

Thermo Fisher Scientific: Lesson #7 – Product Owners Don't Write Most Stories

One of the predictions from my earlier research into Agile hardware development was that Product Owners would write a much smaller percentage of Stories than is typical in software development.

The Thermo Fisher Scientific engagement validated that prediction.

In software development, Story-writing responsibilities tend to follow a fairly predictable pattern. The Product Owner owns the Product Backlog and is responsible for deciding what functionality will be developed and in what order.

Because most Sprint work is described through User Stories, and because User Stories describe user-facing behavior, Product Owners typically write a large percentage of the Stories that enter the backlog.

Technical Stories certainly exist in software development, but they usually represent a minority of the overall work. These Stories are often written by developers because they require technical knowledge that Product Owners may not possess.

This arrangement works well because software Product Owners frequently come from Product Management, Marketing, or other business-oriented backgrounds. They understand customer needs and product behavior, but are often less involved in the technical implementation details.

The Thermo Fisher project operated differently.

As described in previous articles, the team produced almost exclusively Technical Stories. The work focused on components, subsystems, interfaces, prototypes, and engineering deliverables rather than on user-facing functionality.

As a result, there were very few User Stories for the Product Owner to write.

The second difference involved the Product Owner himself.

Dr. Michael Belford was not a traditional Product Owner drawn from a marketing or product-management background. He was a physicist with deep technical knowledge of the product and also participated directly in the development effort as a part-time team member.

This combination of factors produced exactly the pattern that my research had suggested.

Because all of the Stories were Technical Stories, every team member was capable of writing them. Story creation naturally became a shared responsibility across the team. The Product Owner participated in Story writing, but he did not dominate it.

In fact, because there was only one Product Owner and multiple engineers, most of the Stories were written by team members rather than by the Product Owner.

The Product Owner still owned the Product Backlog. He still made prioritization decisions. He still determined what work would be performed and in what order.

What changed was the distribution of Story-writing responsibility.

The lesson was clear.

Lesson #7: In hardware development, Product Owners often write only a minority of the Stories. Story-writing responsibility is distributed much more evenly across the Product Owner and the engineering team.

This pattern occasionally appears in highly technical software environments where Product Owners possess strong engineering backgrounds. In hardware development, however, it is much closer to the norm.

The Thermo Fisher experience provided an early validation of this prediction and reinforced the broader conclusion that many Agile practices must be adapted when applied to hardware engineering organizations.

In the next article, I will discuss another prediction that proved correct: why Quality Assurance specialists typically do not test the deliverables produced by individual Stories.

For those interested in the full case study, the complete Thermo Fisher Scientific story is available on my website at <https://kevinthompsonphd.com/storage/papers/LessonsLearned-AgileHardware-v6.pdf>.