

How to get what you need in an Agile environment: the Basics

(The first in a series of articles on this topic)

Author: Rezah Reid, Business Analyst @ Futuregrowth

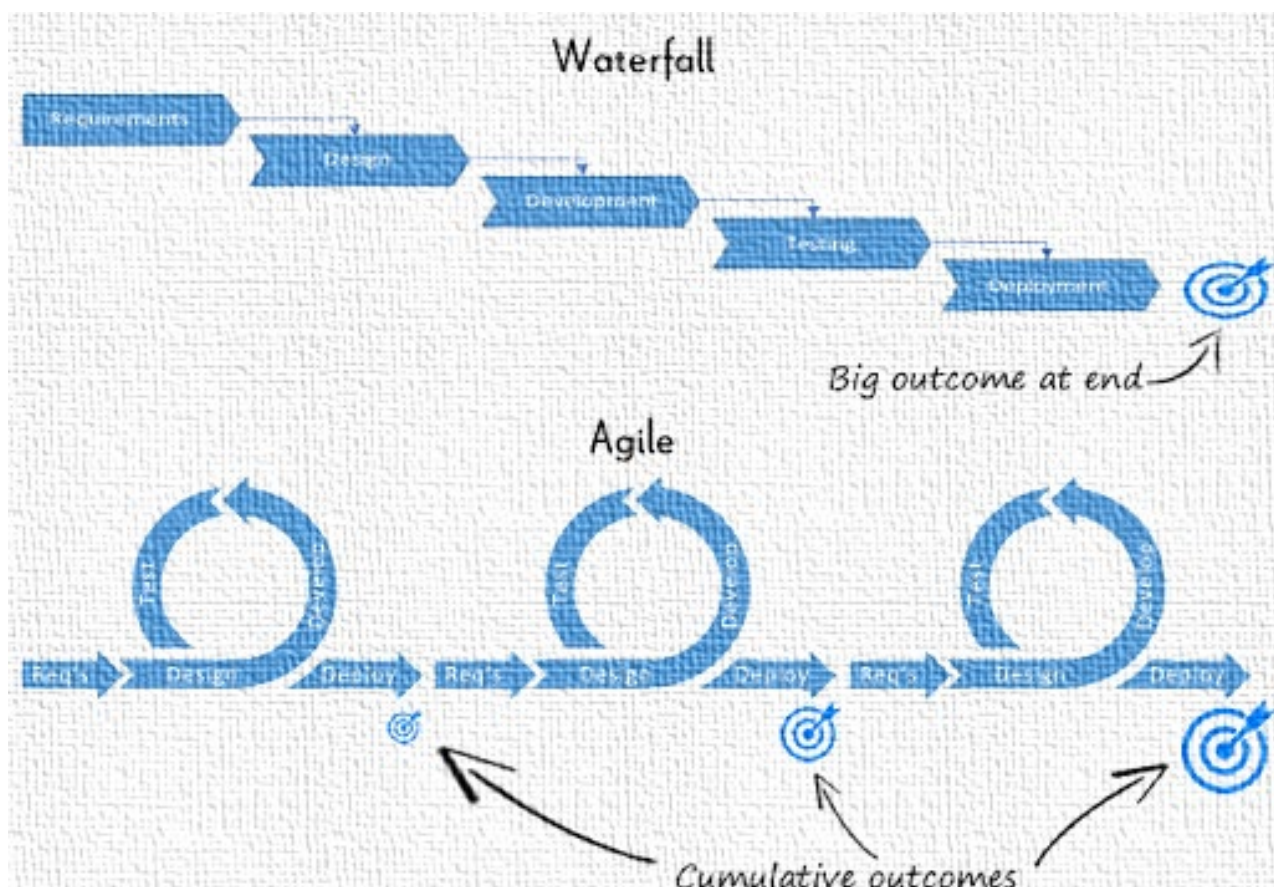
Published: June 2020

What is Agile?

Agile business and Agile development processes have been getting a lot of attention, perhaps nowhere more so than in the Enterprise Solutions space where we are constantly working on co-creating an agile process that puts the focus on business value and client engagement.

The distinction is generally made between "agile" and "waterfall" approaches to software development. "Waterfall" refers to the traditional engineering format of having big planning and analysis stages up front and then building a solution in isolation, in a series of cascading steps. This is sometimes called "big bang" development since it takes so long for an outcome that it might be dated or completely off target by the time it is delivered. "Agile" methods, on the other hand, build towards business goals in small increments, getting feedback and pivoting accordingly after each stage. This makes them more customer centric, with the goal of constantly delivering and assessing value along the way.

This diagramme illustrates the different approaches:



Source: <http://crmsearch.eu/agile-versus-waterfall-crm.php>

I am what you might term born agile, since my entire career has been within Agile teams (I'd rather not say how many years but let's just say my Agile career can legally drink and drive a car - hopefully not at the same time though!). I have worked on the deployment of Rational Unified Process (RUP) to Extreme Programming (XP) to SCRUM to Kanban, with a seam of Lean Thinking running through it all, where the

NEW WORDS?

Rational Unified Process (RUP): is an iterative software development process framework intended to be tailored by the software project teams that will select the elements of the process that are appropriate for their needs. (<https://en.wikipedia.org/>)

Extreme Programming (XP): is the most specific of the agile frameworks regarding appropriate engineering practices for software development.

(<https://www.agilealliance.org/>)

SCRUM: is structured in a way that allows teams to incorporate practices from other frameworks where they make sense for the team's context.

(<https://www.agilealliance.org/>)

Kanban: is a workflow management method designed to help you visualize your work, maximize efficiency and be agile.

(<https://kanbanize.com/>)

Lean Thinking: is a business methodology that aims to provide a new way to think about how to organize human activities to deliver more benefits to society and value to individuals while eliminating waste.

(<https://en.wikipedia.org/>)

method may have flexed but the intent has always been the same - satisfying client needs through prioritisation of the highest value work items.

Executives opting for Agile methods are often seduced by the promise of shippable products within two-week sprints, better tracking and capacity management and more engaged teams. A good way to view the Agile process is as an upside down funnel, filtering business needs (from projects to bugs) into a state that the development teams can work with in a logical, prioritised and sustainable manner. This involves a layer of planning, negotiation and analysis that can sometimes be easy to overlook. Teams within businesses often have competing concerns, and development teams find themselves fielding requests from across the organisation, with requests often landing en masse and concurrently. This is where an understanding of the process becomes useful for people outside of the team.

With this in mind, I would like to offer a view of how things are prioritised, the importance of focus in order to bank completed work and the best ways of interacting with development teams in order to achieve your objectives.

What are the benefits?

The key benefits of bringing about business change through software development, project management or change management in an agile way are rooted in the Agile values, which include:

- transparency;
- client engagement;
- continuous improvement;
- focus on quality; and
- business value (or solving the right problem rather than of solving the problem right).

My exposure to these methodologies prompts me to think of the benefits in terms of orchestration. The internet is awash with memes around what happens when a team does not act in concert and members fail to perform their specialist roles in a manner that allows others to succeed. Software development and change management involve a complex choreography of both individuals and the team - that should result in improved business processes and stakeholder productivity. (Just google "gymnastic fails" to see the disastrous consequences of being slightly out of step with a carefully choreographed dance).

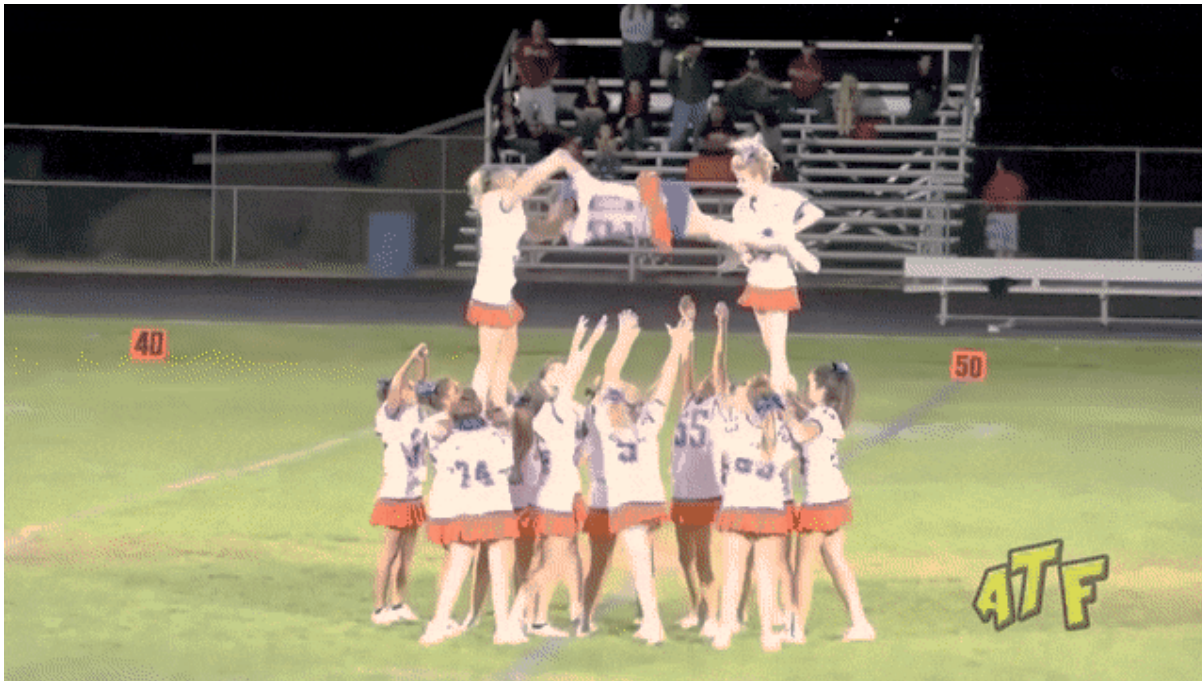
The key to smooth orchestration is to realise that we are all on the same side, be that part of the business side or the development side. We are supporting each other towards an outcome of improved efficiencies and greater profitability – and that this is a team sport and not an individual contest. For us to succeed as a business, we all need to hit our marks and ensure that the handoffs are clean, and that they effectively set up the next steps towards realising the desired outcome for the business.

NEW WORDS?

Sprint: In Agile product development, a sprint is a set period of time during which specific work has to be completed and made ready for review – usually a two-week period in our context.

(<https://searchsoftwarequality.techtarget.com/definition/Scrum-sprint>)

Agility for its own sake is pointless, but agility harnessed to produce a seamless chain of events, in which the outcome is greater than the sum of the parts, is a valuable objective. To this end, it is important to discuss the various elements of the choreography of these projects and how we ascertain where to focus our collective effort.



Source: <https://gfycat.com/definiteidolizedarcherfish-videos-fail-football>

To begin with, where do we set our sights?

There are three horizons in the planning stage, as discussed in the International Institute of Business of Business Analysis (IIBA)'s Agile Extension:

1. *Strategic:* At this level, projects get prioritised across the business using a matrix of factors to determine whether the project will yield the highest possible benefit for the cost of the resources deployed in achieving it.
2. *Initiative:* Here, a roadmap is devised that develops a picture of the steps and phases required to realise the vision that has resulted in this project being prioritised. This is the jumping off point for deeper analysis.
3. *Delivery:* At this horizon, the planning is focused on enacting a two-week cycle of development that will result in a realised business benefit from the deployment of production software.

Orchestration within the Futuregrowth Ecosystem

At Futuregrowth, we take a portfolio approach towards project prioritisation, managed through our Steerco platform. In order to get your development requirements prioritised, it helps to know the lay of the land. Work requests get discussed with a business analyst who can assist you to log the request in the correct format and to the correct level of detail. This ensures that rework and repeated conversations are kept to a minimum. Depending on the scope of your idea/requirement, this could kick off an entire project or a smaller work item.

Once the business analyst has assisted you with your initial requirement, the request is logged and it then needs to be prioritised. This process is accelerated when all of the required information is at hand, and can then be refined. Engagement with the subject matter expert or requestor is key at this stage.

The development team then assesses the prioritisation in terms of an organisation-wide perspective and, once the work item is close enough to the top of the list, a deeper cycle of analysis is kicked off. This feeds the development cycle through a series of planning sessions that ensure that every sprint contains the highest business value items (with the lowest amount of risk).

Sound familiar?

By now, this process should sound slightly familiar to any investment analyst:

- Deals are sourced through engagement with stakeholders, contextual analysis (including company research, legal and financial analysis);
- Deals are created in some central repository;
- Deals are initiated, resources are deployed, and then the review process comes into play to ensure quality and business value.

Substitute “requests” for “deals” and you could be discussing the Agile software development process. It is evident that the two processes and their proponents have much to learn from each other. This form of cross pollination is at the very core of the Agile mind set.

How to stand out

The orchestration of the complex choreography involved in getting the highest return for our investment through development requires stimuli. The most important of these is *detail*. In order to get your idea implemented, it is critically important to be able to express your vision as a business case, clearly and succinctly.

The prioritisation process is more than just a comparison of business benefits in absolute financial terms. Freeing up investment analyst time for research and analysis, and reducing financial and reputational risk for the business are always front of mind. Being able to demonstrate expenditure saved or profits realised in hard currency goes a long way to getting the project noticed and off the ground.

The second fuel source is *engagement*. Being willing and able to engage on your ideas and to provide detail and context, and considering yourself part of the team that collectively adds colour and dimension to the conceptual benefits, can result in this work gaining priority above other less defined or developed reveries.

What's next?

This topic is too broad to cover in one sitting. In the next article in this series, I plan to provide more detail around each of the horizons at which planning and prioritisation occur, and, more specifically, the process of developing a business request to the degree of sophistication needed to action it effectively within a development sprint.

Published on www.futuregrowth.co.za/newsroom.

Futuregrowth Asset Management (Pty) Ltd (“Futuregrowth”) is a licensed discretionary financial services provider, FSP 520, approved by the Registrar of the Financial Sector Conduct Authority to provide intermediary services and advice in terms of the Financial Advisory and Intermediary Services Act 37 of 2002. The fund values may be market linked or policy based. Market fluctuations and changes in exchange rates may have an impact on fund values, prices and income and these are therefore not guaranteed. Past performance is not necessarily a guide to future performance. Futuregrowth has comprehensive crime and professional indemnity in place. Performance figures are sourced from Futuregrowth and IRESS.