

WHITE PAPER

Why Operational Intelligence Needs More Than Dashboards

Connected operational questions need a system that understands relationships, not another view of the same data

Martin Jack

Founder and CTO, BessWave

July 2026

Executive summary

Most mid-market operations run on data that is spread across many systems. Tickets live in one system, devices in another, contracts and customers in a third, sites and applications somewhere else. Each system has its own report or dashboard, and each answers the questions it was built to answer. The trouble starts with the questions that cross those boundaries, because those are the questions that describe how the operation actually runs.

This paper makes a simple case. Dashboards and business intelligence tools are good at showing known metrics on a fixed layout. They are not built to answer the connected, one-off operational questions that come up every day, the ones that join a device to a ticket to a site to a customer. Answering those today means exporting from several systems, lining the records up by hand, and building yet another report. The work is repeated on every new question, and it falls to the few people who understand all the systems.

BessWave is an operations intelligence layer that sits above the systems a team already runs. It connects to those systems, models how their records relate to one another, and lets people across the team ask a question in plain language. Every answer shows the sources behind it. The result is not a faster dashboard. It is a different way to reach the connected answers a dashboard was never designed to give.

The timing matters. Operational data keeps growing, systems keep multiplying, and the pressure to answer faster keeps rising. Adding one more dashboard does not close that gap. The rest of this paper explains why, and what changes when operational intelligence is built around relationships and plain-language access rather than fixed views.

The problem is not a shortage of dashboards. It is that the most useful operational questions cross the boundaries a single dashboard is built on.

The problem: operational data is fragmented

Start with a concession. Fragmentation is not a sign that anything was done wrong. It is the natural result of buying good tools. A team picks the best system for tickets, the best system for device management, the best system for contracts, and each one does its job well. Over time the operation ends up with a handful of capable systems, each holding one slice of the picture.

The slices are not the problem on their own. The problem is that the relationships between them carry most of the meaning, and no single system holds those relationships. A device belongs to a site. A site belongs to a customer. A ticket points at a device. A contract covers some assets and not others. Read any one system and you see a list. Read across them and you see how the operation is actually behaving. That second view is the one people need, and it is the one that is hard to see.

The entities in these examples come from one kind of operation. A different operation connects different systems and different records. The point holds regardless of the specific entities. Value lives in the relationships between systems, and those relationships are exactly what falls through the cracks when data is split across them.¹

To answer a connected question today, someone has to become the integration layer by hand. They export from each system, match records that use different identifiers, reconcile the gaps, and only then can they begin to answer. That reconciliation is quiet, recurring work, and it is a real cost. Estimates of the time knowledge workers spend simply locating and gathering information run close to a fifth of the workweek.² By the time an answer is ready, the underlying data has often moved. The next question starts the process over.

Knowledge workers spend close to a fifth of the workweek just locating and gathering information. That is time fragmentation takes before anyone has answered a single question.

Why current tools fall short

It is fair to ask why the tools already in place do not solve this problem. Most operations already have dashboards, business intelligence tools, and increasingly some form of AI search. Each is good at a specific job. None is built for connected operational questions. It is worth taking them one at a time.

Dashboards. A dashboard is a fixed view of known metrics. It is excellent when the question is stable and repeats: how many tickets are open, how is uptime trending, which sites are over threshold. The strength of a dashboard is also its limit. Someone has to know the question in advance and build the view for it. When a new question comes up, and in operations new questions

¹MuleSoft, 2025 Connectivity Benchmark Report, based on a survey of 1,050 IT leaders conducted October to November 2024. The report finds that data silos create business challenges for roughly 90 percent of organizations, and that on average only about 29 percent of an organization's applications are integrated. The figures are cited here to indicate that fragmentation across systems is a common condition, not as a claim about any specific operation.

²McKinsey Global Institute, "The social economy: Unlocking value and productivity through social technologies," July 2012. MGI reports that the average interaction worker spends nearly 20 percent of the workweek looking for internal information or tracking down colleagues who can help with specific tasks. The figure is used here only to indicate that locating and reconciling information across systems is a real and recurring cost, not to make a claim specific to any one operation.

come up constantly, the dashboard cannot answer it. A request goes to whoever owns the reporting, and the team waits for a new panel to be built.

Business intelligence tools. BI tools are more flexible than a single dashboard. A skilled analyst can join data sources and answer a wide range of questions. That flexibility comes at a cost. It requires modeling the data, writing the queries, and maintaining the pipelines, and it requires a person who knows how to do all three. BI answers connected questions, but only for the people who can drive it, and only after the work of joining and modeling is done. The long tail of everyday operational questions rarely justifies that effort, so it goes unasked.

Basic AI search. Plain-language search over records is a real step forward, and it removes the need to know a query language. On its own, though, it treats records as separate fragments and retrieves the ones that look similar to the question. That works for finding a document or a single fact. It struggles with questions that depend on how records relate, because similarity is not the same as relationship. Asking which customers have unresolved incidents tied to covered assets is not a search for similar text. It is a walk across several connected systems, and simple retrieval does not make that walk.

The common thread is that each tool answers isolated questions well but connected questions poorly. The gap is not intelligence. It is structure. None of these tools holds the relationships between systems, so none of them can reason across those relationships on demand.

The gap is not intelligence. It is structure.

The BessWave approach

BessWave closes that gap by making the relationships the center of the product, not an afterthought. Four ideas describe how it works.

Relationship-aware intelligence. When BessWave connects to a system, it does more than store the records. It builds an operational graph that models how those records relate to one another across systems. A device connects to its site, its tickets, and its contract, and the graph holds those links directly. A question that spans several systems is answered against that connected picture rather than against a pile of separate records. This is the difference that lets BessWave answer the connected questions that defeat isolated tools.

Cross-system answers are the point. This is worth stating plainly, because it is where the value of connecting systems comes from. A single connected system is useful, but it puts BessWave in its weakest position, because there are few relationships to cross. The value grows sharply when two or more systems are connected, because the answers that matter most are the ones that join them.

A graph built from your data. The operational graph is not a fixed template that assumes a particular set of entities. It is built from whatever a team connects. The entities used as examples in this paper are illustrative. Your graph reflects your systems and your records, and it changes as you connect more.

Grounded retrieval. To answer a question, BessWave combines search with the operational graph, an approach the research community calls graph-based retrieval. Rather than pulling back the fragments that look most similar to the question, it uses the relationships in the graph to gather the

records that actually bear on the answer. Graph-based retrieval has been shown to improve answers on complex, corpus-wide questions compared with similarity search alone.³

Conversational access with sources. A person asks in plain language and reads a plain-language answer. There is no query to write and no view to build in advance. Every answer carries its sources, so the reader can see which systems it came from and decide whether to act on it. The plain-language front and the sourced back are what let the whole team use the tool, not only the few who know the systems.

One point underlies all of this. BessWave uses strong, general language models for the language part of the work, and it treats the model as one component that can be swapped as better models appear. The product is everything around the model that makes an answer reliable on real operational data: the connectors, the operational graph, the grounded retrieval, and the sources behind every answer.

The model is one component. The system is the product.

Example questions this answers

The clearest way to see the difference is in the questions themselves. The examples below are illustrative. They reflect one kind of operational dataset, and a different operation would ask different questions across different systems. What they share is that each one crosses systems, which is exactly why a single-system report cannot answer it.

Which locations have aging devices with recurring tickets? Without this, you keep paying to patch hardware that should have been replaced, and the failures concentrate on the sites least able to absorb them. The answer joins devices, their age, their sites, and the ticket history against each, which no device inventory or ticket system holds on its own.

Which customers have unresolved incidents tied to covered assets? Without this, you renew accounts that are quietly at risk, or you absorb SLA exposure you could have caught first. The answer depends on which assets a contract actually covers, a relationship that lives between systems.

Where is ticket volume rising fastest against assets near end of contract? Without this, the renewal conversation starts after the trouble is already visible to the customer, instead of before, while you still have room to act. The answer joins ticket trends, assets, and contract dates to surface the renewals worth attention before they lapse.

Which sites carry the most risk from a single point of failure? Without this, the first time you learn a site depends on one fragile device is the day it fails. The answer walks the relationships between sites, the devices they depend on, and the tickets those devices generate.

Each of these is a routine operational question. Each is hard to answer today because the answer is spread across systems that do not share a view. BessWave answers them in one step, in plain language, with the sources attached.

³Edge et al., "From Local to Global: A Graph RAG Approach to Query-Focused Summarization," Microsoft Research, April 2024, arXiv:2404.16130. The paper finds that graph-based retrieval improves the comprehensiveness and diversity of answers over naive vector retrieval on complex, corpus-wide questions. BessWave applies a relationship-aware operational graph for the same reason.

Security and trust

Operational data is sensitive, and a tool that reads across a team's systems has to earn trust before it earns use. BessWave is built with that in mind.

Isolation. Each customer's data is isolated by tenant. Answers for one organization are drawn only from that organization's connected systems.

No training on your data. Customer operational data is not used to train or fine-tune BessWave models. The language models BessWave uses for the language part of the work do not learn from the operational data they help answer questions about.

Sources on every answer. Every answer shows the systems it was drawn from. This is a deliberate design choice. It lets a reader verify an answer rather than take it on faith, and because each answer carries its sources, each response leaves a traceable record of what it was based on.

Independent assurance. BessWave is being built toward SOC 2 from the start rather than retrofitted later. SOC 2 is an independent audit of a service organization's controls, commonly covering areas such as security, availability, and confidentiality.⁴ BessWave is working toward it and is not yet certified.

Put simply, trust here is built into the structure, not just promised. Tenant isolation, no training on your operational data, and sources on every answer are properties of how the platform is built, not settings a user has to remember to turn on.

Fast time to value

BessWave sits above the systems a team already runs. There is no wholesale data migration and no new system of record to stand up, because BessWave reads from the systems already in place rather than replacing them. Value begins when connected systems can answer a question that neither system could answer alone, and reaching that point does not depend on a long embedded rollout first.

Business outcomes

The point of all this is not the architecture. It is what changes in the working day.

Decisions while they still matter. A connected question that used to mean exporting, matching, and building a report becomes a plain-language question with a fast answer. The point is not speed for its own sake. It is that the decision gets made while it can still change what happens, rather than after the moment has passed.

Skilled time returned to real work. Much of the reporting in an operation exists only to answer one-off questions that no standing dashboard covers. When the team can ask those directly, the people who used to assemble those reports get that time back for the work only they can do.

⁴SOC 2 is an independent audit of a service organization's controls, commonly covering areas such as security, availability, and confidentiality, defined by the AICPA. BessWave is building toward SOC 2 from the start and is not yet certified.

Answers reach the whole team. Because a person does not need to know the systems to ask, an operational answer stops being gated behind the one specialist who understands the data. More of the team can act without waiting in that queue.

Risk caught early. Cross-system questions that were too much trouble to ask simply went unasked, and the risks hiding across systems went unseen with them. When those questions are easy, the operation sees the exposure and the patterns early enough to act on them.

Reporting that stops scaling with every question. Dashboards remain the right home for the stable metrics a team watches daily. The difference is that they no longer have to carry every ad hoc question, so the pile of custom reports and the requests to build new ones both stop growing with each new thing someone needs to know.

We are not going to put invented numbers against these outcomes. The honest claim is directional: where a team spends real effort exporting, reconciling, and reporting across systems, that is the effort BessWave is designed to remove.

How the tools compare

The distinction is easiest to read side by side. The comparison below is between dashboards and business intelligence tools on one side and BessWave on the other.

Dimension	Dashboards and BI tools	BessWave
What they answer well	Known, repeating metrics on a fixed view	Connected, one-off questions across systems
New questions	Need a new view or query built first	Asked directly in plain language
Relationships across systems	Joined by hand or by an analyst	Modeled in an operational graph
Who can ask	People who own reporting or know the tool	Anyone with appropriate access
Freshness	As current as the last export or refresh you built	Refreshed from connected systems on a schedule you set
Source traceability	Depends on how the report was built	Every answer shows its sources

When dashboards are enough

It is worth being clear about where a dashboard is the right tool. When the question is stable, watched every day, and made of metrics from a single system, a dashboard is hard to beat. BessWave is not trying to replace that. Dashboards are for the questions you already know you will ask again tomorrow. BessWave is for the connected questions you cannot predict.

The shift

The shift underway is not from one dashboard to a better dashboard. It is from fixed views toward conversational operational intelligence: plain-language access to answers that cross systems, grounded in relationships, and backed by visible sources. Dashboards will keep doing what they do

well. Connected questions need something built around relationships and plain-language access. That is the category BessWave is building.

The clearest way to evaluate BessWave is on your own data: connect the systems you already run, ask the cross-system questions your team has been answering by hand, and review the answers with sources attached.

Martin Jack

Founder and CTO, BessWave

martin.jack@besswave.com | besswave.com