CASE STUDY 10 · DATA ENGINEERING, GOVERNANCE & BI

# Rebuilding a marketing data foundation: Redshift to Snowflake, governed definitions and a new dashboard suite

A company-wide warehouse migration gave marketing a hard deadline: when Amazon Redshift shut down, every marketing dashboard would stop refreshing. I used the deadline to do more than move tables—redefining the ETL and business rules, documenting them in an Atlan data dictionary, feeding an AI semantic layer, and relaunching a suite of Tableau dashboards on clean, modeled Snowflake data.

**Jennifer Newsome** · Senior Marketing Analytics Manager

| MY ROLE | PARTNERS | METHODS | TECH STACK |
|---|---|---|---|
| Project owner for the marketing migration: business rules, SQL logic, validation, Atlan definitions, dashboard rebuild and enablement | Data engineering, an analytics-engineering consultant, data project management, marketing operations, revenue operations, web analytics | Dimensional modeling, ETL rule design, parity testing and UAT, data governance, dashboard information design | Amazon Redshift, Snowflake, dbt Cloud and MetricFlow, Stitch, AWS S3, Atlan, CoCo, SQL, Tableau Cloud, Glean, Salesforce, Marketo |

*Company, team and proprietary field details have been anonymized; technology vendors are named. Charts and dashboard figures use illustrative data, not actual results.*

THE BOTTOM LINE

The migration succeeded because it was treated as **a business-definition project, not a lift-and-shift**. Twelve overlapping legacy query datasets were consolidated into four governed Snowflake sources of truth, every rule change was traced from SQL to Snowflake to Atlan, and the dashboards were relaunched on a model that an AI agent can now query using the same definitions people use.

01 · THE CHALLENGE

## A hard deadline on top of fragile reporting

Marketing measured campaign and funnel performance through a set of Tableau dashboards fed by custom SQL running against Amazon Redshift. Over time, the reporting had grown into twelve similar but inconsistent datasets built from multiple complex queries over Salesforce objects. They produced different answers for the same question, were slow to debug and update, and had eroded the team's confidence in the numbers.

Then the warehouse itself became the constraint. The organization was moving from Redshift to Snowflake on a fixed shutdown date, after which every marketing dashboard would stop receiving data. Parts of the old

warehouse were already degrading: models stalled, extracts failed, and at one point dashboards showed no new data for days.

- **Business rules were changing at the same time.** Revenue operations redefined when an opportunity counts as sales-qualified, a new partner-referral lead stream had to be kept out of marketing performance, and a separately run business line had to be excluded.
- **Change capacity was limited.** Data engineering was spread thin and moving to two-week sprints, and much of the build ran through a single external consultant—so every requirement had to be unambiguous and traceable.
- **Some answers required the model.** Deduplicated MQL reporting and multi-touch campaign attribution could not be reproduced from CRM reports alone.

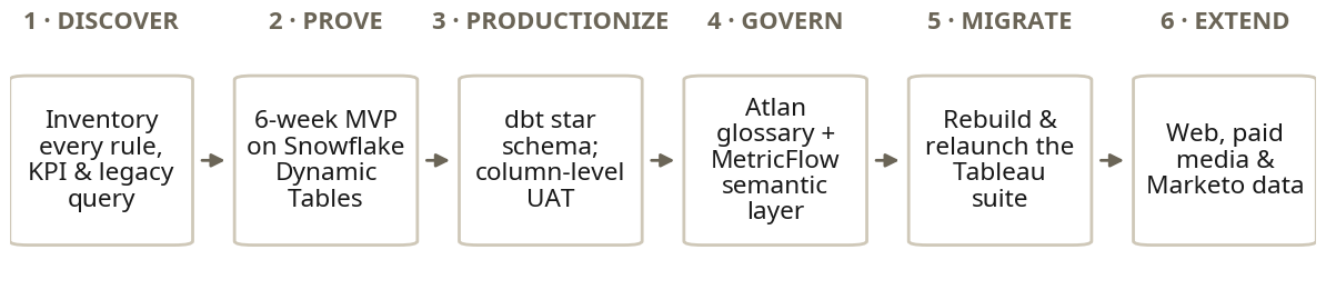
02 · STRATEGY

## Fix the definitions first, then move the data

I had built and maintained the original marketing query logic, so I started by turning it into documentation the data team could build from: every business rule, KPI, lifecycle stage and Tableau custom SQL query, including logic mistakes I had found that were excluding or miscounting campaigns. That document became the reference specification for the new model.

### Six phases from inventory to extension

Project sequence across roughly a year of foundation work and a deadline-driven migration




**01 Discover**
Inventoried business rules, KPI definitions and the legacy datasets, and mapped which dashboards depended on each.

**02 Prove**
A six-week MVP proved the approach on Snowflake Dynamic Tables with a daily target lag, connecting Tableau directly to reporting tables.

**03 Productionize**
Data engineering rebuilt the logic as a dbt project—staging, intermediate and mart layers—with column-level parity testing against the legacy Tableau SQL.

**04 Govern**
I defined and verified the business glossary in Atlan; data engineering linked it to dbt documentation and a MetricFlow semantic layer.

05 **Migrate**
I triaged, rebuilt and relaunched the Tableau dashboard suite on shared Snowflake data sources, with training and documentation.

06 **Extend**
Web analytics, paid-media and Marketo data are being brought into the same warehouse to enable full-funnel journeys.

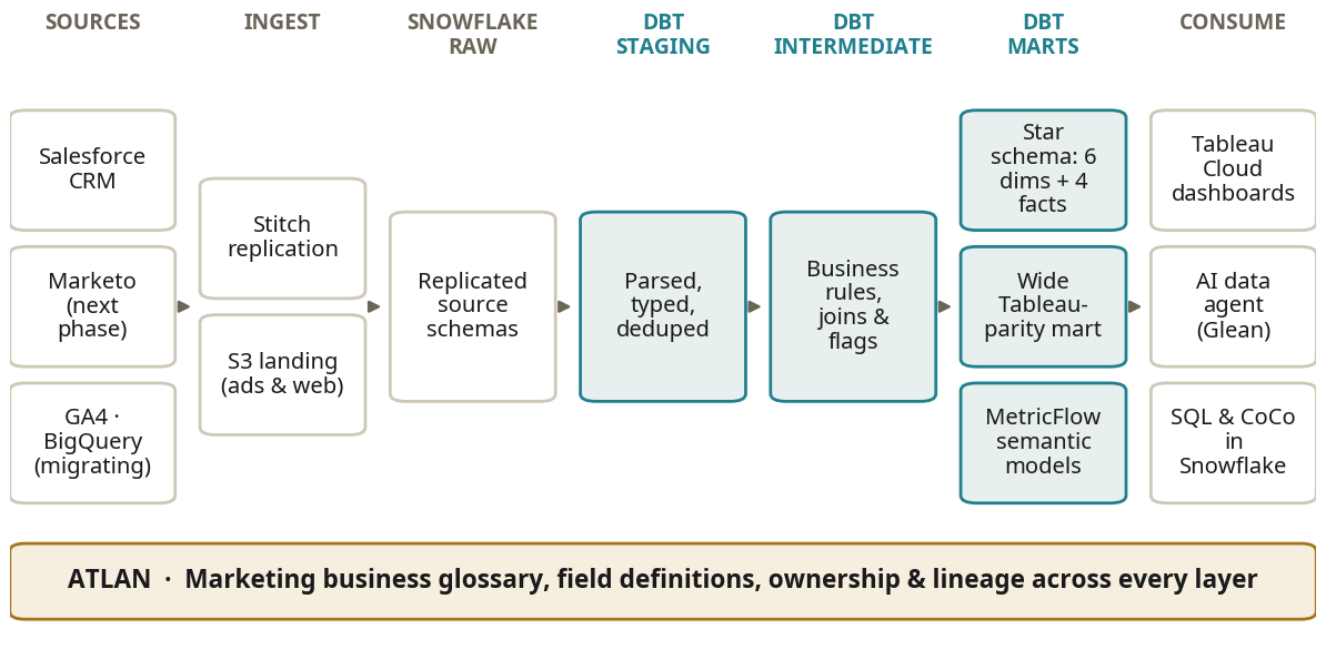
03 · ARCHITECTURE

# A layered model with business rules in one place

Salesforce data is replicated into Snowflake through Stitch, with ad-platform and web data landing through AWS S3. dbt Cloud then transforms it in three governed layers—staging parses, types and deduplicates; intermediate applies business rules, joins and flags; marts publish a star schema plus a wide reporting mart that preserves parity with the legacy dashboards. Atlan documents definitions, ownership and lineage across every layer.

### End-to-end marketing data architecture

Simplified architecture; web, paid-media and Marketo sources are in the next phase



| OBJECT | GRAIN | PURPOSE |
|---|---|---|
| fct_mql_events | One row per MQL person (lead or contact) | Deduplicated MQL volume, priority and source |
| fct_sql_opportunities | One row per opportunity | SQL, SQO and pipeline counts and flags, dated by SQL date |
| fct_campaign_members | One row per responded campaign member | Campaign reach and responses, dated by first response |
| fct_campaign_influence | Influence row × contact role × responded member | Multi-touch sourced and influenced pipeline |
| Dimensions (6) | Accounts, contacts, leads, campaigns, opportunities, calendar | Descriptors, parent–child campaign hierarchy and a fiscal-calendar time spine |
| Wide reporting mart | One row per MQL with pipeline context | Column-level parity with legacy Tableau custom SQL |

04 · ETL AND BUSINESS RULES

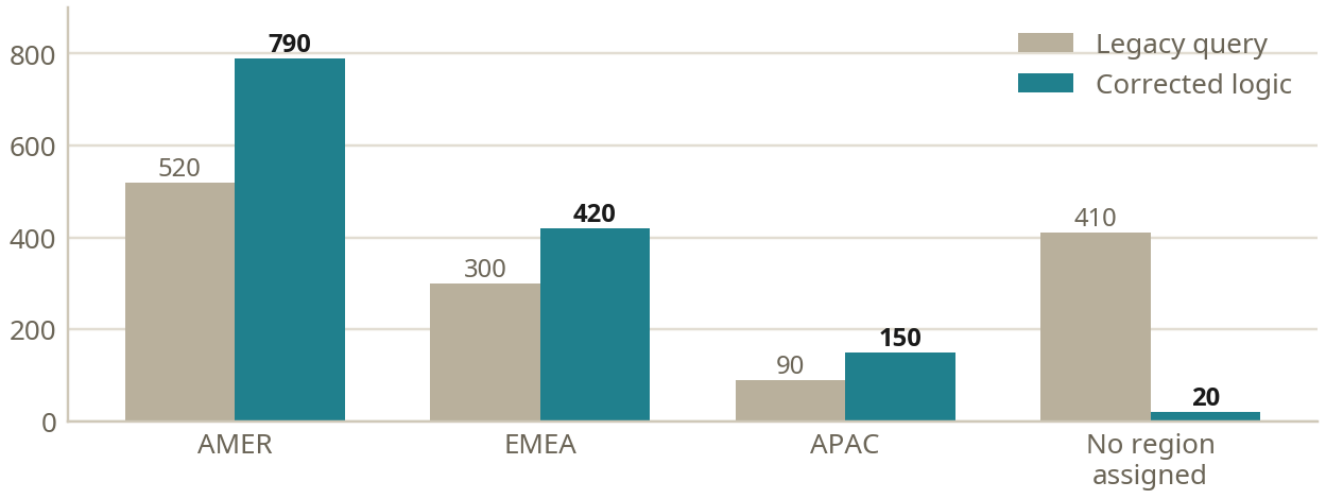
## Every rule change traced from SQL to Snowflake to Atlan

I ran rule changes through a single field-change log. Each entry captured the issue, affected data mart and field, category (business-process change, bug fix or new feature), old logic, new logic and business impact—with separate checkpoints for my dashboard SQL queries, the Snowflake model and the Atlan data dictionary. The log fed Jira tickets for the data team and ensured a rule never changed in one place but not the others.

| RULE CHANGE | TYPE | WHAT CHANGED AND WHY |
|---|---|---|
| SQL timing redefined | Business process | Sales-qualified status moved from opportunity-created date to the date SQL criteria were met, aligned with a revenue-operations process change. The created date was kept as a reference field. |
| Region assignment fix | Bug fix | A column alias in the legacy Redshift SQL collided with a physical column, so every contact MQL fell into “No region assigned.” Fixed with explicit source logic for leads and contacts. |
| Sourced vs. influenced | Bug fix | Renewal and services opportunities are grouped as influenced and never labeled marketing-sourced, reducing unassigned records. |
| Campaign target region | New field | Campaign managers plan by target region, not prospect headquarters, so a rule derives target region from the campaign naming convention. |
| Partner-referral leads | New flag | Leads referred by an external partner program are flagged and excluded so they don’t inflate marketing performance. |
| Separate business line | Exclusion | Pipeline and campaigns from an independently run business line are excluded from marketing reporting. |
| Account type | New field | Account relationship type added wherever account data appears, so marketers can see whether they reach customers, partners, prospects or free users. |

**One alias bug hid most contacts' regions**

MQLs by region before and after the region-logic fix · illustrative data




### Example: redefining a marketing-qualified SQL

The SQL timing change cascaded through every dependent field: days from MQL to SQL, the marketing-SQL flag and its 180-day attribution window, the 0–30, 30–60, 60–90 and 90–180-day influence buckets, and the WHERE clause that defines which campaign responses can influence an opportunity.

```
-- Before: legacy Redshift custom SQL, anchored on opportunity creation
CASE WHEN o.latest_mql_date IS NOT NULL
      AND DATEDIFF(day, o.latest_mql_date, o.created_date) BETWEEN 0 AND 180
     THEN TRUE ELSE FALSE END AS is_marketing_sql

-- After: Snowflake / dbt, anchored on the date SQL criteria were met
CASE WHEN o.latest_mql_date IS NOT NULL
      AND o.stage_name <> 'Consideration'
      AND DATEDIFF(day, o.latest_mql_date, o.sql_date) BETWEEN 0 AND 180
     THEN TRUE ELSE FALSE END AS is_marketing_sql
```

05 · VALIDATION

## Proving parity before anyone noticed a change

- **Count reconciliation.** I compared Redshift and Snowflake MQL, SQL and campaign counts and documented each discrepancy—missing lead records, incomplete fact-table loads and missing fields—so the data team could fix causes, not symptoms.
- **Field-level gap analysis.** A validation workbook mapped every field in my existing SQL queries to the new mart columns, exposing fields that still needed to be added.
- **Root-causing edge cases.** One gap traced to how converted leads were deduplicated: the model kept only the contact record. Teammates used CoCo, Snowflake's AI coding assistant, to trace issues like this quickly in SQL.
- **Freshness diagnostics.** When Salesforce-backed models stalled, we diagnosed it with dbt job logs and Stitch freshness fields.
- **Column-level parity.** The production model was built to match the legacy Tableau custom SQL column by column, with user acceptance testing before cut-over.

06 · ATLAN DATA DICTIONARY

## A business glossary built for people and machines

Working with data project management and data engineering, I defined the marketing business glossary in Atlan, personally verifying more than 120 fields with business definitions, rules and ownership. Each term carries a certification status, business-name aliases, the model it lives in, its kind (entity, dimension or measure), a category, a plain-language definition, business rules, the calculation and owners.

| TERM | KIND | HOW IT IS DEFINED |
|---|---|---|
| Days MQL → SQL | Measure | Days between latest MQL date and SQL date; attribution counts only opportunities where the MQL falls within 180 days before the SQL date. |
| Campaign Influence Window | Rule | A response influences an opportunity if it falls 0–180 days before qualification or while the opportunity is open. Implemented in a WHERE clause, so it defines the dataset rather than a column. |
| Campaign Influence Date | Dimension | When the CRM wrote the influence record—a processing timestamp, not a business event. Documented so no one uses it as a response date. |

Terms are organized into categories such as Audience Segmentation, Campaigns & Attribution, Demand Generation & Funnel Performance, Key Date Fields and Revenue & Opportunity Metrics. Writing down what a field is not—as with the influence date—proved as valuable as defining what it is.

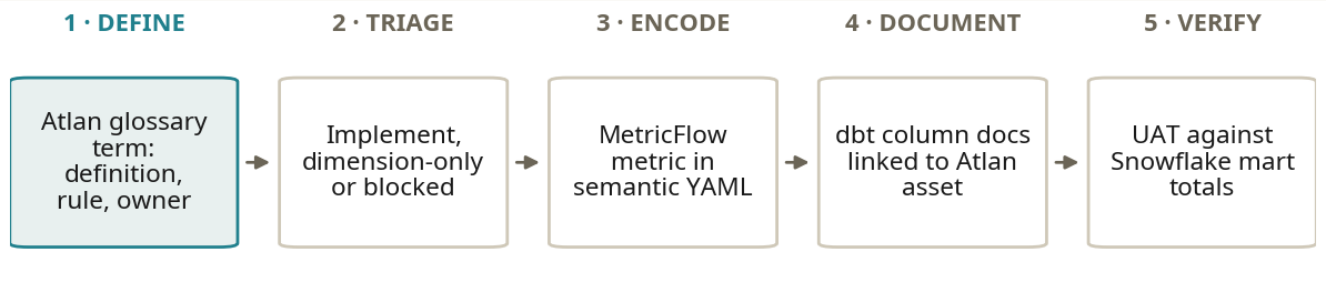
07 · AI SEMANTIC LAYER

## Definitions an AI agent can use

The glossary feeds a dbt Semantic Layer built with MetricFlow on top of the thin fact tables. Phase 1 exposes core funnel KPIs—total MQLs, marketing SQLs and SQOs, days MQL→SQL and MQL→SQO, opportunity count, pipeline dollars, deduplicated campaign pipeline and responded campaign members—with region, segment, priority and source as dimensions rather than duplicate metrics.

### From business definition to governed metric

The Atlan-to-dbt workflow used to publish each metric




A natural-language AI data agent in Glean answers marketing funnel and attribution questions using the governed dbt metrics and Atlan metadata as context, so a question like “marketing SQOs by region last quarter” resolves to the same definition as the dashboard. Metrics that still need agreed definitions—conversion-rate ratios and priority filters—were deliberately deferred rather than approximated.

08 · THE DASHBOARD SUITE

# Triaging, rebuilding and relaunching on clean data

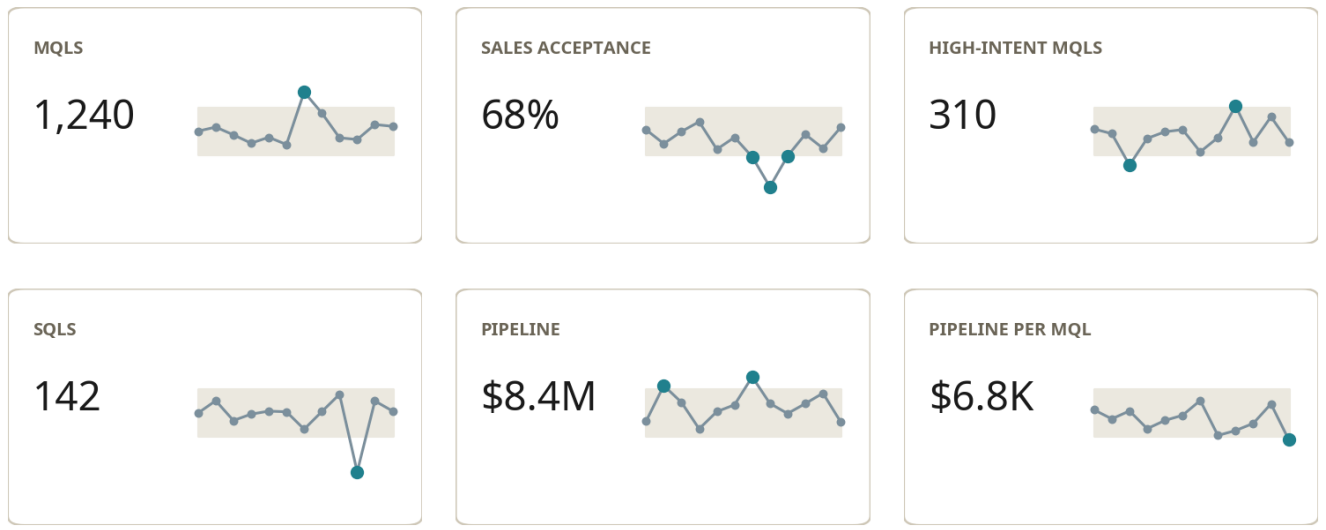
I owned the dashboard migration plan. I inventoried the legacy data sources to retire, mapped every dependent dashboard, and decided for each whether to migrate, rebuild or deprecate—weighing usage, customization and technical debt.

| DASHBOARD | DECISION | RATIONALE |
|---|---|---|
| High-intent MQL performance | Rebuild on the new datasets | Recently built, so a clean rebuild was simpler than renaming every field |
| Campaign performance | Migrate in place | Heavily used and customized; new dataset fields were named to match so sources could be swapped with minimal disruption |
| Weekly pipeline (MQLs, SQLs, SQOs) | Complete rebuild | Built on aging datasets and logic; rebuilt to better standards |
| Quarterly performance summary | Deprioritize; redesign later | No recent usage |

- **Shared, governed sources.** I created shared Snowflake data sources in Tableau Cloud with daily extract refreshes, reorganized Tableau projects and reconnected dashboards.
- **Clean retirement.** Non-migrated dashboards were marked with alert text and moved to an archive folder rather than silently disappearing.
- **Relaunch with enablement.** The Marketing Pipeline dashboards—weekly pipeline performance, MQL and SQL profiles, and campaign and source performance—launched with a training video, a quick guide, a full user guide and team announcements.
- **Operational transparency.** I documented the Salesforce → Snowflake → Tableau refresh schedule so users know when data is current.

### KPI tiles with trend context

Illustrative layout: headline values with a trend band that highlights unusual weeks · illustrative data



09 · RESULTS

# What changed

- **Twelve legacy datasets → four governed sources of truth** in Snowflake: MQLs, SQLs, campaigns with members and MQLs, and campaigns with sourced and influenced opportunities.
- **Core reporting moved off Redshift.** Campaign datasets and their dashboards migrated and became operational; MQL and SQL datasets were migrated and validated; the rebuilt pipeline dashboard suite launched shortly after the shutdown.
- **Known bugs fixed in the move,** including region misassignment for contacts and sourced-versus-influenced labeling.
- **A shared vocabulary in Atlan** with more than 120 verified fields, linked to dbt documentation and a governed MetricFlow semantic layer.
- **AI-ready analytics.** A natural-language data agent can query marketing metrics using the same certified definitions as Tableau.

10 · REFLECTION

# What I would carry forward

- **A migration is the cheapest time to fix definitions.** Moving broken logic faithfully would have preserved the bugs; documenting rules first let us fix them once.
- **Trace every change in three places.** A single log tracking dashboard SQL, warehouse model and data dictionary prevented definitions from drifting apart.
- **Write requirements as if the builder has no context.** With a consultant and a sprint-based team, precise old-logic/new-logic specifications kept work moving.
- **Govern for machines as well as people.** Certified definitions in Atlan made the semantic layer and AI agent trustworthy, and deferring undefined metrics avoided confident wrong answers.

SKILLS DEMONSTRATED

Data warehouse migration · Dimensional modeling · ETL & business-rule design · SQL · Parity testing & UAT · Data governance (Atlan) · Semantic layer design · Tableau dashboard development · Change management & enablement