

BEYOND THE POINTS

Sprint Velocity Is Not a Programme Status Report

Why turning a team planning aid into executive assurance corrupts both the metric and the work



Giovanni Leonardi

February 2012

Sprint Velocity Is Not a Programme Status Report

Programme · No. 165

Originally written February 2012 | Re-edited and re-rendered for the WhereBeyond Edition, 2025

“Velocity measures a team's relationship with its own estimates; it does not measure the programme's relationship with reality.”

The Green Report Built from Forty-Seven Points

The programme board received a new dashboard. Team A had completed forty-seven points against a forecast of forty-two. Team B had completed thirty-eight against forty. Team C had increased its velocity for the third sprint in succession.

The page looked reassuring. It was also almost meaningless.

Team A had re-estimated several unfinished stories after discovering more work. Team B had spent four days helping operations diagnose an environment fault, work that appeared nowhere in its backlog. Team C had split large stories into smaller ones and therefore counted more completions without delivering more usable scope. Meanwhile, the programme's critical data decision was six weeks late.

Yet the dashboard was green because the numbers were moving in the approved direction.

Sprint velocity is becoming the new status report: a team planning aid lifted out of context, aggregated for executives and treated as evidence of programme progress. In making that move, organisations are repeating the reporting error Agile methods were supposed to correct. They are substituting activity for outcome, only with newer vocabulary.

A Local Measure Escapes Its Boundary

Velocity can be useful within a stable team. It records how much estimated work that team completed in a sprint. Over several iterations, the team can use its own history to judge how much work it is likely to take on next.

Its value depends on boundaries.

- The estimates belong to one team.
- The scale is relative, not absolute.
- The number supports planning, not performance appraisal.
- Changes in team composition, story shape or definition of done can alter it.

Once velocity leaves those boundaries, its meaning deteriorates rapidly. Twenty points in one team cannot be compared sensibly with twenty points in another because the units were created by different people. A rising number may indicate improvement, but it may equally indicate easier work, larger estimates, weaker completion criteria or pressure to show growth.

The programme report removes all that context. It preserves the number and discards the conditions that made the number interpretable.

Velocity measures a team's relationship with its own estimates; it does not measure the programme's relationship with reality.

Why Management Keeps Recreating Status

The appeal is understandable. Programme leaders need a view across several teams. Traditional reports based on percentage complete have repeatedly concealed late integration and optimistic forecasts. Working in sprints appears to offer a more current source of evidence. Velocity is numerical, frequent and already available.

The strongest case for using it at programme level is therefore practical: imperfect measures may still reveal trends. A sudden fall can expose interruption. Persistent volatility can indicate unstable demand. If every team reports in a common rhythm, management can see trouble sooner than it did through monthly narrative reports.

That case has merit. The error is not looking at velocity. The error is asking velocity to answer questions it cannot answer.

A programme board does not primarily need to know whether teams are processing estimated items faster. It needs to know whether the intended capabilities are becoming usable, whether dependencies are clearing, whether assumptions remain valid and whether the programme can still meet its commitments.

Velocity can prompt a question about those matters. It cannot resolve one.

The Number Changes the Behaviour

Metrics used for planning behave differently when attached to judgement. The moment a programme office asks why velocity fell, teams learn that lower is bad. When senior managers praise an increase, higher becomes the unofficial target. Once the target is understood, the estimate ceases to be an honest planning conversation.

A composite team began with an average velocity of twenty-nine. After two quarters of executive reporting, it averaged forty-six. The increase looked like a fifty-nine per cent improvement. Delivered releases had not accelerated.

What changed?

1. Stories were estimated more generously.
2. Technical tasks that had once sat beneath stories were counted separately.
3. Partially tested work was accepted as complete at sprint end and defects were added to the next sprint.
4. Time spent helping another team was excluded because it reduced the reported total.

Nobody had ordered the team to manipulate the metric. The reporting system had simply made the incentive obvious.

This is the familiar danger of every status regime. People adapt the representation of work to protect the appearance of control. Agile language does not make the danger disappear.

Report Evidence the Board Can Use

Programme reporting should begin with decisions and outcomes, then use team measures only as supporting evidence.

A board needs a concise view of:

- Usable capability. What can a real user now do that could not be done before?
- Critical flow. Which end-to-end items have moved through build, integration, acceptance and release?
- Dependencies. Which decisions, interfaces or shared resources are constraining several teams?
- Forecast. What range of scope is credible by the required date, based on actual completion?
- Exposure. Which assumption has weakened, and what decision now follows?

These questions are harder than asking whether velocity rose because they force a programme to define value and confront uncertainty. They also produce information that can change executive action.

Velocity should remain visible to the team and, where useful, to programme leadership. But it should be accompanied by its context and never converted into a common productivity scale. A team whose velocity falls because it is removing a critical integration risk may be protecting the programme. A team whose velocity rises while producing unaccepted features may be endangering it.

Do Not Modernise the Wrong Habit

The move from monthly status reports to sprint dashboards looks progressive. It is not progressive if the governing question remains, “Are people producing enough?”

Agile delivery offers a more valuable possibility: frequent evidence that the programme's beliefs are right or wrong. A demonstration can reveal that users reject a workflow. An integrated release can expose a dependency. A tested increment can improve a forecast. These are signals about reality, not merely about effort.

The board should ask what has been learned, what has become usable and what decision must now be made. If the answer is replaced by a chart of points, the programme has modernised the presentation while preserving the old management reflex.

Sprint velocity is not the problem. Turning it into executive reassurance is.