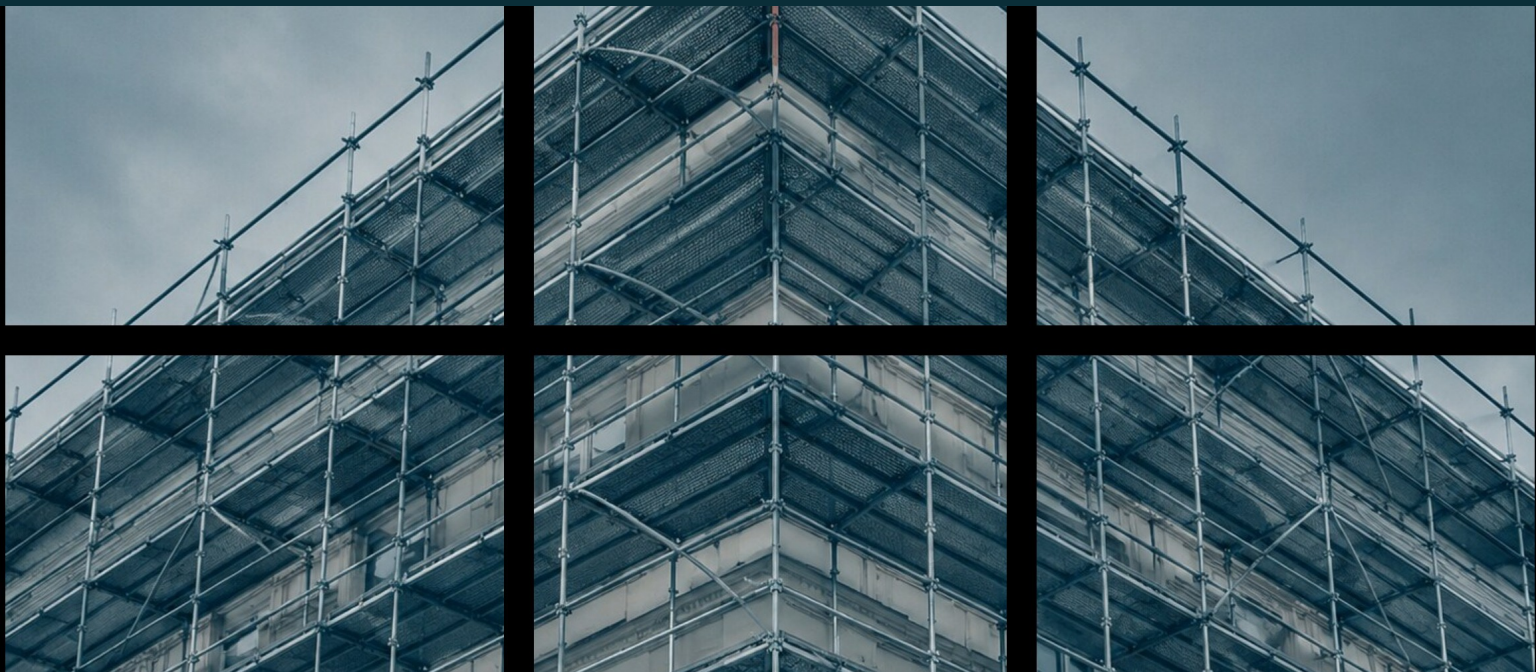


BEYOND HANDOVER

The Offshore Team Delivered Code, Not Understanding

In the 2004-05 outsourcing wave, the real gap was not distance but interpretation



The Offshore Team Delivered Code, Not Understanding

Transformation · No. 49

Originally written April 2005 | Re-edited and re-rendered for the WhereBeyond Edition, 2025

“The work was offshore, but the ambiguity remained stubbornly onshore.”

The Hand-Off That Looked Complete

The weekly pack said the offshore team had delivered everything assigned to it. Twenty-seven change requests closed. Four interface amendments completed. Defect turnaround down from six days to three. On paper, the new arrangement was working exactly as the business case had promised: lower unit cost, more development capacity, longer working day, and a visible queue of work moving steadily through the machine.

Then the acceptance meeting began.

The sales operations manager did not dispute the code. The screens opened. The overnight batch ran. The report produced numbers. What she challenged was the meaning. A field labelled customer type had been interpreted as the billing category rather than the selling channel. A rule described as "apply discount after approval" had been implemented as a system permission rather than a commercial judgement. The offshore team had built what the document said. The domestic team had meant what the document assumed.

That is the uncomfortable lesson emerging from the current outsourcing wave. The visible risk is distance. The more important risk is interpretation.

The Problem Was Never Simply Geography

Much of the 2004-05 conversation about offshore delivery is still framed around communication mechanics: time zones, accents, travel budgets, video links, specification quality, and service-level reporting. These matter. A poor handover across five and a half hours of time difference is still a poor handover. A design question left unanswered on Wednesday can easily become three idle days by Monday.

But the deeper pattern is more awkward. Many organisations are discovering that offshore delivery does not create ambiguity; it exposes ambiguity that was previously absorbed by proximity.

When analysts, developers, testers, and business users sit in the same building, a great deal of interpretation happens informally. Someone overhears a constraint. A developer walks to a business desk. A tester knows that a phrase in the requirements document should not be read literally because the operations team has always used it differently. The organisation calls this speed. Often it is undocumented translation.

Move the development work to Bangalore, Pune, Krakow, or Manila, and the translation layer becomes visible by disappearing. The offshore team asks for precision. The onshore team sends documents. Both sides behave rationally, and still the result disappoints.

The work was offshore, but the ambiguity remained stubbornly onshore.

This is why some outsourcing arrangements feel paradoxical in their first year. Productivity measures improve while business confidence falls. Code volumes rise while rework remains stubborn. The cost per development day drops, but the cost of clarifying what should have been meant by the original request quietly increases.

What the Better Programmes Are Learning

The strongest operators are not treating this as a cultural problem in the lazy sense of the phrase. They are treating it as a design problem in the delivery model.

The weak model says: write better specifications, send them offshore, manage the supplier harder. The better model asks: where is interpretation happening, who owns it, and how is it tested before code is produced?

A practical example is the difference between a signed-off requirements pack and a working interpretation pack. In one composite finance transformation, an onshore team sent a 42-page change specification to its offshore partner. The supplier estimated 310 development days. After three clarification calls, the estimate barely moved. After a structured interpretation session, however, nine assumptions surfaced that had never been written down: two about exception handling, three about product hierarchy, one about month-end timing, and three about user authority. The estimate rose to 345 days, which caused some irritation. The first release then avoided the familiar late-stage churn in which apparently small defects turn out to be disagreements about business meaning.

That rise in estimate was not waste. It was hidden work becoming visible early enough to govern.

For this reason, the next level of offshore maturity will not be won by procurement alone. Rate cards have had their day in the sun. The sharper question is whether the operating model contains the disciplines that distance demands:

- A named onshore interpreter who is accountable for business meaning, not merely document transmission.
- Requirement walkthroughs that test understanding through examples, exceptions, and sample data before estimation is frozen.
- Acceptance criteria written in business language as well as system language.
- A deliberate overlap window in the working day for questions that cannot wait for the next formal call.
- Measures that track clarification cycles and rework causes, not only delivery volume and defect counts.

None of these disciplines is exotic. That is precisely the point. The failure is rarely that organisations do not know what to do. It is that they try to buy distributed delivery without redesigning the interpretive work that made local delivery function.

The Serious Objection

There is a fair objection to this argument. Offshore suppliers can and should improve their domain knowledge. If a supplier takes responsibility for application maintenance or recurring change, it cannot remain forever dependent on the customer to explain every term of art. Capability matters, and some delivery centres are investing heavily in sector training, business analysis, and client-specific knowledge bases.

That objection is right, but incomplete. Domain knowledge cannot compensate for an organisation that has never made its own decisions explicit. A supplier can learn the vocabulary of insurance, retail banking, manufacturing, or telecoms. It cannot reliably infer which local exception the finance director still expects to survive, which sales rule is political rather than logical, or which manual workaround everyone has stopped mentioning because it is too familiar to notice.

The better ambition, then, is not to make offshore teams behave as though they are in the corridor. It is to stop relying on the corridor as a substitute for clarity.

What Leaders Should Notice Now

The current enthusiasm for offshore delivery is understandable. Budgets are tight, technology estates are expanding, and boards are asking why software work cannot be industrialised in the same way as other repeatable services. There is real value here. The mistake is to think that the value comes from moving work alone.

Offshore delivery works best when leaders distinguish between tasks that can be transmitted and judgement that must be shared. Coding a defined change can travel. Clarifying a contested business rule cannot simply be shipped with the same confidence. Testing whether a document has been understood is management work, not administrative overhead.

The organisations that learn this early will still pursue labour arbitrage, but they will not confuse it with transformation. They will invest in the roles, ceremonies, and artefacts that make meaning portable. Those that do not will keep receiving what they asked for and wondering why it is not what they needed.