www.pipelinepub.com
Volume 22, Issue 7

Telcos Have More Data than Anyone. So Why is AI Failing?

By: Danielle Rios

The telecom industry's own research association recently surveyed 110 operators across 72 companies. Ninety-five percent said intent-based, AI-driven operations are the future. Fifty-eight percent admitted they lack the technology stack to get there.

That gap is not a funding problem. It is not a talent problem. It is a context problem — and until the industry names it correctly, every AI investment will underperform.

When Every System Says Green, But Revenue Says Otherwise

Ask a network operations team a simple question: "Why are customers reporting service degradation when every dashboard shows green?" Then watch what happens.

One engineer pulls network alarms. Nothing. Another checks configuration. Everything is normal. Someone else opens the customer care system and starts reading complaints that have no clear cause.

Every system is telling the truth. But none of them is telling the same story.

This is the operational reality inside most Tier-1 operators. The data exists, the systems are connected, and the investment in analytics is real. Yet a question that should be answerable immediately — what is wrong, where, and what is it costing — takes hours or days to answer. In many cases, it is only answered once the revenue damage is already done.

What Integration Gets Wrong

For decades, the industry has treated complexity as a connectivity challenge. Middleware, APIs, and data pipelines have been built so systems can communicate. That work was necessary. It moved data where it needed to go. It did not solve the problem operators are now trying to answer.

The industry did not solve complexity. It moved it. Integration connects systems. It does not align with what those systems *mean*.

There is a structural reason this gap has persisted. The systems telcos run were built by different vendors, at different times, to solve different problems. Each vendor had every incentive to make their system best-in-class, but no incentive to make it coherent with anyone else's. Interoperability between vendors would have commoditized their implementations. The semantic gaps between systems are not an oversight. They are a structural outcome of how the industry was built.

A "cell" in the network is not the same "cell" in planning. A "customer" in billing is not the same "customer" in customer care. The words are identical. The definitions are not. You can have perfect pipelines and still have no shared understanding of what the data describes.

That is the gap. And it is exactly where most AI initiatives fail.

When Systems Agree, and The Network Still Fails

Zain Sudan, a mobile operator in Sudan and part of the Zain Group, encountered this gap in a way that made it impossible to ignore. They had cell sites that appeared fully operational across every monitoring system: no alarms, correct configuration, and clean health checks. The sites were actually dormant, carrying zero traffic.

Every system reported accurately based on its own logic. OSS showed network availability. Planning tools confirmed deployment. Customer care captured complaints. Revenue systems reflected declining usage. None of the systems could attribute the issue. None could identify the cause. The problem was not hidden in the data. It was hidden *between* the systems.

Technicians spent up to 48 hours tracing a single instance: pulling reports from OSS, cross-referencing planning tools, escalating to customer care, checking revenue systems, and trying to manually reconcile signals that were never designed to align. Each system gave a consistent answer. None of those answers agreed with the others. By the time the issue was identified, revenue had already been lost, and customer experience had already degraded.

A Context Layer Above The Stack

Zain Sudan addressed this by introducing a platform that encodes and normalizes the entities, processes, actions, and relationships that define how the business operates.

Nothing was replaced. Existing systems remained in place. What changed was how their data was interpreted.

A cell site stopped being just a network object in OSS. It became a business entity connected to the subscribers it served, the traffic it carried, the infrastructure it depended on, and the revenue it generated. Those relationships were explicitly defined and consistently applied. Once that model existed, data from across OSS, BSS, and network domains could be correlated in real time against a shared understanding.

The systems did not change what they reported. They changed how those reports were understood together.

With shared semantics in place, previously disconnected signals could be interpreted as part of the same event. A power fluctuation, a transmission constraint, and a drop in throughput no longer appeared as separate alerts. They pointed to a single underlying issue. Root cause analysis that previously required manual investigation across multiple systems could be resolved in minutes, because the relationships between data points were already defined.

At Zain Sudan, diagnosing underperforming sites dropped from up to 48 hours to 30 minutes.

From Performance to Revenue

The most significant shift was not operational. It was economic.

Before, underperforming sites were treated as technical issues. The financial impact was indirect and largely invisible. With a unified data context, that separation disappeared. The operator could identify not only that a site was underperforming, but also how much revenue it was losing and which subscribers were affected.

Issues were no longer ranked by technical severity alone. They were ranked by business impact. Engineering effort aligned directly with business outcomes.

Why This Matters for AI

Here is what most AI vendor pitches do not tell you. A better model on top of a semantically incoherent stack is still incoherent — and now it acts faster.

Hallucination in AI is an architecture problem. When a retention agent coordinates billing, provisioning, and care to execute an offer, it queries systems that have three different definitions of the same customer. The agent resolves that contradiction at inference time — which means it hallucinates on the most consequential operational decisions in the business.

Training a better model misses the point entirely. The fix is a semantic foundation that makes impossible actions architecturally unrepresentable.

A dashboard can tolerate ambiguity. An AI agent cannot. A dashboard shows 10 alerts and leaves it to a human to decide what matters. An AI agent needs to know whether those 10 alerts describe 10 problems, one problem, or no problem at all — and it needs to know before it acts.

This is not a future risk. McKinsey's December 2025 survey of top telco executives found that only 57 percent are scaling AI across multiple domains — and that number has barely moved in a year. The bottleneck is not analytical capability. The bottleneck is that AI inherits whatever context it is given, and in most telco stacks, that context is contradictory.

Rethinking The Foundation

Operators already have the data required to run more efficient networks. They already have the analytical tools. What they lack is a shared model of what that data means across the organization. Without that, systems remain internally consistent but collectively incoherent.

Establishing shared meaning does not require replacing existing systems. It requires defining the existing relationships and applying them consistently — across OSS, BSS, and network domains simultaneously.

The operators who solve this now are the ones whose AI investments will deliver. The ones who do not will spend the next several years adding more capable models to an incoherent foundation and wondering why the results keep disappointing.

If your network showed you a problem right now, would your systems agree on what it was? The operators who can answer yes are not just running better networks. They are the only ones positioned to let AI run them.