

BEYOND VELOCITY THEATRE

Velocity Is Becoming the Status Report—and That Is the Problem

Why a useful team measure is being recruited to satisfy programme control in the early agile enterprise



Giovanni Leonardi

April 2012

Velocity Is Becoming the Status Report—and That Is the Problem

Programme · No. 167

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

“Velocity can help a team plan its next iteration; it cannot tell an executive whether the organisation is becoming more capable.”

The number that travelled upstairs

On Monday morning, a programme board receives a new slide. Six software teams have completed their latest fortnightly iterations. One has moved from 31 points to 38; another has fallen from 27 to 21. The slide colours the first green and the second amber. Nobody asks whether the estimates changed, whether the teams are comparable, or whether either team delivered something a customer can use.

By Wednesday, managers are asking the amber team how it will recover seven points.

This is how sprint velocity is becoming the new status report. A measure designed to help a team understand its own delivery rhythm is being lifted out of that context and turned into an executive assurance device. The old red-amber-green machinery has not disappeared. It has simply found a more modern-looking number.

Why this is happening now

Agile delivery is moving beyond isolated teams and into larger programmes. That is welcome. But the expansion exposes a collision between two management traditions.

Programme offices are accustomed to plans that offer apparent comparability: milestones, percentage complete, earned value, defect counts and traffic lights. Iterative delivery replaces some of that apparent certainty with frequent inspection and adaptation. Executives still need assurance, but the familiar pack no longer tells the whole story.

Velocity arrives at precisely the right moment to satisfy that appetite. It is numerical, regular and easy to plot. It appears to answer three questions at once:

- Are the teams productive?
- Is the programme accelerating?
- Will the release arrive on time?

In fact, it answers none of them reliably outside the estimating team. Velocity is the amount of work a particular team completes in an iteration, expressed through that team's own relative estimates. It can support short-range planning when the team is reasonably stable and its estimating habits are consistent. That is useful. It is not a universal unit of output.

The mechanism of distortion is straightforward. Once a local planning measure becomes a target, the easiest route to improvement is often to change the scale. A team can split stories differently, revise what it calls a point, carry less unfinished work across the

iteration boundary, or become more generous in estimation. The chart rises while the underlying rate at which useful capability reaches operations remains unchanged.

A measure ceases to be diagnostic when the people being measured must make it look reassuring.

The comparability trap

Consider two composite teams working on the same policy administration programme in 2012. Team A has seven people and completes 42 points in an iteration. Team B has five people and completes 24. Team A appears 75 per cent more productive.

Yet Team A estimates a routine interface change at eight points; Team B would call comparable work three. Team A also counts testing inside the story, while Team B records part of it separately. After three iterations, Team B has put two complete changes through user acceptance and into the scheduled release. Team A has accumulated more points but still depends on a delayed data conversion decision.

The point totals disclose less than the dependency.

Velocity can help a team plan its next iteration; it cannot tell an executive whether the organisation is becoming more capable. Nor can the arithmetic be repaired by normalising points across teams. Standardising the scale may make the report tidier, but it turns a relative estimating aid into a disguised work-measurement system. Teams then spend time calibrating estimates for management consumption instead of using estimates to expose uncertainty.

The strongest defence of programme-level velocity is reasonable: leaders require some common measure, and an imperfect number may be better than anecdote. Without a quantitative view, weak delivery can hide behind the language of adaptation.

The answer is not to abandon measurement. It is to measure at the level of the decision being made. A programme board deciding whether a release is credible needs evidence about completed capability, unresolved dependencies, operational readiness and forecast uncertainty. It does not need the private estimating currency of six teams added together.

Replace the single signal with a small control set

The practical correction is modest. Keep velocity where it is informative, and build programme assurance from evidence that survives movement between teams.

1. Leave velocity with the team. Use the recent range, not a demanded upward trend, to decide how much work to bring into the next iteration. A falling figure should prompt inquiry into team change, uncertainty or impediments, not an automatic performance judgement.
2. Show completed capability. At programme level, report which end-to-end changes have met their agreed acceptance conditions and which are still partial. “Three of five policy changes accepted by users” is less flattering than 214 points completed, but it is much harder to misunderstand.

3. Expose the constraints. Name the decisions, environments, specialist roles and external dependencies that govern the release. Record how long each has remained unresolved and who can clear it.
4. Forecast as a range. Use the team's recent delivery pattern to present plausible release outcomes, with assumptions visible. A range invites a decision; a single date invites ritual reassurance.
5. Ask what changed in the operating position. The board should know whether the programme has reduced risk, increased usable capability or learned something that changes the plan.

This control set is not as compact as one rising line. That is its virtue. Complex programmes rarely fail because leaders lacked a simple number. They fail because the simple number concealed the decision that mattered.

The governance question behind the metric

The misuse of velocity is therefore not chiefly an agile problem. It is a governance problem. Organisations are introducing iterative methods while preserving reporting systems built to reward conformance to an approved plan. When the two meet, agile language is absorbed into the old assurance ritual.

We should resist the temptation to make every team measure legible to the board. Some measures are valuable precisely because they remain local, contextual and useful for conversation. Executive governance should not demand a larger version of the team's instrument panel. It should ask for evidence proportionate to executive decisions.

The test is simple. If velocity rises next month, what decision will the programme board make differently? If the answer is merely that the slide will remain green, the measure is not governing the programme. It is decorating the status report.