
RevPipeline360

Microsoft Fabric Lakehouse Build Architecture

Author	Anthony Orji
Role	Principal BI & Systems Analyst
Stack	Microsoft Fabric · PySpark · Direct Lake · Power BI · DAX
Workspace	WS_RevPipeline360_Lab
Version	1.0 · 2026

This document describes the end-to-end architecture of RevPipeline360 — a Microsoft Fabric analytics solution built on a Bronze/Silver/Gold medallion lakehouse pattern. It covers data ingestion, notebook-driven ELT transforms, pipeline orchestration, semantic model design, DAX measures, and Power BI reporting.

Executive Summary

RevPipeline360 is an end-to-end Microsoft Fabric analytics solution built on a Lakehouse pattern. Source CSV files are ingested from OneDrive into Bronze tables, cleansed and typed into Silver via a PySpark notebook, then transformed into a curated Gold star schema via a second PySpark notebook. A Data Pipeline orchestrates the Silver and Gold notebook activities in sequence. A Direct Lake semantic model connects directly to the Gold layer, with a full DAX measure library powering a Power BI Revenue Intelligence report.

Key Implementation Highlights

- **Notebook-driven ELT** — all transforms run inside Fabric via PySpark using CREATE OR REPLACE TABLE for idempotent rebuilds
- **Pipeline hardening** — retries and timeouts configured to handle Spark capacity and session throttling
- **Semantic model relationship design** — inactive relationships with USERRELATIONSHIP/TREATAS where multiple date paths exist
- **Campaign date-key enhancement** — start_date_key and end_date_key added to gold_dim_campaigns to support fiscal year slicing

High-Level Architecture

Layer	Component	Description
Source	OneDrive folder	CSV files — Salesforce, Finance, HR sample tables
Bronze	dbo.bronze_*	Raw landing tables, 1:1 with source files, loaded via Dataflow Gen2
Silver	dbo.silver_*	Cleaned, typed, and standardized via NB_Silver_Transform notebook
Gold — Dims	dbo.gold_dim_*	Curated dimensions: date, customers, products, users, campaigns
Gold — Facts	dbo.gold_fact_*	Curated facts: opportunity, leads, invoices, GL, budget, headcount
Pipeline	PL_RevPipeline360_Refresh	Run_NB_Silver_Transform → (On Success) → Run_NB_Gold_Transform
Semantic Model	SM_RevPipeline360	Direct Lake — relationships, inactive paths, DAX measures
Report	RevDashboardV4	Revenue Intelligence: Campaign to Revenue — FY slicer, KPIs, funnel

Data Assets and Schema

Gold Dimensions

Table	Key Columns
<code>gold_dim_date</code>	date_key (PK), calendar_date, year, month, quarter, week, fiscal_year, fiscal_month
<code>gold_dim_customers</code>	master_customer_id (PK), customer_name, industry, region, active_flag
<code>gold_dim_products</code>	master_sku (PK), product_name, product_family, category, uom
<code>gold_dim_users</code>	salesrep_id (PK), rep_name, manager, territory, region
<code>gold_dim_campaigns</code>	campaign_id (PK), campaign_name, channel, start_date, start_date_key, end_date, end_date_key

Gold Facts

Table	Key Columns / Description
<code>gold_fact_opportunity</code>	opportunity_id, customer_id, salesrep_id, is_won, amount, close_date_key, created_date_key
<code>gold_fact_opportunity_line</code>	opportunity_line_id, opportunity_id, product_sku, quantity, unit_price, discount_pct, line_amount
<code>gold_fact_leads</code>	lead_id, campaign_id, lead_source, created_date_key, converted_to_opportunity_id
<code>gold_fact_campaigns</code>	campaign_id, campaign_name, channel, start_date_key, end_date_key
<code>gold_fact_invoice_header</code>	invoice_id, opportunity_id, customer_id, invoice_date_key, due_date_key, status
<code>gold_fact_invoice_line</code>	invoice_id, product_sku, quantity, unit_price, discount_pct, line_amount, invoice_date_key
<code>gold_fact_gl_postings</code>	gl_id, posting_date_key, customer_id, account_code, account_type, amount
<code>gold_fact_budget_monthly</code>	month_date_key, region, budget_revenue, budget_cogs, budget_gross_profit, budget_gm_pct
<code>gold_fact_headcount_monthly</code>	month_date_key, region, headcount, hires, attrition, attrition_rate

Notebook Architecture

NB_Silver_Transform

Cleanses and standardizes all Bronze tables into Silver. Each table is rebuilt using CREATE OR REPLACE TABLE — ensuring idempotent execution on every pipeline run. Transformations include TRIM on all string columns, CAST for type enforcement, NULLIF to normalize empty strings to NULL, and TO_DATE for date standardization. Row count validation is applied after each table build.

Example — silver_customers:

```
spark.sql("""
CREATE OR REPLACE TABLE dbo.silver_customers AS SELECT
TRIM(master_customer_id) AS master_customer_id,
TRIM(customer_name) AS customer_name,
NULLIF(TRIM(parent_customer_id),'') AS parent_customer_id,
TRIM(industry) AS industry,
TRIM(region) AS region,
TRIM(state) AS state,
CAST(active_flag AS INT) AS active_flag
FROM dbo.bronze_customers
""")
```

NB_Gold_Transform

Builds the curated Gold star schema from Silver tables. All Gold dimensions are built first, then Gold facts are constructed with enrichment joins to dimensions. Date keys are generated by joining to gold_dim_date on the calendar date. The full statement list runs sequentially via a Python loop.

Pattern — gold_dim_campaigns (with dual date keys):

```
CREATE OR REPLACE TABLE dbo.gold_dim_campaigns AS SELECT
TRIM(sc.CampaignId) AS campaign_id,
TRIM(sc.CampaignName) AS campaign_name,
TRIM(sc.Channel) AS channel,
TO_DATE(...) AS start_date,
ds.date_key AS start_date_key,
TO_DATE(...) AS end_date,
de.date_key AS end_date_key
FROM dbo.silver_sf_campaigns sc
LEFT JOIN dbo.gold_dim_date ds ON start_date = ds.date
```

```
LEFT JOIN dbo.gold_dim_date de ON end_date = de.date
```

The dual date-key pattern on gold_dim_campaigns enables fiscal year overlap slicing without a bridge table, supporting the Active Campaigns (FY) DAX measure.

Pipeline Orchestration

The pipeline `PL_RevPipeline360_Refresh` in workspace `WS_RevPipeline360_Lab` contains two notebook activities chained in sequence:

- **Run_NB_Silver_Transform** — executes the Silver cleansing notebook
- **Run_NB_Gold_Transform** — executes on success of the Silver activity

Governance and Hardening

- Retry policies and timeout settings are configured on both notebook activities
- Retry logic mitigates intermittent Spark/Livy session creation failures (HTTP 430 TooManyRequestsForCapacity) on shared capacity
- Interactive notebook sessions should not run in parallel with pipeline execution to avoid session conflicts
- Initial pipeline runs may take longer due to compute cold-start on shared Fabric capacity

Semantic Model — Relationships

SM_RevPipeline360 uses Direct Lake mode, connecting directly to Gold tables in the lakehouse. Relationships follow a standard star schema with deliberate handling of ambiguous multi-path scenarios.

Active Relationships

From Table (column)	To Table (column)	Cardinality
gold_fact_opportunity (customer_id)	gold_dim_customers (master_customer_id)	* → 1
gold_fact_opportunity (close_date_key)	gold_dim_date (date_key)	* → 1
gold_fact_opportunity (salesrep_id)	gold_dim_users (salesrep_id)	* → 1
gold_fact_opportunity_line (opportunity_id)	gold_fact_opportunity (opportunity_id)	* → 1
gold_fact_opportunity_line (product_sku)	gold_dim_products (master_sku)	* → 1
gold_fact_leads (campaign_id)	gold_dim_campaigns (campaign_id)	* → 1
gold_fact_leads (created_date_key)	gold_dim_date (date_key)	* → 1
gold_fact_invoice_line (invoice_date_key)	gold_dim_date (date_key)	* → 1
gold_fact_invoice_line (product_sku)	gold_dim_products (master_sku)	* → 1
gold_fact_gl_postings (posting_date_key)	gold_dim_date (date_key)	* → 1
gold_fact_budget_monthly (month_date_key)	gold_dim_date (date_key)	* → 1
gold_fact_headcount_monthly (month_date_key)	gold_dim_date (date_key)	* → 1

Inactive Relationships and Resolution

- **gold_fact_opportunity (created_date_key)** — inactive; activated via USERRELATIONSHIP in time-intelligence measures
- **gold_fact_invoice_header (due_date_key)** — inactive; primary path uses invoice_date_key
- **Campaign start/end date keys** — no active date relationship to campaigns dim; fiscal overlap handled in DAX via FY boundary variables
- **Lead to Opportunity attribution** — no direct campaign_id on gold_fact_opportunity; TREATAS projects converted opportunity IDs from lead data for channel-attributed pipeline measures

DAX Measures

All measures are organized under a dedicated MyMeasures table in the semantic model.

Lead & Conversion Measures

Measure	Formula / Logic
Leads	<code>COUNTROWS (gold_fact_leads)</code>
Converted Leads	<code>COUNTROWS (FILTER (gold_fact_leads, NOT ISBLANK (gold_fact_leads[converted_to_oppo</code>
Lead Conversion Rate	<code>DIVIDE ([Converted Leads], [Leads])</code>

Opportunity Measures

Measure	Formula / Logic
Opportunities	<code>COUNTROWS (gold_fact_opportunity)</code>
Won Opportunities	<code>CALCULATE ([Opportunities], gold_fact_opportunity[is_won] = 1)</code>
Win Rate	<code>DIVIDE ([Won Opportunities], [Opportunities], 0)</code>
Pipeline Amount	<code>SUM (gold_fact_opportunity[amount])</code>
Won Amount	<code>CALCULATE ([Pipeline Amount], gold_fact_opportunity[is_won] = 1)</code>
Avg Deal Size (Won)	<code>DIVIDE ([Won Amount], [Won Opportunities], 0)</code>

Attribution Measures

Measure	Formula / Logic
Pipeline Amount (By Channel)	<code>VAR OppIds = FILTER (VALUES (gold_fact_leads[converted_to_opportunity_id]), NOT</code>
Pipeline Amount (Attributed to Leads)	<code>CALCULATE (SUM (gold_fact_opportunity[amount]), USERRELATIONSHIP (gold_fact_op</code>

Campaign Measures

Measure	Formula / Logic
Campaigns (Distinct)	<code>DISTINCTCOUNT (gold_dim_campaigns[campaign_id])</code>
Active Campaigns (FY)	<code>VAR SelectedFY = SELECTEDVALUE(...) - fiscal year boundary overlap filter across</code>

Modeling Constraints & Resolutions

No campaign_id on gold_fact_opportunity

Issue: The opportunity fact does not carry a campaign foreign key. Campaign/channel slicers cannot directly segment opportunity-based pipeline.

Resolution: Lead-based attribution via converted_to_opportunity_id. TREATAS and USERRELATIONSHIP measures compute channel-attributed pipeline for marketing analytics views.

Ambiguous relationship paths through Dim Date

Issue: Multiple fact tables have two or more date foreign keys pointing to gold_dim_date, creating ambiguous filter paths.

Resolution: One active relationship per fact table (primary analytical date). Alternate date roles set inactive and activated via USERRELATIONSHIP in specific measures.

Spark session throttling during pipeline runs

Issue: Gold notebook activity intermittently failed with Livy session creation errors (HTTP 430 TooManyRequestsForCapacity) on shared Fabric capacity.

Resolution: Retry policies and timeout settings on pipeline activities. Avoid overlapping interactive notebook sessions during scheduled runs. Optional wait step between activities.

Power BI Report — Revenue Intelligence

RevDashboardV4 is the primary report connected to SM_RevPipeline360 via Direct Lake. The Campaign to Revenue page delivers a full lead-to-revenue funnel view with fiscal year slicing.

KPI Cards (top row)

KPI	Value	Measure
All Leads	1,200	[Leads]
Lead Conversion Rate	27%	[Lead Conversion Rate]
Revenue (Won from Leads)	\$18,226	[Won Amount]
Win Rate (Lead-Sourced)	82%	[Win Rate]
Pipeline Amount (Lead-Sourced)	\$78,445	[Pipeline Amount]

Visuals

- **Leads by Lead Source** — horizontal bar chart; Partner, Paid Social, Trade Show, Web, Email channels
- **Lead-to-Revenue Funnel** — stacked bar; Leads → Converted Leads → Won Opps
- **Leads by Fiscal Year** — line chart; 2023→2024→2025 trend
- **Top 5 Campaigns by Leads** — horizontal bar chart sorted descending
- **Campaign Effectiveness** — scatter plot; Lead Source x conversion rate x lead volume bubble size
- **Fiscal Year Slicer** — button slicer filtering all visuals by FY 2023 / 2024 / 2025