

Semantic Layer Starter Kit

Define revenue once, serve it to dashboards and AI agents. Copy, adapt, ship.

Step 1 — Define it with a human

Name: revenue. Definition: gross revenue, refunds excluded, recognized on order date. Grain: one row per day per region. Owner: named and accountable.

Step 2 — Model it (dbt MetricFlow)

```
semantic_models:

  - name: orders

    defaults:

      agg_time_dimension: ordered_at

    entities:

      - name: order_id

        type: primary

    dimensions:

      - name: ordered_at

        type: TIME

        type_params:

          time_granularity: day

      - name: region

        type: CATEGORICAL

    measures:

      - name: order_total

        agg: SUM

        expr: line_total

    metrics:

      - name: revenue

        description: Gross revenue, refunds excluded,
```

```
    recognized on order date

    type: SIMPLE

    type_params:

    measure: order_total
```

No dbt? A SQL view gets you 80% there:

```
CREATE OR REPLACE VIEW marts.revenue_daily AS

SELECT

    DATE_TRUNC('day', ordered_at) AS day,

    region,

    SUM(line_total) AS revenue

FROM orders

WHERE order_status != 'refunded'

GROUP BY 1, 2;
```

Step 3 — Test it like production code

Sanity bounds (never negative), grain checks (no dup day/region, no future dates), and scheduled reconciliation against finance.

Step 4 — Serve it to everyone, including agents

BI connects natively; agents get the same metrics by name through an API — they read metrics, never compute them.

Step 5 — Govern changes

Version definitions, require owner sign-off, deprecate with notice. Agents break silently on unversioned changes.