mondoo

CASE STUDY

GUARDRAILS, NOT GATES

How a long-established global manufacturer enforces compliance as code with Mondoo — on every change, before infrastructure is ever built.

Internal standards do not fail because they are wrong. **They fail because nothing enforces them.** The organizations that treat cloud governance as a documentation exercise write standards almost nobody reads.

The ones that read the problem correctly build something different: guardrails that enforce policy automatically, where developers already work, without slowing them down. The manufacturer in this case study is in the second group, and Mondoo is the platform those guardrails run on.

AT A GLANCE

ORGANIZATION: A long-established global manufacturer in the transportation sector

SECTOR: Manufacturing and transportation, with multiple business units, including a financial-services arm

SCALE: Thousands of software projects and development teams distributed worldwide, deploying into thousands of separate cloud projects

ENVIRONMENT: Cloud infrastructure defined in Terraform, shipped through CI/CD pipelines in GitHub; an Active Directory holding millions of identities

ENGINEERING MODEL: Autonomous, high-velocity teams that own their own infrastructure — no change-control boards

MONDOO'S ROLE: Policy-as-code enforcement in every CI pipeline, checking Terraform against company-specific standards before infrastructure is built

01
THE COMPANY

A TRADITIONAL MANUFACTURER WITH A MODERN ENGINEERING ORGANIZATION

Despite its image as a traditional industrial company, this manufacturer runs a remarkably modern engineering organization. Thousands of independent teams build and deploy their own applications — everything from supply-chain ordering and time-tracking systems to customer-facing product APIs. Each team owns its own infrastructure, deployed into thousands of separate cloud projects, and ships at high velocity without heavyweight change-control boards or burdensome security review.

That autonomy is a deliberate strategy, and it works. But it created a growing problem: when any team can decide what its infrastructure looks like, it can also decide how secure — or insecure — that infrastructure is.

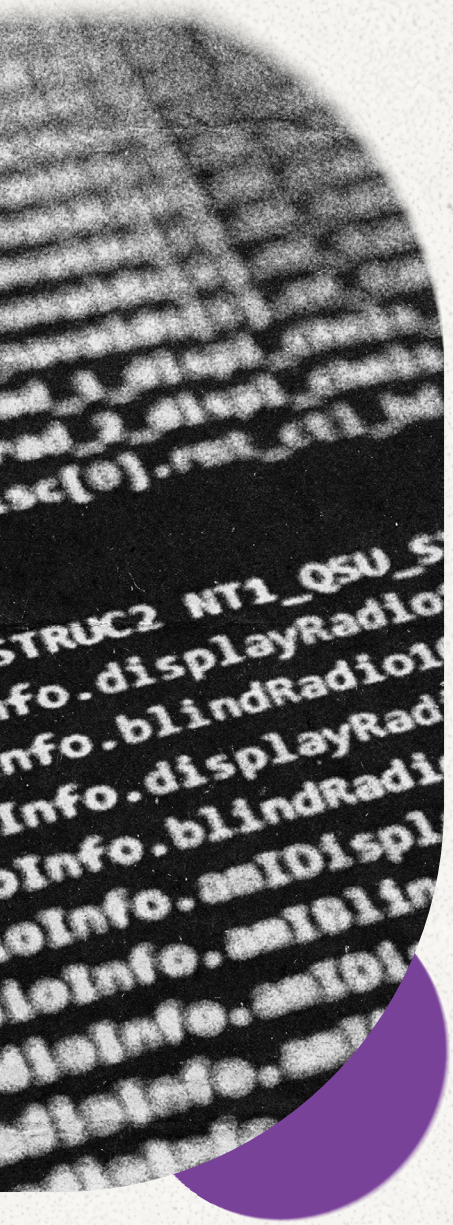
02
THE CHALLENGE

FREEDOM WITHOUT GUARDRAILS

The company had written down internal standards: a production database should have replication and backups; a development instance shouldn't; service accounts shouldn't use over-privileged cloud defaults; instances should be right-sized for their environment. The team calls this "little c" compliance — not an external framework like SOC 2 or ISO 27001, but the company's own definition of what its infrastructure should look like. Like most internal standards documents, almost nobody read them, and nothing enforced them.

The consequences were real and expensive:

Outages: A single misconfiguration once took down a critical ordering system used by the company's business partners. With orders halted, partners' downstream operations stopped and they turned to alternative suppliers — revenue simply lost.



Downtime in production facilities costs millions of dollars.

Waste: Development environments running production-grade replication, oversized instances, and terabytes of unnecessary data retention added up to significant avoidable cloud spend.

Risk: Teams reusing default, effectively root-level cloud service accounts — a known anti-pattern the cloud team suspected was widespread but had no way to find or prove.

WHY THE PREVIOUS TOOL FAILED

The company had already tried policy scanning with a popular open-source infrastructure-as-code checker. It ran across the entire business — **and accomplished almost nothing, for two reasons**.

First, it couldn't express company-specific policy: its built-in rules reflected generic industry best practices, but the company needed its own best practices enforced — its database standard, not a checkbox standard.

Second, it never blocked anything: findings were buried in thousands of pages of build logs while pipelines stayed green. Developers never saw a problem, so nothing changed.

03
THE SOLUTION

POLICY AS CODE, ENFORCED WHERE DEVELOPERS LIVE

The company deployed Mondoo into every CI pipeline. Every software change across the business now runs through Mondoo, which scans the Terraform code that builds infrastructure — catching misconfigurations before they ever reach production.into the existing workflow. Crucially, the frameworks and their controls are built into the platform as traceable code, so there are no mapping tables to build and no derivations to write up by hand.

CUSTOM POLICIES, WRITTEN BY THE EXPERTS

Subject matter teams author their own policies. The database team — the first major adopter — has written roughly a dozen policy packs covering every database technology the company runs: ~20 checks for PostgreSQL, a similar set for BigQuery, and so on. Policies apply contextually: when a pull request spins up a Postgres database, the Postgres policy runs against exactly that code.

> “We said production databases must have replication. Do they? For the first time, the database team can see the answer in one view — across thousands of cloud projects.”
>
> - TECHNOLOGY LEADER, CLOUD INFRASTRUCTURE TEAM

FEEDBACK WHERE DEVELOPERS WORK

When a developer — perhaps in India, with the platform team asleep in the US — opens a pull request that violates policy, the pipeline goes red immediately, in GitHub, in their existing workflow. No scanner portal to learn, no ticket to file, no security team in the loop. They read the finding, fix the Terraform, push a commit, and the pipeline turns green. They’ve corrected themselves back into the guardrails, in their own time zone, with minimal lost velocity.

EXCEPTIONS WITH INFORMED CONSENT

At enterprise scale, every rule has legitimate exceptions — perhaps 20 projects out of thousands genuinely need replication in development. The company’s previous homegrown exception process was a convoluted tangle of internal web apps. With Mondoo, developers request exceptions in-app, and requests route automatically to the right subject matter expert: database exceptions go to the database team, cloud-platform exceptions to the cloud team. The requesting team never approves its own exception, and every decision leaves an audit trail. Once approved, the pipeline re-runs and goes green.

A FORCING FUNCTION FOR LEGACY DEBT

This is no greenfield environment — it is about as brownfield as it gets. The goal wasn’t just to stop new violations but to steadily drain decades of accumulated configuration debt. CI enforcement does this automatically: touch a dormant

repository for any reason — even a typo fix in a README — and Mondoo flags every existing violation in that project before the change can merge. Problems eventually fix themselves, without anyone assigning a remediation backlog.

04
SCALE & INTEGRATION

BUILT FOR GENUINE ENTERPRISE SCALE

Integration demands were extreme. The company's Active Directory holds millions of identities — anyone who has ever worked with the business in any capacity. New employees belong to hundreds of groups on day one, blowing past assumptions baked into standard identity protocols. Role-based access control had to be airtight: in a company where security data could hint at an unannounced product program, people must see only what they're responsible for.

The entire Mondoo deployment is also 100% Terra-form-managed — zero button clicks — covering identity integration, organizational structure, and workspace layout. The structure supports top-down corporate policy with controlled business-unit autonomy: the financial-services arm, which operates almost as a separate company, can layer its own stricter requirements on top of corporate baseline policy but can never override it. Full enterprise integration was delivered in under a year.

> "Mondoo is the fastest-moving vendor we've worked with – the only one that's fully integrated with all of our systems. We have giant vendors who've told us they haven't built that in years."

- TECHNOLOGY LEADER, CLOUD INFRASTRUCTURE TEAM

05
RESULTS

70,000+

CI scans per month — a project scanned roughly every 30 seconds

3,800+

software projects with infrastructure code discovered and brought under policy

22,000

policy findings remediated

GUARDRAILS THAT WORK

Enforcement has only recently been switched on, and the early data shows the model working. A software project is scanned for compliance roughly every 30 seconds, around the clock. ~31,000 scans flagged violations that, under the old tooling, would have shipped invisibly [confirm timeframe: 28 days vs. all-time]. Active findings are holding flat while remediations climb — even as more policies roll out. The divergence of those two lines is the signature of effective guardrails: new violations are being prevented at the source while legacy issues are steadily fixed.

Just as importantly, the company kept what it values most. Teams still ship at high velocity, still control their own destiny, and still never sit in front of a change-control board.

The guardrails replaced bureaucracy, not autonomy.

06
THE TAKEAWAY

STANDARDS CAN'T BE WRITTEN. THEY HAVE TO BE ENFORCED.

Documentation is the floor; enforcement at the source, continuously and at scale, is what changes infrastructure. This manufacturer is proof that the policy-as-code operating model works in a real, brownfield enterprise — thousands of autonomous teams, millions of identities, decades of configuration debt — without a single change-control board. Mondoo turns internal standards into guardrails, and guardrails into outcomes. **In that order.**

mondoo.com
email
linkedin