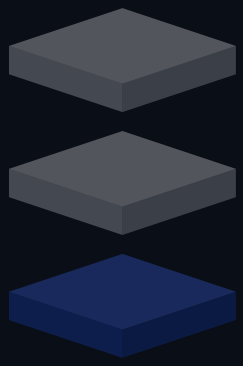
Ariavor



A stalled Power BI rollout, taken over and finished

A mechanical engineering company whose Power BI implementation had stalled, with inconsistent models and source systems that had never been connected. It now runs on one KPI framework, a report catalogue and a budgeting process built into the model.

SECTOR
Mechanical engineering

STARTED
09/2025, ongoing

COMPANY
50—250 people

FOCUS
Finance and budgeting

OCTOBER 2026

AT A GLANCE

One-off files, replaced by a catalogue with owners

The Power BI rollout had stopped halfway. Reports had multiplied as one-off files, the models underneath them disagreed, and several source systems had never been connected. We audited what existed, made it consistent and put one binding set of KPI definitions under all of it, the budget included.

No outcome figures are published for this engagement: nothing exists that would let you check them, so what was built is described in full instead. Anonymised by sector and size, because the confidentiality that protects this client would protect you too. Sector and outcomes are real; identifying detail is not.

Before and after

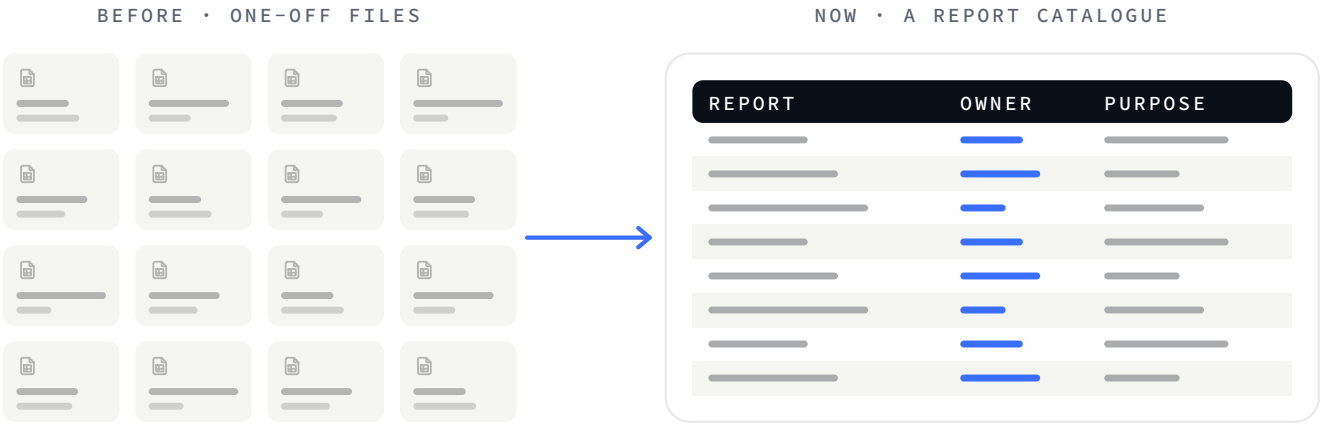


Figure 1 · Before, reports were one-off files with no stated owner or purpose. Now each report sits in one catalogue with its owner and its purpose. Illustrative: no client report names are shown.

CLIENT

A mechanical engineering company of 50 to 250 people, with a Power BI implementation that had been started and then stalled.

MANDATE

Take over the implementation, make what existed consistent, connect the systems it was missing, and bring the budgeting process into the model.

How the takeover ran

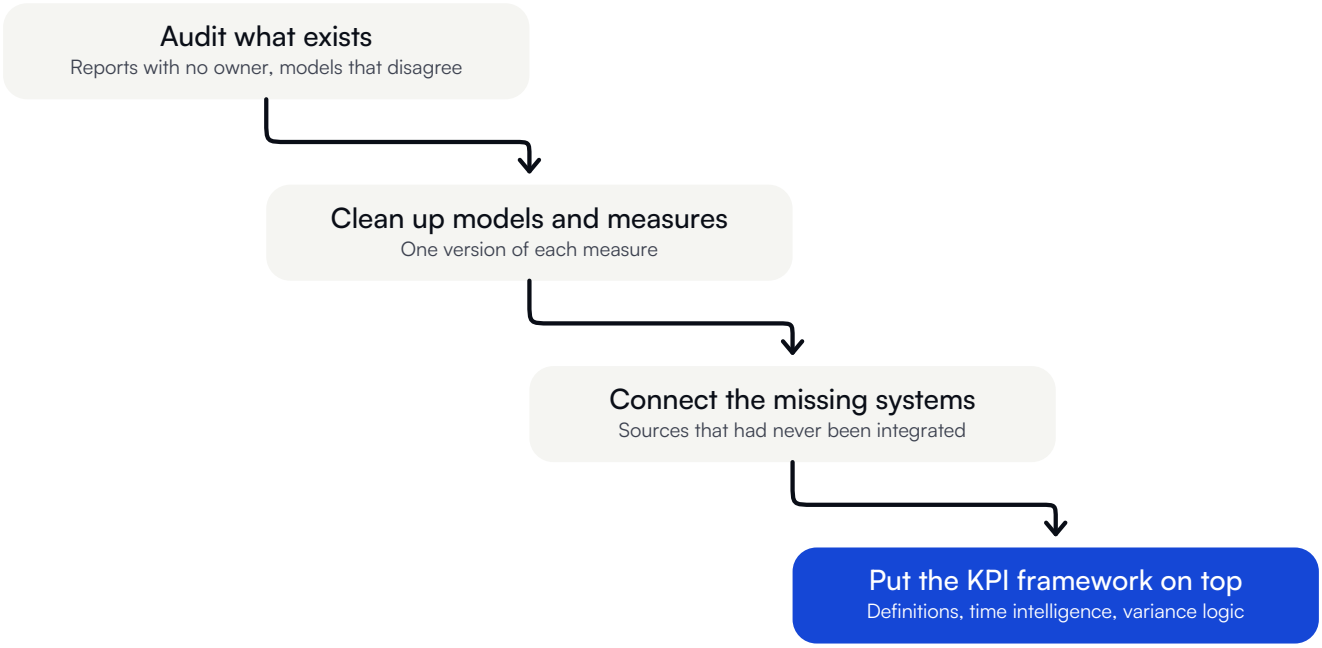


Figure 2 • Each step builds on the one before: nothing new was added until what existed was consistent.

The architecture

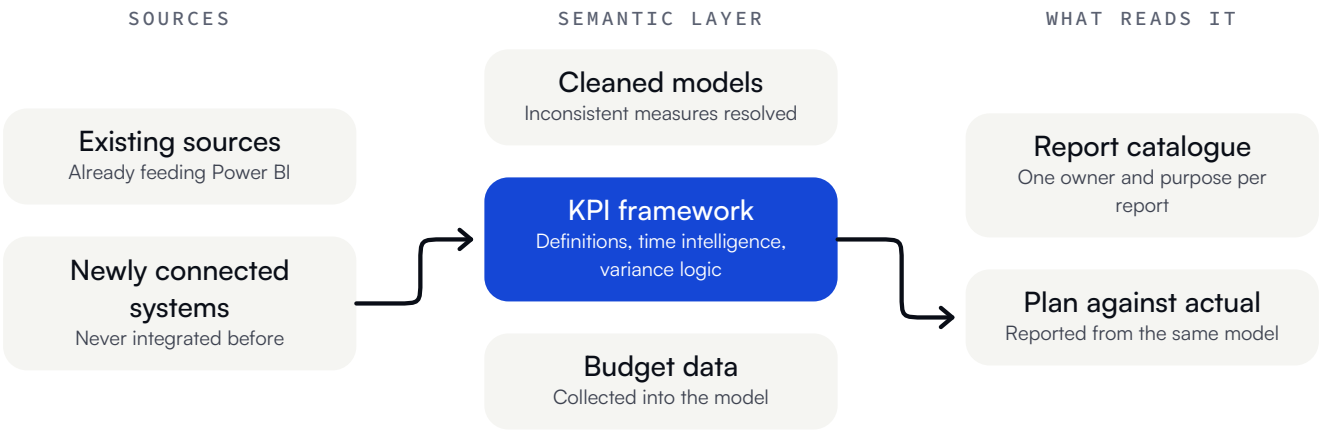


Figure 3 • Existing and newly connected sources feed cleaned models under one KPI framework, which the catalogued reports and the budget reporting both read.

Architecture decisions

DECISION	REJECTED ALTERNATIVE	WHY
Audit before building	Adding new reports on top	The models underneath disagreed, so every new report would have inherited the disagreement.
A catalogue with an owner per report	More one-off files	A report nobody owns is a report nobody corrects, and the sprawl was the original problem.
One binding KPI framework	Definitions kept per report	Time intelligence and variance logic decided once mean every report answers the same way.
Budget inside the model	Budget kept in a separate workbook	Plan and actual read from one place, so plan against actual needs no reconciliation.

The budgeting process, end to end

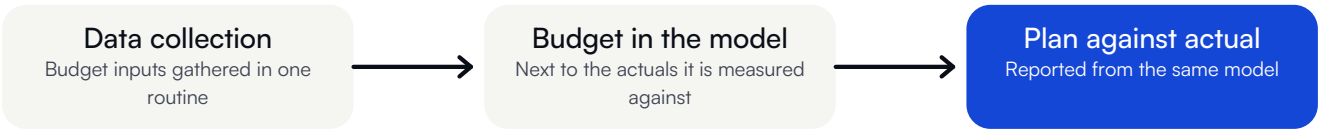


Figure 4 • The budgeting process redesigned from data collection through to plan-against-actual reporting.

How the platform is run

Refresh routines

Scheduled, with a known order, so figures are current when people open them.

Release routines

Changes reach the reports through a defined release rather than ad hoc.

How the platform is run (cont.)

Naming conventions

Models, measures and reports named the same way throughout.

Documentation and handover

Written down so the client's team can run the platform themselves.

What the client has today

- A report catalogue in place of the one-off files, with an owner and a purpose for each report.
- Consistent models and measures, with the source systems that were missing now connected.
- A binding KPI framework as the single source of truth for definitions, time intelligence and variance.
- A budgeting process that runs from data collection to plan against actual inside the model.
- Refresh and release routines, naming conventions and documentation for a clean handover.

WHERE IT STANDS

A rollout that had stopped now has one set of definitions, an owner for every report and the budget in the same model as the actuals.

STACK

Power BI · DAX · Semantic modelling · SQL · Dataflows · Data governance

Niklas Hoeppener • Founder • info@ariavor.com • ariavor.com