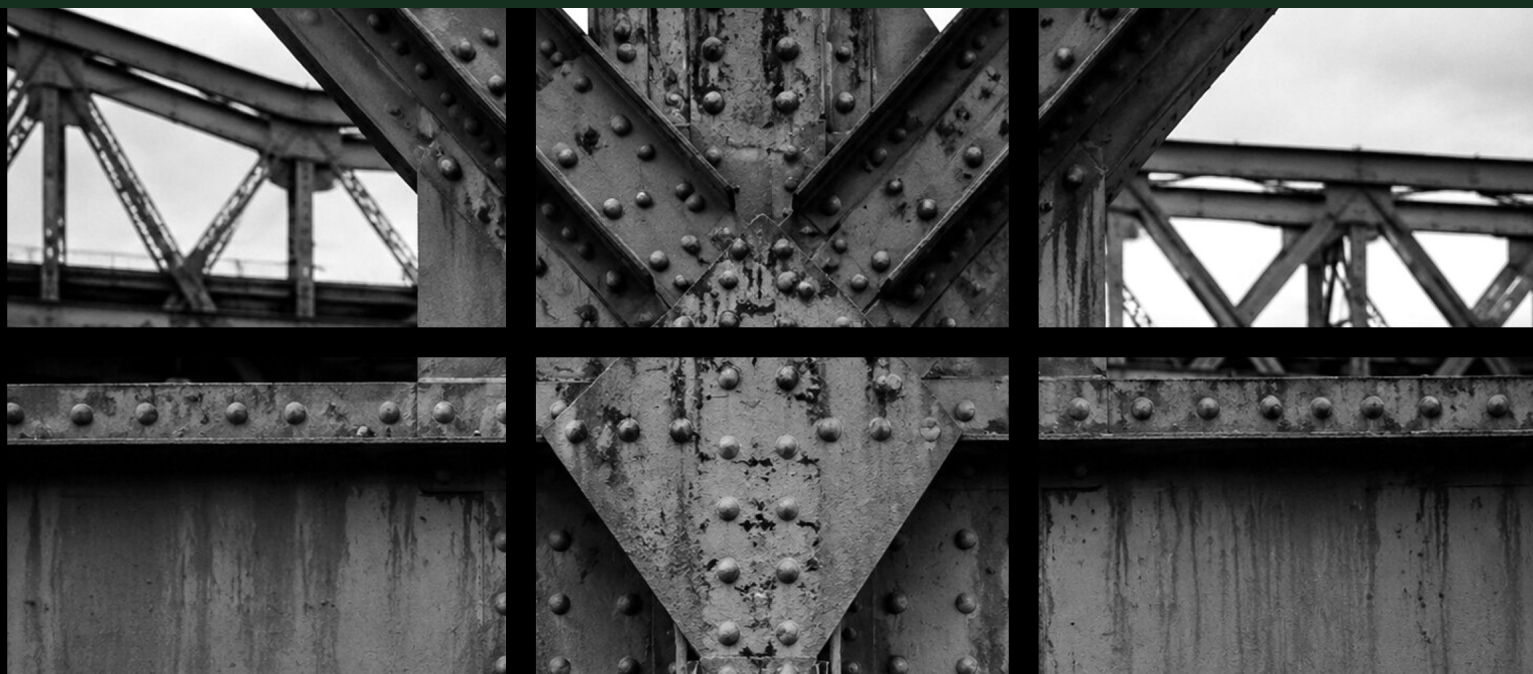


BEYOND THE PLAN

The Waterfall That Never Fell — Why Sequential Delivery Persisted Against All Evidence

A Practitioner's Perspective on the Resilience of Failed Models



Giovanni Leonardi

June 2007

The Waterfall That Never Fell — Why Sequential Delivery Persisted Against All Evidence

Programme · No. 94

Originally written June 2007 | Re-edited and re-rendered for the WhereBeyond Edition, 2025

“Waterfall persists not because it works but because it tells a story that governance structures want to hear — a story of certainty, sequence, and control.”

1. A Model Nobody Invented

The curious thing about waterfall is that nobody designed it. Winston Royce's 1970 paper — the one universally cited as the origin of the sequential model — actually argued against it. His paper presented the linear sequence of requirements, design, coding, testing, and deployment as a flawed approach, then spent the remaining pages describing how to mitigate its failures through iteration and feedback. The industry read the first diagram and ignored the rest.

Thirty-seven years later, waterfall remains the dominant delivery model for complex technology programmes in large organisations. Not because it has been refined or improved — but because the alternatives, while demonstrably more effective, are structurally incompatible with the way most organisations govern, fund, and assure their investments.

This is a commentary on why that incompatibility persists, and what it tells us about the real obstacles to improving programme delivery.

2. The Evidence Is Not in Dispute

The case against sequential delivery for complex programmes is not new and it is not subtle. The Standish Group's CHAOS reports, whatever one thinks of their methodology, have documented for over a decade that large, sequentially-delivered IT programmes fail at rates that would be unacceptable in any other domain. Overruns of 50% to 100% on cost and schedule are routine. Scope is cut to meet deadlines. Benefits are redefined after the fact to match whatever was actually delivered.

The pattern is structural, not incidental. Sequential delivery assumes that requirements can be fully understood before design begins, that design can be completed before construction starts, and that the thing built will match the thing specified months or years earlier. For simple, well-understood problems with stable requirements, this assumption holds. For the complex, uncertain, interdependent challenges that characterise most enterprise technology programmes, it fails — reliably and predictably.

This is not a controversial observation. Practitioners know it. Programme managers know it. Even the sponsors who approve the business cases know it, though they are less likely to say so. The question is not whether sequential delivery fails for complex work. The question is why we keep using it.

3. Three Structural Reasons Waterfall Survives

3.1 The Governance Lock-In

The most powerful force sustaining waterfall is the governance infrastructure that has been built around it. Stage gates, investment approvals, OGC Gateway Reviews, PRINCE2 checkpoint reports — all assume a sequential progression through defined phases. The business case is approved at the start. Requirements are baselined. Design is signed off. Each gate asks: have you completed the previous phase? May you proceed to the next?

The governance model does not merely describe the delivery approach. It enforces it. An organisation cannot adopt iterative delivery while its investment process demands fixed scope, fixed cost, and fixed timelines approved twelve months in advance.

This creates a lock-in that is extremely difficult to break. The delivery teams may understand that iterative approaches would produce better outcomes. The programme board may privately agree. But the investment approval — the mechanism by which money flows — requires a sequential narrative: we will spend this much, deliver this scope, by this date. That narrative is incompatible with the honest answer, which is: we will invest in solving this problem, learn as we go, and adjust scope and timeline based on what we discover.

No treasury function in a FTSE 100 company or government department is currently equipped to approve an investment on those terms. Until that changes, waterfall will persist — not because delivery teams choose it, but because funding processes demand it.

3.2 The Comfort of the Plan

Waterfall produces something that iterative approaches do not: a detailed plan that stretches from today to a defined end state. This plan is almost certainly wrong — the further it extends, the more fictional it becomes — but it serves a powerful psychological function. It tells senior leaders that the future is knowable, that the programme is under control, that someone has thought through every step.

The alternative — acknowledging that complex programmes cannot be planned in detail beyond a few iterations — is deeply uncomfortable for leaders whose careers have been built on the appearance of control. A programme director who presents a detailed twelve-month plan to the board is seen as competent and prepared. A programme director who presents a three-month plan with a direction of travel for the remainder is seen as uncertain, unprepared, or — worst of all — not in control.

“Waterfall persists not because it works but because it tells a story that governance structures want to hear — a story of certainty, sequence, and control.”

This is not a criticism of senior leaders. It is a recognition that organisations reward certainty and punish ambiguity, even when certainty is false and ambiguity is honest. The incentive structure favours the confident plan over the truthful one. Until organisations learn to distinguish between genuine confidence and performative confidence, the detailed-but-fictional plan will win every time.

3.3 The Skills and Structures We Have

The third reason is simply inertia. Organisations have spent decades building teams, roles, career paths, and supplier relationships around sequential delivery. Business analysts write requirements documents. Solution architects produce high-level designs. Developers code to specifications. Testers execute test scripts derived from requirements. Project managers track progress against a baselined plan.

Each of these roles is defined by its position in the sequence. Each depends on the completion of the previous stage for its inputs. The organisational structure is the waterfall — not as a conscious choice, but as an accumulated consequence of decades of hiring, training, and promoting people to fill roles that assume sequential flow.

Moving to iterative delivery does not just change the methodology. It changes what people do every day, how teams are composed, what skills are valued, and how careers progress. A business analyst who writes a complete requirements specification before development begins has a clear deliverable, a clear role, and a clear moment of completion. A business analyst embedded in an iterative team, refining requirements continuously as the team learns — that is a fundamentally different job, and many people who are good at the first are neither trained for nor interested in the second.

4. What Would Have to Change

The path away from sequential delivery is not primarily a methodology question. It is a governance, funding, and organisational design question.

4.1 Fund Problems, Not Solutions

Investment approval must shift from funding a defined solution to funding the exploration of a defined problem. This means approving a budget envelope and a problem statement, with stage-gate reviews that assess learning and value delivered — not compliance with a pre-approved specification. Some organisations are experimenting with this. None, to my knowledge, have embedded it as the standard approach for complex programmes.

4.2 Govern Learning, Not Compliance

Governance reviews should ask: what have you learned? What has changed? What are you doing differently as a result? These are harder questions than "are you on track against the plan?" but they are the questions that actually predict programme success. A programme that is learning and adapting is more likely to deliver value than one that is rigidly following a plan that was wrong from the start.

4.3 Restructure Around Flow

Teams must be organised around the delivery of value, not around phases of a sequence. This means cross-functional teams that contain all the skills needed to take a piece of

work from idea to deployment, rather than specialist functions that hand work from one group to the next. It means breaking down the walls between analysis, design, build, and test — and accepting that this will make many existing role definitions obsolete.

5. The Uncomfortable Truth

Waterfall will not be displaced by better methodologies. It will be displaced — if it is displaced at all — by changes to the governance, funding, and organisational structures that sustain it. The methodology is a symptom. The disease is an institutional preference for the appearance of control over the reality of learning.

Every large organisation knows that its complex programmes fail at unacceptable rates. Every large organisation knows that the sequential model contributes to those failures. And every large organisation continues to use it, because the alternative requires changes to governance, funding, and organisational design that nobody with the authority to make them is yet willing to champion.

The waterfall has not fallen because nobody has pulled the lever. The lever exists. It is labelled "uncertainty." And in most boardrooms, it remains untouched.