

Why Agile?

By Alex Gray, Lean & Agile Trainer/Coach

Whether you are just starting to find out about Agile, or already considering it as a potential new way of working, 'Why Agile?' is a very sensible question. The answer we will provide in this white paper consists of looking at available ways of working, highlighting their strengths and weaknesses, and listing out the benefits that an Agile way of working can bring to your business.

Our traditional way of working (synonyms: **waterfall**, the defined process) has some inherent problems. It cannot cope with change or uncertainty, and it limits innovation.

Let's think about how the traditional process works.



We start off by **defining all the requirements**. We give our customers one single opportunity to define all their requirements. Their reaction will be to spend as long as they can on defining all the possible things they could ever need.

Great. So we have defined everything that the customer could possibly imagine they might require. The output of this is a **Detailed Requirements Specification**, a document, which sometimes is hundreds of pages long.

We get the customer to sign off that this is everything they want, and they can never change this, without being inflicted by the dreaded 'Change Control'.

This tome of a Detailed Requirements Specification is then passed to the **Analysis team**.

The analysis team doesn't always have much context about the requirements. Instead, the information it has to hand is the DRS. So the team spends a long time analysing the DRS, and producing some large and lengthy **Analysis Document**. If the customer is really lucky, they will be forced to sign this off. They might not understand it, but it is important that we lock them down and threaten them with 'Change Control' again.

The same process might happen again between the Analysis and the **Design**, yet another document will have been created.

We all well know that documentation is not a very good communication method. It seems though that it is our preferred way to pass information around.

At the next stage, the document gets passed over to the **Developers**. By now, the customer's requirements might be third or fourth hand and have made a number of deviations in these documents, being several months in the making.

The developers will do their best, with the information given to them, and pass it on to the **Test team**. The test team will use the same documents the developers had as input, but they might read them in a different way, and run tests with their understanding.

Developers will have one understanding of the requirements, and Testers another. Some toing and froing and, eventually (often very late), we will have something to show to the customer.

Developers will have one understanding of the requirements, and Testers another.

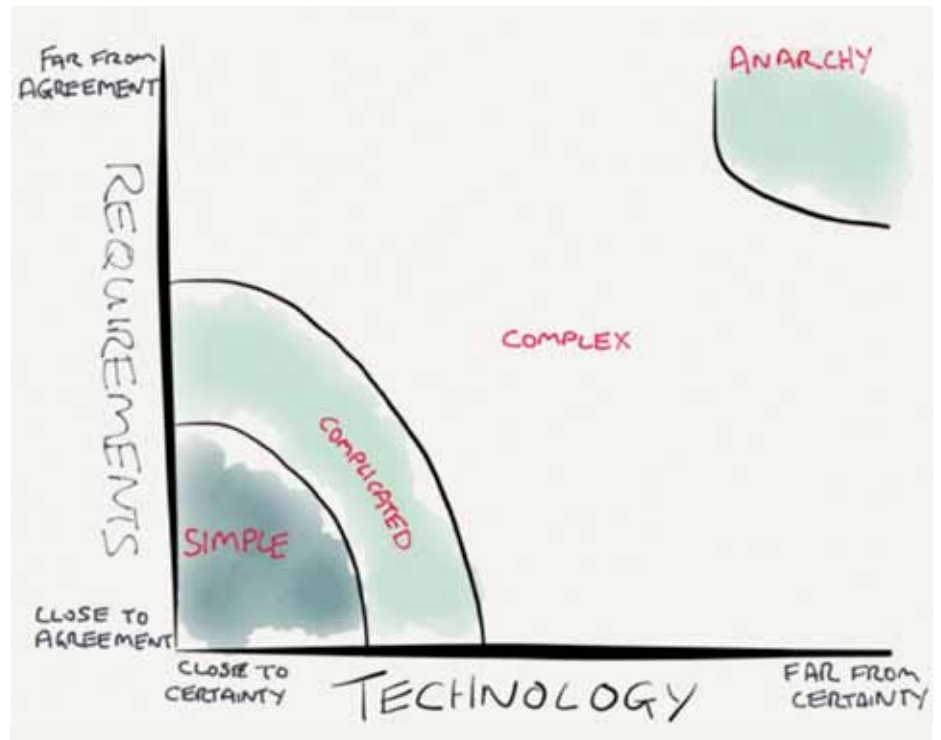
What we show to the customer is not fully tested. Testing is traditionally the last thing we do, and, by virtue of being the last thing we do, it is often the thing that is squeezed for time the most. Can you just work the weekend to get testing done? Can you run tests at night, can you just test 'the happy path'?

We may now be several months into a project, and it is time to show the product to the customer. The covers are removed and this may be the first time the customers has seen a working product, their response might as well be one of the following: 'that is not what I wanted', 'that is not quite how I expected it to work', 'that is not what we meant', etc.

All the communication through documents leads to Chinese whispers, and ultimately leads to a product the customer didn't want. And do you know something?

Oftentimes that customer didn't really know what they wanted in the first place. It was really hard for them to imagine the product and describe it in a document, especially when they don't know exactly what they want.

Stacey Model



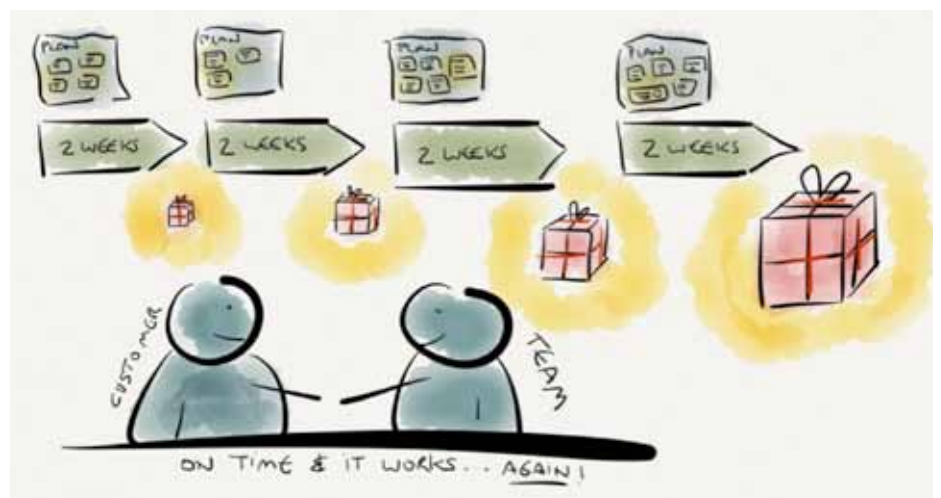
There are some cases where using a traditional method is the best way to work. If we know exactly what the requirements are, and exactly what the technology is, or how we will build it, then the traditional process will work very well. Agile will work here too, but is not absolutely necessary.

Agile processes accommodate change. So as soon there is some uncertainty in the requirements, or some uncertainty as to how we will deliver the product, which - let's face it - is most of the time, then this is the sweet spot for Agile!

By using Agile, we try to change and improve a number of aspects, ranging from recognising that we can't define all requirements up front, through to involving the customer on a continuous basis, and up to delivery of the products and services.

When employing Agile, this is what we want to achieve:

Agile Model



- We don't pretend that we can define all the requirements up front, or know exactly how we will deliver something. We learn as we are going through iterative and incremental development.
- We engage our customer throughout the process, in order to improve communication. This can be a big mind-set change for customers, and they need to be prepared to spend less time upfront doing requirements specification, but more time throughout the delivery phase, working with the delivery team.
- We deliver things on time and matched to the agreed level of quality. Testing is not left until the end: we test continuously through the development.
- We try and deliver things as soon as possible. We want to learn whether we are developing the right product for our customers as soon as possible. Or we want to learn if the technical solution will work as soon as possible. These key pieces of feedback are highly valuable and used to drive the future direction of the product.
- We want to have the freedom to innovate new solutions, and new products.
- We want to have freedom to identify how, as a team, we can improve our development people and process.

Being Agile is about delivering the right product to our customers so they can achieve their desired outcome as soon as possible.

This is **why** we do Agile.

If you too want to collect the benefits that Agile can bring to your organisation, cutting out lengthy delivery times without eventually offering customers what they want, then consider adopting Agile today.

About the Author

Alex is a professional, versatile and enthusiastic Lean & Agile Trainer / Coach with 20 years' experience in varied IT projects and roles. A member of the BCS Agile Expert Panel, and an author of the BCS Foundation in Agile Practices syllabus, examination and course materials.



About Radtac

Radtac unlocks your organisation's potential to accelerate value delivery for less cost. The only Agile specialist company offering the full combination of training, consulting, culture transformation and staffing solutions.



Contact Radtac:

+44 (0)207 953 9853
enquiries@radtac.co.uk

www.radtac.co.uk

[Radtac Agile training courses](#)

Content and images courtesy of Radtac.

Professional certifications designed to help individuals and their organizations perform more effectively.

Project & Programme Management

IT Governance, Service Management & Cyber Security

Business Management & Change

www.apmg-international.com

www.apmg-cyber.com



Discover our complete portfolio online:

www.apmg-international.com / www.apmg-cyber.com

A global network of accredited training and consulting providers at:

www.apmg-international.com/AccreditedProviders



@APMG_Inter



@Cyber_APMG



company/apmg-international