Exploring Adoption and Use of Agile Methods: 
A Comparative Case Study

Nancy L. Russo  
Northern Illinois University  
nrusso@niu.edu

Siamak Shams  
Infoglow for Agile Transformation  
shams@infoglow.com

Guy Fitzgerald  
Loughborough University  
g.fitzgerald@lboro.ac.uk

ABSTRACT
In this paper we describe the adoption of agile methods in two organizations. A participant observation method was used to gather in-depth data on the introduction and initial use of agile methods at a bank and a global law firm. An agile usage model, built on theories of innovation diffusion, information systems development, and agile methods, is used to explain post-adoption use of the agile methods in the two organizations. Both organizations had identified problems in meeting business needs with existing waterfall-type methods, and the agile practices and tools introduced at the organizations were similar. Yet the organizations achieved very different results. Compatibility, sociological factors, and top management support were associated with greater agile method use and more positive outcomes.

Keywords
Agile usage, method adoption, success factors.

INTRODUCTION
The traditional approach to information systems (or software) development has frequently been superseded in organizations by what is termed the agile development approach. Agile is an umbrella term for iterative and incremental software development that seeks to deliver working software in relatively rapid timescales. Such approaches include methodologies, such as Extreme Programming (XP), Scrum, Crystal, Dynamic Systems Development Method (DSDM), Lean Development, and Feature-Driven Development (FDD). Where traditional IS development methods focused on documentation, planning, and control of the development process, agile development processes put the focus on the incremental delivery of working, tested software created through collaboration between stakeholders, with short iteration cycles. The adoption of agile is often in response to the perceived failure of the traditional approach, usually in relation to the quality of software developed using the traditional approach, its slow speed of delivery, its difficulty in handling changing circumstances and requirements, and its ineffective engagement of business stakeholders. These issues coupled with today's increasingly fast moving and competitive business environment have encouraged many organizations to turn to agile approaches for their software development needs.

This paper reports on a study of two cases of agile adoption in organizations. These two cases relate to significantly different companies, with different backgrounds, cultures, and approaches and they exhibited significantly different outcomes regarding their agile adoption, including its longer term sustainability. They were chosen for their distinctiveness, with the research objective of ascertaining whether, and what aspects of agile methods, are dependent on organizational, cultural and historical factors of the organization, and whether success and sustainability of the agile approach also exhibits such dependencies. A theoretical model, proposed by Senapathi and Srinivasan (2012), is used to analyze these cases. The model was chosen because of its basis in innovation theory and its encapsulation of social, organizational and cultural factors as well as technical. The study is exploratory rather than confirmatory and intended to to help develop our knowledge of agile adoption in the context of the organization.
BACKGROUND

According to Dyba and Dingsoyr (2009) agile has had "a huge impact on how software is developed worldwide", and as a result is the focus of intense interest from both practitioners and researchers. Early papers tended to focus on agile concepts and adoption issues (e.g. Cockburn and Highsmith, 2001; Boehm, 2002; Highsmith, 2002; Boehm and Turner, 2004; Beck and Andres 2005). Since then there have been a variety of more focused studies, for example on agile in distributed environments (e.g. Ramesh, Cao, Mohan, and Xu, 2006), post-adoption issues (e.g. Cao, Mohan, Peng, and Ramesh, 2009, and Mangalaraj, Mahapatra, and Nerur, 2009), team characteristics and dynamics (e.g. Paivarinta, Sein, and Peltola, 2010, Moe, Dingsoyr and Dyba, 2008; Lee and Xia, 2010), and cultural issues (e.g. Chan and Thong, 2008; Misra, Kumar and Rahman, 2009; Cram, 2012). Whereas the literature identifies many benefits of the agile approach some also recognize a number of drawbacks and limitations, including that it may not be suitable for all application areas and contexts (e.g. Turk, France, and Rumpe, 2002).

Dingsøyr, Nerur, Balijepally, and Moe (2012) identified a dearth of theoretically based contributions and urge agile researchers to embrace a more theory-based approach to their research in the future. Lee and Xia (2010) additionally conclude that there is relatively little empirical research on agility constructs such as the dimensions, determinants, and their effects upon performance, and "as a result, there is a lack of understanding about how organizations can effectively implement an agile development approach". In the light of these calls this research focuses on the theoretical factors underpinning agile and their contribution to performance and adopts the Agile Usage Model, proposed by Senapathi and Srinivasan (2012), as its analytical framework. The model’s underlying concepts are derived from diffusion of innovation theory (Rogers, 2003), the Extreme Programming evaluation framework (Williams, Layman, and Krebs, 2004), and the wider research on information systems methods and implementation. The Agile Usage Model proposes that factors related to the agile method itself (innovation factors), characteristics of the teams that are expected to use the method (sociological factors), the particular agile practices and tools (technological factors), and organizational factors will impact the acceptance and use of agile methods. The amount and intensity of use are expected to affect (and be affected by) observed improvements in the development process in terms of productivity, quality, and customer satisfaction. Senapathi and Srinivasan (2012) acknowledge that the model does not include an exhaustive set of factors that contribute to successful agile method use but nevertheless it attempts to provide a theoretically based approach to the study of use in different agile contexts and thus has been adopted in this paper.

The constructs and components of the Agile Usage Model will be discussed in more detail in the following section together with the research model and approach utilized. The case study organizations will then be described according to the model dimensions, followed by a discussion. Finally, conclusions and reflections on the study and some limitations will be discussed.

RESEARCH MODEL

The Agile Usage Model (Senapathi and Srinivasan, 2012), shown in Figure 1, asserts that certain factors (innovation, sociological, technical, and organizational) contribute to agile method use. Relative advantage and compatibility, from the diffusion of innovation model (Rogers, 2003), are used to represent the innovation factors. Relative advantage refers to the degree to which the adopted agile method is perceived as better than the development method used previously. Compatibility refers to the fit between the agile method and the existing context, including work practices, past experiences, and the expectations of the organization. When an agile method is compatible with the organizational context and development environment and is perceived to have an advantage over the previous method, it is more likely to be used. The sociological factors in the model include experience, technical competence, and attitude regarding the consequences of continuing to use the agile method and regarding innovation in general. Both the particular agile practices in use and the availability of tools to support these practices are relevant to the successful adoption and use of agile methods. The key role of top management support in the successful implementation of information systems development methods is supported by the model, as well as the importance of method champions for adoption success (Conboy, Pikkarainen, and Wang, 2007).

In this model the successful adoption of agile methods is measured in terms of organizational assimilation and demonstrated by the degree of use. The breadth, or horizontal usage, is measured by the number of projects or teams using the method. The depth of usage refers to whether project teams are applying all of the defined practices that make up the method. Agile effectiveness is defined by Senapathi and Srinivasan (2012) as “any specific improvements in the overall systems development process as a result of continued usage of agile practices as perceived by the organization” (p. 1260). Successful organizational outcomes such as improved productivity and quality of software development are expected to be positively correlated with use of the method. In addition, successful outcomes are likely to lead to even higher levels of method use,
either through application of the methods to more projects or adoption of additional agile practices, represented by the bi-directional arrow in the model.

![Diagram of Agile Usage Research Model](Image)

**Figure 1. Agile Usage Research Model (Senapathi & Srinivasan, 2012)**

**RESEARCH METHOD**

A case study-based research approach using participant observation (Jorgensen, 1989) was chosen for this study. Various studies of software development, including agile methods, have adopted this approach (e.g. Moe, et al., 2008; Boden, Nett, and Wulf, 2007; Sharp and Robinson, 2004). McAvoy and Butler (2007) suggest that “To investigate the types of issues found in software development teams requires an approach where a team is examined in context and over an extended period" (p. 557). In this study one of the researchers was a participant in the introduction of the agile method and was able to observe the results of the changes introduced in two organizations. The researcher was a leader, in both cases, of a team in the introduction of agile and not only participated in team activities, meetings, formal and informal discussions, etc. but also worked with people at all levels of the organization, including senior technical and business managers.

Data collection consisted of participant observer logs, interviews with individuals in a variety of roles in the agile adoption process, review of organizational documentation, and data from secondary sources which reported on the agile adoption and use processes. From this, detailed histories of the agile method selection, introduction, and use were created. For the purpose of this analysis, the detailed histories were examined by all three researchers in order to identify the relevant information related to each of the model constructs. Because the Agile Usage Model was not used as an *a priori* framework in conducting the case studies, there were not necessarily identical measures obtained from the two organizations. Based on the data available, however, some information regarding all of the factors in each organization was found.
Our analysis of the two case studies is presented in the following section. For each case study organization a brief introduction is provided before discussing the innovation, sociological, technological, and organizational factors, the use of agile methods, and the outcomes.

**ANALYSIS OF THE CASE STUDY ORGANIZATIONS**

The two case study organizations were selected for this analysis due to their different contexts and experiences with the deployment of agile methods. Each organization’s experiences are described according to the framework provided by the Agile Usage Model that is illustrated in Figure 1.

**Case Study Organization #1: Egg Bank**

Egg Bank, the UK’s first purely on-line bank, was initially established in 1998 as the telephone-banking service of Prudential Insurance. The bank’s long term strategy was to become a market leader by offering a number of unique products and services. In less than two years, the bank had managed to establish itself in the credit card and savings market sectors by reacting to changes in market conditions and being responsive to customers’ requirements. However, the success of the bank in meeting its primary goals was marred by a growing number of failed and late projects, in addition to increased competition from other well established banks which had begun to deliver competitive products and services. A gap between business and IT sectors was identified, and Egg IT practices, particularly lengthy cycles for software delivery, were deemed to be inhibitors to achieving business success. According to Richard Duvall, founder of Egg Bank, “Agility is essential to survival” (Business Management, 2006). The new agile development approach at Egg was first used on a project to facilitate promotions of new services to existing customers. Another early project was to modify the on-line credit card application process to support the use of electronic signatures.

**Innovation Factors: Relative Advantage and Compatibility**

During interviews with project and release managers it was suggested that prior to introduction of agile methods, the delivery of projects was more important than methodical approaches to software development. Many software projects had been mothballed or abandoned, mainly as a result of difficulties with software production and support. Some of the reasons given for this were over-complicated architecture, lack of progress monitoring, lack of reuse, and growing operational distance between the software development and business teams. One manager said: “we have not had one accurate estimate in producing software ....” Some software development and support managers also attributed the difficulty with software provision to the lack of a cohesive approach and method for software development and support. The financial losses that occurred due to problems with delivery and support of the software were reported by a senior manager to run into millions in the 3-year period. Furthermore, unlike most of Egg bank’s competitors who possessed multiple channels of interaction, online software products were Egg’s only interaction channel with its customers.

The shortcomings associated with existing practices were described by Darren Martin, the then director of technology, as: “We had an old ‘waterfall’ style of development here, with requirements and specifications going through to technical design and development in a linear manner. It was a pretty long process, considering that we have a relatively small amount of people serving a significant number of customers” (Watson 2003). Egg bank as a business was entirely reliant on IT, but in terms of their reaction to events, there remained a chasm between the business and IT approaches. The gap, which was behavioral as well as one of proximity, would have to be bridged by bringing the two sides closer to each other. Egg IT was made up of dispersed and specialized teams with strict protocols and ceremonies for communication and cooperation. These barriers would have to be removed to enable all staff to operate as one team.

**Sociological Factors: Attitude, Experience, and Technological Competence**

According to Richard Duvall, founder, Egg bank had been practicing “business agility” since its beginning, “…rapidly creating new products and services to meet the needs of customers, and interacting with customers in ways that suit and adapts to them” (Business Management, 2006).

Egg bank executives decided to align IT with existing business agility endeavors through the introduction of agile software development for all new projects. Egg initially decided to employ .NET technologies on low risk projects due to their inexperience with new technologies and shortage of in-house skills on .NET and agile methods. Egg’s IT staff had unique experience with their ecommerce infrastructure, but limited experience with the agile approach.
Technological Factors: Agile Practices and Tool Support

Agile Egg Framework (AEF) is a hybrid method based on XP, while combining many of the best practices from methods such as Scrum, DSDM and others. At the heart of AEF were three phases: Pre-Project, Development Iteration to Release, and Post-Project Activities. Pre-Project Activities include priority definition; high level solution, architecture, and risk definitions; user stories; business value assessment; and release planning. Development Iteration and Release Activities include the management and implementation of user stories brought forward from the previous phase, design storming, test driven development, spikes, pair programming, re-factoring, stand-up meetings, collective code ownership, and retrospectives. Post-Project Activities include transition to live activities and project retrospectives.

Egg used what were at the time state of the art technologies for internet development such as Microsoft .NET and automated web application testing tools. Egg IT produced a product named Story Studio to capture user stories and activities relating to the stories. Story Studio is also used as a reporting and monitoring tool for product development iterations. Some of the other tools used by these projects include the following: NUnit to create test fixtures and write unit tests, NDoc to create API documentation, CruiseControl.NET for Continuous Integration and build process reporting, FxCop to automate the process of analyzing code for coding standards, Visual Source Safe for source control and open source Wiki for organizing published documents and reports relating to projects. Comparing the legacy development tools and platforms with the newly introduced tools, a frequent comment by Egg developers was that the success of agile methods was heavily dependent on these tools.

Organizational Factors: Top Management Support and Method Champion

Egg Bank’s business units and the highest levels of executives championed the introduction of agile methods to business, and subsequently, to IT functions. The change program cost over £12m and encompassed all aspects of the company at all levels. The change program, which was led by the technology director, ranged from the refurbishment of a new building to the introduction of new software development methodologies. Members of Egg’s board of directors were directly involved in championing the move to agile, even to the point of relocating to be closer to the new building during the transition. The Technology Director, CTO, and CIO championed the move to agile at the organizational level, while every agile team was led by a team leader who received coaching and training. An agile coach was available to teams for 6 months.

Usage: Horizontal and Vertical Use

“The initial project … was deliberately small-scale to test the technology’s capability, according to Ken Woghiren, head of architecture at Egg” (Glick 2003). Use grew from this project, to five projects, and ultimately all new projects were agile. All aspects of AEF were adopted. Therefore it can be said that use was both broad (across projects) and deep (all defined agile practices used.)

Impact: Productivity, Quality, and Customer Satisfaction

In terms of the impact of the proposed change, the director of technology stated that: “If we kept on growing at the speed we've been growing before, we'd have to have a 500-strong technology organization. Instead, we've kept the actual number to around 100 people, which means we can keep them very focused and passionate, as well as keeping our development costs low” (Watson, 2003). The introduction of agile methods had, according to Darren Martin, had a positive impact on development cost, which had dropped by 13%, equivalent to £3 million for the first year.

Carol Moseley, the technical change manager at the bank, gave the example of increased productivity that resulted from the use of agile for the Egg Card site which had been built by three developers in just three months. This task would have taken many more months, had the legacy technologies been used. "It's noticeable that we're able to develop things more quickly," she remarked (Glick, 2003). The pilot team delivered products with zero defect reports, attributed to higher level of attention to User Acceptance Testing and Unit Testing.

Case Study Organization #2: Global Law Firm

Global Law Firm (GLF) came into being as a result of number of mergers and acquisitions which brought together law practices from UK, Germany and other parts of Europe, making GLF one of the largest law firms in the world, providing legal advice to businesses and governments throughout Europe, the Middle East, Asia and the United States. Client data and case information were held predominantly in proprietary databases of commercial off-the-shelf (COTS) applications. COTS packages are were also used for billing and business intelligence purposes. GLF’s approach to software development had traditionally been to customize and extend the acquired COTS products. GLF faced an ever-growing number of COTS applications, complexity of operations, and issues with data integrity. In 2004, GLF embarked on a program of unification of
clients’ and case data storage and the production of an “Integration Platform” application suite to integrate and if necessary replace the multitude of COTS databases with a single data repository and common access and distribution mechanisms.

Two years after the initial sanctioning of this project, the team in charge of the delivery of the application could not demonstrate sufficient progress, causing a change in key personnel. In March 2006, the researcher was employed to help select, introduce and implement agile methods for production of the Integration Platform suite of software. The researcher was tasked to salvage the work of the preceding two years and lay the foundations for future work. The project was also intended to spearhead agile methods for use in other projects.

**Innovation Factors: Relative Advantage and Compatibility**

GLF faced ever escalating costs for the purchase and maintenance of existing COTS packages. Other problems were the proliferation of data sources leading to an accumulation of inaccurate, redundant and contradictory data, and the absence of a consistent approach for the provision of software solutions. Each group followed their own unique approach to software provisioning defined as a variation of the Waterfall method. A typical project life cycle included lengthy requirements gathering. Key components of all applications had to be verified by product sponsors who met once a quarter. Furthermore, the life cycle activities were conducted in isolation with little or no contact between stakeholders. The lack of a systematic approach to software had become a bottleneck for the firm’s IT which prevented timely responses to changing business needs. An increasing number of software project failures to deliver appropriate solutions on time – including the Integration Platform project which had not delivered a solution after two years – were seen as a justification for the introduction of software solutions in an iterative and incremental fashion.

**Sociological Factors: Experience, Attitude, and Technical Competence**

At the time the case study was conducted, all developers and testers at GLF were on short-term contracts. GLF is located in London with access to a large pool of software developers skilled in the latest technologies. Whereas the ability to hire in experience had its benefits, it also led to a lack of continuity, lack of domain knowledge, lack of ownership, risk avoidance by developers and a tendency to follow plans without due consideration to consequences.

While some managers believed that agile methods would provide a systematic approach to software development previously lacking in the organization, others indicated that the system already in place was adequate and change was only required to enable and ensure the more efficient use of available resources.

**Technology Factors: Agile Practices and Tool Support**

The agile method introduced at GLF was based on Scrum. Practices included the provisioning of a constant stream of prioritized backlog of work, prioritization and grouping of backlogs into product releases, project control artifacts, and XP processes for coding, testing and refactoring.

It was concluded that appropriate tools were required to deliver agile practices at GLF. The tools, some of which were already in place, included Version for requirements capturing and prioritization, task specification, and monitoring progress, NUnit for all unit and integration testing tasks, NDoc for document creation, Cruise Control continuous integration builds of the software, FxCop to automate the process of analyzing code for coding standards, Visual Source Safe for source control, and a Wiki for organizing the published project metadata, documents and reports relating to projects and development standards.

**Organizational Factors: Top Management Support and Method Champion**

Software development teams did not have executive level representation within the firm. The lack of influence and sponsorship of IT at the highest level was reflected in the absence of a unified approach and overall vision for IT software development and provisioning. Management’s support for agile was tacit at best. Within the agile team there were self-educated “enthusiasts” who attended conferences and developer forums; however, the team leaders promoting agile had no experience deploying it.

**Agile Usage: Horizontal and Vertical Use**

At GLF there was limited use of agile methods vertically and no horizontal spread of use beyond the first project. The management team decided not to adopt some of the key agile practices such as a 40-hour working week, pair programming, and test-driven development. The agile method was used only for the Integration Platform Project, and was faced with challenges and obstacles from the start. Only three months into the project GLF’s management decided to remove the
business analyst (the customer representative) and tester from the Integration Platform team on the grounds of the excessive cost of maintaining the team as it was. This project was the first and only agile project at GLF.

**Impact: Productivity, Quality, and Customer Satisfaction**

The new agile project, offered a much reduced cycle of software delivery time, each cycle including a portion of the overall analysis, design development, testing and release. Owing to the rapid delivery of the working software, the users and sponsors of the product were able to reduce the parts of the requirements which were deemed irrelevant.

According to interviews with IT management and team members, the successful delivery of software iterations in 4-6 weeks (instead of cycle time of 18 months) on the Integration Project was primarily due to the team composition and its ability to resolve issues internally without calling on external resources.

**DISCUSSION AND CONCLUSIONS**

The analysis of these two cases contributes to our understanding of agile adoption and sustainability in two distinct organizations. Both appeared to have strong incentives for improving their approach to software development, with the limitations of their traditional approaches exposed. The other main similarity was that both organizations had an effective range of tools available to support their use of agile methods, although GLF relatively less so. However, Egg Bank’s focus on business agility, innovation and strong top management support led to higher levels of adoption. These factors are classified as innovation (compatibility), sociological (attitude) and organizational (management support) factors in the model. The high level of horizontal and vertical use of agile practices resulted in positive outcomes in terms of cost of developing software, productivity, quality, and customer satisfaction. Whereas, at GLF, the full use of agile practices was not sustained over the course of the Integration Platform project at least partly due to a lack of senior business management support and understanding, together with a number of organizational cultural factors. Another area of difference was the absence of methodology champions in GLF, whereas their presence in Egg Bank was seen to be significant.

Of course broad generalizations cannot be made, but a second contribution is that the analysis confirms the utility of the Agile Usage Model (Senapathi and Srinivasan, 2012) and the importance of the social and organizational (often called cultural) factors as highly relevant to agile method adoption and use. Further, in reflecting upon the Agile Usage Model, whilst it proved useful and provided insights, there were some factors identified in the case studies that it did not highlight, such as the relationship between usage and effectiveness. Further, it did not specifically draw out the role of decision making practices in the organizations, the management style, including recognition and reward structures, hierarchy and bureaucracy, organization size, and overall enterprise governance. Also, in comparing the factors with others highlighted in the literature, there is an absence of consideration of team autonomy and dynamics, and distributed teams. Thus, although the model has demonstrated some utility, there are a number of additional factors that are likely to be of importance and future research will attempt to enhance the model and the analysis.

**REFERENCES**