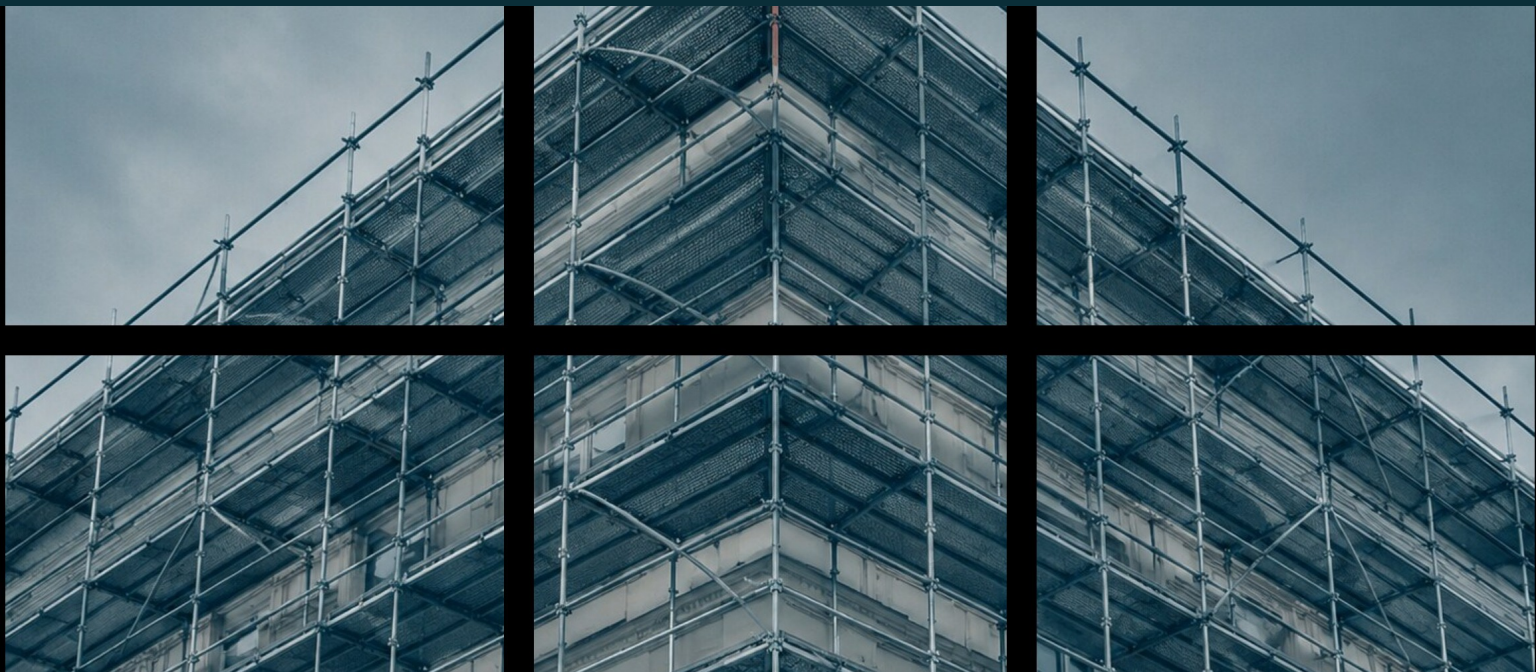


BEYOND MANAGING RESISTANCE

# Designing Past the People Who Know the Work

Why operational transformation keeps designing around the people whose knowledge would make it work



Giovanni Leonardi

May 2007

# Designing Past the People Who Know the Work

Transformation · No. 108

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

---

*“We are, most of us, fluent in method. We are far less fluent in humility — in the discipline of assuming that the people closest to the work know something we do not.”*

---

## 1. The room where the work was not

---

There is a room I have walked into more times than I can count. It has whiteboards down one wall and a large printed process map — the to-be state, laminated, colour-coded, admired — pinned to the other. Around the table sit the programme director, two consultants, a business analyst, someone from IT, someone from finance, and a change manager whose job is to prepare the organisation for what this room decides. The design is nearly finished. It is elegant. It has been signed off by a steering group. And three floors down, or in a service centre two hundred miles away, are the several hundred people who will actually have to do the work this map describes — none of whom has been in the room.

This is the quiet scandal at the centre of most transformation, and after years of watching it recur I have stopped calling it an oversight. An oversight is an accident. This is a pattern too consistent, too structurally reinforced, and too easily predicted to be accidental. We exclude the people who do the work not because we forget them, but because the machinery of a transformation programme is built in a way that pushes them out. And that exclusion, more than technology risk, more than scope, more than any of the things our risk logs obsess over, is the single most reliable predictor that the benefits case will not be met.

*The frontline is not consulted late because we ran out of time. It is consulted late because everything about how a programme is structured makes late the natural moment — and by then, consultation is theatre.*

## 2. Why the work gets left out of its own redesign

---

If this were merely a matter of good intentions, it would have been solved long ago. Every methodology tells us to engage stakeholders. Every change manager knows the phrase. The reason it persists is that the incentives of a programme run the other way.

- The programme is rewarded for pace, and involvement reads as delay. A steering group measures a programme against its plan. Nothing on that plan says understood the work as it is actually done. The milestones are design sign-off, build, test, go-live. Bringing forty people from the floor into the design does not advance a milestone; it threatens one. So the rational programme director defers it — not out of contempt, but because the reward system is blind to it.

- Design happens in the language of the designers. The to-be map is written in the vocabulary of process, system, and target operating model. The work, as it is actually done, is written in a different language — of workarounds, of the customer who always calls back, of the step that the manual says takes two minutes and reliably takes twenty. These two languages do not meet, and the room only speaks one of them.
- The people who do the work are treated as a risk to be managed, not a source to be mined. Notice the vocabulary. We speak of managing resistance, of the impact on the business, of readiness. Every one of these frames the frontline as an obstacle standing between the design and its benefits. When your entire language casts a group as the thing to be overcome, you do not think to ask them how the work should be built.
- Hierarchy makes the knowledge invisible. The person who knows most about how an order is actually processed is very often the person with least standing to be heard. Their knowledge is tacit, undocumented, and low-status. The programme, staffed by people of high standing, mistakes the absence of documentation for the absence of knowledge.

None of these forces is a failure of character. That is exactly why exhortation does not fix them. You can tell a programme director a hundred times to engage the frontline earlier, and the plan, the steering group, and the language of the room will quietly undo the instruction every time.

### 3. What it costs, and where it shows up

The cost is not abstract, and it does not appear where we look for it. It appears after go-live, disguised as something else.

Consider a composite that will be familiar to anyone who has lived through a large operational change. A shared-services consolidation is designed to take a fragmented back-office function — say, the processing of supplier invoices spread across a dozen sites — and pull it into a single centre running a single system and a single process. The business case is clean: a headcount reduction of thirty per cent, a cost-per-invoice falling from a benchmarked figure of, say, £4.20 to a target of £2.80, payback inside two years. The to-be process is designed, correctly, for the standard invoice — the one that arrives matched to a purchase order, in the expected format, requiring no judgement.

The trouble is that the standard invoice was never the problem. The people who did the work knew — and could have said, had anyone asked — that something like a third of invoices are not standard. They arrive without a purchase order, or against a contract with unusual terms, or from a supplier whose account was set up years ago in a way no system since has quite understood. In the old fragmented world, these were handled by a handful of long-serving clerks who simply knew what to do, because they had done it for fifteen years. The new design has no place for that knowledge. The exceptions do not disappear; they queue. Within three months the centre is drowning in a backlog of the awkward third, suppliers are calling because they have not been paid, and the £2.80 target has quietly become £3.90 as temporary staff are drafted in to clear the queue.

None of this shows up as a design failure. It shows up as a transition issue, or a data quality problem, or lower-than-expected process maturity — anything but the truth, which is that the design was built without the knowledge that would have made it work, and that knowledge was sitting, unconsulted, in the heads of the people the programme was busy making redundant.

---

*“The exception you did not design for does not vanish. It waits for you on the far side of go-live, and it brings the benefits case down with it.”*

---

#### 4. The objection worth taking seriously

---

There is a serious case against everything I have just argued, and it deserves to be met at its strongest rather than waved away.

It runs like this. You cannot design a process by committee. The frontline sees its own patch and not the whole; ask forty people how the work should be done and you will get forty incompatible answers, each optimised for a local convenience that the enterprise cannot afford. Worse, involvement raises expectations — the moment you ask people to shape the design, they believe they have a veto, and when the design does not match their preference they feel betrayed, and you have manufactured the very resistance you hoped to avoid. Transformation, on this view, sometimes requires imposing a discipline that no one close to the current mess would ever have chosen. Design needs a vantage point above the work, not inside it.

I take this seriously because parts of it are true. The frontline does see its own patch. Involvement can curdle into a sense of ownership that makes the necessary imposition harder. And a process assembled from local preferences would indeed be a monster.

But the objection quietly assumes that there are only two options: design above the work in ignorance of it, or design by the work and be captured by it. That is a false choice, and it is the false choice that does the damage.

#### 5. Extraction, not consultation

---

The distinction I have come to rely on is between consulting the frontline and extracting from it — and it is the whole game.

Consultation asks people what they want. That is where the objection above lands its blows: it invites preference, it implies a vote, and preferences do not aggregate into a process. Extraction asks something entirely different. It asks: show me how the work is actually done — not how the manual says, but what really happens when the awkward invoice arrives. It treats the frontline not as a constituency to be satisfied but as the only reliable source of a specific and irreplaceable kind of knowledge: what the work is really like. The designer still designs. The vantage point above the work is preserved. But it is now informed by the texture of the work rather than by a laminated fiction of it.

In practice this is not a grand act of participation. It is small and unglamorous.

1. Map the work before you map the process. Spend real time — days, not an afternoon workshop — watching the awkward cases being handled by the people who handle them. The deliverable is not a wish-list; it is an honest inventory of the exceptions, the workarounds, and the tacit rules that keep the current mess functioning.

2. Name the exceptions and design for them explicitly. The standard case will look after itself. The design stands or falls on whether the awkward third has a home. Every exception surfaced in step one gets an owner in the to-be world, or a deliberate, eyes-open decision to stop handling it.
3. Keep the knowledge before you remove the people. Where a change reduces headcount, the tacit knowledge in those heads is an asset the programme is about to destroy. Capturing it – through the people themselves, while they are still there and while they can be treated with enough respect to want to help – is not sentiment. It is the difference between a design that survives contact with reality and one that does not.

Notice that none of this hands the frontline a veto. It hands the designer something they did not have: an accurate picture of the work. That is the reconciliation the objection missed. We are not choosing between authority and involvement. We are choosing between designing with our eyes open and designing with them shut.

## 6. The honest account

---

So here is the honest account, the one the programme documents never quite give.

We do not exclude the people who do the work because we underestimate them as individuals. We exclude them because our plans, our governance, our language, and our hierarchies all conspire to make their knowledge invisible at precisely the moment it is most needed – and then we are surprised, on the far side of go-live, when the benefits we promised evaporate into backlogs and temporary staff and transition issues that were never transitional at all.

The organisations that break this pattern are not the ones with the best change-management slide decks. They are the ones that have understood something almost embarrassingly simple: that the work is a form of expertise, that this expertise lives in the people who do it, and that no amount of design elegance in a room three floors up can substitute for it. That understanding cannot be delegated to a change workstream. It has to be held by the people who run the programme, because it is a decision about where the programme looks for the truth.

We are, most of us, fluent in method. We are far less fluent in humility – in the discipline of assuming that the people closest to the work know something we do not. Until that changes, we will keep building beautiful maps of rooms that no one who does the work would recognise, and we will keep calling the result a surprise.