

Agile methodologies and the emergence of the agile organization: A software development approach waiting for its time?

Glen Litton Van der Vyver, Andy Koronios and Michael Lane

Division of Information Systems, Faculty of Business, University of Southern Queensland,
Toowoomba, Qld, Australia, 4350

vandervy@usq.edu.au

lanem@usq.edu.au

School of Information and Computer Science, University of South Australia, The University
of South Australia, Mawson Lakes Adelaide, SA 5095, Australia

Andy.Koronios@unisa.edu.au

Abstract

Agile methodologies such as Extreme Programming (XP) have emerged as a major thrust in the computer software development arena. Almost contemporaneously, an organisational theory highly aligned with the values of agile methodologies has mutated from the 'restructuring' paradigm. This paper presents an overview of agile methodologies and the key features of the agile corporation, exploring the synergistic relationship they engender. The paper examines whether agile methodologies are used in practice and what organisational factors facilitate or inhibit their adoption. The paper concludes that agile methodologies are gaining widespread acceptance but there is often a misalignment with organisational culture and values. The importance of web systems have helped agile methods forge a place for themselves but there are still a number of factors mitigating against their large-scale success. In some corporations, agile methods need to be adapted to survive. In many others, their time will only come when changes in structure, culture and values have occurred.

Keywords

Agile corporation, Organisational Culture, Transformation, Organic corporation, Agile methods, Extreme programming.

Introduction

In the business parlance of the new millennium, a term that is becoming rather prominent is *agility*, perhaps the most important dimension of corporate survival in a fiercely competitive, rapidly evolving marketplace. The concept of the rapidly mutating corporation is not new. Indeed, some of these ideas have been around for two decades. People like Drucker (1988), Hammer (1990), Peters (1993) and Hammer (2001), argued that corporations would face a highly competitive global marketplace and would need to adapt in order to survive. Some of the most important adaptive techniques that emerged from this literature include:

- Fewer organisational layers.
- Continuous learning.
- Worker empowerment.
- Team-based work.
- The importance of boundary-spanning roles.
- Devolution of authority.
- Multi-tasking, multi-skilling and shifting work roles.

We will show that the concept of the *agile corporation* incorporates many of these imperatives, sometimes in a rather different form. Zain, Kassim and Mokhtar (2000) define agility as the capability of the company to manage change quickly and easily. They argue that agility is IT driven and revolves around the capability of IT to provide strategic direction to the corporation. Agility will soon no longer be just one of a number of options available to the corporation but one of the few viable ways of competing as *traditional modes of business operations are being unshackled in favour of high speed transmissions and interactive communications across national boundaries . . .* (Zain, Kassim and Mokhtar 2000, p. 70).

It is not the intention of this paper to critique the evolution of the theory of the corporation. What we do consider interesting, however, is the fact that the need for the agile corporation should still be a point of discussion today. Although there have been recent setbacks, few people doubt that the Internet will drive the global economy and that the way we do business will continue to change. This type of business will require corporations to adopt new structures and techniques. We would argue that the emergence of the Internet as the pervasive instrument driving the global commerce of the future has forced corporations and institutions to re-think the extent to which they have truly adapted to paradigm shift (Morris & McManus 2002). For many, the question is not really whether they are ready to adapt to the challenges of the first decade of the new millennium but rather whether they have adapted to the first wave of change – the wave of the final decade of the old millennium.

From the perspective of Information Systems development, there has been much change during the last two decades. New techniques such as Rapid Application Development (RAD), Joint Application Development (JAD) and Object Oriented (OO) development have emerged but, somehow, they often remain on the periphery in many organizations. What has emerged is an often uneasy mixture of older (tried and trusted?) and newer methods. Perhaps one of the key problems in this regard has been that IT has offered corporations a variety of new techniques and tools but these have not been grounded in a unified, comprehensive methodology or structure. The traditional SDLC and its tools, on the other hand, were well grounded in theory and research. Thus, the new tools, such as RAD, were often used in an ad hoc manner and management did not understand their potential to affect organizational change. Of greater significance, management did not know how the new tools and methods should be managed and controlled.

It is fortuitous that the last decade has seen the largely contemporaneous emergence of the theory of the agile corporation and the theory of agile programming methods, for example *Extreme Programming* (Beck, 1999). Thus, a variety of techniques that have been evolving over the last ten years can now be grounded in a larger, structured method, in effect a

methodology for developing Information Systems quickly. At the same time, a theory of organisational structure has been emerging which will apparently have goals and values congruent with those of the new agile programming methods.

Most managers and corporate strategists understand that their companies need to become agile in order to compete in the global world of the 'Net' there have conflicting views as to how Information Technology could deliver the required change (Morris & McManus 2002; Zain, Kassim & Mokhtar 2002). We argue that guidance is to be found at the confluence of theory relating to agile corporations and theory relating to agile programming. Agile methods can deliver the individual tools and the encompassing paradigm to help facilitate corporate transformation into the *Agile Corporation*. The agile corporation can, in turn, deliver the structure necessary for the success of agile methodologies.

This paper will examine the key characteristics of agile corporations and why agile programming methods are ideally suited to these corporations. We will examine how individual IT professionals who are using agile methods perceive the organisational structures within which they are operating. This research is part of a larger study and reports on the first phase of the study. This phase examines the extent to which IT professionals who are using agile methodologies feel constrained and restricted by the organisational structure within which they work.

The Agile Corporation

The concept of the *agile corporation* has been the subject of a relatively small body of serious research and much of that research has focused on manufacturing. This is not surprising because manufacturing organizations were the first to feel the effects of the transition to the post-manufacturing economy (Das 2001; Sanchez & Nagi 2001).

Amos (1996) argues that the term *agility*, when applied to the corporation, first made an appearance around 1991. Of fundamental importance, the term came from the shop floor, not from management theorists. It constituted an attempt to understand and comprehend what was happening on the part of those who saw the effects of massive change first hand. Corporate agility therefore became popular as a method whereby the corporation could remain competitive in the face of unrelenting change. (Goldman, Nagel, Preiss and Dove, 1991).

As is often the case with a broad concept, the term *agility* does not lend itself to easy definition. At the simplest level, it relates to the capacity of the organisation to mutate, to adapt to unpredictable and continuous change. The truly agile company is able to derive competitive advantage at times when whole industries undergo structural shift; it is able to profit from the very presence of change as its competitors flounder, unable to keep pace (Dove, 1994a; Goldman & Nagel, 1993; Das 2001; Sanchez & Nagi 2001).

Dove (1994b) argues that agility is the core requirement for the new organisation. The transformation to agility (from the perspective of a theorist writing in 1994) would be accomplished by means of Business Process Reengineering (BPR) and Total Quality Management (TQM). The final goal was an organisation with the following attributes:

- A lean organisation with few levels.
- A learning organisation.
- Mass customisation – focus on the customer.
- An emerging *virtuality* – i.e. the virtual corporation.

Agility demands that the corporation position itself to thrive in uncertain situations. It does not just happen. Corporate transformation is required and this must be driven strategically, from the top. Of paramount importance, agility is above all dependant on people and information technology. Technology, management and the workforce must function as a interdependent system, always in touch externally and internally (Dove, 1994b; Goldman & Nagel, 1993; Morris & McManus 2002). Agility impacts structure but process is the most important factor. Processes must be minimalist, adaptive and, most important of all, rapid.

Perhaps it would serve us well to revisit the metaphors of mechanistic versus organic organisations. The mechanistic organisation thrives on scale, forward planning, bureaucracy and predictability. The organic organisation thrives on adaptation, uncertainty, customisation and adhocacy. The global, Internet-driven marketplace is not at all suited to the mechanistic organisation, which is slow to adapt. In this marketplace, we do not even know the rules of the game. Forward planning is almost impossible and the organisation must be able to reinvent itself at lightning speed. Agility is the appropriate paradigm. Oliver (2002, p. 7) takes up this cry, proclaiming '*Organic Organisations Rule*'. Oliver takes the radical approach that biotech will replace IT as the dominant technology – and soon. Within this framework, the organic metaphor is of course highly appropriate. In some ways, it reflects a post-organic reality, where the corporation is able to adapt so quickly that it is a driver of change (Raffaelli 2002).

While there is widespread agreement that agility is something more complex than the restructuring paradigm which preceded it, there is by no means unanimity on how to best achieve agility. Oliver (2002, p.8) believes that agility comes from within: *Certain organisations, like living organisms, developed complex self-regulating mechanisms that ensure survival and success*. He argues that the key driver of agility is strategy. Indeed, strategy has been the driver of the three phases of adaptation to the new economy. In the first phase, strategy was used to restructure the organisation. In the second phase, strategy was used to put in place information systems to manage the increasingly complex, global activities of the organisation. The current phase, which Oliver calls *strategy as organism*, is focused on achieving agility. The far-flung branches of the organisation are 'wired' to the centre, always looking for environmental change, always monitoring the environment. When a change or opportunity is identified, the organisation as a whole reacts instinctively, with great speed. The goal of strategy is longevity and the most important strategic processes are daily analysis of events, replication of previously successful strategies, feedback loops and the rapid removal of people, processes, programs or products that stand in the way of corporate strategy – Oliver calls these *mutants*.

Weill, Subramani and Broadbent (2002) believe that strategic agility is best achieved via devoting resources to building IT infrastructure. One of the most important, and most difficult, decisions relates to the extent of IT infrastructure that is required. Too much or too little infrastructure can be equally disastrous. Getting the balance right requires collaboration between business units and IT professionals. Morris and Johnson (2002) believe that the agile organisation is characterised by the centrality of information infrastructure. In order to achieve agility, organisations are turning to e-commerce and virtuality – virtual teams, virtual structures, and even virtual corporations. The strategic value of IT infrastructure is the fundamental platform achieving for agility in the new economy (Sambamurthy, Bharadwaj & Grover 2003). In the agile organisation, E-commerce drives the business, even those functions that conduct more traditional exchanges.

This paper will focus on people and IT, the two key enablers of the organisation transformation to agility. Kiechel (1993) examines the changes that will be required in the workforce:

- The average size of the corporation will decline. Fewer people will be employed by more companies;
- There will be a multiplicity of organisational forms;
- Some of the most successful organisation will comprise a loose network of specialists;
- Horizontal division of labour will become more important than vertical division;
- The core business activity will relate to providing a service and not making a product; and
- The nature of work itself will change to incorporate decision-making and higher order thinking skills, teamwork, constant learning and less emphasis on the traditional eight hour work day.

Shapiro (1995) believes that workers are ‘signing up’ to the new contract. They are reinventing themselves by constantly upgrading their skills and foregoing the protection of job security and union membership. The agile organisation must be built around an agile workforce and the most dynamic organisations have already begun the journey that will transform the workplace (Shafer 1999). However, this comes at a price to the corporation, however. Workers want more – interesting and challenging work. Peters (1993) argues that organisations are well placed to make work more rewarding. Work has become more project-oriented and less task-oriented. This provides workers with regular new challenges, variety and a sense of achievement but it comes at a price – workers must re-invent themselves every few years and passionately support the corporate vision, even though they may not be around for more than a few years, or even a few months. In the new corporation, creativity is crucial, and there exists a ‘curiosity quotient’, whereby companies are built around the capacity to think creatively and innovate (Peters 1993; Bridger 2002). Workers are rewarded for exploring and thinking beyond the ordinary. They are given wider authority and more complex, diffuse performance expectations - and those who best leverage knowledge, creativity and effort achieve the highest rewards.

The notion of the agile workforce has been discussed as central to creating agile organisation which achieves superior environmental responsiveness in contexts of turbulence and change (Bridger 2002). The agile workforce exhibits five capabilities: intelligence, competencies, collaboration, culture, and information systems. Thus, our point of departure in this study is the fit between agile organisations and appropriate methodologies to develop the right information systems that complement and support agility in the organisation.

Agile Methodologies

Agile methodologies have emerged in response to the changing nature of organizations and the way organizations do business. Agile methodologies place an emphasis on adaptation rather than the optimization of processes (Highsmith, 1998). In the networked knowledge based economy, adaptation is significantly more important than optimization. Achieving results in such a business context requires:

- An adaptive mindset which accepts the limits of imposed order.
- Revised development practices to support self organization and flexibility – i.e. the adaptive life cycle.
- Rethinking management especially how we create collaborative environments..

Complex system research directs us to the fundamental issues facing agile organizations (Cockburn, 1998). Is an organisation's ability to adapt quickly a direct function of its ability to collaborate? And even more so, is that ability to adapt a function of an organization's collaborative structure? We would argue that this is indeed the case and that using agile methodologies helps to build a collaborative, decentralised structure, where knowledge is shared and organisational learning is optimised. At the core of agile methodologies there are four key values identified by the Agile Alliance Group (2003):

- Individuals and interactions are valued above processes and tools.
- Working software is valued above comprehensive documentation.
- Customer collaboration is valued above contract negotiation.
- Responding to change is valued above following a plan.

Extreme Programming

There are a number of agile methodologies that are recognized but *extreme programming (XP)* is by far the most widely adopted agile methodology (Fowler, 2000). The XP methodology has 12 core practices (Beck, 1999). Figure 1 illustrates how these twelve core practices of XP fit into the exploration, planning, iterations to release, productionizing, implementation and death phases of the system development life cycle (Abrahamsson, Salo, Ronkainen & Warsta 2002, p. 19). The terms used for the various phases of system life cycle in XP although somewhat different still emphasize the standard analysis, design, build, implementation, maintenance phases in an ongoing cycle until the ultimate death of a system. However, in XP there is an emphasis on priority of customer/client stories which represent the business requirements for each system release. The XP approach involves many small releases of system functionality/features with developers writing the tests for system functionality/features before actually writing the specific code. The involvement of customer from the inception of a project through to the Customer/client acceptance testing before production release of code ensures strong buy in by the Customer/client.

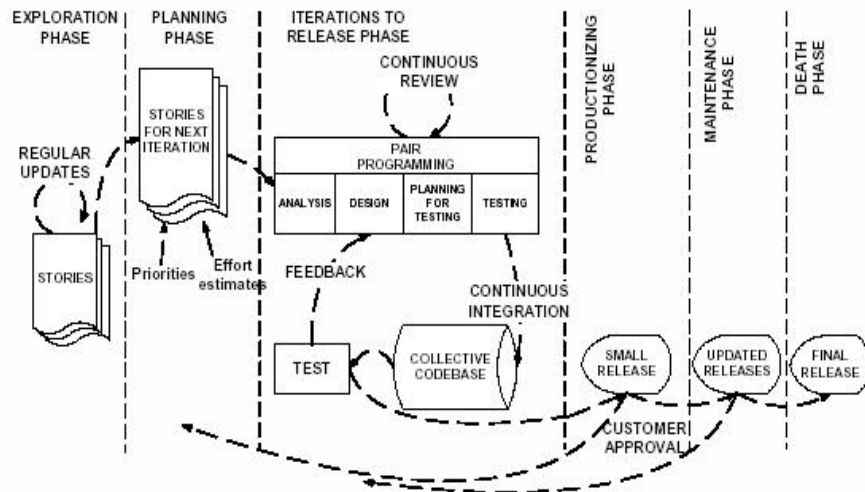


Figure 1: Different activities of XP associated with the phases of the system development life cycle (source: adopted from (Abrahamsson et al., 2002))

There is also a strong emphasis on the team and collaborative development. In this approach, there is knowledge transfer between the customer and developers very much in line with the RAD approach. But XP goes much further and adopts a collaborative approach within the development team by using the ‘whole team’ approach and ‘paired programming’.

This collaborative approach to software development means that the agile methodologies are well suited to the current business environment which continues to change at a dramatically increasing pace (Cockburn & Highsmith, 2001). Agile methodologies such as XP accommodate many of the important adaptive techniques listed in the previous section on agile organizations emphasizing the synergy between the characteristics of an agile organization and agile methodologies such as XP.

Agile methodologies fit with Agile Organizations

The synergy between agile methodologies and the important adaptive techniques of the agile organization would suggest the following outcomes:

Few organizational layers. The emphasis on customer collaboration means that in an XP project all of the stakeholders (sponsor, customer, user, and developer) are on equal footing, part of the same team (Cockburn & Highsmith, 2001). Merging their different experiences and expertise with goodwill allows the combined group to change directions quickly so they can produce more appropriate results and less expensive designs. XP subscribes to the world view that organizations are complex adaptive systems. This is best achieved in a decentralized organizational structure where individuals interact in self organizing ways, guided by a set of simple generative rules, to creative innovative emergent results.

Continuous learning. XP encourages a self adaptive process where there is continuous learning in the team (Fowler, 2000). The continuous learning is achieved through regular reviews of the process at the end of iteration. In the reviews, the team would ask the following questions: What did we do well? What have we learned? What can we do better? What puzzles us? To deepen the process of self adaptability and learning within the organization, teams would conduct a formal review of major project milestones involving all the relevant key stakeholders in the process.

People orientation that is, people are more important in software development than the process or technology. XP requires shared responsibility where the developers have equal footing with management.

Team based focus. All contributors in an XP project sit together as members of one team (Abrahamsson et al., 2002). The team must include a business representative, a client / customer who provides the requirements and sets the priorities and direction of the project. It is important that the customer involved in a XP project is a 'real' end user who knows the system domain and its requirements. The team will comprise of programmers, and system testers who will help the customer define the acceptance tests. Business analysts will be involved assisting the customer in defining requirements. Furthermore there is commonly a coach who helps keep the team on track and facilitates the process. Finally, there maybe a manager in the XP team whose role is providing resources, handling external communication and coordinating activities.

Knowledge Sharing. Everyone in an XP team contributes in any way they can. The best teams have no specialists - only general contributors with special project skills. With pair programming, all production software in XP is built by two programmers, sitting side by side, at the same machine (Abrahamsson et al., 2002). This practice ensures that all production code is reviewed by at least one other programmer, and results in better design, better testing, and better code. It may seem inefficient to have two programmers doing "one programmer's job", but the reverse is true. Research into pair programming has shown that pairing programming while producing less code per programmer more importantly produces better quality code with fewer bugs (Williams et al., 2000). The notion of pair programming extends the sharing and transfer of knowledge in a XP project.

Devolution of Authority. As mentioned earlier, there is shared responsibility between the developers and management in a XP project and developers. The developers have the authority to make all technical decisions.

Multi-tasking, Multi-skilling and Shifting work roles. The concept of pair programming facilitates multitasking, multi-skilling and shifting work roles in the development of software. This practice of using two programmers working in close collaboration ensures that all production code is reviewed by at least one other programmer, and results in better design, better testing, and better code. It is a somewhat controversial practice that has met some resistance from programmers that are used to working independently. It is important to pair the right blend of personalities. It does take practice and time to do well and produce results. Paired programming, in addition to providing better code and testing, also serves to communicate knowledge throughout the team. As pairs switch everyone gets the benefit of the specialized knowledge that exists within a team. Programmers learn from other programmers in the team, individuals' skills improve and they become more valuable to the team. The following comment by Nosek (1998) illustrates the potential positive outcomes of team based paired programming approach of XP.

"To the surprise of managers and participants, all of the teams outperformed the individual programmers, enjoyed the problem solving process more, and had greater confidence in their solutions" (Nosek,1998, p. 109)

There is little doubt that the characteristics of agile methodologies such as XP are well suited to an environment that encapsulates the attributes of an agile organization. We would argue that agile methodologies can facilitate organisational change but, before they can take hold, the organisational structure and values must provide a safe, nurturing environment. Thus, in those organisations that have not embraced agile structures, values and procedures, agile programming methodologies are likely to encounter significant resistance. Moreover, they

will not cause organisational change because the core values of the organisation will be too strongly in opposition. Where agile methodologies are introduced into the wrong type of organisation, they will cause conflict and frustration. Agile methodologies can only work well where the corporation is at least open to change.

Research Questions

This paper reports on research outcomes from the first phase of a larger project to examine practical issues relating to the adoption of agile methods into the corporation. We put forward the argument that agile methods are closely aligned to an agile, adaptive organisation which is highly responsive to changing circumstances. A pilot study was carried out to explore the following research questions:

RQ1: To what extent, are agile methods being used successfully in organisations?

RQ2: What are the key facilitators for the adoption of agile methods in an organisation.

RQ3: What are the inhibitors for adoption of agile methods?

Research Method Employed

A qualitative approach was deemed appropriate to guide the focus of the first phase of the study given that there is little existing empirical research on agile methodologies and the agile corporation (Miles & Huberman, 1994; Yin, 1994). A qualitative approach allowed the researchers to explore the research questions in an interpretative manner inline with our research objectives. It was necessary to capture rich domain knowledge from experienced practitioners, to identify the key issues in relation to the research questions, and develop an appropriate conceptual model that reflects the reality of current practice.

This exploratory first phase of the research provided some insights and will assist in refining the data collection instruments and the development of the interview protocol for a number of case studies that will be examined in the next phase of the research.

A convergent interview was conducted on an online XP discussion forum to solicit the opinions of practitioners. Convergent interviewing is an exploratory technique which was used to identify the key issues in relation to the adoption of agile methods of software development in agile organisations from real life practice. The convergent interview technique allows the researcher to identify and confirm the key issues when there is little previous theory available (Dick 1990). In the next phase of this study we will conduct more structured interviews with respondents that have agreed to be involved in the next phase of the study. The novel approach of conducting the convergent interview technique using an online discussion forum added an extra dimension to the technique. That is, such approach allowed the respondents to interact and comment on key issues raised by other respondents in relation to the general research questions posted on the online discussion forum. Of the practitioners who responded to the convergent interview conducted online, many are acknowledged experts in using agile methodologies such as XP.

Web based information systems were the focus as the unit of analysis for the first phase of this study. It was felt that it was appropriate to focus on systems which are more likely to highlight and draw out the key issues in relation to the fit between agile methodologies and the agile organisation. The responses to the questions were collated and stored in a text file. These were then analysed using the qualitative data analysis software package NVivo 1.3. This software package allowed for the exploration of raw data and identification and coding

of the common themes as well as the identification of relationships between themes in a rigorous manner. While there are some limitations in the approach used, it is felt that the richness of the data collected far outweighed the methodological shortcomings of such an approach.

Findings

In addressing the three research questions posed in the first phase of this study, it was found that organisations are adopting agile methodologies but practitioners report significant problems. The analysis of the data collected from the IT practitioners who responded to our qualitative opinion survey revealed a number of issues that facilitated or inhibited the successful adoption of agile methodologies in web based information systems. The focus on web based information systems is particularly relevant as organisations need to be agile in developing systems that support their business activities in a knowledge based, networked global economy. The main issues that emerged were categorised as facilitators and inhibitors of the adoption of agile methodologies in agile organisations. Key facilitators of the adoption of agile methodologies in organisations are summarised in Table 1 and a discussion of these factors follows.

Table 1 Key organisational facilitators for the adoption of agile methodologies (Source: this research)

Facilitator	Agile methodology technique
Ability to adapt quickly to change	Agile methodologies acknowledge that customer requirements will change
Market pressures demand short time frames for releases	Agile methodologies such as XP are based on small iterations of 1 week or more between releases of working software
Ability to get instant feedback from customer	Short time frames between releases of working software allow developers to gather quick feedback from the customers and users
Organisation processes demand high quality bug free software	Continuous testing and integration regime of agile methodologies such as XP enforces the delivery of high quality bug free software

The capability of agile methodologies such as XP to adapt quickly to the changing requirements of a customer and business environment was considered a very important factor by practitioners. Web based information systems are evolutionary in nature and experience many micro and macro changes. XP accommodates and acknowledges that there will be frequent changes in an information system. The following comments by practitioners emphasise the ability of XP to adapt to change in a business environment.

I worked on a few large web dynamic web sites for about a year -- content management, shopping-cart-style ordering, user registration, mass email based on searches, that kind of thing. Features tended to get added to the live site every 1-3 weeks. We always had some version of each site that we could quickly push to the live site if we needed to (e.g. to fix a bug or rush a feature) (T285:289).44

The capability of XP to facilitate frequent releases of working software underpins the capability of agile methodologies to adapt to change by allowing the developers to release high quality working software in short time frames. The following comment by a practitioner emphasises the value of frequent releases of working software for web based information systems.

I find the frequent releases to be a good fit for web development since it keeps the feedback cycle short and get the product in front of your customer quickly. Once the customer sees it, then you can get the real feedback on how you are doing (T158:163).

Furthermore, frequent releases of working software have the added advantage of keeping the feedback loop between the developer and customer/user short. Frequent releases allow the customer to provide real feedback to the developer almost instantaneously and keep the project on right path. The following comments by practitioners note the value of quick feedback in keeping a project on the right path.

Feedback is important. Because new releases are adopted immediately, feedback comes quickly. This makes steering the project easier (T7

The continuous testing and integration regime of an agile methodology such as XP enforces the delivery of high quality working software whilst still addressing the issue of needing to deliver working software in short time frames.

I have found that XP helps to maintain very high levels of quality through unit tests, simple design and test driven design. The high quality gives you a lot of confidence in your website which is an advantage since a lot of websites are flaky (T174:177).

Another practitioner comment emphasises the overall advantages of XP highlighted in Table 1 (T97:116):

Speaking broadly, I think XP's advantages for fast-paced web projects are even stronger than for other types of project:

- *Having all the running copies of the program under your direct control makes frequent releases much easier.*
- *Because new releases are adopted immediately, feedback comes quickly. This makes steering the project easier.*
- *In fast-paced environments, priorities often change rapidly. Releasing frequently means that when a project suddenly gets shelved, relatively little work sits around unreleased.*
- *Because your competitors are just one bookmark away, that makes the ability to adapt quickly much more important.*
- *Since bugs can be very, very public, XP's ability to drastically reduce bug counts and bug severity is a big win.*

The main inhibitors that emerged from the responses of practitioners related to the ability of many organisations to adapt to agile methodology such as XP. These inhibitors are listed in Table 2 and a discussion of the findings in relation to these inhibitors follows.

Table 2 Key organisational inhibitors for the adoption of agile methodologies. (Source: this research)

Inhibitors	Organisation Characteristics
Traditional Waterfall development mindset	Organisations steeped in waterfall development mindset are reluctant to adopt agile methodologies such as XP
XP requires a disciplined approach	XP enforces a disciplined approach to systems development may not sit well with agile organisations pushing for the next product development release
Unrealistic expectations concerning pace of development	Agile organisations may have unrealistic expectations with which agile methodologies such as XP can meet market demand

One practitioner noted an organisation where the manager was talked into adopting a waterfall approach to a large web project. This resulted in very poor outcomes in terms of the finished product. Interestingly, management was in favour of adopting an agile methodology such as XP but corporate politics dictated a waterfall approach for the project. The following comment by practitioners outlines the problems that emerged in using a traditional waterfall approach for a web project. These comments contrast starkly with comments included previously relating to the suitability of XP for web projects.

Right now, I'm doing some consulting for an organization that had someone else develop their web site (mostly intranet and extranet). They did it waterfall: big contract, big requirements doc, possibly big design doc, big price tag. The project is very late, parts don't function, other parts function very differently than what the organization needs, and still other parts are almost physically painful to use. The customers have literally never seen the people who are actually creating the system. The saddest part is that the customers' manager, while not at all experienced in software development, is a big fan of approaches like XP: incremental progress, iterative steering, and no solo work. The developers had to argue him into the waterfall, big-bang approach.

Another issue that emerged from the responses of the practitioners was that in some organisations there was significant pressure to relax or abandon the disciplined approach of an agile methodology such as XP. This is due to the intense pressure to deliver products and services in the market place given the short windows of opportunity that are now commonplace. XP is a disciplined approach and management needs to be aware that core practices of XP, while flexible, cannot be comprised because of market pressures to deliver in incredibly short time frames.

XP demands doing things right from the beginning. This pays off in the long run, but initial progress on an XP project feels slower than just jumping in and hacking stuff out. On fast-paced projects, this initial feeling of slowness makes XP adoption much harder than in environments where people are accustomed to taking a longer view of things (T128:132). XP requires some discipline. It also requires managers and customers to support that discipline, instead of undermining it with pressure, fits of anxiety or cajoling to "skip testing, just this once". (T310:312).

Interestingly, one practitioner noted that one of his customers wanted to release new features immediately despite the significant risk involved in releasing software that has not been rigorously tested. The practitioner also noted that the client in question was no longer a customer because of his unrealistic expectations in terms of delivery time frames for new features and enhancements.

Conclusions and implications for future work

Agile methodologies are gaining widespread acceptance because of the need for development approaches that can deliver working, high quality software in short time frames to meet the needs of agile organisations. Within the agile organisation, systems created for the web are a crucial determinant of competitive advantage. This research clearly indicates that agile methods such as XP are highly favoured by practitioners involved in developing systems for the web.

Based on the responses of the practitioners involved in this study, the following conclusions were made:

1. There is substantive evidence that agile methods are being used widely, and successfully.
2. The practitioners involved in this study identified four factors that facilitate the adoption of agile methods within organisations: ability to adapt quickly to change, short time frames for releases, instant feedback from customer and high quality, bug-free software. The first three of these are significant in that they would commonly be found in an agile corporation. This type of corporation would thrive on rapid adaptation and corporate clients would in all likelihood be intimately involved in systems development. Feedback from customers would therefore be immediate. Agile corporations would also be competing in environments where systems were mutating continuously to meet the challenges of new competitors, new markets and new opportunities. Agile methods can deliver the systems to meet these demands. An interesting point that emerged from the findings is the indication that high quality, bug-free software is now demanded by clients, regardless of the fact that development teams are working at 'lightning' speed. In iterative approaches such as RAD, it was often accepted that a trade-off for speed was the presence of some errors. These errors were resolved during late iterations of the system construction or even post implementation. This is probably acceptable for systems that will be used by internal users, but the reality of systems developed in the agile corporation is that they are placed on the web, in the sight of customers and competitors. Upon implementation, they can be immediate sources of profit (or loss) and they must work.
3. Agile methods are clearly well aligned with the agile corporation in general, but there are points of potential tension. Perhaps paradoxically, the practitioners involved in this study highlighted the fact that agile methods require a disciplined approach. These are not ad hoc methodologies that can be adapted to every organisational contingency. They are able to deliver good systems relatively quickly but they have a cycle of their own. Because these methods are associated with speed, corporations tend to take them for granted and push the limits. When the speed at which agile methods are able to deliver quality product becomes too slow and corporations push for product releases at very short intervals, the methodology is compromised. Although these problems are a potential source of concern, most practitioners in this survey identify corporations who are committed to the waterfall model as the biggest problem. Many agile practitioners are working in environments that are ambivalent to their approach at best and openly hostile at worst.

Agile methods are clearly impacting on applications development but more research is required to determine whether they are suited to a variety of project types. Practitioners participating in the study emphasized the value of these methods in development for the web, but their suitability to a broad range of information system development needs to be investigated. There were examples of practitioners favouring traditional methods over agile methods, which indicate that further research needs to be done on the penetration of agile methods within the broader IT profession. Subsequent phases of our research will develop and/or refine suitable metrics for measuring the success of agile methods, the organisational fit of these methods and the extent to which these methods add business value over and above more traditional methods.

Agile methodologies have clearly been waiting for the right climate to come into their own. The importance of web systems and the arrival of the agile corporation have helped agile methods forge a place for themselves but there are still a number of factors mitigating against large-scale success of agile methods.

References

2003, *Agile Alliance*, Available: [<http://www.agilealliance.com>].

Abrahamsson, P., Salo, O., Ronkainen, J. & Warsta, J. 2002, *Agile software development methods*, Available: [<http://www.inf.vtt.fi/pdf/publications/2002/P478.pdf>] (12th August 2002).

Amos, J. 1996, 'The Transformation to Agility', *UMI*.

Bridger, D. J. 2002, 'Workforce agility: the new employee strategy for the knowledge economy', *Journal of Information Technology*, vol. 17, no. 1, pp. 21-32.

Beck, K. 1999, *Extreme Programming Explained: Embrace Change*, Addison-Wesley, Reading Mass.

Cockburn, A. 1998, *Non-Linear, first Order Components in Software Development*, Available: [<http://crystalmethodologies.org/articles/panlc/peopleasnonlinearcomponents.html>] (1st February 2003).

Cockburn, A. & Highsmith, J. 2001, 'Agile Software Development: The Business of Innovation', *Computer*, September 2001.

Das, A. 2001, 'Towards theory building in manufacturing flexibility', *International Journal of Production Research*, vol. 39, no. 18, pp. 4153-4177.

Dick, B. 1990, *Convergent Interviewing*, Interchange, Brisbane.

Dove, R. 1994a, 'The meaning of life & the meaning of agile', *Production magazine*, no. November.

Dove, R. 1994b, 'Plumbing the agile organisation', *Production*, vol. 106, no. 12, pp. 14-15.

Drucker, P. E. 1988, 'The coming of the new organisation', *Harvard Business Review*, no. January-February, pp. 45-53.

Fowler, M. 2000, *The New Methodology*, Available: [<http://www.martinfowler.com/articles/newMethodology.html>] (1st January 2003).

Goldman, S. L. & Nagel, R. N. 1993, 'Management technology and agility: the emergence of a new era in manufacturing', *International Journal of Technology Management*, vol. 8, no. 1/2, pp. 18-38.

Goldman, S.L., Preiss, K., Nagel, R.L. and Dove, R. 1991, *21st Century Manufacturing Enterprise Strategy: An Industry-led View*. Lehigh University, Bethlehem PA.

Hammer, M. 1990, 'Re-engineering work: Don't automate, obliterate', *Harvard Business Review*, no. July-August 1990, pp. 104-112.

Highsmith, J. 1998, *Adaptive Software Development: An Evolutionary Approach to Managing Complex Systems*, Dorset House Publishing.

- Kiechel, W. I. 1993, 'How we work in the year 2000', *Fortune*, vol. 127, no. 10, pp. 38-52.
- Miles & Huberman 1994, *Qualitative Data Analysis*, Sage, USA.
- Morris, S. & McManus, D. J. 2002, 'Information Infrastructure Centrality in the Agile Organisation', *Information Systems Management*, Fall 2002, pp. 8-12.
- Nosek, J. T. 1998, 'The Case for Collaborative Programming', *Communications of ACM*, vol. 41, no. 3, pp. 105-108.
- Oliver, R.W. 2002, 'Instinctive Strategy: 'Organic Organizations Rule'', *Journal of Business Strategy*, September/October 2002.
- Peters, T. 1993, *Liberation Management*, Pan, London.
- Raffaelli, C. 2002, 'Tecology: How agile organisations adapt to change', *Employee Benefit New*, April 1 2002.
- Sambamurthy, V. Bharadwaj, A. & Grover, V. 2003, 'Shaping Agility through Digital Options: Reconceptualising the Role of Information Technology in Contemporary Firms', *MIS Quarterly*, vol. 27, no. 2, pp. 237-263.
- Sanchez, L. M. & Nagi, R. 2001, 'A review of agile manufacturing systems', *International Journal of Production Research*, vol. 39, no. 16, pp. 3561-3600.
- Shafer, R. A. 1999, 'Only the agile will survive', *HR Magazine*, no. HR in the 21st Century.
- Shapiro, E. 1995, *Fad-surfing in the boardroom*, Addison-Wesley, Reading, MA.
- Weill, P., Subramani, M. and Broadbent M. 2002, 'Building IT Infrastructure for Strategic Agility', *Sloan Management Review*, Fall 2002
- Williams, L., Kessler, R. R., Cunningham, W. & Jeffries, R. 2000, 'Strengthening the case for Pair Programming', *IEEE Software*, no. July-August 2000, pp. 18-25.
- Yin, R. K. 1994, *Case Study Research - Design and Methods*, Sage, Thousand Oaks, California.
- Zain, M., Kassin, N. M. & Mokhtar, E. 2002, 'Use of Information Technology and Information Systems for Organisational Agility in Malaysian Firms', *Singapore Management Review*, vol. 25, no. 1.