

# Agile Practices in Practice: A Theory of Agile Adoption and Process Evolution

Brendan Julian, James Noble, and Craig Anslow

Victoria University of Wellington, New Zealand  
(julianbren, kjx, craig)@ecs.vuw.ac.nz

**Abstract.** As teams and organisations make the difficult shift to agile ways of working, there has been relatively little investigation of how they adopt and use agile practices. To aid those teams looking to move to agile, we should examine how others have done so and what practical value they found. We studied teams which adopted agile practices across a spectrum from taking on a whole methodology at once, to adopting a couple of practices and then committing to continuous assessment and improvement of their ways of working. Experienced teams favoured agile-based, team-oriented practices suited to their particular needs over technical practices and plan-driven defined methodologies.

**Keywords:** agile adoption, agile software development, continuous improvement, grounded theory, industry applications

## 1 Introduction

Agile practices have had a marked impact on the technology industry around the world, supporting increased communication, quality, and productivity in teams and their products when implemented successfully [6, 27]. Yet adopting and effectively using agile ideas and practices still presents a significant challenge to teams and organisations [3, 13, 21, 22, 25]. There also seems to be a disconnect between extant literature discussing what agile is and the comparatively smaller amount of research into how agile is applied in industry [7]. In order to aid those teams looking to become agile, it is important to examine how others have already done so and to share their experiences with the wider agile community.

We sought to understand agile use and evolution in multiple teams around Wellington, New Zealand by analysing the changes they made to their processes over time. We wished to identify trends in how teams adapt and modify their process to fit their work, and to understand the success of those changes.

In this paper, we present a Grounded Theory study of the adoption and evolution of agile practices in several local organisations. We conducted semi-structured interviews with twenty-two agile practitioners in twenty organisations, each with varying levels of expertise across different roles and types of work. Based on this data, we present our theory on the motivations and means of agile adoption and the ongoing process of adapting and improving on agile practices to suit the specific needs of the teams and their work.

## 2 Methodology

### 2.1 Grounded Theory

Grounded Theory (GT) is a sociological research method proposed by Glaser and Strauss in 1965 [11] and formalised in 1967 [12]. Rather than verifying an existing theory, GT seeks to build a theory from data where little or no established theory exists. GT is a commonly used method for qualitative data analysis, including in the software development industry [1, 17, 18]. Glaser and Strauss's GT assumes the researcher is unfamiliar with the area of interest to aid discovery of a theory, so to account for our own prior knowledge we instead chose to adopt Charmaz's Constructivist Grounded Theory (CGT) variant [4]. CGT theories are co-constructed by the researcher and participants through investigation of the area of interest, rather than discovered. CGT promotes using prior knowledge and experience as a means of more effectively examining data that may require esoteric domain knowledge. As we already have experience with agile practices we considered CGT to be well suited to our study.

### 2.2 Data Collection

We conducted twenty-two interviews with participants from twenty different organisations, nineteen within the tech industry and one outside it. Interviews were 30 to 90 minutes in length and followed a semi-structured format. After each interview, the data was coded [19]. Codes were then used to inform further interviews and modify questions as part of constant comparative analysis [10]. Table 1 gives aliases and roles for the participants.

Among those interviewed in this study, most were working in small-mid sized teams, using some form of Scrum [28]. Few, if any, were using "pure" Scrum. A number were using some form of Lean such as Kanban [31], or a combination of the two, referred to as Scrumban [23]. We also observed some use of scaled agile frameworks such as Scrum of Scrums [32] and SAFe [24].

### 2.3 Theory Development

Through a process of constant comparison, we used previously gathered data to inform further data gathering, and used that new data to aid in interpreting previous data. We took the codes drawn from the interview notes and grouped them into general descriptive categories of observed practices and modifications via card sorting [33]. We further developed the theory by grouping cards by categories of changes and the driving factors behind them, focusing on identifying changes in participants' methodologies. For example, the initial code of "buy-in" developed further into concepts such as "willingness to make changes", "capacity for change" and "commitment" as we gathered more information. Our information gathering continued until we were confident we had reached theoretical saturation, the point at which little to no new information was being uncovered. Though we conducted twenty-two interviews, we began to reach saturation at

P#	Role	P#	Role
P1	Product Owner	P12	Project Coordinator and Logistics
P2	Scrum Master	P13	Facilitative Team Lead
P3	Team Lead	P14	Technical Business Analyst
P4	Intermediate Developer	P15	Consultant/Agile Coach
P5	Consultant	P16	Team Lead/Business Director
P6	Mobile Developer	P17	Lead Developer
P7	Full Stack Developer	P18	Management Consultant
P8	Release Manager	P19	Client Delivery Manager
P9	Scrum Master/Agile Coach	P20	Test Consultant/Account Manager
P10	Project Manager/Scrum Master	P21	Solutions Architect
P11	Scrum Master	P22	Digital Producer

**Table 1.** Interview participant aliases and roles

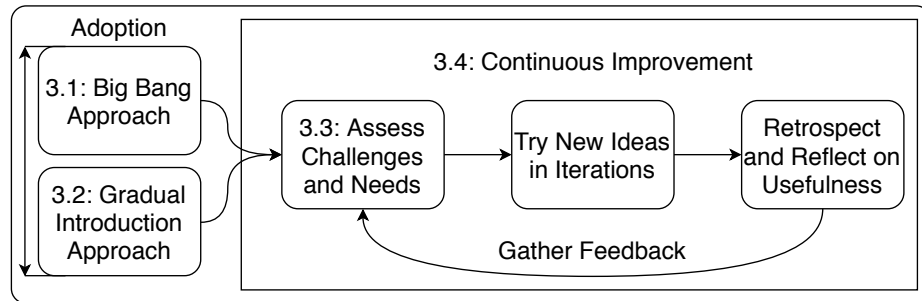
around eighteen interviews, past which interviewees were providing their own perspectives on concepts we had already previously identified. These changes allowed us insight into how and why teams chose to develop their practices over time, from adoption through to constant evolution.

Once data collection was complete and theory development was well underway, we began to compare our findings to other similar studies and ideas. In particular we noted several similarities between our discoveries and the proposed Agile Fluency Model by Larsen and Shore [2] with regards to teams assessing their ideal level of agility, as well as several interesting comparisons and contrasts to Hoda’s work on agile adoption [15]. We discuss this further in Related Work.

### 3 A Theory of Agile Process Evolution

We identified two general means of adopting agile practices, which we termed the “big bang” and “gradual” approaches. With a big bang [26], teams begin by adopting all of a by-the-book process to learn the practices and then begin to modify them. In contrast, teams following a gradual approach take on and integrate a couple of agile practices at a time alongside their existing, non-agile ones through a longer transition period. These approaches describe two ends of a spectrum wherein some teams performed a wholly big bang adoption across the whole organisation while others simply added stand-ups and issue tracking to their existing work, with many variations in between.

As teams developed expertise and understanding beyond adoption, they focused on continuous improvement, iterating on their process over time to ensure it fitted their needs. Adopting agile was not seen as a one-off event by interviewees, rather it requires a willingness to change and commitment to improving *how* a team works. We present a model in Figure 1 which visualises this process, demonstrating how teams which adopt agile through either approach tend to converge towards the goal of adaptability and continuous improvement, making adjustments as needed to ensure that their practices fit their work.



**Fig. 1.** Our model of Agile Process Evolution. Adoptions of agile methods by teams sit along the spectrum between a “big bang” or “gradual” approach. Following adoption, teams shift to a focus on continuous improvement, iteratively reflecting on and modifying their practices to better fit their current work.

### 3.1 The Big Bang Approach

Those in leadership or consultant roles (such as P1, P5, and P9) often shared the idea that newer teams should start with an established framework that has extensive literature and successful examples behind it.

*“That’s what I always recommend to people; start with Scrum and then evolve it”* - P1, Product Owner

In a big bang adoption, a team starts out following the same large set of general practices from day one. Even if there is an initial productivity hit while learning the practices, it was seen as being better to get it out of the way:

*“You need to start a project the way you mean to carry it on... In terms of that basic agile structure you want to have that in place but you want to be agile with that. It would evolve some change over time but you want to have an understanding of the method and roles as close to day one as possible.”* - P18, Management Consultant

The team should gain experience with all of the practices involved in a chosen framework, learn their value, and understand why they’re used, and then begin to assess which practices work well and which could be improved or replaced.

*“[Scrum] is the best one to start with, because it gives them a bit of discipline. It’s a framework they can hang on to and follow the rules, and it helps them get into an iterative, fast feedback style, and they can then learn to break the rules.”* - P5, Consultant

Proponents of big bang felt that the core practices of frameworks such as Scrum were valuable guidance for teams looking to adopt agile, even if they weren’t completely suited to their needs, and that attempting adoption without

some guidance was needlessly difficult. For example, one organisation (P10, typically Scrum-oriented) explicitly attempted to build an agile framework from the bottom up with no starting practices, rather adopting practices as they felt they would be needed. Quickly, they found that although they were able to identify issues with or gaps in the methodology built so far, the lack of structure made it difficult to identify how to best rectify the issues that arose:

*“We have tried putting nothing in place and taking on things as we need them. That hasn’t worked out so well. There was a lack of structure, structure is really important. Without that, things start to fall apart, things get lost. Some people think agile means you’ve got no structure, but really I think you need more rigour, discipline and structure to actually make it work.”* - P10, Project Manager/Scrum Master

### 3.2 The Gradual Adoption Approach

In contrast with the big bang approach, P2, P4, P8, P12, P13, P14, P15 and P19 spoke of integrating agile processes into an organisation’s existing non-agile, methodology, and gradually making changes to more agile methods. The first practices to be introduced were almost always iterations, stand-ups and retrospectives, accompanied by some form of task tracking and visualisation. For some teams (P8, P12, P13) those practices alone were well suited to their needs. Others (P2, P4, P14) continued to adopt practices to become fully agile teams, usually transitioning into something akin to Scrum or Kanban.

P19 shared their experience of a transition and felt that for an organisation lacking in agile expertise, attempting to take on too much in one go could be overwhelming, and instead the organisation looked at adopting and expanding their practices over time.

*“It was small bits at a time and I don’t think that was for any other reason than if you try and do it all in one hit it just becomes overwhelming and it fails. Where if you can do a bit, perfect it and that becomes the norm then move on to the next bit. First of all it was breaking into squads and then you were part of a squad of a business unit and then you do things like daily stand-ups and you just go through it.”* - P19, Client Delivery Manager

P15 related the adoption method back to a team’s capacity and needs. A team may not be capable or even have a use for fully adopting Scrum up front, but they may find great value in simply getting everyone in the same room every day for a stand-up to improve communication.

*“What I really try and help teams do is feel out a good sense of... what they’re capable of in any given time. An organisation might say “We’re gonna have a digital transformation, an agile transformation.” For me it’s figuring out what their purpose is, their overall purpose. You start to look at and observe tiny moves, suggesting small things. Will a stand up every day serve?”* - P15, Consultant/Agile Coach

Although fostering a commitment to the practices and driving change among team members was sometimes difficult, interviewees (P5, P6, P9, P10, P13, P14) noted that over time, and with greater experience, commitment increased. P13 noted that team members began organically making use of the task tracking board that they put in place without prompting:

*“The biggest put off is the idea of having to have everything sorted and we’re in an environment where things can change on the day so you’ve got to be flexible. So I wanted something that was flexible but also visual... Within a few weeks or so of us [maintaining a Kanban board] as a regular activity, my colleagues started putting up stuff as well, which was great because they were taking ownership of it” - P13, Team Lead*

Simply having progress be visible and traceable allowed team members to plan and organise their work more effectively, as the team organically embraced the board over time. Along with this tracking board, this team had adopted stand-ups and regular iterations in their non-tech, part-time environment. While a full agile transformation would be surplus to requirements, simply improving communication and visibility of information was incredibly valuable.

### 3.3 Challenges to Adoption

**Resistance.** In trying to drive a transformation, experienced individuals (P1, P2, P5, P6, P9, P10, P11, P14) noted it was often difficult to effect widespread change if either the people or environment were resistant to it. They recognised an “if it’s not broken, don’t fix it” mentality, that a process that isn’t actively hindering progress is good enough, even if there is an opportunity to improve. This attitude was seen as particularly prevalent in government departments, where existing processes are inflexible and change is looked on with suspicion.

*“The business don’t understand the DevOps and agile model here. Basically, we play the agile game totally for the teams...but then, the upper parts of the business haven’t really bought in. They’re coming around but they still see it as kind of a fad” - P2, Scrum Master*

It was felt (P2, P5, P9, P10, P13, P14) that some team members may need to see the effectiveness of a change in order to accept it, but making effective changes without some level of acceptance was difficult. Those same practitioners observed significant differences in attitude over time as newer adopters who had been previously resistant learned the effectiveness of a new way of working.

**Outside forces and regulatory compliance.** Those working in or with government departments or highly regulated organisations (P2, P8, P12, P14) showed frustration at trying to implement agile in an existing rigid framework.

*“One thing was a resourcing problem. They wanted development and testing done at the same time, which is fine, but they didn’t want to bring developers over to do testing because the pay gap is so different, but they also didn’t have enough testers” - P2, Scrum Master*

This team could fully adopt agile within themselves, but required a translation layer, ala “Agile Undercover” [16], to document and demonstrate work as required by the existing process. They could implement what agile practices they could (such as, iterations and issue tracking) within the existing framework while still fulfilling their obligations to their overseeing body.

*“Often, depending on the organisation, there’s the governance structure, and in government that can be quite heavy. At the moment there are two forums that things have to go through to be signed off [on top of the team’s iterative processes]” - P12, Project Coordinator.*

This solution was not considered perfect, but at least implementing more flexible practices helped improved team morale and effectiveness to some extent, even if such efforts were hamstrung by rigid existing processes.

**Needs of the work.** Interviewees focused on the importance of team practices over technical ones when it came to adopting agile ways of working. While technical elements such as pair-programming, test-driven development and so on were valued, the primary motivations were almost always about improving communication through practices like daily stand-ups and end-of-iteration retrospectives, and achieving a regular, fast delivery cadence that would allow them to frequently check their work against their clients’ expectations.

Regarding existing frameworks such as Scrum or Kanban, participants found them to be useful guidelines rather than rules. Every organisation has different needs and will find value in different practices. Frameworks should be taken as a valuable source of ideas rather than followed blindly.

Participants P5, P6, P9, P10, P11 and P15 considered it vital to understand the purpose and potential usefulness of a set of practices to make better decisions about their applicability to the team’s context. Through using the practices, teams gain experience and understanding of how applicable they are to their work and enable them to make informed decisions about how to modify the practices from there on. Gaining some experience and context would then allow teams to make more informed choices and have better discussions about what works well or what might need changing.

*“Then they can have a meaningful, evidence-based discussion about what they can achieve... We’re trying to build a self-organising team, capable of setting their own priorities. It’s quite amorphous, you’ve got to avoid just following the process, avoid a cargo cult. It does need a certain prescription[sic] result at the beginning to build up that muscle memory, this is why we have stand-ups, this is why we have retrospectives, sprint planning.” - P15, Consultant/Agile Coach*

The primary boons of conducting stand-ups and retrospectives were facilitating open communication and gathering feedback from every member of the team. Having everyone in the same room, at the same time, discussing the same project would allow everyone to stay informed and discuss their current work while retrospectives were used to reflect on work completed as well as the practices used to do so, or as P9 put it *“what went well, what didn’t go well, and what fell to shit.”* Several interviewees noted that over time, doing the same kind of stand-up or retrospective could become stale. They proposed that teams should seek to mix it up and try different styles in order to keep interest up, and ensure high quality and consistent feedback.

*“You’ve got to change it up now and then, otherwise people pretty much become bored with it and don’t participate as well”* - P9, Scrum Master/Agile Coach

Different teams would hold planning meetings at various scales, some with the immediate team each week all the way up to three-monthly department-wide planning sessions.

### 3.4 Focus on Continuous Improvement

Post-adoption, all teams moved into a overall practice of continuous improvement. Teams recognised that no set of practices is ever perfect and committed to regularly assessing how they work, often as part of retrospectives, in order to ensure their methods were well suited to their work and to improve and develop over time as a team.

**Nature of work determines the process.** More mature teams, those who have been employing agile successfully for a significant length of time (P1, P2, P3, P6, P9, P10, P11) tended to focus on ensuring their practices are adapted to the work, rather than trying to force their work to conform to their established practices.

*“If something like the definition of done or the definition of ready isn’t quite working for us, then maybe we look to improve that. It’s a constantly evolving process. It’s about making the process work for that team.”* - P6, Mobile Developer

For example, one large organisation (P9 and P11) operates an overall scaled agile framework on a four-week release train. A particular team was operating on a two week Scrum schedule, however, they switched to a four week Kanban schedule to align with the cadence of the rest of the organisation and account for irregular releases from their supplying vendor.

*“So Scrum doesn’t really suit us, because we’re getting stuff fed in, Kanban makes it easier to limit our work in progress so we can try and get through things as efficiently as we can”* - P9, Agile Coach



Almost all teams used estimation in some form when planning and prioritising their work, though specific methods differed. Several interviewees (P9, P21, P22) shared that they were moving away from their earlier hours-based estimation to more abstract points which allowed them to more accurately prioritise work relative to other work rather than attempting to time-box it.

*“Lately we have made a concerted effort to move away from hours-based estimations in favour of story points. Largely the motivation here was that hours-based estimates were proving difficult and often inaccurate or unreliable.”* - P22, Digital Producer

Changes can also be related to how a team performs a particular practice such as issue tracking. P9’s team found that using the digital solution Jira suited their purposes better than their previous on-paper solution as the team had distributed elements which didn’t have access to the physical board. While the physical solution was great for those who could interact with it, it was impossible for off-site members to actively contribute. On the other hand, P16’s organisation moved initially from a paper solution for tasks and BitBucket for tickets to Jira, but later switched back again:

*“It was originally in Jira but it made a lot more sense to take the stories out of Jira and give them their own identity. Jira is shit really, it’s fine for low level tickets and capturing screenshots but the other thing that we’ve used which is working really well is [a spreadsheet based Scrum board]”* - P16, Team Lead/Business Director

Especially when working with distributed teams or off-site clients, it was generally considered to be better to adapt the practices employed to account for difficulties rather than try to push through them with existing processes.

*“We get down to [vendor] at least twice a week to do refinement sessions. I’m not sure that’s a strict Scrum ritual, it’s a thing you’re supposed to do but Scrum doesn’t seem to guide you on how to do it, so we [visit vendor fortnightly] and spend an hour going through our draft stories.”*  
- P1, Product Owner

Organisations working in fields with significant regulatory components must factor in the time needed to complete the necessary testing and documentation into their release schedule.

*“Probably the biggest impact for us is regulatory. The documentation alone is massive... We break it down quite granularly... We can treat it a bit more agile... We can shuffle things around to keep ahead of the game”*  
- P8, Release Manager

Each of these examples demonstrates the pragmatism of the agile practitioners we interviewed, and their focus on doing what works for them.

Despite differences in methods of adoption, most interviewees (P1, P2, P3, P5, P9, P10, P11, P13, P14) felt it was important to constantly adapt their practices and continuously improve on how they performed their work.

*“You need to adapt... That’s what’s good about agile in general, ’cause the principle is never stay static. You’ve always gotta question the process and adapt the process to fit your environment, your people, your way of working.”* - P1, Product Owner

Change was a universally accepted fact of the working world, for better or worse, and a means of assessing shortcomings and rectifying them was considered highly valuable. Attitudes to changes varied quite widely, as did opinions on drivers for those changes. Through the interviews, participants were prompted on specific changes they made to their work-flow processes, though they were highly specific to the team and project at hand. Suffice to say time should always be taken, often in retrospectives, to examine aspects of the current process that may not be a good fit, and to not be afraid to try something else that may work better.

Most of the interviewees attested to using a form of Scrum or something Scrum-like (P1, P2, P3, P6, P7, P8, P9, P10, P11), but none were using pure, by the book Scrum. While teams made use of sprints, daily stand-ups, end-of-sprint retrospectives, start-of-sprint planning, and some means of issue tracking, they often deviated by dropping some practices or bringing in others.

*“The most important thing is the values and principles. We use Scrum largely, but the values and principles always help us find what isn’t working.”* - P10, Project Manager/Scrum Master

**Commitment to “being agile”.** The phrase “Don’t do agile, be agile” and the concept of agility are not exactly new [5, 14, 34], though it was explicitly brought up by nine of the twenty-two participants.

*“Agile is a mindset...Doing agile is only half of it, the rest is being agile”*  
- P6, Mobile Developer

The core idea here is to not simply follow the motions of the process for the sake of doing so but to employ the agile ideals to assess how you work, as much as the work you do. As P9, an agile coach, noted: *“If it’s not working, why the hell are you doing it?”* A practice doesn’t necessarily have to be bad to warrant changing or tweaking. Agile practices are just practices like any other and therefore are only as useful as their practitioners allow them to be. Simply performing the rituals will likely not produce successful results without a commitment to making proper use of them. As an example from P11, retrospectives are largely useless if those participating are unwilling or unable to provide constructive criticism of the work completed and the process used to do so.

*“Agile might potentially allow you to expose weaknesses in a better way. Agile might give you an option to break down [issues]. The guy that was a grumpy old bastard last week under Waterfall is still the same grumpy old bastard this week, you’ve just changed the methodology. [Agile] required a change in culture, people and mindset as much as a change in methodology”* - P11, Scrum Master

Effective implementation of agile ideas requires a commitment to make the most of the practices put in place, with a view to continuously improving on and modifying those practices to best suit the work at hand. Just as no team examined was using pure Scrum, rather some variation which they found better suited them, no one process will necessarily fit a given team on any given job.

## 4 Evaluation

**Glaser’s Criteria** Glaser provides four criteria for assessing the validity of a theory [9, 20], which is tied to the soundness of the research method and the data gathered by that method:

**Fit:** We adopted a CGT methodology in order to employ our previous understanding of what agile methodologies *are* to help investigate *how* they are employed in industry. The categories we developed were based on the data gathered, using our previous experience to interpret the data.

**Work:** By evidencing the theory, we have formed a clear link between interview results and our developed theory, which explains the data we have gathered, shows general patterns and provides new insights. Opinions shared by participants were generally very consistent. Even though they may adopt agile differently, they all focus on the concept of continuous improvement of their work and processes. The work criterion describes a theory that provides a solid explanation for identified phenomena. Owing to the consistency between our twenty-two participants, we consider this to be the case with our study.

**Relevance:** Our questions were predominantly aimed at how teams decide to adopt and adapt agile. The responses we gathered show how teams who adopt agile through different means trend towards the same end goal and find similar value in their different practices.

**Modifiability:** We developed our theory with each interview adding new perspectives and information, and it is open to further refinement.

### 4.1 Threats to Validity

There are two key limitations we encountered in this study:

**Scope:** We interviewed twenty-two participants from around Wellington, New Zealand. While this provides a good spread for the local context, data gathered is only representative of the opinions and experiences of our participants and may not necessarily represent other organisations or locales.

**Lack of Direct Observations:** While interviews are valuable, we believe this study could have benefited from observations to help understand how a team makes the changes they choose to make and why. Interviews can shed some light on this but are limited by what the interviewee can recall at the time.

## 5 Related Work

### 5.1 “Agile Fluency Model”, Larsen and Shore

The Agile Fluency Model [2] proposes how teams should look at the process of developing their agile practices. Citing the broad range of ideas on “how agile is supposed to be done” they examined agile methods in theory and practice. The model is comprised of five steps describing different levels of fluency; pre-agile, focusing, delivering, optimising, and strengthening.

One of the core elements of the Agile Fluency Model [2] is the idea that different levels of the model will suit some teams well and others may be a better fit for other teams. While different teams we studied had achieved differing levels of agility, they had all found value in their adopted practices.

Several teams (including P2, P8, P12 and P13) working within existing processes looking to employ iterations, stand-ups and retrospectives benefited from increased communication between team members and improved understanding of progress, which aligns well with the Focusing step.

More thoroughly agile teams (including P1, P6, P16 and P18) found increased fluidity to their work, with autonomous team practices improving flow. Prioritisation of work based on delivering the most immediate value became the main focus, embodying the Delivering step.

Mature teams (including P9, P10 and P11) identified that at agile practices had become second nature they were able to rapidly adapt to changes while continuously improving their practices. These teams were often either part of an overall transformation or leading it, which fits the Optimising step.

The ideas that agile frameworks are not one-size-fits-all, and that agile is not a silver bullet for any issue, would agree with the Agile Fluency Model. That said, to fit our observed teams into the model’s defined steps would be somewhat inaccurate. Rather, a spectrum from new teams to fluent teams better describes our data. Moving from one level to another in the Agile Fluency Model does not appear to be a generational leap but a series of small steps and occasional missteps that develops a team’s practices over time.

### 5.2 “Agile Transitions in Practice”, Hoda and Noble

“Becoming agile: a grounded theory of agile transitions in practice” [15] studied many organisations that have undertaken, or are undertaking adoption of an agile framework, and found that the teams do so in differing ways with varying levels of success. The authors acknowledge that the process of adopting agile is daunting and fraught with challenges surrounding existing organisational culture, people, processes and tools. As we have, Hoda and Noble identify a gap between existing guidance on adopting agile and useful advice for teams undertaking such a fundamental change to their way of working. Through a Grounded Theory study with 31 participants across 18 organisations in five countries, the authors developed a theory of agile adoption as a transition much like our model’s idea of gradual adoption. Rather than a simple series of steps, this paper posits

that transitioning to agile involves a complex network of concurrent changes in five dimensions; software development, team, management approach, reflective, and cultural practices. In our research we found the latter four to be most prevalent as our interviewees generally focused on team-oriented practices over tech-oriented ones.

While Hoda’s theory for agile transitions focuses on the process of adopting agile, the core concepts identified generally agree with both our findings on adoption and on continuous improvement. The paper implies that the organisations observed only employ what we have described as the gradual adoption approach and big bang adoptions were absent, even with a sample size larger than that of our study. Our observations agree that adoption is not a one-and-done affair: even among those following a big bang adoption, the initial introduction of agile ideas is not the end. Whether teams adopt processes over time or many at once, there is still a process of iteration and improvement required for team members and other staff to learn the processes involved and to grow their practices: the key difference is time scale.

Our theory of continuous improvement demonstrates that agile transitions do not stop once an adoption is considered complete, but instead teams continue to develop, regardless of their initial adoption approach. To ensure long term success in the use of agile, teams must develop and continuously improve on their processes both during and beyond the initial adoption.

### 5.3 Scrum and “Scrum-but”

Some participants (P1, P5, P9, P10) used the phrase “Scrum-but” to mean they use the core concepts and rituals of Scrum, but modify those which they have found to lack usefulness through experience. “Scrum-but” tends to carry negative connotations in literature, described as the “reasons why teams cant [sic] take full advantage of Scrum” [30] and typically refers to a number of anti-patterns that may crop up in Scrum [8].

Interviewees generally acknowledged that “Scrum roles, events, artefacts, and rules are immutable and although implementing only parts of Scrum is possible, the result is not Scrum per se [29].” Interviewees often found pure Scrum to be unsuitable for their environments and preferred to take Scrum values and core practices and adapt them to their needs. These teams would use the phrase to demonstrate that they recognise that what they’re using isn’t exactly Scrum, but rather an adaptation of the framework to the particulars of their environment. The prominence of “Scrum-but” demonstrates that though the core ideas of a framework such as Scrum are valuable, it should be treated as a framework, meant to be adapted to the purposes of a given team and that every practice may not be applicable to every team in the same way.

## 6 Conclusion

In this study we identified two methods for adopting agile methods in an organisation: big bang and gradual adoption.

With the big bang approach, teams adopt the entirety of an agile framework such as Scrum in order to learn the rituals and their value. Participants who had more extensive experience with agile techniques, such as P5, P9, P10, and P11 favoured this up-front style. While this approach often has an initial productivity hit, it gives teams a solid foundation to build upon from there on.

With the gradual adoption approach, teams seek to introduce specific agile practices one or a few at a time, typically beginning with iterations, stand-ups and/or retrospectives. Over time, more practices are introduced as the team feels they need them. Those interview participants, like P2, P8, P12, P13, and P14, who were somewhat new to agile themselves, preferred this approach as they felt it was more manageable to learn.

Adoption, moving from non-agile to agile methods, is not the end of the agile journey. We found that teams then embark on a process of continuous improvement where they iteratively improve upon their current set of agile practices, adding, removing, or modifying them as needs be to better suit the particular needs of their work and environment. In addition to assessing their work, teams also assess *how* they work, gathering feedback on the process they employ to understand what works well and what doesn't, with a view to addressing any shortcomings or taking advantage of opportunities for learning and improvement. As team members gain more experience with their practices, they become more receptive to further changes and more capable of contributing to and driving those changes.

More experienced teams typically favour autonomy in their work, flexibility in their methods, and actively assess and update them over time. They seek to ensure their practices are a good fit for their work, often picking up other ideas or making incremental improvements, rather than trying to force their work to adhere to their established practices. Teams appreciate that an established and well-honed set of practices helps align team members towards the same goals and keep everyone on track.

Future research could explore how the style of adoption (big bang vs gradual transition) impacts how team capabilities develop and how quickly, or take a closer look at how exactly teams go about assessing their needs, defining their goals, and then adapting their practices to suit their work.

## Acknowledgements

We thank the interviewees who participated in this study. We thank the reviewers for the insightful feedback and comments. This research was partially funded by Victoria University of Wellington, University Research Fund Application #216549 – User Centered Secure Programming.

## References

1. Adolph, S., Hall, W., Kruchten, P.: Using grounded theory to study the experience of software development. *Empirical Software Engineering* **16**(4), 487–513 (2011)
2. Agile Fluency Project LLC: The Agile Fluency Model (2018). URL <https://www.agilefluency.org/model.php>. Last accessed 18 October 2018
3. Biddle, R., Meier, A., Kropp, M., Anslow, C.: MyAgile: sociological and cultural effects of agile on teams and their members. In: *Proceedings of the International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*, pp. 73–76. ACM (2018)
4. Charmaz, K.: *Constructing Grounded Theory*. Sage (2014)
5. Cockburn, A.: *Agile Software Development*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA (2002)
6. Coram, M., Bohner, S.: The impact of agile methods on software project management. In: *Proceedings of the International Conference and Workshops on the Engineering of Computer-Based Systems (ECBS)*, pp. 363–370. IEEE (2005)
7. Dreesen, T., Schmid, T.: Do as you want or do as you are told? control vs. autonomy in agile software development teams. In: *Proceedings of the Hawaii International Conference on System Sciences (HICSS)*. IEEE (2018)
8. Eloranta, V.P., Koskimies, K., Mikkonen, T.: Exploring scrumbut: An empirical study of scrum anti-patterns. *Information and Software Technology* **74**, 194–203 (2016)
9. Glaser, B.: *Theoretical sensitivity. Advances in the Methodology of Grounded Theory* (1978)
10. Glaser, B.G.: The constant comparative method of qualitative analysis. *Social Problems* **12**(4), 436–445 (1965)
11. Glaser, B.G., Strauss, A.L.: *Awareness of Dying*. Routledge (2017)
12. Glaser, B.G., Strauss, A.L.: *Discovery of Grounded Theory: Strategies for Qualitative Research*. Routledge (2017)
13. Gregory, P., Barroca, L., Sharp, H., Deshpande, A., Taylor, K.: The challenges that challenge: Engaging with agile practitioners concerns. *Information and Software Technology* **77**, 92–104 (2016)
14. Harrison, N.: Beyond agility: Organizational patterns of successful software teams. In: *Agile Vancouver conference*, pp. 15–16 (2006)
15. Hoda, R., Noble, J.: Becoming agile: A grounded theory of agile transitions in practice. In: *Proceedings of the International Conference on Software Engineering (ICSE)*, pp. 141–151. IEEE Press (2017)
16. Hoda, R., Noble, J., Marshall, S.: Agile undercover: when customers don't collaborate. In: *Proceedings of International Conference on Agile Software Development (Agile)*, pp. 73–87. Springer (2010)
17. Hoda, R., Noble, J., Marshall, S.: Grounded theory for geeks. In: *Proceedings of the 18th Conference on Pattern Languages of Programs (PLoP)*, p. 24. ACM (2011)
18. Hoda, R., Noble, J., Marshall, S.: Developing a grounded theory to explain the practices of self-organizing agile teams. *Empirical Software Engineering* **17**(6), 609–639 (2012)
19. Holton, J.A.: The coding process and its challenges. *The Sage Handbook of Grounded Theory (Part III)*, 265–89 (2007)
20. Holton, J.A.: Grounded theory as a general research methodology. *The Grounded Theory Review* **7**(2), 67–93 (2008)

21. Kropp, M., Meier, A., Anslow, C., Biddle, R.: Satisfaction, practices, and influences in agile software development. In: Proceedings of the International Conference on Evaluation and Assessment in Software Engineering (EASE), pp. 112–121. ACM (2018)
22. Kropp, M., Meier, A., Biddle, R.: Agile practices, collaboration and experience – an empirical study about the effect of experience in agile software development. In: Proceedings of the International Conference on Product-Focused Software Process Improvement (PROFES), pp. 416–431. Springer (2016)
23. Ladas, C.: Scrumban: Essays on Kanban Systems for Lean Software Development. Modus Cooperandi Press (2009)
24. Leffingwell, D.: SAFe® 4.0 Reference Guide: Scaled Agile Framework® for Lean Software and Systems Engineering. Addison-Wesley Professional (2016)
25. Meier, A., Kropp, M., Anslow, C., Biddle, R.: Stress in agile software development: Practices and outcomes. In: Proceedings of International Conference on Agile Software Development (XP), pp. 259–266. Springer (2018)
26. Mencke, R.: A product manager’s guide to surviving the big bang approach to agile transitions. In: Proceedings of International Conference on Agile Software Development (Agile), pp. 407–412. IEEE (2008)
27. Pikkariainen, M., Haikara, J., Salo, O., Abrahamsson, P., Still, J.: The impact of agile practices on communication in software development. *Empirical Software Engineering* **13**(3), 303–337 (2008)
28. Schwaber, K., Beedle, M.: Agile Software Development with Scrum, 1st edn. Prentice Hall PTR, Upper Saddle River, NJ, USA (2001)
29. Schwaber, K., Sutherland, J.: The scrum guide. *Scrum Alliance* **21** (2011)
30. Scrum.org: What is ScrumBut? (2018). URL <https://www.scrum.org/resources/what-scrumbut>. Last accessed 20 October 2018
31. Sugimori, Y., Kusunoki, K., Cho, F., Uchikawa, S.: Toyota production system and kanban system materialization of just-in-time and respect-for-human system. *The International Journal of Production Research* **15**(6), 553–564 (1977)
32. Sutherland, J., Viktorov, A., Blount, J., Puntikov, N.: Distributed scrum: Agile project management with outsourced development teams. In: Proceedings of the Hawaii International Conference on System Sciences (HICSS), pp. 274a–274a. IEEE (2007)
33. U.S. Department of Health & Human Services: Card Sorting (2018). URL <https://www.usability.gov/how-to-and-tools/methods/card-sorting.html>. Last accessed 20 October 2018
34. Zieris, F., Salinger, S.: Doing scrum rather than being agile: A case study on actual nearshoring practices. In: Proceedings of the International Conference on Global Software Engineering (ICGSE), pp. 144–153. IEEE (2013)