Thermo Fisher Scientific: Lesson #10 – Why Velocity Is Different in Hardware Development

One of the predictions from my Agile Hardware research was that Velocity would behave differently when teams estimated work in Person-Days rather than Story Points.

The Thermo Fisher Scientific engagement validated that prediction.

Velocity is one of the most important concepts in Scrum. Simply stated, Velocity is the amount of work completed by a team during a Sprint.

Velocity is important because it determines how much work a team can reasonably plan for the next Sprint.

This planning model differs significantly from traditional project management.

In a traditional project plan, tasks are identified, estimated, sequenced according to dependencies, and assigned to available resources. The resulting schedule predicts when the work will be completed.

Scrum takes a different approach.

Rather than building a detailed task schedule for the Sprint, Scrum begins with a forecast of how much work the team can accomplish. The team then fills the Sprint with Stories until that forecasted capacity is exhausted.

To make this process work, the team needs a forecast of its next Sprint Velocity.

How that forecast is produced depends heavily on the units used for estimation.

When Velocity is measured in Story Points, the only practical forecasting method is extrapolation from history. Common approaches include using the Velocity from the previous Sprint or averaging the Velocities from several recent Sprints.

This is necessary because Story Points are intentionally abstract. Since they are not tied to any measurable quantity, there is no direct way to calculate future Velocity from team availability.

Person-Days are different.

Because a Person-Day represents eight hours of engineering effort, future Velocity can be estimated directly from the team's expected capacity.

The simplest approach is to apply a focus factor to the available staff. For example, a team of five people working in a two-week Sprint might be expected to devote only a fraction of its available time to Story work after accounting for meetings, interruptions, and other activities.

A more sophisticated approach, and the one I prefer, is a resource model.

The resource model explicitly accounts for factors such as:

- Sprint ceremonies
- vacations and holidays
- training commitments
- part-time assignments
- expected interruptions
- other known availability constraints

The result is a forecast of how many Person-Days of effort the team can realistically apply to Stories during the Sprint.

This was the approach we used at Thermo Fisher Scientific.

I walked the Scrum Master, Ray, through the process, and he was able to produce the Velocity forecast with little difficulty.

The resulting forecasts proved both intuitive and useful.

This experience reinforced another conclusion from my Agile Hardware research.

Lesson #10: When work is estimated in Person-Days, Velocity should also be measured and forecast in Person-Days. The most effective forecasting technique is a resource model that accounts for the actual availability of team members during the Sprint.

The lesson is not that historical Velocity becomes irrelevant. Historical data remains useful for validating forecasts. However, when estimates are expressed in units of effort, future Velocity can be calculated directly rather than inferred solely from past performance.

In the next article, I will discuss Sprint Planning itself and how the process differed from the software-oriented Scrum model.

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.