The Hidden Cost of Knowledge Reconstruction

Neal McCormick

Semantic Foundry

There is a line item in almost every transformation budget that nobody talks about honestly.

It is not labelled discovery. It is not labelled knowledge reconstruction. It appears as workshops, consulting engagement fees, architecture reviews, requirements gathering, stakeholder interviews, current-state analysis. It looks like the necessary first phase of doing serious work.

What it actually is, in almost every case, is the organisation paying to understand itself.

Again.

—

The previous piece in this series argued that organisations cannot read themselves — that the knowledge of how they actually operate is distributed across people, processes, and systems in a form that nobody can fully assemble into a coherent picture.

This piece is about what that costs.

Not in abstract terms. In budget terms. In the kind of numbers that appear on programme business cases and get approved by CFOs who have no idea they have approved the same numbers before.

—

Consider a typical large-scale modernisation programme in a regulated industry. Financial services. Insurance. Healthcare. A legacy platform that needs replacing, or a strategic capability that needs building, or a regulatory requirement that needs implementing.

The programme is approved. The budget is committed. The team is assembled.

For the first six to nine months, almost nothing gets built.

Subject matter experts are interviewed. Existing documentation is reviewed. Systems are analysed. Conflicting accounts of how the business actually works are reconciled. Business rules that exist in three places and say different things are traced back to their origins. The people who understand particular domains are identified, calendared, and asked to explain what they know to people who will need to understand it in order to build something new.

This is not delivery. This is archaeology.

And it consumes, on the programmes where anyone has bothered to measure it, a significant portion of the total programme budget. Before a single capability changes. Before a line of new code is written. Before anything the programme was approved to deliver has been delivered.

—

Here is the part that should give every CFO pause.

That knowledge — the knowledge being reconstructed at such expense — already existed in the organisation.

The business rules that are being traced back to their origins were encoded in a system that the previous programme built. The business process that is being mapped was documented when the system before that was implemented. The subject matter experts who are being interviewed learned what they know from the people who were interviewed in the programme before this one.

The organisation has already paid to produce this understanding. Possibly several times. On previous programmes, with previous teams, with previous consultancies who charged day rates to build a picture of how the organisation works and then took that picture with them when the engagement ended.

Most transformation budgets are paying to reconstruct knowledge the organisation already possesses.

That is the money shot. That is the number that should appear on a line of its own in every programme business case, with a note explaining how many times the organisation has paid it before.

—

The cost compounds in ways that go beyond the direct budget.

The subject matter experts who are being interviewed during the discovery phase are not doing their actual jobs during those interviews. The people with the deepest domain knowledge are the most operationally essential — which means they are also the most expensive to pull into workshops, the most resistant to having their calendars consumed by a programme team, and the least likely to have the time to explain things as thoroughly as the programme team needs. The knowledge transfer that happens in these sessions is partial by design. You get what there was time to ask, not what there was knowledge to give.

The artefacts produced by the discovery phase — the process maps, the requirements documents, the architecture assessments — begin aging the moment they are written. By the time the programme is halfway through delivery, some of what was documented in the discovery phase is already out of date. Nobody maintains it because maintaining it is not the programme's job. Nobody updates it because everyone assumes someone else will. The next programme will have to go through the same process, starting from artefacts that are partially accurate and partially misleading.

And then there is the cost that is hardest to measure but most significant in the long run. The things that were not captured.

The discovery phase, however thorough, recovers what the team knew to ask about. It does not recover what nobody thought to ask. The edge case that the most experienced operator handles differently from everyone else. The business rule that applies only in a specific set of circumstances that the programme team never encountered. The dependency that exists between two systems that nobody thought to mention because everyone assumed everyone else knew about it.

Those gaps show up in production. They become incidents. They become rework cycles. They become the reason the programme that was supposed to close in eighteen months is still running at month twenty-four, addressing problems that a more complete discovery would have surfaced before the first line of code was written.

—

None of this is a secret. Everyone who has run a large programme knows this is how it works. The discovery phase is budgeted for because it is expected. The overruns are managed because they are anticipated. The knowledge gaps show up in production because they always do.

What is rarely asked is whether this has to be the case.

The cost of knowledge reconstruction is not the cost of doing transformation. It is the cost of not having preserved what previous transformations produced. It is a tax on the absence of something — a governed, persistent account of what the organisation actually does — that would make the reconstruction unnecessary.

The question is not how to make the discovery phase cheaper. The question is why the organisation needs a discovery phase at all.

That question has an answer.

The next piece is about why every modernisation programme starts with discovery — and why that has come to be treated as normal when it is anything but.

Neal McCormick is the author of The Last Great Dig.