

Thermo Fisher Scientific: Lesson #1 – Product Owner as Team Member

"Do you have any experience in Agile hardware development?" I was asked at the go/no-go decision meeting.

I was completely up-front with them.

"No," I replied. "And I don't know anyone who does."

At that point, Agile hardware development was largely unexplored territory. We had decades of experience to draw upon in Agile software development, but very little experience applying Scrum to purely hardware products.

"You will be pioneers," I added. "If you decide to proceed, everyone needs to understand that there may be challenges along the way."

To their credit, they were willing to accept the risk, and the project received approval.

We planned the engagement in the same way we would approach a software project: begin with Scrum training, write high-level Epics for Release Planning, draft the Release Plan, write and refine Stories for the first Sprint, and then begin Sprint execution.

There was an additional challenge. The team members had never worked together before, and most had learned only shortly before the training that they would be using Scrum.

I give them a great deal of credit for their open-mindedness. They were willing to learn and give this unfamiliar process a chance.

At the time, I had not yet developed a dedicated Scrum-for-Hardware course. As a result, I used my standard Scrum training material, changed one slide, and discussed hardware-specific issues at the appropriate points throughout the class.

The Product Owner for the project was Dr. Michael Belford, a physicist who would also participate in the development work as a part-time team member. The project itself involved the development of a new mass spectrometer for Thermo Fisher Scientific.

This led to one of the first important lessons from the engagement.

Lesson #1: In hardware development, the Product Owner is often a team member.

The reason is that ownership of a hardware product typically requires deep technical knowledge of the product itself. The person making decisions about priorities and tradeoffs frequently possesses the same specialized expertise required to contribute directly to the development effort.

This differs from the software model, where Product Owners often come from a product-management background and focus primarily on user needs, market requirements, and business outcomes rather than technical implementation details.

Both models are valid, but they involve different tradeoffs.

In the software world, a full-time Product Owner can often support two or three teams simultaneously. In the hardware world, a Product Owner who is also contributing technical work is unlikely to support more than a single team.

That was the first major lesson we learned from applying Scrum in a hardware-development environment.

In the next post, I will discuss another important discovery: why Release Planning turned out to be far more important in hardware development than many Agile practitioners would expect.

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>.