

DATA STRATEGY TRACKS

DataOps

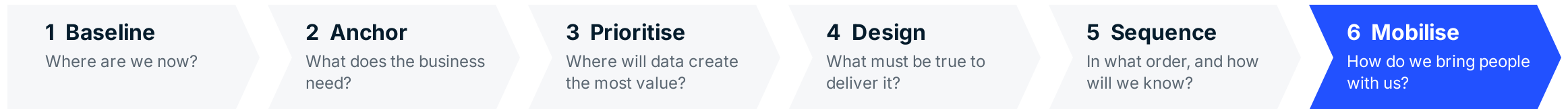
DataOps is a working practice that brings communication, integration, automation, observability and operational discipline to the flow of data between the people who produce it and the people who use it.

THE SO WHAT OF THIS TRACK

Measure how long it takes to deliver a trusted dataset and how often it breaks, then fix the practices that move those two numbers before buying anything new.

Part of stage 6: Mobilise

A strategy that is right on paper still fails if every new dataset takes a quarter to arrive and nobody trusts it when it does. DataOps is how the data team turns the roadmap into a steady flow of usable, decision-quality data. It belongs in the strategy because it sets the delivery speed and the level of trust that every other objective depends on.



QUESTIONS THIS TRACK ANSWERS

- How long does it take today to turn a request for data into a trusted, usable dataset?
- How often do pipelines break or deliver wrong data, and who notices first?
- Which software delivery habits can the data team adopt this year?
- Who owns each data product end to end, from source to consumer?
- Where should DataOps start so that it shows value quickly?

1

The method

Deliver trusted data at the speed the strategy needs

The friction DataOps removes

Traditional delivery fails on both sides of the pipeline.

Producing trusted data

- Hand-built pipelines that only one engineer understands
- Changes deployed by hand, tested in production
- Breaks discovered by the business, not by the team
- Every request treated as a new project

Consuming decision-quality data

- Long queues before a new source or field arrives
- No clear definition or owner for the data received
- Reports that disagree, so meetings debate the numbers
- Analysts rebuilding the same extracts in spreadsheets

SO WHAT

If both columns sound familiar, the bottleneck is how data is delivered, not which platform delivers it.

The outcomes leaders want but rarely measure

Outcome	What to measure	Why it matters
Speed	Time from request to a trusted dataset in use	Sets how fast the strategy can move
Throughput	Usable datasets and data products delivered per quarter	Shows capacity, not just effort
Quality	Data incidents reaching consumers	Trust is lost one wrong number at a time
Rework	Share of effort spent fixing or rebuilding	Rework is capacity the strategy never gets
Reuse	Consumers per data product	Reuse is where delivery pays back

SO WHAT

Pick two of these, measure a baseline this month, and put them on the data scorecard.

Lessons borrowed from software delivery

Software teams solved the same friction years ago. Most of the habits transfer directly.

Version everything

Pipeline code, transformations, schemas and tests live in version control with review.

Test automatically

Every change runs tests on structure, volume, freshness and business rules before it ships.

Deploy continuously

Small changes move through development, test and production by pipeline, not by hand.

Observe in production

Monitor freshness, volume, schema drift and quality, and alert the team before consumers notice.

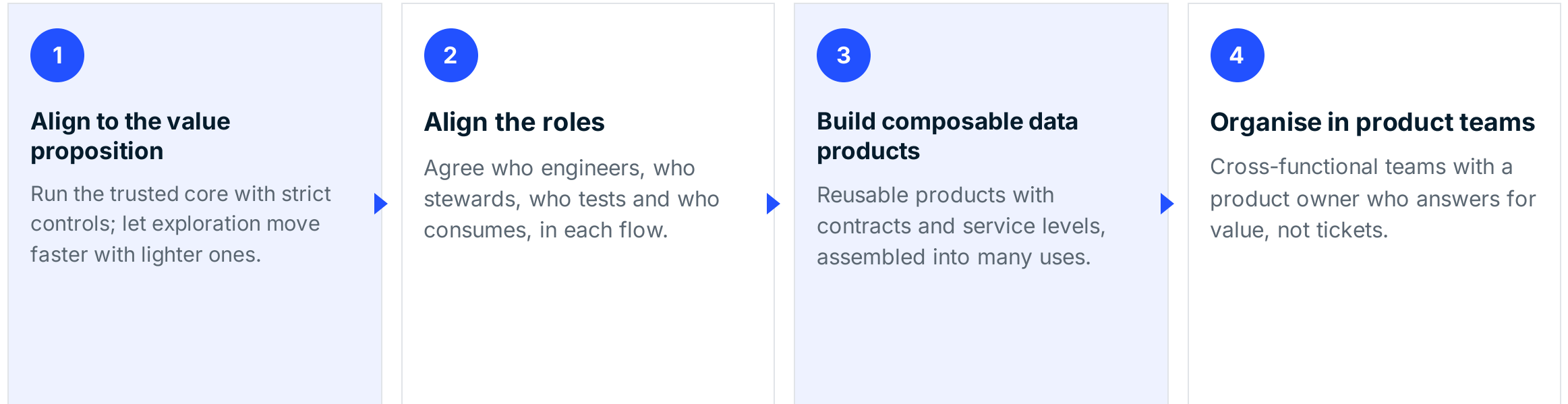
Work in small batches

Ship a usable slice in weeks, then extend, rather than a large release after months.

Learn from incidents

Blameless reviews turn each break into a test or a control so it does not recur.

Four foundations for introducing DataOps

**SO WHAT**

The four foundations change how work is organised; tools only make them cheaper to run.

An illustrative data product team

Role	Accountable for	Sits in
Product owner	Value, priorities and consumer satisfaction	The business domain
Data engineers	Pipelines, tests and deployment	The product team
Analytics engineer	Models, metrics and semantic definitions	The product team
Data steward	Definitions, quality rules and access	The business domain
Platform engineer	Shared tooling, environments and observability	The central data team
Consumers	Feedback and acceptance of each slice	The business domain

SO WHAT Placing the team close to the business that uses its products shortens every feedback loop.

Start small, with sponsorship

One domain, one product

Choose a data product a sponsor cares about and that breaks often today.

Measure before and after

Baseline delivery time and incidents, so the improvement is visible.

Ride on other changes

Pair DataOps with literacy and governance work already in flight.

Scale the pattern

Turn what worked into templates, then spread team by team.

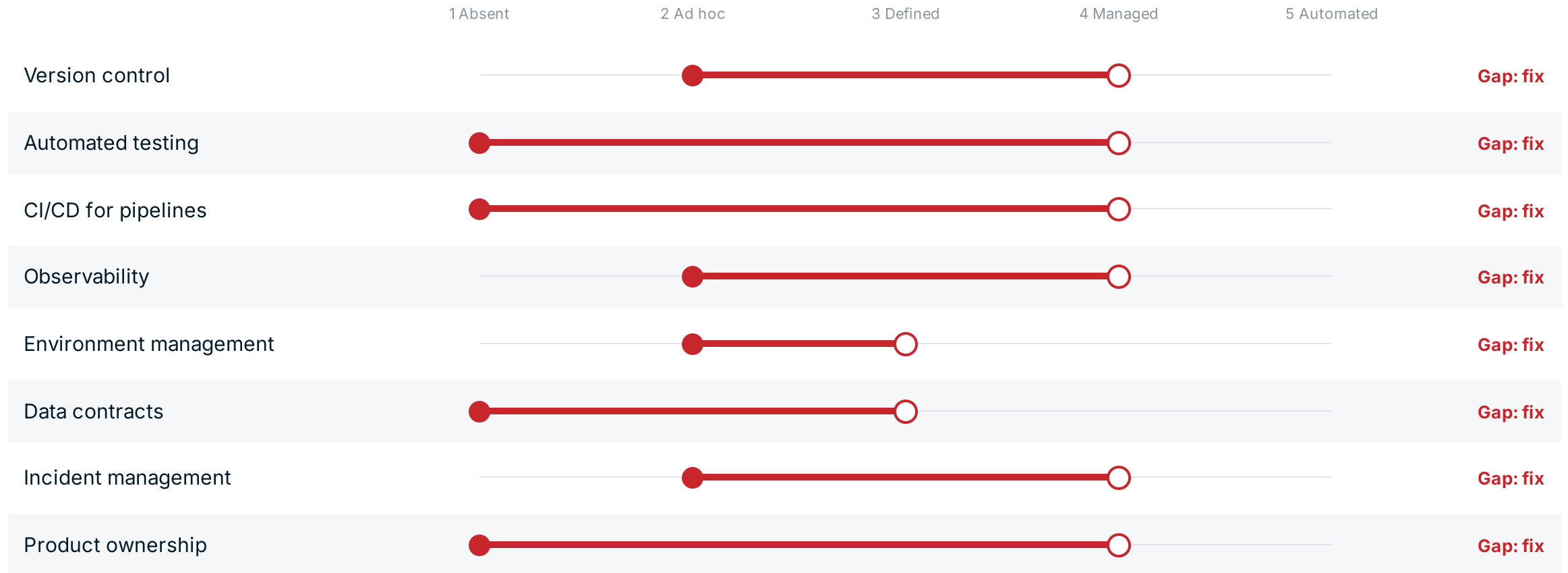
2

Tools and evidence

Score it, measure it, see it in six industries

DataOps practice gap

Rate where each practice is today and where the strategy needs it. Deficits are listed in order, with a first fix.



Filled dot: current level. Open dot: level the strategy needs. Illustrative defaults; score your own.

Closing the gaps that matter

Capability	What it covers	First fix
Version control	Pipeline code, transformations and schemas are versioned and reviewed	Move every pipeline into a shared repository with peer review on merge
Automated testing	Changes are tested for structure, volume and business rules before release	Add freshness, row-count and null tests to the ten most-used pipelines
CI/CD for pipelines	Changes move between environments by an automated pipeline	Automate deployment for one data product end to end, then template it
Observability	Freshness, volume, schema drift and quality are monitored with alerts	Alert the owning team on failure before the morning reports run
Environment management	Separate development, test and production with representative data	Stand up a test environment with masked, representative data
Data contracts	Producers and consumers agree schema, meaning and service levels	Write contracts for the critical feeds into the scorecard data products
Incident management	Breaks are logged, triaged, fixed and reviewed without blame	Start an incident log and a fortnightly review that adds a test per incident
Product ownership	Each data product has an owner accountable for value and service	Name a product owner in the business for each priority data product

SELF-CHECK

Is this in place today?

Score each statement 1 (not at all) to 4 (fully, evidenced).

#	Statement	Score
1	We know how long it takes to deliver a trusted dataset, and we track it.	
2	Pipeline changes are versioned, tested and deployed automatically.	
3	We usually find out about data breaks before our consumers do.	
4	Each priority data product has a named owner and a service level.	
5	Producers and consumers agree contracts for critical data feeds.	

ILLUSTRATIVE

How it looks in six industries

Industry	Situation	The move	Measure
Manufacturer	Plant quality and delivery data arrive late and differ by plant.	A product team for the delivery-performance data product, with automated tests on every plant feed.	Days from plant month-end to trusted delivery figures
Retail bank	Supervisory findings on risk data aggregation and manual report fixes.	Version-controlled, tested pipelines for regulatory metrics, with an incident log reviewed monthly.	Data incidents reaching regulatory reports
Retailer	Stores and online plan stock from separate, conflicting extracts.	One stock-position data product with a contract and freshness alerts, owned by merchandising.	Freshness of the stock position at trading start
Hospital group	Bed and discharge data are compiled by hand each morning.	An automated patient-flow pipeline with tests on admissions and discharge feeds.	Time from shift change to a trusted bed position
Telecom operator	Separate billing and CRM stacks break churn models whenever a field changes.	Data contracts on billing and CRM feeds, with schema-drift alerts to the owning teams.	Churn model runs lost to upstream changes
Insurer	Actuarial and regulatory reporting relies on manual reconciliations.	Automated reconciliation tests in the claims and policy pipelines, deployed by pipeline.	Analyst hours spent on manual reconciliation

Measures for the data scorecard

Measure	Definition	Lead or lag
Lead time to trusted data	Days from an accepted request to a dataset in use with definitions and tests	Lead
Deployment frequency	Pipeline changes released to production per month	Lead
Automated test coverage	Share of priority pipelines with automated quality tests	Lead
Contracted feeds	Share of critical feeds covered by a data contract	Lead
Incidents reaching consumers	Data breaks found by consumers rather than by monitoring	Lag
Time to restore	Hours from detection of a data break to a corrected dataset	Lag
Rework share	Share of team effort spent fixing or rebuilding existing pipelines	Lag
Data product reuse	Average number of consuming use cases per data product	Lag

3

Making it happen

Traps, the first 90 days and where SCIKIQ fits

Traps to avoid

Tool first

A new orchestration tool is bought and the same manual habits move onto it. Counter-move: Change the practices on one product first; choose tooling once the way of working is proven.

Big bang

DataOps is launched across every team at once and stalls in training. Counter-move: Start with one domain and one sponsor, measure the gain and spread the template.

No consumer in the loop

Pipelines are faster but still deliver data nobody asked for. Counter-move: Put consumers and a business product owner in the team and accept work slice by slice.

Speed without trust

Releases accelerate and incidents rise with them. Counter-move: Gate releases on automated tests and track incidents alongside lead time.

Same rules everywhere

Exploration is slowed by controls designed for regulatory reporting. Counter-move: Set control levels by value proposition: strict for the trusted core, lighter for experiments.

The first 90 days

Days 1-30: Baseline and choose

- Measure lead time, incidents and rework on current delivery
- Choose one priority data product and its sponsor
- Name the product owner and form the team

Days 31-60: Put the practices in place

- Move the product's pipelines into version control with review
- Add automated tests and freshness alerts
- Write the first data contracts with source owners
- Start the incident log and review

Days 61-90: Prove and template

- Automate deployment end to end for the product
- Report before-and-after measures to the sponsor
- Turn the setup into a template for the next two teams

Where SCIKIQ fits

Virtual Data Fabric

Connects and federates sources in place, so new data products start without building another copy pipeline.

Semantic Data Layer

Business objects and semantic models give every data product one shared definition for consumers.

Outcome and Observability

Measures outcomes and feeds learning back, so the team sees what changed and what it delivered.

Governance Gate

Checks that access and actions are compliant before they run, so faster delivery does not bypass control.

Take it further

- The SCIKIQ Data Strategy Framework: the cascade this track plugs into
- Strategy Cascade and Scorecard Workbook: add this track's objectives and measures
- Run the interactive tool and self-check at scikiq.org/data-strategy#track-dataops

scikiq.org/data-strategy