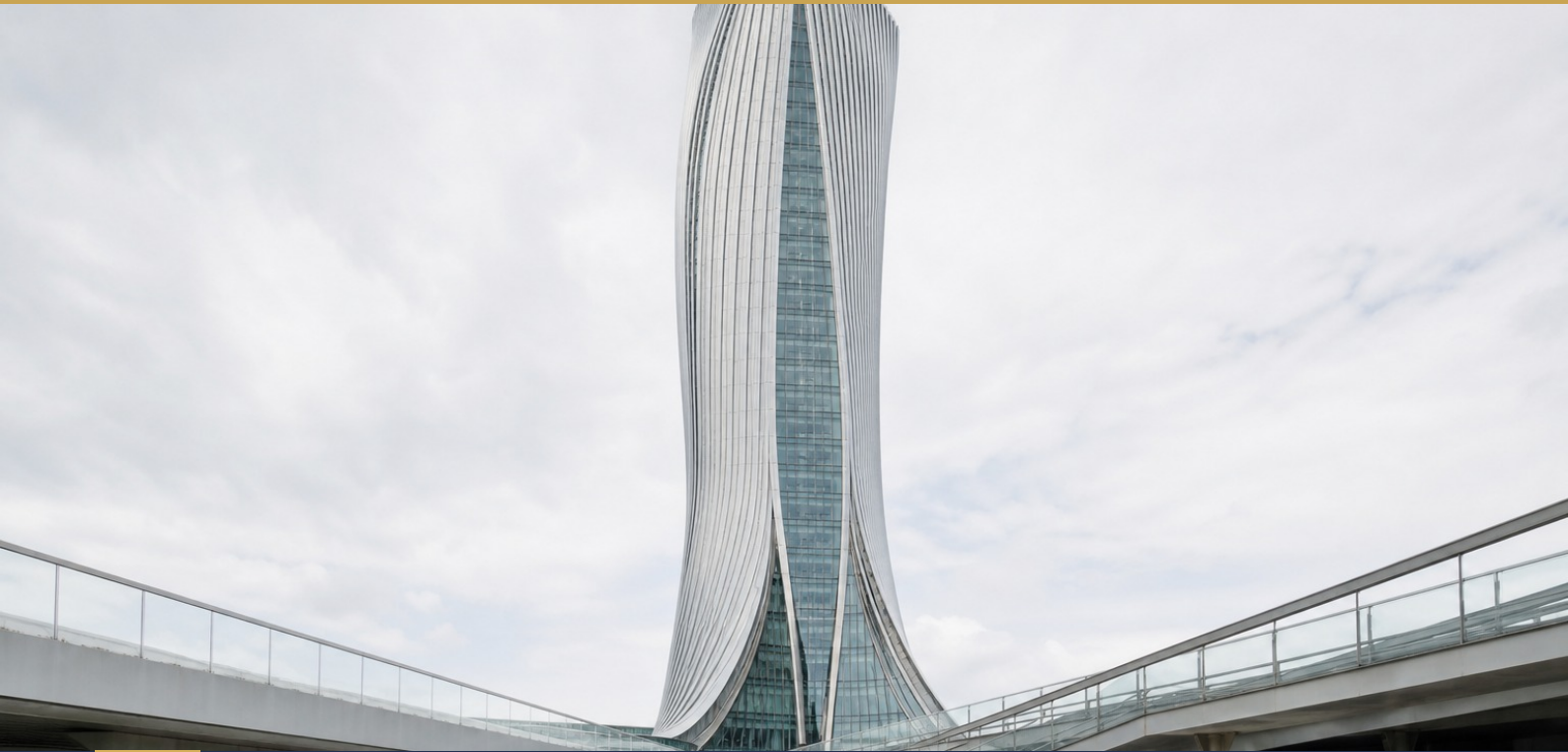




BEYOND THE HIRE

The Data Scientists Have Arrived — Now What Do We Do With Them?

The Missing Operating Model for Enterprise Data Science

The Data Scientists Have Arrived — Now What Do We Do With Them?

Leadership · No. 75

Originally written May 2017 | Re-edited and re-rendered for the WhereBeyond Edition, 2025

"A data scientist without a business owner who will stake a decision on the model's output is an academic in a corporate building."

1. The Notebook-to-Production Gap

The hiring drive worked. Across financial services, insurance, telecommunications, and retail, data science teams have grown from a handful of curious analysts to dedicated departments in the space of three years. The job postings attracted exactly the right people — PhDs in statistics and machine learning, experienced researchers, talented graduates from the wave of new analytics programmes. By any measure of recruitment, the strategy succeeded.

And yet, in a growing number of these organisations, a quiet pattern is emerging. The data scientists are building models that never leave the laboratory. The prototypes are impressive — demonstrated in PowerPoint, praised by the executive sponsor, filed away. The models that do reach anything resembling production are maintained through heroic manual effort: a scheduled script running on a data scientist's laptop, a spreadsheet refreshed weekly, a one-off analysis that becomes permanent by accident. The gap between what these teams can build in a notebook and what the organisation can actually use in a decision is, in most enterprises, enormous — and it is widening.

The conventional diagnosis is that the organisation needs better data scientists, or more of them. This is precisely wrong. The constraint was never talent. It is the absence of an operating model — the infrastructure, the engineering discipline, the organisational structure, and the decision rights — that could take a model from a Jupyter notebook to a system that reliably informs or automates a business decision.

2. The Talent Illusion

The phrase "data scientist" entered the executive vocabulary with the force of a mandate. Since the Harvard Business Review declared it the sexiest job of the twenty-first century, boards and leadership teams have treated the hiring of data scientists as the act that creates data science capability. The logic runs: we have a data problem; data scientists solve data problems; therefore, hire data scientists.

What this logic misses is every precondition that makes data science productive. A data scientist without reliable access to clean, governed, well-documented data is an analyst who happens to know Python. A data scientist without a path to deploy a model into a production system is a researcher who produces proofs of concept. A data scientist without a business owner who will stake a decision on the model's output is an academic in a corporate building.

We have, across much of the enterprise landscape, created the conditions for frustration rather than the conditions for value. The data scientists know this — attrition in enterprise data science teams is already becoming a leading indicator, and exit interviews tell a consistent story. The talent is not leaving because the problems are uninteresting. They are leaving because the organisation cannot translate their work into anything that matters.

3. What "Model to Production" Actually Requires

The gap is not mysterious. It is a set of specific, practical, largely engineering problems that most organisations have not yet recognised as their responsibility. In my experience, the organisations that have begun to close it have converged on a similar set of requirements, though few have named them formally.

Data engineering as a distinct discipline. The most consequential hire most data science teams have not yet made is not another data scientist — it is a data engineer. The work of building and maintaining the pipelines that move data from source systems into a form that models can consume reliably, repeatedly, and at the right latency is a full engineering discipline. It is not a task that data scientists should be doing, and it is not a task that traditional database administrators are equipped for. It sits somewhere between software engineering and infrastructure, and most organisations have no job description for it. The result is predictable: data scientists spend the majority of their time — seventy or eighty per cent is the figure I hear most often — finding, cleaning, joining, and reshaping data, leaving a residual fraction for the work they were hired to do.

An engineering path from model to system. A trained model is an artefact — a file, a set of coefficients, a fitted function. Moving it from a data scientist's notebook into a system that serves predictions reliably, at the required speed, with appropriate error handling, logging, and fallback behaviour, is a software engineering problem. It requires someone to write the serving layer, to package the model, to integrate it with the consuming application, to think about latency and throughput. In most organisations today, this work either does not happen at all — the model stays in the notebook — or it happens through a painful, ad hoc handoff between the data science team and an application development team that has never seen a statistical model before.

The organisations making progress here have recognised that this is not a one-off integration task but a repeatable engineering function. They are building small platform teams — sometimes just two or three engineers — whose job is to provide a standard way to take a model from development to deployment: a common serving framework, a consistent way to version and package models, a deployment pipeline that a data scientist can use without becoming a software engineer. This is not glamorous work. It

is infrastructure. And it is the single most impactful investment most data science programmes are not making.

Business ownership of models. A model without a business owner is an orphan. Someone in the business — not in the data science team, not in IT — must own the decision that the model informs. They must be accountable for whether the model is used, how its outputs are interpreted, what happens when it is wrong, and when it should be retrained or retired. Without this ownership, models drift into irrelevance. The data science team builds a churn propensity model; the marketing team ignores it because they do not trust it, or because it arrives in a format they cannot use, or because nobody has the authority to change the campaign process based on its output.

This is a governance problem as much as a technical one. The organisations that have solved it tend to have created explicit roles — a model sponsor, a decision owner — and explicit processes for agreeing, before a model is built, what decision it will inform, what action will change, and who is accountable for the outcome.

4. The Operating Model Nobody Designed

What these requirements amount to, taken together, is an operating model for data science — and it is the thing that almost no organisation designed before it started hiring. The leadership assumption, in most cases, was that data science would work like other specialist functions: hire the experts, give them a mandate, let them produce. What distinguishes data science from most specialist functions is that its output is not a report or an analysis but an artefact that must be embedded in a system and sustained over time. It is closer to software engineering than to consulting, but it has been organised as though it were consulting.

The result is a structural mismatch. The data science team sits in a part of the organisation — often reporting to a Chief Data Officer, sometimes to the CTO, occasionally to a business unit — that has no engineering capacity of its own and no direct authority over the systems into which its models must be integrated. Every deployment becomes a negotiation, a project, a request into another team's backlog. The feedback loop between a model's performance in production and the data scientist who built it is either non-existent or painfully slow. The team cannot iterate because it cannot observe.

The constraint on enterprise data science is not talent, tools, or algorithms. It is the absence of the engineering discipline, the data infrastructure, and the decision rights that connect a model to a business outcome. Until leadership treats this as an operating-model problem rather than a hiring problem, data science teams will continue to produce impressive work that never leaves the laboratory.

5. What Leadership Owes the Investment

The corrective is not complicated, but it requires leadership to do something harder than signing hiring requisitions: to design the operating model that makes the talent productive.

This means investing in data engineering before — or at least alongside — data science. It means building or commissioning the platform capability that provides a repeatable path from model to production. It means establishing business ownership of models, with named sponsors and agreed decision points, before the first line of code is written. And it means measuring the data science function not by the number of models built or proofs of concept demonstrated, but by the number of business decisions that are demonstrably better because a model is in production and being used.

We are, as an industry, roughly three years into the enterprise data science experiment. The early returns are mixed not because the science is wrong but because we treated hiring as the hard part, when hiring was in fact the easiest step in a much longer chain. The data scientists have arrived. The question now — the one that will determine whether the investment pays off or quietly dissipates — is whether leadership will build the machinery around them that lets them do what they came to do.