

AGLPM4 – Unit 1 - ACTIVITY 2: OBSERVE

What makes an Agile Leadership Success Story

This document is an excerpt from the book:
“Agile Project Management for Government “
 Authored by Brian Werham
 Published by Maitland & Strong
 Reproduced with permission under license



PMCAMPUS.com / Mokanova Inc

© 2003-2016 Mokanova Inc and its licensors. All rights reserved for all countries.
PMCAMPUS.com is a trademark and service of Mokanova Inc.

PMI, PMBOK, PMP are owned and registered marks of the Project Management
Institute, Inc.

Single use only. Do not download, print, duplicate, and share

Executive Summary

Top officials on both sides of the Atlantic have too often failed to provide agile leadership. The seductive siren call of huge fixed price contracts to deliver technology usually ends up in disaster. In one case a supplier is fired. In another there is simply a resigned acceptance by a government of a flawed solution. Government customers and their suppliers can end up in death-embraces – where neither party can admit that a project is undeliverable. As one newspaper commentator succinctly stated:

“Yet another outsourcing company collects profits when all goes well and the state picks up the pieces if the company fails. Soon much of the state may be too atrophied to step in.”¹

Governments must stop pretending that the business risks of large project failure can be managed by suppliers. Governments must manage these risks. They must stop trying to outsource mission-critical work to be built in large, indigestible deliveries.

There are agile success stories out there: US Veteran Affairs, the FBI, the UK Ministry of Defense, the UK Government Digital Service, Housing Benefits in Australia – all over the world these pockets of excellence demonstrate that governments can be agile. For example, the New Zealand government instituted a disaster compensation system within three days of the Christchurch earthquake. The team responsible for the software used an agile approach to visually track their work on a continual basis. Releases of working software were scheduled on a half daily and sometimes even hourly basis. The system paid out more than NZ\$200m, and ensured economic continuity in the face of a natural disaster.²

I present cases of projects that governments around the world have implemented successfully using agile approaches, such as a safety critical defense project, a benefits payment project, and a federal criminal and homeland security project among many others.

I am proposing that the spread of agile thinking in governments will be accelerated by the adoption of 9 specific Agile Leadership Behaviors that I identify in this book. These 9 Agile Leadership Behaviors are a necessary foundation that will pave the way for agile success. If the concept behind a project is bad, then the approach should change. Cancellation of the project early with little harm done may be the best decision. The ability to change direction when facts are uncovered that upset prior ideas is a fundamental characteristic of an agile approach.

Also identified here are 6 Barriers to Agile Success. Agile thinking addresses the current *addiction to process* and mega-project mania to reduce risk and deliver on time. By thinking differently about how they agree their objectives up-front on projects, go about procurement, and carry out project audits, governments around the world will overcome these 6 Barriers to Agile Success.

This is the first large-scale research that has been published on agile project management for

government, and I have been helped enormously by Chief Information Officers in governments around the world. They have seen how agile can deliver, and they want the call for change to be loud and clear.

There are many sources for best practice guidance on the agile approach. I have chosen to describe three in this book because they have different perspectives and strengths: the Dynamic Systems Development Method framework from the not-for-profit DSDM Consortium; the Scrum method described in the writings of Schwaber and Sutherland; and the eXtreme Programming (XP) techniques developed by Kent Beck.

I give examples of how these have been combined to get the rounded approach to agile that government needs. By reading this book you will be exposed to just enough jargon to enable you to sit down and talk to agile experts and ensure that your team processes will work under your leadership.

But although best practice guidance can help project processes, it is no substitute for leadership!

Introduction

The agile approach is best summed up as being a way of incrementally delivering change so as to get the earliest possible benefit, get feedback early on what works, and change direction accordingly. I argue in this book that governments around the world have for many years been doing the exact opposite with their technology developments. They have commissioned large projects that progress along a predetermined and unfaltering course, deliver late (if at all) and provide little or no benefit.

I have decided to lay out these arguments in the first part of this book using the Harvard MBA case study approach to compare and contrast the agile and non-agile approach. I avoided the classic book structure of 'history, theory, examples' because the first question people have been asking me when I told them about this book was "Can agile be used in governments?" Therefore I have turned that classic sequence of explanation on its head.

I start with fully attributed examples of government success stories. These are from around the world, including the USA (where the Federal Government is in the vanguard of demonstrating success with use of agile on some huge projects) and also the UK and Australia. These are real stories and are fully referenced. The case studies in this book actually happened and are fully attributed.

The central tenet of the agile approach is that we must be scientists. When we start a project we have a hypothesis that the outcome will be beneficial. We must test that hypothesis as the project progresses. Regular delivery of testable product provides the basis for ensuring that our projects are on the right track.

Much is made of the word "agile" in government today. The US Government says: "Government IT needs to be more agile, more responsive and more accountable to the citizens."³ In the UK the government has vowed "to be more agile, more fleet of foot".⁴ So, is agility anything more than a nebulous concept? If a government wants to be more agile what must it do?

Tom Gilb was one of the first to propose an incremental, agile approach to developing software. Rather than have large, clumsy, slow and ultimately risky projects that took years to complete, he proposed an *evolutionary approach* he termed "Evo":

"Evo is a technique for producing the appearance of stability. A complex system will be most successful if it is implemented in small steps and if each step has a clear measure of successful achievement as well as a 'retreat' possibility to a previous successful step upon failure. You have the opportunity of receiving some feedback from the real-world before throwing in all resources intended for a system, and you can correct possible design errors."⁵

In a 1985 paper, "Evolutionary Delivery versus the 'Waterfall model'", Gilb introduced the EVO method as an alternative to the waterfall which he considered "unrealistic and dangerous to the primary objectives of any software project". Gilb based EVO on three simple principles:

- ◆ Deliver something to the real end-user
- ◆ Measure the added-value to the user in all critical dimensions
- ◆ Adjust both design and objectives based on observed realities.⁶

Figure 1 shows a simple conceptual model that helps put this book into the context of government strategy. The outside bubble represents the desire by politicians and top management to be both *lean* and *agile*.

Lean Government has all unnecessary and wasteful ‘fat’ trimmed off. This is a process that not only boosts efficiency but also increases quality of output. Lean initiatives are generally internally initiated and maintained.

Agile Government is able to change direction quickly when faced with unforeseen or unforeseeable circumstances. This reduces risks of failure. Just as an athlete may fall attempting to jump over a hurdle that is set too high, in an agile world we set the hurdles at a comfortable height and at regular intervals. *Agility*, then, corresponds to setting short, realistic targets and reacting fast to changing circumstances.

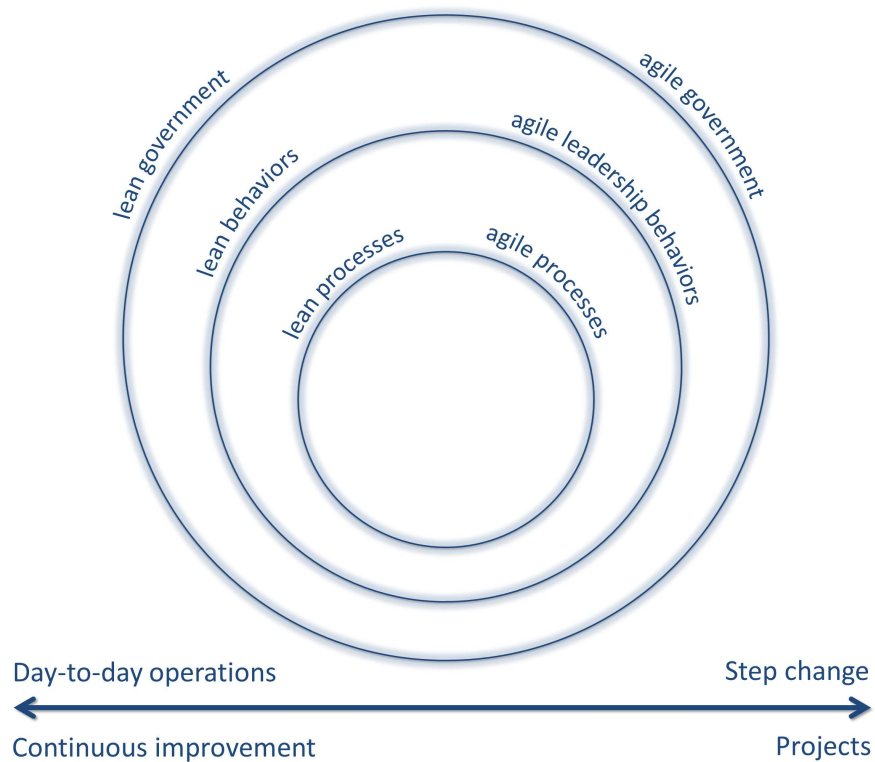


Figure 1: What does *agile* mean?

Both the US and the UK have plans to increase agility. The White House 25 Point Plan aims to increase quality and efficiency in US Government Information Technology (IT).⁷ The UK Cabinet Office IT Strategy has similar objectives, with five out of the fourteen points specifically relating to the adoption of agile approaches in development.⁸

When Vivak Kundra was sworn in as Chief Information Officer (CIO) for the US Government in 2009 he inherited \$27bn (and that is billion not million!) in IT projects that were behind schedule and over budget. His \$1bn cancellation of the Military Human Resources System was just one of several actions he took to try to take control of a spiraling, out of control IT project budget. From 2001 to 2009, IT spending nearly doubled, growing at an annual rate of 7%. But from 2010 onwards Kundra capped the IT Budget. Spend was forecast to rise to \$104bn by 2013, but the new forecast was just \$79bn – a saving of \$25bn per year.⁹

His 25-Point Plan called for a “modular approach to development using an iterative development process”. It intensified previous attempts to move to an agile approach.¹⁰

Agile behaviors reduce the reliance on premature agreement of detail before development work commences. The proponents of this approach (often called *Agilists*) argue that as development gets underway new requirements appear that were not considered previously. Conversely the development teams discover problems (and opportunities) that can inform strategic decisions. They say that one should not imagine that a detailed specification for a system can be written years in advance of the development taking place and being implemented for use.

On the face of it, adopting an agile approach appears to be at odds with typical government bureaucratic approaches. I argue that although the turnaround to a new way of thinking will be a challenge, there is already evidence of success.

This book is about the adoption of agile in government and how to overcome the barriers to its introduction. The application of this book is relevant at local levels, not just central government. Some geographically local projects are of a staggering size. The Mayor of London, for example, spent £161.7m in setting up a congestion charge plan for the city.¹¹ Public bodies have planned significant technology projects. The US National Digital Information Infrastructure and Preservation Program were allocated \$100m in funding from Congress in 2000.

The trans-Atlantic interaction between technology developers and project managers in the US and the UK is a central theme of the book. There has been a continual and fruitful interaction between the Governments of the US and the UK in the development of computers. The US Navy played a pivotal role in the British development of the first large-scale vacuum tube driven computer at Bletchley Park in England, which broke encoded Nazi war messages.¹²

Other authors have argued that agile processes can be scaled up to large projects.¹³ But I propose here that a bottom-up push by agilists will take time and will run into organizational inhibitors. What is needed is leadership, especially at the strategic level. Although many agile concepts are complementary to existing approaches, and there has been more continuity in the development of project management approaches than many recognize, a change in leadership thinking is needed. It is the emphasis and strategic philosophy of management that needs to evolve to encourage agile and allow it to thrive in a government environment.

In delivering large projects in both public bodies and large corporations, I have had to work hard to make large, inflexible procurements more incremental and customer focused. Leadership of others, such as lawyers and procurement executives, played a crucial role in steering these projects to success.

My experiences in leading large teams in the USA (in North Carolina and New York) and in Canada mirrored those in the UK and Europe. I led projects that would now be termed agile that delivered the first automatic code generators for Windows-based computers. These projects revolutionized user-friendliness, decreased training requirements, and reduced error rates by replacing mainframe terminals at large public and private sector organizations. The key was flexibility in setting the team goals, and agreeing what the business was going to realize by delivering working solutions incrementally from an early stage.

Any practical project manager will know that it is better to deliver an imperfect solution early than wait forever for perfection. The banker J.P. Morgan is reputed to have said: “I want it Thursday, not perfect!” The trick is to know what level of imperfection can be handled by the business and traded off against the early realization of the benefits of the solution that the project is going to deliver. Bill Gates knew this when deciding on the right moment to release the replacement for Windows 3.1. He called it Windows 95. It was officially named for the year of its release (1995), but my sources at the time told me that it was named after an internal slogan: “95% ready, not 100% perfect”.

Part I, then, does not start with theory – it contains proof of success. It tells stories of effective use of agile in the face of huge challenges. These stories are provided to give you the incentive to say “Yes – we can also be agile! Tell me how I can lead my colleagues so that they can also have agile successes!” In these cases, I see how projects around the world have used popular agile best practice guidance to achieve agile success. As previously stated, I focus on three sets of best practice: the DSDM framework, the Scrum method and eXtreme Programming techniques.¹⁴ I have chosen to examine these three in this book because they have different perspectives and strengths, and, as we shall see later, they have been used together to great effect:

- ◆ DSDM provides an agile framework that can be applied to any type of project. It can be used to run IT or non-technology projects such as incremental construction or engineering. The DSDM framework provides practical guidance on agile governance processes, operational implementation, and project management together with team-level structures and techniques.
- ◆ Scrum is a method which provides guidance on technology development via a set of processes and practices at the team level. The Scrum method takes an unpretentious, empirical approach to the development of products which is easy to follow.
- ◆ eXtreme Programming (XP) techniques help IT developers work together, become more productive, and create high quality computer software.

There is a growing body of opinion that these three can be used to contribute to success on large government projects. Craddock, Richards, Tudor, Roberts, and Godwin (2012) have proposed a promising approach for using the DSDM framework with the Scrum method:

“One or more aspects of the DSDM Agile Project Framework may be used to supplement Scrum ... where they make the use of the Scrum (method) easier (or) more effective.”

As we shall see, where management has in mind a time and budget limited project to deliver change into operations, the DSDM framework may be successfully used as a wrapper around the Scrum method to create a hybrid of the best of both sets of guidance. This ensures that all those who may be impacted by the new system (the *stakeholders*) are engaged with appropriately. In the same way,

when using the DSDM framework and the Scrum method together on a project involving IT development, it can be useful to include some XP techniques because the Scrum method does not give guidance on specific software development techniques.

Many organizations embarking on agile projects for the first time feel that they have to make an exclusive choice, and adopt one set of best practice guidance only. This can lead to a very limited set of processes and some blind spots. I suggest here that you should consider using the best of all three of these sets of best practice, and incorporate good agile thinking from elsewhere whenever possible.

Keeping an open mind to new ideas and fresh evidence is a great agile leadership quality.

Part II then proceeds to give guidance on the leadership dimension. It explains the genesis of the Agile Manifesto and the related 12 Agile Manifesto Principles which define what agile is, and what it is not. I make the argument that it is the leadership perspective, not the process perspective that is most critical, and I propose 9 Agile Leadership Behaviors that you should follow.

Your adoption of these behaviors will reduce risk and encourage the use of the agile approach in your organization. Each of the nine chapters that follow examines one of these leadership behaviors in the context of government regulations, rules, unhelpful and inconsistent 'best practice' guidance, and organizational inertia.

More evidence of agile project successes around the world is provided and contrasted with the problems of the traditional *waterfall* approach on past government projects. The waterfall approach to project management requires each step of a project to be completely finished before proceeding to the next. For example, design then development then testing before use can start.

I put forward two main arguments. First, that agile project management provides a much better way of running most technology projects than waterfall approaches. Second, that without agile leadership, governments cannot become agile. Some practical advice is given in these chapters on specific improvements to make in the use of the DSDM framework and the Scrum method in your organization. At the end of each chapter I provide a list of agile leadership exercises you can do right now – even if you are working in a waterfall environment!

Part III of this book identifies 6 Barriers to Agile Success. These are the potential blockers to the adoption and spread of the agile project management approach. The 1990s was dominated by a desire to use large complicated design methods. The last decade was dominated by huge prime supplier outsourced contracts which crushed any incipient agility in many government offices. And procurement, regulations, and outdated approaches to project audit still remain the main inhibitors of agile adoption in government.

One interesting piece of news I discovered while carrying out the research for this book is that agile is being introduced into high schools in New Zealand. Final year students are required to understand and practically use an iterative development life cycle and the concepts of *test driven development* of technology. Students will be required to demonstrate agile teamwork.

The new syllabus is very broad – ranging from how mp3 players work to e-commerce and the impact of technology on society. Although traditional technical skills are being taught, the stress is on getting a “taste of the discipline to find out if it suits them or not.” The syllabus is now implemented, and researchers are tracking the students through the system to assess the results.¹⁵

The output of the first agile graduates into work and higher education is expected shortly not just in New Zealand, but from schools around the world. Governments need to be ready to make use of their knowledge, energy and enthusiasm...

Part I

Stories of Agile Success in Government

When I discuss the concept of the agile approach with leaders in governments in different countries, I get a lot of interest. These people understand the scale of culture change that is needed if the agile approach is to spread throughout government, and they want to know how to convince their colleagues. That is why I wrote this book.

The best way for me to convince you to adopt the 9 Agile Leadership Behaviors in the middle part of this book is to start with some real-life agile success stories. Whether you are a trainee or a senior director or politician, these case studies will provide you with the evidence that you need to lead your teams to agile success.

These events actually happened. I present a *warts and all* account of each one. Each is fully attributed – no anonymous case-studies appear in this book. I haven't selected small, experimental projects. These are all large, hairy beasts. As we progress through each case, I will introduce agile concepts and some Jargon.

¹ {After G4S, who still thinks #417}

² {Office of the Auditor-General 2012 #428}

³ {Jeff Zients 2010 #240}

⁴ {London Evening Standard 3 June 2011 #168}

⁵ {Gilb 1976 #413}

⁶ {Gilb 1985 #427}

⁷ {Kundra 2010 #157}

⁸ {UK Cabinet Office 2011 #167} and {UK Cabinet Office 2011 #375}

⁹ {van Roekel 2012 #364}

¹⁰ {Kundra 2010 #157}

¹¹ {London 09/02/2012 #169}

¹² {Hamer 1998 #215}

¹³ Many books spend most of their time discussing the detail of the various 'flavors' of agile before discussing scaling them up. See for example a bottom-up discussion in {Leffingwell 2007 #170}. Other books, such as {Larman 2009 #172} and {Larman 2010 #171} spend many pages on detailed discussions of scaling to large teams without taking into account the requirements of large organizations.

¹⁴ DSDM and Scrum: {Craddock 2012 #330}, and {Glass 2001 #136}

¹⁵ {Dinning 2009 #261} and {Bell 2010 #260}