



Agile environments and the large organisation: A Grounded Theory

PhD RESEARCH PROPOSAL

Prepared by

Peter John Alkema (9405456K)

Submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, in fulfilment of the requirements for the degree of Doctor of Philosophy in Engineering.

Supervisors: Dr. S Levitt, Prof. K Nixon, Mr. A Soicher

April, 2016

CONTENTS

Page

DECLARATION	4
LIST OF FIGURES.....	5
LIST OF TABLES	5
1 INTRODUCTION	6
1.1 Research context	9
1.2 Purpose of the study.....	10
1.3 Problem statement	13
1.4 Identified gaps	16
1.5 Proposal Structure	17
2 LITERATURE REVIEW.....	19
2.1 Introduction	19
2.1.1 Leadership of traditionally large organisations	19
2.1.2 Agile Software Development	21
2.1.3 Social inquiry through GT	22
2.2 Previous Work	23
2.3 GT and literature reviews.....	30
3 RESEARCH QUESTIONS	30
3.1 Core Question.....	30
3.2 Related and sub questions.....	32
3.3 Agile and Management	33
3.4 GT and the research questions.....	35
4 MOTIVATION.....	36
5 METHODOLOGY: GT.....	39
5.1 Background	39

5.2	Suitability	40
5.3	Practicalities	42
5.3.1	Interviews	42
5.3.2	Computer Tool Support	43
5.4	Design	43
5.4.1	Initial coding	44
5.4.2	Concurrent data generation.....	45
5.4.3	Memos.....	45
5.4.4	Constant comparative analysis.....	46
5.4.5	Theoretical sampling	46
5.4.6	Theoretical sensitivity	47
5.4.7	Intermediate coding.....	48
5.4.8	Identifying a core category	49
5.4.9	Advanced coding.....	50
5.4.10	Generating theory.....	51
6	ETHICS	51
6.1	Guidelines and approvals.....	51
6.2	Research participants.....	52
6.3	Anonymous participant information.....	54
6.4	Interview questions.....	55
7	DELIVERABLES AND PROJECT SCHEDULE	56
8	PROJECT BUDGET	59
9	FINAL REPORT	60
	REFERENCES.....	62
	APPENDIX.....	66

DECLARATION

This proposal is my original work and has not been presented for a degree in any other university.

A handwritten signature in black ink, appearing to read 'P. Alkema', with a large, stylized loop at the beginning and a trailing flourish.

PETER J. ALKEMA

LIST OF FIGURES

Figure

Figure 1: Project environments are co-created	6
Figure 2: Agile and management environment.....	14
Figure 3: Leadership, Agile and social inquiry	19
Figure 4: Contextual relationship between management and Agile	32
Figure 5: Focus of the core research question	33
Figure 6: GT methods	44
Figure 7: GT conceptual framework and coding activities	48

LIST OF TABLES

Table

Table 1: Programme of work	56
Table 2: Summary project budget	59

1 INTRODUCTION

Large companies that adopt Agile software engineering methods aim to improve the way their IT projects are delivered. Agile offers faster delivery and improved software quality which means lower costs and less risk for organisations that are continually under pressure to do more with less. Suncorp is one of the largest banking and insurance companies in Australia, having 20,000 employees of which 2,000 are IT personnel. In 2007 they started a five year journey of Agile adoption led by the new CIO and after two years had already started achieving benefits. As presented by Phil Abernathy at Agile 2009, a key learning in the journey was the need for a “symbiotic relationship between line management and the Agile teams” [1 p.314]. The broader issues and solutions in this relationship during the Suncorp journey are outlined in his paper “Hook, line and sinker: the role of line management in relation to Agile teams”. Building and expanding on these findings, the proposed research aims to develop an explanatory theory of Agile environments in large organisations. Critical to this are the managers who influence these environments who can understand how they can better support such journeys and projects.

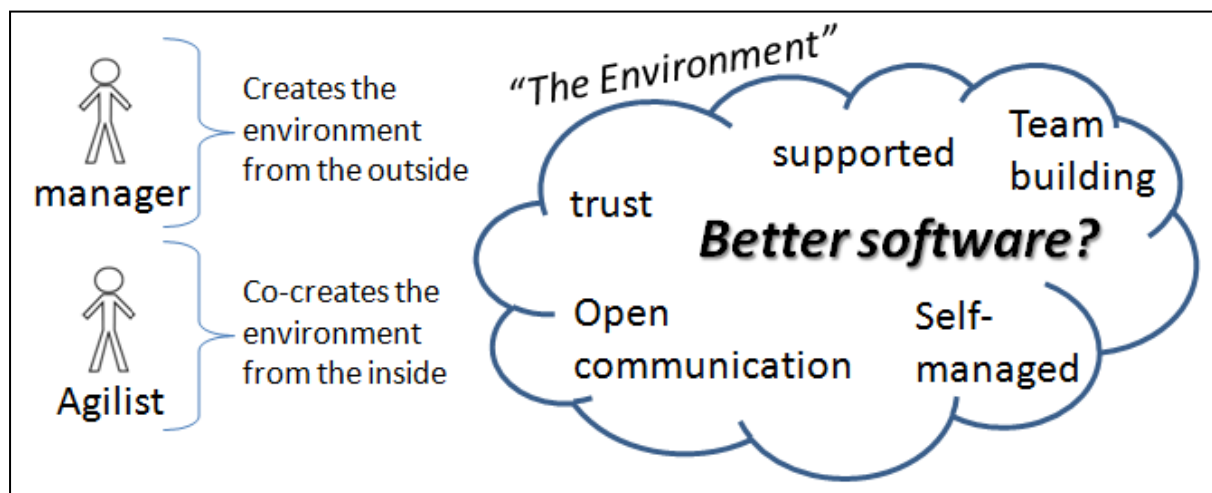


Figure 1: Project environments are co-created

As shown in Figure 1, Agile teams and line management both have a role to play in establishing and maintaining successful project environments. Roles inside such teams are well defined in Agile methodologies and although specific methods differ they are always guided by the Agile Manifesto [2]. Suncorp developed its own Agile

handbook to best suit their needs which they packaged into an open source knowledge hub which is freely available. In contrast to the largely internal focus of project teams, the role of management in respect of these project teams is to “give them the environment and support they need, and trust them to get the job done” [2]. In the Suncorp Agile journey, although it was led from the top it became clear that the traditional line manager’s handbook differs from the Agile way of working and “Daunting Dysfunctions” arose as a result. Although these appear generally well handled in the Suncorp journey it highlights that managers need an updated handbook so they know how to successfully set up and support Agile project teams.

Large organisations control the working environment and specific project environments through their management structures who implement common processes. These processes ensure consistent, predictable results from large groups of workers. This is extremely effective in operational and highly manual work areas where tasks are repetitive and easily defined. It is critical to understand the degree to which the working and project environments are influenced by management. The industrial age was fuelled by a highly efficient factory floor where blue-collar workers were controllable variables using conveyor belt production schedules and production calculations. The command and control management hierarchy is derived from the military where absolute discipline and regulated flow of information is essential for success in battle.

“Suncorp, like any large organisation, is hierarchical and structured around lines of business. [...] The IT department has three layers of middle management [...] the business analysts, developers, testers, architects and project managers all report to different line managers.” [1 p.314]

By contrast, the Agile way of building software encourages self-organisation and a large degree of autonomy by project teams. This approach has resonated well with IT teams that want to build software quicker and collaborate closely with their customers as well as other teams across organisational boundaries. However, this could be seen as disruptive or even a threat to the highly structured environment that

management has created, and it might therefore become difficult for these software projects to be delivered.

Two examples of managerial dysfunctions on Agile projects at Suncorp was “back door conflict resolution” and “corridor interventions”. Line managers exerted their authority counter-productively and this affected trust and relationships. Their management handbook in these instances was based on the transactional leadership style of reward and punishment rather than the Agile principle, “trust them to get the job done”. To help prevent such dysfunctions Suncorp developed a social contract for project teams to counteract such influence; it even states “We have zero tolerance for bullying” [3]. Characterising managers as bullies is an unfair generalisation but it is clear that the required symbiotic relationship between Agile teams and line managers required significant focus on the people aspects of software development. Ultimately they were successful;

“Teams that have gone Agile show an almost immediate increase in safety and morale. They feel appreciated, empowered, valued. This benefit, while hard to quantify, is firmly believed by senior leaders to be the key to sustainable success.” [1 p.319]

To achieve further success in other companies, the proposed research will use a method of social inquiry called Grounded Theory (GT) to generate data and answer a core research question;

How do managers in large organisations create the supportive and trusting environment required by Agile software development teams?

Using this methodology, the structure of such an answer to this research question will be;

“The final product of a GT study [which] is an integrated and comprehensive GT that explains a process or scheme associated with a phenomenon.” [4]

This is how GT takes a study beyond observations and conclusions and achieves usefulness for research. It is the explanation of the theory itself; the creation of meaning and new knowledge for its readers. If these readers are Agile project team members they will gain a valuable outside-in perspective of their environments and the dynamics of their collective relationship with management. If these readers are the line managers involved with Agile Software Development projects they will gain the necessary insight to achieve the important 5th principle of Agile.

1.1 Research context

Large organisations with traditional management structures are also being challenged by smaller companies, particularly those who use technology effectively and differentiate themselves in their industry. Ismail et al claims that the rise of the “exponential organisation” [5] is due in part to the compound effect of digitisation of technology and the democratisation of information. This book describes a radical new vision for today's enterprise which can achieve exponential growth by embracing technology which is digitized, deceptive, disruptive, dematerialized, demonetised and democratised. This has led to the rise of “Killer apps” which are so called because they “kill” the competition and make their use so unavoidably compelling it almost creates an unfair advantage particularly for start-ups who then give these apps to their tech-savvy customers and leap frog traditionally large organisations by orders of magnitude. Originally observed by Tom Goodwin these are examples of exponential organisations;

“The world's largest taxi firm, Uber, owns no cars. The world's most popular media company, Facebook, creates no content. The world's most valuable retailer, Alibaba, carries no stock. And the world's largest accommodation provider, Airbnb, owns no property.” [6]

Fuelling this accelerated change is innovative, intuitive software that empowers its users with unprecedented access to useful information. Ever shortening cycles of disruption to traditional business models and rapid changes in how this information is used means the software industry has had to become far more responsive in its

methods. Agile Software Development methodologies are based on the Agile Manifesto [2 , 7] which claims to be “uncovering better ways of developing software” and displays a preference for people, close collaboration with customers and iterative, incremental IT development.

This increased people-centric mindset in the software industry is significant; it is also what largely drives the essence of this study. The importance and potential impact of the human factor in technology is however not a new idea and featured in Fred Brooks’ seminal work “The Mythical Man Month” [8] written 40 years ago. In it he claimed, “the quality of the people on a project, and their organisation and management, are much more important factors in success than are the tools they use or the technical approaches they take” [8]. The Agile Manifesto, written 25 years later, states that software development should value “Individuals and interactions over processes and tools”. The Agile way of building software appears to have finally acknowledged Brooks’ hypothesis that the coder is actually more important than the code. An article on the history of the Agile Manifesto describes how

“companies have to rid themselves of ... make-work and arcane policies. This freedom from the inanities of corporate life attracts proponents of Agile Methodologies, and scares the ... traditionalists. ... scare corporate bureaucrats”. [9]

The corporate bureaucrats are traditional managers who are threatened by new work methods such as Agile. The exponential organisation succeeds by leveraging technology but generally they build a culture from the start that incorporates Agile. Conversely, managerial structures of large organisations actually defend the structure and status quo against disruptive work methods like Agile. Exponentials have re-written the rules of business success in numerous industries by thriving in shortening cycles of change and meeting increased customer expectations.

1.2 Purpose of the study

Pre-dating the information age, and even the concept of a modern day, commercial organisation, is the concept of leadership; how to achieve results through people. The application of leadership in organisations varies greatly but is claimed by Landis [10] to be “one of the most important components” of its success. Malo [11] describes the eight most important theories¹ of leadership of which the management (or transactional) model is the most common in organisations. According to the management theory of leadership, people perform best when there are clearly defined consequences for their actions, both as individuals and as a group. Borrowed from the success of military chains of command on the battlefield, the hierarchy of line management and subordinates in a company achieves this application of power from the boardroom to the cubicle. Just as the skeleton supports the human body, such structure is necessary but sometimes (especially in traditionally large organisations) becomes an overriding obsession which reflects in the culture, strategy, processes and even the design of its systems and software.

One of the most significant shifts in management thinking is the rise, and improved understanding, of knowledge worker roles such as doctors, lawyers and – most pertinent to this study – software engineers. Peter Drucker is the best known expert on management theories and first used the term “knowledge worker” in his book *The Age of Discontinuity* [12]. In it he differentiated these new roles from the manual workers of the industrial age because the knowledge worker’s main skill is their ability to work with information. Agile incorporates this new thinking and moves away from managing software engineers as task-driven resources with input-output methodologies to churn out code, like widgets on a conveyor belt. Another principle of Agile is “The best architectures, requirements, and designs emerge from self-organizing teams”. This is completely at odds with the transactional theory of leadership which requires a chain of command to ensure that subordinates are told what to do and measured on objectives they have largely not been consulted on. This could be a serious conflict for the traditionally large organisation; a new way of building much needed and better quality software which relies entirely on a fundamentally new approach to managing people.

¹ These eight theories of leadership are Great Man, Trait, Contingency, Situational, Behavioural, Participative, Management and Relationship

This study will examine how (and if?) Agile environments can be effective in organisations that are still largely reliant on traditional management practices. Are Agile and its practitioners flexible enough that these apparent constraints pose no serious threat to its success? Is there a link between how organisations harness and build technology and the way they lead the people who achieve that for them? The proposed research aims to generate an explanatory theory of Agile and Management by interviewing and conducting focus groups with practitioners and managers who have relevant experience. These insights and the semi-structured data arising from the interviews will be analysed using GT, a rigorous qualitative research method that aims to understand the primary concern of people living or working in a specific circumstance.

Barney Glaser and Anselm Strauss developed the GT research methodology as a result of their study of the experience of dying at Columbia University in New York. Their book “The Discovery of GT” [13] laid the groundwork for future generations of grounded theorists and its application beyond social sciences. It has become used more recently in Information Systems (IS) research where it is gaining traction for studies with people-centric objectives. A study of 50 IS journals between 1985 and 2007 found an increasing prevalence of GT articles and concluded it to be “an appropriate research methodology in IS” [14 p.145]. A selection of four GT studies in Software Development show this simply by the socially minded nature of their topics; “understanding lack of trust” [15], “transition and adoption” [16], “how people manage” [17] and “the experience of” [18]. While there is precedent for using this research methodology in software engineering, this proposal also establishes why it is appropriate for this study.

The leadership teams of organisations are required to implement management plans and practices to achieve their strategic objectives. The dominant model of leadership in organisations is transactional; people need to be told what to do and if they do it they are rewarded, if not they are punished. According to its manifesto, Agile software engineering is based on a different philosophy; one of trust and self-organisation. This study aims to understand how the introduction of Agile software

development is impacted by such management practices in its immediate environment. The study also aims to examine the converse, ie in organisations where leadership practices are more closely aligned with the Agile Manifesto, it follows that it would be easier to introduce and run successful Agile software projects.

1.3 Problem statement

According to Boehm and Turner's [19] research, Agile works well at lower levels on smaller projects that are specific to a certain area such as custom web applications. However – as is usually the case – if the institutionalised system development methodology is Waterfall then trying to achieve broader adoption across bigger and more complex projects becomes a challenge. They examined the results of a series of workshops held in March 2004 by the University of Southern California Centre for Software Engineering. Barriers to successful Agile adoption that were categorised as real problems – and relevant to this proposed study – were HR policies, roles and responsibilities, required co-location and customer access.

According to that research there are problems relating to the human factor when introducing Agile practices into organisations that are run with traditional management styles. "People issues are by far the most critical in improving management of engineering and development personnel. Addressing them is vital to the adoption and integration of Agile methods and practices" [19 p.36]. A quick review of the principles of Agile yield a number of people specific phrases; "work together", "face-to-face conversation" and "the team reflects" however other system development frameworks would also claim these ideals are important. However, there is one uniquely Agile principle that will be the primary focus of this study because it directly challenges the transactional model of leadership. "Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done." If management controls the environment and operates differently, is the team getting the support and trust it needs? Understanding this environment and the forces and factors at play requires consideration of how the Agile team (which is ultimately guided by the Agile

Manifesto) is acted on by its management (which is evidently guided by different leadership principles).

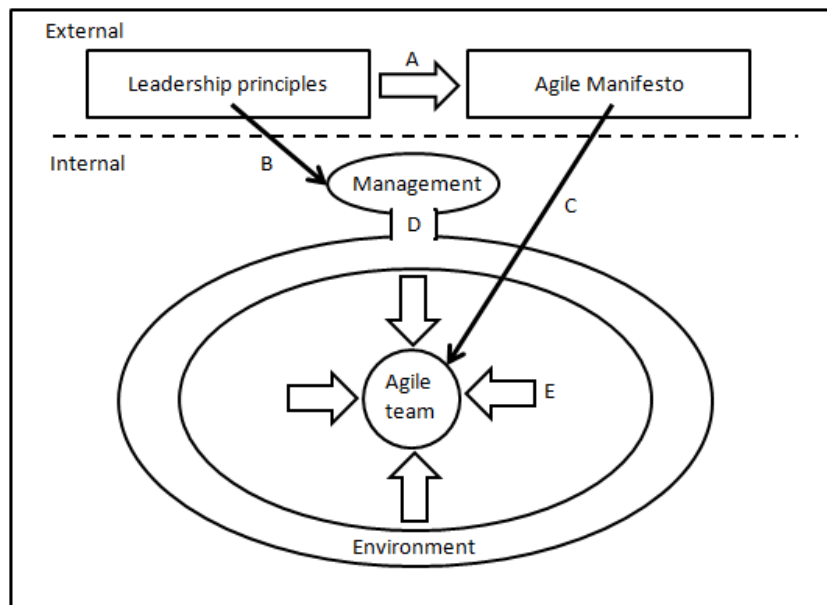


Figure 2: Agile and management environment

External to an organisation (at the top left of Figure 2), leadership principles have emerged from how people coexist in the pursuit of a common objective. Software development teams are comprised of people who also coexist in pursuit of the common objective of building software and the Agile Manifesto appears to return to leadership principles to best achieve this. This is not to say that organisations by definition lack leadership or that their vision and strategies are devoid of leadership principles. A great many leaders have emerged from the traditional enterprise and many companies demonstrate strong leadership values in their culture and management practices. Agile principles are thus informed (A) by leadership principles and re-balance the approach to software development from task driven to people-led.

Management is the internal structure of an organisation that is tasked to achieve its goals and (especially in traditionally large organisations) usually inherits (B) the transaction style of leadership. Agile teams aim to reflect the manifesto (C) as far as possible but are in turn subject to the environment created by management (D).

These teams cannot create or dictate the nature of the environment around them, much like an employee cannot change his / her manager. If the influencing forces (B) and (C) are highly correlated it follows that the environmental factors (E) which exert on Agile teams will perpetuate the harmony of this system. If management creates an environment which is not aligned with Agile principles then these forces will work against Agile teams. Or, if the leadership principles that inform management practices are misaligned with the Agile principles that guide Agile Software Development, then the conflict will be evident in the environment of that project team.

If management was guided by both the Agile manifesto and similar leadership principles then the only balancing required is internally. Managers would simply have to reconcile any differences as rather being a continuum of choice in the daily decisions about how they oversee Agile teams. This is the specific situation the author is in; both as a manager with transactional leadership responsibilities for people and teams, on the other hand heavily influenced by the Agile Manifesto and the people-centric ways of developing software that it embodies. A successful approach thus far has been to reconcile any differences constructively and in the best interests of the team. A good example is performance management; as a manager one is required to conclude a yearly detailed performance contract with subordinates, however as a leader of Agile software development the guiding principle is “trust them to get the job done”.

Richard Branson inspirationally claimed there is no point hiring bright people and telling them what to do; you should hire bright people so they can tell you what to do. This is also true of software developers; they usually know more about the job they are doing than their manager. Both of these positions are important and can be held together in a natural tension; the daily answer is usually a combination of both (but not a compromise between them). But what about Agile teams operating under managers where there is low correlation between Agile principles and their management practices? In the tradition of all GT studies, this problem can be resolved to a simple question of “what is going on here?” What is going on when

Agile software teams are working in environments created and heavily influenced by transactional management?

1.4 Identified gaps

According to a comprehensive systematic literature review in 2010, “there has not been enough focus by primary study researchers on the area [of human factors in software engineering]” [20 p.78]. An initial search² in relevant databases yielded 401 (2000-2010) results which were then further filtered to 67 papers as the scope of that research. These 67 papers were reviewed in terms of software engineering management and its project phases, human roles and human factors. The last two categories covered management and organisational aspects which correlate well to the leadership focus of this study. The author’s own search across the same years, (excluding the “human factors” search string) produced the following results: 302,000 (IEEEExplore), 201,000 (ACM) and 93,000 (Scopus). This indicates a disproportionately small focus on the human factors of software engineering research. Given that only the human roles and factors categories of the above research include leadership practices of organisations, the coverage of this specific topic in the literature is even smaller by comparison.

A more rigorous study [21] focused specifically on Agile Software Development and presents an overview of research in the same period (10 years) from the inception of the Agile Manifesto (2001). Given that Agile methods were still being developed and matured during this period, the research also reflected a constantly shifting subject matter, and the paper re-iterates calls for a structured approach to the expanding streams of interest. During the initial years of this period, the focus was understandably on the adoption of Agile methods, while more recently the focus has shifted to the control of projects and social aspects such as team communication, trust and self-organisation. It is surprising that the increased emphasis on people aspects in the Agile Manifesto did not appear to generate a proportional increase in this aspect of the research much earlier in the period. The authors of the manifesto

² The search string (“Software Development Process” OR “Software Development” OR “Software Engineering” AND “Human factors” OR “Human issues”) was conducted on Scopus, IEEEExplore, ACM Digital Library and Google Scholar.

were quite visionary in attempting to rebalance their field accordingly, but the quantity of research about the people centric aspects of their work has evidently taken some time to catch up. Total publications on Agile Software Development in ISI Web of Science grew tenfold from under 25 in 2001 to 275 in 2009. Arguably this rapid increase means that any aspect of Agile is still under-researched; this comparatively young field of research simply hasn't matured enough to make meaningful comparisons that point to conclusive gaps.

Management research has also identified the need for further studies on how leadership affects IS projects. Faraj and Sambamurthy proposed a theoretical model in their study of IS project leadership and pointed out the "growing realisation that IS projects are characterised by challenges in communication, co-ordination, learning and negotiation" [22 p.238]. These "peopleware" topics are best handled by the project leaders and Faraj et al offer various reasons why we still don't fully understand the application of leadership in IS projects;

- "First, most studies of leadership have been in broader organisational contexts, unlike focused work contexts such as IS development.
- Second, IS projects represent knowledge work contexts [...] Traditional styles of leadership [...] might not be very effective in such contexts" [22 p.238]

These two reasons are highly applicable to the large organisations context of the proposed study and the four areas of challenges listed above are intensified on an Agile project due to the added complexity of using a relatively new project delivery methodology. The quoted study analyses findings related to directive and empowering leadership styles in IS project settings and creates a useful model for further understanding of these factors in team and project success.

1.5 Proposal Structure

Chapter 2: Relevant prior research is summarised in the 3 main areas of interest; leadership, Agile and social enquiry. Starting from a broad perspective, this prior

work is narrowed down to specific, comparable examples especially where the proposed topic was recommended as future work.

Chapter 3: The intended research questions position the proposed study more specifically within the body of relevant prior work. A core question narrows the focus of the research without reducing the possibility of sub-questions or related topics emerging in line with the GT method of research.

Chapter 4: The motivation for this study is rooted in the desire to know more about the human factors of software development. Particularly under the conditions of power and influence that exist in the environment of Agile teams in today's large enterprise.

Chapter 5: Provides an overview of the chosen methodology GT, so called because it is grounded in the data and enables the emergence of a theory from the collection and analysis of semi-structured data. Definitions as well as practical aspects such as computer tools and interviews are also dealt with.

Chapter 6: The ethics process is outlined (non-medical), given that the research will involve the participation of human subjects, all of whom will have to give consent to participate in the research. It is affirmed that no exceptions to the process are required and the approval is expected to be fairly standard.

Chapter 7: The overall project plan for the 4 year study is given as well as some communications practicalities with regards to supervisor support. I also commit to completing some conference presentations and journal articles related to the research; this will ensure contribution to knowledge throughout the study.

Chapter 8: Estimated costs for the full period of the study are given and confirmation is provided that no funding is required and all travel and accommodation (if required) will be paid for by the researcher. Furthermore there is a bursary in place whereby the researcher's employer (FNB) covers the annual costs of tuition and related expenses although this has to be re-approved annually.

Chapter 9: The contents and layout of the final report is given which importantly complies with the style guide of the School of Engineering and the Built Environment. Using the GT method of research means that the structure (and representation) of the theory which emerges may manifest in the actual headings of the report and one example is given from a related study.

2 LITERATURE REVIEW

2.1 Introduction

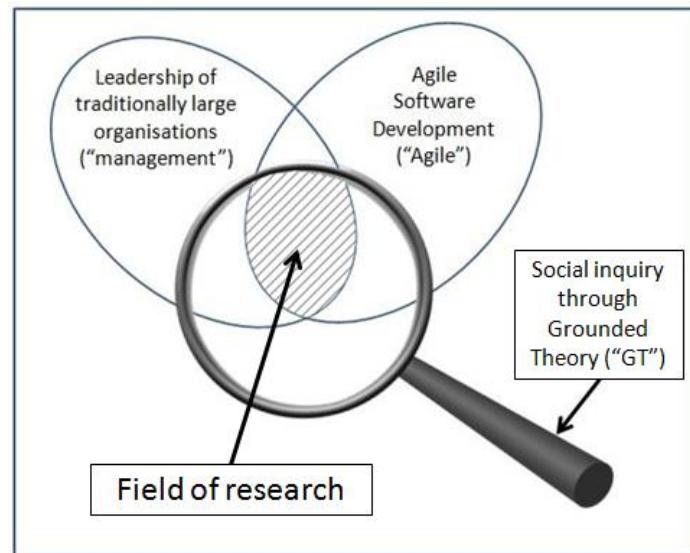


Figure 3: Leadership, Agile and social inquiry

Two overlapping topics create the field of research for this study; (1) leadership of traditionally large organisations ("management") and (2) Agile Software Development ("Agile"). GT is the method of inquiry through which this world will be examined and a preliminary review of relevant previous work on all three topics has been completed.

2.1.1 Leadership of traditionally large organisations

Multi-national, listed and / or Fortune 500³ companies need to provide direction to the large number of people they employ. The size of the group of people they control (typically 10,000 or more) requires formal arrangements so that direction can flow down the organisation and communication back upwards. Typically at the top there is a board which is answerable to the shareholders; a Chief Executive Officer sits on this board and is mandated to convene and appoint an executive committee that works directly for him / her. Below this, management structures are put in place throughout the company so that every single employee is contractually bound to obey the instructions of someone placed in authority over them. 'Spans and layers' provide a measure of organisational design; the wider a span, the less layers are required, the more layers, the more complex the interactions. A span is the number of direct reports and the layers are the number of successive people (each reporting to the other) below a person. A typical measure is the number of hierarchical layers from CEO down to the lowest level, and conventional management theory holds that "seven hierarchical layers is enough or more than enough for all but the largest corporations" [23 p.131]. All such large organisations will have an IT department and be involved with developing software, whether for their own internal use or for sale into the markets within which they operate. The people and teams in these IT departments (centralised or decentralised) are also employees of the organisation and as such, are subject to its management practices and influenced by the nature of its leadership.

In the book "Leadership in organizations: Current issues and key trends" [24], Graeme Salaman contributes a chapter discussing management and leadership in the context of the enterprise. He discusses the concept of a "competent manager" and builds an argument that emphasizing certain core competencies can redefine and re-align management more usefully within large organisations. Leadership, almost sceptically it appears, is seen as "making up for the weaknesses and deficiencies of organisation" [24 p.74]. If managers must become competent to be effective within the enterprise, then the role of leaders – in order to correct these deficiencies – is to transform the organisation. However, Salaman argues that "the

³ The Fortune 500 is an annual list of the 500 biggest corporations in the US, as measured by gross income

cult of the charismatic individual leader” [24 p.76] has largely undermined this project and it is thus back to the competent manager to now also become a leader and transform the organisation from within. “Leadership competencies” have been added to the competency framework for managers and Salaman concludes that functionally, management and leadership “are remarkably similar in that both offer to resolve the failures of organisation by avoiding and individualising them.” [24 p.77] The context of IS projects also highlights the comparison of leaders and managers; especially given the transformative and disruptive nature of Agile.

2.1.2 Agile Software Development

Agile is a new approach to building software which has emerged in the last 10-20 years to counter increasing dissatisfaction with the established, plan-driven methods such as Waterfall. The Standish Group’s Chaos report;

“The Agile process is the universal remedy for software development project failure. Software applications developed through the Agile process have three times the success rate of the traditional Waterfall method and a much lower percentage of time and cost overruns.” [25 p.25]

However, Agile and Waterfall are very broad frameworks, within which there are many specific methodologies, often unique to certain software packages, programming languages, industries and types of projects. An example of this is the Waterfall methodology Oracle Unified Method (OUM) which is based on Unified Process (UP) but intended specifically for use by project teams that are implementing Oracle software. The proposed study is not intended as a critique of either method or a comparison between the two; the increasing popularity of Agile is clear but it still faces challenges. One example is its relevance in the large enterprise, where typically the dominant method is Waterfall and thus Agile’s credibility is derived partly from being a promising new alternative. Even The Standish Group’s Chaos Report should be seen in the context of doubts about its figures and definitions [26], although this was prior to the above quotation.

Agile methods are essentially team focused, and give projects very practical tools and approaches to delivering software better but by nature therefore it generally emerges at the bottom of such an organisation. Extensions of Agile such as SAFE (Scaled Agile Framework) [27] have attempted to position Agile for use across broader spans and higher numbers of hierarchical layers but has been met with decidedly mixed feelings from practitioners. Occasionally, Agile will be brought into an organisation at a very senior level (such as at Suncorp) and a culture of top-down support is created for its principles and methods, depending on what is relevant. Unfortunately this is sometimes a blunt instrument of doing Agile by instruction, rather than embedding it into the culture and processes through a process of natural, but intentional adoption.

2.1.3 Social inquiry through GT

As with the rise of human centric principles and methods in software development, so too there has been an increased need in such circles to borrow from the research methods of sociology, the study of human society and its patterns. Social inquiry is mostly used to explore, describe or explain the world around us by making observations, and interpreting what has been observed. GT is an explanatory means of social scientific research and thus seeks “to indicate causality between variables” [28 p.95]. We are interested in understanding what causes things to happen; in the present discussion, how are social variables related in the world of the software developer? There are different philosophical standpoints the researcher can take when asking and answering this question. Positivist sociology seeks to understand the natural laws that govern society and the interaction between the humans that live and work within it. It assumes that these laws exist outside of, and independently of the subjects of their research; gravity is still there whether the apple falls from the tree or not. Interpretive sociology places the human mind at the centre of its research and proceeds on the basis that the world around it is a function of that person’s interpretation of it. It is our consciousness that defines us; the falling apple is important because it is meaningful (or not) to me, it is not important just because it proves the existence or causal effects of gravity. GT as a framework eventually split along these lines with its original two co-founders Glaser and Strauss adopting

strongly positivist and interpretivist positions respectively. As expanded below in the methodology section, the proposed study will follow the Straussian variation of GT and the researcher positions himself in this work as being interpretivist. In the context of this literature review and its background for the proposed study, GT is more than a set of methods. It is a means of social inquiry that informs our understanding of the enterprise as a social construct within which we are interested in how the IT minds at work actually operate and interact as social beings.

2.2 Previous Work

The interview “Agile at IBM: software developers teach a new dance step to management” [29] offers practical insight into how Agile improves cycles of innovation at a global software services firm. Rob Purdie (Agile Practice Lead for IBM Design Lab in New York City) describes how speed, better quality, avoidance of waste and increased predictability have improved the way they deliver digital marketing solutions for their clients. Borrowing from the language of the Agile Manifesto he highlights the need for “leaders and teams to work and learn ... in an open and transparent environment” [29 p.29]. The article mentions the principles of Scrum [30] which emphasize the self-organising nature of teams in these environments and the reduced importance of external / top-down managerial control:

1. “Management does not interrupt the team during a work cycle.
2. The team reports to the customer, not the manager.
3. The team estimates how much time work will take.
4. The team decides how much work it can do in an iteration.
5. The team decides how to do the work in the iteration.
6. The team measures its own performance.” [29 p.29]

These new “dance steps” are likely to be uncomfortable for traditional managers who will perceive a loss of control and be threatened by the existence of these self-organising teams. Evidently at the IBM Design Lab this is not a primary concern and they successfully focus more on “radical management approaches that promote swift and continuous innovation” [29 p.26].

Whether a traditional manager can learn such new “dance steps” is also an important sociological question; many did make a successful transition at Suncorp. In another study a model was proposed for assessing an individual’s ability to succeed as a leader on an Agile project. The model used the Myers Briggs Type Indicator (MBTI) methodology in an “inquiring systems perspective to identify traits that will enhance the scope and effectiveness of leaders in an Agile environment” [31 p.85]. It was found that specific behavioural characteristics aligned well with the nature of Agile projects and the requirements on leadership roles in that environment. By implication it was found that, “A very rigid and controlling type of manager, with a successful track record managing process driven software projects, may not be the right candidate to manage an Agile project” [31 p.84]. The contrast between traditional management styles and progressive leadership approaches is again brought into focus on Agile projects where the delivery method is relatively new, but the benefits of adopting it are significant. Bonner also suggests that existing leadership categories may not be enough to understand what is required of Agile leaders.

Other examples of large enterprises that are using Agile has been studied in the Netherlands where the results of co-existence with plan driven methods was less positive. Waardenburg and Vliet [32] interviewed 21 Agile practitioners and developed a GT of “When Agile meets the enterprise”. They identified two major challenges that emerge when adding Agile projects to a traditionally large organisation; increased landscape complexity and lack of business involvement. In both the organisations they studied, they found that “Agile practices were implemented at a development team level, with the intention to grow into the organisation” [32 p.2170]. One of the suggestions for further research was the interaction between IT management and IT development as they found these groups differed on how this would actually happen. In these two instances from the Netherlands, the classic clash between semi-autonomous self-organising teams and external, top-down controlling management was evident.

The need for top down support of IT projects by senior managers was re-iterated by Neufeld et al in their paper “Charismatic leadership and user acceptance of information technology.” While the specifics of Agile were not included as differentiating characteristics, they conducted extensive research and found “in this body of literature spanning more than 30 years is the finding that top management behaviour – however it may be manifested or measured – is associated with important outcomes” [33 p.496]. This is a broad position and their study aimed to enrich this relationship by focusing on the nature and impact of charismatic leadership on such projects. Another paper, in the International Journal of Productivity and Performance Management also linked project success to leadership performance but highlighted the lack of empirical evidence to support this. Nixon et al observed that;

“in the general management literature, it is widely recognized that the performance of the functional manager’s leadership style contributes to the success of the organization he or she manages. The project manager’s performance is ignored when identifying project success factors.” [34 p.206]

Here again we see the tension between; organisation and project, manager and leader, progressive and traditional styles. As with Neufeld, these authors also supported the primary classification of leadership styles into either transformative or transactional, reinforcing this broad grouping which also underpins the proposed study. As with Salaman above, Nixon et al also do not see a discontinuity between a manager and a leader and states that “leadership is a leading behavioral trait exhibited in effective managers” [34 p.214]. By definition therefore, the manager is not necessarily a rigid construct of the enterprise, he / she can become a leader and adapt to the situational requirements of their managerial role. This is good news for enterprise managers who find themselves creating and managing Agile environments. According to the above literature it is possible that they can adopt the leadership styles that such an environment needs.

A further example of research on Agile adoption in large organisations is a questionnaire based study [35] at Nokia which covered seven different countries in

Europe, North America and Asia. The 1000 respondents were largely in Agile project team roles and the empirical evidence showed clearly that support of these new methods increased with the duration of exposure to them. One of the goals of that research was to highlight challenges with Agile; they found the biggest of these to be “Agile deployment and adaptation of Agile methods to fit the organization”. Some of the suggestions for further research in this paper propose interviewing as a way of getting to grips with the adoption of Agile. This suggests that such underlying issues are better understood with more qualitative research methods like interviews and focus groups. The Nokia research is important and its methods allow for a high number of respondents, but the quantitative results can obscure the true essence of what is really going on in the human interactions of such endeavours. In this regard, one such suggestion in their study is to consider the effect of organisational culture, which is often directly associated with the nature of its management practices.

This association is observed by Hoda in her study [36] of 23 software organisations across New Zealand and India, “Senior management has been found to influence organizational culture, which in turn influences the adoption of Agile methods in an organization”. She interviewed 58 Agile practitioners over four years and developed a GT of the roles in self-organising Agile teams to assist such teams perform better. The study also found that senior management plays a crucial role in creating an environment which is conducive for self-organising teams. Specifically their research

“found that self-organizing Agile teams (practicing Scrum or combinations of Scrum and XP) require a conducive organizational culture marked by freedom, openness, trust, and an informal organizational structure. In contrast, an organization with a hierarchical organizational structure and an environment of restricted, formal, and indirect communication restricts the teams’ ability to self-organize.” [37 p.12]

Interestingly, reference is also made to an adoption of XP at IBM [38] where senior managers were sceptical because of the informal, game associated nature of XP’s terminology; this is in contrast to the experience elsewhere in IBM, specifically at the Design Lab given above. This shows that large, multinational organisations like IBM

are not homogenous and there can be vastly unrelated and different experiences in separate parts of the same whole. Caution is thus required when drawing conclusions about an individual large organisation. Accordingly, the Nokia work could have been enriched by segmenting the data across countries or even down to co-located teams but ethics and anonymity requirements may have restricted this.

Hoda's GT work on self-organising teams demonstrates how the science of social inquiry can effectively be applied to the systems development world. Similar GT research [15 , 18 , 39 , 40] offers socially sensitive but practical and re-usable theories that might actually be more useful than software development methodologies. Adolph and Krutchen argue this point in "Generating a Useful Theory of Software Engineering" [41] and highlight the reductionist nature of most software methodologies which "limit, and even impede the way individuals actually create software" [41 p.47]. This is because;

"The social processes are not a side effect of the software development process; the social processes are the control mechanism [...]. Put in modern Agile shorthand 'people trump process' " [41 p.48]

While they acknowledge that a unifying, master theory of software engineering is unlikely to exist, they support GT as a way of developing mid-level theories about the human factors of software engineering. Additional studies such as the proposed one will add to the collection of these theories that;

"better capture the messiness [...] that characterises software development. Empirically generated theories capture what people are really doing rather than what experts believe they should be doing." [41 p.49]

Senior managers in large and complex enterprises also need practical advice to assist them with changes and challenges to their world. If, as Laanti et al claims [35] "Agile methods are here to stay" then practical strategies are required so that the value of Agile is not obscured by the evident difficulties of adopting it. Boehm and Turner offer such advice in "Management challenges to Implementing Agile

Processes in Traditional Development Organisations” [19]. A core question they ask is “How do you merge Agile, lightweight processes with standard industrial processes without killing agility?” They list a range of considerations such as different lifecycles of integrated systems, varied approaches to gathering requirements, process standards ratings, documentation, deployment and risk management.

In agreement with Adolph and Krutchen above, they claim, “People issues are by far the most critical in improving management of engineering and development personnel”. Their subsequent suggestions for addressing such people issues reads like a how-to guide for adopting many of the principles of the Agile Manifesto;

7. “Understand how communication works within development teams
8. Translate Agile and software issues into management and customer language ... because of the ‘eyes glaze over’ response
9. Emphasize value ... acknowledge the value of each team member
10. Pick good people and reward the results of pilot projects
11. Reorient reward systems to recognize both individual and team contributions”

These authors expand in detail in their book “Balancing Agility and Discipline” [42] in which they tell the story⁴ of an elephant and a monkey that lived nearby a village and worked in very different ways to find and bring back food for the people. Eventually they combined forces; although it was difficult at first, they acknowledged their complementary skills and developed a model whereby they could work together, “secure in their mutual trust and the appreciation of well-fed villages”. This story is an obvious depiction of hybrid models of software development but an equally powerful metaphor for how it is possible to achieve both agility and discipline. If Agile is seen as adversarial to Waterfall, this will be a source of conflict and a cause of people issues because management is usually comfortable with Waterfall. Even though the

⁴ The elephant had established well-worn paths to find food for the village and could bring back enough for days at a time. Despite being loved and appreciated by the people, he became unable to cope with their demands for rarer ingredients for new recipes. The monkey was adaptable to the new requirements and became more popular than the elephant as he quickly flitted about the forest gathering the exotic food. But the insatiable requirements outstripped the monkey’s abilities; he grew tired and the villagers became impatient.

IBM Design Lab uses Agile and other areas of IBM does not, perhaps it is a large enough organisation that they operate independently and therefore avoid a turf war. Management structures that are used to Waterfall and plan-driven methods may exert control by defending what they know and by default, oppose new methods and the people that try and introduce them.

One of the managerial roles in organisations that may oppose new methods because it is often disrupted by the adoption of Agile is the project manager. Although operating at a lower level than senior managers, they still play a significant role in setting up environments within which Agile teams are expected to operate. In “A project manager’s survival guide to going Agile” [43], Agile coach Michele Sliger addresses the confusion and disorientation that project managers feel by mapping PMBOK knowledge areas to Agile practices and offers new definitions of the project manager’s job which better suits the self-managed team environment, as researched in detail by Hoda [36], above. Interestingly Sliger points out the compatibility of plan-driven and Agile methods as well as the compliance of Agile to CMMI, if properly implemented. She also draws on the concept of servant leadership as the ideal style of leader for an Agile project. Some of the earliest work [44] on servant leadership was by Robert K. Greenleaf who developed the theory at AT&T. Written in 1977 as a response to the leadership crisis in America at that time, the ethos of a servant leader;

“begins with the natural feeling that one wants to serve, to serve first. Then conscious choice brings one to aspire to lead. That person is sharply different from one who is leader first [...] The difference manifests itself in the care taken by the servant first to make sure that the other people’s highest priority needs are being served.”

The servant-leader-project-manager thus places primary focus on the people in the environment they are setting up and supporting. This style of leadership aligns perfectly with the Agile principle of creating the trusted and supportive environment for the people to work in successfully. Out of that will come the self-managed team

which then requires the empowering and directing leadership to achieve overall project success.

2.3 GT and literature reviews

The role, sequence and nature of literature reviews in GT studies has been widely contested. If a theory is to emerge from the data during the process, the researcher should not be influenced by existing hypotheses or results on the same or closely related topic. The positivist tradition of GT supports the completion of an initial, but limited literature review to assess the current state of relevant study which also ensures the researcher's credibility with the participants. A more extensive literature review will then also be conducted once the theory has started to emerge during data collection and analysis. Performing an extensive literature review later in the process means that relevant work is searched for and read with the developing theoretical framework in mind. The proposed study will also be guided by traditional scientific research methods to ensure that the initial literature review is also substantial enough.

3 RESEARCH QUESTIONS

3.1 Core Question

The proposed study will centre around a crucial “golden thread” of enquiry that guides the research journey and ensures consistent relevance;

How can large organisations create the supportive and trusting environments required by Agile software development teams?

The core question above is also a re-arrangement of the 2nd sentence of the 5th Agile principle;

“Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.” [2]

Some guiding principles inform the above approach;

1. The literature review indicates there are challenges to achieving this environment principle and hence most evidence from the field is expected to be negative or rather tend to answer the question “how does management fail to create the environment [...]” etc. The core research question is however written in the positive as its answer (the explanatory, grounded theory) will be a more constructive contribution to the body of knowledge than an abstract description of what is already largely self-evident.
2. In the exact wording of the Agile principle in the manifesto, “support” and “trust” are not direct adjectives of the environment however the question uses them as adjectives, ie “create the supportive and trusting environment”. Their association with the environment is implicit by their presence in this principle and provides two valuable ‘hooks’ or dimensions with which to engage the data.
3. The full set of Agile values and principles inform the application of the various specific Agile methodologies that have emerged; Scrum, DAD, XP etc. Said differently; if the values and principles are in place then the methodology itself has a higher chance of being implemented successfully. The core research question goes beyond this corollary and asks how to actually implement the principle in the first place, ie how to “create” the environment, which assumes it was not pre-existing.
4. Agile methodologies are also oriented to an inside-out perspective of the project environment; they focus on the primary elements of the project itself. The core research question reverses this frame of reference to outside-in and balances the emphasis of the study with the broader universe outside the project environment ie the large organisation.

3.2 Related and sub questions

The relationship between the management of project environments in large organisations and their Agile projects is thus the broader context of interest. If management is informed by transactional leadership principles, and Agile projects are informed by Agile principles then a 2D grid of these elements would describe such a relationship. Agile principles (A) on the left of Figure 4 below are well defined and their presence may or may not be in conflict with various management practices (B) in the same environment. Answering the core question above also requires a broader understanding of the causal relationships that exist and how they bring about the social interactions that are observed on the project.

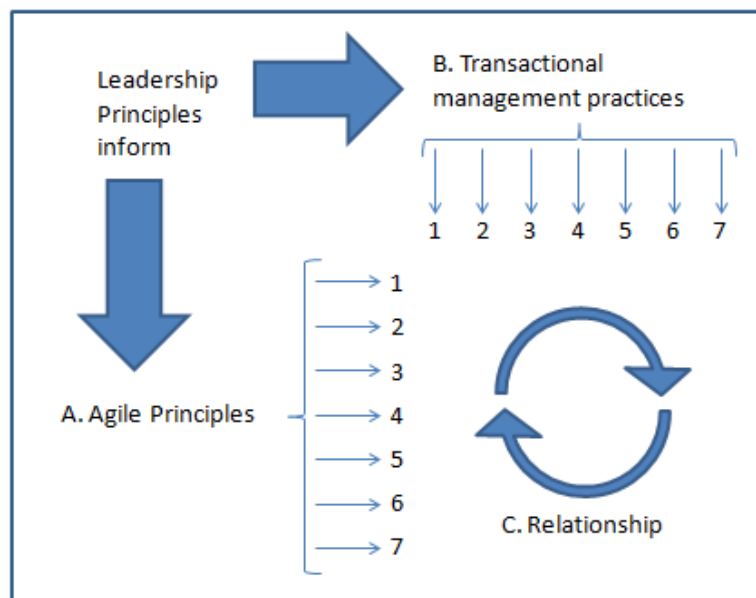


Figure 4: Contextual relationship between management and Agile

Related questions could thus include;

1. Are transactional management practices in conflict with Agile principles?
Why? What examples are there?
2. Do transactional management practices and Agile principles have any correlation on successful / unsuccessful Agile projects? Why? What examples are there?

3. What causal effect do transactional management practices have on Agile principles? What examples are there?
4. Do any Agile principles have a causal effect on management practices? Which ones and why? What examples are there?

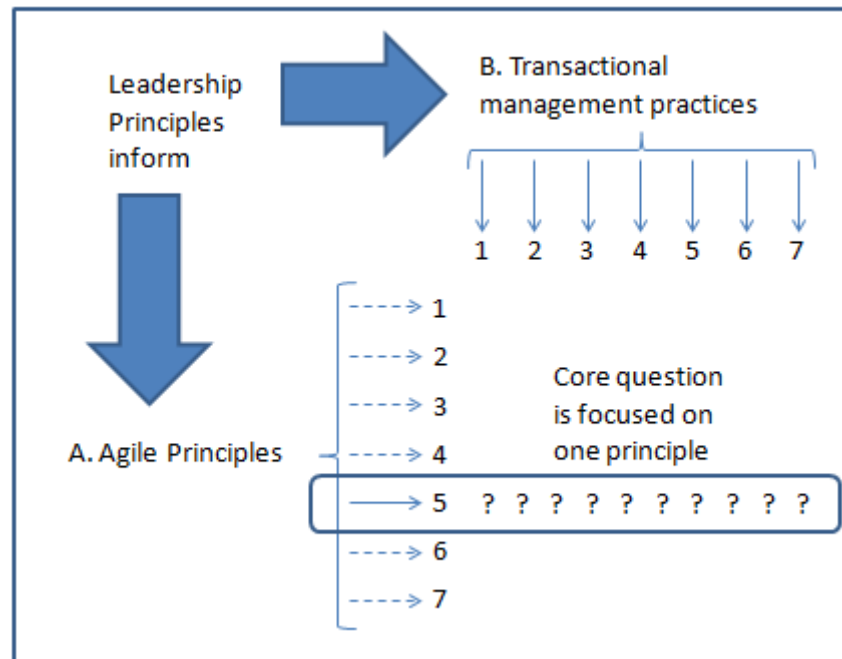


Figure 5: Focus of the core research question

The core research question focuses on the environment principle and thus sub questions could include;

1. Which management practices most / least influence the successful application of this principle?
2. Does this correlate to / cause the success / lack of success on Agile projects?
3. How do you build projects around motivated individuals?
4. How do senior managers become equipped to do this successfully?
5. What does trust mean in the context of Agile environments?
6. Can motivated individuals in this context get the job done if the environment and support they need is not in place?

3.3 Agile and Management

In the study mentioned above, “When Agile meets the Enterprise” [32] Waardenburg and Vliet present “a grounded theory of the challenges of using Agile methods in traditional enterprise environments”. While the most important challenge they considered was the co-existence of plan-driven methods they also looked at the impact of business management on the Agile environment. This was done under a broader category of challenges called “Lack of Business Involvement” but highlighted issues pertinent to the proposed study which is well summarised by one of the research participants;

“The problem with business stakeholders is the fact that they are still used to old habits. They feel their job is done when [...] all requirements are elicited. It is really a case of we-them mentality. They throw their list of requirements over the fence and expect their responsibility and involvement stops there.”
[32 p.2164]

This comment contrasts with Agile principles which require the ongoing and active management of the environment within which projects are being run. The above research further concluded that lack of business involvement was a clear challenge to Agile projects and recommended mitigating actions.

Another relevant study [45] considered the daily stand up meeting (DSM) which is an important aspect of many Agile methodologies to improve team collaboration and ensure regular focus on progress. In the DSM, team members gather on a daily basis to discuss plans and progress on the project which is intended to embed the self-organising nature of Agile teams. Using four case studies and one GT study, various positive and negative aspects of the DSM were analysed to understand their effect on the project. Of significant relevance to the current study is that DSMs were actually perceived to be negative when “authoritative managers [...] use the meeting to obtain status information, mainly useful to themselves.” It also emerged that the DSM was used to blame developers for poor code, who became defensive and attacked the managers for overselling the product. Although this study was

conducted in smaller organisations it is again clear that it is important to understand the nature and impact of business management on Agile projects.

Just because a project runs according to an Agile methodology and subscribes to the Agile manifesto, it does not mean it is immune from factors around it that operate on different principles. As supported by the two studies above, the main proposition of the proposed work is that the nature of the relationship between the Agile project and the management around it is crucial to understand; both to mitigate negative impacts but also to harness positive ones. Another participant in the Waardendburg study stated that lack of business involvement was “one of the hardest things to cope with when practising Agile”. Although the DSM is a popular Agile practice, Stray found that project members were generally neutral in their attitude to it. Both studies highlight the importance of the project environment and specifically the effect that management and business stakeholders have on it. The research questions for the proposed study will contribute to this area of research even further by asking more specifically about the nature of business involvement.

3.4 GT and the research questions

A project will be affected by a conflict between the principles of transactional leadership in enterprise management and the principles of the Agile manifesto. The questions listed above probe these issues to generate the data which the GT methodology then develops and abstracts into a useful theory. How the researcher does this fundamentally depends on their position in the ‘emergence versus forcing’ debate between Glaser and Strauss and the influence of the scientific method on the study. This consideration is adequately summed up by Birks and Mills;

“Glaser’s stance is based on his belief that for a problem to be of relevance, it must come from those for whom it has significance [...]. Strauss and Corbin’s position is typically much more pragmatic and relevant to the contemporary, professional research situation. Current requirements for the conduct of research, some of which echo the dominance of the scientific method, do of course impose the need for the researcher to demonstrate a focused research

topic. Whether this topic is expressed as a question, problem statement or hypothesis will be determined by the research design adopted by the researcher.” [4 p.21]

This study adopts the Straussian approach to GT (as indicated previously) and thus requires a sufficiently narrow core research question at this stage of the research. The methodology section below describes the GT process of using the research question to generate and initially code the data, then abstract it through higher levels of coding and ultimately produce an explanatory theory which answers these research questions.

4 MOTIVATION

The objective of this study is to improve Agile project environments in large organisations and help managers understand how they must disrupt their own thinking about leadership before they can disrupt their markets with innovative, Agile developed technology. Their ability to build successful, supportive and trusting environments will ensure that Agile projects flourish and their investment in the social capital of such projects will leverage the networks of assets and resources on which they depend.

The sociology of software development projects offers insights into the world of the people factors and the human interactions on a project. These dynamics can influence overall success, quality of product, duration of project and other key measurables. In the January 1982 issue of Computerworld, Barry Boehm asserted this argument in an article about managing software development costs (alongside an advert for a DT100 with a “high resolution” of 25x80!). “Personnel attributes and human relations activities provide by far the largest source of opportunity for improving software productivity” [46 p.61-63]. More recently, but prior to the Agile Manifesto, the importance of the social nature of software engineering was also put forward by Alfonso Fuggetta;

“Software engineering is primarily a social and creative process, where the creativity, skill, and co-operation of developers, users, and procurers determine the quality and effectiveness of the developed software.” [47 p.135]

Much more recently, in 2015, Wohlin et al formulated a theory of software engineering built on three sources of intellectual capital; human, social and organisational. The social capital component is the sum total of a network of relationships existing across, outside and within the boundaries of the project. The value offered by this element in the software engineering process is dependent on how well these networks leverage the resources and assets available in them. Thus;

“An organisation supports the creation of social capital when it brings its members together in order to undertake their primary task, to supervise activities, and to co-ordinate work, particularly in the context requiring mutual adjustment.” [48 p.230]

The concept of social capital in their theory of software engineering is broken down into internal and external aspects. Examples of internal are; “solving problems together”, “give each other feedback” and “common goals”; these describe the networks inside the environment of the project. External social capital examples are “collaboration with other units”, “collaboration with customers” and “collaboration with product owners and programme managers”. Typically in large organisations, such collaboration would have to occur across organisational boundaries, which are set by the spans and layers of the hierarchical structure of the company. Managers can either encourage such collaboration or negatively control the flow of information and stifle the social capital that software engineering projects need to succeed.

Managers in large organisations are trained to control their environments, within which they can hold people accountable (and be held accountable themselves) to achieve specific objectives. Understanding the relationship they have with Agile projects will help them create environments which are conducive for Agile while still achieving the expectations of their managerial roles. Answering the question “what is going on” with Agile and leadership in large organisations will inform a way of

thinking for senior managers about how they engage and oversee Agile projects. It may be that they have initiated the adoption of Agile; the insight they will gain from research on this in similar environments will assist them to be more successful in supporting those teams.

“Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done” [2]. This Agile principle appears to contradict the transactional management style of control and in this context is seen as a threat. This research will inform a better way for Agilists and managers to understand each other’s paradigms and find a better way that meets both their needs. The transactional manager would rewrite the above Agile principle as, “Incentivise subordinates to deliver according to deadlines. Tell them how to do it, set boundaries for their work and effectively monitor their progress.” Exponential organisations and high growth tech-based start-ups [49] are non-traditional and have been highly Agile in how they harness the power of software to achieve their goals.

A theory of Agile software development environments in large organisations will achieve a number of specific objectives;

1. The people influencing such environments with limited experience of Agile will gain an understanding of
 1. Agile principles and how their managerial roles can enable more successful projects;
 2. How they can achieve organisational imperatives as well as the benefits of innovative technology and
 3. Communication and trust-building methods that embody Agile principles and create the environment required by their software development teams.
2. Agile teams with limited experience of influencing upwards and outwards will gain an understanding of
 1. The influence of transactional leadership principles on managerial practices;

2. How they are positioned in an a broader organisational environment and how they are able to influence it and
 3. How to work effectively in a broader, plan driven environment while still achieving Agile project goals.
3. Traditionally large organisations aiming to harness innovative, Agile developed technology will
1. Run better project teams, despite the principles of these environments being at odds with their broader managerial ethos;
 2. Achieve a balance between their structure and the supportive environment needed by Agile teams;
 3. Reduce conflict between Waterfall and Agile teams and successfully address the people aspects of a hybrid approach.

5 METHODOLOGY: GT

5.1 Background

GT was first encountered by the author when researching the human aspects of software engineering. Studies of interest correlated with the author's observations in IT that there is not enough understanding of the coder behind the code.

Programming languages are essentially building blocks of logic, but it is still human beings that have to assemble them. Software engineering is thus also a social and creative process; relevant tools are needed from the humanities where there is greater expertise in understanding people. A new lens is required to properly understand the peopleware of Agile projects; a credible sociological method that is also rigorous and disciplined enough to withstand academic scrutiny. GT is starting to gain traction in IS studies, and was particularly well used in a study of self-organising Agile teams [36] where it offered a totally different angle on the human aspects of software development. The philosophical roots of this social inquiry method emphasise the importance of the researcher's position in their own work. The proposed study is not intended to be neutral and objective, but must proceed from, and relate back to the author's stand point on these issues.

5.2 Suitability

While GT is an appealing method and there is evidence of its growing popularity in IS, its suitability for use in software engineering research must also be confirmed. Three main reasons are given in “Grounded Theory for Geeks” [50] to arrive at a conclusion that “GT is appropriate as a research method to explore the human and social aspects of Software Engineering”. Each of these reasons are considered in light of the research objectives of the proposed study.

1. “Firstly, GT, used as a qualitative method, allows the study of social interactions and behaviour.” The primary interest with this research project is to explain what is happening when Agile is adopted in a traditional management environment. This world is about people, ie social issues, and the interactions and behaviour of the people that inhabit it are fundamental to understanding what is going on.
2. “Secondly, GT is useful when studying a relatively little explored areas” (sic). The state of research on human aspects of Software Engineering has been outlined above and given its (relative) infancy, is thus suitable for study using GT. While there is anecdotal evidence supporting observations in this area, there is insufficient formal and sophisticated qualitative research in the proposed area.
3. “Thirdly, GT allows generation of concepts and categories which have immense ‘grab’ for practitioners.” Any research that will really help managers achieve better success in creating Agile environments, will need to have high impact. GT moves beyond numerical results and speaks a rich and textured language from the people who speak through the data to the people who read the theory which emerges from it.

There are two main streams of thinking in GT that inform very different approaches to its usage. Although the two founders co-authored the original GT texts, they eventually disagreed on the very fundamentals they previously assumed were

equally valid positions on a continuum of options. Researchers differ as to how important this split is, either it is two strands of the same philosophical approach to qualitative research or they are so different and irreconcilable that one must declare a position early on in GT research as to which is used. Conservatively in favour of the latter, the Straussian version of GT will be used which is based primarily on the second of six contentions outlined in “Glaser or Strauss? Considerations for selecting a grounded theory” [51]. This relates to how the researcher is positioned in that “Strauss directs the researcher to be personally engaged with the research in an attempt to better describe and understand the world as the participants believe it to be”. The “detached observer” of Glaser’s first generation GT is inconsistent with the author’s professional immersion in the subject of interest as well as his firsthand experience of the issues under research.

The practical implications of this are that the researcher’s own position must be clearly understood, stated and then related to the emerging theory with sufficient theoretical sensitivity. An entire chapter of “Grounded Theory: A practical guide” [4] is devoted to this question and provides for the researcher;

“techniques to promote a reflexive approach to grounded theory research [that] will assure the quality and ethical integrity of your work by identifying how you and your history are present in the research process.” [4 p.62]

Most importantly the researcher’s methodological position determines whether they are a neutral collector of data from participants or if they also generate data with participants. Especially in the semi-structured interview process, the researcher is able to guide the interview in different directions, thus affecting emphasis in areas they feel strongly about. Additionally, the analysis of the data after interviews can be affected by the researcher’s position; the researcher has a relationship with the data, each are affected by the other. Memos are an important GT method and offer a valuable audit trail in the construction process of a grounded theory. The researcher’s reflexivity can be examined in the memos throughout the research process to determine how objectively the relationship with the data and participants is maintained.

5.3 Practicalities

5.3.1 Interviews

It is envisaged that research participants would be interviewed face to face and over Skype as part of the project. Focus groups would also be held and observations conducted of artefacts and key elements of Agile environments. Interviews would be semi-structured with guideline questions (see section on ethics clearance) which would be provided to them beforehand. In alignment with GT methods, the discussion with each research participant would be allowed to diverge and develop according to the flow of the interview. However, the broad themes of leadership, management and Agile would be maintained and used to guide the discussion appropriately. Two rounds of interviews will be planned as per the programme of work below. The first will cover all eligible participants and the second will be selective based on which data and topics I would like to revisit to confirm the emerging theory.

The core research question is stated in the positive sense but participants, focus groups and observational work will not be engineered accordingly. Examples of successful Agile environments in large organisations will be preferred for investigation but a positively stated theory can still be indirectly inferred from unsuccessful environments. This may not be as conclusive but the theoretical sampling will ensure there are sufficient points of triangulation about what to omit or do differently to convert an unsuccessful Agile project environment to a positive one. Equally there is no clear cut definition of a successful Agile project environment so this is also subject to interpretation (but can also be triangulated to some degree).

5.3.2 Computer Tool Support

It is expected that extensive use of computer tools would be used in the qualitative data analysis and organisation of references for the research project. Nvivo would house and manage all the data, memos and research sources, most importantly the interview recordings and transcripts. The organisation of references will be done on EndNote with integration to Nvivo and Microsoft Word which will be the word processor used for all documentation. In line with other GT studies, data analysis, constant comparison, coding and core categorisation would not be limited to Nvivo, but it is expected that Microsoft Excel would also be used. While some GT work has not worked well on computer tools such as Nvivo, the advice given in these studies was that tools have to be introduced from the start. All the computer tools above have been used extensively to ensure sufficient learning curve and appropriate usage. An important insight from most GT literature is that the researcher still needs to engage directly with the data. Much like on an Agile project, the visual and tactile nature of wipeboards and sticky notes make them just as useful as the most advanced computer based tools.

5.4 Design

Birks and Mills list the main methods of GT as;

“initial coding and categorization of data; concurrent data generation or collection and analysis; writing memos; theoretical sampling; constant comparative analysis using inductive and abductive logic; theoretical sensitivity; intermediate coding; identifying a core category; and advanced coding and theoretical integration.” [4 p.10]

Many studies claim to be GT however are selective in their application of these methods. As a novice GT researcher, use of this design is both encouraging and challenging. On one hand it provides a roadmap for the research project and a comprehensive tool box for building and using the sociological lens I desire to rigorously interrogate my field of interest. As a novice GT researcher the author will

be on unfamiliar territory and thus rely heavily on practical guidance from experts to correctly apply its methods.

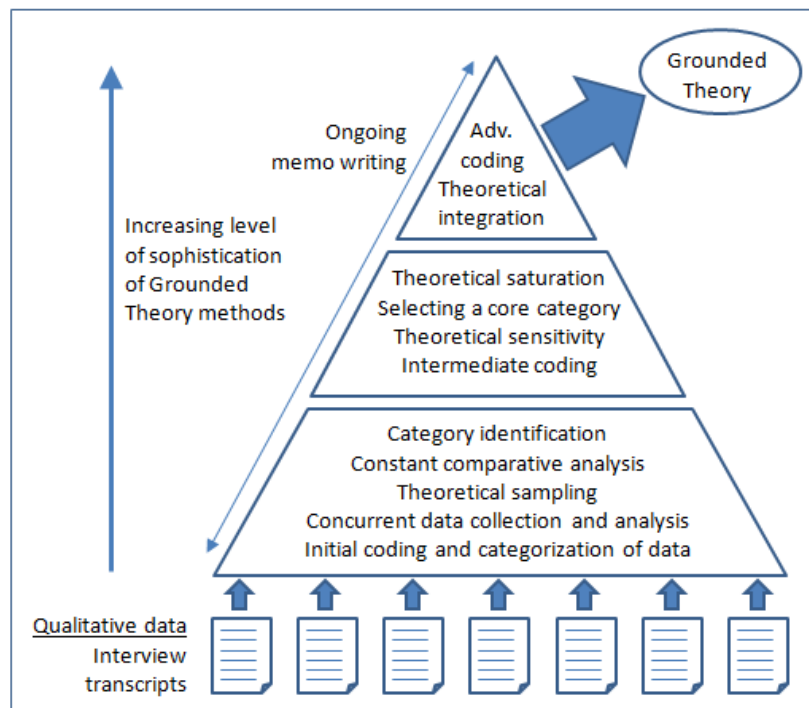


Figure 6: GT methods

The primary GT methods are shown in Figure 6 as a pyramid model with increasingly sophisticated methods being used as the study progresses. The primary qualitative data is the transcripts from the interviews, to which the GT methods are applied in three major stages as shown by the sections, ultimately reaching the apex when the theory itself is generated and presented in the final report. Subsequent sections briefly introduce these primary GT methods and explain how they will be applied to the proposed research. Much reliance is placed on the final sections of chapter 1 of Mills and Birks [4] and any practical considerations are highlighted where necessary.

5.4.1 Initial coding

Once an interview with a research participant is completed, the transcript can be subjected to initial coding (sometimes just called coding) as the first step of analysing the body of primary data. Words or groups of words of the interview are scrutinised for important “identifying anchors that highlight the key aspects of the data being

gathered and analysed”⁵. First generation grounded theorists were quite abstract in their notion of coding, and even what codes are themselves. Subsequent literature is a lot more understandable and practical. In this study I plan to audio record the Skype interview and automatically download a transcript in Nvivo, which is a paid service that saves a tremendous amount of time. The transcript and audio of the interview are then linked and available as a combined internal source in Nvivo where it is then possible to code “line by line”. Nvivo works on the concept of nodes which can be grouped or linked together, ie an individual Nvivo node would be considered an initial GT code.

5.4.2 Concurrent data generation

This aspect of the research is what differentiates GT from other types of qualitative data collection and analysis. The GT researcher is required to work iteratively by collecting some data, analysing and coding it and repeating this all again.

Interestingly, GT research has many hallmarks of Agile Software development, particularly this iterative approach to data collection and analysis. Much like iteration in Agile offers a way of incorporating constantly changing requirements, so GT data collection and analysis allows the evolution of a theory which is perpetually grounded in the data. Other methods follow a linear process of collecting all the data then analysing it in one go or developing and stating a theory up front and then collecting and analysing data to test the theory. Practically in GT this means that the data will be recorded, transcribed, coded and analysed for each interview as they are completed without waiting for all data from the interviews to be available before analysis starts. As a researcher this allows an understanding of what is going on to evolve as an involved and committed participant in the research process itself.

5.4.3 Memos

Although on the surface, memos are similar to a common generative writing process for any study or writing project, increased reading of GT has shown how fundamental it is to this methodology. Memos are essentially semi-structured journal entries that

⁵ Lynlee Howard-Payne’s Post-grad workshop at Wits on Grounded Theory based on her own use of this research method in studying medical male circumcision.

document the researcher's journey through their field. These memos are the researcher's record of thinking and can document any aspect, but at this stage should capture the links between categories of codes and also record the initial perceptions and insights from interviews. Like any generative writing exercise they should be completed as soon after the researcher's judgment has been exercised and without any self-criticism of content. Importantly, according to Mills and Birks, [4] "Your memos will in time transform into your grounded theory findings". Fundamental to GT is the idea that theory is not so much stated up front for testing by research but rather allowed to emerge during the research process such that it is grounded in the data itself. As such, it is the content of the GT researcher's memos which will describe the golden thread that leads from initial question to final theory.

5.4.4 Constant comparative analysis

Codes and categories (groups of initial codes) are observed and developed from interviews, then they are compared against each other within the same interview and across others. This constant comparison between all the elements that arise from the research process causes the data to be abstracted to higher levels of the pyramid in Figure 6 above. Rather than generating a large set of data then analysing it all in one go (linearly like Waterfall), GT requires that small chunks are digested, understood and incorporated in the emerging theory as the researcher progresses. A common feature of GT presentation is the use of quotes from interviews to substantiate aspects of the theory as it is being explained. Although the final theory is the highest level of abstraction of the data, this technique in explaining it shows that constant comparative analysis has taken place. Raw data at the bottom of the pyramid is pulled to the top and compared with the final product to calibrate and verify it. Every part of the landscape of a GT project must make sense in the context of the rest of it; this is only possible when constant comparative analysis takes place consistently and continuously.

5.4.5 Theoretical sampling

The researcher makes ongoing decisions about which data to analyse and as part of their constant comparison will switch accordingly. These ongoing judgments are supported by the writing of quality memos because they record the structured thinking that guides the ongoing changes and decisions in the research process. The theory emerges during this dynamic process of collecting and analysing data and deciding what data to collect and when. A category is said to become saturated when no new insight is obtained from collecting and analysing further data. Categories which are becoming saturated become clear through this iterative process of theoretical sampling and highlights which categories require more information about their properties, conditions of existence, dimensions and relationships with other categories. With regards to the planned interviews, theoretical sampling requires a high degree of flexibility in scheduling. As the theory of Agile project environments emerges, interviews may be repeated and previously analysed data re-assessed. In the programme of work I have planned to do an initial set of interviews with all participants and then a second round with certain participants, as chosen based on this concept of theoretical sampling.

5.4.6 Theoretical sensitivity

As a researcher taking an interpretivist approach to social inquiry in my topic, it is crucial to decide and declare how I position myself philosophically before entering the field. From this stand point, theoretical sensitivity guides my research journey and ensures I remain attuned to the data and the emergent theory as I proceed. To achieve this, my self-awareness must be coupled with insight into my field and my own history of inquiry, criticism and thinking about the world around me. As a researcher I am the sum total of my experiences, especially of the software development industry and the application of its methods in business. I cannot be completely objective and stand outside of my work as a neutral observer but I can account for this in my research so that my results have integrity. I must be sensitive to this in the research process and it thus informs how I decide which data is more important.

5.4.7 Intermediate coding

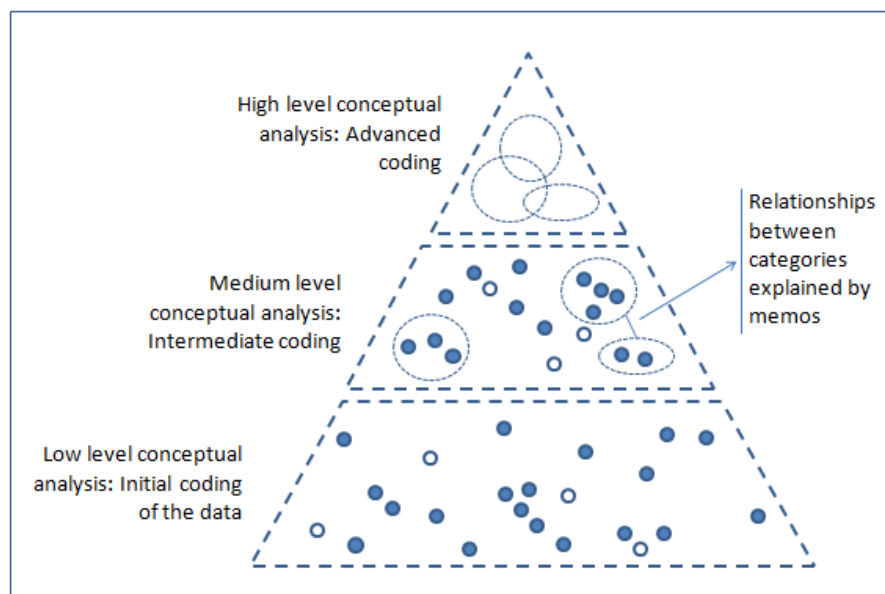


Figure 7: GT conceptual framework and coding activities

The initial coding step at the bottom of the pyramid develops the anchors for abstracting the raw data from the interview transcripts as described above. Although grouped into logical categories, these codes are often unrelated and only arbitrarily linked to each other. “Intermediate coding reconnects the data in ways that are conceptually much more abstract” [4 p.12], plus it relies on, and builds on the preceding steps and methods, thus increasing the level of sophistication in the research. The goal of a GT study is to develop “a high level conceptual framework that possesses explanatory power supported by advanced analytical processes” [4 p.91]. Intermediate coding is thus a medium level stage where the categories of concepts developed at the low level are now linked, integrated or collapsed entirely. Relationships between categories can be fully developed in memos that assist the researcher in coherent, theoretically sensitive and logical analysis of the data and its emerging theory.

The structure and output of a GT study is an attractive research model for this project but the historical development of its terminology regarding coding can be

confusing. Birks and Mills [26] creates a logical grouping of the major names for different coding:

4. **Initial coding:** Coding and comparing incidents (Glaser & Strauss 1967), Open coding (Glaser 1978, Strauss & Corbin 1990, 1998) and Initial coding (Charmaz 2014)
5. **Intermediate coding:** Integrating categories and properties (Glaser & Strauss 1967), selective coding (Glaser 1978), axial coding (Strauss & Corbin 1990, 1998) and focused coding (Charmaz 2014)
6. **Advanced coding:** Delimiting the theory (Glaser & Strauss 1967), theoretical coding (Glaser 1978, Charmaz 2014) and selective coding (Strauss & Corbin 1990, 1998,).

The clear structure of this grouping is easier to understand but should not obscure the original theoretical intentions and nuances of each. The richness of methods and theoretical tension between grounded theorists also makes it such an interesting means of social inquiry

5.4.8 Identifying a core category

The research process follows logical steps of ever increasing abstraction of the data through more and more sophisticated methods as shown in the previous two figures. During this process, categories emerge from the grouping of concepts that unify and integrate the data, eg in Hoda's work on self-organising Agile teams [36], these were the actual roles identified in such teams; mentor, co-ordinator, translator, champion, promoter and terminator. The properties, dimensions and relationships between these categories is the explanatory essence of this work. The fact that roles emerged as the categories themselves was testimony to the validity of the method and the theoretical sensitivity of the researcher in her work. The core category is one that emerges as the "main concern or problem" [52] for research participants and in that study was the research problem itself; self-organising Agile teams. All other

categories became sub-categories to this and the thrust and relevance of all results, discussion, future research, related topics and observations was related back to this.

The idea of developing a core category is one of the difficulties that is anticipated in this study which is also the author's first use of GT. The intention is to study the phenomenon of management and Agile in traditionally large organisations, however one must be sensitive to the data and flexible enough to allow a valid and meaningful core category to emerge. This central idea must have impact and significance but equally not deviate too much from the sufficiently narrow core research question originally articulated. The core category may emerge as being very close to the intended research problem which may be the result of unintended manipulation of the data or theoretically insensitivity.

5.4.9 Advanced coding

At the top of the pyramid diagrams above, the most sophisticated method of ultimately converting data into theory is accomplished by advanced coding. This is how GT takes a study beyond observations and conclusions and achieves usefulness for research. It is the explanation of the theory itself; the creation of meaning and new knowledge for its readers. Advanced (or theoretical) coding is the bridge between the findings of a study and the explanation of those findings through hypothesis. In practical terms it elevates the core category from evidence-based to theory-based; it allows the researcher to present their findings in a way that hypothesises about what is going on. In doing so, there is a fine balance between creativity that presents theory in an imaginative way, and discipline that demonstrates theoretically sound research. There are a couple of techniques that will be considered for achieving this.

The first is storyline, which is one example of the disagreements between Glaser and Strauss, the original co-founders of GT. It is commonly used in interpretive methodologies and ethnography to convey findings of a more descriptive nature. Another option is a strong reliance on diagramming and visual modelling to explain relationships between categories and concepts which can even be extended to a

final, all-encompassing high level model of what is going on. Diagramming has been used in this proposal as well as much of the pre-work discussion documents with supervisors. Suitably, for this topic it is also a very common method in software engineering and in particular, Agile product design. The primary goal of this step in the research is to credibly elevate (and concentrate) the empirical results of the research into an elegant and practical theory, “whilst still respecting all the data by offering a way to explain connections and conceptualise relationships [between the data]” [4].

5.4.10 Generating theory

Theory in GT is defined as “an explanatory scheme comprising a set of concepts related to each other through logical patterns of connectivity” [4]. This is somewhat different from the predictive nature of scientific theory; the measure of GT is its ability to explain, rather than predict, concepts. Throughout the GT research process, categories of codes are refined, related, described and constantly compared to each other and the rest of the data. A core category should also have arisen which provides the thrust of the theoretical scheme and a point will be reached when these categories are saturated and the further collection of data adds nothing new to the emerging theory. This marks a shift from coding activities to the generating of theory such that the researcher steps back from the data and adds the explanatory essence (this is also most likely when wipeboards and sticky notes are more useful than computer tools). High quality, sorted memos that track the full journey are also crucial at this point; they are the field notes that now become the text book. Together with the outputs from advanced coding described above, the process of generating the theory is the act of creating the knowledge; explaining what is going on.

6 ETHICS

6.1 Guidelines and approvals

Full ethics clearance for human research through interviews will be sought for this study according to the Wits “Guidelines for Human Research Ethics Clearance

Application (non-medical)". It is not expected that permission from any authorities will be required as individuals are being interviewed for their experiences of the adoption and use of Agile in the environment within which they work. There are no institutions that are being researched in their own right however it is likely that institutions involved in Agile, and projects, locations etc where participants have worked will come up during the research. Any such references will be made in generic terms and specific information about commercially sensitive aspects of these operations will not be included in the results of the research. There is also no requirement to gain physical access to premises, all interaction with research participants will be virtual; over Skype, email and limited online surveys.

Signed, formal consent forms will be gathered from suitable people who are willing to participate. According to the university guidelines, this covers a description of the research, a list of what has been communicated to them and a clear statement about what the participant is consenting to by becoming involved in the research. A specific provision will be included for allowing the participant to agree to the interview process over Skype as well as that the interviews will be recorded which is necessary for the data gathering and analysis. Signed and dated forms will be retained for all participants in the research and as required, an example of one form will be submitted as part of the ethics clearance process. It is not expected that any of my research participants will be under the age of 18, thus the Assent Form and involvement of any of the parents will not be required. "Standard Operating Procedures for the Human Research Ethics Committee (HREC, non-medical) at the University of the Witwatersrand, Johannesburg" defines the governance of submissions. Section 3.5.1 deals with documentation required for application to HREC for non-medical research and a new online process is now available for the submission of these documents electronically. This process will be complied with accordingly and an initial review of both the online submission process and the governance requirements does not indicate that there will be any requirements for exceptions.

6.2 Research participants

Initial communication about my research has taken place with a wide group of suitable and interested Agile practitioners, managers and relevant thought leaders. Background reading of similar work suggested possible delays regarding the recruitment of research participants and the intention has been to get this process going as early as possible. I placed notices on relevant Twitter and LinkedIn forums briefly explaining my research and requested interested people to get in touch with me after which I exchanged emails and assessed their suitability. From this exercise I have a list of 60 suitable and willing people and I have had exploratory, unstructured Skype interviews with two on this list, mainly as practice and initial experimentation for a typical GT interview. It is expected that the number of research participants required and which is possible for this study will be less than the 60 on this list. Subject to ethics approval, the next step is to scrutinise their suitability and formally complete the ethics clearance process with them. The “snowball” method will also be used to find suitable research participants; as participants on the list are interviewed they will be asked for suitable people in their network that can be interviewed and the list will “snowball”.

The list of research participants will also depend on the approved topic, hypothesis and questions; this will be finalised through the completion and approval of this proposal. These aspects were broadly defined and communicated when I first engaged interested research participants and some of these interactions also helped guide and structure my thinking. I started with quite a wide field of interest, which has become more specific and will thus be a filter for which people to include in the research – based on their experience and interest. For example the research is now focused on understanding what is going on with Agile and the management of Agile environments in large organisations, meaning that I will require only research participants who have knowledge and experience of this kind of environment.

Worth noting is that an online email management system called Getresponse will be used to communicate with research participants which is more commonly used for digital marketing to signed up subscribers. It is equally suitable for research communication with consenting participants and provides a very professional process and output for communication. It also allows for the secure management of

participant information, the tracking of their receipt of the communication and the template-based creation of such emails. Two examples of the formal emails I have sent to the full list of 60 interested research participants are in Appendix A. Their contents are an indication of the developing nature of this research at those points in time.

6.3 Anonymous participant information

Drawing on common practice in similar studies [15 , 16 , 17], the final report will also include a list of the included research participants, with all information made completely anonymous. While ensuring the anonymity of the research participants this detail is also important because it aims to demonstrate that the list of participants are suitable to the objectives and scope of the study, based on their experience and role. Of those three studies, the following are some of the common headings that I would intend to include in this list;

1. Relevant experience in years, in some cases differentiating the various Agile roles that individuals may have held;
2. Primary / current Agile role; Scrum master, coach, manager, developer, analyst;
3. Agile methods experienced; Scrum, XP, Kanban;
4. Number of employees in their current organisation.

The last metric is crucial to the focus of this study and I would expect to include others that demonstrate the intended scope of traditionally large organisations. These could be total revenue, countries of operation, nature of business etc but should not be sufficient detail to triangulate the name of the company. All three studies included a generic number for each participant, this is partly so that quotations in the final report can be referenced to the anonymous information of that person. Senior managers would be included in the participants, in order to gain a customer and business stakeholder view. This data will be crucial to balance the weight of research in both directions across the boundary of the Agile environment, ie outside-in and inside-out.

6.4 Interview questions

For the purpose of ethics clearance, the intended interview questions are included below and will loosely guide my Skype discussions with participants. These complement the research questions above, and are intended to prompt discussion which will generate sufficient data for GT analysis.

1. Please tell me about Agile in the organisation where you work / coach / consult? What is the organisational culture there? How long has Agile been used, how was it adopted?
2. What are the Agile projects in that environment that you have been involved with? What Agile methods are used and why / how successfully?
3. What was your role on these projects, how long were you involved and what was the results / effect / impact of your involvement.
4. How was the environment around these projects set up? Who set it up, what role did senior / line / project management have in doing this?
5. What issues did you / the Agile team / others in the environment experience as a result of interactions with management and / or responses to their involvement with the project?
6. How was management perceived among the Agile team? How well did management demonstrate understanding / acceptance / knowledge of the Agile principles?
7. Describe an example that you observed first hand of a positive / negative interaction between the Agile team / project and management in the immediate environment? What impact did this have on the team?
8. How would you say the topics of trust, communication, leadership and support were handled on the project / by management / in the environment?

In developing the above questions the exploratory Skype discussions with two of the prospective research participants has been very useful. This greatly assisted in understanding what are the real dynamics and expected key topics of such a conversation. The ethics of human research is a key feature of the writings about GT

and also features strongly in the results and reports of studies that use its methods. GT is a methodology that originated in the medical field where clinical trials and other human research is very frequent and related issues are well understood.

7 DELIVERABLES AND PROJECT SCHEDULE

Major tasks and deliverables for the research project are set out in Table 1 below which covers the five years of the study. GT studies are notoriously iterative so it is expected that work activities will overlap, particularly the data collection and analysis. Despite this, it is equally important to ensure the work stays on track and progress is made according to a realistic schedule. The letter confirming my PhD registration (dated 4th March 2015) requires submission of the final report by 1st March 2019. The plan below loosely incorporates the various GT methods and research steps required to achieve this. Specific dates are indicated where relevant and Q1 = Jan-Mar (incl.), Q2 = Apr-Jun, Q3 = Jul-Sep and Q4 = Oct-Dec.

Table 1: Programme of work

Tasks and deliverables	2015	2016	2017	2018	2019
Do preliminary literature review and draft proposal	Q4				
Complete and submit research proposal		Q1			
Complete and submit online ethics approval process		Q1			
Present final proposal for approval to proceed with research		Q2			
Finalise participants list and		Q2			

Tasks and deliverables	2015	2016	2017	2018	2019
schedule of round 1 interviews					
Receive initial batch of signed research participant consent forms		Q3			
Conduct round 1 interviews, analyse and code data, theoretical sampling, memoing and written follow-ups with participants		Q3-Q4			
Complete literature survey, draft chapter			Q1-Q4		
Finalise and schedule round 2 interviews (selected participants)		Q4			
Conduct round 2 interviews, analyse and code data, theoretical sampling, memoing and written follow-ups with participants			Q1-Q2		
Compare emerging theory with literature, update literature survey draft chapter			Q3-Q4		
Write up GT Part A & B				Q1	
Write up GT Part C & D				Q2	
Write up Abstract, Introduction, Conclusion and finalise literature				Q3	

Tasks and deliverables	2015	2016	2017	2018	2019
survey chapter					
Finalise report; contents, figures, tables, lists of abbreviations, references, bibliography and acknowledgments				Q4	
Format, bind and submit report with copies according to university requirements					1 st March
Process corrections and post-submission requirements Oral presentation, examination as required					Q1-Q2

To ensure regular focus and consistent discipline of completing the plan, there will be a weekly email status report to supervisors, covering;

- Maximum five specific items of work;
 - completed the previous week and
 - due the coming week;
- General issues, decisions required and questions;
- Relevant work and deliverables to be attached as required.

In the spirit of efficient and effective communication with my supervisors, the intention will be that everything is channelled through this weekly email. For the student, this will be the “drumbeat” of the project and for my supervisors, this will ensure emails are kept to a minimum. At certain periods in the study more communication will be required, as well as face to face meetings and these can be arranged as required. Memoing, ongoing reading in all relevant fields and relevant

blog posting (as a public form of generative writing) will continue throughout the period of the research project. I also plan to complete at least 3 peer reviewed articles in relevant journals, (to be agreed with supervisors), and present at 2 appropriate conferences (1 local, 1 international) on related topics.

8 PROJECT BUDGET

No funding is required for this research as most costs will be paid for by the researcher's employer who has provided a study bursary. The bursary is subject to re-approval in each year of study and all expenses (books, transcripts, research related expenses) will need to be submitted on the company's expense claim system for checking and approval. Travel to conferences and associated accommodation expenses will need to be paid for personally, unless such travel can be combined with a legitimate business objective in which case it will be directly paid for by the employer. The table below provides a high level summary of the total costs of the project, of which c.R30,000.00 has already been spent.

Table 2: Summary project budget

Items	Total Cost
1. Transcripts of 50 interviews (average R2000.00 each)	R100 000.00
2. 10 books (average R400.00 each)	R4 000.00
3. 3 conferences: 1. 2 local (no cost) 2. 1 international (travel and accommodation)	R50 000.00
4. Professional editing services	R5 000.00

Items	Total Cost
5. Binding, paper, ink, photocopies and related	R2 000.00
6. Nvivo yearly renewal (R1500 p/a for 2016-2019)	R6 000.00
7. EndNote once off purchase	R2 700.00
8. Contingency	R10 000.00
Total	R179 700.00

9 FINAL REPORT

The preliminaries, chapters, structure and layout of the final report will comply with the style guide of the School of Engineering and the Built Environment. Section 2 covers the structure and form of these, providing the essential elements as below, in the order they would appear.

- Title and title page
- Candidate's declaration
- Abstract
- Dedication
- Acknowledgements
- Contents
- List of figures
- List of tables
- List of symbols
- Nomenclature / list of acronyms
- Introductory chapter
- Central chapters

- Concluding chapter
- References
- Bibliography
- Appendices

Importantly, the central chapters section “are the thesis – they make the candidate’s contribution to knowledge”⁶. In this study this portion will present the main theory that has emerged in the research and which broadly achieves the objectives and answers the questions listed above. In Hoda’s final report on a GT study of self-organising Agile teams [36] the individual chapters of this section were the actual roles (ie main categories of coded data) that were determined through the research. My final report may adopt a similar approach but equally a core category may arise and dominate the results of the study in which case this will be the central concern and golden thread of the thesis. Just as the stringent GT methods guide the research process from the start, so too the final results should reflect the nature and philosophy of this means of social inquiry. Ultimately the final report will comply with faculty requirements and aim to present the results in a clear and logical fashion that will withstand academic scrutiny and prove useful to readers who will benefit from the new knowledge it creates.

⁶ Section 2.2.2: *Style Guide for Theses and Dissertations: Graduate School of Engineering and the Built Environment*. The final report will also comply with the booklets *Reports & Presentations – A Communications Manual for the Engineer (Blue Book)* and *School of Electrical and Information Engineering (Red Book)*

REFERENCES

- [1] Abernathy, P. *Hook, Line and Sinker: The Role of Line Management in Relation to Agile Teams*. in *Agile Conference, 2009. AGILE '09*. 2009.
- [2] Alliance, A., *Agile manifesto*. Online at <http://agilemanifesto.org>, 2001. **6**(6.1).
- [3] *Social Contract*. 2011.
- [4] Birks, M. and J. Mills, *Grounded theory: A practical guide*. 2011: Sage publications.
- [5] Ismail, S., M. Malone, and Y. Van Geest, *Exponential organizations: Why new organizations are ten times better, faster, and cheaper than yours (and what to do about it)*. 2014, Diversion Books.
- [6] Goodwin, T., *The Battle Is For the Customer Interface*, in <http://techcrunch.com/2015/03/03/in-the-age-of-disintermediation-the-battle-is-all-for-the-customer-interface/>. 2015, TechCrunch.
- [7] Fowler, M. and J. Highsmith, *The agile manifesto*. Software Development, 2001. **9**(8): p. 28-35.
- [8] Brooks, F.P., *The mythical man-month*. Vol. 1995. 1975: Addison-Wesley Reading, MA.
- [9] Highsmith, J. *History: The Agile Manifesto*. 2001.
- [10] Landis, E.A., D. Hill, and M.R. Harvey, *A Synthesis of Leadership Theories and Styles*. Journal of Management Policy & Practice, 2014. **15**(2): p. 97-100.
- [11] Maloş, R., *THE MOST IMPORTANT LEADERSHIP THEORIES*. Annals of Eftimie Murgu University Resita, Fascicle II, Economic Studies, 2012: p. 413-420.
- [12] Drucker, P.F., *The age of discontinuity: Guidelines to our changing society*. 2011: Transaction Publishers.
- [13] Glaser, B.G. and A.L. Strauss, *The discovery of grounded theory: Strategies for qualitative research*. 2009: Transaction Publishers.
- [14] Matavire, R. and I. Brown, *Investigating the use of "Grounded Theory" in information systems research*, in *Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and*

- Information Technologists on IT research in developing countries: riding the wave of technology*. 2008, ACM: Wilderness, South Africa. p. 139-147.
- [15] Dorairaj, S., J. Noble, and P. Malik. *Understanding Lack of Trust in Distributed Agile Teams: A grounded theory study*. in *Evaluation & Assessment in Software Engineering (EASE 2012), 16th International Conference on*. 2012.
 - [16] Gandomani, T.J. and M.Z. Nafchi, *An empirically-developed framework for Agile transition and adoption: A Grounded Theory approach*. Journal of Systems and Software, (0).
 - [17] Adolph, S., P. Kruchten, and W. Hall, *Reconciling perspectives: A grounded theory of how people manage the process of software development*. Journal of Systems and Software, 2012. **85**(6): p. 1269-1286.
 - [18] Adolph, S., W. Hall, and P. Kruchten, *Using grounded theory to study the experience of software development*. Empirical Software Engineering, 2011. **16**(4): p. 487-513.
 - [19] Boehm, B. and R. Turner, *Management challenges to implementing agile processes in traditional development organizations*. Software, IEEE, 2005. **22**(5): p. 30-39.
 - [20] Pirzadeh, L., *Human Factors in Software Development: A Systematic Literature Review*. 2010.
 - [21] Dingsøyr, T., et al., *A decade of agile methodologies: Towards explaining agile software development*. Journal of Systems and Software, 2012. **85**(6): p. 1213-1221.
 - [22] Faraj, S. and V. Sambamurthy, *Leadership of information systems development projects*. IEEE Transactions on Engineering Management, 2006. **53**(2): p. 238-249.
 - [23] Jaques, E., *In praise of hierarchy*. Harvard Business Review, 1990. **68**(1): p. 127-133.
 - [24] Storey, J. and J. Storey, *Changing theories of leadership and leadership development*. Leadership in organizations: Current issues and key trends, 2004: p. 11-38.
 - [25] *Manifesto, CHAOS*. 2011, The Standish Group.
 - [26] Eveleens, J. and C. Verhoef, *The rise and fall of the Chaos report figures*. IEEE Software, 2010. **27**(1): p. 30-36.

- [27] Brenner, R. and S. Wunder. *Scaled Agile Framework: Presentation and real world example*. in *Software Testing, Verification and Validation Workshops (ICSTW), 2015 IEEE Eighth International Conference on*. 2015.
- [28] Mouton, J. and E. Babbie, *The practice of social research*. Cape Town: Wadsworth, 2001.
- [29] Randall, R.M., *Agile at IBM: software developers teach a new dance step to management*. *Strategy & Leadership*, 2014. **42**(2): p. 26-29.
- [30] Denning, S., *Why Agile can be a game changer for managing continuous innovation in many industries*. *Strategy & Leadership*, 2013. **41**(2): p. 5-11.
- [31] Bonner, N.A., *Predicting leadership success in agile environments: An inquiring systems approach*. *Academy of Information and Management Sciences Journal*, 2010. **13**(2): p. 83.
- [32] van Waardenburg, G. and H. van Vliet, *When agile meets the enterprise*. *Information and Software Technology*, 2013. **55**(12): p. 2154-2171.
- [33] Neufeld, D.J., L. Dong, and C. Higgins, *Charismatic leadership and user acceptance of information technology*. *European Journal of Information Systems*, 2007. **16**(4): p. 494-510.
- [34] Nixon, P., M. Harrington, and D. Parker, *Leadership performance is significant to project success or failure: a critical analysis*. *International Journal of Productivity and Performance Management*, 2012. **61**(2): p. 204-216.
- [35] Laanti, M., O. Salo, and P. Abrahamsson, *Agile methods rapidly replacing traditional methods at Nokia: A survey of opinions on agile transformation*. *Information and Software Technology*, 2011. **53**(3): p. 276-290.
- [36] Hoda, R., J. Noble, and S. Marshall, *Self-Organizing Roles on Agile Software Development Teams*. *Software Engineering, IEEE Transactions on*, 2013. **39**(3): p. 422-444.
- [37] Hoda, R., J. Noble, and S. Marshall, *Supporting self-organizing agile teams what's senior management got to do with it?*, in *Lecture Notes in Business Information Processing*. 2011, Springer Verlag. p. 73-87.
- [38] Grossman, F., et al. *One XP experience: introducing agile (XP) software development into a culture that is willing but not ready*. in *Proceedings of the 2004 conference of the Centre for Advanced Studies on Collaborative research*. 2004. IBM Press.

- [39] Jantunen, S. and D.C. Gause, *Using a grounded theory approach for exploring software product management challenges*. Journal of Systems and Software, 2014. **95**: p. 32-51.
- [40] Javdani Gandomani, T., et al., *The impact of inadequate and dysfunctional training on Agile transformation process: A Grounded Theory study*. Information and Software Technology, 2015. **57**: p. 295-309.
- [41] Adolph, S. and P. Kruchten. *Generating a useful theory of software engineering*. in *Software Engineering (GTSE), 2013 2nd SEMAT Workshop on a General Theory of*. 2013.
- [42] Boehm, B. and R. Turner, *Balancing agility and discipline: A guide for the perplexed*. 2003: Addison-Wesley Professional.
- [43] Sliger, M., *A project manager's survival guide to going agile*. Rally Software Development Corporation, 2006.
- [44] Greenleaf, R.K., *Servant leadership*. 1977, New York: Paulist Press.
- [45] Stray, V., *An Empirical Investigation of the Daily Stand-Up Meeting in Agile Software Development Projects*. 2014, University of Oslo.
- [46] Boehm, B., *Keeping a Lid on Software Costs*, in *Computerworld Magazine*. 1982. p. 61-63.
- [47] Fuggetta, A., *Rethinking the modes of software engineering research*. Journal of Systems and Software, 1999. **47**(2–3): p. 133-138.
- [48] Wohlin, C., D. Šmite, and N.B. Moe, *A general theory of software engineering: Balancing human, social and organizational capitals*. Journal of Systems and Software, 2015. **109**: p. 229-242.
- [49] Fried, J., D.H. Hansson, and M. Linderman, *Getting Real: The smarter, faster, easier way to build a successful web application, 37signals*. 2009, Chicago.
- [50] Hoda, R., J. Noble, and S. Marshall, *Grounded theory for geeks*, in *Proceedings of the 18th Conference on Pattern Languages of Programs*. 2011, ACM: Portland, Oregon, USA. p. 1-17.
- [51] Howard-Payne, L., *Glaser or Strauss? Considerations for selecting a grounded theory study*. South African Journal of Psychology, 2015.
- [52] Glaser, B.G., *Theoretical sensitivity: Advances in the methodology of grounded theory*. 1978: Sociology Pr.

APPENDIX

The first two emails I sent to interested / prospective research participants were an introduction to myself and then an update on the research project.

There are key benefits of using a tool such as the Getresponse email management system:

9. Opening of emails can be tracked;
10. Information is kept well organised;
11. Templates ensure consistent and professional communication;
12. Participants can very easily unsubscribe from the project.

From: Peter Alkema <peter.alkema@telkomsa.net>
To: *Prospective Research Participant*
Sent: Wednesday, 14 January 2015, 1:17
Subject: Update #1: Introduction (Agile Research Participation)

Agile Research Participation

[[firstname]],

Update #1: Introduction

Thank you very much for agreeing to participate in my PhD research which will be focused on how leadership impacts Agile Software Engineering teams. I will be staying in touch with you via email and my intention is to keep you updated and ensure that communication with my research participants is efficient and organised.

Your involvement will be 2-3 Skype interviews and minor follow up information via email. In due course I will also need written consent for joining the research and I'll also be sending you ethical approvals from my university once the project kicks off. I am very grateful for your support but you can opt out at any time just please let me know.

I am still looking for more research participants so would be most grateful if you can pass on this request to other Agile practitioners in your network, user groups etc. You will be contributing to this exciting IT field and if, like me, you are passionate about improving software engineering then I hope you are looking forward to this.

Just a short word about the method I hope to use... Grounded theory is a systematic methodology mainly used in social sciences but increasingly finding application in many other fields, including software engineering. It is a rigorous process to develop a theoretical framework from qualitative data such as interviews.

This is where you come in... During our Skype interviews you'll be sharing your experiences of Agile, and how you have understood / observed / influenced the impact of leadership. This is intentionally broad and allows for a diverse group of people to share their experiences, out of which a theory of Agile & Leadership will emerge.

Yours sincerely,

A handwritten signature in black ink, appearing to read 'P. Alkema', with a large, stylized loop at the end.

Peter Alkema

PS. The research rationale.....

The strong emphasis that Agile places on human interactions requires that we understand much better how different models of leadership affect Agile teamwork and output. Agile values human interaction; Leadership is all about inspiring people. Do we know enough about how these work together?

From: Peter Alkema <peter.alkema@telkomsa.net>
To: *Prospective Research Participant*
Sent: Thursday, 2 July 2015, 23:28
Subject: Update #2: Staying in touch (Agile Research Participation)

Agile Research Participation

[[firstname]],

Update #2: Progress

Earlier this year I sent an introduction email thanking you for offering to participate in my research about Agile Software Development and Leadership. The purpose of this email is primarily to stay in touch but also to give you some insights and a brief progress update on my research.

I've been mostly reading up in the literature, refining research themes and learning more about Grounded Theory as a qualitative research methodology. It's always an exciting journey to venture into new areas and explore the work that other people have already been doing.

I'm really fascinated by Grounded Theory; it's main premise is that research is also about generating new theory from data, not just testing existing theory. This method is popular in the humanities but growing in the sciences and a good summary of how to use it in software engineering is [here](#).

On the IT leadership side of my research I have also used the opportunity to start blogging some thoughts and you can read my articles at <http://www.peteralkema.com> - I'd love some feedback! Every day I realise how important are the leadership and human aspects of IT and that's why we are all passionate about Agile, it makes software development about people as well as technology.

As previously indicated, the research activity with you will start later in the year, once my proposal is accepted and I have ethics clearance from the university. If you would not like to participate any more, please let me know or simply unsubscribe by clicking the link at the bottom of this email.

Yours sincerely,

A handwritten signature in black ink, appearing to read 'P. Alkema', enclosed within a thin black rectangular border.

Peter Alkema

PS. The Agile Manifesto and Leadership.....

While the manifesto does not actually mention the word "leadership", the wording and underlying principles correlate with leadership practices. "Individuals and interactions over processes and tools" and in the 12 principles we find concepts such as trust, sustainable, face to face, environment and support.