

James Saffron



James Saffron Consulting

White Paper Enterprise Agile Adoption

James Saffron

Consulting is formed of technologists who are proven experts in their fields, delivering world class projects across financial services. Our leadership team consists of a CTO who has risen through the ranks from development, and a recognised test expert in finance. We are experts in innovative Agile Transformation for our clients bringing a wealth of benefits such as hugely reducing time to market and exceeding stakeholder expectation.

We help companies build environments where change is encouraged, new technologies and platforms are discussed and the culture is that of success.

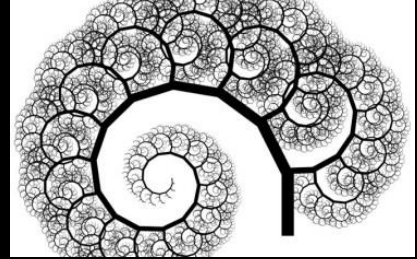
This example will showcase just one success story and show you how easily this could be applied to your organisation.





Enterprise Agile Adoption:

How the success of agile delivery in multiple technology teams scales to enterprise level



Background

MOST ORGANISATIONS employ an annual or biannual planning cycle to achieve targets. For IT, this implies finalising and approving a backlog of IT work at the beginning of a year. Through the year, you are no doubt familiar with the myriad of reports that start at project level and roll up to programme, department and ultimately enterprise level.

Executives don't have the time or inclination to observe projects or programmes from close quarters. They do not monitor progress day-to-day. They want a snapshot of how a department, programme and all projects are doing at a given point in time. This allows them to observe, approve and manage funding, resource allocation, and track against annual plans.

Business Outcome Focussed

AGILE DELIVERY measures success on business outcomes of a project or feature delivery. Some of the key metrics being:

- What ROI was achieved?
- What was the increase in production throughput?
- How many systems were consolidated and rationalised?
- What savings were made across a specified time period?
- What was the increase in business opportunities?
- How much was saved in procurement costs?

MISCONCEPTIONS OF AGILE

- Agile methodologies are not suitable for the biggest problems and organisations?
- The cooperative, iterative and user-focused approach to developing software is often seen as something for smaller teams and organizations.

MISCONCEPTIONS ANSWERED

- The cooperative, iterative and user-focused approach to developing software works with smaller firms and scales like a **fractal to larger** organisations
- In Agile, the same trade-offs and methodologies apply to different levels of scale in the organization. For example, a single scrum team may consist of **seven to nine people** and plan in two-week iterations / sprints with user stories. While a single Agile **program** may consist of **seven to nine scrum teams** and plan in one-quarter iterations with customer features.
- As an enterprise or project grows, Agile will cause teams to lose sight of big-picture goals, such as managing demand, architectural runway, database standards, dependencies, and strategic planning.
- Planning at the **development team level** scales to planning sessions at the **Project** and **Portfolio level** with agile enterprise metrics that retain the focus on company KPI's.



Enterprise Agile Planning

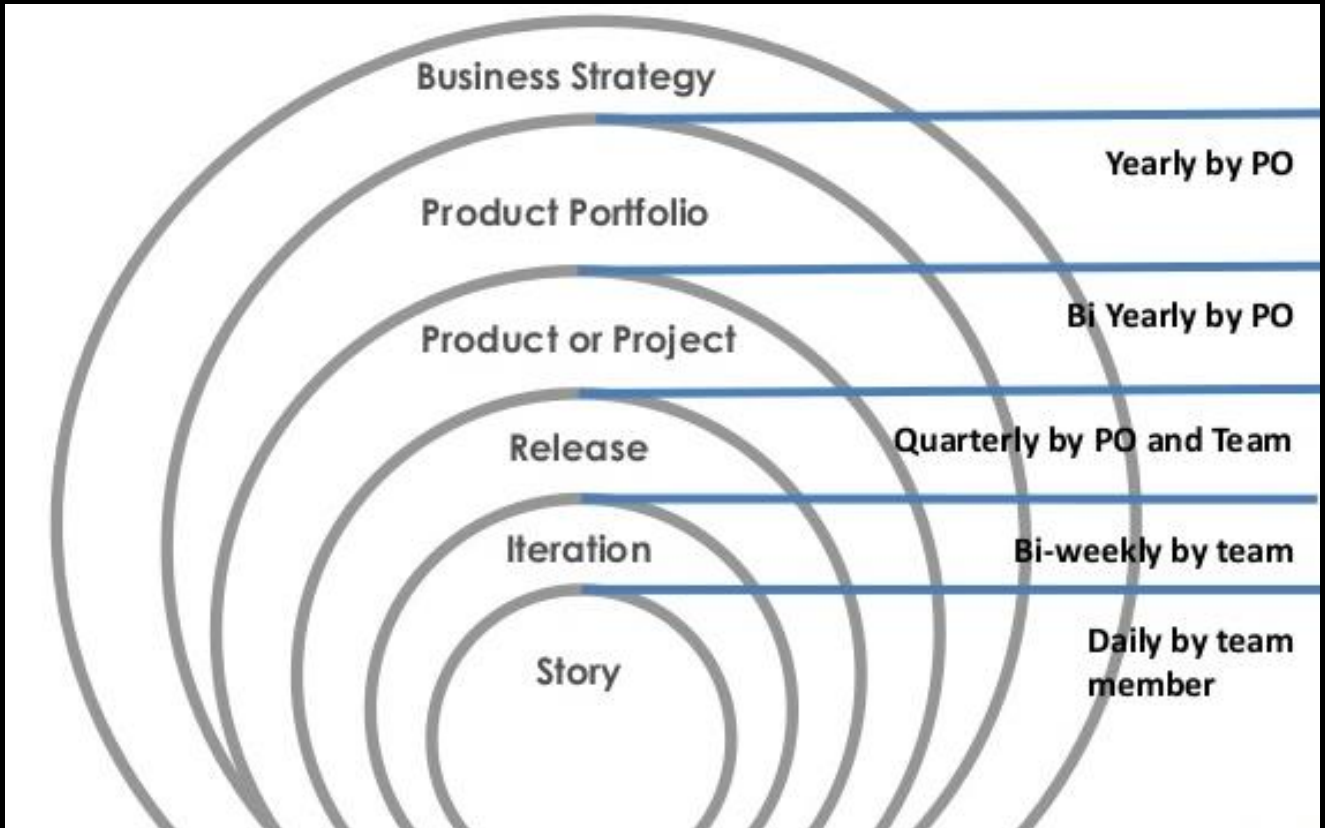


Figure 1. Levels of Agile Planning

AGILE PORTFOLIO PLANNING

Agile portfolio planning at scale implies that the strategy is defined by the enterprise and decisions on how much to invest are driven by value measurement as projects are incrementally released.

MOVING AWAY FROM TRADITIONAL PLANNING

Organizations are moving away from traditional planning in order to capture maximum value and avoid unnecessary overheads / budgetary spending.

Strategic planning with an agile mindset can provide a transformational shift that if scaled to an enterprise level can be a game changer.

Then and only then will you see improved operational effectiveness through the continuous alignment of the business and technology teams.

The conversation moves from where are we with a project, to how much value have we captured so far? If we have captured 80% of the value and only spent 50% of the budget should we stop there and go for something with a higher value to effort ratio?

STRATEGIC LEVEL PLANNING

This takes the overhead out of constantly having to refactor capacity / resource management, simplifying the process and empowering the portfolio level to make decisions aligning with the corporate strategy.

PORTFOLIO LEVEL PLANNING

Portfolio Managers are empowered to determine which projects to work on, so long as there are monthly check-ins with executives to make sure that the organization is on track and in alignment.

PRODUCT LEVEL PLANNING

The Product owner here defines a product roadmap of delivery based on the Portfolio plan. This roadmap plans multiple **releases** in a prioritised backlog of product features.

SPRINT / ITERATION LEVEL PLANNING

Here the team working with the Product Owner and their Scrum Master define short fixed-length subsets of releases, typically 2-4 week time frames that each deliver value to the roadmap.



Enterprise Agile Artefacts

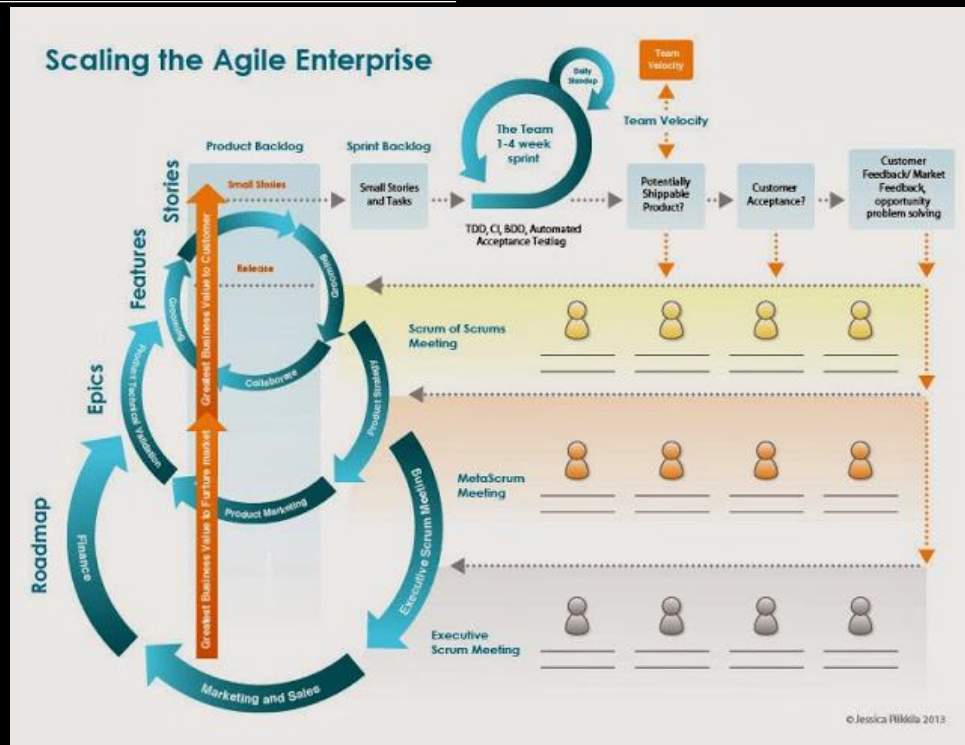


Figure 2 Enterprise Agile Artefacts

Roadmap

The **product roadmap** is a strategic product planning tool that shows how the product is likely to grow across several major releases. This creates a continuity of purpose, facilitates stakeholder collaboration, helps acquire funding, and makes it easier to coordinate the development and launch of different products.

Ideally this is a goal-oriented product roadmap that clearly states the benefits provided by each major release.

Epics

These describe the reasons why a project has been initiated, the objectives that the project will achieve, and the metrics that will be used to measure its success. Epics are comprised of a large collection of features.

Features

These comprise:

- Capabilities that the Product Owner is interested in
- An outline of the value to users
- A collection of user stories

Stories

Stories describe how various stakeholders will interact with the solution and the needs they have in performing their assigned tasks and activities. They should contain a set of documented outcomes that will comprise the tests for successful delivery.

Product Backlog

The Product Backlog is an ordered list of everything that might be needed in the final product. In other words, a wish list of requirements. All items are described in simple, non-technical, business language, and all of them are presentable to every stakeholder.

The Product Backlog is always dynamically changing and improving, it is never complete. Product development should not wait until the backlog is complete before starting delivery of the requirements. The first Sprint can be started as soon as the Product Backlog has enough stories defined.

Sprint Backlog

The Sprint Backlog is created during the Sprint Planning event, which is the first event in a Sprint. During the Sprint Planning event, the development Team collaborates on creating the Sprint Backlog, based on their estimated work and the estimated capacity of the Team.

The Sprint should have a **Goal**, which will help describe the real meaning of the items and direct the efforts of the Development Team.

The items in the Sprint Backlog are frozen after the Sprint Planning, and the Development Team will focus on delivering an Increment of "Done" based on this plan. Although the stories are frozen during the sprint it might be necessary to get more information and justify or remove some of the items. This should be done with the Product Owner.



Measuring the value to the company

Measuring Delivered Value is the most important emphasis on metrics for Agile software development. Merely tracking to a plan is not sufficient to manage risk. Thoughtless allegiance to the original plan, in fact, could assure failure if market conditions (mission parameters) or assumptions made while building the plan have changed. The iterative nature of Agile methods protects sponsors and developers from this trap.

In measuring the value to the enterprise we suggest metrics such as:

- Release burndown
- Sprint burndown
- Number of Features Accepted
- % of Features Completed
- User Satisfaction on delivered stories
- Feature Cycle Time – how long similar sized features take to be delivered to production
- Defects discovered after deployment
- Revenue/Feature

And to avoid metrics such as:

- Lines of code
- Story points delivered (Velocity)
- Hours Worked
- Number of defects in development

RELEASE BURNDOWN

With each completed sprint, the delivered functionality grows, and the release burn-down chart depicts this progress.



Figure 3. Release burndown chart

SPRINT BURNDOWN

As stories in the backlog of work are completed, the chart displays the rate and amount of progress.

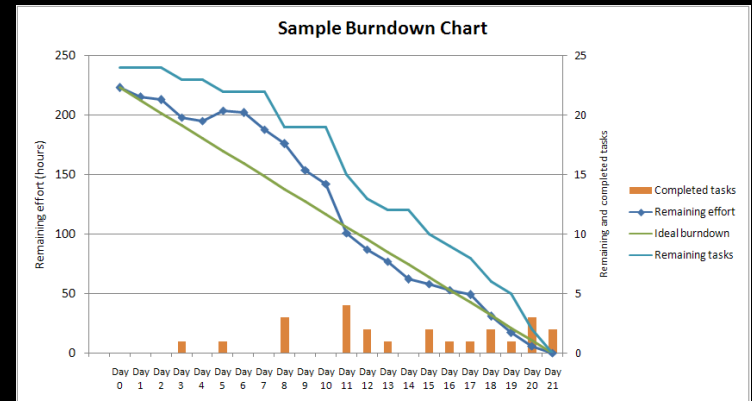


Figure 4. Sprint burndown chart

EARNED VALUE CHART

Earned Value reporting **measures whether the amount of money spent** through so far in a project **justifies the amount of work completed** at this point in time.

There are a few variables here:

Budget – the estimated cost of your project. This is usually decided at the beginning of the project, and reviewed infrequently or not at all.

Actual cost (AC) – the proportion of the original budget your team have spent so far.

Planned Value (PV) – the proportion of your project scope that was expected to have been delivered by this time.

Earned Value (EV) – the 'real' value of the scope that has actually been delivered so far.

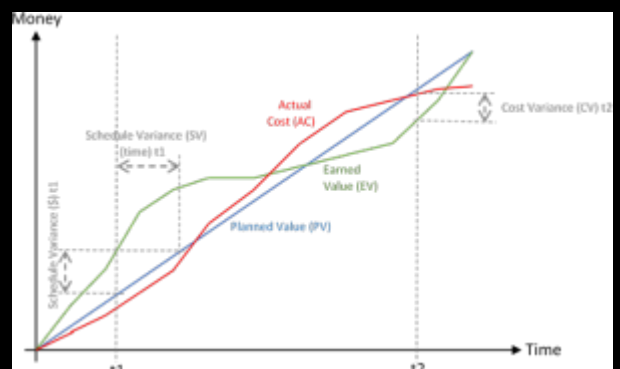


Figure 5. Earned value chart



A Success Story: Enterprise Agile Adoption

How the success of agile delivery
in multiple technology teams
spread to the enterprise

The Challenge

OUR CLIENT'S Agile adoption had spread throughout the technology department but now needed maturing. The client needed help with:

- How to organise and safeguard delivery across multiple delivery teams
- How to align delivery with various business functions
- How to safeguard delivery across the enterprise for traditional, Agile and third party projects
- How to report on relevant metrics for the monthly board pack that relates to the organisation's KPI's

Solution

- Consolidation to four key portfolio streams
- Business Product Owners – working as Product Managers of those portfolios
- Product Owners from the business leading individual feature delivery teams within that portfolio
- Multiple Scrum teams per portfolio – with twice monthly joint planning and review sessions
- Each Portfolio inherits board level KPI's and each feature development across and within teams is prioritised according to the ROI versus estimated effort



Metrics

- Executive Cross Portfolio pack contains per portfolio:
 - Release burndown ■ Sprint burndown
 - % of Features Completed
 - User Satisfaction
 - KPI vs ROI vs Cycle Time
- Drilldown to Portfolio level contains per scrum team in the Portfolio:
 - Sprint burndown
 - % Velocity in story points
 - % features completed Feature Cycle Time
 - Defects discovered after deployment

Results

- Delivery focused on actual business benefit
- Transparent metrics
- Aligned deliveries
- Empowered high performing teams
- Key Business Owner engagement
- Agile responsiveness to change



Conclusion

More and more, organisations and their systems are moving away from traditional waterfall development practices in favour of Agile methods. Agile methods are effective for shortening delivery cycles, delivering incremental value as early as possible in the development lifecycle and managing costs. If the benefits of agile are to be realised effectively for these organisations, however those leading these enterprise wide agile development programs and portfolios must be fluent in metrics used to monitor these programs.

We continue to learn new and inventive ways of demonstrating progress and diagnosing performance from agile implementers. The value of this approach is that it represents a narrative driven by real-world experience.

With its different rhythms, patterns, roles and processes, Agile Scrum is inherently more measurable than prior documentation-oriented, indirect, waterfall-based measures of progress.

Of course, the best measure by far comes directly from *working software and solutions*. It's best for teams, Agile Release Trains, managers, program management and portfolio managers to pivot most of their measuring attention to this critical fact. Other metrics outlined above – are subordinate to the objective and overriding goal of keeping the focus on rapid delivery of quality, working solutions that deliver the identified value and return on investment.

Nevertheless measures are important in the context of enterprise and to that end we provide some of the approaches that we have implemented successfully in practice in this white paper.

Summary

THIS IS JUST ONE of many examples that we have to highlight the impactful changes we can make by helping your enterprise in its digital transformation journey

- Agile methodologies are suitable for the biggest problems and organisations in this world.
- Agile enterprise delivery focusses success criteria on business outcome of a project or feature delivery. Some of the key metrics
- The approach outlined here provides transparency and the effective means to observe, approve and manage funding, resource allocation, and track against annual plans at enterprise, department and programme-level.
- Agile portfolio planning at scale implies that the strategy is defined by the enterprise and decisions on how much to invest are driven by value measurement as projects are incrementally released.

To transform your business get in touch with us:

Email: info@jamesaffron.co.uk
Mobile: 07984 442883

James Saffron

