

From Data Pipelines to Decisions: The Missing Link Is Trust

Why the consumption layer and not the warehouse is where trust is won and lost.

There is a version of the modern-data-stack conversation that goes like this:

"We have the warehouse, the catalog, the lineage, the tests, the data contracts, so we are in good shape."

It sounds compelling.

It also doesn't match what the CFO sees at 9:47 on a Tuesday morning, when two directors walk into her office holding two dashboards that disagree about last quarter's margin by eight hundred basis points.

The failure mode is not the warehouse. The warehouse is fine. The failure mode is that trust doesn't live in the warehouse. Trust lives in the report the executive is actually looking at when she has to make a call, and that report sits in Power BI, or Tableau, or Looker, or Qlik, four layers downstream of every beautifully-governed dbt model you've built.

And as AI assistants move inside every BI tool (Copilot in Power BI, Pulse and agents in Tableau, the "ask your data" box in Looker and Qlik), that unmanaged layer is starting to talk. In natural language. With confidence and at scale.

The modern data stack has been built for the people who build the stack. The layer where trust is won or lost has been left largely unmanaged.

The Trust Problem Nobody Wants to Name

Go look at any Fortune 500 BI estate honestly, and you will find the same picture: tens of thousands of reports, built by hundreds of authors, on top of dozens of overlapping datasets.

Across our customer deployments, we now govern more than 1 million BI assets, which is one of the largest real-world samples of what's actually happening in the consumption layer today. Here is what it looks like up close:

- **Dashboards are duplicated.** Often silently. An analyst rebuilds a report she couldn't find, the original keeps running, and now two "sources of truth" are diverging in parallel.
- **Datasets are cloned.** The same fact is calculated three different ways by three different teams, each convinced that theirs is the canonical one.
- **Reports are orphaned.** The author left. The logic is undocumented. It still feeds the Monday exec readout.
- **Costs are invisible.** Nobody can say which 2% of queries are driving 60% of the spend, and most teams find out only when finance escalates.

This is a trust problem. The business is making decisions on the consumption layer while the data team is investing in everything upstream of it. And every one of those patterns gets sharper the moment an AI assistant starts retrieving from the same estate. A duplicate, a stale refresh, an orphaned workbook, an unused report, they don't just sit there anymore. They get quoted. **Trust doesn't live in the warehouse. It lives in the report the executive is actually looking at.**

Why Data Observability Stops Short

The last five years of the modern data stack have been extraordinary for data quality. We now have tests, lineage, freshness monitors, anomaly detection, and data contracts at the warehouse layer. That work is real, and it matters. But data observability, as the category has been practiced, has a clear boundary: it stops at the warehouse.

A freshness check on a staging table does not tell you that a VP is looking at a dashboard whose underlying dataset is three weeks stale. Column-level lineage doesn't tell you that seventeen near-duplicate versions of the same revenue workbook are floating around Tableau with slightly different filter logic. A perfect semantic layer does not tell you that 40% of your published reports haven't been opened in 180 days, or that the one report that is opened constantly is owned by a contractor who left last year.

Everything that happens after the data lands - how it's modeled into a report, who built it, who uses it, what it costs, whether it's one of five copies, whether it's trusted enough to stake a decision on - has been a blind spot.

BI Observability is the discipline of closing that blind spot.

Defining BI Observability

We define BI Observability as continuous, platform-spanning visibility into the consumption layer of the data stack: the reports, dashboards, datasets, users, and costs that sit inside Tableau, Power BI, Looker, Qlik, Sigma, and Spotfire.

Concretely, it covers four things that data observability does not:

- 1. Asset visibility.** A single inventory of every report, dataset, and workspace across every BI tool in the enterprise, not as a monthly extract, but as a live layer.
- 2. Behavioral monitoring.** Who is creating, editing, sharing, and consuming what, and how that behavior shifts when people leave, teams reorg, or a vendor pushes a pricing change.
- 3. Quality and redundancy signals at the report layer.** Duplicate and near-duplicate detection, schema drift on published datasets, stale and abandoned assets, PII exposure, cost spikes tied to specific datasets.
- 4. Governance that doesn't kill self-service.** Guardrails that surface risk and waste without shutting down the analyst who just wants to answer a question before her 3 pm meeting.

That last point matters. The last generation of governance tried to centralize every published asset through a review gate and quietly died because the business routed around it. BI Observability is the opposite move: let self-service happen, and use observability to understand, rank, and remediate what comes out the other side.

And increasingly, it is the layer that gives the AI assistants inside every BI tool something governed to retrieve from, so the natural-language answer they return is worth reading.

What Trust Actually Requires

Steal a page from the software world, and it becomes clearer. Nobody ships a production service without observability. Everyone needs logs, metrics, traces, and SLOs. The reason is not compliance theater. The reason is that when something breaks, you need three things:

- **Consistency:** the system behaves the same way today as it did yesterday, and you can prove it.
- **Auditability:** when a number looks wrong, you can trace it back to the source, the author, and the lineage in minutes, not days.
- **Context:** you know how this report fits into the bigger picture: how it's used, what it costs, what depends on it, and what changes when you touch it.
- **Ownership:** you know who is responsible for

These are the same three properties a business leader needs before she'll bet a decision on a dashboard. Take any one of them away and trust collapses. Most BI estates today are zero-for-three.

They're also, not coincidentally, the three properties an AI assistant needs to ground its answers. A language model retrieving from an estate without consistency can't give the same answer twice. Without auditability, the answer it gives can't be verified. Without context, it can't tell a live report from a graveyard. BI Observability is the layer that gives trust - human or machine - something to stand on.

What This Looks Like in Practice

The regional VP preparing for Monday's board readout used to open Tableau or Power BI and hunt through a folder or workspace for the right workbook or report. That's changing fast.

More and more often, she opens the AI assistant built into her BI tool - Copilot, Pulse, the "ask your data" box - and just asks: *"What was revenue by segment last quarter?"* Or: *"Pull me the churn deck from Q3."*

She's not browsing the catalog anymore. She's asking the assistant to find the report or to answer directly.

Which sounds like progress. Without BI Observability underneath, it isn't. The assistant runs on top of the same messy consumption layer as the rest of the business, and the trust problem doesn't go away - it gets faster.

Without BI Observability, that Monday looks like this:

- The VP asks for "revenue by segment," and the assistant returns an answer pulled from one of seventeen near-duplicate workbooks and reports, **with no indication that the others exist or disagree.**
- She asks for the churn deck and gets a confidently-named workbook whose underlying dataset failed to refresh on Saturday. **Nothing in the chatbot flags it.**
- She re-asks the same question twenty minutes later and **gets a different number**, because a different retrieval path surfaced a different report.
- She only learns the CFO's assistant returned a different answer when the meeting goes sideways.

AI didn't cause those problems, but it amplified them. You cannot ground a natural-language interface in a BI estate where 30% of reports are duplicates with conflicting definitions, 15% are stale, and 5% quietly expose PII, at least not without getting confident, articulate wrong answers at scale.

With BI Observability running underneath, the same Monday looks different. The assistant is answering out of a governed, deduplicated estate. Near-duplicate workbooks and reports have already been flagged, ranked, and consolidated. In our deployments, similarity detection alone surfaces the redundancy that drives a meaningful share of the \$8.2 million in avoidable BI spend we've found across customers. The Saturday refresh failure has opened an alert where the finance team already looks, and the affected report is clearly marked stale. The "Revenue by Segment" logic drift has been traced to the dbt model change that caused it, with downstream workbooks and reports ranked by usage so remediation starts with the one the VP is actually asking about. The PII column that got exposed overnight has been caught before the audit or the chatbot finds it.

None of this work is new to the business. It's work that was happening before - badly, manually, and after the fact. Observability moves it into the background, where it belongs. And it's what makes the AI layer on top of BI trustworthy enough to use.

The Road from Dashboards to Decisions

BI Observability is not the end state. It is the prerequisite for everything the industry wants to build on top of BI next: LLM-driven analytics, natural-language querying, autonomous agents that draft narratives from dashboards, the "decision intelligence" that operations leaders now expect.

None of that works on an unobserved estate. You cannot safely point a large language model at a BI environment where 30% of the reports are duplicates with conflicting definitions, 15% are stale, and 5% quietly expose PII. You get faster wrong answers, with a confident voice attached.

The path from dashboards to decisions runs through observability. Trust first, then automation.

This Isn't an AI Story, but an Infrastructure One

AI is already inside every BI tool. The question is no longer whether the business will use it. It is whether what it says can be trusted. That answer doesn't live in the model. It lives in the layer underneath.

The part of the problem that actually blocks the business is not the reasoning layer on top. It is the messy, sprawling, undergoverned consumption layer the reasoning layer has to retrieve from. Cleaning that up is unglamorous, enduring work. It is also the work that determines whether every AI initiative on the roadmap reaches production or quietly dies in proof-of-concept.

Every prior generation of enterprise data tooling has followed the same arc. Warehouses became useful when we added observability on top of them. Applications became reliable when we added APM on top of them. The same thing is about to happen to BI, because it has to. Too many decisions now run through it, and too many AI assistants now retrieve from it, for the status quo to hold.

We call that layer BI Observability. The companies that build it first, and the vendors who help them, are the ones whose dashboards, and whose AI assistants, executives will actually trust.