Product Owners – The key to Safe Agile

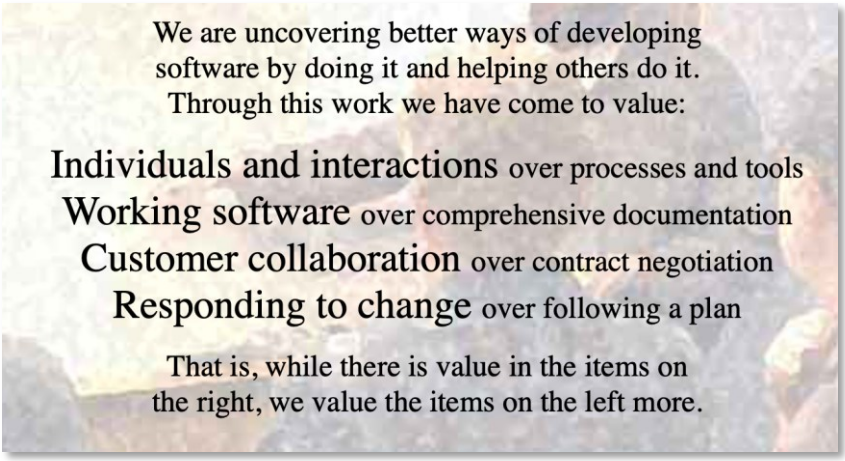


Bill Blackburn develops the theme of Agile in the context of high integrity and safety systems, exploring the role of the Product Owner. This role is pivotal in Agile and is relied upon to orchestrate and oversee the build aspects of the lifecycle, ensuring that requirements are reflected in the product and that relevant stakeholders are represented.

The Safety-Critical Systems Club (SCSC) ran an event – Safe Agile Developments – on 1st May 2025, held at the Wellcome Collection in London. See page 29 in this edition of the newsletter for the event report. This one-day seminar looked at techniques and approaches for managing Agile developments for safety-related systems.

This seminar was for those introducing or using Agile techniques with high integrity and safety-related developments, in e.g. Healthcare, Policing, Transport, Government or Defence areas. It explored the balance between agility and the demands of systems operation in situations where safety and business criticality requirements applied. The discussions touched on the challenges of blending the safety engineering disciplines with an Agile approach. Speakers and participants explored how the Agile manifesto can be interpreted and adopted/adapted.

To give its full title, the Manifesto for Agile Software Development" [1] has its origins in 2001 when a group of experts sought to establish a practice-driven philosophy for more effective software delivery and came up with their four key values. This is often quoted by agile evangelists, however the manifesto has much more to say elaborated through a set of 12 principles.



A thread running through the Agile Manifesto is about having the right skilled and empowered people involved throughout and collaborating in self-organising teams. Done poorly, with Agile evangelists proclaiming the benefits of agile vs "old school" waterfall can result in (FR)Agile delivery and chaos. Not a good recipe in the world of high integrity and safety.

So how can appropriate working practices be established that provide a level of confidence for the safety community?

Evolution of Agile



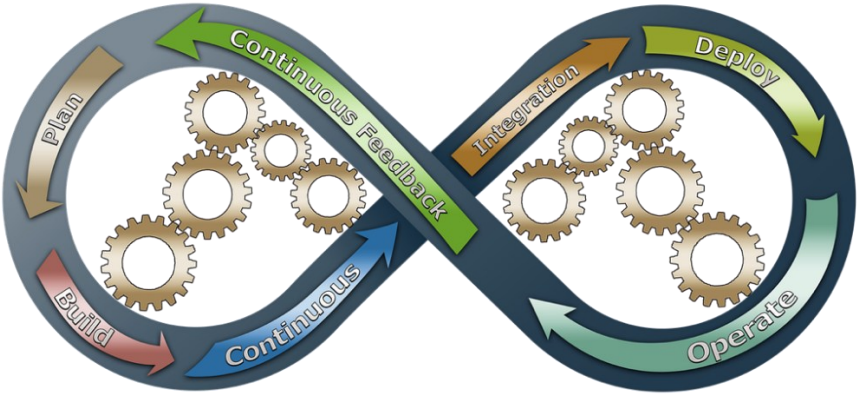
My paper called Scaled Agile for High Integrity Services [2], at the SCSC seminar talked about the evolution of Agile from the groundwork done by Tom Gilb in the 1980s through to Scaled Agile Framework that has become a well-respected set of tools and techniques covering the range of needs across portfolios, programmes and projects.

Gilb's work centred on the concepts of evolutionary delivery [3], and he also complemented this with his detailed book on Software Inspection [4]. It might appear odd to mix the flexibility of agile and the rigour of inspections however Gilb's approach was more about programme level evolutionary delivery rather than software development where the Agile manifesto tends to focus. In a large space systems project I found that the discipline of formally run inspections was extremely helpful to the development team who were able to use the cadence to drive their build iterations.

During the Safe Agile Development seminar, we discussed the various roles that are common across the Agile methods and in particular the Product Owner. With the focus on incremental delivery of a "product" towards a releasable state, the Product Owner ensures that requirements are reflected in the product backlog and oversees the sprint refinement sessions and delivery of product increments.

The Product Backlog contains user stories that translate functional and non-functional requirements into descriptive explanations and acceptance criteria. These are prioritised and elaborated with stakeholders through analysis and design sessions. Once features can be demonstrated, 'show & tell' sessions allow early feedback and refinement. Requirements are typically documented as functional or non-functional statements or outcomes to be achieved. They should avoid prescribing solutions or unnecessarily constraining design and build, though wider architectural principles may provide context.

User Stories are commonly captured in web-based collaborative tools such as Azure DevOps or Jira, which are industry standards used by business and technical teams to support the cycles of build, integration and deployment, as illustrated in the diagram below taken from Microsoft DevOps documentation [5].



Nothing fundamentally unsafe so far?

Platforms and Products

Building a product entirely from scratch is very different from building a product on an existing platform. Platforms need architectural and design skills that are wider than the skills needed for product increments once a platform exists, which can take years to mature. Most sectors including those where safety is paramount such as aerospace and automotive have taken a platform-based approach for many years. Modular and component-based products are commonplace to promote reusability and the advantages that can bring around cost management.

Recent examples exist in the electric vehicle domain where manufacturers are investing heavily in platforms. For example, in May 2025 Volkswagen [6] announced, "The Scalable Systems Platform (SSP) will represent the next generation of an all-electric, fully digitalized, highly scalable mechatronics platform based on a uniform system architecture." Platforms can underpin a whole range of products and not just those of one manufacturer but shared with collaboration partners.

This has some downsides from a safety viewpoint, if for example a component is used in a different way or if there is an inherent issue not apparent in one product, but manifests itself over time across the product range.

The Toyota accelerator pedal used across numerous vehicles springs to mind and resulted in worldwide vehicle recalls in the period 2010-12, one of the largest in automotive history. Along with reputational damage to one of the world's largest corporations [7]. In this case, the safety issue was unintended acceleration.



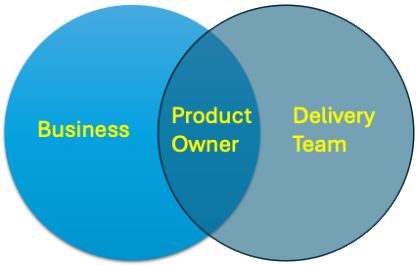
Product Manager / Product Owner

Product Managers handle the overall product lifecycle often from inception to end-of-life and that generally follows an established path. This role is fairly commonplace. For example, in 1998, as part of its initiative ambitiously titled Smart Acquisition, the UK Ministry of Defence introduced an approach termed Integrated Project Teams (IPT). This was part of a new way for defence equipment acquisition. These weren't really dealing with projects and were in fact Integrated Product Teams, where the IPT lead played the client-side role of a Product Manager with supplier-side equivalents. IPTs eventually operated across the whole lifecycle and in fact were pivotal in through life support for equipment. A helpful analysis is provided in the NAO review Implementation of Integrated Project Teams [8].

The Agile Product Owner can be regarded as a Product Manager, or potentially one of a handful of roles working with a product team, depending on the nature of the system or service.

It is best to think of the Agile Product Owner as the responsible owner for the lifecycle stage within which the build is undertaken, through to acceptance and transition/release into operational use. Following which, the Product Owner handles the ongoing evolution, cycling through product increments (or not if the budget has run out). Generally the Product Owner would sit on the business or client side with counterparts in the in-house or outsourced delivery team.

During build, the Product Owner is the person who orchestrates the process by managing the product backlog and getting the right stakeholders involved at the appropriate points. The Product Owner works collaboratively with the Scrum Master (in Scrum terminology, the scrum team are a self-organising delivery team) who oversees and coordinates the resources within the scrum team or other specialists as required.



In the Scaled Agile Framework (SAFe) [9], the definition is:

The Product Owner (PO) is the Agile team member primarily responsible for maximizing the value delivered by the team by ensuring that the team backlog is aligned with customer and stakeholder needs.

Reads like a nice simple definition but lurking within are wide implications and demands on skills. Furthermore, recruiting the right Product Owner can be a challenge, particularly so if looking for one already with a safety engineering background. Ensuring that a Product Owner coming into an organisation dealing with high integrity or safety systems is familiar with the disciplines and requirements. Realistically the Product Owner needs to be skilled at awareness level across a range of domains and safety aspects would need to be treated in a similar manner. Keeping in mind that the Product Owner would be relying upon those skilled and qualified people with safety responsibilities and manage them as stakeholders to be involved at the right time.

Requirements Traceability and Safety User Stories

From a safety engineering viewpoint, the relevant safety requirements (standards, policies, etc) need to be identified and agreed from the outset. The necessary skilled resources need to be lined up and available to provide direction and potentially act in verification & validation roles. This is within the Product Owner responsibilities.

"Realistically the Product Owner needs to be skilled at awareness level across a range of domains and safety aspects would need to be treated in a similar manner"

Agile methods rely on "User Stories" and these are written, reviewed and approved then used to direct build activities whether those are manufacture or other form of product development. User Stories are generally thought of as functional specifications written as acceptance criteria usually in the structure:

> "As a [type of user], I want [goal] so that [reason/benefit]" Taking the earlier accelerator pedal scenarios: "As a **driver** I want to be able to **control engine power** so that **the vehicle speed can be varied**".

User Stories can equally be written as non-functional requirements and instruct how aspects such as safety requirements are to be implemented using an equivalent structure:

> "As **an automotive safety engineer**, I want the **engine control system to comply with ISO 26262**, so that **risks X, Y, Z in the Product Omega FMECA are mitigated**"

More elaborate acceptance criteria would be required, but that's done through refinement sessions as analysis and design progresses. User stories are progressed through the backlog and the Product Owner would need to ensure that those representing essential requirements are prioritised alongside the functional aspects.

Functional user stories tend to be implemented through sprints where functionality is configured/built, and the first stages of testing are performed.

Product Assurance

Agile methods tend to skip over the broader aspects of Product Assurance and the disciplines of verification and validation. Testing within sprints is performed and the integration and build activities ensure that those aspects are assured. This is where a Product Assurance Manager needs to step in, working alongside Product Owner or part of the product team.

"Agile methods tend to skip over the broader aspects of Product Assurance and the disciplines of verification and validation"

My experience where the role is scoped as a "Test Manager", their perspective is subsequently somewhat narrow, and the resulting test-based assurance is not sufficient for high integrity products. Product assurance on the other hand is a set of carefully planned and executed activities running in parallel with the build/configuration, integration and release cycle that runs through the core of Agile methods.

Product Assurance standards are particularly well-defined in the aerospace sector for example, ISO 22893 [10], and these can be readily applied alongside Agile build as found in research [11] that examined how agile attributes and principles can help achieve high European Cooperation for Space Standardization (ECSS) compliance levels.

A Few Takeaways:

This article has been based on personal experience across a range of sectors, while keeping an eye on the ups and downs of Agile project delivery.

- Be wary of evangelists, especially those who pin up sheets proclaiming waterfall bad / Agile good
- Agile requires highly skilled and empowered people who know how to collaborate where the Product Owner takes a leadership role and needs at least awareness-level of safety knowledge
- Platforms upon which to be Agile are vital. Business, information and technology architecture is needed to create a platform upon which product increments can happen
- With the right pre-requisites in place, safety engineering can be applied. The Agile team need to appreciate the demands of safety, and the safety engineer needs to feel comfortable with Agile approaches, adapting techniques where appropriate

References

[1] "Manifesto for Agile Software Development" Jim Highsmith et al, Agile Alliance https://Agilemanifesto.org, February 2001

[2] Scaled Agile for High Integrity, Bill Blackburn SCSC Safe Agile Seminar, May 2025, https://scsc.uk/r3107

[3] Evolutionary Delivery vs the "Waterfall Model", Tom Gilb ACM SIGSOFT Software Engineering Notes Vol 10 No 3 Jul 1985 https://dl.acm.org/doi/pdf/10.1145/1012483.1012490

[4] Software inspection, Tom Gilb and Dorothy Graham, Addison Wesley, 1993 ISBN 0-20163181-4

[5] Microsoft DevOps https://learn.microsoft.com/en-us/devops/overview/

[6] Volkswagen Future Plan – Scalable Systems Platform https://www.volkswagen-group.com/en/articles/volkswagen-future-plan-brand-gives-first-glimpse-of-entry-level-electric-model-at-works-meeting-18997

[7] FBI press statement on Toyota Motor Corporation https://archives.fbi.gov/archives/newyork/press-releases/2014/criminal-charge-against-toyota-motor-corporation-and-deferred-prosecution-agreement-with-1.2-billion-financial-penalty

[8] Implementation of Integrated Project Teams, National Audit Office 2022 https://www.nao.org.uk/reports/ministry-of-defence-implementation-of-integrated-project-teams/

[9] Scaled Agile Framework (SAFe) Product Owner, https://framework.scaledAgile.com/product-owner Scaled Agile Inc.

[10] ISO 22893:2022 Space systems — Software product assurance (SPA)

[11] An Agile Compliance Framework for the European Cooperation for Space Standardization Al-Mhdawi et al, 2023 IEEE Aerospace Conference

Image Attribution:
leading image: AI generated by ChatGPT
Agile manifesto from https://agilemanifesto.org
DevOps Image Dirk Wouters on Pixabay
Pedal adapted from: ID 105521154 | Car Pedals © Asawin Kanakasai | Dreamstime.com

Bill Blackburn, Process Renewal Group Limited and SCSC Member

Bill has a background in mechanical engineering and worked across aerospace, manufacturing, defence and other high integrity sectors for over 40 years. He is presently engaged in advisory roles with clients in UK policing and capability development in Europe. He has extensive experience in business and information architecture, with a focus on process and data management.