **Case [E]:** The project schedule goes like this: first, we begin with the exploration phase, conducting interviews with customers and employees, and collecting relevant data. After that, we move into the design phase, where we define the design criteria and sharpen the overall vision. Then we kick off the sprints. (I.E.6)

While the active agile project work concluded with the end of the sprints, management instituted a three-month implementation phase. This phase encompassed complex and

labor-intensive implementation activities, such as obtaining approval of management and works council, conducting formal staffing procedures, or integrating concepts such as new job profiles or performance targets into the organization's IT systems.

In none of these five cases did the agile units challenge these deadlines. They did not contest the authority of the internal and external Zeitgebers. Thus, across all five cases, the deadlines established by management to reduce uncertainty about the timely delivery of outcomes induced agile units to entrain. In cases [A1] and [A2], the agile software development teams, which developed software for sales agents and the internal customer service unit, respectively, aligned their activities with the biannual software release dates, which ensured that regulatory deadlines were met. In case [B2], the agile software development team, which developed tariff calculators for end customers, treated the institutionalized tariff launch dates as binding. The agile software development team in case [B3], which developed a new customer portal for end customers, structured work around the management-imposed deadline for phasing out the legacy system. In case [E], the agile project team tasked with developing restructuring concepts for a corporate client advisory unit adhered to both the overall project deadline and the intermediate deadlines by which interim work steps had to be completed.

> **Case [A1]:** Strictly speaking, the context in which my team operates isn't truly agile. What I try to do is to present it as agile to the team. In reality, I always have a fixed scope and a fixed deadline. (I.A1.2)
>
> **Case [B2]:** We develop premium calculators, which I'm sure you've already heard about from [I.B2.1]. Right now, we have a few fixed release dates, but we've kept them to a minimum. For example, with tariff updates, like car insurance. November is typically when most people switch policies, so these dates are relatively set. (I.B2.3)
>
> **Case [B3]:** We have a somewhat unusual situation this year. (…) Specifically, there are certain parts of our application that must be replaced by a fixed deadline. So, we know exactly what the scope is and when the deadline is. We have to be finished by then. (I.B3.4)
>
> **Case [E]:** I'd say the processes and structures are more traditional, more of a classic, plan-driven project with a milestone plan and fairly strict deadlines, where we applied agile methods. (I.E.7)

In each of the five cases, entraining to non-agile deadlines supplanted the open-ended and cyclical temporal patterns inherent to agile practices that embody an event-time conception at the process level with temporal patterns that reflected the linear and finite nature of clock-time conceptions in non-agile units. Agile units were left with a constrained timeframe to produce final outcomes. This gave rise to the tensions illustrated at the beginning of the chapter in Table 13, namely the limitations of their flexibility and, in some cases, their responsiveness. In the following, I detail, on a case-by-case basis,

how agile units endeavored to restore flexibility constrained by non-agile deadlines through various compensatory flexibilizing practices, and how in some cases these practices inadvertently reduced their responsiveness.

In case [A1], the agile software development team encountered limitations to its flexibility, as it had to implement many stakeholder requirements into a sales software within six months, constrained by a strict deadline. Often, stakeholders refused to prioritize their requirements, leaving the agile team with a fixed scope of work that had to be delivered by a non-negotiable deadline.

> **Case [A1]:** Just last week, I had a discussion with the board member of the life insurance division. During the steering committee meeting, he stated that he absolutely wants the following requirements implemented by the January release. (…) And I said: "I don't think we can make that happen." And he replied that he wouldn't accept that. We just have to figure out a way to deliver. Then I told the steering committee: 'Guys, this has nothing to do with agile when both the scope and the deadline are fixed! Where's my flexibility?' Well … basically, it just doesn't exist. (I.A1.2)

Limitations on flexibility posed a tension for the agile software development team, as it perceived them to jeopardize the quality of the sales software. From the agile team's perspective, the combination of a fixed scope of work and a rigid deadline created situations in which its capacity was entirely devoted to planned requirements. This left the agile team unable to handle unplanned tasks, such as correcting urgent software defects that were often reported by sales agents at short notice. Without the capacity to handle unplanned tasks, the agile team was concerned that the sales software's quality and reliability would decline because it would no longer be able to resolve defects promptly. By maintaining flexibility, the agile team sought to ensure both the quality of the sales software and the legitimacy of agile practices built on their superior performance.

> **Case [A1]:** So, if the board insists and I am forced to deliver those four additional requirements by the January release, we'll end up with fifty new bugs once it goes live. At that point, nobody will want to use the software because it's riddled with defects. If sales agents can't work with it and think it's terrible … And it really was terrible four years ago! I remember a kickoff event where the board literally said, 'The software is a haunted house!' Back then, we had 450 bugs. We've fixed them step by step. After the last release, I think there were around 20 left, mostly minor issues like missing punctuation. So, the major problems are gone. These are exactly the kinds of discussions I have with [the non-agile units]. My responsibility is to bring a system into production with as few bugs as possible, and I stand firm on that in every discussion. (…) So, I tell them: 'With all due respect to the board, this isn't about agile. Either they want an agile team and value that our software is no longer a haunted house, or they don't.' (I.A1.2)

Because the deadlines were tied to regulatory requirements, failure to meet them could incur sanctions or prohibit sales, rendering them non-negotiable. Consequently, the agile team sought to restore its flexibility by reducing the scope of work.

> **Case [A1]:** Strictly speaking, the context in which my team operates isn't truly agile. What I try to do is to present it as agile to the team. In reality, I always have a fixed scope and a fixed deadline. For example, when legal requirements come into effect at the turn of the year, I can only adjust the scope, not the deadline. (I.A1.2)

Flexibilizing through work scope reductions was effective in case [A1] for two reasons. First, the agile team consistently referred to a prioritization framework that had been specifically introduced for agile teams. Under this framework, tasks were classified into three prioritization categories, commonly referred to as "buckets." Notably, in case [A1], the regulatory compliance unit perceived the prioritization framework primarily as a mechanism for capacity control, ensuring that all tasks, irrespective of their designated priority level, were executed. By contrast, the agile team interpreted tasks in the second and third priority as negotiable and potentially open to postponement.

> **Case [A1]:** The [regulatory compliance unit] has introduced three baskets. The first basket contains the must-have items. No matter what, those have to be implemented. From experience, they can generally estimate that the scope of this basket is manageable. They're actually very good at estimating. For example, when a new product is introduced, they can roughly calculate how many resources it will require and whether we can handle it. The second basket includes requirements that can be implemented if capacity allows. And the third basket, and even the second basket, includes items that can be removed and implemented in the next release. (I.A1.1)

Second, the product owner of the agile software development team actively engaged in ongoing negotiations regarding the team's scope of work. During the biannual planning process, she transparently communicated how many requirements the team could realistically implement with its given capacity. When stakeholders requested new or unplanned requirements, she ensured that they replaced existing items instead of adding to the team's workload. Throughout the agile work process, she maintained continuous dialogue with non-agile units, enabling them to determine which lower-priority requirements could be postponed to accommodate emergent higher-priority tasks.

> **Case [A1]:** I act as a buffer between [our agile team and the non-agile units]. That means I try to allocate requirements across our sprints so we can meet the board's goals. And if we can't, I just let them know. That's simple. (…) Occasionally, something new comes in during the process. I handle that pretty simply. If something new is added, something else has to come out. And the later we are in the release cycle, the fewer things we can remove. It's fairly straightforward. (I.A1.2)

In the regulatory compliance unit, the product owner was known as a skilled negotiator who effectively advocated the agile team's flexibility needs in discussions about the scope of requirements.

> **Case [A1]:** [Name of the product owner], she's a tough negotiator. I have a lot of respect for her, but … she's not easy to deal with. In the end, she usually wins out." (I.A.1)

Negotiations over scope reductions were supported by a distinctive feature of case [A1]: the product owner held a dual role. In addition to serving as the agile team's product owner, she also held the role of business owner for the sales software. Her key responsibility in this second role involved identifying and prioritizing the sales agents' requirements. Thus, the product owner independently determined which requirements of sales agents, and how many, the agile team would implement. She also had the authority to deprioritize requirements and postpone them to future software releases when unplanned tasks demanded priority. Being able to adjust the agile team's scope of work without protracted or contentious negotiations with non-agile units facilitated the enactment of flexibilizing practices. It explains why reducing the scope of work was an effective means of restoring flexibility in case [A1], whereas it proved less feasible in case [A2], despite the agile teams operating under nearly identical conditions.

> **Case [A1]:** As the business owner [of the sales software], I am also one of the stakeholders for myself in my role as product owner. In practice, that makes things quite convenient. I always have to decide. Should this change be implemented in my application? That's my responsibility. If someone wants us to add something and I think, 'Sales won't be able to handle this' or this will hurt performance, then I simply say, 'No, that won't go in.' Politely, of course. But that's the line I take. So, in that sense, it's quite practical. My colleagues sometimes find it a bit annoying, but well, that's how it is. For me, as a product owner, the clear advantage is that I have more flexibility when it comes to handling my own priorities. (I.A1.2)

The dual role further served to prevent reductions in work scope from undermining the agile team's responsiveness. In general, the ease of deferring sales agents' requirements carried the risk that they would be systematically deprioritized, resulting in reduced responsiveness of the agile team to the needs of sales agents as the software's end users. The product owner, however, emphasized that she retained the most critical requirements of the sales agents for implementation. The substantial reduction in defects over recent years, as indicated by an earlier quote, supports the product owner's self-assessment. Thus, the dual role made it possible to balance reductions in work scope with the agile team's need for responsiveness.

> **Case [A1]:** I often have to try to scale back my own requirements to make room for other priorities. Prioritizing is naturally difficult for me, because it always means weighing my own needs against those of others. Sometimes, I manage to push things through … When it comes to the important topics, I make sure they get done. (I.A1.2)

Although the agile team in case [A1] was often able to reduce its work scope, there were instances in which it could not be adjusted. In particular, regulatory requirements could neither be scaled down nor deferred to future software releases. At times, the scope of regulatory requirements alone constrained the agile team's flexibility. To address this

tension, the agile team applied a second flexibilizing practice, complementing work scope reductions with an increase in team size. Departing from the Scrum Guide's recommendation of a maximum of eleven team members, the agile team augmented its capacity by hiring additional staff. This enabled the agile team to implement the large, fixed set of requirements while maintaining flexibility to address unplanned tasks, including the correction of unforeseen software defects.

> **Case [A1]:** When we can't reduce the scope of work, we have to escalate it to the executive board. The board either accepts it, reprioritizes it, or rejects it. And if they reject it, the problem comes straight back to me. Then I have to find solutions and say, 'Okay, we need to expand the team again. Otherwise, we just won't manage. It's simply too much.' That's ultimately why the team kept growing. According to Scrum, a team should have somewhere between three and eleven or twelve members. I don't recall exactly. But we're at 22, double the size! Still, it's necessary because of the fixed deliverables we're required to complete in a single release. (I.A1.3)

However, expanding the Scrum team to enhance flexibility also entailed drawbacks. As the team grew, coordination demands increased, leading the agile team to extend daily standups from 15 to 30 minutes to ensure thorough information exchange and effective resolution of potential impediments. To maintain efficiency during the extended meetings, daily standups became increasingly formalized. For example, team members had to submit impediments in advance, allowing the Scrum master to prepare and circulate an agenda prior to the daily standup meeting. This formalization allowed team members to leave early if they had nothing to contribute, thereby reducing the time they had to devote to communication.

> **Case [A1]:** Managing such a large team is a tremendous challenge. Take the daily standups, for example … Normally, they should last no more than 15 minutes. But unfortunately, we had to organize them differently. In the end, we meet for half an hour every day. With so many people … Even if everyone only says a single sentence, 15 minutes are already gone, and that's before we even get to impediments or things that aren't going well. So, we split it up. We spend the first 15 minutes on the board: What's coming up? Where do we stand right now? Then, for the remaining 15 minutes … I asked the team to let me know before the standup if they have an issue that needs a longer discussion. That means, in the last 15 minutes, what we call the follow-up, we discuss these pre-registered topics. Exactly. That way, I can prepare, know who will be asked, and decide who should stay in the meeting and who can leave early to save time. It really takes a lot of organizational skill to keep 22 people coordinated. (I.A1.3)

Because of the higher communication efforts, enlarging the Scrum team as a means to enhance flexibility was inherently limited. It could be applied only as long as the additional communication effort resulting from team growth was manageable. In case [A1], the agile team seemed to be approaching a critical threshold, beyond which the flexibility gained from team expansion would be outweighed by the added communication effort.

> **Case [A1]:** Of course, we're trying to support by employing even more developers and more people. But the team already has 23 members. That means the team is already far too big for what we're supposed to be doing. (I.A1.2)

In case [A2], the agile software development team operated under conditions nearly identical to case [A1], including the same deadlines. Remarkably, this agile team was unable to recover the flexibility constrained by entraining to the deadlines of the regulatory compliance unit, which ultimately undermined its responsiveness.

Like the agile team in case [A1], the agile team in case [A2] received numerous requirements from various non-agile stakeholders, including the customer service unit as the end user of the software, as well as the product management and regulatory compliance units. As noted earlier in this chapter, the agile team had entrained to the deadlines set by the regulatory compliance unit, which coincided with biannual software release dates to ensure compliance with critical regulatory deadlines. These deadlines imposed a six-month limit on the timeframe for implementing all requirements. With limited options to postpone tasks that were difficult to complete on schedule, the agile team's flexibility was significantly constrained.

> **Case [A2]:** If something is supposed to take effect on January 1st, then it needs to be included in the current [November] release. And that's actually the key pressure point. The next major release isn't until the summer of the following year. If we had more frequent release dates, I could argue differently. That's the issue. It's not even the content that constrains me. What really constrains me is knowing that certain things have to be ready by January 1st, whether for sales purposes or due to legal requirements. That means everything has to be finished by mid-November so it can go live in January and meet legal regulations. And in between, I have no options unless I invest heavily in an additional rollout. That's the situation. That's really the main issue. (I.A2.1)

Notably, while the agile team in case [A1] flexibilized the scope of its requirements, the agile team in case [A2] did not. Unlike case [A1], the agile team in case [A2] perceived no opportunities to negotiate its work scope. Although the requirements assigned to the agile team were categorized into three priority levels, these categories offered little differentiation in terms of urgency or importance from the team's perspective. Stakeholders expected all requirements to be implemented by the next software release and communicated accordingly. Consequently, the agile team was unable to identify lower-priority requirements that could be deferred, limiting its capacity to adjust the work scope in response to changing needs. This fostered the perception within the agile team that both the scope of tasks and the timeframe for completion were rigidly fixed, leaving little room for flexibility.

> **Case [A2]:** [Our stakeholders] prioritize the requirements internally. And then we're handed a list of 50 items, just tossed over the fence, with a prioritization that basically means: everything has to

> be done. (…) They now refer to this prioritization as ‘baskets.’ The first basket contains the absolute must-haves. Without those, nothing moves forward. The second basket includes urgent requirements, which, of course, are also all expected to be completed. And the third basket, as they understand it, contains the rest: requirements we’re supposed to take on to fully utilize our capacity. (I.A2.1)

Without perceived opportunities to adjust the scope of work, agile team members faced high workloads that undermined the effective adoption of agile practices. Early signs of their diminished effectiveness emerged during sprint planning events. Team members ceased making realistic assessments of the work they could complete within a single sprint. Feeling obliged to implement every requirement within the scope of an upcoming software release, they consistently overcommitted to unmanageable workloads. This behavior was not only directly observable during sprint planning, but the agile team also acknowledged and critically reflected on it after the software release.

> **Case [A2]:** At the end of the sprint planning, the Scrum master notes that the team has committed to a total of 216 story points. He emphasizes: ‘216 is more than we can realistically complete in a three-week sprint.” One team member expresses confidence, saying they’ll be able to finish their tickets today. Still, the Scrum master remains skeptical: ‘I mean, it’s too much. Even if we really gain momentum … it’s just not realistic!’ After that, he asks the team to decide whether they really want to commit to that many tasks. No one responds. (Vignette from observation O.A2.7)

> **Case [A2]:** We heard comments like, ‘It has to be done anyway, so I might as well include it in the sprint.’ These kinds of remarks came up quite early on. That was generally how people assessed what could be accomplished. It wasn’t very encouraging. (I.A2.1)

Since the scope of planned work could not be reduced, the agile team’s only apparent option for reducing workload was to decline unplanned tasks. Whereas requirements from the product management and regulatory compliance unit were regularly planned within biannual planning processes, those from the customer service unit were often emergent, and thus communicated to the agile team at short notice. Although technically feasible, implementing emergent requirements during an ongoing release cycle necessitated revisiting completed process steps, which imposed a substantial workload on the agile team. With capacity already fully allocated to planned tasks, the agile team systematically declined emergent requirements from the customer service unit, ultimately forfeiting its responsiveness to the evolving needs of the software’s end users to maintain manageable workloads.

> **Case [A2]:** The customer service unit requested the development of an insurance product, let’s say a funds-based life insurance policy, with a single premium and specific features so we could prepare it for sale. Everything was going great. Then, two months later, they decided: ‘Oh! Maybe we also want a supplementary insurance.’ At first glance, the supplementary insurance seemed like a minor task, but it required its own actuarial data. That meant going back to a point two months ago, when we thought everything was complete, and rebuilding the database schema. We also had to reimplement all the new fields and reconstruct the entire test environment, since we

> effectively added an entirely new section to the database. It was technically feasible, of course, but it set the team back by at least two or three days at that point. (…) However, if they change the original requirements so fundamentally that we have to revisit two months' worth of work, it becomes problematic. (…) We simply don't have the time to make any substantial adjustments two weeks before the release. If we had known six or eight weeks earlier, we could have incorporated these requirements from the beginning. (I.A2.1)

While the agile software development team could decline these emergent requirements to prevent further workload increases, it still had to cope with changes arising within planned requirements. Delays or other deviations that disrupted the original schedule intensified pressure on the team, creating even greater workload peaks to which it could not respond adequately. A concrete example occurred when an enterprise-wide system migration forced the postponement of a scheduled software release, which in turn caused delays in interdependent non-agile units. For instance, the accounting unit delivered the latest quarterly and monthly closing data later than expected; data that the agile team required to develop test cases. Consequently, the agile team also fell behind schedule in preparing testing activities, handing test cases over to downstream testing units several weeks late.

> **Case [A2]:** At the beginning, we were still working on a different schedule. Then, in the summer, (…) the major system migration was postponed by three weeks, even though we were already working on it. (…) Apparently, we didn't raise the alarm bells at that point, and maybe we should take the blame for that, because the whole plan was based on having those three weeks. We placed a lot of trust in the team, and things actually looked quite promising. (…) We didn't know about the backlogs and workloads the migration (…) had caused. Then, at some point, there was a crisis meeting about [the accounting unit]. It was only a few weeks ago, six or eight weeks ago, that we found out none of our topics had been implemented by the accounting unit yet because they hadn't finished the annual financial statements. Then it became clear that we were three, four, or five weeks behind schedule. That meant our colleagues in actuarial and statistics would also get their test cases three or four weeks late. But somehow, it ended up being exactly those three or four weeks. The three weeks we lost at the beginning were also missing at the end. If we had noticed everything three weeks earlier, we would have had an extra three weeks to get everything back on track. (I.A2.1)

Because of the significant delays in testing activities, the time available to address defects identified during testing was substantially reduced. Nevertheless, despite the limited timeframe, all defects had to be corrected before the scheduled software release, resulting in severe workload peaks for the agile team during the final sprints of the six-month release cycle. In an effort to manage its substantial workload, the agile team employed a last-resort flexibilizing practice: it freed up temporal capacity by canceling a significant portion of the ceremonies prescribed by the Scrum framework. This symbolized the extent to which the constraints on flexibility and responsiveness undermined the effectiveness of agile practices.

> **Case [A2]:** At the end of a sprint review, the product owner noted that the sprint contained an excessive number of tasks, none of which could be postponed. Attempts to reduce the scope proved unsuccessful, and the sprint remained overloaded. Consequently, to free up capacity for task execution, the product owner decided to cancel the sprint retrospective and all daily standup meetings, except for a single one necessary to prepare the next sprint review. The team approved this decision. (Vignette from observation O.A2.11)

At this point, the agile team recognized that, through targeted reductions in work scope and the cancellation of agile ceremonies to free up capacity, it had effectively abandoned agile work practices. While this approach appeared to be pragmatic in light of the team's limited flexibility, it was perceived as a substantial tension. The agile team criticized that it could no longer genuinely practice agile methods, and that it became largely unable to respond flexibly to changes and emergent tasks.

> **Case [A2]:** You were involved in the last sprint plannings and reviews. You saw the heavy load we had to carry … what kind of ship we were sailing in the end. We weren't really working in an agile way anymore; we were just being pragmatic. Our goal was to cope with these mountains of work and basically deliver some kind of functional outcome. (I.A2.1)

As in previous cases, the agile software development team in case [B2] experienced limited flexibility, as both the scope of work and the timeframe for its completion were clearly defined. The agile team's scope of work was delineated by detailed requirements to ensure consistency between the tariff calculators, the underlying insurance tariffs, and the software used to sell and administer insurance policies. The completion of these clearly defined requirements was bound by deadlines for the launch of new insurance tariffs, which were dictated by market expectations and could not be negotiated. Thus, the agile team's flexibility was constrained both in terms of time and task content.

> **Case [B2]:** A directive comes from management, and we have to apply a tariff based on the car color blue. That's how it is. It comes from management, and we implement it. That's it. We can't … Some new team members suggested, 'Let's do it differently', or proposed changing the timing: 'Let's do it later.' We said, 'No.' They insisted, 'Yes.' We replied: 'Let management decide.' Then they came back, 'Alright, we'll do it as planned. That doesn't work because the overall process has dependencies. The insurance line units have applications. The sales agents have applications. They all follow the same fixed deadline. All applications must be released at the same time to ensure the new tariff can be calculated with the updated parameters. This is not something we will change. (I.B.3)

Meeting fixed requirements within tight, predefined timeframes periodically placed the agile team under intense pressure, with workloads peaking particularly in the weeks leading up to a deadline. This pressure was directly observable. During the final sprints before a market launch, the agile team deliberately overcommitted to a high volume of tasks to ensure that all required functionalities were delivered on time, fully aware that this would generate considerable strain.

> **Case [B2]:** In the final sprint planning before the market launch, an insurance expert notices that the team now has to handle 252 story points, which exceeds their capacity. He comments, 'That's obviously a huge amount.' He then removes the story points for testing, ongoing tasks, and release preparation. In the end, he notes, 'It still has to get done.' He reassures the team that, excluding tests, ongoing work, and release preparation, the sprint only accounts for about 140 to 150 story points, trying to make the workload seem more manageable. Later, during small-group discussions, backend developers reflect: 'This is going to be a brutal sprint. That's a lot.' After the small-group sessions, the Scrum master emphasizes again: 'As [the insurance expert] already said: We have no choice. We have to tackle this entire workload.' (Vignette from observation O.B2.9)

Notably, the agile team preserved flexibility by flexibilizing both the scope of its work and the size of its team. The work scope reductions differed from those observed in case [A1]. The agile team did not negotiate the scope of requirements with non-agile units, but instead adhered to a core principle of agile practice: delivering a minimum viable product. When developing the premium calculators under intense deadline pressure, the agile team often temporarily lowered quality expectations, reducing the effort required to fulfill the defined scope of requirements. For example, by the time of a tariff launch, customers had to be able to create and download documents that contained individualized offers. The agile team ensured that the functionality for creating and downloading these documents was operational. However, it lowered the quality of the documents' content, which did not yet meet the company's usability standards at the time of the tariff launch. Once workloads eased in subsequent sprints, the team revised the documents to improve user-friendliness.

> **Case [B2]:** During our October release, one of the items that went live was a PDF. That release included a lot of things, but with this PDF, we said: 'Okay, we know it's not perfect from the user's perspective. It's clear enough, but not exactly user-friendly. People will have to think a bit more than we'd like them to.' So, we released it anyway, then polished it over the next week or two, and released the improved version afterwards. (I.B2.5)

The intuitiveness of the agile minimum viable product approach for preserving flexibility prompts the question of why the agile teams in previous cases [A1] and [A2] did not adopt the same strategy. In case [B2], the minimum viable product approach was facilitated by the agile team's capacity to release software on a weekly basis. Quality reductions thus represented only short-term limitations for users, typically lasting only a few weeks. This enabled the agile team to implement all defined requirements by the deadline at lower quality levels, while gradually improving their quality afterwards. In contrast, in cases [A1] and [A2], where software releases were scheduled only twice a year, the constraints imposed by reduced quality levels persisted for users over a much longer period. As illustrated by a team member from case [A2], fixed deadlines became particularly challenging when combined with limited release opportunities. Because quality deficits

could not be corrected promptly, temporary reductions in quality were less viable, offering an explanation for why agile teams in cases [A1] and [A2] did not flexibilize the quality of the software they developed.

> **Case [B2]:** Of course, that's compatible, at least from my point of view. We have deadlines, sure, but the product can always be developed further. Just because you've finished something today (…) doesn't mean it's really finished tomorrow. You can start improving it again the very next day. (…) And you could release every week. You could even release twice a week. (I.B2.5)

> **Case [A2]:** If something is supposed to take effect on January 1st, then it needs to be included in the current [November] release. And that's actually the key pressure point.-The next major release isn't until the summer of the following year. If we had more frequent release dates, I could argue differently. (…) [But] in between, I have no options unless I invest heavily in an additional rollout. That's the situation. That's really the main issue. (I.A2.1)

Flexibility was not restored solely by lowering software quality. In addition to adjusting the quality levels of its deliverables, the agile team temporarily expanded its team size. During data collection, we primarily focused on one agile team developing premium calculators for life insurance products, while two other agile teams within the same line unit developed calculators for legal protection, property, and car insurance. Because the launch dates for different insurance products varied, workload peaks rarely coincided across teams. Consequently, it became common practice for teams experiencing high workloads to receive support from developers in other agile teams. This arrangement provided additional capacity when needed, enabling the teams to manage peak workloads more effectively and preserve their flexibility.

> **Case [B2]:** We're organized by business lines. Sometimes, extra support was needed in certain sub-teams. In those cases, we said: 'Alright, some people from other teams will have to be reassigned temporarily, otherwise we won't be able to meet the release deadline.' (I.B2.3)

While effective in alleviating workload peaks and preserving the flexibility of agile units, the temporary reassignments of developers also produced drawbacks. Frequent movement of staff between agile teams undermined team cohesion and gradually destabilized team structures. Because agile practices were implemented differently across teams, developers required time to adapt before contributing effectively. In addition, increases in team size extended daily standups, sprint planning sessions, and retrospectives, thereby intensifying the communication efforts of agile team members.

> **Case [B2]:** That's when our team structure became unstable. Certain people had to be reassigned. Who would it be? Who gets pulled out of their team, joins a new one, and suddenly has to adapt to a new Agile Manager? Some of the teams grew so large that it became hard to keep track. The daily standups were huge. And then there were all the issues around knowledge transfer, the lack of clear lead roles … (I.B2.3)

In case [B3], the agile software development team was required to meet a deadline for replacing a legacy system imposed by management. The agile team experienced constraints on its flexibility because it was required to deliver a fixed set of software functionalities by that deadline. Although management did not expect fully perfected software by the replacement date, it did require the new customer portal to meet at least the functional standards of the legacy system. Significant deviations from these standards, to which customers had grown accustomed, were potentially detrimental to customer satisfaction and had to be avoided. The non-negotiable scope, combined with the requirement to meet these functional standards within the deadline, resulted in a workload the agile team perceived as excessive, and thus became a major source of strain.

> **Case [B3]:** The project lead reports back from the steering committee meeting. He describes the fixed deadline for phasing out the legacy software as a major pain point for the agile team. By September 30, when the legacy system is to be shut down, all current software components have to be migrated to the new customer portal to avoid decreases in customer satisfaction. He reassures the team that he has openly communicated to management the extremely heavy workload they are currently facing. (Vignette from observation O.B3.2)

The agile team responded to the constraints on its flexibility, as well as the resulting workload increases, by flexibilizing the scope of its tasks. Similar to case [B2], it temporarily reduced the quality of the customer portal. The agile team prioritized the complete implementation of core functionalities, even if they did not fully meet its quality standards. At the explicit request of the project lead, the team suspended ongoing improvements and refinements and postponed them until after the replacement deadline, when restrictions imposed by non-agile deadlines would no longer apply. Until then, the agile team departed from its iterative development and reverted to plan-based practices, executing tasks strictly sequentially and adhering closely to quarterly plans, despite having deliberately detrained from them in the previous year (see Chapter 5.1.2).

> **Case [B3]:** The project lead emphasizes the importance of quarterly planning for the current year [2022]: ‘This quarterly planning … We’re on a really tight schedule, and everything planned for Q1 has to be completed. Otherwise, we’ll have a problem. By the end of Q3, the key functionalities of the software must be implemented so that the legacy system can be shut down. (…) So, it’s better to save your iterations for later and just move on to the next topic.’ (Vignette from observation O.B3.2)

The agile team recognized that its chosen way of flexibilizing was effective in managing excessive workloads, but curtailed its agile capabilities. Team members reflected that the specific task of replacing a legacy system had effectively forced them into a plan-based mode of working, leaving limited scope to respond to emergent customer needs or continuously improve software functionalities. Their perception of this as a tension was

underscored by the hopes and plans they expressed to return to agile ways of working once the deadline had passed.

> **Case [B3]:** What we're essentially doing is replacing technical debt. Once that's cleared, we can shift to a different way of working, one that aligns more closely with what we, as an [agile team], have truly committed to: focusing strongly on the customer, measuring customer feedback, and optimizing based on KPIs [key performance indicators]. We want to move toward those kinds of cycles, so we can develop our features in a more targeted way, without the rigid timelines and quarterly planning structures we currently face. (I.B3.4)

In case [E], the agile team showed little awareness of the tensions arising from non-agile deadlines. While it conceded that the managerially imposed deadlines rendered the project "semi-agile" (see Table 13), it did not explicitly address tensions related to limitations in flexibility or high workloads. However, such tensions surfaced both in interviews with members of non-agile units and through participatory observations of collaborative exchanges. In line with earlier cases, the timeframe for developing restructuring concepts in case [E] was strictly constrained by management-imposed deadlines for both the overall project and intermediate process steps. The tightly scheduled deadlines deprived the agile project team of the temporal flexibility to carry out tasks with the intended level of depth. This became especially evident during the exploration phase, when the agile team sought to elicit stakeholder needs and expectations, which served as the basis for developing tailored restructuring concepts. To this end, the agile team had scheduled a large number of interviews with diverse project stakeholders, including the designated end users, employees of the corporate client advisory unit. However, because the deadline permitted only a narrow window for exploration, the team was unable to conduct all the interviews originally planned. With the exploration phase capped at two weeks by the deadline, the agile team cancelled interviews, especially those with corporate client advisory unit employees, justifying this by the significant time pressure they encountered.

> **Case [E]:** I was invited to an interview once. They asked if I wanted to participate, and I said, I'd be happy to. Then I got an email back [laughs] saying the results had to be presented in three days, and there wasn't enough time for that many interviews. So, I was dropped. I just thought: Typical! (I.E.16)

By cancelling interviews, the agile team effectively employed a flexibilizing practice to reduce its workload and make it manageable under the tight deadlines. Due to this specific work scope reduction, the agile team forfeited the opportunity to gain a comprehensive understanding of the expectations and needs of the corporate client advisory unit. Only four of the more than 60 employees were interviewed. Although the agile team

maintained ongoing communication with the corporate client advisory unit, interactions primarily involved a few employees with whom they had close personal relationships and who responded quickly to requests, thereby supporting the timely completion of project tasks. Thus, from the very beginning, the agile team's responsiveness to customer needs was limited as it did not adequately identify the needs of key stakeholders under the tight deadlines.

> **Case [E]:** What we got from the employee interviews was more of a general sense of their mood. (…) Although … the [head of the corporate client advisory unit] asked me afterwards: Well … if you only talk to four people, that's not really a representative picture of the corporate client advisory unit overall, is it? (…) Sometimes, I really wish there had been more involvement, especially on broader topics affecting the whole corporate client advisory unit. And more attention to average perspectives! Even when we did talk to people, I said, 'Make sure we're not always talking to the same people, so we actually get an average picture. Not just [name of an employee] who's great, yes. But make sure we don't only get her point of view.' (I.E.5)

Likewise, the final deadline for implementing the newly developed business processes and team structures in the corporate client advisory unit constrained the agile project team's flexibility. Pressured by management to deliver solutions within five months, the team was restricted in its flexibility to choose the most appropriate solution. Under the non-agile deadline, it was compelled to prioritize quickly deployable solutions over those that reflected the needs of the corporate client advisory unit. Consequently, flexibility restrictions translated into limited responsiveness to end-user needs. These tensions became evident during the conceptualization of new performance targets for employees. In addition to quantitative metrics, such as sales figures or the number of advisory calls, employees expressed a desire to incorporate qualitative measures, for example customer satisfaction with the advisory services. The agile project team explored options for integrating customer satisfaction as a performance measure, including an interactive voice response (IVR) system that would allow customers to evaluate their advisory experience immediately after a call. However, because IVR had not previously been adopted within the organization, its implementation was considered time-intensive. After presenting their proposals to the board member sponsoring the project, who explicitly emphasized rapid implementation as his first priority, the agile team abandoned the IVR solution, judging it too risky to meet the final deadline. In doing so, they prioritized adherence to the deadline over the development of a user-centered solution for the corporate client advisory unit.

> **Case [E]:** The agile project team presents the preliminary concept for the new performance targets to the executive board member ahead of the sprint review. This presentation is one of the acceptance criteria defined for the user story. The board member responds enthusiastically to the

> concept, which includes both quantitative performance metrics (such as the number of accepted advisory calls and sales successes) and qualitative metrics (such as customer-assessed service quality). Nevertheless, the board member concludes the meeting with an urgent remark: 'Let me emphasize once again … this is very important to me … If there are technical issues that prevent us from implementing these performance metrics by January 1, 2022, we'll need to consider alternatives. I want to create momentum. I want to see a lot of ready-to-use solutions. I don't want to risk losing the 2022 fiscal year because we're still stuck in technical implementation.' During the debriefing, the project team reflects on the board member's final comments. Although team members are generally satisfied with the feedback, one member sums it up bluntly: 'IVR will be dropped!' Given the effort already invested in developing a solution that incorporates qualitative performance metrics, particularly to meet the needs of the corporate client advisory unit, some team members react with dismay and confusion. The same team member clarifies: 'Given the requirement that everything has to be ready by January 1, we have to drop the IVR. You'd need to validate the entire technical setup, and that's a hot-button issue with the works council. The risk that it won't be ready by January 1, which is extremely important to the board, is just too high. I think we have to drop it, even though it's a great idea. But the risk is simply too high.' (Vignette from observation O.E.61)

Overall, the five cases in this first pattern demonstrate how agile units entrained to non-agile deadlines, thereby reducing the uncertainty of timely delivery in non-agile units. Following their entrainment, agile units experienced limitations to their flexibility and increased workloads, which could mutually reinforce each other; tensions they sought to mitigate by using various flexibilizing practices. These flexibilizing practices included work scope reductions, liberations of temporal capacity, and team size expansions. Notably, while these practices helped agile units in all cases to cope with peak workloads, they did not always restore their core agile capabilities. A comparison of cases [A1], [A2], and [E] indicates that work scope reductions further compromised agile capabilities by limiting responsiveness when they resulted in systematic deprioritization of the needs of certain stakeholders. In case [A2], the agile unit could reduce its workload only by declining emergent requests from the customer service unit. In case [E], the agile unit opted to reduce its work scope by cancelling interviews with the corporate client advisory unit. These actions constituted a systematic deprioritization of end-user needs that lowered agile units' responsiveness toward them. In contrast, in case [A1], the product owner's dual role as the key end-user representative allowed her to determine which tasks could be deprioritized without undermining the agile unit's responsiveness. Another comparison of the case [B2] with the remaining software development cases [A1], [A2] and [B3] reveals that work scope reductions by lowering quality levels effectively restored agile capabilities only when rapid release rhythms, as evident in case [B2], allowed to address quality deficits quickly and iteratively. Flexibilizing practices, and work scope reductions in particular, are thus a double-edged sword that can restore the effectiveness of agile practices under specific circumstances but also entail the risk of exacerbating limitations of agile capabilities if these circumstances do not hold.

5.2.2. Pattern 2: Complementary use of detraining and harmonizing

Cases [C] and [D] form the second pattern (illustrated in Figure 11). In this pattern, the release management units acted as the time-setting actors, using deadlines to mitigate ***quality concerns***. While these deadlines were intended to ***entrain*** the agile units to non-agile temporal patterns, they failed to do so. Anticipating that non-agile deadlines would limit their ***responsiveness,*** the agile units ***detrained*** from the deadlines, thereby maintaining divergent temporal patterns. By detraining, they experienced ***uncertainty about the timely delivery*** of outcomes, as non-agile units could reject late submissions. To reduce this uncertainty, the agile unit in case [D] eventually ***entrained*** to the release process deadlines. In case [C], the agile unit continued to detrain and mitigated uncertainty through ***harmonizing*** practices.

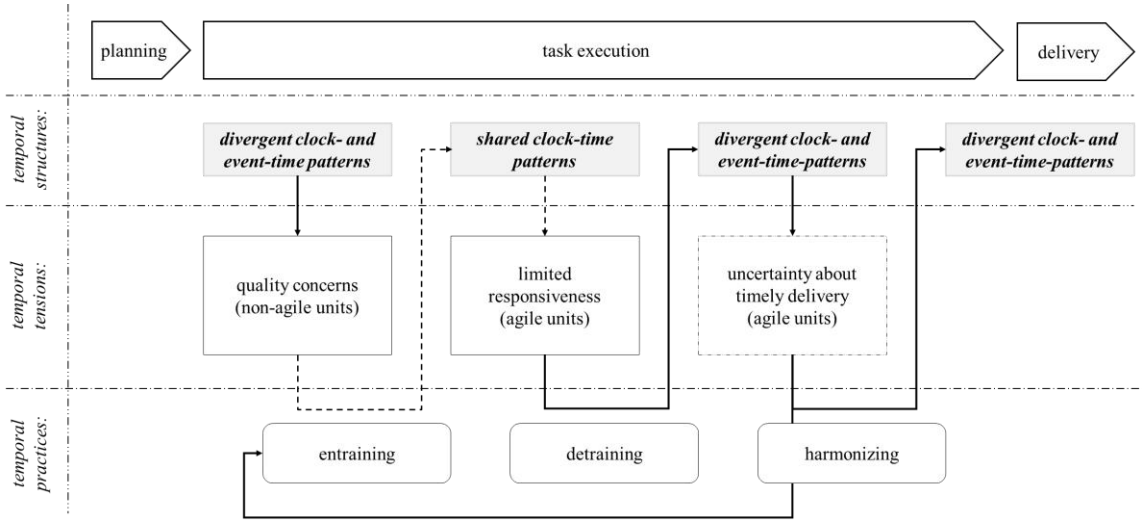

Figure 11: Complementary use of detraining and harmonizing practices

Since the agile units in cases [C] and [D] were organized as projects, there was, in principle, a management-imposed project duration with a defined end date. The agile project in case [C], launched at the beginning of 2020, was originally given a two-year timeline by management, which was eventually prolonged by a year. In case [D], management prescribed a project duration of two and a half years, which the agile project team deemed unrealistic from the outset. By the time of data collection, the agile team had applied for a second project extension. Although the presence of these deadlines

suggests that management aimed to reduce uncertainty about the timely completion of agile projects, they held little relevance for the agile project team.

> **Case [C]:** The project managers, the product owners, and the Scrum master prepare the presentation for the steering committee. They review the project status and include in the presentation that the project completion date, the end of the year 2022, is not at risk. (Vignette from observation O.C.12)

> **Case [D]:** [Management] assumed we would be completely finished in two and a half years. But as soon as we broke ground, everyone involved in the project (…) knew that wouldn't happen. (I.D.1)

The specific deadlines that created tensions for the agile project teams in cases [C] and [D] were those established by the non-agile release management units to govern software release processes. The agile project teams depended on the infrastructure provided by non-agile release management units for software deployment. Access to this infrastructure was contingent on following established release processes, which also entailed compliance with specific deadlines. As shown in Figure 12, these deadlines defined the milestones by which particular activities had to be completed. The deadlines aligned with the typical stages of a plan-driven software development process. First, in case [C], the agile team was required to meet an "end of specification" deadline, submitting the software's technical specifications six months before the scheduled release date. Second, in cases [C] and [D], the release management units set an "end of development" deadline, indicating when all development work had to be completed.

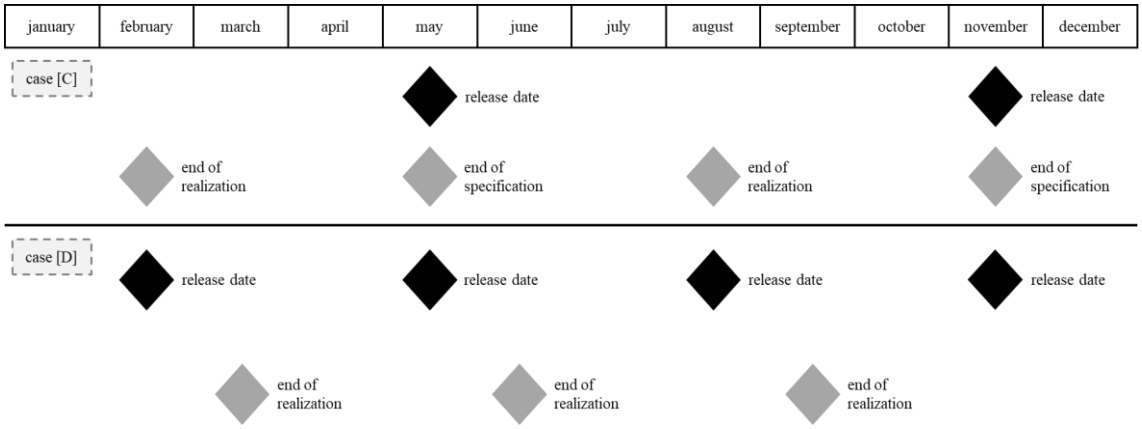

Figure 12: Deadlines of release processes in cases [C] and [D]

> **Case [C]:** I believe the specifications should be completed by the end of November. And we're talking about exactly six months ahead. The bank's ideal planning, specifically from the release management unit, is that specifications are completed six months in advance. (I.C.19)

> **Case [C]:** Major releases theoretically take forever to prepare. The next major release is scheduled for November. The "end of realization" (…) is at the beginning of August. So that's about three and a half months beforehand (I.C.2)

> **Case [D]:** In total, I usually block out seven to eight weeks [before the release] – at least two months – before I (…) get it ready and actually move it into production. And for an agile project, eight weeks with several sprints is quite a lot of time. (I.D.5)

Unlike the deadlines set by management in the first pattern, those imposed by the release management units were not primarily intended to reduce uncertainty about the timely completion of agile deliverables. In contrast to management, which acted as the dominant Zeitgeber in the first pattern, the release management units were not responsible for the organization's overall success. Their responsibility lay in ensuring that software scheduled for release was adequately tested and posed no risk to the stability of the organizational software landscape. Given that testing activities involved a substantial workload, the release management units used deadlines to ensure that adequate time was available to conduct the required tests thoroughly. Consequently, the tension that motivated release management units to set deadlines primarily reflected quality concerns.

> **Case [C]:** With our major release … We have quite a lot of testing work. (…) That means, for the release, especially in units where we have strong project dependencies, we can't really say, we'd offer these (…) releases [without deadlines] that easily. (I.C.20)

> **Case [C]:** Ideally, you're supposed to know a year in advance how your software will work and specify it accordingly, so that everyone has enough time. (I.C.21)

> **Case [D]:** Of course, what has to be managed is the risk that comes with [software releases]. (…) And the most important thing, especially for an insurance company that depends on technology, is that the systems run – that they run stably and as error-free as possible. That's why you need broad test coverage. (I.D.10)

> **Case [D]:** Our major releases (…) always have an integration testing period beforehand. You can assume, and this is a validated number, that roughly 30 domains are involved in a major release. These 30 domains are either specific applications or projects. (…) So, the integration test itself takes quite a while. Our integration testing period lasts about six to seven weeks. It also depends on the time of the year. For example, we stretch the testing during summer holidays, like right now, and shorten it if we're releasing in winter. Often, six weeks is enough. After that, there's usually a week for a consolidation test or a system test in another environment. Altogether, that means I usually block seven to eight weeks, at least two months, before I even start the integration test and get it ready to go live in production. (I.D.5)

In case [D], the release process deadlines affected only collaboration between the agile project team and the release management unit. In contrast, in case [C], they also influenced collaboration with the data warehouse unit. The agile project team in case [C] relied on interfaces developed by the non-agile data warehouse unit, which were delivered to the agile team through software releases. Since the data warehouse was a core application whose modifications could impact the entire software landscape, new interfaces could only be introduced in major releases. Therefore, to develop these interfaces, the data warehouse unit had to adhere strictly to the release management deadlines. Consequently, the agile team was required to submit interface specifications

by the end-of-specification deadline, enabling the data warehouse unit to complete interface development by the end-of-development deadline and submit them for testing on time.

> **Case [C]:** The data warehouse is a central hub in the bank's software landscape. If something goes wrong there, it causes problems everywhere. So, we need clear quality assurance, and we also need defined points in time when data fields and interfaces are changed. That doesn't really work in an agile way, or at least it is very difficult. So, we have two development platforms that are somewhat independent, and their schedules need to be synchronized (I.C.28)

While the release management units set deadlines to allow sufficient time for testing new software, thereby mitigating quality and stability concerns, these deadlines created drawbacks for the agile project teams. In both cases, the agile project teams perceived a tension between their inherently cyclical and open-ended temporal patterns and the linear, finite temporal patterns imposed by the deadlines. In case [D], it criticized that the long lead times resulting from the release process deadlines prevented it from responding to changes quickly.

> **Case [D]:** When people ask us if we're working agile, we always say, 'Yes, we're trying to. But there are conditions that just don't really allow us to work in an agile way.' We have these deadlines, and they're really challenging for us (…) because the mechanisms are just so slow. We have to finish development six to eight weeks [before a release], and then we have to test. We're stuck in these traditional waterfall processes. (I.D.1)

In case [C], the release process deadlines compelled the agile team to specify both the software it intended to release and the interfaces it needed from the data warehouse unit six months in advance. This contradicted not only the agile project team's short planning cycles but also, crucially, its iterative development approach. Finalizing technical specifications half a year before deployment limited opportunities to adjust for short-term requirement changes or incorporate lessons learned from prior sprints. Often, adapting to evolving requirements, pursuing optimization opportunities, or correcting defects required new specifications that could be realized only in a later software release, thereby delaying responses by up to a year. Thus, the early deadlines embedded in the release processes significantly limited the agile project team's responsiveness.

> **Case [C]:** The standard release process at [the bank] basically requires us to have everything fully specified by May for deployment in November. And that really doesn't fit at all with our agile approach, which is to implement things step by step: specify, implement, specify, implement, and so on. Of course, we don't want to throw out what we've already conceptualized. We want to build on it. And that's where things get really difficult, because we deliberately decided against conceptualizing and implementing one big chunk. Instead, we're doing it in lots of small steps. (I.C.12)

> **Case [C]:** But in an agile environment, where you develop things incrementally, [deadlines are] extremely difficult! Especially when authorities like Deutsche Bundesbank or the ECB [European Central Bank] introduce new requirements on short notice. Often, you don't have a complete solution from the start, where you can say: 'This is how we'll do it.' And when small changes come up later, it becomes a huge issue because they weren't part of the original specification. I always find it incredibly hard to know up front what the final outcome should look like. And if you later realize: 'Oops! Maybe we should have done that differently' … well … by then, it's basically no longer wanted and gets postponed to the next release … which is just nonsense! (I.C.21)

In both cases, the agile units refused to compromise their responsiveness. Consequently, they resisted the release management units' efforts to entrain them to the linear and finite temporal patterns by setting deadlines. They preserved their agile capabilities by deliberately detraining from the release process deadlines. In case [C], this meant that the agile project team deviated from both the end-of-specification deadline and the end-of-development deadline.

> **Case [C]:** Wait! There's actually an 'end of specification' deadline before the 'end of realization'. I almost forgot about it because we never really stick to it anyway. (I.C.2)
>
> **Case [C]:** This formal deadline was … Was it at the beginning of June? It's crazy. You just think: This is such an insane lead time!' And it'll take another two or two and a half months [to the release]. In reality, this means that in the meantime, people are rolling out lots of minor releases, production changes, or whatever. Because they need changes in between, they can't always wait six or nine months for something to go live. (I.C.2)

Similarly, in case [D], the agile project team disregarded the end-of-development deadline. It continued to develop software beyond the point at which it was supposed to hand it over to the release management unit for testing.

> **Case [D]:** We have fixed dates for software releases. We start counting backwards from the release date, seven weeks out. One week before the release is the 'Kons' phase, where everything is tested again in an environment that closely mirrors production. Before, there are six weeks of integration testing. So, in the end, you count back seven weeks from the release date, and by that point, development should be complete so the software can be handed over. Normally, development stops at this point. Normally! But not for us! (I.D.12)

Choosing detraining over entraining, however, carried negative consequences for the agile project teams in both cases. In case [D], continuing development beyond the deadline reduced the time available for testing, often forcing the agile project team to suspend or omit certain tests. As a result, software defects occasionally went undetected or could not be fixed in time for the release. In the early stages of the project, when the software was not yet in use, this did not pose a major problem, as any defects had no impact on end users or the broader organizational software landscape from which the software was still largely disconnected. Nevertheless, even at this early stage, agile project team members cautioned that these quality limitations would become untenable

once the software entered productive use and was fully integrated into the organizational software landscape.

> **Case [D]:** I told the team: once [the software] goes into production, we can't just keep tinkering with it. We need to make sure that everything we release has been thoroughly tested over several weeks. That means checking new functionalities as well as making sure the old ones still work. Because once a production bug slips through, we're in real trouble. (I.D.12)

Despite internal warnings, the agile project team initially continued to deviate from release process deadlines, even after the software was in productive use within the insurance line units. As a result, the quality concerns anticipated by some project team members eventually materialized. Because testing phases were shortened, defects could be reliably identified or corrected, leading to malfunctions after new software functionalities were released and disrupting insurance line operations. Thus, by detraining from release process deadlines, the agile project team created a fundamental uncertainty about whether functional software could be delivered to users on the scheduled date. In light of these negative experiences, the agile project team ultimately abandoned the practice of detraining from release process deadlines and began to entrain to them, thereby accepting the resulting limitations on its agile capabilities.

> **Case [D]:** We learned the hard way that sometimes we stretched the integration testing phase too much. We kept developing for too long, and then it came back to bite us in production. Users had to wait two weeks and were told, 'You can't use this function yet.' We reflected on that quite seriously and decided we wouldn't repeat it. Now we strictly follow the six- to eight-week integration test period prescribed by the company, without exception! We drop tasks if needed and no longer debate whether a feature should be added. Once the six weeks are up, nothing else goes on. That used to be debated, but no longer. (I.D.1)

Similarly, in case [C], the agile team's decision to detrain from the release process deadlines led to uncertainty about whether software could be released on time. The agile project team was aware that both the release management unit and the data warehouse could reject specifications or software submitted after the respective deadlines. In such cases, fully developed software could not be released until six months later, or the development of certain software functionalities had to be postponed by half a year because the data warehouse unit would only provide the required interfaces in the subsequent software release.

> **Case [C]:** We're deliberately not always following the existing processes. (…) In theory, they could just say, 'You didn't follow our process. That means you're not compliant, so we won't approve it. Tough luck! You'll have to wait six months before you can get anything live again.' (I.C.4)

Unlike in case [D], where uncertainty ultimately led the agile project team to comply with the release process deadlines, the agile project team in case [C] sought to reduce this uncertainty by using harmonizing practices. It acknowledged that it could not deviate unilaterally or uncoordinatedly from established deadlines; instead, the scope and feasibility of any deviations had to be discussed and aligned with the needs of both the data warehouse and release management units. To achieve this, the team engaged in negotiations over deadline deviations at multiple hierarchical levels, as well as in anticipatory coordination.

The agile project team recognized that negotiating deviations from deadlines without management support could be a futile endeavor. Thus, in an initial step, it sought management approval to detrain from the release process deadlines. In doing so, the team capitalized on the fact that management had intentionally endorsed agile practices to enable faster software delivery in light of rapidly evolving regulatory demands. Since the release process deadlines constrained the agile project team's responsiveness, they curtailed the benefits that management had expected from adopting agile practices. Accordingly, management granted the agile project team permission to deviate from the established release process deadlines.

> **Case [C]:** The colleagues in [the agile project] managed to present their way of working as agile, both within the bank and to the decision-making bodies. And in doing so, they secured something like a carte blanche to operate flexibly. It really is a bit of a carte blanche, because when it comes to some of the deadlines for reporting to release management, they've managed to gain more leeway by simply using the 'agile' label. (I.C.19)

However, negotiating over deadline deviations alone was not sufficient. Although management had granted permission, the agile project team was unable to freely depart from the deadlines that were considered binding by its collaboration partners, namely the data warehouse and the release management units, which continued to enforce compliance.

> **Case [C]:** We received the explicit message: You are allowed to try it out. You are allowed to decide what needs to be postponed. But it wasn't communicated clearly enough to some of my colleagues who are still operating in the 'old world'. It wasn't made clear that the resulting implications, which inevitably follow, are acceptable to management. I can understand these colleagues when they say, 'But you're not allowed to do that.' My response is simply, 'Well, actually, we are.' Maybe more effort should have been made to build a bridge for them. But unfortunately, that's just how it is. (I.C.1)

For the non-agile units, the lack of formalization regarding management's permission for the agile project team to deviate from deadlines proved challenging. As highlighted by a

member of the agile project team in the quote above, management had not transparently communicated the permission to the non-agile units, leaving them largely unaware of the agile project team's leeway. Moreover, management had not established formal rules or processes governing deviations from release process deadlines. Thus, the non-agile units lacked procedural guidelines that provided orientation on the extent to which deviations from deadlines were permissible.

> **Case [C]:** Well, I think the agile methodology is still at a very early stage. Not all the processes … how shall I put it? Well, the corresponding processes for agile projects are not yet as fully developed as they are for the plan-based ones. And yes … But I believe that maybe over time, as people come to understand what is agile and what is plan-based, and recognize the different processes are needed for each, things will improve. Well, every beginning is difficult. (I.C.20)

Without revised processes, the data warehouse unit and the release management unit continued to enforce their established deadlines. Consequently, the agile project team had to engage in negotiations with both of them. Management's prior approval to deviate from established deadlines served as a necessary precondition, enabling the initiation of bilateral negotiations regarding the possibility, scope, and limits of deviations that the data warehouse and release management units would tolerate. These negotiations were not one-time events but recurred repeatedly within each software release.

> **Case [C]:** So, we have to take the initiative and say, 'Look, we can't adhere to this process. How can we still resolve this in a reasonable way? And how can we still obtain your approval, even though we're not following the formal process?' That responsibility has to come from us. (I.C.4)

> **Case [C]:** Deadlines for delivering certain artefacts like specifications or implementation milestones are sometimes simply not feasible. That's why we discuss this regularly with release management, at least once a month, to negotiate certain exceptions, because we are an agile project. We work in sprints and have different deadlines, or rather, we would like to have different deadlines. (I.C.14)

The end-of-specification deadline was negotiated between the agile project team and the data warehouse unit. Adherence to this deadline was critical for the data warehouse unit because it relied on detailed technical specifications of interfaces to plan for and allocate both personnel and temporal resources. Deviations from the end-of-specification deadline created uncertainty regarding the scope of work for upcoming software releases, severely constraining the data warehouse unit's ability to plan effectively and utilize its resources efficiently. Chapter 5.1.1 provides a more detailed discussion of these tensions.

During negotiations with the data warehouse unit, the agile project team sought to reach a compromise that would, on the one hand, provide them with flexibility to deviate from the end-of-specification deadline, while on the other hand, accommodate the data warehouse unit's planning and capacity management requirements. The agreed

compromise involved the agile project team delivering a preliminary specification of interfaces by the scheduled deadline, which could then be iteratively refined after the deadline had passed. The preliminary specification enabled the data warehouse to determine the capacity needed to support the agile project team's tasks. Within these capacity boundaries, the agile project team was able to adapt the specification and incorporate insights gained during ongoing sprints, thereby enhancing its responsiveness.

> **Case [C]:** From a system-technical perspective, having a specification is a prerequisite to being allowed to develop something and move it to the test environment. So, once the code is developed, it needs to be moved to the test environment, and for that, we need approval. Then, our colleagues from the data warehouse unit said, 'Okay, let's handle this more cleverly by creating a so-called change for each sprint. Within this change, we can incrementally update an initial specification. By setting up this change, we're then authorized to develop and transport code.' That was … well … an operationalization of a sprint-based approach that turned out to be very, very helpful. (I.C.3)

The primary objective of the data warehouse unit, maintaining a steady workload for its developers, created flexibility for the agile project team. The data warehouse unit tolerated changes to the initial specifications submitted by the agile project team, provided that developers had not yet begun implementing the software functionalities described in these specifications. Because data warehouse developers implemented the specified functionalities sequentially, many specifications could be refined and adapted over extended periods, sometimes up until shortly before the end-of-realization deadline, by which all development work had to be completed.

> **Case [C]:** Now here's what I believe. If a developer has a hundred tasks scheduled for the end of November, they won't start all hundred at once. To me, it's not one big waterfall but rather multiple smaller waterfalls. The developer needs to be able to work at any time, ideally, continuously from the start of the development period until the deadline, completing everything within that timeframe. For example, if the last task only takes one week and they get it exactly one week before the deadline, that's just in time. That's why we managed to stay more flexible than others. Often, we specified tasks much later than usual. And surprise, surprise … We still always met the release date. Because of this, we never had a situation where someone couldn't work; there was always enough material available. Developers received tasks on time, so there was never any downtime. (I.C.19)

Negotiations between the agile project team and the data warehouse unit resulted in a temporary exemption from the formal early deadlines for submitting technical specifications. However, instead of adhering to these formal deadlines, the agile team was expected to meet later, informal deadlines aligned with the point at which data warehouse developers began implementing the specifications. These informal deadlines were highly dynamic, as they depended on the pace of developers' progress. Consequently, the agile project team often struggled to determine how long it could continue refining its initial

specifications, which created a need for anticipatory coordination with the data warehouse unit, which required significant temporal coordination effort.

The increased need for coordination was addressed through the introduction of a brokering role. The data warehouse unit appointed one employee to serve as the primary liaison with the agile project team. The main objective of this role was to make the dynamic, informal deadlines transparent to the agile project team and to ensure compliance with them. To this end, the broker maintained close contact with the agile team. By regularly attending both agile routine events and meetings of the data warehouse unit, the broker remained informed about the activities and plans of both sides. This enabled timely communication of evolving or shifting informal deadlines, ensuring that the agile team could adapt its activities accordingly.

> **Case [C]:** My role is simply to coordinate between the agile project team and the data warehouse. This includes coordinating and monitoring whether all deadlines are being met. That's probably why I'm mentioned by many people. I'm the one who ensures all deadlines in the project are actually adhered to. I follow up to check if colleagues have planned everything accordingly and point things out, like, 'Look, this is the current release plan. You need to keep that in mind. Development has to be completed by then. Specifications need to be ready by then.' Luckily, we have plenty of meetings in the project. We meet at least three times a week, with standups, our dailies, from Tuesday to Thursday. On top of that, we have planning sessions and retrospectives. And, of course, we also communicate directly outside of those meetings. So, there are definitely enough levers in place to make sure everyone sticks to the deadlines. (I.C.14)

While the agile project team and the data warehouse unit could informally renegotiate the deadline for submitting technical specifications, no such flexibility existed regarding the deadline marking the end of development activities. The end-of-realization deadline not only signaled the conclusion of development but also marked the start of testing. For the release management unit, this deadline was particularly critical, as it ensured sufficient time for tests. These tests were designed not only to verify the functionality of newly developed software but also to detect and prevent adverse effects on the organization's existing software landscape. Whereas the data warehouse unit could accommodate delays due to the sequential nature of its development work, stability testing required all components to be completed simultaneously. Consequently, collaboration with the release management unit, the unit primarily responsible for coordinating testing, offered no comparable flexibility for informal deviations from the end-of-realization deadline. Consequently, the agile project team required formal mechanisms to accommodate deviations from the end-of-realization deadline. One such mechanism was the formal process of late changes. Established by the release management unit, this procedure allowed new software to be developed or existing software to be modified even after the

end-of-realization deadline for a given release had passed. In this way, it enabled rapid responses to critical and urgent requirements, such as the implementation of short-term regulatory changes.

> **Case [C]:** If a change needs to be made after the end of realization date (…), it's considered a late change. This could also be triggered by a requirement from the Bundesbank. It's not just a matter of 'we feel like implementing something now.' Sometimes, the request comes straight from the top. According to the release plan, there's a specific date by which changes must be implemented. If a change comes in after that date, it has to go through a large committee. The request must be submitted formally via email to release management, and it can be escalated all the way up to the unit head. (I.C.17)

Late changes involved a considerable workload, as they required repeating previously completed testing activities to ensure system stability. Consequently, the release management unit approved them only in exceptional circumstances, specifically when the risk to system stability was minimal and the underlying development or modification was both urgent and important. Since the regulatory reporting software developed by the agile project team was not yet in productive use at the time of data collection, late changes posed little risk to the stability of the broader organizational software landscape. Because the software had not yet interacted with existing systems, any defects resulting from late changes were not expected to affect ongoing operations. The agile project team leveraged this argument to negotiate the possibility of developing and modifying software after the end-of-realization deadline, thereby normalizing the exceptional late change process as a routine mechanism within the project.

> **Case [C]:** The project is technically fairly separate from the other software systems. The long-standing systems are, of course, untouchable. They are the holy grail. And heaven forbid something goes wrong! Then everything has to be tested by specialists from every angle. The agile project is more self-contained. It isn't critical for overall banking operations. That's also the argument we make with release management: 'Colleagues, if we need special conditions, please grant them! We're not putting the bank's entire reporting at risk.' But we always have to back that up, which is why the discussion comes up again and again. (I.C.14)

Since late changes by the agile project team posed little risk to the stability of the organizational software landscape, they raised minimal quality concerns for the release management unit and were therefore often approved. This allowed the agile project team to develop or modify software for an upcoming release over an extended period, supporting a more iterative approach and faster responses to change. However, just as deviations from the end-of-specification deadline never became standardized and had to be constantly renegotiated with the data warehouse unit, the release management unit also refrained from defining standard rules for deviations from the end-of-realization deadline.

Consequently, each late change required a new agreement on the permissible delay beyond the original deadline.

> **Case [C]:** Sometimes, we implement late changes with special approval. (…) But we always have to negotiate the deadlines. (I.C.2)

For each late change, the release management unit reserved the right to assess its potential impact on overall system stability. Depending on the perceived risk, the timeframe within which the agile project team could deviate from the end-of-realization deadline varied. The risk assessment considered, among other factors, whether the software affected by the late change interacted with other systems in the organization and the extent to which prior testing by the agile project team indicated a reliable implementation. As a result, approvals for late changes remained case-by-case decisions and required ongoing negotiation with the release management unit.

> **Case [C]:** Most of the time, it's about looking at dependencies. What can still be allowed? What counts as a minor change we can handle? And what would affect other software? The key question is: How long do I still allow late changes? Where's the cutoff point? And when do I just close the door from a release management perspective (…) so that I can deliver a more stable version into production. (…) Usually, it's a discussion with the business unit and the project lead. How far can we go and still accept changes? That decision mostly depends on test progress and system stability. If I see there are still plenty of defects in an agile project and testing hasn't moved far enough, then I close the door quickly. But if the system already looks stable and everything has been thoroughly tested, then I might say: 'Let's keep the door open a little longer.' In the end, it also comes down to experience and gut feeling – how far you're willing to go in taking on change. (I.C.27)

In collaboration with both the release management unit and the data warehouse unit, the agile project team could deviate from established deadlines only through repeated negotiations. While these negotiated deviations helped preserve the team's responsiveness, they increased the required coordination effort. Team members involved in these ongoing negotiations perceived the resulting coordination effort as burdensome. During sprint retrospectives, deviations from the established deadlines of the software release process were occasionally identified as a source of tension, prompting some project team members to advocate for stricter adherence. Consequently, deadline negotiation emerged as a relatively fragile practice, employed only as long as the agile project team judged that the benefits of preserving its agile capabilities outweighed the costs of increased coordination.

> **Case [C]:** During a sprint retrospective, a team member who was recently assigned the product owner role criticizes the heavy coordination required when renegotiating deadlines. She explains that late changes require intensive coordination with upstream units. In her view, this tension could be reduced by adhering to the regular deadlines defined by the software release process, rather than seeking deviations, especially when implementing new data fields; 'Maybe, when it comes to new data fields, we could just stick to the normal end-of-specification deadlines that exist in

> this company.' Another team member asks how she defines the normal deadline. He points out that if it means the standard release deadline, meeting it would be impossible: 'By normal, you mean within the company? That's basically utopian.' The product owner clarifies that she does not expect the deadlines to be met for all data fields, only for those that require coordination with upstream units. She emphasizes again that simply following the standard process would reduce the coordination effort. Another team member sighs and concludes: 'That's the dilemma of Waterfall processes across the bank versus the agile approach in our project.' (Vignette from observation O.C.26)

Overall, the second pattern demonstrates how agile units responded to the tensions caused by non-agile units by coupling detraining with harmonizing practices. To preserve their responsiveness, agile units deliberately detrained, choosing not to comply with release process deadlines that governed their collaboration with the release management units and, in case [C], with the data warehouse unit. As detraining generated uncertainty regarding the timely release of software, the agile team in case [D] ultimately ceased to detrain and began to entrain to the release process deadlines. In case [C], the agile unit reduced this uncertainty through harmonizing practices, engaging in negotiations over deadline deviations, and in anticipatory coordination to determine the extent of deviation that allowed it to maintain responsiveness while remaining tolerable to the release management and the data warehouse units. Both cases illustrate that detraining from non-agile deadlines entails costs, which manifest either as the materialized uncertainty of timely software delivery or as the increased temporal coordination effort required to implement harmonizing practices.

5.2.3. Interim conclusion

Chapter 5.2 focused on the tensions that units experienced when confronted with deadlines set by non-agile units, which conflicted with their event-time conception at the process level. It identifies two distinct responses of agile units to such tensions. Unlike in the previous chapter, agile units did not engage in recurring dynamics between entraining and detraining. Instead, they committed more permanently to either entraining or detraining and attempted to mitigate the respective drawbacks of these temporal structuring practices by complementing them with flexibilizing or harmonizing. In the first pattern, agile units entrained to the deadlines of non-agile units and compensated for the resulting loss of flexibility by flexibilizing work scopes, team sizes, and temporal capacities. In the second pattern, agile units compensated for the disadvantages of

detraining, namely the uncertainty about whether they may participate in a software release, through harmonizing. While flexibilizing practices are newly identified in this chapter, harmonizing practices are similar to those in Chapter 5.1. Agile units sought to sustain harmonious collaboration with non-agile units despite disregarding their deadlines. This is achieved, as in Chapter 5.1, through anticipatory coordination as well as ongoing negotiations about the extent to which non-agile units tolerate deviations from deadlines. Figure 13 illustrates the two approaches.

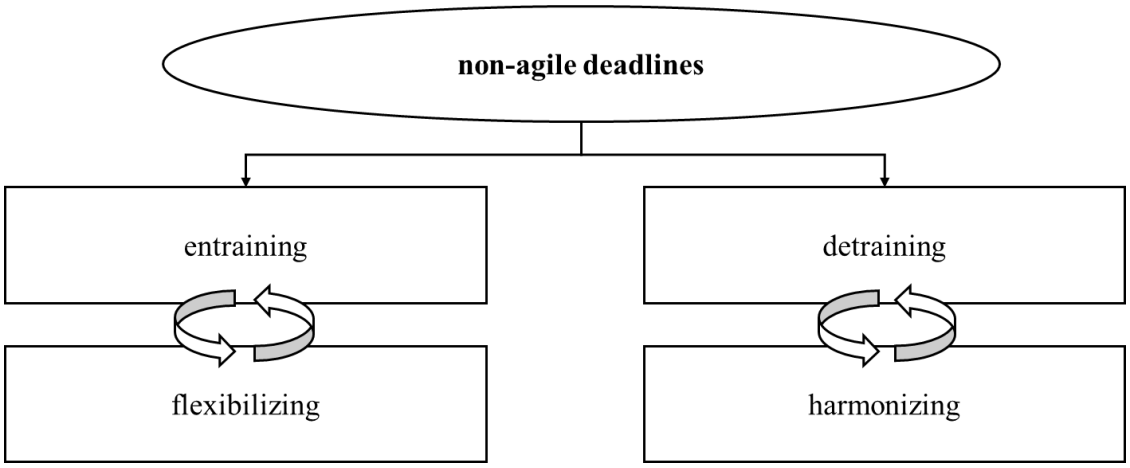

Figure 13: Complementary use of practices to respond to non-agile deadlines

Why do agile units adopt these different approaches? The key contextual distinction between the two patterns lies in the Zeitgeber. In the first pattern, management or influential external actors, such as the market or regulatory authorities, serve as Zeitgebers for the agile units. In the second pattern, release management units assume this role. These two Zeitgebers differ, first, in their power within the organization, and second, in the underlying tension they seek to resolve by setting deadlines. Given the power of management, it is not surprising that the agile units in the first pattern entrained to non-agile deadlines. However, the release management units, though not occupying the same hierarchical position, still hold considerable power over agile units because they control the release infrastructure on which agile units depend. The choice of different responses may therefore be shaped less by hierarchical power alone and more by the underlying tension that each Zeitgeber seeks to address.

The central tension for management in the first pattern is the uncertainty that agile units' outcomes will not be delivered on time. Deadlines, sometimes determined by external actors, are tied to revenue and cost considerations. Even minor deviations may result in

economic disadvantage for the organization that management must absorb. The core tension for the release management unit in pattern 2 stems from the uncertainty over whether released software may adversely affect the organizational software landscape. Unlike management, release management can prevent this uncertainty from materializing into actual disadvantages, even when agile units disregard deadlines. They achieve this by refusing to approve software submitted after the deadline, thereby shifting the tension from themselves onto the agile units. The extent to which agile units are willing to tolerate this uncertainty rests in their own discretion. This is reflected in the behaviors of the two agile units in this pattern: one entrained to release management deadlines in response to the displaced uncertainty, while the other mitigates it through harmonizing practices while maintaining detraining. Accordingly, the ability of agile units to choose detraining depends on the extent to which non-agile units can deflect their potential negative consequences and shift them back onto the agile units.

This chapter has shown how agile units address the tensions arising from non-agile deadlines, which embody finite temporal patterns that conflict with their inherently open-ended temporal patterns, by complementing entraining practices with flexibilizing practices and detraining practices with harmonizing practices. The next and final findings chapter turns to the third central difference between the temporal structures of agile and non-agile units: their divergent release cycles.

5.3. Divergent release rhythms: Clashes of speed and perfection

Divergent temporal structures also become salient at the conclusion of work processes. During the delivery stage, tensions arise from the differing release rhythms of agile and non-agile units, that is, the preferred cadence at which work outcomes are made available to end users. These rhythms are shaped by the frequency of potential release opportunities, such as software deployment dates or, more broadly, any organizational event through which the outputs of agile units are put into productive use. Slow release rhythms imply that organizations provide only a few release dates per year. Thus, output can be made available to end users less frequently and is therefore perceived as being

delivered more slowly. In contrast, fast release rhythms imply that organizations offer many release dates per year, allowing outputs to be delivered more quickly. While the term release originates from software development, in this chapter, I use it more broadly to refer to the point at which work outcomes are made accessible to end users, including the implementation of outcomes from the agile restructuring project described in case [E].

Agile work practices, as defined by their underlying values and frameworks, emphasize the rapid delivery of work outcomes to end users. In line with this principle, the agile units across the eight cases aimed to release their outcomes every two to three weeks. However, they were not always able to determine the frequency of release dates independently. During the delivery stage, agile units depended on the release infrastructure of the organizations in which they operated, particularly on centralized release management units responsible for coordinating corporate release processes.

None of the five organizations from which the eight cases were drawn operated with a single, unified release process. Instead, each organization maintained two distinct types of releases: minor releases and major releases. Minor releases enabled the deployment of outcomes that were not interdependent with non-agile units, such as software components without dependencies on the broader IT landscape or restructuring concepts not subject to codetermination requirements. These releases typically required minimal effort and could therefore be executed at the discretion of the agile units, allowing deployment in short intervals. In contrast, when software developed by agile units affected the broader IT landscape or when restructuring concepts required review or approval by the works council, deployment had to occur through major releases. Major releases were resource-intensive, involving extensive testing and formal approval procedures, and as a result, took place less frequently. While some of the eight cases primarily relied on minor releases, others were more dependent on the less frequent opportunities provided by major releases. Table 14 compares the duration of agile sprint cycles with those release rhythms that were most relevant for the release of outcomes in each case.

Case [B1] must be excluded from the comparison of release cycles. At the time of data collection, the agile project in case [B1] had not yet deployed software and was therefore not yet reliant on the release infrastructure of the organization in which it operated. Thus, it remained unclear whether the agile project was ultimately able to plan and execute

Table 14: Agile and non-agile release rhythms

Case	Rhythm of agile sprint cycles		Rhythm of (dominant) non-agile release cycles	
[A1]	2 weeks	We have two-week sprints. (I.A1.3)	6 months	From a product perspective, particularly when it comes to regulatory topics, everything has to go live by January 1st. That's a release date we absolutely have to meet. The mid-year release focuses more on backend features. Those two releases are still in place. (I.A1.1)
[A2]	3 weeks	We have adopted Scrum: three-week sprint cycles (I.A2.1)	6 months	Right now, we still have two … I always talk about two big releases that we have, which also include model changes. And they go through regression testing, too. But basically, there are just two of them: one in the spring, one in the fall. And those two release dates include the most important stuff. (I.A2.1)
[B1]	2 weeks	We have two-week [sprint cycles] in our unit. (I.B1.5)	not yet known	We don't have a fixed go-live date right now because first, we need to reach a certain stage before we can actually deliver the product to the customers or users. (I.B1.7)
[B2]	3 weeks	We have three-week sprints. They work pretty well for us. (I.B2.3)	1 week	[We] could release every week. [We] could even release twice a week, provided the change has been requested and approved. (I.B2.5)
[B3]	2 weeks	We work in two-week sprints, so our cycles are relatively short. (I.B3.3)	1 week	A Scrum team member reports on his tasks: "We had a release today, but it was only focused on tracking. There's nothing we can actually show. Next week, we're planning a release for 'personal data' and 'contact information.' We worked on the layout and improved the wording. (Vignette from observation O.B3.5)
[C]	2 weeks	I have an IT project that I want to run in two-week sprints and ideally also go live in two-week sprints. (I.C.13)	6 months	Basically, with every release we have at our bank, with every major release, we continue to develop the software. And we have it twice a year. (I.C.19)
[D]	3 weeks	Of course, you want to release as soon as possible after each sprint. (I.D.10)	3 months	And our major releases take place roughly four times a year. It's not always tied to quarters, how they are scheduled. But it always happens four times a year. And the [agile project] participates in almost every release (I.D.5)
[E]	2 weeks	We have these two-week sprints. (I.E.3)	3 months	The slowdown happens because of the staffing topic. That just takes a long time. You can easily count on two months before we even get to implementation (I.E.4)

minor releases independently, or whether it would be constrained to the three major release dates[11] referenced by the non-agile units with which it collaborated.

In two of the eight cases, [B2] and [B3], the agile units were able to release work outcomes in alignment with the rhythms of their sprint cycles. In both cases, the agile teams developed software, namely insurance premium calculators in case [B2] and a customer self-service portal in case [B3], largely independently of the broader software landscape of their organization. As a result, they were able to plan and execute software releases according to their own needs, without being constrained by the company's three annual major release dates. In case [B3], observations of the biweekly sprint reviews indicated that the agile team was often able to release software functionalities on a weekly basis. In case [B2], the agile team had to account for the organization's major releases, as the introduction of new insurance premiums required synchronized updates across several software applications, including sales, administrative, and premium calculator software. Nonetheless, the agile team was able to release new functionalities independently and on a weekly basis, thereby maintaining a release rhythm consistent with the goal of delivering working software at the end of each sprint.

In the remaining cases, [A1], [A2], [C], [D], and [E], the agile units were unable to release outcomes in alignment with the rhythms of their two- or three-week sprint cycles. The agile teams in cases [A1], [A2], [C], and [D] developed software applications that were tightly integrated into the broader software landscape of their respective organizations. In cases [A1], [A2], and [D], the agile teams depended on non-agile units that provided specialized software components. These components needed to be integrated with the sales and administration software developed by the agile teams to ensure full usability for end users. In case [C], the reporting software developed by the agile project team strongly depended on the data warehouse unit, which provided technical interfaces to connect the regulatory reporting software with the organization's data warehouse. Across these four cases, agile units could not develop software independently from the broader landscape. Thus, they were tied to their organization's major releases, which occurred biannually or

[11] The organization-wide frequency of major releases in the organization where cases [B1], [B2], and [B3] were situated was described by an employee from a non-agile unit in case [B1]: "There were components that were deployed immediately, but when it came to components that really had broader impact, there were three major releases a year, outside of what we call the frozen zone. I'm not sure if you're already familiar with [the releases] in our company. We can't deploy throughout the entire year. (…) That's an internal policy at the [insurance company]. (I.B1.2)

quarterly, resulting in release rhythms that were substantially slower than those envisioned by agile practices. Similarly, in case [E], the agile team was unable to make most of its outputs available at the end of each sprint. Many restructuring concepts, such as new role profiles or performance targets for employees in the corporate client advisory unit, were subject to internal revision and approval processes involving both the works council and management. Because these procedures could not be completed within a single sprint, the agile team in case [E] primarily worked toward a single major release, internally referred to as a "big bang" launch. Consequently, delivering outcomes according to the rhythm of the agile project team's two-week sprints was precluded.

This chapter focuses on the cases [A1], [A2], [C], [D], and [E], in which differing release rhythms are evident. Chapter 5.2.1 first explains the fundamental issue underlying the differing release rhythms of agile and non-agile units, namely, conflicting priorities of speed and perfection. It shows that this conflict was initially resolved at the expense of the agile units, which must entrain to the slower rhythms of major releases. Chapter 5.2.2 then illustrates how, in two of the five cases ([C] and [D]), the agile units negotiated with non-agile release management units to accelerate the major release rhythms, thereby developing new temporal patterns.

5.3.1. Pattern 1: Entraining

Figure 14 illustrates the first pattern. It is based on an inherent conflict between time urgency and time patience. The pattern shows how agile units, who have inherently fast release rhythms, in response to ***quality concerns*** raised by time-patient release management units, entrained to their slower release rhythms. By ***entraining***, they abandoned their own faster rhythms, which led to ***limitations of speed***.

This chapter begins by outlining the conflict between speed and quality, which shaped the tensions that arose from divergent release rhythms in both agile and non-agile units. The divergent release rhythms of agile and non-agile units reflect their distinct temporal orientations. As outlined in Chapter 2.2, agile practices accommodate higher levels of time urgency, whereas plan-based practices are associated with greater time patience. These conceptualized differences of their temporal orientations are substantiated by the

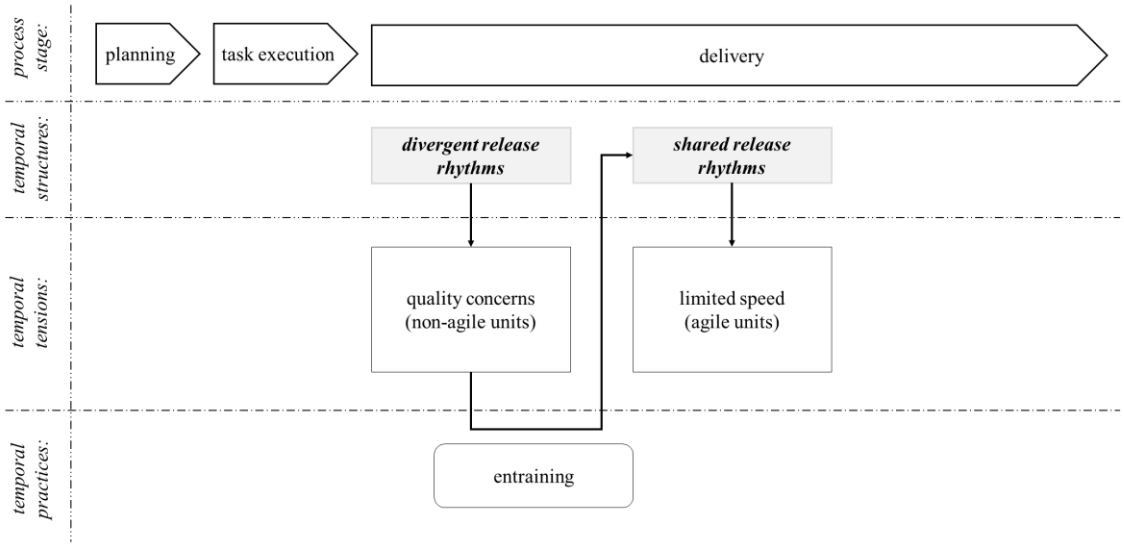

Figure 14: Entraining

empirical findings of this study. In the five cases where agile units were unable to release their outcomes according to their preferred rhythms, I observe diverging priorities regarding the speed and quality with which agile and non-agile units sought to deliver outcomes. Table 15 presents illustrative evidence of these contrasting priorities.

Agile units prioritized rapid delivery and accepted lower initial quality levels in exchange for faster development, aiming to improve quality iteratively through continuous customer feedback and frequent releases. In contrast, non-agile units preferred a slower delivery pace to ensure a higher initial level of quality, which they associated with greater stability of the solutions developed by agile units. In the agile software development contexts of cases [A1], [A2], [C], and [D], quality was understood as software that was as error-free as possible and fully functional for end users. It further implied that new or modified software did not compromise the functionality of the broader software landscape. In case [E], which involved an agile restructuring project, solution quality was primarily assessed by its ability to meet managerial requirements while also safeguarding employees' interests, as represented by the works council. In this context, solution quality required the formal approval of both management and the works council.

Table 15: Conflicts of speed and quality in agile and non-agile units

Case	Exemplary quotes for speed-perfection tradeoffs
[A1]	We try not to deploy every month, but more like once every three months. The main reason is that we need more time to stabilize things. Before rolling out a new feature, we usually set aside two to three weeks just for bug fixing—no new requirements, just focusing on cleaning up issues from the last feature. Because the more new stuff we add, the more it piles up, and then we risk going live with a sluggish application. Sometimes we even have to 'borrow' the UAT environment for that, which management really doesn't like. And of course, it adds to the cost. (I.A1.1)
[A2]	We're not yet at the point where we're continuously delivering MVPs, sprint results, and their follow-up developments into production. Right now, we still have two … I always talk about two big releases that we have, which also include model changes. And they go through regression testing, too. (…) But those two big releases are really the main bottleneck. (I.A2.1)
[C]	In the agile project, we said, 'We want to deliver quickly. And if a bug slips in under pressure, then we'll just fix it afterwards. (I.C.1)
	Traditional software development, especially when it comes to regulatory reporting, requires thorough testing cycles, lots of coordination, and a high level of stability. In that kind of environment, there are just too many dependencies. Agile, on the other hand, works best in settings where fast feedback is possible and where regular adjustments can be made. Ideally, you're working in one- or two-week sprints and deploying at the end of each. But that simply doesn't work when you're dealing with things like nightly batch processing in banking systems, where a small change at the front end can have a knock-on effect across the entire bank. (I.C.28)
[D]	If we were strictly following Scrum, we'd ideally release after every sprint. But that just doesn't really work in the kind of stable, tightly controlled environment you find in an insurance company. (I.D.4)
[E]	I think the non-agile teams are still a bit skeptical and unsure, probably both. Agile just wasn't part of their picture here before. And I think the skepticism comes from early doubts that this kind of approach would actually deliver results. People were used to traditional project management, where the goal was to end up with a complete, polished solution. That doubt around the iterative model is still around – the sense of, 'So, we're just going to throw something out there and see what happens.' (I.E.2)

The non-agile emphasis on stability played a significant role in shaping the release rhythms that became established in the organizations where agile units operated. Across all software development cases, maintaining the stability of the organizational software landscape was a key priority. The emphasis reflected the critical role of software in banks and insurance companies, which relied on it to execute core operations. Products were delivered exclusively through software, and advisory, sales, and service functions were also largely supported by software applications. Because instability of software could severely disrupt day-to-day business operations, these organizations implemented extensive testing procedures, including usability tests to ensure proper functionality and integration tests to safeguard against adverse effects on the organizational software landscape.

> **Cases [A1] and [A2]:** An insurance company deals in intangible goods. And what we ultimately produce is software – software applications. It doesn't matter whether it's for internal operations or a sales tool used by field agents when visiting customers. In the end, what we deliver is always a software solution. That's why IT and the systems behind it play such a central role. Think about it: when we launch a new insurance product, say, a new disability policy, it's not a physical item. It's a promise, a commitment to take on risk. And that promise is both managed and sold through software. (I.A.1)

> **Case [C]:** We have what we call major releases. These are changes that require integration testing. We schedule two large rollouts per year for these changes because they impact the entire bank and have to go through thorough testing. (I.C.28)

> **Case [D]:** The key thing that has to be managed is the risk involved in each release. You can't just roll out changes whenever you feel like it. Some things can only be done during the major releases. But smaller updates, like adjusting a piece of text, a letter template, or a parameter, can be done in between. What matters most, especially for a tech-driven insurance company, is that the systems run reliably, stay stable, and have as few errors as possible. That's why we need strong test coverage with lots of automated tests, so the risk of accidentally introducing serious bugs stays low. (I.D.10)

The strong emphasis on stability was further reinforced by the strict regulatory requirements for IT systems in banks and insurance companies. These regulations mandated that all software changes undergo rigorous testing, with the results thoroughly documented.

> **Case [C]:** [C] has an agile approach. But the share of agile projects in our bank is quite still small. The main reason is the regulatory environment in banking, which, I'd say, prevents many projects that might normally be agile from being truly agile. You can't just do ad-hoc releases that easily. Plus, the documentation requirements in banking are far more extensive than in a start-up. Our regulations are on par with those in the pharmaceutical industry. You have to follow strict rules that an internet start-up, like a delivery service, simply doesn't face. These requirements cover everything from planning all the way through to go-live. Testing, approvals … it's all very demanding. (I.C.27)

> **Case [C]:** This very controlled, carefully managed release process comes from the requirements set out in BAIT [Supervisory requirements for IT in financial institutions]. I'm familiar with them from various other regulations BaFin has issued related to IT supervision in banking. The same rules apply to insurance companies as well. It's basically the same framework. (I.C.2)

Across all cases, testing and documentation activities extended over several months. In cases [A1] and [A2], participatory observations revealed that the agile teams were expected to complete up to five regression testing cycles to meet organizational standards for software stability and functionality. Each cycle could last up to three weeks, resulting in an overall testing phase spanning several months. Similarly, in cases [C] and [D], release management units allocated at least two months to complete and document the required testing procedures.

> **Case [C]:** The testing phase takes at least two more months. IT testing begins around mid-February, followed by business testing throughout March and April. By May, we're mostly wrapping up final fixes on the most critical issues. (I.C.19)

> **Case [D]:** You can assume that around 30 areas are involved in a major release. That's a confirmed figure. These 30 areas represent specific applications or projects. For example, [D] counts as one

> area here, even though it actually includes two applications: the risk manager and the administrative software itself. Because of this, everything is quite concentrated and extensive, involving many components. As a result, the integration testing phase takes quite a while, around six to seven weeks. After that, there's another week for consolidation or system testing in a different environment. So, all told, we block out seven to eight weeks, at least two months, before we even begin […] getting everything ready for production. For an agile project, eight weeks includes multiple sprints and is definitely a long time. (I.D.5)

The time- and labor-intensive nature of these testing activities resulted in slow release rhythms, typically limited to two or three major releases per year (see Table 14). The limited number of major releases allowed organizations to meet high quality and stability standards without overburdening staff in the release management units. Consequently, slow release rhythms with only a few major releases became established practice across all organizations, applying equally to both agile and non-agile units. This was particularly evident in cases [C] and [D], where agile project teams, being among the first of their kind in their respective organizations, were constrained by the established major release processes. Across these cases, no faster release rhythms had yet emerged to accommodate the needs of agile units.

> **Case [C]:** First, the [biannual release rhythms] are well-established. Saying 'well-established' isn't quite accurate; it's more of a fixed, ingrained process involving the entire bank. You can't just change it overnight. (I.C.13)

> **Case [D]:** It actually started with us working in three-week sprints. The assumption among the team was that the releases and release dates would revolve around us. But that's not true! We have a lot of legacy systems. We're just a small cog in the machine, and we have to align with the overall company schedule. (I.D.7)

In cases [C] and [D], the agile units could not release software with interdependencies with the organizational software landscape outside the strictly limited number of major release dates. Thus, the limited number of software releases effectively entrained the agile project teams to slower release rhythms.

> **Case [C]:** From my perspective, the project has somewhat moved away from agility and become tied down by the releases. Because originally, the plan was to go live with something every two weeks… or maybe every four weeks, I don't know. But that's not what we're doing. (I.C.13)

> **Case [D]:** Agile projects are always eager to move fast, so to speak. But they have to - or at least until now had to -submit to the usual, traditional release process. (I.D.5)

In cases [A1] and [A2], the situation differed. In principle, the agile units were able to release software outside the biannual major software releases. However, each additional release entailed significant costs for the agile teams, as every release triggered testing and documentation requirements that they had to conduct and fund themselves. By assigning costs to each release, the release management in these cases created an effective incentive

for agile units to entrain to slower release rhythms. In light of the high costs, the agile teams voluntarily refrained from releasing software at a fast rhythm. In case [A2], the agile team fully entrained to the organization's slow release cycles. In case [A1], the agile teams conducted two additional releases per year to accelerate the delivery of new software components. However, to keep the costs manageable, they limited the number of additional releases to two.

> **Case [A1]:** We try not to deploy every month, but more like once every three months. The main reason is that we need more time to stabilize things. (…) Sometimes we even have to 'borrow' the UAT environment for that, which management really doesn't like. And of course, it adds to the cost. (I.A1.1)

> **Case [A2]:** It's not really the content that restricts me. What really limits me is knowing that certain things need to go live in sales by January 1st, or that it's a legal requirement. So, I have to be done by mid-November to make sure it can launch in January or to meet the legal deadlines. And I don't have another window in between, unless I'm willing to pay a lot for a production update. (I.A2.1)

Even in the restructuring project of case [E], the stability expectations of the predominantly non-agile organizational context determined the rhythm at which the agile project team could deliver outcomes to end users. Unlike in the previous cases, a release in case [E] did not involve deploying software that, following extensive testing, was put into productive use. Instead, a release referred to the handover or presentation of new business processes and team structures that employees in the corporate client advisory unit were expected to adopt in the future. A small portion of these concepts, specifically those aimed at optimizing minor advisory processes, was to be shared immediately upon completion. These did not require works council approval, as they did not affect employee rights. However, the majority of the concepts developed by the agile project team, including, for example, new role profiles or performance targets, were subject to co-determination. Such concepts could neither be presented to the corporate client advisory unit nor formally handed over until lengthy approval procedures had been completed. Initially, consistent with agile principles of presenting preliminary results as early as possible, the agile project team was unaware of this restriction and had already begun organizing an information session to present the new team structures and business processes to the corporate client advisory unit. Upon advice from the works council, the team postponed the session by two months, rescheduling for a time when the approval process had been completed and the outcomes were considered final and thus "safe" to communicate. In doing so, the agile project team effectively entrained to the slower delivery rhythms.

> **Case [E]:** During a daily standup meeting, an employee shares an update from a discussion with the product owner and a representative of the works council. It turns out that at the planned information session for employees of the corporate client advisory unit, scheduled for the end of the project's conceptual phase, most of the work results cannot be shared because the works council will not have reviewed and approved them by then: "We can't communicate anything about the organizational structure before the works council has approved it." Since only a small portion of the results is not subject to co-determination and could be presented, the employee questions the point of holding the information session so early and suggests postponing it to a much later date. Another project team member expresses disappointment: "I think it's a shame that we always talk about values like transparency and openness. But the things colleagues really want to know can't be shared." (Vignette from observation O.E.61)
>
> **Case [E]:** The information session ultimately takes place at the end of November, two and a half months later than the date the agile project team had originally planned. (Vignette from observation O.E.89)

Prohibiting the early communication of concepts that were subject to co-determination had become established practice within the organization as a means of preventing instability. Specifically, there was concern that if employees were informed prematurely about concepts such as new role profiles or performance targets, they might challenge them by mobilizing the works council, potentially leading to withholding of approval. By delaying the communication of project outcomes, the organization sought to avoid giving employees leverage to compel change to the developed concepts. Notably, many members of the agile project team themselves viewed concept changes negatively. This helps explain why entraining to slower delivery rhythms by postponing the information session, which represented the actual release in case [E], was willingly accepted.

> **Case [E]:** Since I know the team in the corporate client advisory unit, I support the current strategy of only releasing the concepts after the works council has approved them. Given the way this group is, I don't think it could have been handled any other way. They would have just gone straight to the works council anyway. Naturally, that would have sparked discussions, and that might have ended up being counterproductive for the project's outcomes. (I.E.17)

With the exception of the agile team in case [A1], which organized two additional release dates and thus did not fully conform to the slower, non-agile release rhythms, all agile units experienced a reduction in the speed at which they could deliver work outcomes as a consequence of entraining to slower release rhythms. This created a significant source of tension for the agile units. Although agile practices were intended to deliver outcomes quickly, the slow release rhythms prevented this advantage from being realized.

> **Case [A2]:** If you start thinking about it now, the go-live date is probably January 1, 2022. And that lead time is really too long. (I.A2.2)
>
> **Case [C]:** But since we're mostly tied to the major releases, we can only manage to develop and implement new things at a limited pace. Right? (I.C.18)
>
> **Case [D]:** We work in three-week sprints, but we only have four releases per year. So, this whole incremental or iterative approach kind of falls apart, because I can't actually release anything until three months later. In other words, the constraints we have wipe out the benefits of the project

> methodology. Theoretically, I could say, 'Hey user, you gave feedback in the review. Great! I'll build it, test it, and deliver it in three weeks. Awesome, right?' But I can't. Because I have to wait for the next major release. (I.D.2)
>
> **Case [E]:** The slowdown happens because of staffing procedures. That just takes a long time. You can easily count on about two months before we even get to start implementing anything. (I.E.4)

Chapter 5.2.1 demonstrated that agile and non-agile units exhibited differing levels of time urgency and time patience, which translated into distinct release rhythms preferred by each unit. The time patience observed in non-agile units resulted in a greater emphasis on achieving high initial levels of quality rather than on rapid delivery. To meet these quality standards and satisfy underlying stability requirements, the banks and insurance companies studied mandated time-consuming testing and approval procedures before work outcomes could be released. These procedures permitted only a limited number of releases per year, leading to a slow release rhythm. Accelerated processes tailored to the needs of agile units were either entirely absent (cases [C], [D], and [E]) or so costly that they were only used to a limited extent, if at all (case [A1] and [A2]). Consequently, the agile units either fully (cases [A2], [C], [D], and [E]) or partially (case [A1]) entrained to the slower release rhythms and experienced severe constraints on their delivery speed.

5.3.2. Developing new temporal patterns

The second pattern builds upon the first. Figure 15 largely reflects the structure of the first pattern. The fast release rhythms of agile units gave rise to ***quality concerns*** within non-agile units, particularly release management units. To accommodate the need for high quality and stability of the organizational software landscapes, the release management units limited the number of software releases, thereby forcing the agile units to ***entrain*** to slower release rhythms. This enforced entrainment entailed disadvantages. It ***limited the speed*** at which agile units were able to deliver their outcomes to end users, a core agile capability. Two of the five agile units in the first pattern were unwilling to endure these limitations of their speed. Consequently, they sought to collaborate with release management units to ***develop new temporal patterns*** that reflected their need for faster delivery. The second pattern focuses on the collaborative processes of developing new temporal patterns.

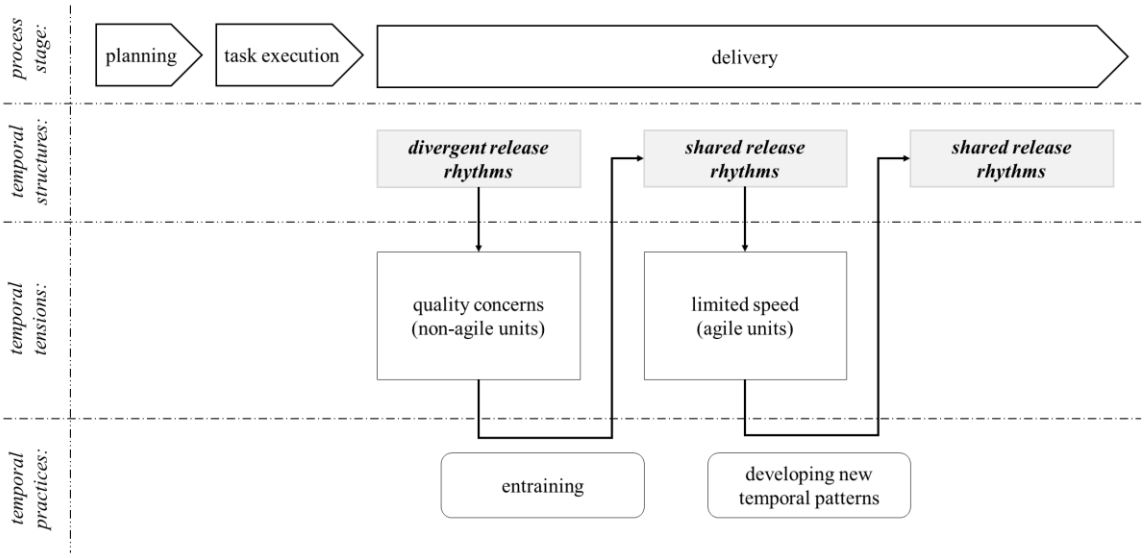

Figure 15: Developing new temporal patterns

To address the limitations on delivery speed, the agile project teams in cases [C] and [D] initiated efforts to develop new temporal patterns, namely, accelerated release rhythms. In both cases, the agile project teams entered into negotiations with the release management units to increase the number of major releases each year. The agile project teams recognized that the release management units could neither lower their stability requirements nor reduce the intensity of their testing procedures. As a result, they sought to develop new release schedules that accommodated their need for faster delivery without compromising the stability standards set by release management. In both cases, the release management units were open to enabling faster release rhythms for the agile project teams. They recognized the broader organizational shift toward agility and acknowledged that the established release processes were incompatible with agile practices. In order to better support agile projects and to future-proof their own processes, the release management units willingly engaged in negotiations to develop new temporal patterns.

> **Case [C]:** Was it really a problem? Maybe not. The release manager, [name], was actually pretty supportive. He wanted the process to be more agile. It's kind of funny because he said, 'We can't keep working like it's the 90s all the time. It's not appealing anymore, it's no fun, and it's just outdated.' In the end, we got good support from him. He could have made a lot of protests or caused trouble back then, but he didn't. (I.C.2)

> **Case [D]:** The more agile a company becomes, the more pressure – or maybe just the desire – to come up with solutions grows. (…) So, [name], two colleagues, and I decided to figure out how we could offer faster release rhythms. (I.D.5)

A central argument made by the release management units for limiting the number of major releases was that the capacities of testing environments, used to validate software

functionality and ensure stability under controlled conditions, were fully utilized by the tests required for major releases. Because the agile project teams relied on the shared testing infrastructure provided by the release management units, they were unable to carry out the necessary tests independently. As such, they could not accelerate the release rhythms on their own.

> **Case [D]:** We'd actually planned to start doing faster releases back in March last year, but we quickly realized it wasn't going to happen because the testing environment just wasn't up to it. (I.D.6)

To resolve delivery speed constraints without generating new tensions around quality or stability within the release management units, the agile project teams focused primarily on expanding the capacity available for testing activities. In case [C], the agile team noticed that the release management unit occasionally carried out additional releases between the two scheduled annual major releases. They inferred that the testing environment was not continuously occupied by major release testing activities. Instead, there appeared to be two roughly two-month time windows each year during which the environment was available for supplementary testing. The agile project team coordinated with the release management unit to identify these time slots and ultimately secured the privilege to conduct additional releases during these open periods. This privilege could only be superseded by a small number of higher-priority projects.

> **Case [C]:** I read through it and thought, 'Ah, we could do this, and that too.' Then I talked to a few people to see what was actually possible. I also checked in with [the release management colleague] and asked, 'Hey, when can we use the ITE? It's not booked all the time. When there's a major release, sure, it's blocked, and that's fine. But in between … how many open slots are there?' So, we started applying for releases in all the gaps – for my project, or rather, our project. That's how it went. Now the others have to figure out when they can go. And yeah … some topics have higher priority, and if they request a slot, they'll get precedence. (…) But for now, we've got approval for two to three upcoming [special releases]. In those cases, others will have to let us go first. (I.C.2)

> **Case [C]:** Another specific feature at [the bank] is that we have what's called an integrated test environment, the ITE. There's only one of them in the entire bank, and it's available year-round. But – though I'd have to do the math – around eight or nine months of the year, it's reserved for major releases. Once it's booked for those, there's no flexibility. The remaining three to four months are basically open, with no major releases scheduled. And that's when we typically have a window to offer something, specifically, to schedule special releases that need integrated testing on the ITE. That's basically the standard process we have to go through to get a special release approved. (…) The [project manager] reached out really early and pretty much signed up for every slot that was available. (…) The demand for special releases, and the fact that some projects consistently need them, just isn't something you can ignore. It's been that way for years. (I.C.22)

In case [D], the agile project team also identified a way to increase the number of major releases despite similar intensive testing requirements. Since the insurance company in case [D] scheduled one more major release per year than the bank in case [C], there were

fewer available time slots when the existing testing environment was not in use. However, the agile team identified an unused environment that had previously been used for an old software application. This additional environment made it possible to conduct tests independently of the company's main testing infrastructure and became the foundation for negotiating faster release rhythms in case [D].

> **Case [D]:** This is something I've been regularly discussing with release management, IT management, and others for almost two years now. The issue is that they have a dedicated testing environment, with set procedures and a tightly planned annual release calendar. It's basically booked solid. So, what we need is a separate testing environment. I've now identified one – we've found it – but now we have to set up all the systems so we can take a detour through that environment to get into production, without disrupting everyone else who's following the standard release schedule. And it's a huge undertaking. In principle, everyone's on board. Other projects have said, 'Oh! That's interesting,' because of course I looked for allies. 'That's great, we want that as well.' And the company wants it too. So, in theory, everything's great. But in practice, it's incredibly complex to set up an environment that actually works as a proper test setup. Because even if we're taking this workaround, we still have to make sure the software we put into production works reliably for our users. (I.D.2)

> **Case [D]:** So, we had a testing environment that was used for [an old application]. Since it wasn't really being used anymore, the plan was to shut it down to save money. But then we said, 'Wait! Hold on! As long as the environment is still available, we can just repurpose it!" (I.D.6)

In both cases [C] and [D], the agile project teams initiated efforts to expand testing capacity. This enabled them to conduct testing activities more frequently and to increase the number of major releases without compromising the stability requirements set by the release management units. In case [C], increases in testing capacity resulted in the introduction of up to four additional releases per year alongside the two scheduled major releases, allowing for the deployment of new software functionalities. In case [D], the expanded capacity made it possible to conduct additional releases on a monthly basis.

> **Case [C]:** We have a standard in-house release cycle, basically an IT release cycle. Formally, there are two releases a year when things go live in production. But then, there are also special and exceptional releases that bring the total to about six releases per year on average. Only two of these are officially guaranteed, though. For the rest, you really need strong negotiation skills. (I.C.1)

> **Case [D]:** We've now found a way to roll out what we call our fast-track releases. That doesn't mean daily releases, and I don't think we need to go that fast. But we're now able to release about every four weeks, which I believe is really important, especially for the [insurance line units]. (I.D.9)

Limited testing capacity was a key, though not the sole, reason why release management units generally resisted faster release rhythms. Each additional release required not only increased testing efforts from the agile project teams but also added workload for the release management units. In case [C], for example, the release management unit needed to prepare data from the bank's end-of-day processing to support the mandatory integration tests for each release. This data enabled the reconstruction of the software

landscape within the testing environment, facilitating validation of software functionality and preventing adverse effects on the company's overall software landscape. In case [D], the setup and maintenance of an additional testing environment imposed further workload on the release management unit.

> **Case [C]:** There's always a lot of work involved in a special release, especially if we're using it fully, like when we actually need the ITE. Another factor is the end-of-day-processing, which we call batches, that have to run. Lots of people are involved and have to take specific actions. When they're busy with that, they can't work on other tasks. It simply costs money, so it needs to be approved. (…) When the department head (…) came on board, his attitude was basically, 'Make everything possible.' Everyone wanted to release whenever they wanted. I really pushed back hard and said, 'Look, my team can't handle this. It's just too much! We need to change something!' (I.C.22)

> **Case [D]:** Especially when it comes to the IT folks managing the surrounding systems … Take a central system like 'partner', where all the customer data is stored, or 'text', which handles all correspondence, printing insurance contracts, and everything. They'd have to support another environment on top of the many they already manage. So just because of my requests, they'd have to keep a fifth environment up and running. Naturally, that's extra work for them, and they're definitely not thrilled about it. (I.D.2)

> **Case [D]:** But we'll probably need to invest some more money now to get [the testing environment] properly set up again. I think it's been sitting idle for about a year. So, all the new applications or updates from the past year will have to be installed on it in the correct versions. And that's likely going to tie up some staff resources. (I.D.6)

To keep the additional workload for the release management units manageable, the accelerated release rhythms did not replace the standard rhythms established within the organizations. Instead, they were implemented exclusively in collaboration with the two agile project teams in cases [C] and [D]. While the new release rhythms constituted new temporal patterns, they were not adopted as a broader organizational standard. In case [C], even the agile project team could not consistently rely on accelerated release rhythms. Major releases outside the standard schedule required prior approval. Accordingly, the agile project team had to formally request permission from the release management unit to carry out any additional major releases. This approval process not only involved considerable administrative effort but also required the agile project team to provide clear, well-founded justifications for each additional major release. A general reference to the need for faster release rhythms in agile practices was not considered sufficient. However, since the project had a regulatory focus, such justifications were relatively easy to provide.

> **Case [C]:** But of course, [additional major releases] bring some overhead because you have to justify to project management why an interim release is needed. (…) There are lots of internal processes at [the bank] that need to be followed, and naturally, that takes time. (I.C.12)

> **Case [C]:** Sure, everyone at the bank knows we're an agile project now. That argument is gaining more acceptance. But especially early on, it didn't really carry much weight. Luckily, we're working in a regulatory context, and the bank has to implement new regulatory requirements. Management naturally has a strong interest, and is required, to provide timely information to the financial supervisory authority. So that gives us a pretty solid argument to push for what we need. (…) There are changes that mean our interface has to be live by the end of the year for certain regulatory reports. That's a really strong case. We're fortunate to always have the regulatory angle to fall back on. So, if nothing else works, that usually does the trick [laughs]. (I.C.12)

In case [C], additional major releases faced a limitation beyond the need to demonstrate sufficient importance and urgency. Only a restricted set of software functionalities could be deployed. The testing environment was available for roughly one month, significantly shorter than the standard three-month testing phase for regular releases. As a result, the scope of testing had to be reduced. To maintain system stability under these conditions, the release management unit ensured that additional major releases included only a small number of changes. This also meant that only one project or unit was permitted to deploy new software in a given release window. These precautions were intended to minimize interdependencies and helped maintain the stability of the organization's software landscape, despite the compressed timeframe for testing.

> **Case [C]:** And then we have several smaller special releases. They usually happen within about a month and tend to have a very narrow focus, mostly regulatory requirements or mandatory issues – very targeted. (I.C.27)

> **Case [C]:** I believe they've already had, or planned, two or three additional releases this year. We weren't involved in any of them. (…) Honestly, that probably wouldn't have worked. If we had been involved, I'm not sure the bank would have approved it as an additional release just for this project. They might have said, 'No, this is already too big to be treated as an additional release for a single project.' (I.C.19)

In case [C], the release management unit retained the authority to reject requests for additional major releases, for example, if there was no compelling justification, if time constraints prevented adequate preparation for testing, or if the release was deemed too large in scope. In such cases, the agile project team was not able to enact the accelerated release rhythms. The possibility of rejection meant that the new release rhythms were anchored in event time rather than clock time. Since there was no guarantee that an additional major release would be approved, the release rhythms were irregular and dependent on contingencies such as the urgency and importance of respective software functionalities, the scope of the changes involved, and the workload of the release management team.

> **Case [C]:** There are some things where we say, 'Okay. It's not absolutely necessary right now.' That was especially true at the start of the project when we were running things in parallel. We thought it would be better for the project if we could release sooner. But if release management

> says, 'We can't do it because of dependencies, or because they don't want to make the releases too big, or maybe they can't provide the batches or technical processing we need, or the infrastructure isn't in place, or it would just be too expensive to do it only for us,' then we have to wait for the next major release. That said, all of the releases I've been involved with – meaning additional releases – I think we've been part of most of them. There have been a few exceptions where we had to accept that it wasn't possible at the time and wait for the next major or minor release." (I.C.12)

In case [D] as well, although the reactivated testing environment made additional monthly major releases technically possible, the release management unit deliberately chose not to implement such a regular and frequent release rhythm. Instead, like in case [C], the release rhythm was anchored in event time. Additional major releases were offered on demand and only during longer intervals between regularly scheduled major releases.

> **Case [D]:** I'll have to check, but the target date we're aiming for is October 7th. (…) If all goes well, there won't be a regular schedule. Releases will happen on demand, that's the plan. (…) The idea is, if someone has something ready and we can do it … Say there's no major release coming up and I want to fill the gaps between major releases with software that's ready and adds value … Then we want to offer it on demand. (I.D.5)

The main reason for this decision was the release management unit's concern that running additional major releases alongside regularly scheduled ones could lead to version inconsistencies. This refers to situations in which different components of a system operate on different versions of the same software, which can result in difficult-to-trace errors, system incompatibilities, or even security vulnerabilities. To prevent such issues, parallel testing and release activities were avoided. Additional major releases were not permitted if their timing was too close to a regularly scheduled major release. In contrast to case [C], where both quality concerns and workload pressures shaped the event-driven release rhythms, case [D] was guided solely by concerns about software quality and system stability.

> **Case [D]:** Running a fast-track release alongside a major release is probably going to be tight because that would create version inconsistencies. (…) So right now, the chance of having a fast-track release running in parallel with a major release is pretty low. (…) It's not really about capacity, since capacity is often on the [agile project's] side anyway, but about avoiding conflicts. We absolutely can't afford version inconsistencies! We have to keep the software versioning consistent and clean; otherwise, if the version that's supposed to come next gets deployed before the major release, it becomes tricky to fix things afterward. Not impossible, but tricky. And as we all know, software is prone to bugs. You can never fully test everything beforehand. Errors pop up in production every day. To fix those with hotfixes or patches outside the regular release cycle, the versioning has to be spot on, or else we just keep going in circles. (I.D.5)

In case [D], the agile project team and the non-agile release management unit had already completed negotiations about accelerating release rhythms. However, the first additional major release that the agile project team could actually use to speed up software delivery was not scheduled until several months after the end of data collection. In case [C], the

accelerated release rhythms, though triggered by specific events rather than clock-time-based intervals, proved effective. They allowed the agile project team to deploy software more frequently, alleviating the tension caused by the limited delivery speed. This was achieved without compromising quality or, more importantly, the stability of the broader software landscape, and without placing an excessive burden on the release management unit.

> **Case [C]:** We've definitely been releasing more often, but only in parts, mostly in the less critical areas. For a lot of things, we still have to wait for the major releases. But I do get the sense that we're pushing into production more frequently now, which is something the business values and that's also technically doable. So, it hasn't created extra overhead, just the kind you can handle. And the desire, or the ability, to get deliverables into production earlier has really been met. (I.C.28)

5.3.3. Interim conclusion

This chapter has highlighted a third difference in the temporal structures of agile and non-agile units. Diverging priorities, whether emphasizing speed of implementation or the pursuit of perfected outcomes, translate into distinct release rhythms. Agile units tend to favor faster release rhythms with frequent release opportunities that allow them to deliver results to users quickly. In contrast, non-agile units prefer slower release cycles with fewer release opportunities, which enable more extensive testing and help meet their high standards for quality and system stability.

Across the five cases in Pattern 1, the tensions between fast and slow release rhythms were initially resolved in favor of the non-agile units. Agile units were unable to release their deliverables independently and instead relied on infrastructure or decisions controlled by potentially more powerful actors, such as senior management, release management units, or works councils. As a result, the agile units entrained to the slower release rhythms established by non-agile units. While this alleviated non-agile units' concerns regarding quality and system stability, it introduced new tensions for the agile units, who now faced constraints in how quickly they could deliver output, restricting a core capability of agile practices.

The agile units responded to this tension in different ways. In cases [A1], [A2], and [E], they simply endured the limited delivery speed. In [A1] and [A2], the agile teams could theoretically have carried out additional major releases but chose not to, due to the high

costs this would have entailed. In case [E], I observed that members of the agile project team began to prioritize the quality and stability concerns of the non-agile units after becoming aware of them, rather than resolving their own speed-related tensions. As a result, they accepted entrainment to the slower release rhythms without resistance. In contrast, in cases [C] and [D], the agile units were unwilling to accept the limitations of their delivery speed. After an initial episode of entraining, they entered into negotiations with the release management units and developed new temporal patterns. The faster, event-time-based release cycles that emerged from these negotiations alleviated the speed tensions experienced by the agile units, without resurrecting quality concerns or significantly increasing the workload for non-agile units.

6. Discussion

This dissertation aimed to generate new insights into how actors with divergent temporal structures coordinate their interdependent activities. This chapter discusses the findings and outlines their theoretical implications. The research objective of this dissertation was to contribute to temporality research. Chapter 6.1 addresses the theoretical implications for this body of research. However, the findings also have relevance for research that examines the specific empirical context of this dissertation: agile practices. Over the past decades, agile practices have developed into an independent research stream, especially within management and information systems research. Chapter 6.2 discusses the implications of the findings for this body of literature. The discussion begins with a brief summary of the main findings.

Chapter 5.1 focused on the different planning cycles of agile and non-agile units. While agile units employ short planning cycles that align with their two- or three-week sprints, the planning cycles of non-agile units span considerably longer periods. The findings reveal four patterns in how agile units navigate the tensions arising from different planning cycles: uncertainties in resource allocation, limitations of flexibility, and waiting times. In six cases (patterns 1 and 2), agile units navigated resource allocation and flexibility tensions by alternating between entraining, developing new temporal patterns (only in pattern 2), and detraining. The third and fourth patterns do not display such dynamics. In pattern 3, the agile unit entrained to non-agile planning cycles once and attempted, unsuccessfully, to resolve resulting flexibility tensions by eliminating schedule disruptions. In pattern 4, the agile unit forwent entraining altogether, thereby avoiding flexibility tensions and experiencing recurring resource allocation tensions. Regardless of the dynamics of temporal structuring, waiting times occurred across all four patterns. These were jointly resolved by agile and non-agile units through the (often combined) use of buffering and harmonizing practices.

Chapter 5.2 focused on the tensions that arise for agile units from the deadlines imposed by non-agile units. These deadlines represent temporal patterns that, due to their finite nature, are strongly anchored in clock time and conflict with the cyclical and open-ended temporal patterns characteristic of agile practices. Agile units responded to this conflict in two distinct ways. In the first pattern, agile units entrained to non-agile deadlines,

attempting to offset the associated reductions in flexibility by applying flexibilizing practices. While this approach succeeded in some cases, in others it resulted in the systematic deprioritization of end-user requirements, thereby lowering the responsiveness of agile units. In the second pattern, agile units detrained from the deadlines imposed by non-agile units. However, disregarding the deadlines generated uncertainty regarding the timely delivery of work outcomes. This uncertainty prompted one agile unit to revert to entraining to the deadline, while a second agile unit mitigated it through harmonizing practices.

The final empirical chapter, Chapter 5.3, examines responses to different release rhythms. While the fast release rhythms of agile units led to quality concerns in non-agile units, the slower release rhythms of non-agile units limited the speed of agile units. Pattern 1 revealed that all agile units initially had to entrain to the slower release rhythms that were dictated by release management units. However, in the second pattern, two agile units leveraged the tensions resulting from their entraining to collaboratively negotiate new temporal patterns with release management units. Chapter 5.3 provides a detailed account of these negotiation processes.

6.1. Implications for temporality research

In Chapter 3, I identified three overarching research gaps that have received insufficient attention in prior literature on temporality. First, research on synchronizing practices, commonly employed to address temporal coordination challenges, has predominantly focused on their benefits while largely overlooking the associated costs and potential drawbacks (Blagoev & Schreyögg, 2019). In addition, prior work has concentrated almost exclusively on the actions of time-setting actors while leaving the agentic role of non-time-setting actors in synchronizing processes largely unexplored. Chapter 6.1.1 addresses this gap by outlining this dissertation's contributions to understanding both the negative consequences of synchronizing practices and the agentic role of non-time-setting actors. I discuss the negative consequences of entraining revealed by the findings and show how non-time-setting actors integrate them into their decision-making to determine whether to entrain or resist entrainment. In doing so, I illuminate the role of non-time-setting actors' decision-making in entrainment processes. Moreover, the findings extend

the narrow perspective on the development of new temporal patterns by comparing episodes in which they are unilaterally imposed with episodes in which they are multilaterally negotiated. In this way, this dissertation further contributes to our understanding of why actors accept or resist synchronization.

Second, although the idea that actors do not necessarily require shared (i.e., synchronized) temporal structures to coordinate effectively has gained traction in temporality research (Blagoev & Schreyögg, 2025; Hilbolling et al., 2022), the literature to date presents only a limited range of sustaining practices that enable coordination while preserving the unique temporal structures of different actors. Existing studies have primarily focused on single-case contexts in which interdependencies were negligible or actors operated in temporally open-ended processes without time pressure, limiting their applicability to contexts characterized by strong interdependencies and significant time constraints. In Chapter 6.1.2, I discuss how the findings of this dissertation extend previous knowledge on sustaining practices. I highlight buffering and harmonizing practices as novel sustaining practices that emerge in contexts characterized by strong interdependence and time pressure. Additionally, I discuss how the findings of this dissertation refine our understanding of previously identified brokering practices.

Finally, Chapter 3 highlighted that existing research has paid little attention to the dynamics of temporal structuring and temporal structures. To date, research has focused on single episodes of temporal structuring (Otto et al., 2024; Shipp & Richardson, 2021), leaving unexplored whether and how these episodes coalesce into broader temporal structuring processes that dynamically reshape actors' temporal structures. In Chapter 6.1.3, I discuss this dissertation's contributions regarding the dynamics of temporal structuring. The findings reveal that dynamics are triggered by the emergence and re-emergence of temporal tensions. The chapter also makes an attempt to theorize the changes in temporal structures, drawing on legitimacy conflicts within agile units (Suchman, 1995).

6.1.1. Implications for synchronizing practices

Prior research has largely treated synchronization practices as beneficial. Numerous studies have highlighted the positive effects of entrainment on coordination and organizational performance (Dibrell et al., 2015; Hopp & Greene, 2018; Khavul et al., 2010; Pérez-Nordtvedt et al., 2008; Shi & Prescott, 2012) but have overlooked that entrainment may also entail negative effects. Recent review articles summarizing the state of research on entrainment acknowledge this underrepresentation of potential drawbacks. For instance, Sandra and colleagues (2023) cite only a single study (Luciano et al., 2018) that addresses the negative consequences of entrainment. The absence of a critical perspective that considers not only the benefits but also the drawbacks and costs of entrainment has been explicitly criticized. Blagoev and Schreyögg (2019, p. 1843), for example, question the assumption, implied by the largely positive findings in prior research, that entrainment represents a "purely cost-free state of harmony."

A first theoretical contribution of this dissertation regarding synchronizing practices lies in its consideration of the negative consequences of entrainment. The findings presented in Chapter 5 illustrate several episodes in which agile units entrain to the temporal patterns of their non-agile collaboration partners. Agile units entrain to longer planning cycles, typically dictated by management, in order to secure resources from non-agile units during the planning stage (see Chapter 5.1). Similarly, adhering to deadlines through which non-agile units seek to monitor and control the timely completion of (interim) outputs by agile units, also constitutes entrainment (see Chapter 5.2). Finally, many agile units entrain to the slower release rhythms established by non-agile release management units in order to make their work outcomes available to end users (see Chapter 5.3).

Each of these entrainment episodes resolves a tension for either agile or non-agile units (e.g., uncertainty of resource allocation for agile units during the planning stage, or quality concerns for non-agile units during the delivery stage), but they also generate new tensions for agile units. Chapter 5.1 shows that, after entraining to longer planning cycles, agile units have limited flexibility to respond to unforeseen changes, such as evolving user requirements. Chapter 5.2 demonstrates that overriding the event-time conception of agile work processes constrains incremental development, thereby reducing the flexibility and responsiveness of agile units. Chapter 5.3 illustrates how the rapid delivery of

solutions is slowed down by entraining to the slower release rhythms of non-agile release management units. Overall, entraining to the temporal patterns of non-agile units undermines flexibility, responsiveness, and speed as core agile capabilities (Sharifi & Zhang, 1999), which constitute the very rationale for organizations to implement agile practices (Annosi et al., 2020; Conboy, 2009; Renzl et al., 2021).

The loss of agile units' distinctive capabilities after entraining provides a largely overlooked, internal perspective on the costs of entrainment. The few studies that discuss negative consequences of entrainment adopt a primarily external perspective, emphasizing that the cost of entrainment lies chiefly in the fact that synchronizing with a dominant Zeitgeber simultaneously entails decoupling from other, potentially equally important, Zeitgebers (Blagoev & Schreyögg, 2019; Hilbolling et al., 2022; Zhang et al., 2023). In other words, entrainment is constrained by its applicability to coordination with a *single* Zeitgeber. This drawback becomes particularly salient in contexts characterized by high temporal complexity, where multiple, potentially conflicting Zeitgebers are present at a given moment or over time (Blagoev & Schreyögg, 2025; Hilbolling et al., 2022).

The internal perspective evident in the findings of this dissertation suggests that the drawbacks of entrainment do not lie solely in its limited applicability under conditions of high temporal complexity, but more fundamentally in the abandonment of actors' unique temporal patterns, its *Eigenzeit* (Blagoev & Schreyögg, 2025; Nowotny, 1992). Relinquishing these temporal patterns can become problematic when they underpin the work practices and routines that actors rely on to perform their tasks effectively. Research on routines has shown, for example, that underlying temporal patterns strongly influence the effective enactment of routines (Geiger et al., 2021; Turner & Rindova, 2018). Firefighters, for example, actively seek to maintain their Eigenzeit during operations by creating temporal boundaries, since entraining to the temporal patterns of their environment could impair routine performance (Geiger et al., 2021). The significance of temporal patterns as a foundation for effective routine performance can also be extended to agile practices, as practices can themselves constitute routines (Feldman, 2021; Feldman et al., 2021), and prior research has frequently conceptualized agile practices as routines or routine clusters (Mahringer & Danner-Schröder, 2025; Sailer et al., 2023).

The findings demonstrate that the effectiveness of agile practices diminishes when agile units become entrained to the longer planning cycles, clock-time-based temporal patterns, and slower delivery rhythms that characterize the temporal structures of non-agile units. Although agile practices are formally maintained across all cases, by segmenting the available time into two- or three-week sprints intervals and by conducting Scrum events in accordance with the Scrum Guide (with the exception of case [A2]), the erosion of their temporal foundation leaves agile units largely unable to respond to change with greater speed and flexibility than non-agile units. The continuation of agile practices is not regarded as merely ceremonial (Meyer & Rowan, 1977), since agile units continue to derive benefits from enhanced intra-team communication and coordination fostered by regular Scrum events (Stray et al., 2016). Nonetheless, they fall short of the expected levels of flexibility, responsiveness, and speed commonly attributed to agile practices. This shortfall is particularly problematic because agile practices are employed in highly dynamic contexts where requirements and solution approaches are ambiguous or uncertain (Stacey, 2012), and where successful task execution depends precisely on these agile capabilities. Agile units, therefore, often perceive entrainment as disadvantageous, since the temporal structures imposed by non-agile units are ill-suited to support the enactment of agile practices and, consequently, to the effective accomplishment of their tasks.

This disadvantage of entrainment is rooted in an internal perspective, which, unlike the disadvantages highlighted in prior research, is not concerned with the limited applicability of entrainment in the presence of multiple Zeitgebers. Rather, it reflects that homogeneous temporal structures, which actors adopt by entraining, are not uniformly suited to all tasks (P. R. Lawrence & Lorsch, 1967). By elaborating this internal perspective on the drawbacks of entrainment, this dissertation contributes to incorporating a contingency-theoretical perspective (Burns & Stalker, 1961; Galbraith, 1973; P. R. Lawrence & Lorsch, 1967; Woodward, 1965) into the concept of entrainment.

A consequence of the misalignment between the temporal patterns of the Zeitgeber and the agile practices is the agile units' reluctance to entrain to the Zeitgeber's temporal patterns. Across all three findings chapters, evidence indicates that agile units resist entrainment to the temporal patterns of non-agile units. In Chapter 5.1, resistance to entrainment is evident as most agile units, following an initial episode of entrainment,

deliberately detrained from the longer non-agile planning cycles, reverting to their original sprint-based planning cycles. In Chapter 5.3, resistance is demonstrated by some agile units, who, after being subjected to slow release rhythms, established novel temporal patterns by accelerating these release rhythms. Previous research has identified both detraining and developing novel temporal patterns as forms of resistance to entrainment (McGivern et al., 2018; Shipp & Richardson, 2021). The most active form of resistance, however, is observed in Chapter 5.2, where two of the eight agile units deliberately circumvent entraining to clock-time-based temporal patterns by intentionally disregarding non-agile deadlines (Waller et al., 2001).

These resistances suggest that agile units assess the disadvantages associated with relinquishing their own temporal patterns when deciding whether to entrain or not. Thus, the contributions of this dissertation extend beyond merely accounting for the negative consequences of entrainment, shedding light on the complexity of the decision-making processes that precede it.

Existing research frames entrainment as an agentic decision in which actors consciously determine whether to engage in it (Leroy et al., 2015; Omidvar et al., 2025; Shipp & Jansen, 2011). The drivers or factors that actors rely on to make these conscious decisions, however, have received limited attention and warrant further investigation (Zhang et al., 2023). To date, only Shipp and Richardson (2021) have conceptualized four drivers of entrainment decisions. These drivers (outcome interdependences, the mutuality of power relations, information uncertainty regarding the Zeitgeber's behavioral choices, and the likelihood of future interactions) attribute entrainment decisions exclusively to the relationship between actors and their potential pacers. Thus, they reflect an external perspective that neglects the possibility that actors may judge the pacers' temporal patterns as more or less appropriate for supporting their work practices or accomplishing their tasks. The entraining-detraining dynamics observed in Chapter 5.1 can also be explained by these relational drivers. Agile units detrained at the onset of the task execution stage, following an initial episode of entrainment, because their dependence on management decreased once resources for executing their activities had been allocated during the planning stage. While Chapter 5.1 thus provides, to the best of my knowledge, the first empirical evidence for the relational decision-making drivers conceptualized by

Shipp and Richardson (2021), the resistances observed in Chapters 5.2 and 5.3 cannot be accounted for by these factors.

In Chapter 5.2, the agile units in cases [C] and [D] consciously decide not to comply with the deadlines imposed by release management units as non-agile Zeitgebers. In Chapter 5.3, the same agile units negotiate new temporal patterns following an initial episode of entrainment to slower release rhythms, in order to deploy their software solutions more quickly. From the perspective of Shipp and Richardson (2021), these agile units would be expected to fully entrain to non-agile temporal patterns in both situations. First, the agile units in both cases are highly dependent on the infrastructure provided by the release management units to deploy software solutions. Second, the agile units are situated in an unfavorable power position. In cases [C] and [D], management permits the use of agile practices within the studied projects, yet abstains from positioning agile transformations as a strategic priority for the organizations. Thus, the agile units proceed largely without managerial support when they endeavor to alter the temporal patterns underlying the release processes. The authority to approve or reject faster release rhythms or deviations from established deadlines resides exclusively with the release management units, granting them greater power within the collaborative relationship. Third, agile units face low information uncertainty because of the formalized processes and clearly defined deadlines, and fourth, future interactions with the release management are foreseeable. Although high dependence, limited power, low information uncertainty, and a high likelihood of future interactions would predict full entrainment, I observe how the agile units actively resist by renegotiating deadlines (Chapter 5.2) and developing new temporal patterns (Chapter 5.3).

The agile units' decisions to diverge from non-agile deadlines or to develop novel temporal patterns that accelerate their release rhythms are not driven by the Zeitgeber's relative power or dominance. Rather, these decisions are shaped by the negative consequences of entraining to non-agile temporal patterns, particularly regarding limitations in flexibility, responsiveness, and speed. Thus, actors consider both the relational characteristics of their connections with potential Zeitgebers (Shipp & Richardson, 2021) and the extent to which the temporal patterns adopted through entrainment enable the effective execution of the work practices they have selected when deciding whether to entrain. Accordingly, this dissertation contributes to prior research

by introducing an additional factor that accounts for actors' decisions regarding entrainment, uncovering both a previously overlooked dimension and the intricate complexity inherent in the decision-making situation preceding entrainment. Thus, actors' decisions involve not only selecting the pacer with whom to synchronize (Blagoev & Schreyögg, 2025; Hilbolling et al., 2022) but also assessing whether the temporal patterns adopted through entraining effectively support their work. While the findings reveal that various, potentially conflicting drivers influence entrainment decisions, further research is needed to investigate the interactions among these drivers.

The findings frame entrainment decisions as a dilemma (Putnam et al., 2016). Contexts that demand coordination across boundaries are especially prone to such dilemmas, in which actors must contend with the trade-offs between competing yet disadvantageous alternatives (Kellogg et al., 2006). In the context of this dissertation, the two alternatives lie in achieving effective cross-boundary coordination between agile and non-agile units and in effectively applying agile work practices, both of which may critically influence the success of the agile unit. Choosing to entrain may facilitate coordination with non-agile units, yet it can simultaneously undermine the effectiveness of agile work practices if the temporal structures of the Zeitgeber fail to provide adequate support. Conversely, choosing not to entrain may safeguard the efficacy of agile practices, but at the cost of impeding coordination with non-agile actors.

A further contribution of this dissertation lies in elucidating how actors address and navigate this dilemma of entrainment decisions. Prior research has shown that, in similar dilemmas, actors develop new "ambitemporal" patterns that adequately reflect the deep-level temporal conceptions and orientations of all collaborating parties (Reinecke & Ansari, 2015). The findings of this dissertation identify, alongside the development of new temporal patterns, two additional approaches. First, actors who opt not to entrain and instead maintain their divergent temporal patterns can compensate for the coordination benefits of shared temporal patterns by employing alternative coordination practices. Second, actors who opt to entrain can preserve or restore the effectiveness of their work practices through the use of complementary practices.

In the first approach, actors compensate for the loss of coordination benefits caused by maintaining divergent temporal structures by adopting alternative temporal coordination practices. Without shared temporal patterns, conventional temporal coordination

mechanisms, such as plans or deadlines, cease to function effectively, since at least one collaborating actor no longer adheres to them (Bechky, 2006; Levina & Vaast, 2005; Okhuysen & Bechky, 2009). Chapters 5.1 and 5.2 demonstrate the breakdown of conventional temporal coordination mechanisms. In Chapter 5.1, the detraining of agile units from the planning cycles of non-agile units rendered plans ineffective. The breakdown of plans as coordination mechanisms manifested in waiting times, which emerged from the misalignment between the timing of resource demands and resource availability. In Chapter 5.2, the coordinating function of deadlines was undermined when agile units stopped following them to safeguard their agile capabilities. Although these synchronization-based coordination practices lost their effectiveness, neither the collaboration between agile and non-agile units nor the performance of the agile unit was compromised. The findings indicate that the collapse of plans and deadlines can be counterbalanced by alternative temporal coordination practices. Responding to recent calls in the literature (Blagoev et al., 2024; Zhang et al., 2023), this dissertation demonstrates how actors can align deliberate non-entrainment decisions with the temporal characteristics of their environment. The specific temporal coordination practices through which agile and non-agile units meet their coordination needs while maintaining their distinct temporal patterns are discussed in Chapter 6.1.2.

The second approach indicates that the underlying decision-making dilemma does not require actors to opt against entraining. Alongside the first approach, Chapter 5.2 shows that actors can entrain to a Zeitgeber's temporal patterns while simultaneously ensuring the effective execution of their agile practices through complementary flexibilizing practices. Rather than offsetting the coordination provided by shared temporal structures (Ancona & Chong, 1996; Orlikowski & Yates, 2002), these practices serve to compensate for the limitations on flexibility resulting from entrainment. The flexibilizing practices I observed are intended to create available capacity to accommodate unplanned tasks. They involve both reducing the workload of agile units, either through fewer tasks or lowered quality expectations, and increasing the number of employees within the agile units. Thus, their effectiveness is based on the provision of slack resources. Reducing workloads creates temporal slack (Agrawal et al., 2018), providing agile units with disposable time to address unplanned tasks, such as correcting unexpected software errors. Increasing the number of employees in agile units generates human resource slack (Bentley & Kehoe, 2020; Lecuona & Reitzig, 2014), allowing unplanned tasks to be addressed with

supplementary workforce capacity. Slack resources are widely acknowledged in the literature on organization studies and strategy for enhancing flexibility (Cyert & March, 1963; Mount et al., 2024), and are thus considered a key enabler of agile capabilities (Teece et al., 2016).

While decisions against entraining invariably create coordination tensions that call for complementary coordination practices, flexibilizing practices to support entraining decisions are contingent on the specific case of collaboration between agile and non-agile units. They are effective because entraining to the temporal structures of non-agile units restricts agile units' flexibility, prompting them to respond with suitable practices that restore flexibility. In other cases, however, different practices might be complementary, depending on the specific tensions arising from entrainment. For example, if non-agile units following plan-based practices were to entrain to the temporal structures of agile units, the ensuing tensions would probably arise from limitations in predictability or efficiency rather than restricted flexibility. Accordingly, other complementary practices would be necessary to specifically reinstate these capabilities.

Both approaches indicate that the dilemma inherent in entraining or non-entraining decisions can be overcome by counterbalancing the disadvantages they entail with complementary practices. This builds on the notion that practices, such as entraining or detraining (the effective enactment of a non-entraining decision), should not be considered in isolation (Hui, 2017) but rather as part of a bundle of practices (Schatzki, 2006), in which they interact and mutually complement each other (Smets et al., 2015). Thus, this dissertation advances knowledge on how actors navigate the dilemmas arising from entraining and non-entraining decisions, thereby supplementing the solution proposed by Reinecke and Ansari (2015) of developing new ambitemporal patterns by highlighting the complementary interplay of different practices.

Synchronizing practices not only include entraining to dominant Zeitgebers but also developing new temporal patterns to which actors can synchronize (Granqvist & Gustafsson, 2016; Omidvar et al., 2025). The findings of this dissertation uncover several episodes in which actors develop new temporal patterns. In Chapter 5.1, non-agile units in cases [B1], [B2], and [B3] develop new temporal patterns by shortening planning cycles to address the tensions resulting from entraining to the organization's annual planning cycles. In Chapter 5.3, agile units initiate the development of new temporal

patterns by negotiating accelerated release rhythms to alleviate the tensions caused by entraining to the slower release rhythms of non-agile units. A closer examination of these episodes enhances our understanding of how new temporal patterns are developed.

Developing new temporal patterns is understood as an institutional change process (Granqvist & Gustafsson, 2016; McLeod et al., 2024) whereby temporal tensions are abductively transformed into novel temporal patterns (Feldman et al., 2022; Otto et al., 2024). While these new temporal patterns constitute temporal innovations for the actors synchronizing with them (Shipp & Richardson, 2021), their development, design, and enforcement do not differ from traditional entrainment. The few studies that examine the development of new temporal patterns uniformly describe how such patterns are unilaterally created, decided, and implemented by a single actor (Feldman et al., 2022; Granqvist & Gustafsson, 2016; Jarvenpaa & Välikangas, 2022; Lifshitz-Assaf et al., 2021; Otto et al., 2024). Yet, they neglect that unilaterally imposed temporal patterns, despite being temporal innovations, may not be embraced or effectively adopted by all collaborating actors. When newly developed temporal patterns do not align with actors' broader temporal structures, especially their deep-level temporal conceptions and orientations (Ancona, Okhuysen, et al., 2001; Rowell et al., 2016), actors may resist and detrain from them, even when imposed by a relatively powerful Zeitgeber (McGivern et al., 2018; Shipp & Richardson, 2021).

The findings of this dissertation encourage understanding the development of new temporal patterns not merely as a unilateral process dictated by a single actor (Omidvar et al., 2025), but as a multilateral negotiation in which multiple collaborating actors co-create them. Whereas the shortened planning cycles presented in Chapter 5.1 were developed solely by non-agile units, the accelerated release rhythms in Chapter 5.2 emerged through a negotiated compromise between the agile unit and the non-agile release management units.

Unilaterally imposed versus multilaterally negotiated temporal patterns produce differing levels of actor acceptance and adoption. Chapter 5.1 shows that while agile units formally synchronized with the shortened two- or three-month planning cycles by participating in the relevant planning processes, they remained informally detrained, as they did not consider the outcomes of these processes binding and continued to operate according to their own two- or three-week sprint planning cycles. Consistent with prior research

(McGivern et al., 2018; Shipp & Richardson, 2021), the unilaterally developed temporal structures were not effectively adapted by agile units, despite being designed to alleviate the temporal tensions they experienced. By contrast, the accelerated release rhythms negotiated multilaterally in Chapter 5.2 were effectively implemented by both agile units and non-agile release management units. Through multilateral negotiation, actors were able to identify, understand, and integrate the tensions of all involved parties into the development of new temporal patterns. Thus, the newly developed temporal patterns in Chapter 5.3 represented a compromise, mitigating the speed tensions experienced by the time-urgent agile units while preventing quality concerns for the time-patient non-agile units. The new temporal patterns that emerged were "ambitemporal" (Reinecke & Ansari, 2015), capturing the divergent temporal orientations of different actors, and thereby being effectively adopted by all collaborating actors. While these findings do not preclude the acceptance or use of unilaterally imposed temporal patterns, they indicate that multilateral negotiations are more likely to produce temporal patterns that are effectively adopted, since they better capture the tensions of all collaborating actors.

Moreover, the findings address a question raised by Granqvist and Gustafsson (2016) in their seminal work on temporal institutional work: Does synchronizing with new temporal patterns inevitably cause desynchronization from previously established temporal patterns? The evidence presented in this dissertation contradicts this claim. The new temporal patterns developed in Chapters 5.1 and 5.2 to alleviate the flexibility and speed tensions of the agile units are temporal parallel structures. In Chapter 5.1, the annual planning cycles of management persisted and were not replaced by the shortened two- or three-month planning cycles developed by the non-agile data warehouse unit in case [B1] or by management in cases [B2] and [B3]. Instead, the shortened planning cycles constituted a complementary temporal structure, used in parallel with the annual planning cycles that continued to apply across all units. Thus, agile units were still required to forecast tasks and resource needs over the annual planning horizon, yet could review and adjust their forecasts within the shortened planning cycles as needed. Similarly, in Chapter 5.3, the accelerated release rhythms did not replace the existing temporal patterns. They were enacted only in collaborations between agile units and release management units, while non-agile units continued to release outcomes according to the previously established slower rhythms.

Observing that new temporal patterns ran in parallel with established temporal patterns, instead of replacing them, points to two implications. First, new temporal patterns can be developed and enacted exclusively for collaborations of particular actors, while established temporal patterns remain in place for other actors within the same organization. This is most clearly demonstrated in cases [B1], [B2], and [B3]. While management introduced shortened planning cycles in collaboration with the agile units in cases [B2] and [B3], it did not use them in collaboration with the agile unit in case [B1]. Likewise, in cases [C] and [D], the accelerated release rhythms existed solely as arrangements between the agile units and the non-agile release management units while non-agile units continued to follow slower release rhythms. The findings of this dissertation thus depict temporal patterns as loosely coupled systems, able to persist with relative independence and coexist alongside one another (Orton & Weick, 1990; Weick, 1976). Accordingly, new temporal patterns induced by agile units do not necessitate changes to established temporal patterns within the broader organizational contexts.

Second, understanding temporal structures as loosely coupled systems implies that actors can draw on multiple temporal structures, using them as distinct "temporal repertoires" (Jarvenpaa & Välikangas, 2022; Stonig & Müller-Stewens, 2025) in collaborations with different agile or non-agile actors. Building on prior studies, the findings of this dissertation suggest that developing and enacting different temporal patterns as temporal repertoires can serve as a response to high temporal complexity (Blagoev & Schreyögg, 2025; Stonig & Müller-Stewens, 2025). However, they also indicate that the number of temporal repertoires that actors can maintain is limited. This manifests in cases [B1], [B2], and [B3], where agile units were required to dedicate much more time to planning as they enacted several different planning cycles in parallel, or in cases [C] and [D], where non-agile release management units faced additional workload from stability testing. This raises the question for future research of how many temporal repertoires actors can effectively maintain without becoming overburdened. Particular attention should be given to identifying the conditions that render the parallel maintenance and enactment of temporal repertoires more or less effortful, for instance, the extent of temporal differences or the level of contradiction and conflict among various temporal repertoires.

Overall, this dissertation makes several contributions to the two synchronizing practices introduced in Chapter 3.1: entraining and developing new temporal patterns. The findings

expand knowledge on entrainment by specifically addressing its drawbacks. They demonstrate how relinquishing unique temporal patterns to achieve synchronization with a dominant Zeitgeber can produce harmful effects, thereby enriching the literature on entrainment with contingency-theoretical arguments. Building on this, the findings indicate that actors consider the fit between the temporal patterns of the dominant Zeitgeber and their work practices when deciding for or against entraining. They further show that the negative effects of entraining can be offset through the use of complementary practices, such as flexibilizing practices. In addition to their contributions to entraining, the findings of this dissertation also enhance understanding of the development of new temporal patterns as a second synchronizing practice. They illustrate that multilateral negotiation processes may increase the acceptance of new temporal patterns, as they allow temporal tensions of various actors to be better identified and translated into appropriate temporal arrangements. Finally, the findings show that new temporal patterns do not necessarily supplant established temporal patterns, but can be enacted in parallel. This suggests that actors can draw on a repertoire of distinct temporal patterns to bridge temporal differences with various actors.

6.1.2. Implications for sustaining practices

A further major contribution of this dissertation lies in advancing knowledge about temporal coordination practices that facilitate collaboration among interdependent actors while preserving their divergent temporal structures. Echoing both the findings and the previous discussion chapter, research on temporal structuring suggests that actors can derive advantages from sustaining divergent and heterogeneous temporal structures (Blagoev et al., 2024; Garud et al., 2013; Reinecke & Ansari, 2015). Yet little is known about how actors can realize a harmonious coexistence of their divergent temporal patterns without giving rise to temporal conflicts that undermine coordination and collaboration. By identifying two sustaining practices, this dissertation responds to recent calls in temporality research to investigate temporal coordination practices that allow actors to navigate tensions and conflicts of coordination without relinquishing their distinct temporal structures, thereby offering an alternative to prevailing synchronization practices (Blagoev et al., 2024; Kunisch et al., 2021).

The findings reveal two episodes in which agile and non-agile units retained their distinct temporal structures despite the emergence of temporal tensions. The first of these two episodes (Chapter 5.1) demonstrates that, by detraining, agile units abandoned extended though shared planning cycles and reverted to sprint-based cycles, thereby reinstating temporal differences. Although waiting times arose as a result of reinstated temporal differences, they were not resolved through resynchronization but rather through alternative coordination practices that enabled the sustained coexistence of divergent temporal structures. The second episode, detailed in Chapter 5.2, shows agile units choosing not to comply with the deadlines of non-agile units, effectively enacting detraining and sustaining divergent temporal structures. While the need for coordination between agile and non-agile units remained, it was addressed not by synchronization but through coordination practices that support the coexistence of their differing temporal structures. Across the eight cases, I identify two temporal coordination practices through which agile and non-agile units coordinate in such episodes: buffering and harmonizing.

Buffering constitutes a coordination practice that, to the best of my knowledge, has not yet been explored in prior research on temporality. Examples of buffering practices include the integration of specialized employees from non-agile units into agile units (see, for instance, cases [B2], [B3], and [D]), the creation of slack resources in cases [A1] and [D], and the advance preparation of relevant input in cases [A2] and [E]. Buffering practices aim to reduce the coordination needs (Galbraith, 1973) between collaborating actors during stages of their work processes in which sustaining divergent temporal structures offers inherent benefits. The specific buffering practices observed in this dissertation resonate with the approaches highlighted by Galbraith (1973) for reducing the need for information processing. They are implemented through slack resources or self-contained tasks, such as providing inputs prior to the start of agile task execution or integrating employees from non-agile units into agile teams. The findings of this dissertation thus demonstrate how the classic mechanisms outlined in Galbraith's (1973) seminal work on coordination theory underpin the preservation of divergent temporal structures.

Buffering practices function by allowing actors to exploit synchronous moments in the early stages of their collaboration to coordinate interdependent activities ex ante. In the cases focused on in this study, involving collaboration between agile and non-agile units,

this ex-ante coordination occurs during the planning stage, when agile units are entrained to the temporal structures of non-agile units. It is the ex-ante coordination that distinguishes buffering practices from the isolating practices identified and described in prior temporality literature (Dille et al., 2023; Hilbolling et al., 2022; Oborn & Barrett, 2021), even though both types of practices aim to reduce the need for coordination between two collaborating actors (Galbraith, 1973). By isolating, coordination needs between interdependent actors remain unaddressed, whereas buffering takes them into account, albeit at an earlier point in time, when shared temporal structures facilitate coordination. Buffering practices thus build on the premise that temporal structures change over time, with dynamic transitions between phases of synchrony and asynchrony.

Buffering practices are considered the preferred first-best approach to coordination by agile units. They allow agile units to approximate the ideal of self-organized and autonomous work units (Baham & Hirschheim, 2022; Kremser & Xiao, 2021; Mahringer & Danner-Schröder, 2025). Consequently, it is not surprising that, across all cases, agile units prefer to coordinate by employing buffering practices. Nonetheless, this preference is not accommodated in all cases. Cases [B1] and [E] are examples in which agile units clearly advocate for buffering practices yet are unable to implement them when collaborating with particular interdependent units. These cases shed light on two prerequisites for the successful implementation of buffering practices. In case [E], buffering practices cannot be implemented because the agile units are persistently detrained from the planning cycles of non-agile units. Thus, collaboration between the agile unit and interdependent non-agile units lacks synchronized planning cycles during the planning stage, which impedes the ex-ante coordination that buffering practices are intended to enable. As buffering practices rely on dynamic transitions between synchronous and divergent temporal structures, they cannot be implemented amid persistent temporal differences unless non-agile units have sufficient slack resources. This leads to the second condition, which is particularly pronounced in case [B1]. The implementation of buffering practices requires sufficient resources in non-agile units. For example, in case [B1], the non-agile data warehouse unit declined to second a developer into the agile project team, thereby inhibiting the enactment of buffering. While buffering practices offer solely advantages for agile units, they entail disadvantages from the perspective of non-agile units. For example, staff seconded to agile units are unavailable for the day-to-day operations of non-agile units. The time spent on the advance

preparation of tasks may be perceived as wasted when the input produced is not retrieved by agile units. Thus, buffering practices generate inefficiencies within non-agile units that, particularly when resources in non-agile units are scarce, require compensation, for example, through diminished task quality or additional strain on the remaining staff. Organizations aiming to enable buffering practices must provide adequate resources to prevent resistance stemming from resource conflicts. This entails tolerating some inefficiency, such as maintaining slack resources in non-agile units or allowing specialized staff in agile units to remain partially underutilized.

Where buffering practices are not feasible, agile and non-agile units rely on harmonizing practices to coordinate their interdependent activities during the task execution stage. Harmonizing practices aim to create and leverage ephemeral moments of synchrony that arise even as actors maintain their divergent temporal structures. The notion of achieving harmony without synchronization is not entirely new in the temporality (Ancona & Chong, 1999). Hilbolling and colleagues (2022, p. 148) identify "leveraging serendipitous alignment" as a temporal coordination practice that exploits moments of coincidental alignment among divergent temporal structures for effective coordination. Other authors refer to such moments as windows of opportunity (Tyre & Orlikowski, 1994) or Kairotic moments (Garud et al., 2025). The very definition of Kairotic moments as "intra-temporal opportune moment[s] that [are] distinct from the structured, Chronos-based inter-temporal moments" (Garud et al., 2025, p. 6) suggests that coinciding – harmonious – moments can indeed emerge within divergent temporal patterns. These studies focus on the existence of ephemeral moments of synchrony and the benefits they offer for coordination. However, they assume that these moments are the result of chance or serendipity, making them difficult to plan or control (Garud et al., 2025; Pentland et al., 2025), which limits their usability for active and deliberate coordination.

The findings challenge the assumption that synchronous moments are neither predictable nor controllable. They show that actors across all cases deliberately use harmonizing practices to induce ephemeral moments of synchrony while preserving their unique yet distinct temporal structures. Rather than being exceptional, harmonizing practices are routinely used to coordinate between agile and non-agile units. Across the eight cases, I identify four types of harmonizing practices: (1) anticipatory coordination, (2) assumption of additional tasks, (3) reprioritization of tasks, and (4) negotiations over

deadline deviations. Each of these practices indicates that actors do not merely wait for ephemeral moments of synchrony but intentionally create them by harmonizing their divergent temporal patterns without synchronizing them. Thus, they show that ephemeral moments of synchrony do not emerge by serendipity alone, but can be actively constructed through the agentic decisions of actors (Granqvist & Gustafsson, 2016).

A key prerequisite for intentionally creating ephemeral moments of synchrony is that actors are conscious of both their interdependencies and the temporal differences shaping their collaboration. Interdependencies are often not visible (Pentland et al., 2016), in part because they can emerge dynamically as actors collaborate (Deken et al., 2016; Mahringer & Danner-Schröder, 2025). The findings indicate that temporal differences may also develop emergently, for instance, when agile units detrain from non-agile planning cycles or choose not to comply with non-agile deadlines. Anticipatory coordination enables actors to detect emergent interdependencies and temporal differences, usually through initiating early and regular interaction, such as fixed meetings or attendance at daily standup meetings. This enables the use of additional harmonizing practices, through which temporal differences can be constructively addressed. In some cases, I observe that agile or non-agile units voluntarily assume unplanned tasks, even when doing so entails additional workload, albeit to a limited and unproblematic extent. The decision to assume additional tasks rests with each actor; however, it is typically influenced by their collaboration partners who construct urgency (Granqvist & Gustafsson, 2016) or appealing to social obligations based on strong personal relationships (Granovetter, 1973). Negotiations over deadline deviations involve intensive coordination to determine the extent to which such deviations can be accommodated before they impair harmonious collaboration. Task reprioritization can likewise create ephemeral moments of synchrony, whereby actors advance future-planned tasks or defer current tasks to achieve harmonized collaboration in the present or future. Reprioritizations may sometimes be initiated by the focal actor, while in other instances they are imposed through hierarchical intervention. When reprioritization is enforced hierarchically, tasks planned for the future are usually moved forward to avoid present-day waiting times. When focal actors autonomously reprioritize tasks, they usually defer current tasks and advance future tasks to facilitate harmonized collaboration in the future. This does not reduce the time that passes until the deferred tasks are handled. Waiting times are thus endured (Geiger & Harborth, 2025; Otto et al., 2024), yet they are

no longer perceived as tension, as they no longer constitute unproductive periods in which actors' capacities are unutilized (Bailey & Suddaby, 2023) and therefore do not necessarily translate into delayed delivery of work outcomes.

Conceptually, harmonizing practices differ from the synchronizing practices defined in Chapter 3.1. Although they may result in changes to existing plans, they do not affect the underlying temporal structures. For example, task reprioritizations are a reactive response to emerging waiting times, yet they do not necessitate any change to the planning horizons or planning cycles in which these actors plan their activities.

Why do actors in the case studies of this dissertation coordinate through buffering and harmonizing, rather than through isolating (Dille et al., 2023; Hilbolling et al., 2022; Oborn & Barrett, 2021), brokering (Mell et al., 2021), and negotiating practices (Magrelli et al., 2022; Reinecke & Ansari, 2015) that prior literature identified, when they aim to preserve their unique temporal patterns? The reason may be that the applicability of isolating, brokering, and negotiating practices is restricted to the specific contexts of the case studies in which they were identified. However, these contexts differ from the empirical context of this dissertation, which centers on collaboration between agile and non-agile units.

Isolating practices tend to be effective only in settings where actors collaborate either without strong interdependencies (Dille et al., 2023) or without time pressure (Hilbolling et al., 2022). The case study by Dille and colleagues (2023) demonstrates that isolating practices are feasible in a context where actors coordinate only for achieving a concurrent software release, yet conduct all development work autonomously, without depending on specialized input from collaboration partners. The interdependencies among collaboration partners are sufficiently weak that actors can work in isolation while still releasing the software on schedule. In organizational reality, such cases are rare, given that interdependencies with other units often emerge, even in contexts where units were initially designed to be autonomous (Mahringer & Danner-Schröder, 2025). Unlike in the study of Dille and colleagues (2023), the interdependencies between agile and non-agile units in this dissertation are strong. Agile units cannot carry out their activities entirely autonomously without specialized input from non-agile units. Likewise, they are unable to deliver their outcomes to end users without involving non-agile units, such as release management units. Isolating practices would prevent relevant coordination, which, as

shown by Oborn and Barrett (2021), could reduce the quality of agile units' outcomes. Given the strong interdependencies between agile and non-agile units, it is thus not surprising that they avoid isolating practices and prefer to address coordination needs proactively through ex-ante coordination using buffering practices.

While interdependencies between collaboration partners in the study by Hilbolling and colleagues (2022) were strong, the specific context of their case differs from the collaboration between agile and non-agile units in this dissertation, as it does not involve any time pressure. The interorganizational project studied by the authors spans a long time horizon and lacks a definite endpoint. The lack of time pressure facilitates isolating practices, allowing actors to interrupt their collaboration for indefinite periods when addressing divergent temporal structures through isolation, and resuming only when a window of opportunity for harmonized collaboration arises. By contrast, the agile and non-agile units studied in this dissertation operate under considerable time pressure. Their collaboration is driven by project and work assignments issued by powerful internal (e.g., management) or external actors (e.g., regulatory authorities), which must be completed within prescribed deadlines. Thus, the agile and non-agile units in nearly all cases work toward defined deadlines, which inherently creates time pressure. Agile units, in particular, face time pressure, as each sprint cycle is highly time-constrained (Junker et al., 2023; van Oorschot et al., 2018) and they are expected to deliver value quickly (Ghosh & Wu, 2023). Thus, agile units tend to adopt coordination practices that facilitate ongoing coordination of interdependent activities, using buffering practices to coordinate ex ante or harmonizing practices to actively create moments of alignment that collaboration partners in the study of Hilbolling and colleagues (2022) passively waited for.

The context in which Mell and colleagues (2021) observed brokering practices is likewise different from the empirical context of this dissertation. The authors examine globally distributed teams whose members must coordinate across different time zones. Translating time zone differences into the language of the framework I used to describe temporal structures (Ancona, Okhuysen, et al., 2001), team members perform their tasks asynchronously due to the different timing of their work hours (Espinosa & Carmel, 2003). Although Mell and colleagues (2021) provide strong evidence that brokering practices can bridge timing differences, they seem less suited for coordinating the

temporal differences between agile and non-agile units, which primarily rely on buffering and harmonizing and only occasionally employ brokering.

Unlike the timing asynchronies analyzed by Mell and colleagues (2021), the temporal differences between agile and non-agile units are considerably more complex. Although different planning cycles, deadlines, and release rhythms also constitute temporal patterns (Ancona, Okhuysen, et al., 2001), these differ from the time zone differences examined by Mell and colleagues (2021) in that they are grounded in divergent deep-level temporal conceptions and orientations (Rowell et al., 2016). Different planning cycles arise from differences in temporal depth, different deadlines represent different enactments of clock and event time, and distinct release rhythms result from varying levels of time urgency. Thus, temporal differences between agile and non-agile units are nested across multiple dimensions of temporal structures, rendering them considerably more complex.

Consequently, more complex temporal structures generate more complex coordination challenges. The primary coordination problem highlighted by Mell and colleagues (2021) is that timing incongruences prevent information from being reliably transmitted from one actor to another. This coordination problem is technical in character and can be resolved relatively easily through brokering practices, which ensure information flows between actors working asynchronously. In this dissertation, however, coordination problems are more profound, not simply resulting from temporal differences but residing within them. Even when transmitted reliably, information fails to coordinate if it conveys contested temporal patterns that conflict with actors' surface- and deep-level temporal structures and are thus neither accepted nor shared. Given the different natures of coordination problems, brokering practices are insufficient. Simply transmitting information cannot resolve temporal conflicts between agile and non-agile units and may even risk amplifying them, as it lacks mechanisms to address them constructively.

In their collaboration, the agile and non-agile units observed in this dissertation draw more extensively on buffering and harmonizing practices than on brokering. Buffering allows agile and non-agile units to retain their distinct temporal patterns during task execution by coordinating ex ante in phases of synchronization. In this way, temporal conflicts are avoided rather than actively resolved. When buffering practices are employed, brokering is not required, since no information needs to be transmitted between the actors (Galbraith, 1973). Harmonizing practices are preferred as they extend

beyond information transmission and enable constructive resolution of temporal conflicts. By engaging in anticipatory coordination, actors proactively seek to identify temporal conflicts. By assuming additional tasks and by reprioritizing tasks, they take active measures to align their divergent temporal patterns without synchronizing them, thereby creating harmonious moments for effective collaboration.

As the effectiveness of harmonizing practices depends on actors having information about their divergent temporal patterns, brokering practices appear as complementary to harmonizing practices (Smets et al., 2015). This is reflected in the findings of this dissertation. Where agile and non-agile units coordinate extensively through harmonizing (cases [A1], [B1], and [C]), they also employ brokering practices, evident in the designation of dedicated brokering responsibilities to selected employees in agile or non-agile units. Brokering practices are especially conducive to anticipatory coordination, fostering mutual awareness of (emergent) interdependencies and temporal differences.

Notably, the nature of brokering roles differs between cases (Halevy et al., 2019; Soda et al., 2018). In case [C], the broker attempts to separate the agile and non-agile units, primarily to reduce the high temporal coordination efforts that arise in agile and non-agile units from the need to coordinate across divergent temporal structures. This mirrors brokerage as described by Mell and colleagues (2021), where brokers, because of time zone differences, act merely as intermediaries and are unable to facilitate direct collaboration between actors. In cases [A1] and [B1], however, the nature of brokerage is different. Here, brokers attempt to promote direct interactions between agile and non-agile units, for example, by organizing and moderating regular meetings. Thus, the findings contribute to enriching the concept of temporal brokerage introduced by Mell and colleagues (2021) by revealing different nuances in how brokers facilitate coordination between temporally diverse actors. Given the limited use of brokering practices in this empirical context, this dissertation does not address the question of the potential implications of these different forms of brokering. It is possible that brokering, intended to foster direct collaboration between temporally diverse actors, may enhance harmonizing practices more extensively or effectively than brokering designed to separate such actors. Further research is needed to understand the implications of the different forms of brokering practices.

Finally, prior studies have recognized negotiating practices as a means to sustain divergent temporal structures (Magrelli et al., 2022; Reinecke & Ansari, 2015). As elucidated in Chapter 3.2, negotiating practices preserve differences in deep-level temporal conceptions and orientations, but not necessarily in surface-level temporal patterns. Often, negotiating practices involve the development of new temporal patterns (Fayard, 2024; Reinecke & Ansari, 2015). Resonating with prior literature, agile and non-agile units draw on negotiating practices to develop new temporal patterns (see Chapter 5.3), yet they do not apply them when they seek to sustain their unique temporal patterns.

In conclusion, the findings of this dissertation advance knowledge on sustaining practices by identifying buffering and harmonizing as novel practices that allow actors to coordinate while preserving their distinct temporal patterns. Buffering practices allow actors to take advantage of phases in which their temporal patterns are synchronized for ex-ante coordination, thereby reducing the need for coordination in later phases when preserving divergent temporal patterns is beneficial. Harmonizing practices aim to intentionally create ephemeral moments of synchrony for harmonious collaboration across divergent temporal structures, rather than treating such moments as serendipitous or coincidental. This chapter highlights that buffering and harmonizing practices differ from isolating, brokering, and negotiating practices and are particularly suited when actors (1) have sequential or reciprocal interdependencies, (2) operate under time pressure, and (3) must navigate complex temporal differences in which surface- and deep-level temporal structures are nested. In discussing the functioning of buffering practices, I have already suggested that temporal structures may evolve dynamically over the course of a work process. The findings of this dissertation also provide insights into the dynamics of temporal structuring, which I discuss in the following chapter.

6.1.3. Implications for the dynamics of temporal structuring

Finally, this dissertation enriches the field of temporality research by illuminating the dynamic process of temporal structuring that generates ongoing and recurrent changes in temporal structures. Temporality research has long been grounded in the assumption that

temporal structures are static. For example, early research on entrainment postulated that entrained temporal structures persist even in the absence of a Zeitgeber (McGrath & Kelly, 1986). Reflecting this assumption, temporal structures may enter a state of lock-in, where they become so deeply entrenched that they are hardly subject to change (Blagoev & Schreyögg, 2019; Perlow et al., 2002). Yet, current research acknowledges that temporal structures can, and often must, change dynamically, as they typically provide only temporary solutions to temporal coordination problems (Feldman et al., 2022; Hilbolling et al., 2022; Otto et al., 2024; Shipp & Richardson, 2021), and because lock-in situations often entail considerable drawbacks (Blagoev & Schreyögg, 2019; Perlow et al., 2002). Thus, temporal structuring practices must not be understood as single interventions that produce stable, unchangeable temporal structures; rather, they are recurrently applied, resulting in ongoing change in temporal structures (Ancona & Waller, 2007). Nonetheless, the dynamics of temporal structuring processes remain underexplored, as does the question of how and why temporal structures evolve over time (Blagoev et al., 2024). The findings of this dissertation shift attention from discrete temporal structuring episodes to a broader temporal structuring process, thereby enriching the understanding of sustained coordination among actors with divergent temporal structures and the mechanisms that drive changes in temporal structures.

The findings reveal dynamic temporal structuring processes in agile units, through which they recurrently change their temporal structures[12]. Dynamic temporal structuring processes are evident in Chapters 5.1 and 5.3. In Chapter 5.1, six of eight cases (patterns 1 and 2) repeatedly employ entraining and detraining practices in their work processes, thereby alternating between synchronized and divergent temporal patterns. Chapter 5.3 shows how two agile units (pattern 2) dynamically change their release rhythms by first entraining to slower release rhythms of release management units, and then leveraging the resulting tensions to develop new temporal patterns. Figure 16 illustrates the changes

[12] Not all cases exhibit dynamics in every chapter. In Chapter 5.1, patterns 3 and 4 describe two cases in which the temporal patterns of agile units are either continuously synchronized with those of non-agile units or consistently diverge. In Chapter 5.2, I observe that two out of five agile units maintain their divergent temporal structures, while the remaining five agile units entrain to the temporal patterns of non-agile units once. However, this represents only a single temporal structuring episode, as no further changes are observed after the initial entraining. In order to focus this discussion on the continuous and recurring dynamics of temporal structuring, these cases are not discussed here. The same applies to three cases in Chapter 5.3 (pattern 1), where agile units entrain only once to the release rhythms of the release management units and persistently maintain these entrained temporal patterns.

in planning cycles described in Chapter 5.1, which most clearly reflect the ongoing changes in temporal structures, induced by recurrent enactments of temporal structuring practices.

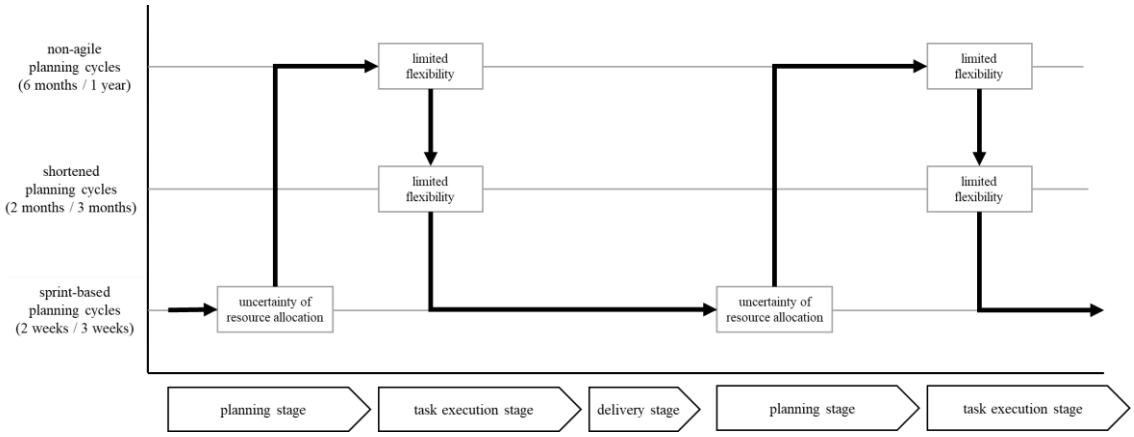

Figure 16: Dynamic changes of planning cycles

As shown in Figure 16, agile units initially plan their activities in sprint-based planning cycles of two to three weeks, but entrain to the longer planning cycles of non-agile units during the planning stage in order to reduce uncertainties related to resource allocation. The planning cycles remain synchronized up to the task execution stage. During the task execution stage, longer planning cycles of non-agile units create new tensions for agile units, limiting their flexibility and, consequently, their ability to respond to schedule disruptions. In three cases, non-agile units develop new temporal patterns by shortening their planning cycles in an attempt to reduce agile units' flexibility tensions while maintaining synchrony. However, these efforts prove ineffective because the shortened non-agile planning cycles constitute a compromise that fails to accommodate the flexibility needs of the agile units. In response to the tension that they cannot respond adequately to schedule disruptions, agile units detrain from non-agile planning cycles, thereby reverting to their sprint-based planning cycles and reestablishing divergent temporal structures. These differences persist until a new non-agile planning cycle begins, where they recreate uncertainties in resource allocation and trigger temporal restructuring to reestablish synchronized temporal structures by entraining.

Changes in temporal structures also emerge in Chapter 5.3, albeit less continuous and recurrent. In two cases, I observe how agile units do not merely entrain once to the release rhythms of non-agile release management units, but reengage in temporal structuring by leveraging the tensions arising from this entrainment to develop new temporal patterns.

Thus, the findings demonstrate how different temporal structuring episodes (entraining, detraining, and developing new temporal patterns) coalesce into a temporal structuring process in which temporal structures are dynamically modified, extending the focus of prior literature, which has concentrated on single episodes of temporal structuring (Hilbolling et al., 2022; Otto et al., 2024).

The findings also provide insight into the drivers of temporal structuring processes. Temporal structuring practices are a response to the tensions that agile units experience in collaboration with non-agile units, resulting from either shared or divergent temporal structures. Agile units enact temporal structuring practices, such as entraining, detraining, or developing new temporal patterns, to resolve emergent tensions by modifying their temporal patterns. Shifting the focus of attention from individual temporal structuring episodes to the broader temporal structuring process reveals that new tensions emerge in modified temporal structures (Hilbolling et al., 2022), either because agile units collaborate with different non-agile units whose temporal demands vary (synchronic temporal complexity) or because their own temporal demands change over time (diachronic temporal complexity) (Blagoev & Schreyögg, 2025; Garud et al., 2013). Renewed tensions elicit temporal restructuring, prompting agile units to adjust their temporal structures to the prevailing temporal demands. Thus, tensions that continuously arise from synchronic and diachronic temporal complexity are the primary drivers of dynamic temporal structuring processes.

Blagoev and colleagues (2024) advocate in their review article that research needs to theorize changes in temporal structures in order to uncover the key mechanisms that drive these changes. To contribute to this call and move beyond the currently descriptive level of analysis, it is worthwhile to take a closer look at the tensions that trigger temporal structuring episodes.

Divergent planning cycles initially create uncertainty about resource allocation in the planning stage, a tension addressed through an entraining episode. This episode reflects the efforts of agile units to attain legitimacy, thereby gaining access to resources and resolving the central tension of the planning stage (Aldrich & Fiol, 1994; Pfeffer & Salancik, 1978). Legitimacy is defined as the “generalized perception or assumption that the actions of an entity are desirable, proper, or appropriate within some socially constructed systems of norms, values, beliefs, and definitions” (Suchman, 1995, p. 574).

Actors attain legitimacy by conforming to the normative expectations of their environment (Aldrich & Fiol, 1994; Scott, 1995; Suchman, 1995). Agile units are required to entrain to the prevalent planning cycles, which have become normative expectations through their frequent and routine application (Orlikowski & Yates, 2002) in order to be recognized as legitimate (Pérez-Nordtvedt et al., 2008). While this provides an explanation for the first temporal structuring episode, it does not capture the dynamics of the broader temporal structuring process. Looking at the temporal structuring process as a whole raises a central question: Why do agile units forgo the legitimacy attained through entraining by detraining from institutionalized planning cycles, given that it will again be needed in the next non-agile planning cycle?

The findings indicate that agile units detrain from non-agile planning cycles to preserve their flexibility. This raises the question of why limitations to their flexibility are so problematic for agile units. Flexibility is widely recognized as a core agile capability (Sharifi & Zhang, 1999) and represents a primary motivation for actors to adopt agile practices (Annosi et al., 2020; Renzl et al., 2021). Thus, constraints on flexibility undermine the distinctiveness of agile practices, and thereby erode their advantages over non-agile practices.

Thus, limitations to flexibility reveal a second legitimacy concern of agile units. Agile units attain legitimacy not only by demonstrating conformity with normative expectations, such as adopting non-agile planning cycles, but also by fulfilling specific performance expectations. The literature recognizes different types of legitimacy, distinguishing between the normative legitimacy outlined above and pragmatic legitimacy (Suchman, 1995). Pragmatic legitimacy is derived from achieving certain desirable outcomes. In order to attain pragmatic legitimacy, agile units must fulfill performance expectations, notably those related to their capability to respond quickly and flexibly to change. Whereas normative legitimacy is rooted in conformity, pragmatic legitimacy stems from distinctiveness or superiority (Suddaby et al., 2017; Taeuscher et al., 2021). Thus, agile units must demonstrate that they outperform non-agile units in terms of speed, flexibility, and responsiveness. However, when agile units have to entrain to the planning cycles of non-agile units and, thereby restricting their flexibility, their distinctiveness and superiority are undermined. To attain pragmatic legitimacy, agile

units are required to preserve or restore their flexibility, explaining their detraining from non-agile planning cycles during the task execution stage.

Both normative and pragmatic legitimacy serve to secure resources, albeit at different points in time. By attaining normative legitimacy, agile units gain access to resources in the immediate present, which they inevitably need to carry out their activities. In contrast, pragmatic legitimacy is critical for obtaining resources in the future. Over the course of agile units' lifecycles, meeting performance expectations becomes an increasingly important criterion for securing legitimacy (Fisher et al., 2016). At an indeterminate future time, the agile unit must thus possess pragmatic legitimacy to continue securing resources and sustain its existence. The findings indicate that agile units pursue pragmatic legitimacy not only to sustain their specific activities but also, notably, to maintain and propagate agile practices across their organizations.

Pursuing normative and pragmatic legitimacy is inherently conflicting, given that the former is based on conformity while the latter depends on distinctiveness (B. H. Lee et al., 2017; Wry et al., 2011). To attain normative legitimacy, agile units must demonstrate conformity by entraining to non-agile planning cycles. At the same time, attaining pragmatic legitimacy requires agile units to demonstrate superior flexibility, speed, and responsiveness, which prompts them to sustain their unique but divergent temporal structures. Accordingly, the findings imply that the repeated alternations between shared and divergent temporal structures are driven by conflicting legitimacy demands and represent a deliberate approach of agile units to balance the attainment of normative and pragmatic legitimacy.

Conflicting demands for normative and pragmatic legitimacy may explain why temporal structures in some cases (Chapter 1, Patterns 3 and 4) do not change but remain stable. In these cases, legitimacy conflicts may be less salient. Case [E] focuses on a four-month agile project which unlike the other projects in the cases [B1], [C], and [D], is allocated resources only once. In the absence of future resource needs, attaining pragmatic legitimacy may be less significant in this case compared to those where agile units are also dependent on future resource allocation. By contrast, in case [A2] with the longest duration of agile practice implementation, the agile unit may already hold pragmatic legitimacy based on prior performance, necessitating less effort to attain it. As a limitation, it must be noted that while the absence of conflicting legitimacy efforts offers

one potential explanation for the stability of certain temporal structures, alternative explanations may exist that the empirical data did not uncover. At this stage, further research is required to identify and isolate the mechanisms that may account for the persistence of temporal structures.

While planning cycles display ongoing alternations between synchronization and divergence, the dynamics of agile units' release rhythms differ. Agile units initially entrain their sprint-based release rhythms to the slower release rhythms of non-agile release management units. Subsequently, they collaboratively develop new release rhythms that constitute a compromise between sprint-based and often semiannual release rhythms. Once established, these new release rhythms endure without undergoing further change. Thus, the changes in release rhythms are less recurrent and less dynamic than those in planning cycles, although legitimacy conflicts persist. Normative legitimacy secures access to resources, requiring agile units to entrain to the slower release rhythms of release management units to be able to use the infrastructure for delivering software. However, the slower release rhythms undermine the agile units' capacity for rapid delivery, thereby restricting pragmatic legitimacy.

The differing dynamics in temporal structure changes can be explained by the degree to which the use of resources is constrained in time. In the context of different planning cycles, agile units aim to secure access to resources allocated by management. These resources are provided for defined periods of time, during which agile units can access and utilize them with relative autonomy. The availability of resources during these periods of time allows agile units to revert to their temporal patterns, thereby preserving flexibility. By contrast, release management units provide a resource that is highly time-bounded. Agile units can use the infrastructure for software releases at single points in time. Thus, when collaborating with release management units, they do not have windows of opportunity to revert to their sprint-based release rhythms, explaining why release rhythms change less dynamically than planning cycles. By contrasting the dynamics of changes in planning cycles and release rhythms, the findings not only contribute to explaining variation in the dynamics of temporal structures but also expand the set of contingencies that may influence entraining and detraining decisions (Shipp & Richardson, 2021) by proposing the temporal constraints of resource use as an additional, non-relational factor.

Overall, the findings enhance our understanding of the dynamics of temporal structuring by shifting the analytical focus from single temporal structuring episodes toward a broader temporal structuring process that is characterized by recurring dynamics that drive the recurrent changes of temporal structures. The findings indicate that actors engage in recurrent temporal structuring to respond to temporal tensions, which emerge and reemerge repeatedly due to both synchronic and diachronic temporal complexities. Accordingly, they advance our understanding of coordination between actors with divergent temporal structures, emphasizing that it is a sustained process of enacting both synchronizing and sustaining practices. Furthermore, the findings suggest that recurring temporal tensions reflect conflicts between normative and pragmatic legitimacy attainment, which is balanced by alternating between shared and divergent temporal structures. Thus, this dissertation responds to the call for theorizing changes of temporal structures (Blagoev et al., 2024).

6.2. Implications for research on agile practices

Agile practices, originating from organizational practice, have in the last two decades become an established field of research in management and information systems. Scholars in this field have focused on defining and conceptualizing agile practices (Abrahamsson et al., 2009; Baham & Hirschheim, 2022; Conboy, 2009; Junker et al., 2023), examining how they shape team effectiveness through mechanisms such as empowerment (Grass et al., 2020), control (Khanagha et al., 2022), knowledge sharing (Ghobadi & Mathiassen, 2016; T. L. Huber et al., 2020), or coordination (Ghosh & Wu, 2023), and investigating their positive effects on outcomes such as team performance (Kude et al., 2019) or innovation (Annosi et al., 2020). The emphasis has clearly been on examining the agile unit, that is, the organizational unit applying agile practices, as an isolated entity in order to understand these practices and their effects.

Only a small body of work has shifted away from viewing the agile unit in isolation, instead considering it as embedded within an organizational context that shapes both the enactment and the outcomes of agile practices (Abrahamsson et al., 2009; Cram, 2019).

To date, research has tended to overlook that agile units often exist within organizations in which plan-based work practices prevail, requiring coexistence and collaboration between agile and non-agile units. The literature on agile transformations concedes that organizations are not fully agile, yet frames the coexistence of agile and plan-based practices as a temporary stage that concludes upon completion of the transformation process (Carroll et al., 2023; Dikert et al., 2016; Paasivaara et al., 2018). Consequently, it offers few solutions for fostering harmonious coexistence and collaboration between agile and non-agile units, as its focus is primarily on how non-agile units can be transformed into agile units. Despite many organizations pursuing agile transformations in response to contemporary management trends (Cram & Newell, 2016), it remains essential to explore how agile and non-agile practices can coexist across organizational units and collaborate effectively. Especially during the early stages of an agile transformation, when agile practices are confined to selected units (Beretta & Smith, 2023), agile and non-agile units must inevitably collaborate. If such collaboration fails, it can undermine the performance of both agile and non-agile units, affect the organization as a whole, and put the agile transformation at risk. More importantly, in most organizations, routine tasks, which are more effectively managed through non-agile work practices, persist alongside the complex and uncertain tasks suited for agile practices (Abrahamsson et al., 2009; Cram & Newell, 2016). Accordingly, most organizations are unlikely to undergo a complete transformation to agile practices, thus requiring ongoing coordination and collaboration between agile and non-agile units.

By focusing on the coexistence and collaboration between agile and non-agile organizational units, this study addresses a significant gap in the literature on agile practices, offering three contributions to this body of work. First, it conceptualizes and empirically validates temporal differences between agile and plan-based work practices. This is not only a prerequisite for understanding tensions in their collaboration, but also advances the conceptual and definitional understanding of agile practices by identifying temporal structures as contextual factors that distinguish agile from plan-based practices (Tripp et al., 2018). Second, the findings of this dissertation highlight that temporal differences constitute a source of tension between agile and non-agile units, posing considerable challenges to their coexistence and collaboration. Resolving these tensions often requires temporal structuring, which alters the temporal patterns of agile units to align with those of non-agile units. Thus, this dissertation presents the emergence of agile

method tailoring as the outcome of the processes through which agile and non-agile resolve the temporal tensions that impede their collaboration. Third, while agile practices are typically implemented to cultivate agile capabilities, this outcome does not always materialize. The findings suggest that temporal differences between agile and non-agile units may explain why agile practices may fail to translate into agile capabilities. The first contribution to research on agile practices lies in the conceptualization and empirical validation of temporal differences inherent in agile and plan-based work practices. Prior research has documented various structural, procedural, and cultural differences between agile and plan-based work practices. Structural differences pertain to the distinct organizational structures of agile and plan-based practices, including differences in specialization, role structures, and degrees of decentralization (Boehm & Turner, 2004; Kusters et al., 2017; Misra et al., 2010; Nerur et al., 2005). Procedural differences refer to the distinct organization of core and support processes, such as development, knowledge management, planning, control, and coordination processes (Boehm & Turner, 2004; Kusters et al., 2017; Nerur et al., 2005; Theobald & Diebold, 2018). Cultural differences reflect the distinct organizational cultural foundations of agile and plan-based practices (Iivari & Iivari, 2011; Zinn et al., 2023). Although temporality underlies social and organizational reality that is implicitly reflected in both processes and culture (Hall, 1983), prior research has not explicitly addressed temporal differences between agile and plan-based practices.

The findings of this dissertation indicate that agile and plan-based practices embody distinct temporal structures along three dimensions (Ancona, Okhuysen et al., 2001). First, they diverge in their surface-level temporal patterns. Plan-based practices exhibit linear temporal patterns, in which activities are conducted sequentially and both the discrete phases and the overall work processes are finite and bounded by deadlines. By contrast, agile practices embody cyclical temporal patterns with activities organized into time-boxed sprints that may be repeated indefinitely at rapid rhythms. These distinct temporal patterns mirror differences in the deep-level temporal conceptions grounded in agile and plan-based practices. At the process level, agile practices are characterized by an event-time conception, whereas plan-based practices exhibit a clock-time conception. Furthermore, agile and plan-based practices differ in terms of the temporal orientations they accommodate. Agile practices reflect low temporal depth, restricting planning to the timeframe of a sprint. Plan-based practices display greater temporal depth, planning for

the entirety of their work processes. Agile practices also cultivate higher levels of time urgency. The principle of delivering an output at the end of every sprint cycle, even if imperfect, instills greater time urgency in agile practices compared to plan-based practices, which, in contrast, deliver fully finalized outcomes only once at the very end of the process.

By describing their specific temporal structures, the findings of this dissertation add to the definitory characteristics that distinguish agile from plan-based practices. In contrasting the temporal foundations of agile and plan-based practices and highlighting their distinctiveness, the results address recurring concerns about the definitional boundaries of agile practices (Conboy, 2009; Junker et al., 2023; Tripp et al., 2018) and advance the conceptual and definitional clarity of agile practices. In this way, the findings complement recent scholarly efforts to define agile practices through the specification of sub-practices (Junker et al., 2023; Tripp & Armstrong, 2018) by introducing their temporal foundations as a salient distinguishing characteristic.

The different temporal structures of agile and non-agile units underpin the second contribution of this dissertation to research on agile practices. Divergent temporal structures are a source of tension that may impede the harmonious coexistence and collaboration of agile and non-agile units. Due to different planning cycles, agile units face uncertainties over resource allocation and waiting times, along with constraints on their flexibility. Different event-time and clock-time conceptions shape the distinct use of deadlines in agile and non-agile units. In non-agile units, this generates uncertainty about the timely completion of outcomes, while in agile units, it limits flexibility and responsiveness. Short release rhythms in agile units, indicative of strong time urgency, give rise to quality concerns in non-agile units, while the slower release rhythms of non-agile units limit agile units' speed.

The findings enhance understanding of the tensions that arise in collaborations between agile and non-agile units in two distinct ways. First, temporal differences between agile and non-agile units represent a previously unexplored source of tension. They add to the structural, procedural, and cultural distinctions that earlier studies have documented (Kusters et al., 2017; Misra et al., 2010; Nerur et al., 2005; Theobald & Diebold, 2018). Second, they help overcome the previously one-sided focus on agile units by analyzing tensions in both agile and non-agile units, revealing that the tensions experienced by non-

agile units are qualitatively different from those in agile units. This insight provides an important foundation for developing viable solutions that ensure collaboration between agile and non-agile units and that are equally accepted and applied by both.

To date, the question of how tensions between agile and non-agile units can be resolved to achieve harmonious coexistence and collaboration remains largely unanswered. Prior studies that identified, classified, and interpreted differences between agile and non-agile units as sources of tensions did not explore practices for overcoming them (Kusters et al., 2017; Misra et al., 2010; Nerur et al., 2005; Theobald & Diebold, 2018). This dissertation addresses this gap. It highlights various practices that facilitate effective collaboration between agile and non-agile units. These include temporal structuring practices (entraining, detraining, and developing new temporal patterns) employed by agile units to modify their temporal structures and create a common ground for collaboration with non-agile units. They also encompass the buffering and harmonizing practices discussed in Chapter 6.1.2, which allow coordination while preserving the distinct temporal structures of agile and non-agile units.

Temporal structuring practices are especially noteworthy in the context of agile practice research. Changes in temporal structures effectively induce modifications to the agile practices themselves. This is particularly evident in Chapter 5.2, where non-agile deadlines overwrite the cyclical and open-ended temporal patterns of agile units, thereby constraining iterative development as a core practice of agile work (Ghosh & Wu, 2023; Junker et al., 2023). Chapter 5.3 further demonstrates how agile units, by entraining to the slower release cycles of non-agile units, cease to deliver minimum viable products. Thus, temporal structuring practices that are applied to resolve temporal tensions effectively constitute agile method tailoring (Conboy & Fitzgerald, 2010; Cram, 2019), which is used to create hybrid practices by replacing agile sub-practices that are deeply embedded in agile temporal structures with plan-based sub-practices.

Observing how agile method tailoring results from the necessity to reconcile temporal differences between agile and non-agile units has implications for research on agile method tailoring. The findings extend current perspectives on the contingency factors of agile method tailoring, which explain why practitioners modify and adapt agile methods (Baham & Hirschheim, 2022; Campanelli & Parreiras, 2015; Kalus & Kuhrmann, 2013). Prior research has explained the need for agile method tailoring primarily based on the

necessity of achieving a fit between agile practices and the strategic goals of the agile unit (Bass, 2016; Campanelli & Parreiras, 2015; Tripp & Armstrong, 2018). The findings show that the efforts of agile units to collaborate with actors in traditional, plan-based organizational contexts, whose temporal structures differ, represent an additional driver of agile method tailoring. Thus, the findings illuminate previously underexplored relational and temporal contingencies that help explain why actors engage in agile method tailoring.

Temporal structuring practices, along with the agile method tailoring they entail, can create a common ground between agile and non-agile units and facilitate effective collaboration. However, actors must be aware that agile method tailoring can dilute the benefits of agile practices (Cram et al., 2019; Beretta & Smith, 2023). Several cases in this dissertation illustrate that agile method tailoring, induced by the enactment of temporal structuring practices, caused agile units to experience lasting limitations in flexibility, speed, or responsiveness. Thus, temporal structuring practices that synchronize agile and non-agile temporal patterns, and thereby modify agile practices, should be used purposefully to create common ground at strategically important moments for collaboration. Yet, coordination between agile and non-agile units should be supported by practices that preserve their distinct practices, such as the harmonizing and buffering practices discussed in Chapter 6.1.2.

This dissertation makes a third contribution by advancing our understanding of when and under what conditions agile practices translate into agile capabilities. Agile practices and agile capabilities are conceptually distinct (Renzl et al., 2021). Agile practices are commonly understood as the set of practices underlying various agile management frameworks. These include, but are not limited to, sprint-based planning, stand-up meetings, iterative development, and retrospectives (Ghosh & Wu, 2023; Junker et al., 2023). Agile capabilities, in contrast, refer to an organization's ability to respond quickly and flexibly to changes in its environment and to recognize such changes as exploitable opportunities (Agarwal et al., 2006; Cegarra-Navarro et al., 2016; Sharifi & Zhang, 1999; Tallon & Pinsonneault, 2011). Agile capabilities are often conceptualized as an integrated set of sub-capabilities, namely flexibility, speed, and responsiveness (Sharifi & Zhang, 1999). The connection between agile practices and agile capabilities is typically framed as an input-output relationship. Organizations, therefore, employ agile practices in the expectation, or hope, of enhancing their agile capabilities (Klarner et al., 2025). Although

agile practices may serve as antecedents of agile capabilities, their adoption does not always yield the expected outcomes (Conboy, 2009; Niederman et al., 2018; Renzl et al., 2021). This raises the question of why agile practices do not consistently enhance agile capabilities (Nguyen et al., 2024) and under what conditions such adoption does, in fact, translate into agile capability development (Renzl et al., 2021).

To date, research has framed the missing link between agile practices and agile capabilities as a problem of scaling agile practices across organizational levels. Niederman and colleagues (2018) observe that while agile practices are implemented at the team or project level, agile capabilities are typically measured at the organizational level. Building on this, Renzl and colleagues (2021) explain the failure to translate agile practices into agile capabilities to problems of scaling them across the organization. This reasoning implicitly rests on the assumption that agile capabilities emerge at the level of the unit where agile practices are applied, but that these capabilities remain unmeasured and become diluted at the organizational level.

The findings of this dissertation challenge this assumption. They demonstrate that, even at the level of individual agile units, the use of agile practices does not translate into agile capabilities. Thus, agile capabilities do not necessarily become diluted at the organizational level; they may fail to develop altogether in the unit that adopts agile practices. Agile practices become disconnected from agile capabilities at the unit level because agile units abandon their unique temporal structures to synchronize with non-agile units. This results in a misfit between their (synchronized) temporal structures and the agile practices themselves, which diminishes the effectiveness of the latter. The loss of effectiveness becomes particularly evident in cases where, for example, agile units entrained to non-agile deadlines and shifted from iterative to sequential development, or when they reported that the slow release cycles of non-agile release management units limited their speed. Thus, the findings of this dissertation demonstrate that agile practices require conducive conditions in order to develop into agile capabilities and emphasize that one such condition is temporal structures that do not conflict with the core values and principles of agile practices.

In summary, the findings of this dissertation make three contributions to research on agile methods. First, by conceptualizing and empirically validating differences in the temporal structures of agile and non-agile units, they advance efforts to sharpen definitions and

criteria for distinguishing agile from non-agile practices. Second, the findings reveal how temporal differences give rise to tensions in collaborations between agile and non-agile units, thus adding to the relatively small body of literature that addresses the coexistence and collaboration of agile and non-agile units. Contributing further to this stream of research, the findings demonstrate the role of temporal structuring and coordination practices in alleviating the temporal tensions that obstruct collaboration between agile and non-agile units. They show that temporal structuring practices can induce agile method tailoring, thereby modifying the agile practices themselves. This broadens current perspectives on the contingency factors underlying agile method tailoring by adding relational and temporal contingencies. Finally, the findings provide insight into the missing link between agile practices and agile capabilities, suggesting that a lack of alignment between agile practices and the temporal structures of agile units explains why the adoption of agile practices does not automatically yield agile capabilities.

6.3. Limitations and future research directions

This dissertation is subject to certain limitations, which I will discuss in this concluding section of the discussion. The core limitations lie in the limited generalizability of case study research, the constraints of the most-similar case selection to explaining cross-case variation, the absence of quantitative data on the effectiveness and performance of the coordination practices, and the data collection being conducted under the specific constraints imposed by contact restrictions during the COVID-19 pandemic.

An inherent limitation of qualitative case study research is its lack of statistical generalizability to a larger population (Siggelkow, 2007). This also applies to the findings presented in this dissertation. While the multiple-case design allows for greater analytical generalizability than single-case designs (Gibbert & Ruigrok, 2010; Yin, 2009), the practices and dynamics I identified for coordinating across divergent temporal structures emerged within the specific context of collaboration between agile and non-agile units. Given that research on sustaining practices and dynamics is still at an early stage, and that case studies continue to reveal different practices or dynamics (Dille et al., 2023;

Hilbolling et al., 2022; Mell et al., 2021; Reinecke & Ansari, 2015), this dissertation does not claim to provide a complete or cross-contextually generalizable account of coordination across divergent temporal structures. Instead, it aims to encourage researchers to explore as-yet under- or unexplored contexts in which actors with divergent temporal structures coordinate. In doing so, future research could either strengthen the transferability of the practices identified here and in prior studies or uncover new practices that prove effective under the specific conditions of future case studies.

A second limitation arises from the selection of cases. This dissertation draws on eight case studies that were purposefully sampled based on their similarity (Patton, 2015). The purpose of this case selection was to identify a context in which temporal differences between highly interdependent actors clearly surface, making it possible to observe and describe coordination across temporal structures. While this case selection provides a degree of robustness, as the eight cases reveal similar coordination practices and dynamics, it constrains the potential to theorize and account for differences across cases (Eisenhardt, 1989; Herriott & Firestone, 1983). This becomes particularly evident where the temporal structuring practices vary between cases. Such variation is evident in each chapter. In Chapter 5.1, six cases exhibit dynamic shifts between shared and divergent temporal structures (patterns 1 and 2), whereas the two remaining cases display stable temporal structures that are either persistently synchronized (pattern 3) or persistently divergent (pattern 4). In Chapter 5.2, the first pattern shows persistent entraining in five cases, coupled with complementary flexibilizing practices, while the remaining two cases in the second pattern display persistent detraining, with its disadvantages counterbalanced through harmonizing practices. In Chapter 5.3, three cases persistently entrain to the slower release rhythms of non-agile units (pattern 1), whereas two cases reveal a dynamic shift from entrained to new temporal patterns (pattern 2). These variations raise the particularly important question of why the patterns identified within each chapter differ.

Although the dissertation offers promising approaches to explaining the different patterns, such as the presence of conflicts between normative and pragmatic legitimacy in accounting for the varying dynamics of planning cycles in Chapter 5.1, or power relations between actors and potential Zeitgebers in explaining different responses to deadlines in Chapter 5.2, further research is needed to test and challenge the robustness of these explanations. Future research could particularly benefit from a case selection that

deliberately contrasts previously theorized boundary conditions by applying a most-different design (A. L. George & Bennett, 2005; Yin, 2009).

In particular, future research is needed to understand when temporal structures undergo dynamic change, and when they remain stable, potentially leading to a lock-in (Blagoev et al., 2024; Blagoev & Schreyögg, 2019). By discussing legitimacy conflicts in Chapter 6.1.3, this dissertation has outlined a rationale for why temporal structures may change dynamically. Yet alternative explanations are conceivable and merit empirical scrutiny. One such explanation, which the selected cases did not allow to be tested, concerns whether, and to what extent, the temporal structures of non-agile Zeitgebers are embedded within agile units themselves. In this regard, two aspects may be of particular importance. First, building on the findings of this dissertation, non-agile temporal structures could, over time, become embedded in agile units through translation processes, in which the agile unit is continuously adapted to its organizational context (Beretta & Smith, 2023). While the case studies selected for this dissertation display only differences in the duration of application of agile practices within the respective agile units, they are relatively small. Future studies could generate stronger contrasts through a targeted case selection and examine the temporal structures of agile units at different stages of their life cycles. Second, the specific deep-level temporal structures of individuals in agile units may determine whether agile units tend to entrain to the temporal structures of a non-agile Zeitgeber, or whether they resist those structures whenever windows of opportunity arise, thereby triggering dynamic shifts between shared and divergent temporal structures to create congruence between deep- and surface-level temporal structures (Rowell et al., 2016). To explore the role of individual temporal structures, future research could draw on established instruments from psychological research to quantitatively assess individual temporal orientations (e.g., Conte et al., 1995) and apply them within a multi-level analysis (Klein & Kozlowski, 2000) to evaluate their potential to explain varying dynamics of temporal structures.

A third limitation concerns the absence of an objective evaluation of the efficiency and effectiveness of the coordination practices and dynamics identified. This dissertation primarily sought to describe how coordination across divergent temporal structures unfolds. Considering the variety of practices and dynamic patterns of coordination, the question inevitably arises whether all these patterns are equally effective under their

respective contextual conditions, or whether their efficiency and effectiveness for coordination and task performance differ. While this dissertation relies on subjective evaluations and assessments of individuals from agile and non-agile units to answer this question, it lacks a more objective perspective that is based on quantitative data. The lack of connection between practices and objectively measured outcomes is a noted limitation in research on organizational practices (Jarzabkowski et al., 2016; Kouamé & Langley, 2018). Future research should aim to bridge the gap between practices and their outcomes, especially in the context of coordinating across temporal differences, the central research focus of this dissertation. Methodologically, researchers could focus on isolating different coordination practices, such as buffering or harmonizing, through targeted experimental designs in order to compare their effects on coordination and task performance. Future case study research might benefit from a mixed-methods approach (Yin, 2009), enabling researchers to capture both detailed descriptions of practices and quantitative data on relevant performance indicators. In the specific contexts of agile units, researchers could leverage agile artifacts such as Kanban boards or burndown charts to quantify relevant performance indicators. For example, burndown charts can provide quantifiable data about task completion, while Kanban boards may allow for measuring task progress over time. When combined with a multiple-case design that contrasts agile units employing different coordination practices, using these artifacts to quantitatively measure performance indicators could yield insights into their effectiveness.

Finally, attention is given to the constraints that shaped data collection. We collected data from June 2021 to September 2023, during a period of strict contact restrictions aimed at controlling the COVID-19 pandemic. These contact restrictions particularly affected the ethnographic component of data collection, which was largely confined to participation in virtual meetings. While it enabled us to observe and describe formal coordination practices, it limited our ability to identify informal practices through which actors coordinate in situ (Jarzabkowski et al., 2012; Okhuysen & Bechky, 2009). Yet, the contact restrictions limited not only our access to the field but also on-site collaboration between agile and non-agile units. As staff were instructed to work remotely, coordination between agile and non-agile units largely took place within the formal meetings we observed. In particular, the participant observation in case [E], where I joined the agile unit as a temporary team member for one month, confirmed that informal interactions between agile and non-agile units were minimal beyond formal meetings. Although this indicates

that we were able to capture the reality of coordination between agile and non-agile units, it raises the question of whether the practices and dynamics presented in the findings of this dissertation would persist in settings that allow for more informal and situational coordination. Future research could address this question by examining the processes and practices used to coordinate across different temporal structures, either through replication under current non-restricted conditions or by comparing coordination in virtual and face-to-face teams.

7. Conclusion

The point of departure for this dissertation was the question of how actors with divergent temporal structures can coordinate their interdependent activities. This question has emerged from the recognition that current scholarship provides only a partial and fragmented understanding of temporal coordination. By examining coordination in the revelatory context of collaborations between agile and non-agile units in the financial sector, this dissertation advances this body of research by integrating previously separated and fragmented perspectives on synchronizing and sustaining practices, moving toward a more holistic account of coordination across divergent temporal structures.

A reflection on the dissertation's contributions reveals a unifying insight: synchronizing and sustaining practices are not distinct approaches to be considered in isolation. Each contribution in this dissertation highlights how synchronizing and sustaining practices intersect and interact in different ways. By illuminating the drawbacks of entraining, the agentic role of non-time-setting actors in entraining decisions, and the superiority of multilateral negotiation processes for the acceptance and adoption of new temporal patterns, this dissertation shows that synchronizing decisions are inherently linked to sustaining decisions. Understanding why actors seek to sustain their unique temporal structures is essential for understanding why they resist entraining or refrain from synchronizing with new temporal patterns. The recognition of buffering as a novel sustaining practice demonstrates how the boundaries between synchronizing and sustaining can blur. Buffering enables actors to coordinate effectively while sustaining heterogeneity in their temporal structures, leveraging moments of synchrony for preemptive coordination. Thus, buffering practices harness the dynamics between synchronizing and sustaining, illustrating how the two approaches can be mutually reinforcing. However, the contribution to the dynamics of temporal structuring most clearly demonstrates the complementarity of synchronizing and sustaining practices. It reveals how actors coordinate effectively when they alternate dynamically between synchronizing and sustaining practices to exploit the benefits of both shared and divergent temporal structures at different times. Altogether, these contributions show that

synchronizing and sustaining practices are not mutually exclusive, but can be combined to achieve effective coordination that persists over time.

This dissertation allows us to transcend the prevailing dichotomy between synchronizing and sustaining practices, reconceptualizing them as intertwined elements of temporal coordination. This reconceptualization invites future research to explore temporal coordination in light of the interplay and dynamics of synchronizing and sustaining practices. It does so with the expectation that broadening the perspective from isolated synchronizing and sustaining practices to an integrated process that recognizes and leverages their complementarities will uncover innovative approaches to coordination that might not be limited to transient episodes but could contribute to lasting improvements in coordination across temporal differences.

References

Abrahamsson, P., Conboy, K., & Wang, X. (2009). 'Lots done, more to do': The current state of agile systems development research. *European Journal of Information Systems*, *18*(4), 281–284. https://doi.org/10.1057/ejis.2009.27

Adams, S. (1995). *Timewatch: The social analysis of time*. Polity Press.

Agarwal, A., Shankar, R., & Tiwari, M. K. (2006). Modeling the metrics of lean, agile and leagile supply chain: An ANP-based approach. *European Journal of Operational Research*, *173*(1), 211–225. https://doi.org/10.1016/j.ejor.2004.12.005

Agrawal, A., Catalini, C., Goldfarb, A., & Luo, H. (2018). Slack time and innovation. *Organization Science*, *29*(6), 1056–1073. https://doi.org/10.1287/orsc.2018.1215

Aguinis, H., & Bakker, R. M. (2021). Time is of the essence: Improving the conceptualization and measurement of time. *Human Resource Management Review*, *31*(2). https://doi.org/10.1016/j.hrmr.2020.100763

Aldrich, H. E., & Fiol, C. M. (1994). Fools rush in? The institutional context of industry creation. *Academy of Management Review*, *19*(4), 645–670. https://doi.org/10.5465/amr.1994.9412190214

Alimadadi, S., Davies, A., & Tell, F. (2022). A palace fit for the future: Desirability in temporal work. *Strategic Organization*, *20*(1), 20–50. https://doi.org/10.1177/14761270211012021

Amis, J., Slack, T., & Hinings, C. R. (2004). The pace, sequence, and linearity of radical change. *Academy of Management Journal*, *47*(1), 15–39. https://doi.org/10.5465/20159558

Ancona, D. G., & Chong, C.-L. (1996). Entrainment: Pace, cycle, and rhythm in organizational behavior. In B. M. Staw & L. L. Cummings (Eds.), *Research in organizational behavior* (Vol. 18, pp. 251–284). JAI Press.

Ancona, D. G., & Chong, C.-L. (1999). Cycles and synchrony: The temporal role of context in team behavior. In E. A. Mannix & M. A. Neale (Eds.), *Research on managing groups and teams* (Vol. 2, pp. 33–48). JAI Press.

Ancona, D. G., Goodman, P. S., & Lawrence, B. S. (2001). Time: A new research lens. *Academy of Management Review*, *26*(4), 645–663. https://doi.org/10.5465/amr.2001.5393903

Ancona, D. G., Okhuysen, G. A., & Perlow, L. A. (2001). Taking time to integrate temporal research. *Academy of Management Review*, *26*(4), 512–529. https://doi.org/https://doi.org/10.5465/amr.2001.5393887

Ancona, D. G., & Waller, M. J. (2007). The dance of entrainment: Temporally navigating across multiple pacers. In B. A. Rubin (Ed.), *Research in the Sociology of Work* (Vol. 17, pp. 115–146). Emerald. https://doi.org/10.1016/S0277-2833(07)17004-7

Annosi, M. C., Foss, N., & Martini, A. (2020). When agile harms learning and innovation (and what can be done about it). *California Management Review*, *63*(1), 61–80. https://doi.org/10.1177/0008125620948265

Antoni, A., Reinecke, J., & Fotaki, M. (2023). Making time to care, and caring for time: 'Tricking time' to cope with conflicting temporalities in a child protection agency. *Journal of Business Ethics*, *188*(4), 645–663. https://doi.org/10.1007/s10551-023-05507-9

Augustine, G., Soderstrom, S., Milner, D., & Weber, K. (2019). Constructing a distant future: Imaginaries in geoengineering. *Academy of Management Journal*, *62*(6), 1930–1960. https://doi.org/10.5465/amj.2018.0059

BaFin. (2016, January 1). *Solvency II*. https://www.bafin.de/DE/Aufsicht/VersichererPensionsfonds/Allgemeines/SolvencyII/solvency_II_node.html

BaFin. (2024, May 29). *Rundschreiben 06/2024 (BA): Mindestanforderungen an das Risikomanagement - MaRisk*. https://www.bafin.de/SharedDocs/Downloads/DE/Rundschreiben/dl_rs_06_2024_MaRisk_pdf_BA.pdf?__blob=publicationFile&v=2

Baham, C., & Hirschheim, R. (2022). Issues, challenges, and a proposed theoretical core of agile software development research. *Information Systems Journal*, *32*(1), 103–129. https://doi.org/10.1111/isj.12336

Bailey, C., & Suddaby, R. (2023). When time falls apart: Re-centring human time in organisations through the lived experience of waiting. *Organization Studies*, *44*(7), 1033–1053. https://doi.org/10.1177/01708406231166807

Bakker, R. M., & Knoben, J. (2015). Built to last or meant to end: Intertemporal choice in strategic alliance portfolios. *Organization Science*, *26*(1), 256–276. https://doi.org/10.1287/orsc.2014.0903

Ballard, D. I. (2009). Organizational temporality over time: Activity cycles as sources of entrainment. In R. A. Roe, M. J. Waller, & S. R. Clegg (Eds.), *Time in organizational research* (1st ed., pp. 226–241). Routledge.

Ballard, D. I., & Seibold, D. R. (2000). Time orientation and temporal variation across work groups: Implications for group and organizational communication. *Western Journal of Communication*, *64*(2), 218–242. https://doi.org/10.1080/10570310009374672

Ballard, D. I., & Seibold, D. R. (2003). Communicating and organizing in time: A meso-level model of organizational temporality. *Management Communication Quarterly*, *16*(3), 380–415. https://doi.org/10.1177/0893318902238896

Bansal, P., & Corley, K. (2012). Publishing in AMJ — Part 7: What's different about qualitative research? *Academy of Management Journal*, *55*(3), 509–513. https://doi.org/10.5465/amj.2012.4003

Bansal, P., Reinecke, J., Suddaby, R., & Langley, A. (2022). Temporal work: The strategic organization of time. *Strategic Organization*, *20*(1), 6–19. https://doi.org/10.1177/14761270221081332

Barbero, J. L., Di Pietro, F., & Chiang, C. (2017). A rush of blood to the head: Temporal dimensions of retrenchment, environment and turnaround performance. *Long Range Planning*, *50*(6), 862–879. https://doi.org/10.1016/j.lrp.2017.02.004

Bartel, C. A., & Garud, R. (2009). The role of narratives in sustaining organizational innovation. *Organization Science*, *20*(1), 107–117. https://doi.org/10.1287/orsc.1080.0372

Bartel, C. A., & Milliken, F. J. (2003). Perceptions of time in work groups: Do members develop shared cognitions about their temporal demands? *Research on Managing Groups and Teams*, *6*, 87–109. https://doi.org/10.1016/S1534-0856(03)06005-5

Bartunek, J. M., & Necochea, R. A. (2000). Old insights and new times: Kairos, Inca cosmology, and their contributions to contemporary management inquiry. *Journal of Management Inquiry*, *9*(2), 103–113. https://doi.org/10.1177/1056492600920

Baskerville, R. L., & Pries-Heje, J. (2004). Short cycle time systems development. *Information Systems Journal*, *14*(3), 237–264. https://doi.org/10.1111/j.1365-2575.2004.00171.x

Bass, J. M. (2016). Artefacts and agile method tailoring in large-scale offshore software development programmes. *Information and Software Technology*, *75*, 1–16. https://doi.org/10.1016/j.infsof.2016.03.001

Batra, D., van der Meer, D., & Dutta, K. (2011). Extending agile principles to larger, dynamic software projects: A theoretical assessment. *Journal of Database Management*, *22*(4), 73–92. https://doi.org/10.4018/jdm.2011100104

Bechky, B. A. (2006). Gaffers, gofers, and grips: Role-based coordination in temporary organizations. *Organization Science*, *17*(1), 3–21. https://doi.org/10.1287/orsc.1050.0149

Bechky, B. A., & Okhuysen, G. A. (2011). Expecting the unexpected? How SWAT officers and film crews handle surprises. *Academy of Management Journal*, *54*(2), 239–261. https://doi.org/10.5465/amj.2011.60263060

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *Manifesto for agile software development*. https://agilemanifesto.org/

Bednar, J. S., Galvin, B. M., Ashforth, B. E., & Hafermalz, E. (2020). Putting identification in motion: A dynamic view of organizational identification. *Organization Science*, *31*(1), 200–222. https://doi.org/10.1287/orsc.2018.1276

Beersma, B., Hollenbeck, J. R., Humphrey, S. E., Moon, H., Conlon, D. E., & Ilgen, D. R. (2003). Cooperation, competition, and team performance: Toward a contingency approach. *Academy of Management Journal*, *46*(5), 572–590. https://doi.org/10.5465/30040650

Bentley, F. S., & Kehoe, R. R. (2020). Give them some slack—they're trying to change! The benefits of excess cash, excess employees, and increased human capital in the strategic change context. *Academy of Management Journal*, *63*(1), 181–204. https://doi.org/10.5465/amj.2018.0272

Berends, H., van Burg, E., & Garud, R. (2021). Pivoting or persevering with venture ideas: Recalibrating temporal commitments. *Journal of Business Venturing*, *36*(4). https://doi.org/10.1016/j.jbusvent.2021.106126

Beretta, M., & Smith, P. (2023). Embarking on a business agility journey: Balancing autonomy versus control. *California Management Review*, *65*(4), 93–115. https://doi.org/10.1177/00081256231177718

Berthod, O., Grothe-Hammer, M., Müller-Seitz, G., Raab, J., & Sydow, J. (2017). From high-reliability organizations to high-reliability networks: The dynamics of network governance in the face of emergency. *Journal of Public Administration Research and Theory*, *27*(2), 352–371. https://doi.org/10.1093/jopart/muw050

Bianchi, M., Marzi, G., & Guerini, M. (2020). Agile, Stage-Gate and their combination: Exploring how they relate to performance in software development. *Journal of Business Research*, *110*, 538–553. https://doi.org/10.1016/j.jbusres.2018.05.003

Blagoev, B., Felten, S., & Kahn, R. (2018). The career of a catalogue: Organizational memory, materiality and the dual nature of the past at the British Museum (1970–Today). *Organization Studies*, *39*(12), 1757–1783. https://doi.org/10.1177/0170840618789189

Blagoev, B., Hernes, T., Kunisch, S., & Schultz, M. (2024). Time as a research lens: A conceptual review and research agenda. *Journal of Management*, *50*(6), 2151–2196. https://doi.org/10.1177/01492063231215032

Blagoev, B., & Schreyögg, G. (2019). Why do extreme work hours persist? Temporal uncoupling as a new way of seeing. *Academy of Management Journal*, *62*(6), 1818–1847. https://doi.org/10.5465/amj.2017.1481

Blagoev, B., & Schreyögg, G. (2025). Eigenzeit: A new lens on temporal complexity. *Academy of Management Review*, *50*(1), 93–113. https://doi.org/10.5465/amr.2022.0440

Blount, S., & Janicik, G. A. (2002). Getting and staying in-pace: The "in-synch" preference and its implications for work groups. In H. Sondak (Ed.), *Toward phenomenology of groups and group membership* (pp. 235–266). Elsevier Science.

Bluedorn, A. C. (2002). *The human organization of time*. Stanford University Press.

Bluedorn, A. C., & Denhardt, R. B. (1988). Time and organizations. *Journal of Management*, *14*(2), 299–320. https://doi.org/https://doi.org/10.1177/014920638801400209

Bluedorn, A. C., & Ferris, S. P. (2004). Temporal depth, age, and organizational performance. In C. Fuchs & A. L. Kalleberg (Eds.), *Fighting for time: Shifting boundaries of work and social life* (pp. 113–149). Sage.

Bluedorn, A. C., & Jaussi, K. S. (2008). Leaders, followers, and time. *Leadership Quarterly*, *19*(6), 654–668. https://doi.org/10.1016/j.leaqua.2008.09.006

Bluedorn, A. C., & Martin, G. (2008). The time frames of entrepreneurs. *Journal of Business Venturing*, *23*(1), 1–20. https://doi.org/10.1016/j.jbusvent.2006.05.005

Bluedorn, A. C., & Standifer, R. L. (2004). Groups, boundary spanning, and the temporal imagination. In S. Blount (Ed.), *Research on managing groups and teams* (Vol. 6, pp. 159–182). Emerald. https://doi.org/10.1016/S1534-0856(03)06008-0

Bluedorn, A. C., & Standifer, R. L. (2006). Time and the temporal imagination. *Academy of Management Learning & Education*, *5*(2), 196–206. https://doi.org/10.5465/amle.2006.21253784

Boehm, B., & Turner, R. (2004). *Balancing agility and discipline: A guide for the perplexed*. Addison-Wesley.

Boyd, J. L., & Bresser, R. K. F. (2008). Performance implications of delayed competitive responses: Evidence from the U.S. retail industry. *Strategic Management Journal*, *29*(10), 1077–1096. https://doi.org/10.1002/smj.710

Brown, S. L., & Eisenhardt, K. M. (1997). The art of continuous change: Linking complexity theory and time-paced evolution in relentlessly shifting organizations. *Administrative Science Quarterly*, *42*(1), 1–34. https://doi.org/10.2307/2393807

Brühl, V. (2022). Agile methods in the German banking sector: Some evidence on expectations, experiences and success factors. *Journal of Business Economics*, *92*(8), 1337–1372. https://doi.org/10.1007/s11573-022-01102-y

Burnam, M. A., Pennebaker, J. W., & Glass, D. C. (1975). Time consciousness, achievement striving, and the Type A coronary-prone behavior pattern. *Journal of Abnormal Psychology*, *84*(1), 76–79. https://doi.org/10.1037/h0076259

Burns, T., & Stalker, G. M. (1961). *The management of innocation*. Tavistock.

Burt, R. S. (1992). *Structural holes: The social structure of competiton*. Harvard University Press.

Campanelli, A. S., & Parreiras, F. S. (2015). Agile methods tailoring – A systematic literature review. *Journal of Systems and Software*, *110*, 85–100. https://doi.org/10.1016/j.jss.2015.08.035

Carlile, P. R. (2002). A pragmatic view of knowledge and boundaries: Boundary objects in new product development. *Organization Science*, *13*(4), 442–455. https://doi.org/10.1287/orsc.13.4.442.2953

Carroll, N., Conboy, K., & Wang, X. (2023). From transformation to normalisation: An exploratory study of a large-scale agile transformation. *Journal of Information Technology*, *38*(3), 267–303. https://doi.org/10.1177/02683962231164428

Cegarra-Navarro, J. G., Soto-Acosta, P., & Wensley, A. K. P. (2016). Structured knowledge processes and firm performance: The role of organizational agility. *Journal of Business Research*, *69*(5), 1544–1549. https://doi.org/10.1016/j.jbusres.2015.10.014

Ceschi, M., Sillitti, A., Succi, G., & De Panfilis, S. (2005). Project management in plan-based and agile companies. *IEEE Software*, *22*(3), 21–27. https://doi.org/10.1109/MS.2005.75

Chandler, A. D. (1962). *Strategy and structure: Chapters in history of the industrial enterprise.* MIT Press.

Chang, A., Hatcher, C., & Kim, J. (2013). Temporal boundary objects in megaprojects: Mapping the system with the Integrated Master Schedule. *International Journal of Project Management*, *31*(3), 323–332. https://doi.org/10.1016/j.ijproman.2012.08.007

Chauvin, J., Choudhury, P., & Fang, T. P. (2024). Working around the clock: Temporal distance, intrafirm communication, and time shifting of the employee workday. *Organization Science*. https://doi.org/10.1287/orsc.2023.17558

Chen, J., Miller, D., & Chen, M. J. (2021). Top management team time horizon blending and organizational ambidexterity. *Strategic Organization*, *19*(2), 183–206. https://doi.org/10.1177/1476127019883669

Chen, M. J., & Miller, D. (2011). The relational perspective as a business mindset: Managerial implications for East and West. *Academy of Management Perspectives*, *25*(3), 6–18. https://doi.org/23045047

Chick, V., & Dow, S. C. (1996). Regulation and differences in financial institutions. *Journal of Economic Issues*, *30*(2), 517–523. https://doi.org/10.1080/00213624.1996.11505815

Clark, P. (1990). Chronological codes and organizational analysis. In J. Hassard & D. Pym (Eds.), *The theory and philosophy of organizations: Critical issues and new perspectives* (pp. 137–163). Routledge.

Cloutier, C., & Langley, A. (2020). What makes a process theoretical contribution? *Organization Theory*, *1*(1). https://doi.org/10.1177/2631787720902473

Cohn, M. (2004). *User stories applied: For agile software development*. Addison-Wesley.

Conboy, K. (2009). Agility from first principles: Reconstructing the concept of agility in information systems development. *Information Systems Research*, *20*(3), 329–354. https://doi.org/10.1287/isre.1090.0236

Conboy, K., & Fitzgerald, B. (2010). Method and developer characteristics for effective agile method tailoring. *ACM Transactions on Software Engineering and Methodology*, *20*(1), 1–30. https://doi.org/10.1145/1767751.1767753

Connelly, B. L., Tihanyi, L., Certo, S. T., & Hitt, M. A. (2010). Marching to the beat of different drummers: The influence of institutional owners on competitive actions. *Academy of Management Journal*, *53*(4), 723–742. https://doi.org/10.5465/amj.2010.52814589

Conte, J. M., Dew, A. F., Schwenneker, H. H., & Romano, D. M. (2001). Incremental validity of time urgency and other type A subcomponents in predicting behavioral and health criteria. *Journal of Applied Social Psychology*, *31*(8), 1727–1748. https://doi.org/10.1111/j.1559-1816.2001.tb02748.x

Conte, J. M., Landy, F. J., & Mathieu, J. E. (1995). Time urgency: Conceptual and construct development. *Journal of Applied Psychology*, *80*(1), 178–185. https://doi.org/10.1037/0021-9010.80.1.178

Conte, J. M., Mathieu, J. E., & Landy, F. J. (1998). The nomological and predictive validity of time urgency. *Journal of Organizational Behavior*, *19*(1), 1–13. https://doi.org/10.1002/(SICI)1099-1379(199801)19:1<1::AID-JOB815>3.0.CO;2-E

Cooper, R. G. (2014). What's next?: After Stage-Gate. *Research-Technology Management*, *57*(1), 20–31. https://doi.org/10.5437/08956308X5606963

Cooper, R. G. (2017). Idea-to-launch gating systems: Better, faster, and more agile. *Research-Technology Management*, *60*(1), 48–52. https://doi.org/10.1080/08956308.2017.1255057

Cooper, R. G., & Sommer, A. F. (2016). The agile–stage-gate hybrid model: A promising new approach and a new research opportunity. *Journal of Product Innovation Management*, *33*(5), 513–526. https://doi.org/10.1111/jpim.12314

Cooper, R. G., & Sommer, A. F. (2018). Agile-stage-gate for manufacturers. *Research-Technology Management*, *61*(2), 17–26. https://doi.org/10.1080/08956308.2018.1421380

Cram, W. A. (2019). Agile development in practice: Lessons from the trenches. *Information Systems Management*, *36*(1), 2–14. https://doi.org/10.1080/10580530.2018.1553645

Cram, W. A., & Newell, S. (2016). Mindful revolution or mindless trend? Examining agile development as a management fashion. *European Journal of Information Systems*, *25*(2), 154–169. https://doi.org/10.1057/ejis.2015.13

Crawford, B., Coraiola, D. M., & Dacin, M. T. (2022). Painful memories as mnemonic resources: Grand Canyon Dories and the protection of place. *Strategic Organization*, *20*(1), 51–79. https://doi.org/10.1177/1476127020981353

Crossan, M., Cunha, M. P. E., Very, D., & Cunha, J. (2005). Time and organizational improvisation. *Academy of Management Review*, *30*(1), 129–145. https://doi.org/10.5465/amr.2005.15281441

Cuganesan, S. (2022). Investigating how the clock–event time dialectic shapes the doing of time in organizational change. *Organization Studies*, *43*(5), 749–771. https://doi.org/10.1177/01708406211006252

Cummings, J. N., Espinosa, J. A., & Pickering, C. K. (2009). Crossing spatial and temporal boundaries in globally distributed projects: A relational model of coordination delay. *Information Systems Research*, *20*(3), 420–439. https://doi.org/10.1287/isre.1090.0239

Cunha, M. P. E. (2004). Organizational time: A dialectical view. *Organization*, *11*(2), 271–296. https://doi.org/10.1177/1350508404041616

Cunha, M. P. E. (2009). The organizing of rhythm, the rhythms of organizing. In R. A. Roe, M. J. Waller, & S. R. Clegg (Eds.), *Time in Organizational Research* (pp. 220–237). Routledge.

Cunliffe, A. L., Luhman, J. T., & Boje, D. M. (2004). Narrative temporality: Implications for organizational research. *Organization Studies*, *25*(2), 261–286. https://doi.org/10.1177/0170840604040038

Cyert, R., & March, J. G. (1963). *A behavioral theory of the firm*. Blackwell.

Dahlgren, J., & Söderlund, J. (2001). Managing inter-firm industrial projects-on pacing and matching hierarchies. *International Business Review*, *10*, 305–322.

Danner-Schröder, A. (2020). Focusing on and backgrounding events simultaneously: The past–present–future relationship of the Great East Japan earthquake. *Journal of Management Inquiry*, *29*(1), 92–110. https://doi.org/10.1177/1056492618774848

Das, T. K. (1987). Strategic planning and individual temporal orientation. *Strategic Management Journal*, *8*(2), 203–209. https://doi.org/10.1002/smj.4250080211

Das, T. K., & Teng, B. S. (2001). Strategic risk behaviour and its temporalities: Between risk propensity and decision context. *Journal of Management Studies*, *38*(4), 515–534. https://doi.org/10.1111/1467-6486.00247

Davis, J. P., & Eisenhardt, K. M. (2011). Rotating leadership and collaborative innovation. *Administrative Science Quarterly*, *56*(2), 159–201. https://doi.org/10.1177/0001839211428131

Deken, F., Carlile, P. R., Berends, H., & Lauche, K. (2016). Generating novelty through interdependent routines: A process model of routine work. *Organization Science*, *27*(3), 659–677. https://doi.org/10.1287/orsc.2016.1051

Desai, V., & Madsen, P. (2022). Take your time? How activity timing affects organizational learning and performance outcomes. *Organization Science*, *33*(5), 1707–1723. https://doi.org/10.1287/ORSC.2021.1490

DesJardine, M. R., & Bansal, P. (2019). One step forward, two steps back: How negative external evaluations can shorten organizational time horizons. *Organization Science*, *30*(4), 761–780. https://doi.org/10.1287/orsc.2018.1259

Dibrell, C., Fairclough, S., & Davis, P. S. (2015). The impact of external and internal entrainment on firm innovativeness: A test of moderation. *Journal of Business Research*, *68*(1), 19–26. https://doi.org/10.1016/j.jbusres.2014.05.012

Dikert, K., Paasivaara, M., & Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations: A systematic literature review. *Journal of Systems and Software*, *119*, 87–108. https://doi.org/10.1016/j.jss.2016.06.013

Dille, T., Hernes, T., & Vaagaasar, A. L. (2023). Stuck in temporal translation? Challenges of discrepant temporal structures in interorganizational project collaboration. *Organization Studies*, *44*(6), 867–888. https://doi.org/10.1177/01708406221137841

Dille, T., & Söderlund, J. (2011). Managing inter-institutional projects: The significance of isochronism, timing norms and temporal misfits. *International Journal of Project Management*, *29*(4), 480–490. https://doi.org/10.1016/j.ijproman.2011.02.007

Dille, T., & Söderlund, J. (2013). Managing temporal misfits in institutional environments: A study of critical incidents in a complex public project. *International Journal of Managing Projects in Business*, *6*(3), 552–575. https://doi.org/10.1108/IJMPB-03-2012-0006

Dille, T., Söderlund, J., & Clegg, S. (2018). Temporal conditioning and the dynamics of inter-institutional projects. *International Journal of Project Management*, *36*(5), 673–686. https://doi.org/10.1016/j.ijproman.2018.03.007

Dougherty, D., Bertels, H., Chung, K., Dunne, D. D., & Kraemer, J. (2013). Whose time is it? Understanding clock-time pacing and event-time pacing in complex innovations. *Management and Organization Review*, *9*(2), 233–263. https://doi.org/10.1111/more.12017

Drury-Grogan, M. L. (2014). Performance on agile teams: Relating iteration objectives and critical decisions to project management success factors. *Information and Software Technology*, *56*(5), 506–515. https://doi.org/10.1016/j.infsof.2013.11.003

Durkheim, E. (1965). *The elementary forms of the religious life*. Free Press.

Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*, *50*(9–10), 833–859. https://doi.org/10.1016/j.infsof.2008.01.006

Eddleston, K. A., Banalieva, E. R., & Verbeke, A. (2020). The bribery paradox in transition economies and the enactment of 'new normal' business environments. *Journal of Management Studies*, *57*(3), 597–625. https://doi.org/10.1111/joms.12551

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of Management Review*, *14*(4), 532–550. https://doi.org/10.5465/amr.1989.4308385

Eisenhardt, K. M. (1991). Better stories and better constructs: The case for rigor and comparative logic. *Academy of Management Review*, *16*(3), 620–627. https://www.jstor.org/stable/258921

Eisenhardt, K. M. (2021). What is the Eisenhardt method, really? *Strategic Organization*, *19*(1), 147–160. https://doi.org/10.1177/1476127020982866

Eisenhardt, K. M., & Brown, S. L. (1998). Time pacing: Competing in markets that won't stand still. *Harvard Business Review*, *76*(2), 59–70.

Eisenhardt, K. M., & Graebner, M. E. (2007). Theory building from cases: Opportunities and challenges. *Academy of Management Journal*, *50*(1), 25–32. https://doi.org/10.5465/amj.2007.24160888

Eisenhardt, K. M., Kahwajy, J. L., & Bourgeois, L. J. (1997). Conflict and strategic choice: How top management teams disagree. *California Management Review*, *39*(2), 42–62.

Elfenbein, D. W., & Knott, A. M. (2015). Time to exit: Rational, behavioral, and organizational delays. *Strategic Management Journal*, *36*(7), 957–975. https://doi.org/10.1002/smj.2262

Emirbayer, M., & Mische, A. (1998). What is agency? *American Journal of Sociology*, *103*(4), 962–1023. https://doi.org/10.1086/231294

Espinosa, J. A., & Carmel, E. (2003). The impact of time separation on coordination in global software teams: A conceptual foundation. *Software Process Improvement and Practice*, *8*(4), 249–266. https://doi.org/10.1002/spip.185

Faraj, S., & Xiao, Y. (2006). Coordination in fast-response organizations. *Management Science*, *52*(8), 1155–1169. https://doi.org/10.1287/mnsc.1060.0526

Fayard, A. L. (2024). Making time for social innovation: How to interweave clock time and event time in open social innovation to nurture idea generation and social impact. *Organization Science*, *35*(3), 1131–1156. https://doi.org/10.1287/orsc.2020.0832

Feldman, M. S. (2021). Practice theory and routine dynamics. In M. S. Feldman, L. D'Adderio, B. T. Pentland, K. Dittrich, C. Rerup, & D. Seidl (Eds.), *Cambridge Handbook of Routine Dynamics* (pp. 21–36). Cambridge University Press.

Feldman, M. S., & Orlikowski, W. J. (2011). Theorizing practice and practicing theory. *Organization Science*, *22*(5), 1240–1253. https://doi.org/10.1287/orsc.1100.0612

Feldman, M. S., Pentland, B. T., D'Adderio, L., Dittrich, K., Rerup, C., & Seidl, D. (2021). What is routine dynamics? In M. S. Feldman, L. D'Adderio, B. T. Pentland, K. Dittrich, C. Rerup, & D. Seidl (Eds.), *Cambridge Handbook of Routine Dynamics* (pp. 1–18). Cambridge University Press.

Feldman, M. S., Worline, M., Baker, N., & Lowerson Bredow, V. (2022). Continuity as patterning: A process perspective on continuity. *Strategic Organization*, *20*(1), 80–109. https://doi.org/10.1177/14761270211046878

Fernandez, D. J., & Fernandez, J. D. (2008). Agile project management: Agilism versus traditional approaches. *Journal of Computer Information Systems*, *49*(2), 10–17.

Ferrier, W. J. (2001). Navigating the competitive landscape: The drivers and consequences of competitive aggressiveness. *Academy of Management Journal*, *44*(4), 858–877. https://doi.org/10.2307/3069419

Fisher, G., Kotha, S., & Lahiri, A. (2016). Changing with the times: An integrated view of identity, legitimacy, and new venture life cycles. *Academy of Management Review*, *41*(3), 383–409. https://doi.org/10.5465/amr.2013.0496

Fitzgerald, B., Hartnett, G., & Conboy, K. (2006). Customising agile methods to software practices at Intel Shannon. *European Journal of Information Systems*, *15*(2), 200–213. https://doi.org/10.1057/palgrave.ejis.3000605

Flammer, C., & Bansal, P. (2017). Does a long-term orientation create value? Evidence from a regression discontinuity. *Strategic Management Journal*, *38*(9), 1827–1847. https://doi.org/10.1002/smj.2629

Flier, B., van den Bosch, F. A. J., & Volberda, H. W. (2003). Co-evolution in strategic renewal behaviour of British, Dutch and French financial incumbents: Interaction of environmental selection, institutional effects and managerial intentionality. *Journal of Management Studies*, *40*(8), 2163–2187. https://doi.org/10.1046/j.1467-6486.2003.00416.x

Flyvbjerg, B. (2006). Five misunderstandings about case-study research. *Qualitative Inquiry*, *12*(2), 219–245. https://doi.org/10.1177/1077800405284363

Frankenhaeuser, M., Lundberg, U., & Forsman, L. (1980). Dissociation between sympathetic-adrenal and pituitary-adrenal responses to an achievement situation characterized by high controllability: Comparison between Type A and Type B males and females. *Biological Psychology*, *10*, 79–91. https://doi.org/10.1016/0301-0511(80)90029-0

Friedman, M., & Rosenman, R. H. (1974). *Type A behavior and your heart*. Knopf.

Galbraith, J. (1973). *Designing complex organizations*. Addison-Wesley.

Garud, R., Gehman, J., & Kumaraswamy, A. (2011). Complexity arrangements for sustained Innovation: Lessons from 3M Corporation. *Organization Studies*, *32*(6), 737–767. https://doi.org/10.1177/0170840611410810

Garud, R., Tuertscher, P., & Van De Ven, A. H. (2013). Perspectives on innovation processes. *Academy of Management Annals*, *7*(1), 775–819. https://doi.org/10.1080/19416520.2013.791066

Garud, R., Turunen, M., & Karunakaran, A. (2025). Revisiting exploration and exploitation: Temporal structuring for innovation at work. *Organization Theory*, *6*(1). https://doi.org/10.1177/26317877251346798

Gastorf, J. W. (1980). Time urgency of the Type A behavior pattern. *Journal of Consulting and Clinical Psychology*, *48*(2), 299. https://doi.org/10.1037/0022-006X.48.2.299

Geertz, C. (1973). *The interpretation of cultures*. Basic Books.

Geiger, D., Danner-Schröder, A., & Kremser, W. (2021). Getting ahead of time - Performing temporal boundaries to coordinate routines under temporal uncertainty. *Administrative Science Quarterly*, *66*(1), 220–264. https://doi.org/10.1177/0001839220941010

Geiger, D., & Harborth, L. (2025). The time is right when you make it right: Coordinating routine clusters in sustained crises. *Organization Studies*, *46*(7), 1023–1051. https://doi.org/10.1177/01708406241305754

Gell, A. (1992). *The anthropology of time*. Cambridge University Press.

Gemino, A., Horner Reich, B., & Serrador, P. M. (2021). Agile, traditional, and hybrid approaches to project success: Is hybrid a poor second choice? *Project Management Journal*, *52*(2), 161–175. https://doi.org/10.1177/8756972820973082

George, A. L., & Bennett, A. (2005). *Case studies and theory development in the social sciences*. MIT Press.

George, J. M., & Jones, G. R. (2000). The role of time in theory and theory building. *Journal of Management*, *26*(4), 657–684. https://doi.org/10.1177/014920630002600

Gephart, R. P. (2004). Qualitative research and the Academy of Management Journal. *Academy of Management Journal*, *47*(4), 454–462. https://doi.org/10.5465/amj.2004.14438580

Gersick, C. J. G. (1989). Marking time: Predictable transitions in task groups. *Academy of Management Journal*, *32*(2), 274–309. https://doi.org/10.5465/256363

Gersick, C. J. G. (1994). Pacing strategic change: The case of a new venture. *Academy of Management Journal*, *37*(1), 9–45. https://doi.org/10.5465/256768

Gevers, J. M. P., van Eerde, W., & Rutte, C. G. (2009). Team self-regulation and meeting deadlines in project teams: Antecedents and effects of temporal consensus. *European Journal of Work and Organizational Psychology*, *18*(3), 295–321. https://doi.org/10.1080/13594320701693217

Gevers, J. M. P., Mohammed, S., & Baytalskaya, N. (2015). The conceptualization and measurement of pacing styles. Applied Psychology, 64(3), 499–540. https://doi.org/10.1111/apps.12016

Ghobadi, S., & Mathiassen, L. (2016). Perceived barriers to effective knowledge sharing in agile software teams. *Information Systems Journal*, *26*(2), 95–125. https://doi.org/10.1111/isj.12053

Ghosh, S., & Wu, A. (2023). Iterative coordination and innovation: Prioritizing value over novelty. *Organization Science*, *34*(6), 2182–2206. https://doi.org/10.1287/orsc.2021.1499

Gibbert, M., & Ruigrok, W. (2010). The ‘“what”’ and ‘“how”’ of case study rigor: Three strategies based on published work. *Organizational Research Methods*, *13*(4), 710–737. https://doi.org/10.1177/1094428109351319

Gibbert, M., Ruigrok, W., & Wicki, B. (2008). What passes as a rigorous case study? *Strategic Management Journal*, *29*(13), 1465–1474. https://doi.org/10.1002/smj.722

Gibson, C. B., Waller, M. J., Carpenter, M. A., & Conte, J. M. (2007). Antecedents, consequences, and moderators of time perspective heterogeneity for knowledge management in MNO teams. *Journal of Organizational Behavior*, *28*(8), 1005–1034. https://doi.org/10.1002/job.445

Gioia, D. A., Corley, K. G., & Hamilton, A. L. (2013). Seeking qualitative rigor in inductive research: Notes on the Gioia methodology. *Organizational Research Methods*, *16*(1), 15–31. https://doi.org/10.1177/1094428112452151

Glass, D. C., Snyder, M. L., & Hollis, J. F. (1974). Time urgency and the Type A coronary-prone behavior pattern. *Journal of Applied Social Psychology*, *4*(2), 125–140. https://doi.org/10.1111/j.1559-1816.1974.tb00663.x

Golden, B. R. (1992). The past is the past - or is it? The use of retrospective accounts as indicators of past strategy. *Academy of Management Journal*, *35*(4), 848–860. https://doi.org/10.5465/256318

Graebner, M. E., Martin, J. A., & Roundy, P. T. (2012). Qualitative data: Cooking without a recipe. *Strategic Organization*, *10*(3), 276–284. https://doi.org/10.1177/1476127012452821

Granovetter, M. (1973). The strength of weak ties. *American Journal of Sociology*, *78*, 1360–1380.

Granqvist, N., & Gustafsson, R. (2016). Temporal institutional work. *Academy of Management Journal*, *59*(3), 1009–1035. https://doi.org/10.5465/amj.2013.0416

Grass, A., Backmann, J., & Hoegl, M. (2020). From empowerment dynamics to team adaptability: Exploring and conceptualizing the continuous agile team innovation process. *Journal of Product Innovation Management*, *37*(4), 324–351. https://doi.org/10.1111/jpim.12525

Gutierrez, C., Sloof, R., & Crilly, D. (2024). Time is not money! Temporal preferences for time investments and entry into entrepreneurship. *Organization Science*, *35*(2), 623–643. https://doi.org/10.1287/orsc.2023.1681

Halevy, N., Halali, E., & Zlatev, J. J. (2019). Brokerage and brokering: An integrative review and organizing framework for third party influence. *Academy of Management Annals*, *13*(1), 215–239. https://doi.org/10.5465/annals.2017.0024

Hall, E. T. (1983). *The dance of life*. Anchor Books.

Hampel, C. E., & Dalpiaz, E. (2023). Confronting the contested past: Sensemaking and rhethorical history in the reconstructions of organizational identity. *Academy of Management Journal*, *66*(6), 1711–1740. https://doi.org/10.5465/amj.2020.1132

Hannah, D. P., & Eisenhardt, K. M. (2018). How firms navigate cooperation and competition in nascent ecosystems. *Strategic Management Journal*, *39*(12), 3163–3192. https://doi.org/10.1002/smj.2750

Harris, M. L., Collins, R. W., & Hevner, A. R. (2009). Control of flexible software development under uncertainty. *Information Systems Research*, *20*(3), 400–419. https://doi.org/10.1287/isre.1090.0240

Hassard, J. (1991). Aspects of time in organization. *Human Relations*, *44*(2), 105–125. https://doi.org/10.1177/001872679104400

Haveman, H. A. (1992). Between a rock and a hard place: Organizational change and performance under conditions of fundamental environmental transformation. *Administrative Science Quarterly*, *37*(1), 48–75. https://doi.org/10.2307/2393533

Hawk, A., & Pacheco-de-Almeida, G. (2018). Time compression (dis)economies: An empirical analysis. *Strategic Management Journal*, *39*(9), 2489–2516. https://doi.org/10.1002/smj.2915

Hawk, A., Pacheco-De-Almeida, G., & Yeung, B. (2013). Fast-mover advantages: Speed capabilities and entry into the emerging submarket of Atlantic basin LNG. *Strategic Management Journal*, *34*(13), 1531–1550. https://doi.org/10.1002/smj.2085

Henry, L. A., & Möllering, G. (2023). Sluggish, but innovative? Orchestrating collaboration in multi-stakeholder networks despite low commitment. *Innovation*, *25*(3), 282–304. https://doi.org/10.1080/14479338.2022.2029707

Hernes, T. (2022). *Organization and time*. Oxford University Press.

Hernes, T., & Schultz, M. (2020). Translating the distant into the present: How actors address distant past and future events through situated activity. *Organization Theory*, *1*(1), 1–20. https://doi.org/10.1177/2631787719900999

Herriott, R. E., & Firestone, W. A. (1983). Multisite qualitative policy research: Optimizing description and generalizability. *Educational Researcher*, *12*(2), 14–19. https://doi.org/10.3102/0013189X012002014

Hilbolling, S., Deken, F., Berends, H., & Tuertscher, P. (2022). Process-based temporal coordination in multiparty collaboration for societal challenges. *Strategic Organization*, *20*(1), 135–163. https://doi.org/10.1177/1476127021992705

Hilkamo, O., Barbe, A. S., Granqvist, N., & Geurts, A. (2021). Temporal work by consultants in nascent market categories: constructing a market for knowledge in quantum computing. *Technology Analysis and Strategic Management*, *33*(11), 1303–1316. https://doi.org/10.1080/09537325.2021.1931098

Hobbs, B., & Petit, Y. (2017). Agile methods on large projects in large organizations. *Project Management Journal*, *48*(3), 3–19. https://doi.org/10.1177/875697281704800301

Holt, R., & Johnsen, R. (2019). Time and organization studies. *Organization Studies*, *40*(10), 1557–1572. https://doi.org/10.1177/0170840619844292

Hopp, C., & Greene, F. J. (2018). In pursuit of time: Business plan sequencing, duration and intraentrainment effects on new venture viability. *Journal of Management Studies*, *55*(2), 320–351. https://doi.org/10.1111/joms.12251

Huber, G. P., & Power, D. J. (1985). Retrospective reports of strategic-level managers: Guidelines for increasing their accuracy. *Strategic Management Journal*, *6*(2), 171–180. https://doi.org/10.1002/smj.4250060206

Huber, T. L., Winkler, M. A. E., Dibbern, J., & Brown, C. V. (2020). The use of prototypes to bridge knowledge boundaries in agile software development. *Information Systems Journal*, *30*(2), 270–294. https://doi.org/10.1111/isj.12261

Hui, A. (2017). Variation and the intersection of practices. In A. Hui, T. Schatzki, & E. Shove (Eds.), *The nexus of practices: Connections, constellations, practitioners* (pp. 52–67). Routledge.

Huy, Q. N. (2001). Time, temporal capability, and planned change. *Academy of Management Review*, *26*(4), 601–623. https://doi.org/10.5465/amr.2001.5393897

Iivari, J., & Iivari, N. (2011). The relationship between organizational culture and the deployment of agile methods. *Information and Software Technology*, *53*(5), 509–520. https://doi.org/10.1016/j.infsof.2010.10.008

Im, H. G., Yates, J., & Orlikowski, W. (2005). Temporal coordination through communication: Using genres in a virtual start-up organization. *Information Technology and People*, *18*(2), 89–119. https://doi.org/10.1108/09593840510601496

Janicik, G. A., & Bartel, C. A. (2003). Talking about time: Effects of temporal planning and time awareness norms on group coordination and performance. *Group Dynamics: Theory, Research, and Practice*, *7*(2), 122–134. https://doi.org/10.1037/1089-2699.7.2.122

Jansen, K. J., & Kristof-Brown, A. L. (2005). Marching to the beat of a different drummer: Examining the impact of pacing congruence. *Organizational Behavior and Human Decision Processes*, *97*(2), 93–105. https://doi.org/10.1016/j.obhdp.2005.03.005

Jaques, E. (1982). *The form of time*. Heinemann.

Jarvenpaa, S. L., & Välikangas, L. (2022). Toward temporally complex collaboration in an interorganizational research network. *Strategic Organization*, *20*(1), 110–134. https://doi.org/10.1177/14761270211042934

Jarzabkowski, P. A., Kaplan, S., Seidl, D., & Whittington, R. (2016). On the risk of studying practices in isolation: Linking what, who, and how in strategy research. *Strategic Organization*, *14*(3), 248–259. https://doi.org/10.1177/1476127015604125

Jarzabkowski, P. A., Lê, J. K., & Feldman, M. S. (2012). Toward a theory of coordinating: Creating coordinating mechanisms in practice. *Organization Science*, *23*(4), 907–927. https://doi.org/10.1287/orsc.1110.0693

Junker, T. L., Bakker, A. B., Derks, D., & Molenaar, D. (2023). Agile work practices: Measurement and mechanisms. *European Journal of Work and Organizational Psychology*, *32*(1), 1–22. https://doi.org/10.1080/1359432X.2022.2096439

Kalff, Y. (2022). Managing projects as a mode of temporal ordering: Performative organising of time and temporality in projects. *International Journal of Managing Projects in Business*, *15*(1), 82–101. https://doi.org/10.1108/IJMPB-05-2021-0132

Kalus, G., & Kuhrmann, M. (2013). Criteria for software process tailoring: A systematic review. *Proceedings of the 2013 International Conference on Software and System Process*, 171–180. https://doi.org/10.1145/2486046.2486078

Kaplan, S., & Orlikowski, W. J. (2013). Temporal work in strategy making. *Organization Science*, *24*(4), 965–995. https://doi.org/10.1287/orsc.1120.0792

Karaduman, C. (2021). *Synchronization beyond clocks: The effects of subjective time on the coordination of temporary organizations* [Dissertation]. Tilburg University.

Karau, S. J., & Kelly, J. R. (2004). Time pressure and team performance: An attentional focus integration. In S. Blount (Ed.), *Time In Groups* (pp. 185–212). Emerald.

Karlstrom, D., & Runeson, P. (2005). Combining agile methods with stage-gate project management. *IEEE Software*, *22*(3), 43–49. https://doi.org/10.1109/MS.2005.59

Katila, S., Kuismin, A., & Valtonen, A. (2020). Becoming upbeat: Learning the affecto-rhythmic order of organizational practices. *Human Relations*, *73*(9), 1308–1330. https://doi.org/10.1177/0018726719867753

Kellogg, K. C., Orlikowski, W. J., & Yates, J. (2006). Life in the trading zone: Structuring coordination across boundaries in postbureaucratic organizations. *Organization Science*, *17*(1), 22–44. https://doi.org/10.1287/orsc.1050.0157

Kent, D., & Granqvist, N. (2025). Chasing Storms: Temporal work to goster group engagement under uncertainty. *Academy of Management Journal*, *68*(2). https://doi.org/10.5465/amj.2021.1316

Kerfoot, D., & Knights, D. (1993). Management, masculinity and manipulation: From paternalism to corporate strategy in financial services in Britain. *Journal of Management Studies*, *30*(4), 659–677. https://doi.org/10.1111/j.1467-6486.1993.tb00320.x

Khanagha, S., Volberda, H. W., Alexiou, A., & Annosi, M. C. (2022). Mitigating the dark side of agile teams: Peer pressure, leaders' control, and the innovative output of agile teams. *Journal of Product Innovation Management*, *39*(3), 334–350. https://doi.org/10.1111/jpim.12589

Khavul, S., Pérez-Nordtvedt, L., & Wood, E. (2010). Organizational entrainment and international new ventures from emerging markets. *Journal of Business Venturing*, *25*(1), 104–119. https://doi.org/10.1016/j.jbusvent.2009.01.008

Klarner, P., Ambos, T., & Birkinshaw, J. (2025). How to implement bottom-up organizing: Lessons from agile piloting and scaling. *California Management Review*, *67*(4), 75–94. https://doi.org/10.1177/00081256241304461

Klarner, P., & Raisch, S. (2013). Move to the beat - Rhythms of change and firm performance. *Academy of Management Journal*, *56*(1), 160–184. https://doi.org/10.5465/amj.2010.0767

Klein, K. J., & Kozlowski, S. W. J. (2000). From micro to meso: Critical steps in conceptualizing and conducting multilevel research. *Organizational Research Methods*, *3*(3), 211–236. https://doi.org/10.1177/109442810033001

Klein, K. J., Ziegert, J. C., Knight, A. P., & Xiao, Y. (2007). Dynamic delegation: Shared, hierarchical, and deindividualized leadership in extreme action teams. *Administrative Science Quarterly*, *51*(4), 590–621. https://doi.org/10.2189/asqu.51.4.590

Knights, D., & Morgan, G. (1995). Strategy under the microscope: Strategic management and IT in financial services. *Journal of Management Studies*, *32*(2), 191–214. https://doi.org/10.1111/j.1467-6486.1995.tb00340.x

Kouamé, S., & Langley, A. (2018). Relating microprocesses to macro-outcomes in qualitative strategy process and practice research. *Strategic Management Journal*, *39*(3), 559–581. https://doi.org/10.1002/smj.2726

Kremser, W., & Blagoev, B. (2021). The dynamics of prioritizing: How actors temporally pattern complex role–routine ecologies. *Administrative Science Quarterly*, *66*(2), 339–379. https://doi.org/10.1177/0001839220948483

Kremser, W., & Xiao, Y. (2021). Self-managed forms of organizing and routine dynamics. In M. S. Feldman, L. D'Adderio, B. T. Pentland, K. Dittrich, C. Rerup, & D. Seidl (Eds.), *Cambridge Handbook of Routine Dynamics* (pp. 421–432). Cambridge University Press.

Kude, T., Mithas, S., Schmidt, C. T., & Heinzl, A. (2019). How pair programming influences team performance: The role of backup behavior, ahared mental models, and task novelty. *Information Systems Research*, *30*(4), 1145–1163. https://doi.org/10.1287/isre.2019.0856

Kumar, N., Stern, L. W., & Anderson, J. C. (1993). Conducting interorganizational research using key informants. *Academy of Management Journal*, *36*(6), 1633–1651. https://doi.org/10.5465/256824

Kunisch, S., Bartunek, J. M., Mueller, J., & Huy, Q. N. (2017). Time in strategic change research. *Academy of Management Annals*, *11*(2), 1005–1064. https://doi.org/10.5465/annals.2015.0133

Kunisch, S., Blagoev, B., & Bartunek, J. M. (2021). Complex times, complex time: The pandemic, time-based theorizing and temporal research in management and organization studies. *Journal of Management Studies*, *58*(5), 1411–1415. https://doi.org/10.1111/joms.12703

Kunzl, F., & Messner, M. (2023). Temporal structuring as self-discipline: Managing time in the budgeting process. *Organization Studies*, *44*(9), 1439–1464. https://doi.org/10.1177/01708406221137

Kusters, R. J., Van De Leur, Y., Rutten, W. G. M. M., & Trienekens, J. J. M. (2017). When agile meets waterfall: Investigating risks & problems on the interface between agile & traditional software development in a hybrid development organization. *Proceedings of the 19th International Conference on Enterprise Information Systems*, *2*, 271–278. https://doi.org/10.5220/0006292502710278

Labianca, G., Moon, H., & Watt, I. (2005). When is an hour not 60 minutes? Deadlines, temporal schemata, and individual and task group performance. *Academy of Management Journal*, *48*(4), 677–694. https://doi.org/10.5465/AMJ.2005.17843945

Landy, F. J., Rastegary, H., Thayer, J., & Colvin, C. (1991). Time urgency: The construct and its measurement. *Journal of Applied Psychology*, *76*(5), 644–657. https://doi.org/10.1037/0021-9010.76.5.644

Langley, A. (1999). Strategies for theorizing from process data. *Academy of Management Review*, *24*(4), 691–710. https://doi.org/10.2307/259349

Langley, A., & McGivern, G. (2024). Temporal structuring and project behavior. In *Cambridge Handbook of Project Behavior*. Cambridge University Press.

Lappi, T., Karvonen, T., Lwakatare, L. E., Aaltonen, K., & Kuvaja, P. (2018). Toward an improved understanding of agile project governance. *Project Management Journal*, *49*(6), 39–63. https://doi.org/10.1177/8756972818803482

Larman, C., & Vodde, B. (2008). *Scaling lean & agile development thinking and organizational tools for Large-Scale Scrum*. Addison-Wesley.

Lauer, R. H. (1981). *Temporal man: The meaning and uses of social time*. Praeger.

Laverty, K. J. (1996). Economic "short-termism": The debate, the unresolved issues, and the implications for management practice and research. *Academy of Management Review*, *21*(3), 825–860. https://doi.org/10.5465/amr.1996.9702100316

Lawrence, B. S. (1984). Age grading: The implicit organizational timetable. *Journal of Occupational Behaviour*, *5*(1), 23–35. https://doi.org/10.1002/job.4030050104

Lawrence, P. R., & Lorsch, J. W. (1967). *Organization and environment*. Graduate School of Business Administration.

Lecuona, J. R., & Reitzig, M. (2014). Knowledge worth having in "excess": The value of tacit and firm-specific human resource slack. *Strategic Management Journal*, *35*(7), 954–973. https://doi.org/10.1002/smj.2143

Lee, B. H., Hiatt, S. R., & Lounsbury, M. (2017). Market mediators and the trade-offs of legitimacy-seeking behaviors in a nascent category. *Organization Science*, *28*(3), 447–470. https://doi.org/10.1287/orsc.2017.1126

Lee, G., & Xia, W. (2010). Toward agile: An integrated analysis of quantitative and qualitative field data on software development agility. *MIS Quarterly*, *34*(1), 87. https://doi.org/10.2307/20721416

Lee, H., & Liebenau, J. (1999). Time in organizational studies: Towards a new research direction. *Organization Studies*, *20*(6), 1035–1058. https://doi.org/https://doi.org/10.1177/0170840699206006

Lee, S., & Geum, Y. (2021). How to determine a minimum viable product in app-based lean start-ups: Kano-based approach. *Total Quality Management & Business Excellence*, *32*(15–16), 1751–1767. https://doi.org/10.1080/14783363.2020.1770588

Leffingwell, D., Yakyma, A., Knaster, R., Jemilo, D., & Oren, I. (2016). *SAFe Reference Guide 4.0: Scaled Agile Framework for lean software and sytems engineering* (1st ed.). Addison-Wesley.

Leonard-Barton, D. (1990). A dual methodology for case studies: Synergistic use of a longitudinal single site with replicated multiple sites. *Organization Science*, *1*(3), 248–266. https://doi.org/10.1287/orsc.1.3.248

Leroy, S., Shipp, A. J., Blount, S., & Licht, J. G. (2015). Synchrony preference: Why some people go with the flow and some son't. *Personnel Psychology*, *68*(4), 759–809. https://doi.org/10.1111/peps.12093

Levina, N., & Vaast, E. (2005). The emergence of boundary spanning competence in practice: Implications for implementation and use of information systems. *MIS Quarterly*, *29*(2), 335. https://doi.org/10.2307/25148682

Levine, R. V. (1988). The pace of life across cultures. In J. E. McGrath (Ed.), *The social psychology of time: New perspectives* (pp. 39–60). Sage.

Levinthal, D. A., & March, J. G. (1993). The myopia of learning. *Strategic Management Journal*, *14*(S2), 95–112. https://doi.org/10.1002/smj.4250141009

Leybourne, S. A. (2009). Improvisation and agile project management: a comparative consideration. *International Journal of Managing Projects in Business*, *2*(4), 519–535. https://doi.org/10.1108/17538370910991124

Lifshitz-Assaf, H., Lebovitz, S., & Zalmanson, L. (2021). Minimal and adaptive coordination: How hackathons' projects accelerate innovation without killing it. *Academy of Management Journal*, *64*(3), 684–715. https://doi.org/10.5465/amj.2017.0712

Lin, Y., Shi, W., Prescott, J. E., & Yang, H. (2019). In the eye of the beholder: Top managers' long-term orientation, industry context, and decision-making processes. *Journal of Management*, *45*(8), 3114–3145. https://doi.org/10.1177/0149206318777589

Lindkvist, L., Söderlund, J., & Tell, F. (1998). Managing product development projects: On the significance of fountains and deadlines. *Organization Studies*, *19*(6), 931–951. https://doi.org/10.1177/017084069801900602

Luciano, M. M., Bartels, A. L., D'Innocenzo, L., Maynard, M. T., & Mathieu, J. E. (2018). Shared team experiences and team effectiveness: Unpacking the contingent effects of entrained rhythms and task characteristics. *Academy of Management Journal*, *61*(4), 1403–1430. https://doi.org/10.5465/amj.2016.0828

Luger, J., Raisch, S., & Schimmer, M. (2018). Dynamic balancing of exploration and exploitation: The contingent benefits of ambidexterity. *Organization Science*, *29*(3), 449–470. https://doi.org/10.1287/orsc.2017.1189

Luhmann, N. (1995). *Social systems*. Stanford University Press.

Lumpkin, G. T., & Brigham, K. H. (2011). Long-term orientation and intertemporal choice in family firms. *Entrepreneurship: Theory and Practice*, *35*(6), 1149–1169. https://doi.org/10.1111/j.1540-6520.2011.00495.x

Lupu, I., & Rokka, J. (2022). "Feeling in control": Optimal busyness and the temporality of organizational controls. *Organization Science*, *33*(4), 1396–1422. https://doi.org/10.1287/orsc.2021.1486

Ly, D., Overeem, M., Brinkkemper, S., & Dalpiaz, F. (2025). The power of words in agile vs. waterfall development: Written communication in hybrid software teams. *Journal of Systems and Software*, *219*, 1–22. https://doi.org/10.1016/j.jss.2024.112243

Macan, T. H. (1994). Time Management: Test of a Process Model. *Journal of Applied Psychology*, *79*(3), 381–391. https://doi.org/10.1037/0021-9010.79.3.381

Magrelli, V., Rondi, E., De Massis, A., & Kotlar, J. (2022). Generational brokerage: An intersubjective perspective on managing temporal orientations in family firm succession. *Strategic Organization*, *20*(1), 164–199. https://doi.org/10.1177/1476127020976972

Mahringer, C. A., & Danner-Schröder, A. (2025). Autonomous, yet interdependent: Designing interfaces across routine clusters. *Academy of Management Journal*. https://doi.org/10.5465/amj.2022.0853

March, J. C., & Simon, H. A. (1958). *Organizations*. Wiley.

Marginson, D., & McAulay, L. (2008). Exploring the debate on short-termism: A theoretical and empirical analysis. *Strategic Management Journal*, *29*(3), 273–292. https://doi.org/10.1002/smj.657

Maruping, L. M., Venkatesh, V., Thatcher, S. M. B., & Patel, P. C. (2015). Folding under pressure or rising to the occasion? Perceived time pressure and the moderating role of team temporal leadership. *Academy of Management Journal*, *58*(5), 1313–1333. https://doi.org/10.5465/amj.2012.0468

Massey, A. P., Montoya-Weiss, M. M., & Hung, Y.-T. (2003). Because time matters: Temporal coordination in global virtual project teams. *Journal of Management Information Systems*, *19*(4), 129–155. https://doi.org/10.1080/07421222.2003.11045742

McGivern, G., Dopson, S., Ferlie, E., Fischer, M., Fitzgerald, L., Ledger, J., & Bennett, C. (2018). The silent politics of temporal work: A case study of a management consultancy project to redesign public health care. *Organization Studies*, *39*(8), 1007–1030. https://doi.org/10.1177/0170840617708004

McGrath, J. E. (1991). Time, interaction, and performance (TIP): A theory of groups. *Small Group Research*, *22*(2), 147–174.

McGrath, J. E., & Kelly, J. R. (1986). *Time and human interaction: Toward a social psychology of time*. The Guilford Press.

McGrath, J. E., & Rotchford, N. L. (1983). Time and behavior in organizations. *Research in Organizational Behavior*, *5*, 57–101.

McLeod, J., Zeimers, G., Robertson, J., Ordway, C., McGowan, L., & Shilbury, D. (2024). "Seven weeks is not a lot of rime": Temporal work and institutional change in Australian football. *Journal of Sport Management*, *38*(4), 257–270. https://doi.org/10.1123/jsm.2023-0278

Mell, J. N., Jang, S., & Chai, S. (2021). Bridging temporal divides: Temporal brokerage in global teams and its impact on individual performance. *Organization Science*, *32*(3), 731–751. https://doi.org/10.1287/orsc.2020.1406

Mena, S., & Parker, S. (2024). The temporal structuring of corporate sustainability. *Journal of Business Ethics*, *195*(1), 1–23. https://doi.org/10.1007/s10551-024-05648-5

Merida, A. L., & Rocha, V. (2021). It's about time: The timing of entrepreneurial experience and the career dynamics of university graduates. *Research Policy*, *50*(1). https://doi.org/10.1016/j.respol.2020.104135

Merriam, S. B. (2009). *Qualitative research: A guide to design and implementation*. Jossey-Bass.

Meyer, J. W., & Rowan, B. (1977). Institutionalized organizations: Formal structure as myth and ceremony. *American Journal of Sociology*, *83*(2), 340–363.

Miller, C. C., Cardinal, L. B., & Glick, W. H. (1997). Retrospective reports in organizational research: A reexamination of recent evidence. *Academy of Management Journal*, *40*(1), 189–204. https://doi.org/10.5465/257026

Misra, S. C., Kumar, V., & Kumar, U. (2010). Identifying some critical changes required in adopting agile practices in traditional software development projects. *International Journal of Quality and Reliability Management*, *27*(4), 451–474. https://doi.org/10.1108/02656711011035147

Mohammed, S., Alipour, K. K., Martinez, P., Livert, D., & Fitzgerald, D. (2017). Conflict in the kitchen: Temporal diversity and temporal disagreements in chef teams. *Group Dynamics*, *21*(1), 1–19. https://doi.org/10.1037/gdn0000058

Mohammed, S., & Harrison, D. A. (2013). The clocks that time us are not the same: A theory of temporal diversity, task characteristics, and performance in teams. *Organizational Behavior and Human Decision Processes*, *122*(2), 244–256. https://doi.org/10.1016/j.obhdp.2013.08.004

Mohammed, S., & Nadkarni, S. (2011). Temporal diversity and team performance: The moderating role of team temporal leadership. *Academy of Management Journal*, *54*(3), 489–508. https://doi.org/10.5465/AMJ.2011.61967991

Montoya-Weiss, M. M., Massey, A. P., & Song, M. (2001). Getting it together: Temporal coordination and conflict management in global virtual teams. *Academy of Management Journal*, *44*(6), 1251–1262. https://www.jstor.org/stable/3069399

Moore, W. E. (1963). *Man, time, and society*. Wiley.

Mosakowski, E., & Earley, P. C. (2000). A selective review of time assumptions in strategy research. *Academy of Management Review*, *25*(4), 796–812. https://doi.org/10.2307/259207

Mount, M. P., Ertug, G., Kavusan, K., George, G., & Zou, T. (2024). Reeling in the slack: An integrative review to reinstate slack as a central theoretical construct for management research. *Academy of Management Annals*, *18*(2), 473–505. https://doi.org/10.5465/annals.2023.0087

Muehlberger, U. (2007). Hierarchical forms of outsourcing and the creation of dependency. *Organization Studies*, *28*(5), 709–727. https://doi.org/10.1177/0170840607078119

Nachbagauer, A. (2022). Synchronous and diachronic timing: Insights into managing projects from disaster management and fast-response organizations. *Project Management Journal*, *53*(2), 146–160. https://doi.org/10.1177/87569728221079414

Nadkarni, S., Chen, T., & Chen, J. (2016). The clock is ticking! Executive temporal depth, industry velocity, and competitive aggressiveness. *Strategic Management Journal*, *37*(6), 1132–1153. https://doi.org/10.1002/smj.2376

Nadolska, A., & Barkema, H. G. (2007). Learning to internationalise: The pace and success of foreign acquisitions. *Journal of International Business Studies*, *38*(7), 1170–1186. https://doi.org/10.1057/palgrave.jibs.8400318

Nerur, S., Mahapatra, R. K., & Mangalaraj, G. (2005). Challenges of migrating to agile methodologies: Organizations must carefully assess their readiness before treading the path of agility. *Communications of the ACM*, *48*(5). https://doi.org/https://doi.org/10.1145/1060710.1060712

Nguyen, T., Le, C. V., Nguyen, M., Nguyen, G., Lien, T. T. H., & Nguyen, O. (2024). The organisational impact of agility: A systematic literature review. *Management Review Quarterly*. https://doi.org/10.1007/s11301-024-00446-9

Niederman, F., Lechler, T., & Petit, Y. (2018). A research agenda for extending agile practices in software development and additional task domains. *Project Management Journal*, *49*(6), 3–17. https://doi.org/10.1177/8756972818802713

Nowotny, H. (1992). Time and social theory: Towards a social theory of time. *Time & Society*, *1*(3), 421–454.

Oborn, E., & Barrett, M. (2021). Marching to different drum beats: A temporal perspective on coordinating occupational work. *Organization Science*, *32*(2), 376–406. https://doi.org/10.1287/orsc.2020.1394

Obstfeld, D. (2005). Social networks, the Tertius Iungens orientation, and involvement in innovation. *Administrative Science Quarterly*, *50*, 100–130. https://doi.org/10.2189/asqu.2005.50.1.100

Obstfeld, D., Borgatti, S. P., & Davis, J. (2014). Brokerage as a process: Decoupling third party action from social network structure. *Research in the Sociology of Organizations*, *40*, 135–159. https://doi.org/10.1108/S0733-558X(2014)0000040007

Oellgaard, M. J. (2013). The performance of a project life cycle methodology in practice. *Project Management Journal*, *44*(4), 65–83. https://doi.org/10.1002/pmj.21357

Ofori-Dankwa, J., & Julian, S. D. (2001). Complexifying organizational theory: Illustrations using time research. *Academy of Management Review*, *26*(3), 415–430. https://doi.org/10.5465/amr.2001.4845809

Okhuysen, G. A., & Bechky, B. A. (2009). Coordination in organizations: An integrative perspective. *Academy of Management Annals*, *3*(1), 463–502. https://doi.org/10.1080/19416520903047533

Omidvar, O., Hadjimichael, D., Burke, G. T., Pyrko, I., & Chia, R. (2025). Temporal patterns in management: Integrating perspectives on rhythms of work and organizing. *Academy of Management Annals*. https://doi.org/10.5465/annals.2024.0081

Opper, S., & Burt, R. S. (2021). Social network and temporal myopia. *Academy of Management Journal*, *741*(771), 64–3. https://doi.org/10.5465/amj.2019.1026

O'Riordan, N., Acton, T., Conboy, K., & Golden, W. (2013). It's about time: Investigating the temporal parameters of decision making in agile teams. In H. Linger, J. Fisher, A.

Barnden, C. Barry, M. Lang, & C. Schneider (Eds.), *Building sustainable information systems* (pp. 1–12). Springer. https://doi.org/10.1007/978-1-4614-7540-8_35

Orlikowski, W. J., & Yates, J. A. (2002). It's about time: Temporal structuring organizations. *Organization Science*, *13*(6), 684–700. https://doi.org/10.1287/orsc.13.6.684.501

Orton, J. D., & Weick, K. E. (1990). Loosely coupled systems: A reconceptualization. *Academy of Management Review*, *15*(2), 203–223. https://doi.org/10.5465/amr.1990.4308154

Otto, B. D., Schuessler, E. S., Sydow, J., & Vogelgsang, L. (2024). Finding creativity in predictability: Seizing kairos in chronos through temporal work in complex innovation processes. *Organization Science*, *35*(5), 1795–1822. https://doi.org/10.1287/orsc.2020.14743

Paasivaara, M., Behm, B., Lassenius, C., & Hallikainen, M. (2018). Large-scale agile transformation at Ericsson: A case study. *Empirical Software Engineering*, *23*(5), 2550–2596. https://doi.org/10.1007/s10664-017-9555-8

Patriotta, G., & Gruber, D. A. (2015). Newsmaking and sensemaking: Navigating temporal transitions between planned and unexpected events. *Organization Science*, *26*(6), 1574–1592. https://doi.org/10.1287/orsc.2015.1005

Patton, M. Q. (2015). *Qualitative research & evalation methods: Integrating theory and practice* (4th ed.). Sage.

Pentland, B. T., & Feldman, M. S. (2007). Narrative networks: Patterns of technology and organization. *Organization Science*, *18*(5), 781–795. https://doi.org/10.1287/orsc.1070.0283

Pentland, B. T., Kremser, W., & Goh, K. T. (2025). Path nets: Concurrence and recurrence in the dynamics of organizing. *Academy of Management Review*, *50*(1), 114–137. https://doi.org/10.5465/amr.2022.0412

Pentland, B. T., Recker, J., & Wyner, G. (2016). Conceptualizing and measuring interdepedence between organizational routines. In J. Mooney & B. Fitzgerald (Eds.), *Proceedings of the 37th International Conference on Information Systems (ICIS 2016)* (pp. 1–10). Association for Information Systems.

Pérez-Nordtvedt, L., Payne, G. T., Short, J. C., & Kedia, B. L. (2008). An entrainment-based model of temporal organizational fit, misfit, and performance. *Organization Science*, *19*(5), 785–801. https://doi.org/10.1287/orsc.1070.0330

Perlow, L. A. (1999). The time famine: Toward a sociology of work time. *Administrative Science Quarterly*, *44*(1), 57–81. https://doi.org/10.2307/2667

Perlow, L. A., Okhuysen, G. A., & Repenning, N. P. (2002). The speed trap: Exploring the relationship between decision making and temporal context. *Academy of Management Journal*, *45*(5), 931–955.

Petersen, K., & Wohlin, C. (2010). The effect of moving from a plan-driven to an incremental software development approach with agile practices. *Empirical Software Engineering*, *15*(6), 654–693. https://doi.org/10.1007/s10664-010-9136-6

Pettersen, N. (1991). What do we know about the effective project manager? *International Journal of Project Management*, *9*(2), 99–104. https://doi.org/10.1016/0263-7863(91)90068-7

Pettigrew, A. M. (1990). Longitudinal field research on change: Theory and practice. *Organization Science*, *1*(3), 267–292. https://doi.org/10.1287/orsc.1.3.267

Pettigrew, A. M. (1992). The character and significance of strategy process research. *Strategic Management Journal*, *13*(S2), 5–16. https://doi.org/10.1002/smj.4250130903

Pfeffer, J., & Salancik, G. R. (1978). *The external control of organizations: A resource dependence perspective*. Harper and Row.

Port, D., & Bui, T. (2009). Simulating mixed agile and plan-based requirements prioritization strategies: proof-of-concept and practical implications. *European Journal of Information Systems*, *18*(4), 317–331. https://doi.org/10.1057/ejis.2009.19

Price, V. A. (1982). *Type A behavior pattern: A model for research and practice*. Academic Press.

Project Management Institute. (2021). *Guide to the project management body of knowledge (PMBOK)* (7th ed.). Project Management Institute.

Putnam, L. L., Fairhurst, G. T., & Banghart, S. (2016). Contradictions, dialectics, and paradoxes in organizations: A constitutive approach. *Academy of Management Annals*, *10*(1), 65–171. https://doi.org/10.5465/19416520.2016.1162421

Raaijmakers, A. G. M., Vermeulen, P. A. M., Meeus, M. T. H., & Zietsma, C. (2015). I need time! Exploring pathways to compliance under institutional complexity. *Academy of Management Journal*, *58*(1), 85–110. https://doi.org/10.5465/amj.2011.0276

Rastegary, H., & Landy, F. J. (1993). The interactions among time urgency, uncertainty, and time pressure. In O. Svenson & A. J. Maule (Eds.), *Time pressure and stress in human judgment and decision making* (pp. 217–239). PlenumPress.

Rauch, M. (2025). Between war and peace: How boredom shapes the enactment of idealized futures in extreme contexts. *Academy of Management Journal*, *68*(3), 475–506. https://doi.org/10.5465/amj.2023.0618

Ravasi, D., Rindova, V., & Stigliani, I. (2019). The stuff of legend: History, memory, and the temporality of organizational identity construction. *Academy of Management Journal*, *62*(5), 1523–1555. https://doi.org/10.5465/amj.2016.0505

Reilly, G., Souder, D., & Ranucci, R. (2016). Time horizon of investments in the resource allocation process: Review and framework for next steps. *Journal of Management*, *42*(5), 1169–1194. https://doi.org/10.1177/0149206316630381

Reinecke, J., & Ansari, S. (2015). When times collide: Temporal brokerage at the intersection of markets and developments. *Academy of Management Journal*, *58*(2), 618–648. https://doi.org/10.5465/amj.2012.1004

Renzl, B., Mahringer, C. A., Rost, M., & Scheible, L. (2021). Organizational agility: Current challenges and future opportunities. *Journal of Competences, Strategy & Management*, *11*, 1–10. https://doi.org/10.25437/jcsm-vol11-51

Repenning, N. P. (2001). Understanding fire fighting in new product development. *Journal of Product Innovation Management*, *18*(5), 285–300. https://doi.org/10.1111/1540-5885.1850285

Rhymer, J. (2023). Location-independent organizations: Designing collaboration across space and time. *Administrative Science Quarterly*, *68*(1), 1–43. https://doi.org/10.1177/00018392221129175

Rigby, D. K., Sutherland, J., & Noble, A. (2018). Agile at scale. *Harvard Business Review*, *96*(3), 88–96.

Rindova, V. P., & Martins, L. L. (2022). Futurescapes: Imagination and temporal reorganization in the design of strategic narratives. *Strategic Organization*, *20*(1), 200–224. https://doi.org/10.1177/1476127021989787

Ritter, F., Danner-Schröder, A., & Müller-Seitz, G. (2024). Agile routines enabling efficiency and flexibility: Demarcating and integrating temporal orientations. In C. A. Mahringer, B. T. Pentland, B. Renzl, K. Sele, & P. Spee (Eds.), *Routine dynamics: Organizing in a world of flux* (Vol. 88, pp. 151–177). Emerald. https://doi.org/10.1108/S0733-558X20240000088008

Rowell, C., Gustafsson, R., & Clemente, M. (2016). How institutions matter "in time": The temporal structures of practices and their effects on practice reproduction. In *How institutions matter!* (Vol. 48, pp. 303–327). Emerald. https://doi.org/10.1108/S0733-558X201600048A010

Sailer, P., Loscher, G. J., & Kaiser, S. (2023). Coordinated interdependence: How patterning governs flexibility in a routine cluster. *Journal of Management Studies*. https://doi.org/10.1111/joms.12960

Sandra, D., Segers, J., & Giacalone, R. (2023). How organizations can benefit from entrainment: A systematic literature review. *Journal of Organizational Change Management*, *36*(2), 233–256. https://doi.org/10.1108/JOCM-01-2022-0023

Sasaki, I., Kotlar, J., Ravasi, D., & Vaara, E. (2020). Dealing with revered past: Historical identity statements and strategic change in Japanese family firms. *Strategic Management Journal*, *41*(3), 590–623. https://doi.org/10.1002/smj.3065

Sastry, M. A. (1997). Problems and paradoxes in a model of punctuated organizational change. *Administrative Science Quarterly*, *42*(2), 237–275. https://doi.org/10.2307/2393920

Schatzki, T. R. (2006). On organizations as they happen. *Organization Studies*, *27*(12), 1863–1873. https://doi.org/10.1177/0170840606071942

Schinoff, B. S., Ashforth, B. E., & Corley, K. G. (2020). Virtually (in)separable: The centrality of relational cadence in the formation of virtual multiplex relationships.

Academy of Management Journal, *63*(5), 1395–1424. https://doi.org/10.5465/AMJ.2018.0466

Schriber, J. B., & Gutek, B. A. (1987). Some time dimensions of work: Measurement of an underlying aspect of organization culture. *Journal of Applied Psychology*, *72*(4), 642–650. https://doi.org/10.1037/0021-9010.72.4.642

Schultz, M. (2022). The strategy–identity nexus: The relevance of their temporal interplay to climate change. *Strategic Organization*, *20*(4), 821–831. https://doi.org/10.1177/14761270221125881

Schultz, M., & Hernes, T. (2013). A temporal perspective on organizational identity. *Organization Science*, *24*(1), 1–21. https://doi.org/10.1287/orsc.1110.0731

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The definitive guide to scrum: the rules of the game*.

Scott, W. R. (1995). *Institutions and organizations*. Sage.

Sharifi, H., & Zhang, Z. (1999). A methodology for achieving agility in manufacturing organisations: An introduction. *International Journal of Production Economics*, *62*, 7–22. https://doi.org/https://doi.org/10.1016/S0925-5273(98)00217-5

Shastri, Y., Hoda, R., & Amor, R. (2021). Spearheading agile: the role of the scrum master in agile projects. *Empirical Software Engineering*, *26*(1), 1–31. https://doi.org/10.1007/s10664-020-09899-4

Shi, W., & Prescott, J. E. (2012). Rhythm and entrainment of acquisition and alliance initiatives and firm performance: A temporal perspective. *Organization Studies*, *33*(10), 1281–1310. https://doi.org/10.1177/0170840612453530

Shipp, A. J., & Cole, M. S. (2015). Time in individual-level organizational studies: What is it, how is it used, and why isn't it exploited more often? *Annual Review of Organizational Psychology and Organizational Behavior*, *2*, 237–260. https://doi.org/10.1146/annurev-orgpsych-032414-111245

Shipp, A. J., Edwards, J. R., & Lambert, L. S. (2009). Conceptualization and measurement of temporal focus: The subjective experience of the past, present, and future. *Organizational Behavior and Human Decision Processes*, *110*(1), 1–22. https://doi.org/10.1016/j.obhdp.2009.05.001

Shipp, A. J., & Fried, Y. (2014). Time research in management: Using temporal ambassadors to translate ideas into reality. In A. J. Shipp & Y. Fried (Eds.), *Time and Work* (Vol. 1). Psychology Press.

Shipp, A. J., & Jansen, K. J. (2011). Reinterpreting time in fit theory: Crafting and recrafting narratives of fit in medias res. *Academy of Management Review*, *36*(1), 76–101. https://doi.org/10.5465/amr.2009.0077

Shipp, A. J., & Jansen, K. J. (2021). The "other" time: A review of the subjective experience of time in organizations. *Academy of Management Annals*, *15*(1), 299–334. https://doi.org/10.5465/annals.2018.0142

Shipp, A. J., & Richardson, H. A. (2021). The impact of temporal schemata: Understanding when individuals entrain versus resist or create temporal structure. *Academy of Management Review*, *46*(2), 299–319. https://doi.org/10.5465/amr.2017.0384

Siggelkow, N. (2007). Persuasion with case studies. *Academy of Management Journal*, *50*(1), 20–24. https://doi.org/https://doi.org/10.5465/amj.2007.24160882

Slawinski, N., & Bansal, P. (2012). A matter of time: The temporal perspectives of organizational responses to climate change. *Organization Studies*, *33*(11), 1537–1563. https://doi.org/10.1177/0170840612463319

Smets, M., Jarzabkowski, P. A., Burke, G. T., & Spee, P. (2015). Reinsurance trading in Lloyd's of London: Balancing conflicting-yet-complementary logics in practice. *Academy of Management Journal*, *58*(3), 932–970. https://doi.org/10.5465/amj.2012.0638

Smith, W. K., & Lewis, M. W. (2011). Toward a theory of paradox: A dynamic equilibrium model of organizing. *Academy of Management Review*, *36*(2), 381–403. https://doi.org/10.5465/amr.2009.0223

Soda, G., Tortoriello, M., & Iorio, A. (2018). Harvesting value from brokerage. *Academy of Management Journal*, *61*(3), 896–918. https://doi.org/10.2307/26528883

Sonnentag, S. (2012). Time in organizational research: Catching up on a long neglected topic in order to improve theory. *Organizational Psychology Review*, *2*(4), 361–368. https://doi.org/10.1177/2041386612442079

Sorokin, P. A., & Merton, R. K. (1937). Social time: A methodological and functional analysis. *American Journal of Sociology*, *42*(5), 615–629. https://doi.org/10.1086/217540

Souder, D., & Bromiley, P. (2012). Explaining temporal orientation: Evidence from the durability of firms' capital investments. *Strategic Management Journal*, *33*(5), 550–569. https://doi.org/10.1002/smj.970

Špundak, M. (2014). Mixed agile/traditional project management methodology – reality or illusion? *Procedia - Social and Behavioral Sciences*, *119*, 939–948. https://doi.org/10.1016/j.sbspro.2014.03.105

Stacey, R. D. (2012). *Tools and techniques of leadership and management: Meeting the challenge of complexity*. Routledge.

Star, S. L. (2010). This is not a boundary object: Reflections on the origin of a concept. *Science Technology and Human Values*, *35*(5), 601–617. https://doi.org/10.1177/0162243910377624

Star, S. L., & Griesemer, J. R. (1989). Institutional ecology, "transitions" and boundary objects: Amateurs and professionals in Berkeley's Museum of Vertrebrate Zoology, 1907-39. *Social Studies of Science*, *19*, 387–420. https://doi.org/10.1177/030631289019003001

Staudenmayer, N., Tyre, M., & Perlow, L. (2002). Time to change: Temporal shifts as enablers of organizational change. *Organization Science*, *13*(5), 583–597. https://doi.org/10.1287/orsc.13.5.583.7813

Stjerne, I. S., Geraldi, J., & Wenzel, M. (2024). Strategic practice drift: How open strategy infiltrates the strategy process. *Journal of Management Studies*, *61*(3), 820–856. https://doi.org/10.1111/joms.12895

Stjerne, I. S., Söderlund, J., & Minbaeva, D. (2019). Crossing times: Temporal boundary-spanning practices in interorganizational projects. *International Journal of Project Management*, *37*(2), 347–365. https://doi.org/10.1016/j.ijproman.2018.09.004

Stjerne, I. S., & Svejenova, S. (2016). Connecting temporary and permanent trganizing: Tensions and boundary work in sequential film projects. *Organization Studies*, *37*(12), 1771–1792. https://doi.org/10.1177/0170840616655492

Stonig, J., & Müller-Stewens, G. (2025). When lives are at stake: Managing temporal complexity with a strategy process repertoire. *Organization Science*. https://doi.org/10.1287/orsc.2022.17173

Stover, J. F. (1970). *The life and decline of the American railroad*. Oxford University Press.

Strathman, A., Gleicher, F., Boninger, D. S., & Scott Edwards, C. (1994). The consideration of future consequences: Weighing immediate and distant outcomes of behavior. *Journal of Personality and Social Psychology*, *66*(4), 742–752. https://doi.org/10.1037/0022-3514.66.4.742

Strauss, A., & Corbin, J. (1998). *Basics of qualitative research: Techniques and procedures for developing grounded theory*. Sage.

Stray, V., Sjøberg, D. I. K., & Dybå, T. (2016). The daily stand-up meeting: A grounded theory study. *Journal of Systems and Software*, *114*, 101–124. https://doi.org/10.1016/j.jss.2016.01.004

Sturdy, A. (1997). The consultancy process — An insecure business? *Journal of Management Studies*, *34*(3), 389–413. https://doi.org/10.1111/1467-6486.00056

Suchman, M. C. (1995). Managing legitimacy: Strategic and institutional approaches. *Academy of Management Review*, *20*(3), 571–610. https://doi.org/10.5465/amr.1995.9508080331

Suddaby, R., Bitektine, A., & Haack, P. (2017). Legitimacy. *Academy of Management Annals*, *11*(1), 451–478. https://doi.org/10.5465/annals.2015.0101

Sztompka, P. (1993). *The sociology of social change*. Blackwell.

Taeuscher, K., Bouncken, R., & Pesch, R. (2021). Gaining legitimacy by being different: Optimal distinctiveness in crowdfunding platforms. *Academy of Management Journal*, *64*(1), 149–179. https://doi.org/10.5465/amj.2018.0620

Tajfel, H., & Turner, J. C. (1978). Intergroup behavior. In H. Tajfel & C. Fraser (Eds.), *Introducing social psychology* (pp. 149–178). Penguin Books Limited.

Tallon, P. P., & Pinsonneault, A. (2011). Competing perspectives on the link between strategic information technology alignment and organizational agility: Insights from a mediation model. *MIS Quarterly*, *35*(2), 463–486. https://doi.org/10.2307/23044052

Tang, S., Richter, A. W., & Nadkarni, S. (2020). Subjective time in organizations: Conceptual clarification, integration, and implications for future research. *Journal of Organizational Behavior*, *41*(2), 210–234. https://doi.org/10.1002/job.2421

Teece, D., Peteraf, M., & Leih, S. (2016). Dynamic capabilities and organizational agility: Risk, uncertainty, and strategy in the innovation economy. *California Management Review*, *58*(4), 13–35.

Tereso, A., Ribeiro, P., Fernandes, G., Loureiro, I., & Ferreira, M. (2019). Project management practices in private organizations. *Project Management Journal*, *50*(1), 6–22. https://doi.org/10.1177/8756972818810966

Tessem, B. (2014). Individual empowerment of agile and non-agile software developers in small teams. *Information and Software Technology*, *56*(8), 873–889. https://doi.org/10.1016/j.infsof.2014.02.005

Theobald, S., & Diebold, P. (2018). Interface problems of agile in a non-agile environment. In J. Garbajosa, X. Wang, & A. Aguiar (Eds.), *19th International Conference, XP 2018*. http://www.springer.com/series/7911

Thompson, J. D. (1967). *Organizations in action*. McGraw-Hill.

Tripp, J. F., & Armstrong, D. J. (2018). Agile methodologies: Organizational adoption motives, tailoring, and performance. *Journal of Computer Information Systems*, *58*(2), 170–179. https://doi.org/10.1080/08874417.2016.1220240

Tripp, J. F., Saltz, J., & Turk, D. (2018). Thoughts on current and future research on agile and lean: Ensuring relevance and rigor. *Proceedings of the 51st Hawaii International Conference on System Sciences*, 5465–5472.

Tukiainen, S., & Granqvist, N. (2016). Temporary organizing and institutional change. *Organization Studies*, *37*(12), 1819–1840. https://doi.org/10.1177/0170840616662683

Tuma, N. B., & Hannan, M. T. (1984). *Social dynamics: Models and methods*. Academic Press.

Turner, S. F., & Rindova, V. P. (2018). Watching the clock: Action timing, patterning and routine performance. *Academy of Management Journal*, *61*(4), 1253–1280. https://doi.org/10.2307/26528900

Tushman, M. L., & O'Reilly, C. A. (1996). Ambidextrous organizations: Managing evolutionary and revolutionary change. *California Management Review*, *38*(4), 8–30. https://doi.org/10.2307/4116585

Tyre, M. J., & Orlikowski, W. J. (1994). Windows of opportunity: Temporal patterns of technological adaptation in organizations. *Organization Science*, *5*(1), 98–118. https://doi.org/10.1287/orsc.5.1.98

van Berkel, F. J. F. W., Ferguson, J. E., & Groenewegen, P. (2016). Speedy delivery versus long-term objectives: How time pressure affects coordination between temporary projects and permanent organizations. *Long Range Planning*, *49*(6), 661–673. https://doi.org/10.1016/j.lrp.2016.04.001

van de Ven, A. H., Delbecq, A. L., & Koenig, R. (1976). Determinants of coordination modes within organizations. *American Sociological Review*, *41*(2), 322–338. https://doi.org/10.2307/2094477

van Maanen, J. (1988). *Tales of the field*. University of Chicago Press.

van Oorschot, K. E., Sengupta, K., & van Wassenhove, L. N. (2018). Under pressure: The effects of iteration lengths on agile software development performance. *Project Management Journal*, *49*(6), 78–102. https://doi.org/10.1177/8756972818802714

Vermeulen, P. A. M. (2005). Uncovering barriers to complex incremental product innovation in small and medium-sized financial services firms*. *Journal of Small Business Management*, *43*(4), 432–452. https://doi.org/10.1111/j.1540-627X.2005.00146.x

Vermeulen, P. A. M., van den Bosch, F. A. J., & Volberda, H. W. (2007). Complex incremental product innovation in established service firms: A micro institutional perspective. *Organization Studies*, *28*(10), 1523–1546. https://doi.org/10.1177/0170840607068082

Vinekar, V., Slinkman, C. W., & Nerur, S. (2006). Can agile and traditional systems development approaches coexist? An ambidextrous view. *Information Systems Management*, *23*(3), 31–42. https://doi.org/10.1201/1078.10580530/46108.23.3.20060601/93705.4

Volberda, H. W., Baden-Fuller, C., & van den Bosch, F. A. J. (2001). Mastering strategic renewal: Mobilising renewal journeys in multi-unit firms. *Long Range Planning*, *34*(2), 159–178. https://doi.org/10.1016/S0024-6301(01)00032-2

Waller, M. J., Conte, J. M., & Gibson, C. B. (2001). The effect of individual perceptions of deadlines on team performance. *Academy of Management Review*, *26*(4), 586–600. https://doi.org/10.5465/amr.2001.5393894

Waller, M. J., Giambatista, R. C., & Zellmer-Bruhn, M. E. (1999). The effects of individual time urgency on group polychronicity. *Journal of Managerial Psychology*, *14*(3/4), 268–3946. https://doi.org/10.1108/02683949910263765

Waller, M. J., Zellmer-Bruhn, M. E., & Giambatista, R. C. (2002). Watching the clock: Group pacing behavior under dynamic deadlines. *Academy of Management Journal*, *45*(5), 1046–1055. https://doi.org/10.5465/3069329

Wang, S. L., Luo, Y., Maksimov, V., Sun, J., & Celly, N. (2019). Achieving Temporal Ambidexterity in New Ventures. *Journal of Management Studies*, *56*(4), 788–822. https://doi.org/10.1111/joms.12431

Wang, T., & Bansal, P. (2012). Social responsibility in new ventures: Profiting from a long-term orientation. *Strategic Management Journal*, *33*(10), 1135–1153. https://doi.org/10.1002/smj.1962

Webb, D., & Pettigrew, A. (1999). The temporal development of strategy: Patterns in the U.K. insurance industry. *Organization Science*, *10*(5), 601–621. https://doi.org/10.1287/orsc.10.5.601

Weick, K. E. (1976). Educational organizations as loosely coupled systems. *Administrative Science Quarterly*, *21*, 1–19.

Weick, K. E. (2007). The generative properties of richness. *Academy of Management Journal*, *50*(1), 14–19. https://doi.org/10.5465/amj.2007.24160637

Wenzel, M., Cabantous, L., & Koch, J. (2025). Future making: Towards a practice perspective. *Journal of Management Studies*. https://doi.org/10.1111/joms.13222

Wenzel, M., Krämer, H., Koch, J., & Reckwitz, A. (2020). Future and Organization Studies: On the rediscovery of a problematic temporal category in organizations. *Organization Studies*, *41*(10), 1441–1455. https://doi.org/10.1177/0170840620912977

Wood, M. S., Bakker, R. M., & Fisher, G. (2021). Back to the future: A time-calibrated theory of entrepreneurial action. *Academy of Management Review*, *46*(1), 147–171. https://doi.org/10.5465/AMR.2018.0060

Woodward, J. (1965). *Industrial organization: Theory and practice*. Oxford University Press.

Wry, T., Lounsbury, M., & Glynn, M. A. (2011). Legitimating nascent collective identities: Coordinating cultural entrepreneurship. *Organization Science*, *22*(2), 449–463. https://doi.org/10.1287/orsc.1100.0613

Yakura, E. K. (2002). Charting time: Timelines as temporal boundary objects. *Academy of Management Journal*, *45*(5), 956–970. https://doi.org/10.5465/3069324

Ybema, S. (2010). Talk of change: Temporal contrasts and collective identities. *Organization Studies*, *31*(4), 481–503. https://doi.org/10.1177/0170840610372205

Yin, R. K. (2009). *Case study research: Design and methods* (4th ed.). Sage.

Zerubavel, E. (1976). Timetables and scheduling: On the social organization of time. *Sociological Inquiry*, *46*(2), 87–94. https://doi.org/10.1111/j.1475-682X.1976.tb00753.x

Zerubavel, E. (1981). *Hidden rhythms: Schedules and calendars in social life*. University of California Press.

Zhang, P., Priem, R., Wang, D., & Li, S. (2023). Strategic rhythms: Insights and research directions. *Journal of Management*, *49*(6), 1939–1964. https://doi.org/10.1177/01492063221127910

Zinn, M. M., Müller, C., & Herter, C. (2023). A clash of cultures: How do cultural differences impact agile transformations? *Gruppe. Interaktion. Organisation. Zeitschrift für Angewandte Organisationspsychologie*, *54*(2), 165–176. https://doi.org/10.1007/s11612-023-00687-4

Appendix

Table 16: List of interview partners

Case:	Interview partner:	Affiliation:	Role:	Interview date:	Duration:	Pages:
[A][13]	I.A.1	non-agile	leader	2021/06/21	61 min	17
[A1]	I.A1.1	agile	leader	2021/06/07	50 min	13
[A1]	I.A1.2	agile	leader	2021/06/17	54 min	18
[A1]	I.A1.3	agile	leader	2021/06/24	44 min	14
[A2]	I.A2.1-1	agile	leader	2021/06/11	72 min	18
[A2]	I.A2.1-2	agile	leader	2021/11/15	93 min	32
[A2]	I.A2.2	non-agile	member	2021/06/24	47 min	15
[A2]	I.A2.3	non-agile	member	2021/06/30	35 min	12
[B]	I.B.1	non-agile	leader	2021/09/22	55 min	16
[B]	I.B.2	agile	leader	2021/12/16	48 min	11
[B1]	I.B1.1-1	agile	leader	2021/08/26	38 min	17
[B1]	I.B1.1-2	agile	leader	2021/08/31	58 min	22
[B1]	I.B1.2	non-agile	leader	2021/11/15	62 min	22
[B1]	I.B1.3	agile	leader	2021/10/07	50 min	18
[B1]	I.B1.4	agile	leader	2021/10/07	50 min	18
[B1]	I.B1.5	agile	leader	2021/09/08	57 min	19
[B1]	I.B1.6	agile	leader	2021/09/22	51 min	18
[B1]	I.B1.7	agile	leader	2021/09/06	51 min	16
[B1]	I.B1.8	non-agile	member	2022/05/04	29 min	12
[B]	I.B.3	agile	leader	2022/03/10	78 min	26
[B]	I.B.4	agile/non-agile	member	2021/11/29	45 min	15
[B2]	I.B2.1	agile	leader	2021/10/15	53 min	18
[B2]	I.B2.2	non-agile	member	2021/11/18	41 min	16
[B2]	I.B2.3	agile	leader	2021/11/24	55 min	20
[B2]	I.B2.4	agile	leader	2021/11/25	43 min	15
[B2]	I.B2.5	agile	leader	2021/12/17	59 min	23
[B3]	I.B3.1	agile/non-agile	leader	2022/03/16	32 min	13
[B3]	I.B3.2-1	agile	leader	2021/11/18	45 min	17
[B3]	I.B3.2-2	agile	leader	2021/11/25	30 min	12
[B3]	I.B3.3	agile/non-agile	member	2021/11/29	48 min	18
[B3]	I.B3.4	agile	member	2022/03/17	30 min	12
[C]	I.C.1	agile	leader	2021/07/05	57 min	18
[C]	I.C.2	agile	leader	2021/07/07	63 min	20
[C]	I.C.3	agile	leader	2021/07/20	57 min	18
[C]	I.C.4	agile	leader	2021/07/09	45 min	13
[C]	I.C.5	agile	leader	2021/07/09	45 min	13
[C]	I.C.6	agile	leader	2021/07/12	n.a.	4
[C]	I.C.7-1	agile	member	2021/07/12	42 min	12
[C]	I.C.7-2	agile	member	2022/03/03	15 min	6
[C]	I.C.8	non-agile	member	2021/07/12	37 min	12
[C]	I.C.9	non-agile	member	2021/07/12	37 min	12
[C]	I.C.10	agile	member	2021/07/13	54 min	19
[C]	I.C.11	agile	member	2021/07/19	24 min	8
[C]	I.C.12	agile	member	2021/07/19	31 min	11

[13] Interviews labeled as cases [A] and [B] refer to individuals who were able to provide information on multiple cases within one organization. I.A.1 reported on cases [A1] and [A2]. I.B.1 and I.B.2 reported on the three cases [B1], [B2], and [B3]. I.B.3 and I.B.4 reported on cases [B2] and [B3].

Case:	Interview partner:	Affiliation:	Role:	Interview date:	Duration:	Pages:
[C]	I.C.13	agile	member	2021/07/19	28 min	12
[C]	I.C.14	non-agile	member	2021/07/19	33 min	10
[C]	I.C.15	non-agile	leader	2021/07/19	37 min	13
[C]	I.C.16	non-agile	member	2021/07/19	20 min	8
[C]	I.C.17	non-agile	member	2021/07/20	31 min	10
[C]	I.C.18	non-agile	member	2021/07/21	29 min	10
[C]	I.C.19	non-agile	member	2021/07/22	46 min	14
[C]	I.C.20	non-agile	member	2021/07/23	29 min	7
[C]	I.C.21	agile	leader	2021/07/23	23 min	11
[C]	I.C.22	non-agile	member	2021/07/26	31 min	8
[C]	I.C.23	non-agile	leader	2021/07/26	24 min	8
[C]	I.C.24	agile	leader	2021/07/27	24 min	10
[C]	I.C.25	non-agile	member	2021/08/02	32 min	7
[C]	I.C.26	non-agile	leader	2021/08/02	21 min	7
[C]	I.C.27	non-agile	leader	2021/08/06	18 min	6
[C]	I.C.28	non-agile	leader	2021/08/13	21 min	6
[C]	I.C.29	non-agile	leader	2021/08/13	19 min	6
[D]	I.D.1	agile	leader	2021/07/23	54 min	19
[D]	I.D.2	agile	leader	2021/07/23	60 min	21
[D]	I.D.3	agile	leader	2021/07/26	53 min	20
[D]	I.D.4	agile	leader	2021/07/27	51 min	18
[D]	I.D.5	non-agile	member	2021/07/29	52 min	18
[D]	I.D.6	non-agile	member	2021/07/29	40 min	14
[D]	I.D.7	agile	member	2021/07/27	50 min	14
[D]	I.D.8	non-agile	leader	2021/08/02	45 min	13
[D]	I.D.9	non-agile	leader	2021/08/25	26 min	10
[D]	I.D.10	non-agile	member	2021/09/15	23 min	8
[D]	I.D.11	non-agile	member	2021/08/10	46 min	15
[D]	I.D.12	agile	member	2021/07/27	51 min	17
[E]	I.E.1	non-agile	leader	2021/08/24	28 min	9
[E]	I.E.2	agile	member	2021/08/24	23 min	8
[E]	I.E.3	agile	member	2021/08/27	33 min	12
[E]	I.E.4	non-agile	member	2021/08/26	28 min	10
[E]	I.E.5	agile	member	2021/08/27	52 min	17
[E]	I.E.6	agile	leader	2021/08/30	61 min	18
[E]	I.E.7	agile	leader	2021/09/07	67 min	22
[E]	I.E.8	non-agile	leader	2021/08/26	56 min	17
[E]	I.E.9	non-agile	leader	2021/08/27	35 min	12
[E]	I.E.10	non-agile	member	2021/08/30	22 min	10
[E]	I.E.11	agile	member	2021/09/03	33 min	10
[E]	I.E.12	agile	member	2021/08/27	27 min	10
[E]	I.E.13	agile	member	2021/09/03	26 min	10
[E]	I.E.14	non-agile	member	2021/08/30	11 min	4
[E]	I.E.15	non-agile	member	2021/09/03	18 min	7
[E]	I.E.16	non-agile	member	2021/09/07	24 min	8
[E]	I.E.17	agile	member	2021/09/08	26 min	9
[E]	I.E.18	non-agile	member	2021/08/31	32 min	11

Table 17: List of participatory observations

case	observation number	observation date	occasion	duration	pages of field notes:
[A1]	O.A1.1	2021/07/08	release management jour fixe	22 min	9
[A1]	O.A1.2	2021/10/11	sprint review	27 min	6
[A2]	O.A2.1	2021/06/11	agile method training	98 min	20
[A2]	O.A2.2	2021/06/14	sprint review	92 min	15
[A2]	O.A2.3	2021/06/14	sprint planning	53 min	22
[A2]	O.A2.4	2021/07/06	sprint review	55 min	7
[A2]	O.A2.5	2021/07/06	sprint planning	75 min	9
[A2]	O.A2.6	2021/08/17	sprint review	36 min	8
[A2]	O.A2.7	2021/08/17	sprint planning	105 min	9
[A2]	O.A2.8	2021//09/7	sprint review	60 min	12
[A2]	O.A2.9	2021/09/07	sprint planning	115 min	16
[A2]	O.A2.10	2021/10/20	sprint review	59 min	15
[A2]	O.A2.11	2021/10/20	sprint planning	110 min	23
[A2]	O.A2.12	2021/11/09	sprint review	61 min	15
[A2]	O.A2.13	2021/11/09	sprint planning	27 min	4
[A2]	O.A2.14	2021/11/30	sprint review	90 min	11
[B1]	O.B1.1	2021/10/11	data warehouse jour fixe	10 min	3
[B1]	O.B1.2	2022/01/25	sprint review	60 min	6
[B1]	O.B1.3	2022/02/08	sprint review	43 min	10
[B1]	O.B1.4	2022/02/22	sprint review	45 min	9
[B2]	O.B2.1	2021/11/17	daily standup meeting	15 min	2
[B2]	O.B2.2	2021/11/18	refinement	120 min	14
[B2]	O.B2.3	2021/11/19	daily standup meeting	12 min	3
[B2]	O.B2.4	2021/11/22	daily standup meeting	12 min	3
[B2]	O.B2.5	2021/11/25	daily standup meeting	19 min	2
[B2]	O.B2.6	2021/11/26	daily standup meeting	13 min	2
[B2]	O.B2.7	2021/11/29	daily standup meeting	21 min	5
[B2]	O.B2.8	2021/11/30	sprint review	26 min	5
[B2]	O.B2.9	2021/12/01	sprint planning	85 min	10
[B2]	O.B2.10	2021/12/02	daily standup meeting	15 min	4
[B2]	O.B2.11	2021/12/03	daily standup meeting	10 min	1
[B2]	O.B2.12	2021/12/06	daily standup meeting	17 min	6
[B2]	O.B2.13	2021/12/07	daily standup meeting	20 min	6
[B2]	O.B2.14	2021/12/08	daily standup meeting	20 min	3
[B2]	O.B2.15	2021/12/09	daily standup meeting	20 min	4
[B2]	O.B2.16	2021/12/15	daily standup meeting	10 min	4
[B2]	O.B2.17	2021/12/16	daily standup meeting	17 min	4
[B2]	O.B2.18	2021/12/20	daily standup meeting	16 min	3
[B2]	O.B2.19	2021/12/21	daily standup meeting	15 min	2
[B2]	O.B2.20	2021/12/22	sprint planning	120 min	13
[B2]	O.B2.21	2021/12/23	daily standup meeting	20 min	4
[B3]	O.B3.1	2021/12/14	sprint review	70 min	21
[B3]	O.B3.2	2022/01/11	sprint review	65 min	12
[B3]	O.B3.3	2022/01/25	sprint review	32 min	8
[B3]	O.B3.4	2022/02/08	sprint review	35 min	7
[B3]	O.B3.5	2022/02/22	sprint review	56 min	13
[B3]	O.B3.6	2022/03/22	sprint review	60 min	17
[B3]	O.B3.7	2022/04/05	sprint review	55 min	18
[B3]	O.B3.8	2022/04/19	sprint review	13 min	3
[C]	O.C.1	2021/10/05	daily standup meeting	28 min	9
[C]	O.C.2	2021/10/06	daily standup meeting	11 min	4
[C]	O.C.3	2021/10/14	sprint review	32 min	12
[C]	O.C.4	2021/10/14	sprint planning	35 min	7
[C]	O.C.5	2021/10/19	bug fixing meeting	28 min	9
[C]	O.C.6	2021/10/26	sprint retrospective	32 min	16
[C]	O.C.7	2021/10/27	sprint review	62 min	10

case	observation number	observation date	occasion	duration	pages of field notes:
[C]	O.C.8	2021/10/28	sprint planning	39 min	16
[C]	O.C.9	2021/11/10	sprint review	59 min	11
[C]	O.C.10	2021/11/10	sprint retrospective	17 min	5
[C]	O.C.11	2021/11/11	sprint planning	42 min	8
[C]	O.C.12	2021/11/12	management jour fixe	75 min	37
[C]	O.C.13	2021/11/24	sprint retrospective	49 min	10
[C]	O.C.14	2021/11/25	sprint review	56 min	10
[C]	O.C.15	2021/11/25	sprint planning	41 min	9
[C]	O.C.16	2021/12/08	sprint review	30 min	6
[C]	O.C.17	2021/12/08	sprint retrospective	23 min	5
[C]	O.C.18	2021/12/09	sprint planning	52 min	9
[C]	O.C.19	2021/12/22	sprint review	33 min	7
[C]	O.C.20	2021/12/22	sprint retrospective	21 min	3
[C]	O.C.21	2021/12/22	sprint planning	33 min	4
[C]	O.C.22	2022/01/19	sprint review	43 min	4
[C]	O.C.23	2022/01/19	sprint retrospective	35 min	4
[C]	O.C.24	2022/01/20	sprint planning	29 min	7
[C]	O.C.25	2022/02/02	sprint review	29 min	9
[C]	O.C.26	2022/02/02	sprint retrospective	31 min	7
[C]	O.C.27	2022/02/16	sprint review	50 min	12
[C]	O.C.28	2022/02/16	sprint retrospective	28 min	5
[C]	O.C.29	2022/02/17	sprint planning	60 min	12
[C]	O.C.30	2022/03/02	sprint retrospective	28 min	5
[C]	O.C.31	2022/03/02	sprint planning	23 min	7
[C]	O.C.32	2022/03/16	refinement	38 min	7
[C]	O.C.33	2022/03/16	sprint review	59 min	12
[C]	O.C.34	2022/03/16	sprint retrospective	19 min	4
[C]	O.C.35	2022/03/17	sprint planning	17 min	5
[C]	O.C.36	2022/03/30	refinement	34 min	8
[C]	O.C.37	2022/03/30	sprint retrospective	28 min	6
[C]	O.C.38	2022/03/31	sprint review & sprint planning	35 min	9
[C]	O.C.39	2022/04/13	sprint review	23 min	5
[C]	O.C.40	2022/04/13	sprint retrospective	43 min	10
[C]	O.C.41	2022/04/14	sprint planning	27 min	10
[C]	O.C.42	2022/04/27	sprint retrospective	30 min	6
[C]	O.C.43	2022/04/28	sprint planning	29 min	10
[C]	O.C.44	2022/05/11	sprint review	29 min	5
[C]	O.C.45	2022/05/12	sprint planning	46 min	9
[C]	O.C.46	2022/05/25	sprint review	45 min	8
[C]	O.C.47	2022/05/25	sprint planning	24 min	4
[C]	O.C.48	2022/06/22	sprint review	16 min	4
[C]	O.C.49	2022/06/22	sprint retrospective	21 min	4
[C]	O.C.50	2022/06/23	sprint planning	25 min	6
[C]	O.C.51	2022/07/06	sprint retrospective	10 min	1
[C]	O.C.52	2022/07/06	sprint review & sprint planning	42 min	8
[C]	O.C.53	2022/07/07	sprint retrospective	21 min	4
[C]	O.C.54	2022/07/21	sprint review & sprint planning	30 min	7
[C]	O.C.55	2022/08/03	sprint retrospective	21 min	4
[C]	O.C.56	2022/08/04	sprint review & sprint planning	37 min	9
[C]	O.C.57	2022/08/17	sprint retrospective	28 min	4
[C]	O.C.58	2022/08/31	sprint retrospective	60 min	8
[C]	O.C.59	2022/09/01	sprint review	15 min	3
[C]	O.C.60	2022/09/01	sprint planning	48 min	10
[C]	O.C.61	2022/09/14	sprint retrospective	20 min	4
[C]	O.C.62	2022/09/15	sprint review & sprint planning	37 min	12
[C]	O.C.63	2022/09/28	sprint retrospective	38 min	7
[C]	O.C.64	2022/09/29	sprint review & sprint planning	70 min	11
[C]	O.C.65	2022/10/12	sprint retrospective	60 min	6

case	observation number	observation date	occasion	duration	pages of field notes:
[C]	O.C.66	2022/10/13	sprint review	29 min	6
[C]	O.C.67	2022/10/13	sprint planning	44 min	8
[C]	O.C.68	2022/10/26	sprint retrospective	40 min	4
[C]	O.C.69	2022/11/09	sprint retrospective	25 min	7
[C]	O.C.70	2022/11/10	sprint review	27 min	6
[C]	O.C.71	2022/11/10	sprint planning	52 min	11
[C]	O.C.72	2022/11/22	sprint retrospective	37 min	7
[C]	O.C.73	2022/11/24	sprint review	36 min	9
[C]	O.C.74	2022/11/24	sprint planning	20 min	6
[C]	O.C.75	2022/12/07	sprint retrospective	40 min	4
[C]	O.C.76	2022/12/08	sprint review	30 min	6
[C]	O.C.77	2022/12/08	sprint planning	20 min	5
[C]	O.C.78	2022/12/15	sprint retrospective	53 min	8
[C]	O.C.79	2022/12/16	sprint review	14 min	2
[C]	O.C.80	2022/12/16	sprint planning	40 min	7
[C]	O.C.81	2023/01/11	sprint retrospective	17 min	4
[C]	O.C.82	2023/01/12	sprint review	32 min	6
[C]	O.C.83	2023/01/12	sprint planning	37 min	8
[C]	O.C.84	2023/01/18	sprint retrospective	17 min	5
[C]	O.C.85	2023/01/19	sprint review	43 min	9
[C]	O.C.86	2023/01/19	sprint planning	37 min	6
[C]	O.C.87	2023/02/01	sprint retrospective	26 min	6
[C]	O.C.88	2023/02/02	sprint review	43 min	6
[C]	O.C.89	2023/02/02	sprint planning	25 min	6
[C]	O.C.90	2023/02/16	sprint review	50 min	11
[C]	O.C.91	2023/02/16	sprint planning	60 min	11
[C]	O.C.92	2023/03/01	sprint retrospective	26 min	6
[C]	O.C.93	2023/03/02	sprint review	39 min	8
[C]	O.C.94	2023/03/02	sprint planning	23 min	6
[C]	O.C.95	2023/03/15	sprint retrospective	27 min	7
[C]	O.C.96	2023/03/16	sprint review	35 min	9
[C]	O.C.97	2023/03/16	sprint planning	23 min	6
[C]	O.C.98	2023/03/29	sprint retrospective	29 min	5
[C]	O.C.99	2023/03/30	sprint review	28 min	5
[C]	O.C.100	2023/03/30	sprint planning	25 min	5
[C]	O.C.101	2023/04/12	sprint retrospective	20 min	5
[C]	O.C.102	2023/04/26	sprint retrospective	18 min	5
[C]	O.C.103	2023/04/27	sprint review	35 min	6
[C]	O.C.104	2023/04/27	sprint planning	15 min	3
[C]	O.C.105	2023/05/10	sprint retrospective	35 min	4
[C]	O.C.106	2023/05/11	sprint review	34 min	5
[C]	O.C.107	2023/05/11	sprint planning	15 min	3
[C]	O.C.108	2023/05/24	sprint retrospective	17 min	6
[C]	O.C.109	2023/05/24	sprint review and sprint planning	50 min	9
[C]	O.C.110	2023/06/07	sprint review	50 min	7
[C]	O.C.111	2023/06/07	sprint planning	30 min	4
[C]	O.C.112	2023/06/21	sprint retrospective	56 min	7
[C]	O.C.113	2023/06/22	sprint review	50 min	9
[C]	O.C.114	2023/06/22	sprint planning	20 min	3
[C]	O.C.115	2023/07/05	sprint retrospective	29 min	4
[C]	O.C.116	2023/07/06	sprint review	55 min	10
[C]	O.C.117	2023/07/06	sprint planning	30 min	5
[C]	O.C.118	2023/07/19	sprint retrospective	30 min	7
[C]	O.C.119	2023/07/20	sprint planning	30 min	4
[C]	O.C.120	2023/08/17	sprint review	30 min	5
[C]	O.C.121	2023/08/17	sprint planning	30 min	3
[C]	O.C.122	2023/08/17	sprint retrospective	30 min	9
[D]	O.D.1	2021/07/19	weekly	23 min	6

case	observation number	observation date	occasion	duration	pages of field notes:
[D]	O.D.2	2021/07/20	sprint review	60 min	16
[D]	O.D.3	2021/07/21	sprint planning	48 min	13
[D]	O.D.4	2021/07/26	coordination meeting	45 min	13
[D]	O.D.5	2021/08/02	weekly	32 min	13
[D]	O.D.6	2021/08/12	insurance line unit jour fixe	31 min	4
[D]	O.D.7	2021/08/17	sprint retrospective	60 min	19
[D]	O.D.8	2021/08/18	management jour fixe	90 min	19
[D]	O.D.9	2021/09/03	release management jour fixe	14 min	5
[E]	O.E.1	2021/08/09	synchronization meeting	10 min	5
[E]	O.E.2	2021/08/09	daily standup meeting	15 min	8
[E]	O.E.3	2021/08/09	bilateral interim conversation	40 min	10
[E]	O.E.4	2021/08/09	exchange meeting back office	35 min	14
[E]	O.E.5	2021/08/09	rehearsal of sprint review	150 min	2
[E]	O.E.6	2021/08/10	synchronization meeting	20 min	4
[E]	O.E.7	2021/08/10	daily standup meeting	15 min	4
[E]	O.E.8	2021/08/10	sprint refinement	15 min	3
[E]	O.E.9	2021/08/10	feedback meeting	15 min	1
[E]	O.E.10	2021/08/10	bilateral interim conversation	15 min	1
[E]	O.E.11	2021/08/10	sprint review	90 min	30
[E]	O.E.12	2021/08/10	sprint retrospective	60 min	16
[E]	O.E.13	2021/08/11	synchronization meeting	20 min	1
[E]	O.E.14	2021/08/11	sprint planning	120 min	14
[E]	O.E.15	2021/08/11	bilateral interim conversation	15 min	3
[E]	O.E.16	2021/08/11	agile team meeting	15 min	4
[E]	O.E.17	2021/08/12	daily standup meeting	15 min	3
[E]	O.E.18	2021/08/12	agile team meeting	15 min	4
[E]	O.E.19	2021/08/12	exchange with product owner	5	1
[E]	O.E.20	2021/08/12	exchange with scrum master	5 min	1
[E]	O.E.21	2021/08/13	daily standup meeting	15 min	4
[E]	O.E.22	2021/08/13	sprint refinement	40 min	9
[E]	O.E.23	2021/08/13	bilateral interim conversation	13 min	2
[E]	O.E.24	2021/08/13	agile team meeting	12 min	2
[E]	O.E.25	2021/08/16	synchronization meeting	5 min	1
[E]	O.E.26	2021/08/16	daily standup meeting	16 min	4
[E]	O.E.27	2021/08/16	bilateral interim conversation	6 min	1
[E]	O.E.28	2021/08/16	exchange with corporate client advisory unit	27 min	2
[E]	O.E.29	2021/08/16	agile team meeting	15 min	3
[E]	O.E.30	2021/08/16	exchange with private client advisory unit	27 min	5
[E]	O.E.31	2021/08/17	synchronization meeting	23 min	1
[E]	O.E.32	2021/08/17	daily standup meeting	14 min	3
[E]	O.E.33	2021/08/17	exchange with corporate client advisory unit	29 min	4
[E]	O.E.34	2021/08/17	exchange with corporate client advisory unit	34 min	6
[E]	O.E.35	2021/08/18	daily standup meeting	15 min	4
[E]	O.E.36	2021/08/18	rehearsal of sprint review	140 min	3
[E]	O.E.37	2021/08/18	agile team meeting	7 min	2
[E]	O.E.38	2021/08/19	daily standup meeting	9 min	2
[E]	O.E.39	2021/08/19	sprint review	90 min	13
[E]	O.E.40	2021/08/19	sprint retrospective	53 min	10
[E]	O.E.41	2021/08/19	sprint planning	116 min	33
[E]	O.E.42	2021/08/20	daily standup meeting	17 min	4
[E]	O.E.43	2021/08/20	exchange within the agile project team	44 min	1
[E]	O.E.44	2021/08/20	exchange within the agile project team	12 min	1

case	observation number	observation date	occasion	duration	pages of field notes:
[E]	O.E.45	2021/08/20	exchange within the agile project team	39 min	7
[E]	O.E.46	2021/08/21	bilateral interim conversation	15 min	1
[E]	O.E.47	2021/08/23	bilateral interim conversation	15 min	1
[E]	O.E.48	2021/08/23	daily standup-meeting	15 min	3
[E]	O.E.49	2021/08/23	exchange with management	96 min	43
[E]	O.E.50	2021/08/23	bilateral interim conversation	33 min	2
[E]	O.E.51	2021/08/24	daily standup meeting	14 min	2
[E]	O.E.52	2021/08/24	exchange with works council	28 min	8
[E]	O.E.53	2021/08/25	bilateral interim conversation	5 min	1
[E]	O.E.54	2021/08/25	daily standup meeting	15 min	5
[E]	O.E.55	2021/08/25	exchange with technical organization unit	18 min	5
[E]	O.E.56	2021/08/25	agile team meeting	13 min	3
[E]	O.E.57	2021/08/26	synchronization meeting	10 min	1
[E]	O.E.58	2021/08/26	daily standup meeting	16 min	5
[E]	O.E.59	2021/08/26	agile team meeting	32 min	4
[E]	O.E.60	2021/08/27	daily standup meeting	20 min	7
[E]	O.E.61	2021/08/27	exchange with management	65 min	17
[E]	O.E.62	2021/08/27	agile team meeting	19 min	5
[E]	O.E.63	2021/08/30	daily standup meeting	20 min	5
[E]	O.E.64	2021/08/30	exchange with works council	13 min	2
[E]	O.E.65	2021/08/30	exchange with technical organization unit	40 min	7
[E]	O.E.66	2021/08/30	agile team meeting	11 min	2
[E]	O.E.67	2021/08/30	exchange with management	57 min	16
[E]	O.E.68	2021/08/31	daily standup meeting	13 min	6
[E]	O.E.69	2021/08/31	exchange with works council	19 min	2
[E]	O.E.70	2021/08/31	exchange with transformation management	55 min	15
[E]	O.E.71	2021/08/31	exchange with management	62 min	17
[E]	O.E.72	2021/08/31	exchange with technical organization unit	37 min	2
[E]	O.E.73	2021/08/31	Scrum master meeting	30 min	2
[E]	O.E.74	2021/08/31	agile team meeting	18 min	5
[E]	O.E.75	2021/08/31	exchange within the agile project team	58 min	18
[E]	O.E.76	2021/08/31	exchange within the agile project team	69 min	16
[E]	O.E.77	2021/09/01	daily standup meeting	15 min	5
[E]	O.E.78	2021/09/01	rehearsal of sprint review	89 min	36
[E]	O.E.79	2021/09/02	daily standup meeting	15 min	2
[E]	O.E.80	2021/09/02	sprint review	92 min	38
[E]	O.E.81	2021/09/02	sprint retrospective	90 min	25
[E]	O.E.82	2021/09/02	sprint planning	165 min	50
[E]	O.E.83	2021/09/03	daily standup meeting	15 min	9
[E]	O.E.84	2021/09/03	exchange within the agile project team	23 min	3
[E]	O.E.85	2021/09/03	exchange with technical organization unit	57 min	18
[E]	O.E.86	2021/09/03	agile project team meeting	14 min	4
[E]	O.E.87	2021/09/03	exchange within the agile project team	25 min	13
[E]	O.E.88	2021/10/26	project outcome release event	240 min	11
[E]	O.E.89	2021/10/27	project outcome release event	240 min	9
[E]	O.E.90	2021/11/04	final project retrospective	90 min	22

Table 18: First-order concepts and second-order themes for tensions

First-order concepts:	Second-order themes:
▪ capacities of non-agile units fully utilized by planned tasks ▪ capacity shortages due to staff secondment and absences ▪ high employee specialization prevents staff secondments ▪ insufficient staffing for concurrent resource demands ▪ lack of financial resources for completing project tasks ▪ limitations of financial resources ▪ limited availability of seconded staff ▪ staffing gaps in critical roles ▪ unwillingness of non-agile units to second staff to agile units	capacity and resource shortages
▪ fear of losing status or authority because of agile practices ▪ fear of negative career prospects when transforming to agile practices ▪ resistance against fast changes to interdisciplinary team structures ▪ resistance against speeding up delivery processes ▪ skepticism about the value and benefits of agile practices ▪ resistance against speeding up delivery processes ▪ unwillingness of non-agile units to accept agile planning approaches	resistance against agile practices
▪ friction during project-to-line transition ▪ resistance against abandoning agile practices ▪ resistance against project outcomes ▪ unwillingness of employees to abandon plan-based practices	general change resistance
▪ ambiguity during transitional phases after employee turnover ▪ hierarchical structures prevent effective collaboration within the agile unit ▪ inefficient collaboration due to excessive and unfocused meetings ▪ newly formed teams lack familiarity and cohesion ▪ uneven speaking time distribution in agile routine events	intra-team collaboration inefficiencies
▪ enforced collaboration despite perceived decrease in interdependence relevance ▪ inefficiency due to inconsistent and unclear communication ▪ initial strong stakeholder involvement fails due to poor collaboration ▪ lack early and frequent stakeholder involvement ▪ limited cross-functionality within the organization ▪ restricted cross-team collaboration due to focus on isolated agile units ▪ top-down staff assignment undermines effective collaboration in the agile unit	cross-team collaboration inefficiencies
▪ collaborative conflicts disrupt steering committee meetings ▪ conflicts over preserving egalitarian team dynamics ▪ escalating conflicts due to power struggles ▪ overreach of leadership into team-level agile ceremonies ▪ shared responsibility for the project creates conflicts	power and hierarchy conflicts in collaborative work
▪ excluding stakeholders aggravates unwillingness to cooperate ▪ interpersonal conflicts between key actors ▪ interpersonal conflicts when secondments are revoked ▪ offensive communication style reduces willingness to collaborate	relational conflicts in collaborative work

First-order concepts:	Second-order themes:
▪ agile unit is scapegoated for problems originating in non-agile units ▪ conflicts arising from unmet expectations of non-agile end user units ▪ escalating conflicts due to the miscommunication of project outcomes ▪ different understandings of the adaptation of agile practices ▪ disagreement over the appropriateness of agile methods for certain projects ▪ stigmatization of non-agile work practices ▪ strong disagreements concerning the benefits and weaknesses of agile practices	resistance-induced conflicts in collaborative work
▪ collaborative conflict fueled by unequal rule enforcement across projects ▪ collaborative conflicts due to different levels of formalization ▪ collaborative conflicts due to different release processes ▪ collaborative conflicts due to general processual differences ▪ conflict over uncoordinated planning processes ▪ low agile process standardization creates friction with non-agile units ▪ strong disagreements about optimal planning approaches ▪ specialist disagreement over implementation approach	processual conflicts in collaborative work
▪ intensive discussions to align conflicting expectations regarding release processes	**communication effort to negotiate new temporal patterns**
▪ challenging disengagement of project stakeholders ▪ high communication effort due to ideological debates over agility ▪ high conviction effort to make non-agile units accept project solutions ▪ high conviction effort to make non-agile units accept incremental solutions ▪ intensive communication to agree on priorities ▪ intensive communication to convince product owner of task importance ▪ intensive communication to align understanding of agile practices ▪ negotiations to align on the degree of documentation within agile projects	communication effort to reconcile conflicts and disagreements
▪ high communication effort because of hierarchical communication channels ▪ persistent stakeholder communication without formal mandate	structure-induced communication effort
▪ high communication effort due to excessive number of meetings ▪ increased communication effort in formalized agile processes ▪ prolonged and unstructured meetings due to accumulated problems ▪ prolonged meetings due to high team size	meeting-induced communication overload
▪ conflicting interests between agile units and works council ▪ conflicting demands for standardization and customized solutions ▪ conflicting stakeholder interests complicate development of tailored solutions ▪ conflicting stakeholder interests regarding project goal and methodology ▪ management interests conflict with non-agile stakeholders	conflicting stakeholder interests

First-order concepts:	Second-order themes:
▪ agile unit dependency from non-agile units' input provision ▪ coordination intensity with stakeholders changes over time ▪ coordination needs to avoid adverse effects on software landscape ▪ high coordination requirements because of regulation, security, and compliance ▪ need for ongoing coordination to keep insurance line knowledge updated ▪ need to coordinate with management stakeholders ▪ need to request non-agile employees due to limited in-team capabilities ▪ system interdependencies require coordinated action	limited operational independence
▪ coordination effort for bridging between agile and non-agile temporal structures ▪ high coordination effort to execute late changes ▪ high coordination effort to execute releases at high frequency ▪ high coordination effort to ensure the harmonization of different release cycles ▪ need for synchronization of timelines and tasks with interdependent units	**temporal coordination effort**
▪ challenge of finding meeting slots with key stakeholders ▪ concurrent tasks and responsibilities increase coordination effort ▪ high coordination effort caused by unstructured stakeholder participation ▪ high coordination effort due to the complexity of major releases ▪ multiple project stakeholders increase coordination effort ▪ high need for coordination because of team specialization ▪ negative impact of large team size on coordination complexity ▪ reduced scope of tasks because of high anticipated coordination needs	complexity-driven coordination effort
▪ coordination capacity exceeds actual coordination needs ▪ coordination effort due to the need for data provisioning ▪ effort for reconciling conflicting processes and guidelines ▪ high coordination effort due to excessive task granularity ▪ high coordination effort to replace missing secondments ▪ high coordination needs to reconcile conflicts between actors ▪ proactive follow-up coordinated needed to enforce resource provisioning	procedural coordination effort
▪ budgeting constraints cause delays ▪ delays caused by missing feedback during reviews ▪ postponed delivery of business value due to insufficient technological readiness	delayed delivery due to resource constraints
▪ delayed project completion ▪ non-compliance with project milestones causes delays ▪ non-compliance with testing milestones causes quality deficits that result in delays ▪ postponement of software functionalities to later releases ▪ uncoordinated release timelines cause delays in solution delivery ▪ delays in completing outcome implementation due to extended formal processes ▪ quality concerns result in postponements to later releases	**delayed delivery due to differences in temporal structures**
▪ demotivation from delayed provision of feedback ▪ project teams composed of members with long tenure are less motivated	decreasing motivation

First-order concepts:	Second-order themes:
▪ emotional strain from maintaining confidentiality ▪ emotional strain from perceived inadequacy in fulfilling dual roles ▪ strain from loyalty conflicts between team and organization ▪ fear of losing credibility due to underutilization of requested resources ▪ ongoing self-doubts about own capability of making autonomous decisions ▪ strain from mediating between conflicting stakeholder interests ▪ strong emotional reactions to the communication of project solutions	emotional strain
▪ end user dissatisfaction because of a lack of perfection of agile solutions ▪ end-user dissatisfaction resulting from delays in solution delivery ▪ limited planning ability creates dissatisfaction in non-agile units	**time-related dissatisfaction**
▪ dissatisfaction because of unpaid tasks and obligations ▪ dissatisfaction resulting from unmet expectations around roles and competencies	job-related dissatisfaction
▪ dissatisfaction resulting from unmet expectations regarding provided solutions ▪ end-user dissatisfaction because of software errors ▪ intransparency of project outcomes causes dissatisfaction among end users ▪ lack of alignment of agile solutions with corporate goals	outcome-related dissatisfaction
▪ costs for conducting the agile project are considered too high ▪ high overhead costs due to coordination and leadership demands of agile practices ▪ trial-and-error has high costs at scale	high costs of applying agile practices
▪ high costs for conducting additional releases ▪ high costs for creating additional testing capacity	**high costs for accelerating release rhythms**
▪ high workload due to task assignments during absences ▪ high workload due to capacity shortages ▪ pressure to ensure sufficient workload for assigned staff	high workload from resource and capacity constraints
▪ high workloads for achieving high degree of perfection ▪ late testing causes workload peaks ▪ overhead workload to execute additional releases ▪ obligation of concurrent use of agile and traditional release processes	**high workload from quality versus speed tensions**
▪ high workload because of a lack of task prioritization and planning ▪ schedule disruptions cause workload peaks ▪ increase of workload due to short-term task demands ▪ obligation of concurrent use of agile and traditional planning processes	**high workload due to planning differences**
▪ high workloads resulting from mandatory tasks and deadline pressure ▪ effortful reversal of software development in linear development processes ▪ late task handover increases workloads ▪ rework due to the need of repeating accomplished process steps in linear development processes	**high workload due to temporal constraints**

First-order concepts:	Second-order themes:
▪ high workloads because of concurrent line and project roles ▪ role overload due to combined strategic and operational responsibilities ▪ role overload due to low experiences with agile practices ▪ overload due to multiple story ownership ▪ agile self-organization increases workloads of agile team members	role overload
▪ high workload because projects do not consider long-term operational burdens ▪ high workloads from setting high project goals ▪ overallocation of tasks in managerial planning ▪ overcommitment of agile team during task planning	high planning-induced workload
▪ effort required to onboard new team members ▪ emergency support of other teams increases workload peaks ▪ high workload because of concurrent stakeholder demands ▪ increased workload to substitute input from non-agile units ▪ pressure to assume additional workload to respond to emergent tasks ▪ high testing effort due to system dependencies	high workload from external demands and dependencies
▪ high effort for documentation ▪ high testing effort due to system dependencies ▪ high workloads from daily operative tasks ▪ unexpected workloads of legacy system replacements	high operational workload
▪ agile practices do not match with organizational hierarchies ▪ agile practices used when plan-based practices are more suitable ▪ agile practices are blindly used without evaluating its appropriateness ▪ hybridization of agile and traditional roles leading to methodological inconsistencies	inappropriate application of agile practices
▪ agile units do not use the capacities they applied for ▪ inefficiency of full secondment of non-agile employees ▪ synergies between units are not leveraged ▪ team member capacities are under-utilized	capacity under-utilization
▪ double work prescribed by pre-established processes ▪ inefficient double work due to lack of transparency ▪ early involvement risks wasted effort due to new developments ▪ late involvement risks wasted effort due to a lack of knowledge ▪ effort benefit mismatches in organizing stakeholder communication events ▪ effort-benefit mismatches in participating in agile routine events ▪ extensive review preparation driven by fear of leadership judgment	engagement in inefficient, non-value-adding work
▪ effort-benefit mismatches from high planning efforts ▪ inefficiencies because of high degrees of perfection ▪ inefficient parallel work due to linear development processes ▪ preparation of unused outputs due to asynchronous planning ▪ preparation of unused outputs due to the loss of plan validity	**non-value-adding effort from temporal misalignment**
▪ uneven distribution of capacity demands over time ▪ waiting times cause inefficiencies in agile capacity utilization	**uneven capacity utilization over time**

First-order concepts:	Second-order themes:
▪ absence of feedback and error cultures ▪ lack of proactive learning behavior ▪ limited awareness of sprint deadlines in the team ▪ limited ownership and self-responsibility among employees ▪ transition challenges to empowering agile teams	cultural and behavioral barriers to agile
▪ adoption of agile practices without methodological understanding ▪ insufficient agile literacy among key project roles ▪ insufficient agile literacy among stakeholders ▪ limited appreciation and understanding of Scrum master role ▪ limited understanding of the value and purpose of agile practices ▪ non-agile end users formulate tasks based on descriptions of alleged outcomes instead of their needs ▪ non-agile units do not understand agile role structures ▪ superficial understanding of agile at management level ▪ underestimation of the strictness and rigidity of agile practices	deficits in agile methodological understanding
▪ Scrum master needs support because of limited experience ▪ missing definition of agile practices and agile processes ▪ limited recognition of agile team challenges due to a lack of agile understanding	agile role and process guidance deficiencies
▪ communication failure around nature and structure of the agile approach ▪ confidentiality constraints limit open discussion of project details ▪ hesitancy to speak up and communicate openly in reviews ▪ insufficient communication about allowed process deviations ▪ poor communication of feedback during sprint reviews ▪ restriction of informal communication due to remote work	lack of communication
▪ planning cycles with divergent durations create coordination gaps ▪ lack of alignment with non-agile deadlines and milestones ▪ uncertainty about the effectiveness of coordination	**coordination gaps due to asynchronous planning**
▪ coordination capacity does not meet actual coordination needs ▪ lack of coordination caused by insufficient capacity to manage change notifications ▪ limited project involvement reduces coordination ▪ many meetings limit opportunities for coordination	insufficient capacity for effective coordination
▪ coordination gaps due to a lack of leadership direction ▪ coordination is too strongly dependent on brokerage roles ▪ key agile roles are excluded from management coordination ▪ uncoordinated task assignment to key stakeholders	leadership and role-induced coordination gaps
▪ decentralized team structures inhibit coordination ▪ exclusive focus on the needs of the agile unit ▪ coordination is inhibited by a lack of personal relationship and network ▪ lack of cross-team exchange ▪ lack of transparency about project-relevant information	structural, relational, and informational coordination barriers
▪ autonomous agile decision-making bears risk that regulatory demands are not met ▪ formal rules inhibit agile decision-making empowerment ▪ hierarchical constraints limit agile decision-making empowerment ▪ product owner is excluded from important decision-making ▪ product owner lacks product responsibility	regulation- and hierarchy-induced limitations of autonomy

First-order concepts:	Second-order themes:
▪ interdependencies limit decision-making empowerment in agile units ▪ limited decision-making authority of individual team members ▪ project-induced loss of decision-making autonomy in non-agile units	collaboration-induced limitations of autonomy
▪ anticipated lack of feedback because of decreasing stakeholder participation ▪ biased and unrepresentative feedback due to selective participant involvement ▪ invitation of wrong stakeholders limits the quality of feedback ▪ low stakeholder engagement limits the quantity of feedback received	lack of feedback from low or unrepresentative stakeholder participation
▪ hesitancy to give feedback to avoid blaming the agile unit for mistakes ▪ hesitancy to speak up and communicate openly in reviews ▪ stakeholders need more time than reviews allow to give meaningful feedback	structural and cultural barriers to open feedback
▪ agile routine events limit focus on relevant operative tasks ▪ big team size limits the focus of the agile unit ▪ large meetings with too many attendants limit focus ▪ management participation limits focus in agile routine events ▪ oversized planning meetings and unstructured collaboration limit efficiency	lack of focus in agile routine events
▪ employees in agile units cannot focus on their core tasks due to multitasking ▪ employees juggle project and line tasks due to incomplete secondment ▪ employees in non-agile units cannot focus on one task ▪ involvement in several projects limits focus on agile project tasks ▪ many project requirements limit focus on important tasks	erosion of focus due to multitasking
▪ lack of product owner role causes a lack of focus on relevant tasks ▪ proximity to non-agile home units limit focus of seconded employees ▪ rapid redeployment of product owners limits focus on implementation success	role-induced lack of focus
▪ initiating temporal structuring practices without managerial support ▪ lack of managerial attention during project-to-line transitions ▪ lack of top-down commitment to introducing agile practices ▪ lack of top-down commitment to quick agile decision-making ▪ management has no particular interest in agile practices	lack of management support for agile practices
▪ inadequate team support by Scrum masters ▪ inaction of agile leadership roles during critical issues	agile unit leadership failure
▪ lack of product owner role leads to insufficient task prioritization ▪ missing or inefficient processes for prioritizing stakeholder demands ▪ missing task prioritization for more than one sprint	lack of prioritization
▪ disclosure of relevant information to the agile unit ▪ lack of transparency about forthcoming support of the agile team ▪ selective disclosure of tasks and projects because of competitive dynamics and rivalry between agile and non-agile units ▪ no communication of project goals	strategic and selective information withholding

First-order concepts:	Second-order themes:
▪ complex communication channels are intransparent ▪ no transparency about contact persons in agile units ▪ lack of clarity over information access and timing	intransparent information access
▪ intransparent task progress within the agile unit ▪ lack of awareness of cross-team dependencies ▪ lack of transparency about project outcomes ▪ lack of transparency about project-relevant information ▪ non-agile timelines and deadlines are not transparent to the agile teams ▪ the use of multiple planning tools reduces task transparency ▪ workloads are not transparent to the agile team	lack of transparency in task and project information
▪ resistance to revise prior decisions limits agile flexibility ▪ resource and tasks commitments over long planning horizons reduce flexibility ▪ rigid long-term plans limit flexibility	**limited flexibility due to rigid planning**
▪ long lead times for software releases prevent flexible changes of software ▪ low release frequency limits flexibility ▪ rigid dates and deadlines limit the flexibility of agile units ▪ workload peaks and rigid testing plans reduce flexibility	**limited flexibility due to deadline rigidities**
▪ hierarchical barriers prevent flexible replanning ▪ high specialization prevents flexible change of staff between tasks ▪ rigid team boundaries hinder flexible resource sharing ▪ political pressure limits flexibility of the agile units ▪ rigid processes to ensure compliance with regulations limits flexibility	limited flexibility due to structural barriers
▪ capacity restrictions limit the solutions that can be implemented ▪ coordination with interdependent non-agile units limits flexibility ▪ fixed-scoped tasks of non-agile units limit the flexibility of agile units ▪ flexibility is constrained by legacy systems ▪ manual replanning effort limits flexibility	limited flexibility due to resource and dependency constraints
▪ long tenure fosters process rigidity and blind spots for innovation ▪ strong hierarchical involvement restricts innovativeness	limited innovativeness
▪ ad-hoc project setup limits planning ability ▪ agile approach used to reject long-term planning ▪ failure to plan activities early limits agile team effectiveness ▪ ineffective resource planning due to lack of temporal distribution of workload ▪ misalignment on the scope of required planning ▪ oversimplified effort estimation invalidates planning assumptions ▪ plans are based on rough estimates with limited precision ▪ plans do not adequately consider interdependencies ▪ short, agile planning cycles limit the planning ability of non-agile units	**limited planning ability due to misaligned planning practices**

First-order concepts:	Second-order themes:
▪ assumptions about workloads may not hold true ▪ declining task relevance due to time-sensitive priorities ▪ unexpected high workloads create schedule disruptions ▪ frequent changes limit the validity of long-term plans ▪ interferences between project and release timelines ▪ late communication of agile units disrupt non-agile planning ability ▪ long-term planning has low validity due to limited foresight ▪ short-term task demands disrupt pre-established plans ▪ delivery dates from non-agile units lack dependability ▪ unpredictability of process dependencies limits agile planning ability ▪ schedule disruptions decrease the validity of long-term plans task uncertainty limits the validity of long-term plans	**schedule disruptions from temporal misalignments**
▪ lack of monitoring reduces implementation success ▪ limited knowledge transfer and capacity building among end users ▪ non-agile end users do not accept solutions of the agile unit	limited project success
▪ fixed deadlines limit iterative practices and agile responsiveness ▪ inertia in responding to critical schedule disruptions ▪ lack of iterative practices limits responsiveness of non-agile units ▪ long lead times for releasing software solutions ▪ responsiveness of non-agile units is limited due to low release frequencies ▪ limited responsiveness under deadline pressure ▪ low responsiveness to short-term regulatory changes ▪ deadline pressure restricts stakeholder involvement ▪ innovative solutions are rejected under deadline pressure	**temporal limitations of responsiveness**
▪ attempts to standardize software limit individuality of solutions ▪ insufficient timely information flows limit responsiveness ▪ limited ad-hoc responsiveness in sprint-focused teams ▪ low responsiveness in non-agile units without customer contact ▪ restriction to asynchronous communication limits responsiveness	structural and processual limitations of responsiveness
▪ blocked test environments limit the number of additional releases ▪ complex regulatory requirements limit the number of releases ▪ formal approval processes slow down delivery speed ▪ high costs and workloads for conducting releases slow down release rhythms ▪ testing obligations slow down release speed ▪ infrequent software releases slow down delivery	**limited speed due to low-frequency releases**
▪ end user resistance against project outcomes limit delivery speed ▪ formal approval processes slow down delivery speed ▪ high operative workloads slow down delivery speed ▪ late-stage feedback and acceptance of end users prevent timely delivery ▪ resistance against agile project outcomes slows down delivery speed ▪ testing obligations slow down time-to-market	stakeholder and task-induced limitations of speed
▪ conflicting project management instructions reduce performance ▪ low performance at the start of the project ▪ part-time allocation of employees decreases project performance ▪ reduced operative performance due to capacity shortages ▪ risk of losing functional experience during long project assignments ▪ time-consuming discussions about the use of agile practices lower performance	low project or team performance

First-order concepts:	Second-order themes:
▪ ineffective stakeholder engagement in early phases without tangible deliverables ▪ lack of confidentiality of stakeholder feedback limits engagement ▪ limited appreciation of stakeholder feedback reduces stakeholder engagement	ineffective use of stakeholder feedback
▪ absence of key stakeholders during agile routine events ▪ channels for stakeholder participation are not used ▪ limited willingness to engage in the agile project team ▪ uncertainty about the timing of intensified stakeholder engagement ▪ voluntary engagement with the agile team deprioritized under workload stress	low stakeholder engagement
▪ emergency support of other agile units causes delays ▪ excessive coordination needs prevent the timely completion of tasks ▪ higher priority of other tasks prevents timely completion of sprint tasks ▪ overloading of sprints results in non-completion of tasks ▪ rework takes time and causes delays ▪ sequential multi-team dependencies create bottlenecks in task completion ▪ tasks are not completed in one sprint and transferred to the next	delayed task completion due to workload and dependency tensions
▪ high procedural demands for software changes prevent task completion ▪ uncompleted tasks due to outdated planning heuristics ▪ waiting times lead to non-completion of tasks	**delayed task completion due to planning tensions**
▪ extensive coordination required in planning processes ▪ extensive workload related to planning processes ▪ high coordination complexity increases planning efforts ▪ high effort for conducting recurring and overlapping planning processes ▪ obligation of concurrent use of agile and traditional planning processes ▪ re-emerging planning efforts because of short-term staff secondments	high efforts due to planning complexity
▪ high upfront planning effort to anticipate all eventualities of task execution ▪ high upfront planning effort to create long-term plans	high effort for creating long-term plans
▪ conflicting task prioritization prevents non-agile units from delivering input ▪ corporate-level conflicts over project prioritizations ▪ distributed decision-making causing prioritization conflicts ▪ different prioritization of quality and speed in agile and non-agile units ▪ missing representations of agile units in prioritization processes ▪ prioritization of following long-term plans over responding to changes ▪ unit-level conflicts over project prioritizations	cross-unit prioritization conflicts
▪ autonomous agile decision-making creates risk that regulatory requirements are not met ▪ errors limit stability of the organizational software landscape ▪ errors limit software performance ▪ high complexity of software releases jeopardizes software quality ▪ quality concerns because of limited coordination with interdependent units ▪ quality risks because of limited coordination with interdependent units ▪ uncertainty about future data consistency	quality concerns from coordination gaps

First-order concepts:	Second-order themes:
▪ anticipation that expectations are not fulfilled ▪ end-user quality concerns due to perceived increase in workload ▪ doubts about the solutions because of the agile unit's perceived competence gap ▪ doubts about the solution because of limited representation of non-agile units ▪ solutions of the agile unit are perceived worse than own solutions ▪ quality concerns due to missing secondments of experts ▪ quality concerns due to missing documentation ▪ quality concerns due to suboptimal project sequencing ▪ quality risks from overly customized solutions ▪ intransparency of project outcomes raises quality concerns	competence- and solution-related quality concerns
▪ anticipation of quality risks in future caused by non-adherence to deadlines ▪ errors cannot be reliably identified and corrected when deadlines are not adhered ▪ lack of documentation and knowledge transfer under deadline pressure ▪ ongoing self-doubt whether the own decisions are correct under deadline pressure ▪ task overload under deadline pressure causes errors	**deadline-induced quality concerns**
▪ rapid releases endanger compliance with regulatory requirements ▪ defects in software development prevent quick delivery ▪ errors cannot reliably be identified and corrected in fast release rhythms ▪ external delivery pressure leads to premature go-live ▪ fast releases in high-impact dependencies give rise to quality risks ▪ fear of communicating results too early due to pending approval ▪ agile unit refuses to develop plausibility checks to make fast progress which jeopardizes software stability ▪ non-agile units doubt solution quality because of a lack of perfection	**speed-induced quality concerns**
▪ ambiguities due to the simultaneous use of agile and traditional roles ▪ blurry boundaries between various roles in agile units ▪ conflicting requirements of dual roles ▪ inconsistent understanding of roles between actors ▪ ineffective project decision-making because of unclear responsibilities ▪ lack of definition of additional roles in agile units ▪ lack of understanding of one's own role in the agile unit	ambiguity in role definitions
▪ different expectations about the organizational integration of roles ▪ loss of sense of belonging to non-agile line units ▪ limited product owner role fulfillment ▪ unfulfilled product owner role in the agile team	role execution and integration challenges
▪ constant formation of new agile teams slows down working pace ▪ long lead times for team to grow together slows down pace at the beginning ▪ limited pace in non-agile units because of a lack of focus ▪ slow decision-making processes because of hierarchical involvement ▪ slow decision-making processes because of the lack of team hierarchies ▪ slow decision-making processes due to the non-agile involvement ▪ slow pace due to insufficient technological readiness	slow pace due to organizational and structural barriers

First-order concepts:	Second-order themes:
▪ intense coordination requirements limit the pace of the agile unit ▪ limited perceived pace due to extensive communication needs ▪ limited speed due to the intransparency of communication channels ▪ long duration of communication processes slows down response pace ▪ slow pace because of the steady need to resolve conflicts ▪ slow project progress due to excessive coordination loops ▪ persistent resistance against agile practices undermines agile progress ▪ time invested in communication limits the pace of agile units	slow pace due to intensive coordination and communication
▪ relatively more urgent projects cause a systematic deprioritization ▪ deprioritization conflict due to competing internal demands ▪ deprioritization of functional requirements ▪ deprioritization due to resource constraints and higher-priority projects ▪ rejection of emergent tasks from selected non-agile units ▪ cancellation of interviews with the corporate client advisory unit ▪ unbalanced prioritization due to specific functional backgrounds	systematic deprioritization
▪ agile units do not comply with non-agile deadlines ▪ anticipation of future delays caused by non-adherence of deadlines ▪ lack of deployment synchrony increases uncertainty about timely delivery ▪ management is concerned about timely project completion ▪ non-compliance with release processes creates high risk of delays ▪ open-ended agile development processes cause uncertainty about timely finalization ▪ open-ended agile development processes cause uncertainty about workloads in non-agile units ▪ short, agile planning cycles cause uncertainty about solution delivery	**uncertainty about timely delivery**
▪ early communication of project results might fuel uncertainty of non-agile end users ▪ late communication of agile project outcomes creates uncertainty ▪ uncertainty that stakeholder involvement slows down approval and implementation processes	**uncertainty from stakeholder communication dynamics**
▪ early notification of resource needs for obtaining required resources ▪ need to renegotiate resources creates uncertainty in resource commitments ▪ resource access contingent on synchronized planning cycles ▪ shifting priorities create uncertainty in resource allocation ▪ short-term unavailability of staff in non-agile units ▪ uncertainty in planning resource commitments ▪ underestimation of the duration of implementation processes ▪ underestimation of workloads because of a lack of understanding of operative processes	**uncertainty about resource allocation**
▪ expectation that agile teams produce results more quickly than plan-based teams ▪ high expectations regarding project outcomes ▪ high stakeholder expectations fueled by optimistic promises ▪ underestimation of the duration of implementation processes ▪ underestimation of workloads because of a lack of understanding of operative processes	unrealistic expectations regarding solutions
▪ waiting for relevant project input because of unexpected workloads ▪ waiting for relevant project input due to multitasking obligations	waiting times due to resource scarcity

First-order concepts:	Second-order themes:
▪ loss of project momentum while waiting for available capacities ▪ late communication of agile units' resource needs creates waiting times ▪ deviations from standard planning cycles cause waiting times ▪ postponement of input delivery to later releases ▪ non-compliance with project milestones causes waiting times ▪ waiting for capacities that are bound by specific releases ▪ waiting for relevant input due to planning cycle differences ▪ waiting for relevant project input due to limited availability of non-agile employees	**waiting times due to planning misalignments**
▪ foundational technical work not completed at project start ▪ prevalence of slower processes with longer development times in non-agile units ▪ non-agile units' slow responses create waiting times ▪ waiting times due to conflicting interests ▪ waiting times due to hierarchical communication channels	waiting times due to organizational bottlenecks
▪ backgrounds of pre-established processes are not understood ▪ loss of knowledge after absences ▪ loss of knowledge due to frequent turnover in the agile project team	knowledge gaps and loss
▪ efficiency of processes depends on the presence of specific individuals ▪ errors in estimating the scope of software releases ▪ established monitoring processes do not work in agile practices ▪ inefficiencies when established processes are transformed agile ▪ processes are not clear to all actors involved ▪ rigid adherence to formal documentation in collaboration with agile practices	process inefficiencies

Table 19: First-order concepts and second-order themes for temporal practices

First-order concepts	Second-order themes
Tension: Coordination gaps due to temporal misalignment	
▪ constant requests for preliminary task prioritization	anticipatory coordination
Tension: Deadline-induced quality tensions	
▪ hierarchical enforcement to comply with conventional deadlines ▪ re-entrainment to non-agile deadlines after non-compliance caused adverse effects on software quality	entraining to non-agile deadlines
▪ case-by-case assessment of deadline deviations ▪ negotiation of deadline deviation with release management	negotiation over deadline deviations
Tension: Delayed task completion due to planning tensions	
▪ reprioritization of backlog tasks without interdependencies to circumvent waiting times	reprioritization of tasks
Tension: High workload from quality versus speed tensions	
▪ reactivation of unused testing environment to increase testing capacity ▪ intensive negotiation with non-agile units to accelerate release rhythms	negotiating new temporal patterns
▪ charging costs for conducting additional releases ▪ release of software restricted to major releases	entraining to non-agile release rhythms
Tension: High workload due to temporal constraints	
▪ reduction of the number of requirements to be implemented by the agile unit ▪ negotiation of the scope of requirements with management ▪ suspension of agile iterations in favor of sequential development ▪ postponement of tasks to subsequent software releases ▪ cancellation of interviews with non-agile units to explore their needs	work scope reductions
▪ recruitment of external staff for agile units ▪ flexible reassignment of staff to agile units with peak workloads ▪ gradual handover of task assignments to non-agile units	expansion of team size
▪ cancellation of agile routine events to increase temporal capacities for task fulfillment	liberation of temporal capacities
Tension: Limited flexibility due to deadline rigidities	
▪ suspension of agile iterations in favor of sequential development ▪ reduction of quality level of tasks ▪ priority-based planning allows to flexibly shift tasks within quarters ▪ postponement of tasks to subsequent software releases ▪ suspension of correction of specific software errors ▪ cancellation of interviews with non-agile units to explore their needs	work scope reductions
Tension: Limited flexibility due to deadline rigidities	
▪ rearrangement of task sequences based on evolving knowledge ▪ perseverance of sprint planning to adapt long-term plans as needed ▪ interpretation of long-term plans as non-binding	non-observance of long-term plans
▪ managerial establishment of three-month cycles for revising annual plans ▪ stipulation of two-month planning cycles for detailed planning	dictating new temporal patterns

First-order concepts	Second-order themes
Tension: Limited planning ability	
▪ organization and conduction of workshops to convey ideals of planning practices ▪ preparing roadmap plans for extended periods of time ▪ planning for resource needs over extended periods of time	entraining to non-agile planning cycles
▪ managerial permission to deviate from established deadlines of non-agile release processes ▪ case-by-case assessment od deadline deviations	negotiation over deadline deviations
Tension: Limited speed	
▪ provision of additional releases to accelerate release rhythms ▪ negotiation of additional release dates to accelerate release rhythms ▪ identification of time windows to conduct additional releases ▪ frequent discussion within the release management unit of how accelerated release rhythms may be enabled ▪ hierarchical support to negotiate new temporal patterns ▪ building a coalition to motivate the acceleration of release rhythms ▪ extending testing capacities to accelerate release rhythms	negotiating new temporal patterns
Tension: Schedule disruptions	
▪ adaptation of long-term plans in response to schedule disruptions ▪ reprioritization of tasks despite contradictions to long-term plans ▪ perseverance of sprint planning to adapt long-term plans as needed	non-observance of long-term plans
▪ managerial establishment of three-month cycles for revising annual plans	dictating new temporal patterns
Tension: Temporal coordination effort	
▪ bundling of communication efforts in employees that broker between agile and non-agile units ▪ assignment of responsibility for harmonizing practices to the Scrum master	assignment of brokering responsibilities to team roles
▪ introduction of traditional project lead roles into agile teams ▪ additional roles in agile units to cope with enhanced temporal coordination needs	expansion of agile role structure
Tension: Temporal limitations of responsiveness	
▪ managerial permission to deviate from established deadlines of non-agile release processes ▪ deviations from non-agile deadlines that structure release processes ▪ modification of work outcomes after the deadline has expired ▪ use of late change mechanisms to enable development after deadline expiration	non-compliance with non-agile deadlines
Tension: Uncertainty about timely delivery	
▪ re-entrainment to non-agile deadlines after non-compliance caused adverse effects on software quality ▪ setting deadlines to reduce uncertainty about delivery dates	entraining to non-agile deadlines
▪ negotiations of processes for deviating from established release deadlines ▪ case-by-case assessment of deadline deviations ▪ negotiation of deadline deviations with release management	negotiation over deadline deviations
▪ frequent communication to elicit potentials for deadline deviation ▪ monitoring and reminding deadline adherence during agile routine events ▪ information about upcoming deadlines during agile routine events	anticipatory coordination

First-order concepts	Second-order themes
Tension: Uncertainty from stakeholder communication dynamics	
▪ communication and handover of project outcomes after approval	entraining to non-agile delivery rhythms
Tension: Uncertainty of resource allocation	
▪ synchronization with managerial planning cycles ▪ preparing roadmap plans for extended periods of time ▪ planning for resource needs over extended periods of time	entraining to non-agile planning cycles
▪ substitution of the most relevant with most available staff	substituting uncertain resources
Tension: Uneven capacity allocation	
▪ stipulation of two-month planning cycles for detailed planning	dictating new temporal patterns
▪ communication of potential under-utilization of capacity	anticipatory coordination
Tension: Waiting times due to planning misalignments	
▪ reservation of more resources than actually needed for work	creation of slack resources
▪ preparation of input before the operational work of the agile unit begins ▪ pre-approval of data collection and testing instruments	advance preparation
▪ organization for autonomous actions in the agile unit setup ▪ staffing the agile unit with employees from non-agile units ▪ delegation of decision-making authority for outcome approval to the agile unit	integration of staff and responsibility
▪ improvising without required input from non-agile units	reduction of dependencies
▪ regular conventions of meetings to discuss the priorities of planned and unplanned tasks ▪ transparent communication to identify non-risky opportunities for task postponement ▪ early identification of coordination needs	anticipatory coordination
▪ voluntary assumption of short-term requests of agile units by non-agile units ▪ communication of urgency to persuade non-agile units to assume emergent tasks ▪ immediate response to agile units' needs despite full capacity utilization	assumption of additional tasks
▪ managerial reprioritization for flexible task changes in agile units ▪ reprioritization of backlog tasks without interdependencies to circumvent waiting times ▪ prioritization of emergent tasks of agile units by non-agile units ▪ hierarchical reprioritization of emergent and unplanned tasks ▪ high legitimacy of agile units induces prioritization of agile tasks in non-agile units	reprioritization of tasks