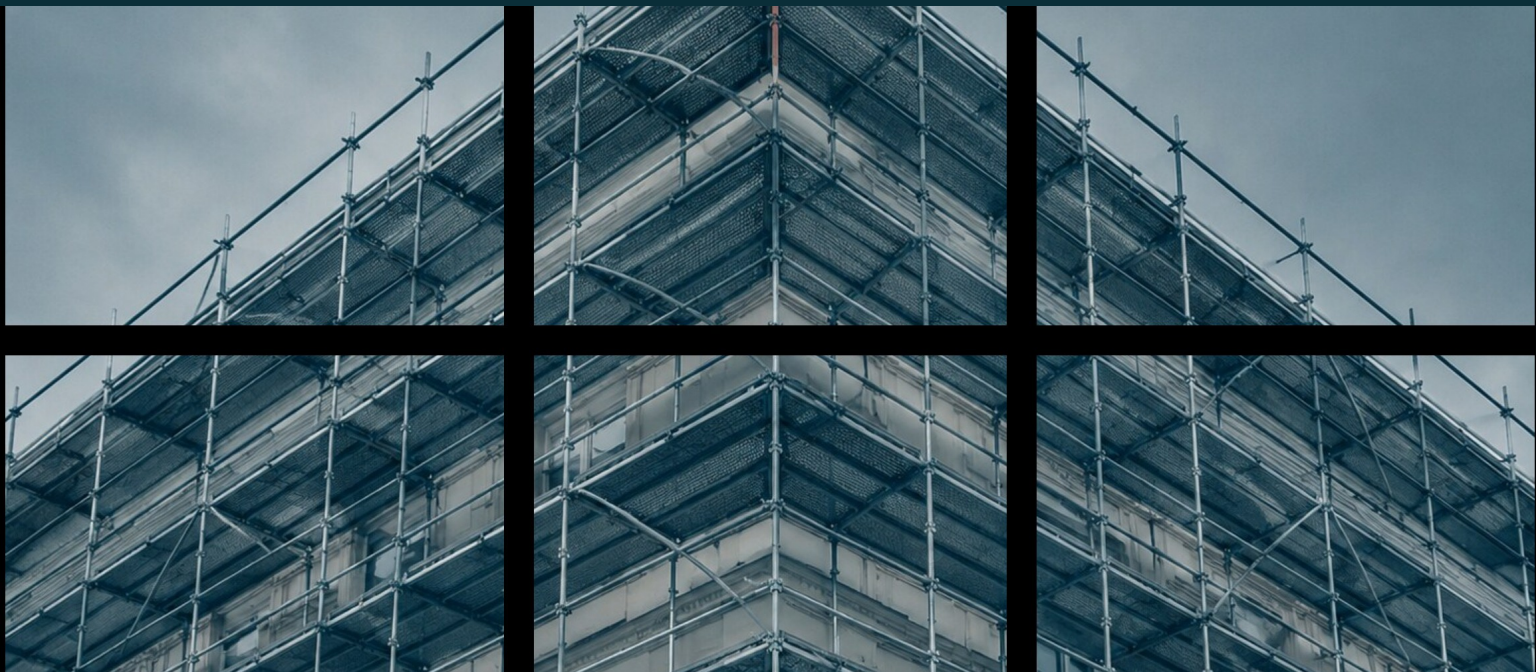


BEYOND OVERCOMING RESISTANCE

Resistance Is Data — and Most Change Programmes Throw It Away

Why the pushback a transformation works hardest to eliminate is the intelligence it most needs to keep



Giovanni Leonardi

January 2006

Resistance Is Data — and Most Change Programmes Throw It Away

Transformation · No. 58

Originally written January 2006 | Re-edited and re-rendered for the WhereBeyond Edition, 2025

“The claim is that all resistance is data, and data is to be interpreted, not obeyed.”

1. The Amber That Never Turns Red

Sit through enough programme steering committees and you learn to read the risk log the way an old hand reads a barometer. Delivery risks swing red and green with the weather. Budget risks are argued line by line. And then there is the one risk that has been amber since the programme began and will still be amber the week it goes live: "people risk" — sometimes rebadged as "adoption risk" or, more honestly, "stakeholder resistance." Its mitigation, quarter after quarter, is a variation on the same phrase: increased communications and stakeholder engagement. Nobody in the room believes the mitigation is working. Nobody proposes a different one. The amber is not treated as a problem to be solved but as a condition to be endured — the organisational equivalent of bad weather you plan around because you cannot change it.

I want to argue that this is exactly backwards, and that getting it backwards is one of the largest unforced errors in transformation work. The resistance a programme spends its energy trying to overcome is not weather. It is data — frequently the most honest, most specific, and least massaged data the whole endeavour will generate. And we throw it away, deliberately, at the very moment it is most useful.

2. Why "Overcoming Resistance" Is Such a Comfortable Story

The dominant vocabulary of change management is a vocabulary of obstacles. We talk of overcoming resistance, breaking down barriers, managing the naysayers, and building a burning platform hot enough that people have no choice but to jump. The models we lean on inherit the same posture. The guiding coalition is assembled in part to outweigh the resisters; the change curve casts objection as a stage to be moved through on the way to acceptance, as though disagreement were a phase of grief rather than a possible statement of fact. In almost every framing the person pushing back is a problem to be processed — informed, engaged, incentivised, and, if all else fails, moved aside.

It is worth asking why this framing is so durable, because its durability is not an accident. It survives because it is convenient for nearly everyone who holds power in a programme.

- For the sponsor, it converts a hard problem into an easier one. "Our design might be wrong" is a threatening sentence that reopens decisions and implicates the people who made them. "Our people are resistant to change" is a comfortable sentence that locates the fault safely in the workforce and asks only for more communication.
- For the programme itself, whose plan and business case were fixed months ago, resistance that arrives late is unwelcome news about a decision that can no longer be

reopened without cost. It is easier to label the messenger than to absorb the message.

- For the change function and the consultancies retained alongside it, resistance is the disease they are paid to cure. A profession organised around overcoming resistance has little incentive to discover that the resistance was right.

None of these actors is behaving badly. Each is responding rationally to the position they occupy. But the cumulative effect is an organisation that has quietly agreed, in advance, to treat every objection as noise — and an organisation that has decided in advance that a signal is noise will never trouble to read it.

The orthodoxy does not fail because it manages resistance badly. It fails because it pre-labels every objection as an obstacle, and so never runs the one analysis that would tell it which objections were warnings.

3. What the Pushback Is Actually Telling You

Treat resistance as data and the first thing you notice is that it is not uniform. It has a shape, a location, and a grain, and each of those carries information that no steering pack and no benefits case will ever give you, because it comes from the only people who have run your design against the reality of the work.

- Resistance concentrated in a single role or function, while the rest of the organisation shrugs and complies, is rarely temperament. It is usually a design flaw specific to that workflow — the new process has broken something that role quietly depends on, and they are the only ones close enough to feel it.
- Resistance loudest among your most experienced people should stop a programme cold. Experience is expensive to acquire and impossible to fake. When the people who know the work best are the ones most alarmed, the odds are high that the design is about to destroy a piece of tacit knowledge or an undocumented control that nobody thought to write down.
- Resistance about sequence and timing — "not like this, not now, not in this order" — is often a true statement about an operational constraint the plan has ignored. It is the cheapest possible warning that the critical path on the slide is not the critical path in the building.
- The suspicious absence of resistance is the most dangerous reading of all. A rollout that meets smooth, uniform compliance has not necessarily won the argument; more often the disagreement has simply gone underground, where it cannot be examined, to resurface at go-live as workaround, shadow process, and quiet non-adoption.

Let me make this concrete, because the argument is worthless in the abstract. Picture a finance shared-services consolidation moving a dozen business units onto a single ledger. One unit's finance team refuses to let go of a spreadsheet they run every month before close — a homemade reconciliation that checks intercompany balances between their unit and two others. The programme logs them as laggards, "emotionally attached to the old way," and schedules extra engagement. The spreadsheet is not in the target design, so it is switched off at cut-over.

Read as data rather than attitude, that spreadsheet was performing a control the new system did not replicate: a manual check that had been catching a recurring intercompany mismatch, on the order of a few hundred thousand pounds each month, and clearing it before it reached the accounts. The resistance was not nostalgia. It was the last line of

defence flagging a hole in the design. The programme forced adoption anyway, and three months later the accumulated mismatch — by then several million — surfaced in the half-year audit, along with the restatement, the auditor's hours, and the awkward question of how it had gone unnoticed. It had not gone unnoticed. Someone had noticed it every month for years, and had told the programme, in the only language available to them, that removing their spreadsheet would let it through. That warning had been the cheapest assurance the programme ever received, and it had been switched off on purpose.

4. The Objection I Take Seriously

There is a serious case against everything I have just written, and it deserves to be put at full strength rather than waved away.

It runs like this. Not all resistance is signal. A great deal of it is self-interest wearing the costume of principle — the manager defending headcount, the specialist protecting the status that comes from being the only one who understands the old system, the team that simply prefers the comfortable to the unfamiliar. Treat every objection as a verdict on your design and you will never ship anything. You will hand the programme to whoever shouts loudest and most fluently, reward the articulate over the affected, and watch necessary change die by a thousand consultations. Sometimes the burning platform is real, the discomfort is the point, and leadership means proceeding through the protest, not pausing to honour it.

Every part of that is correct except the conclusion. It is true that much resistance is self-interested, that some is mere inertia, and that a programme governed by its loudest objector is as badly run as one deaf to all objection. But none of this rescues the orthodoxy, because the claim was never that all resistance is valid. The claim is that all resistance is data, and data is to be interpreted, not obeyed. Reading a signal is not the same as surrendering to it.

The discipline the orthodoxy skips is triage: separating the resistance that encodes a design flaw, a missing control, or a real operational constraint from the resistance that encodes fear, habit, or lost status. That triage is difficult and it is the actual work. But you can only do it if you first stop treating every objection as noise by default — and the "overcome resistance" posture never gets there, because it has already filed the whole category under obstacle. It does not lose the triage. It never opens the case.

And even the resistance that turns out to be pure self-interest is still telling you something you need to know. It marks, with precision, exactly where the programme is redistributing power, whose standing it is quietly demoting, and who has been given no honest reason to cooperate. That is not a verdict on your design, but it is a live map of your sequencing and coalition risk — which is to say, it is data too. The self-interested objector is not always right about the future. They are almost always right about where the pain lands.

“Reading resistance is not the same as capitulating to it; the point is to run the triage the orthodoxy skips, not to hand the programme to whoever protests loudest.”

5. Running the Change as If the No's Were Assets

If resistance is data, the practical consequences are not soft ones. They change how the programme is actually run.

The most immediate is to stop keeping a resistance log and start keeping a resistance register — governed like the risk register it actually is. Every substantive objection gets recorded not as a sentiment to be managed down but as a claim to be tested: what is this person asserting about the design, and is it true? Some claims fail the test and can be set aside with a clear conscience. Some pass, and each one that passes is a defect caught before it reaches production, at a fraction of the cost of finding it afterwards. A programme that disposes its objections this way is doing assurance; a programme that merely counts and communicates at them is doing public relations.

The second consequence is a change in who you listen hardest to. The instinct is to invest engagement in the enthusiasts, because they are pleasant to be around and make the dashboard look green. But the enthusiast, by definition, has already stopped testing your design. It is the credible, specific, experienced objector who is still running it against reality on your behalf — and doing so for free. Treated as an obstacle, that person is worked around or worn down, and the information they hold is lost. Treated as an asset, they are the cheapest quality function the programme has.

The people saying no in a well-run change are frequently doing your assurance for nothing. Overcome them, and you are not removing a risk — you are switching off a detector.

None of this is an argument for programmes that never proceed until every voice is content; those fail too, and they fail slowly and expensively. It is an argument for a different reflex. When the pushback comes — and in any transformation worth the name it will — the first question should not be how do we overcome this? but what does this know that we don't? Most of the time the honest answer will send you back to the design with something you genuinely needed and could not otherwise have found. That is not friction in the machine. It is the machine telling you the truth, at the one moment you can still afford to hear it.