

Cut the crap and get us to the backlog!

I have often written about how absurd Agile is. This paper looks at the silliness in Agile frameworks surrounding backlog creation.

Most popular Agile frameworks start from the point of having a backlog which begs the question of what needs to be done to get to that point. Creating a framework from the point of having an ordered backlog is the Agile equivalent of the "fable" of antipodes. In the fourth century, the fathers of the western church denied that the earth was round. In the twenty-first century, the fathers of Agile deny that there is any work before backlog creation, yet in my experience, most Agile activities begin way before that point. The complex part of an Agile initiative is deciding what to build and what needs to be in the backlog; it's not the development process per se. As agilists, we have allowed statements that are manifestly wrong to go unchallenged and what's more, we reinforce these statements by thorough training and certification.

Consider the following statement from the Scrum Guide 2020. "*A Product Owner orders the work for a complex problem into a Product Backlog*". Exactly how does a Product Owner begin to understand the needed product and, therefore, what work should be in the backlog? What guidance is provided by the framework to help the product owner with this complex problem? Finally, how does the product owner steer the path to the goal or the sun? These are all questions where product owners need help, yet it is not forthcoming.

Development doesn't happen in a vacuum. To fund a team, an organisation or sponsor must know what is needed and at least suggest a path to steer.



The concept suggested by some frameworks that an Agile team owns the product is flawed. For example, will the Agile team set the organisational new product strategy, evaluate competitors' actions, pricing points, and discount strategies? That degree of product ownership in a development team is unlikely. Therefore, they do not own the product alone; ownership is shared with other parts of the organisation.

Products also cannot simply be created because of a customer's suggestion. Products and services must align and support the organisational brand values. For one moment, let's imagine that a team at Bentley Cars has a request for a beach buggy or three-wheeled car. Product marketing would view the request and see that developing the requested new product would damage their prestige brand and politely decline the customer's suggestion. Yet the innovation of new products that align with the brand promise is an imperative.

Consider this statement from Steve Denning. "*Organisations around the world are reinventing themselves so they can innovate more quickly and effectively through the use of*

Agile thinking and practices. But, of course, organisations must reinvent themselves at a pace which aligns with market conditions yet outstrips the speed of their competitors. So, the importance of organisational reinvention with new products and services cannot be understated. So how do the popular Agile frameworks guide this activity?

Again from the Scrum Guide 2020, *"Various processes, techniques and methods can be employed within the framework. Scrum wraps around existing practices or renders them unnecessary. Scrum makes visible the relative efficacy of current management, environment, and work techniques so that improvements can be made."* Given the importance of organisational reinvention, do you think more support or guidance could be given? Which processes should be employed? How does a team wrap Scrum around strategy, product definition and brand reinforcement? How does Scrum make visible the efficacy of current management? Many accept and advance these statements, yet are they the Agile equivalent of the emperor's new clothes?

One answer is to use a framework that links strategy through Agile team delivery to operational change management.

Agile Lineout is one such framework created for non-tech teams but links the strategic plans to the team backlog. It also recommends feedback to the team based on the strategic outcomes from their activities using progression metrics so they can adjust their strategies or pivot. A process that mirrors the Plan -> Do -> Study -> Adjust of the Deming cycle.



Strategic planning, blue ocean thinking, and competitor analysis may be undertaken at a frequency suitable for the organisational context and the market sector. For example, capital goods markets may require a less frequent strategic evaluation than the fashion sector.

In British Telecom, where I recently coached, strategic evaluation and options analysis was a continuous activity. Yet, a quarterly review, progress check and adjustment at the team level were deemed sufficient.

The Strategic plans are broken down into Outcomes and Key Results (OKRs). Using Agile increments, OKRs may be defined for the strategy. For example, a strategic goal may be establishing new markets to provide revenue growth of 20%. This over-arching goal may be further broken down by quarter. For example, the quarter three outcome may be that revenue from the Kenya market reaches £ 20 million.

That £ 20 million quarter-three revenue target for Kenya is further broken down using benefits mapping.

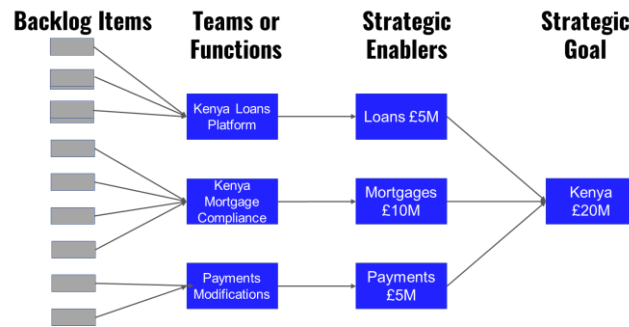
First, the strategic enablers are identified, and in the example below, there are three enablers: Loans, Mortgages, and Payments. Each enabler has its OKR with the team, individually responsible for the identified benefits. In this way, each product owner (the outcome owner may be a better title) may understand how their activity contributes to the strategic goal, and backlog items may be prepared accordingly.

Notice that each backlog is linked to strategy and that the outcome for the team embraces development and realisation.

Each team in this model is genuinely cross-functional, perhaps involving marketing, cultural change or organisational development, in addition to software engineering.

Agile Lineout offers guidelines for OKR definition, quarterly reviews, goal setting, and benefits mapping. These elements help

product owners and teams understand the path to the sun. In addition, Agile Lineout recommends using Tom Gilb's Planguage, a specification language used to describe requirements, designs and plans for a system. Planguage and its sister technique Planalysis enable a team to refine, clarify and question requirements whether they are for software projects or not.



Additionally, the cross-functional team includes responsibilities for the benefits realisation. Agile Lineout team responsibilities do not start with a prioritised backlog or stop at release or deployment but embrace every aspect of outcome delivery. Consequently, during Inspection and Context Assessment activities, the team collectively assess their progress using progression metrics, asking themselves if they will attain their OKRs.

So, for example, if the loans team starts to forecast that they will not reach their £ 5 million goal in their Context Assessment, which is an activity in the Agile Lineout framework, they may decide to change their plans and pivot to reverse their forecast. In this way, Agile Lineout not only states the principles of solution delivery but also provides tools and techniques that teams may use from the outset and then improve upon based on experience.

In conclusion, why not look to employ elements of getting to a backlog from Agile Lineout? Contact us for training or advice on using Agile Lineout to create a backlog for business change activities or software engineering initiatives.

For further information regarding Agile Lineout, please visit <https://agilelineout.com/>. Alternatively, contact Jonathan Ward and [Beneficial Consulting](#), details below.

Jonathan Ward

Email: Jonathan.Ward@Beneficialconsulting.co.uk

Web: <http://www.beneficialconsulting.co.uk>

Mobile Tel: +44 (0) 7802 884598