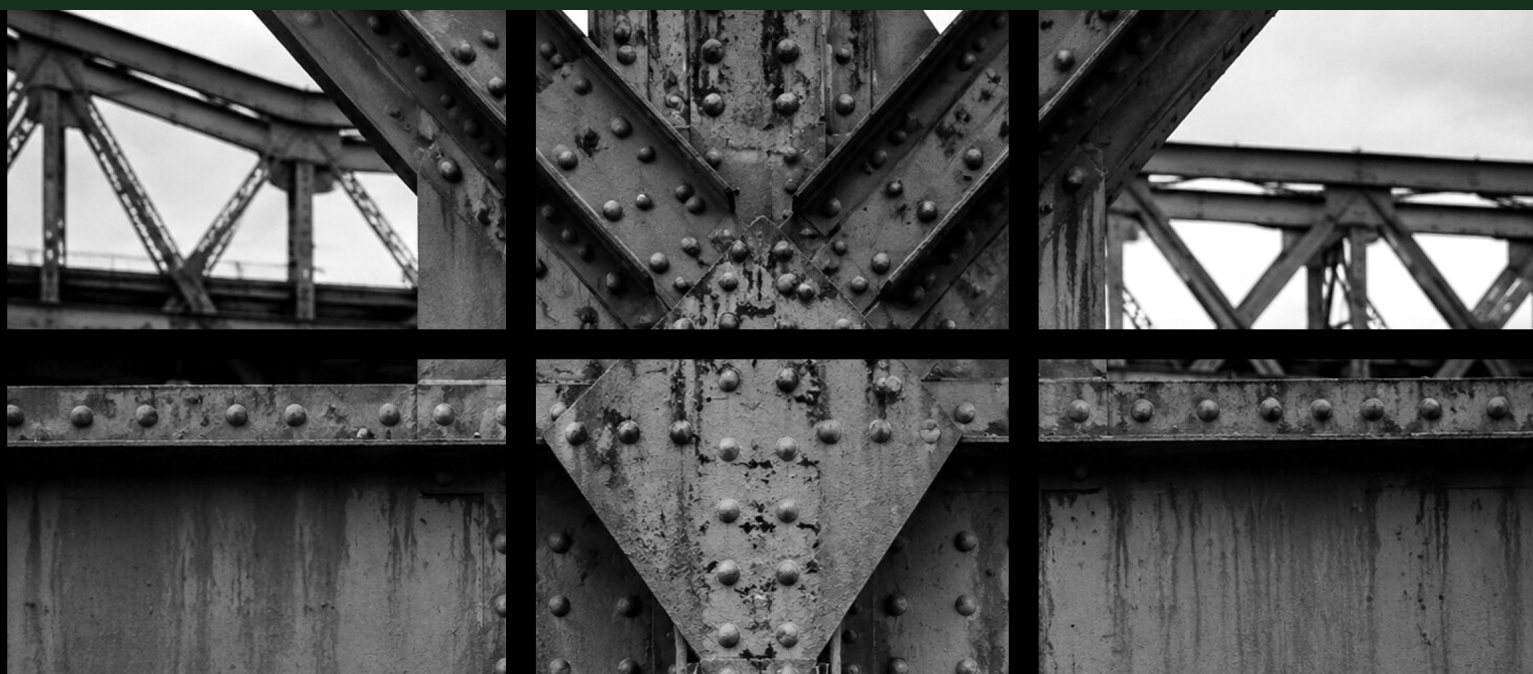


BEYOND THE WATERFALL

The Only Game in Town: Our Unexamined Faith in Sequential Delivery

A Practitioner's Perspective on Programme Delivery and the Certainty We Never Questioned



Giovanni Leonardi

April 2001

The Only Game in Town: Our Unexamined Faith in Sequential Delivery

Programme · No. 5

Originally written April 2001 | Re-edited and re-rendered for the WhereBeyond Edition, 2025

“The plan did not fail because it was badly made; it failed because it promised to know in month one what could only be learned in month nine.”

1. The Question We Were Never Allowed to Ask

On every large programme I have worked inside, the shape of the plan was decided before the first genuine question was asked. Requirements would be gathered, then a design signed off, then the build, then the test, then the go-live — each phase completing before the next began, each gate a point of no return dressed up as a point of assurance. This was not chosen after weighing alternatives. It was simply what delivery was. To propose anything else was not to lose an argument; it was to have failed to understand what the word “programme” meant.

I want to examine that certainty, because I no longer believe it is warranted, and I have come to think the confidence itself is the most dangerous thing we carry into a programme.

Let me be precise about what I am and am not saying. I am not saying that sequence is always wrong, or that planning is a vice. Some work genuinely is sequential; you cannot test what you have not built. What I am questioning is the unexamined status of it all — the way sequential, phase-gated delivery became not a choice we made for particular work but the water we swam in, invisible precisely because it was everywhere.

2. The Comfort of a Plan Is Not the Certainty of an Outcome

The appeal of the sequential model is emotional before it is rational, and we should be honest about that. A fully sequenced plan, laid out across a wall in a project office, is an object of enormous reassurance. It says: we know what we are doing, we know when it will be done, and we know what it will cost. It converts an uncertain future into a document, and a document can be approved, funded, and reported against. For a board nervous about a large commitment — and boards are more nervous this year than most — the plan is not really a delivery instrument at all. It is a comfort instrument.

The trouble is that the comfort and the certainty are not the same thing, and we routinely mistake the first for the second.

“A plan converts an uncertain future into a document. We then make the error of believing the document has removed the

uncertainty rather than merely hidden it.”

The sequential plan front-loads every important decision into the moment we understand the least. Requirements are fixed at the beginning, when the organisation has not yet seen anything real and cannot yet know what it actually needs. The design is committed before a single user has touched a working screen. By the time the programme reaches the point where the truth becomes visible — integration, testing, the first contact with reality — the cost of changing course has been deliberately engineered to be enormous. We call the resulting rigidity "control." It is closer to the opposite.

The plan did not fail because it was badly made; it failed because it promised to know in month one what could only be learned in month nine. And when the learning arrived, the plan had no way to receive it except as a change request, a variance, a problem to be managed rather than a discovery to be welcomed.

3. What Sequential Delivery Quietly Assumes

It is worth naming the assumptions the model rests on, because they are rarely stated and, once stated, are hard to defend.

- That requirements can be known fully and correctly before anything is built.
- That the organisation's needs will hold still for the duration of the programme.
- That the cheapest time to discover a problem is not earlier but on schedule.
- That the risk in a large undertaking is best managed by committing to the whole of it at the start.

Stated plainly, not one of these survives contact with experience. Requirements are discovered, not gathered; they change as the business sees what is possible. Needs do not hold still. Problems found late are ruinously more expensive than problems found early. And committing to the whole of a large undertaking at the outset does not reduce risk — it concentrates it, placing the entire bet on the accuracy of our least-informed judgement.

Every phase gate we pass congratulates us for having reduced uncertainty, when what we have actually done is postpone the moment we are allowed to notice it.

4. The Voices at the Edges

We are not the first to notice this, and it would be dishonest to pretend the alternatives do not exist. They do — they simply live at the edges of respectable practice, tolerated for small projects and quietly assumed to be unsuitable for anything that matters.

The rapid and dynamic development methods of the last decade argued for building in short, timeboxed increments, showing working software early, and letting requirements firm up through contact rather than before it. The spiral model made the case, years ago now, for treating a large undertaking as a series of risk-driven iterations rather than one long march. And the more radical programming disciplines emerging from the development community argue for tightening the loop between building and learning until it is measured in days, not quarters. These are not fringe eccentricities. They are serious

responses to exactly the failure I have been describing.

And yet, inside the large programme, they are almost always waved away with the same phrase: "that doesn't scale." I have said it myself. But I have come to suspect it is less an observation than a defence — a way of protecting the sequential model from scrutiny by asserting, without evidence, that the alternatives belong to a smaller and less serious world. The honest position is that we have barely tried. We have not failed at iterative delivery at scale; we have declined to attempt it, and then cited its absence as proof of its impossibility.

5. The View From Inside

What does this look like from within a programme, rather than from the strategy paper? It looks like a team that knew, privately and early, that the design was wrong, and who carried that knowledge for months because there was no legitimate channel through which a completed phase could be reopened. It looks like a testing period that everyone treated as a formality and that turned out to be the first time anyone had seen the whole thing work, or fail to. It looks like a business sponsor signing a requirements document they did not fully understand, because the process required a signature at that point and offered no mechanism for the understanding to arrive later.

None of these are failures of the people involved. They were, in each case, capable and conscientious. They are failures of a delivery model that asks people to commit to certainty before it is available, and then treats the arrival of the truth as an inconvenience to the plan.

*"We did not lack talented people. We lacked a legitimate way
for what they learned to change what we did."*

6. What I Would Change

I am not calling for the plan to be thrown away, and I distrust anyone who would. Large organisations need to know, in broad terms, what they are committing to and when. The discipline of planning is not the enemy.

What I would change is the status of sequence — from an unquestioned default to a deliberate choice, made work by work, and defended rather than assumed. I would break the large commitment into increments that each deliver something real and each create a legitimate moment to learn and to change course. I would treat the first contact with reality as something to engineer early and often, not to defer to a testing phase near the end. And I would measure a programme not by its fidelity to a plan written when we knew the least, but by the rate at which it converts uncertainty into working, proven capability.

The sequential model gave us the comfort of knowing, on paper, exactly where we would be. The discomfort of the iterative alternatives — the refusal to promise what cannot yet be known, the willingness to be shown wrong early — is not a weakness to be tolerated. It is the whole point. The programmes that will age well are not the ones that clung hardest to the plan. They are the ones that were willing, again and again, to let reality correct them

while correction was still cheap.