



Semantic View

Unlocking Efficient and Trusted AI-powered BI

Tian Gao, Evelyn Li
March 13, 2026

Agenda

- > Why Semantic Models?
- > Snowflake Solution: Native Semantic View
 - Concepts
 - Interfaces
 - Benefits
 - Ecosystems
- > Demo



Why Semantic Models?



Why is a semantic layer needed?

1. BI: Business users shouldn't need to be data engineers. The semantic layer transforms complex physical schemas into intuitive business terms, enabling true self-service without technical friction.
2. AI: LLMs understand SQL syntax but lack your specific business context. A semantic layer provides the deterministic map needed to translate natural language into accurate queries, preventing hallucinations and logic errors.



Challenges without a Semantic Model

1. Concept Mismatch

- Business questions refer to concepts that don't map to a single physical column (“NRR”, “customer health score”, etc.). Everyone reinvents the logic without an explicit definition.

2. Metric Drift

- Sample queries can go stale and one team's definition for a metric quietly diverges from another's. Nobody notices until the board deck has two conflicting numbers.

3. Join Ambiguity

- Text-to-SQL systems could return confidently wrong answer when multiple join paths exist between tables, as the small differences can change the grain and the totals.

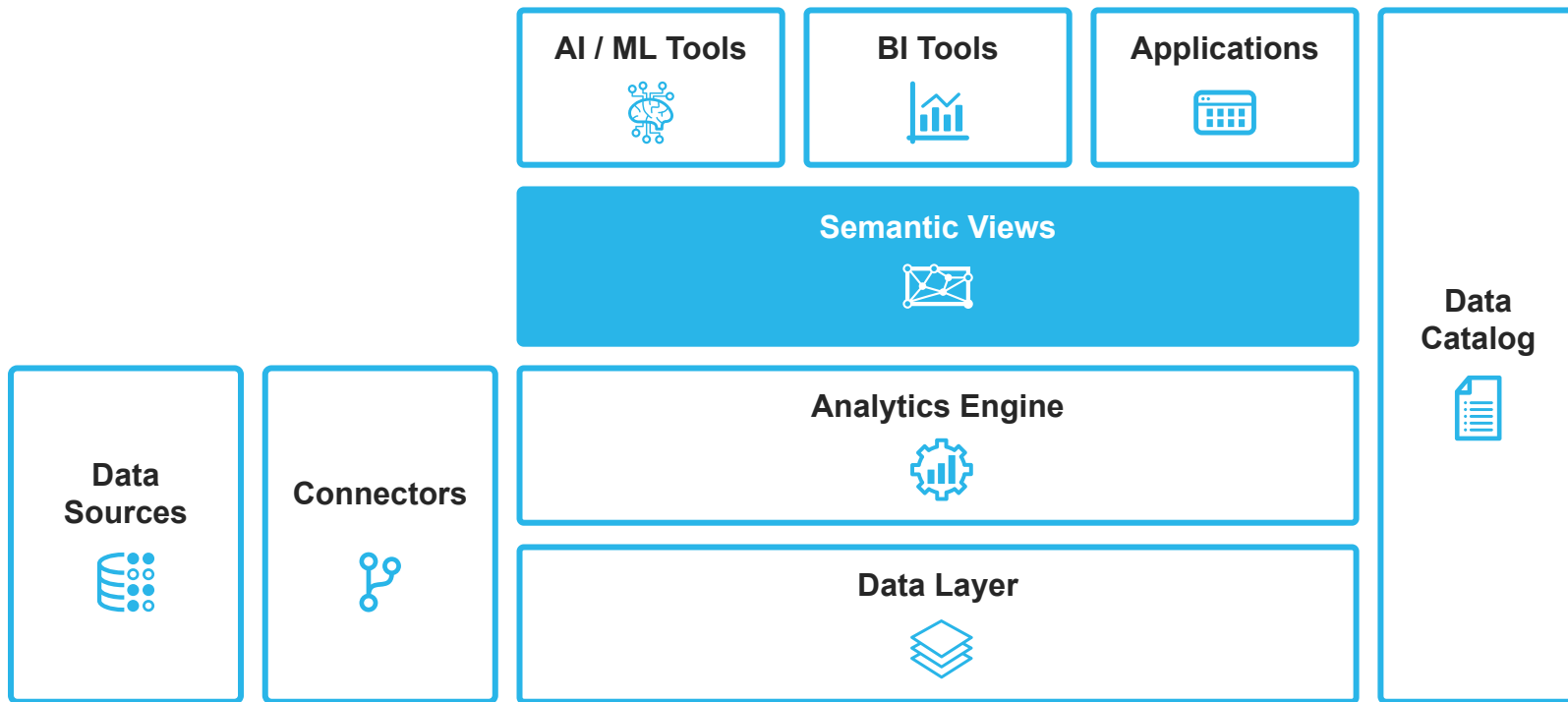
4. Tool Duplication

- Each BI tool and AI agent independently learns meaning from the same raw schema. The same business logic needs maintaining in different places.



Snowflake Solution: Native Semantic View

Semantic View: Universal Interface for Analytics



Semantic View Concepts

Entities

Real-world objects and concepts.

Example: Customer, Order

Relationships

How different entities are related to each other.

Example: customer ↔ order relation

Calculations

Used to analyze entities and relations, support cross-table calculations.

There are 3 kinds of calculations: Metrics, Dimensions, Facts

Metrics

Aggregations that answer business questions.

Example: Revenue means Gross Revenue minus Returns.

Dimensions

Attributes related to entities.

Example: Sale date, customer state

Facts

Non-aggregate calculations on the entity

Example: Sale profit



Semantic View SQL Interface: CREATE

```
CREATE SEMANTIC VIEW Customer_360_View
TABLES (
  // logical table name      AS definition
  region                    AS db.schema.region    PRIMARY KEY (regionkey),
  nation                    AS db.schema.nation    PRIMARY KEY (nationkey),
  customer                  AS db.schema.customer  PRIMARY KEY (custkey)
)
RELATIONSHIPS (
  customer (nationkey) REFERENCES nation,
  nation   (regionkey) REFERENCES region
)
DIMENSIONS (
  // dimension name        AS dimension expression
  region.region_dim       AS region.name COMMENT = 'unique name for each region'
)
METRICS (
  // metric name           AS metric expression
  customer.customer_count AS COUNT(custkey) SYNONYMS = ('num of customers', 'total customers')
)
WITH EXTENSION (
  CA = 'Cortex Analyst data: verified queries, etc (JSON encoded)'
)
```



Semantic View SQL Interface: SHOW, DESCRIBE

SHOW SEMANTIC VIEWS;

NAME	DATABASE_NAME	SCHEMA_NAME	COMMENT	OWNER	EXTENSION
CUSTOMER_360_VIEW	DB_360	PUBLIC		SYSADMIN	["CA"]

DESCRIBE SEMANTIC VIEW Customer_360_View;

OBJECT_KIND	OBJECT_NAME	PARENT_ENTITY	PROPERTY	PROPERTY_VALUE
TABLE	CUSTOMER		BASE_TABLE_DATABASE_NAME	DB_360
TABLE	CUSTOMER		BASE_TABLE_SCHEMA_NAME	PUBLIC
TABLE	CUSTOMER		BASE_TABLE_NAME	CUSTOMER
TABLE	CUSTOMER		PRIMARY_KEY	["CUSTKEY"]
... information about other tables inside the semantic view: REGION, NATION ...				
METRIC	CUSTOMER_COUNT	CUSTOMER	EXPRESSION	COUNT(customer.custkey)
DIMENSION	REGION_DIM	REGION	EXPRESSION	region.name
EXTENSION	CA		VALUE	Cortex Analyst data (JSON encoded)



Semantic View SQL Interface: SELECT

```
// Compute customer count by region
// Form 1: SEMANTIC_VIEW() from clause extension
```

```
SELECT * FROM SEMANTIC_VIEW(
  Customer_360_View
  DIMENSIONS region.region_dim
  METRICS customer.customer_count
);
```

```
// Compute customer count by region
// Form 2: Standard SQL
```

```
SELECT region_dim, AGG(customer_count)
FROM Customer_360_View
GROUP BY region_dim
;
```

REGION_DIM	CUSTOMER_COUNT
AFRICA	29764
AMERICA	29952
ASIA	30183
EUROPE	30197
MIDDLE EAST	29904



Benefits of Native Schema Object

1. Unified Security and Governance
 - Object-level RBAC
 - Replication and DR capabilities
 - Row-access policies and masking policies
2. Strong Validation
 - Validity of base objects and identifiers
 - Relationship constraints
 - Calculation reference rules



Cortex Analyst for Conversational Analytics

Logical tables 18

CATALOG_SALES Edit

- Dimensions 1
- Time Dimensions 0
- Facts 0
- Named Filters 0
- Metrics 6

C_NET_PROFIT Edit ...
Sum of net profit through catalog sales
`COALESCE(SUM(cs_net_profit), 0)`

CATALOG_TOTAL_QUANTITY_SOLD Edit ...
Sum of number of items sold through catalog sales
`COALESCE(SUM(cs_quantity), 0)`

END_DATE Edit ...
Max date for catalog sales
`MAX(date_dim_d_date)`

START_DATE Edit ...
Min date for catalog sales
`MIN(date_dim_d_date)`

TOTAL_SALES Edit ...
Sum of revenue (sales price multiplied by quantity) from catalog sales
`sum(CAST(cs_sales_price * cs_quantity AS decimal(38,2)))`

UNIQUE_CATALOG_CUSTOMERS Edit ...
Unique customers who made a purchase through catalog sales
`COUNT(DISTINCT cs_bill_customer_sk)`

View 4 warnings Edit YAML

TPCDS_UNIFIED

User
Explain the dataset

Cortex Analyst Request ID

This semantic data model contains comprehensive retail sales data from a multi-channel business. It includes information about sales transactions across three channels: physical stores, online web sales, and catalog sales. The dataset tracks customer demographics, product details including brands and categories, inventory levels across warehouses, store information, shipping methods, and returns. You can analyze sales performance, customer behavior, inventory management, profitability across different channels, geographic trends, and product performance. The data supports questions about revenue, profit margins, customer segmentation, seasonal trends, and operational metrics across the entire retail operation.

What is the total revenue from store sales by product category?

How many unique customers made catalog purchases last year?

What is the average net profit per item sold through web sales by shipping mode type?

1 warning Response metadata

User
What is the total revenue from store sales by product category?

Cortex Analyst Request ID

This is our interpretation of your question:
What is the total revenue from store sales by product category over the entire available time period?

Semantic query

```
SELECT
FROM
SEMANTIC_VIEW(
  TEMP.TGA0.TPCDS_UNIFIED_METRICS store_sales.total_sales DIMENSIONS i_category
) -- Generated by Cortex Analyst {request_id: e600efa2-fa62-4048-ac14-e6bec3c1a3}
```

Enter prompt

Run



Semantic View Autopilot (SVA)

Create “First Pass” Semantic View leveraging existing assets, like BI tool metadata, SQL queries, or even documentation.

The screenshot shows the 'Create Semantic Model' dialog in Cortex Analyst. The 'Provide context' tab is selected, showing several optional sources for context: Tableau Data Source, SQL Queries, Snowflake Dashboards (marked 'Coming soon'), SQL Worksheets (marked 'Coming soon'), and Custom Instructions. On the right, the 'Select Tableau Data Sources' section shows a search for 'music_fest.tds' and 'music_fest_2.tds', with the first result selected. Navigation buttons at the bottom include 'Previous: Getting started' and 'Next: Select tables'.

Your semantic model just got smarter. See what's new.
Your semantic model has been optimized using the verified queries you submitted. Here's a summary of the improvements we made.

- 93%** overall SQL accuracy (up from 63% before)
We measured accuracy on 27 verified queries after temporarily removing those exact SQL snippets from your model - so this number reflects real gains in descriptions, relationships, and synonyms rather than rote memorization.
- 189** lines of YAML added (~5 hours of YAML writing you didn't have to do)
41 descriptions, 123 sample values, 58 synonyms and 3 custom instructions.

[See how we did this](#)

Improvement examples
Here are some of your queries which have improved. See how your model now handles rock sales, top-priced songs and genre revenue questions correctly, even with verified queries removed from the Semantic Model.

Question
What are the total sales for rock music?

Cortex Analyst

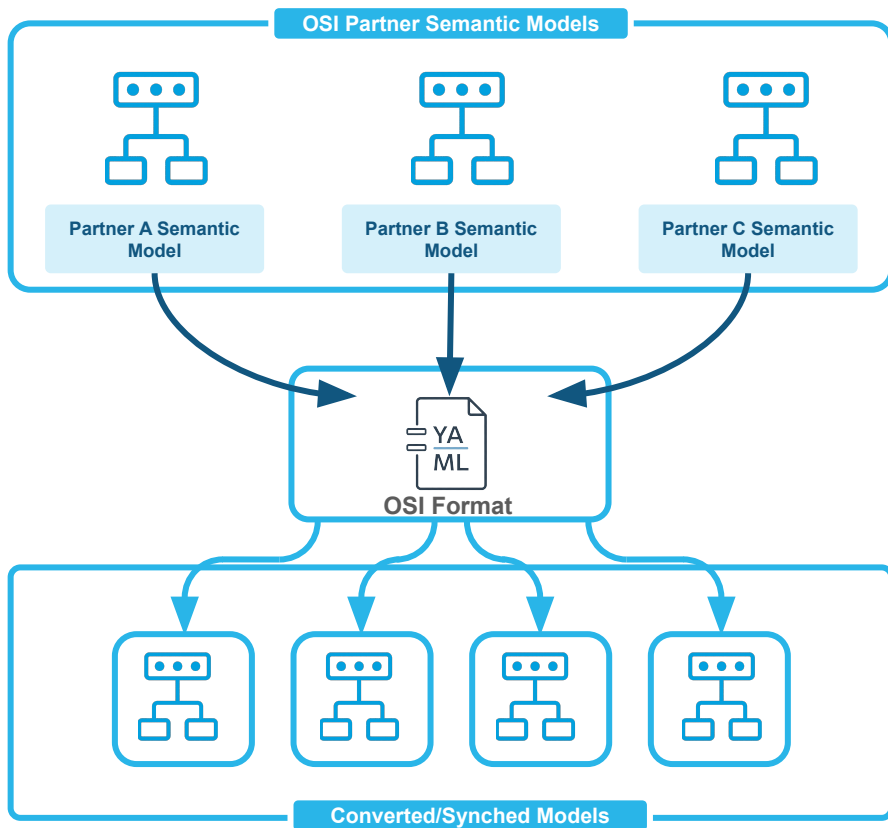
-- No SQL generated

```
SELECT
SUM(IL.UnitPrice * IL.Quantity) AS TotalSales
FROM
Genre G
JOIN Track T ON G.GenreId = T.GenreId
JOIN InvoiceLine IL ON T.TrackId = IL.TrackId
WHERE
G.Name = 'Rock'
```

[Go to optimized Semantic View](#)

SVA will execute a background “optimization” step by looking at Snowflake query history and enriching the model with additional metrics, dimensions, verified queries, and custom instructions.

Open Semantic Interchange (OSI)



A new open standard to:

- provide a universal "language" for semantic metadata
- standardize how business logic and data definitions are exchanged between different tools and platforms

Resources

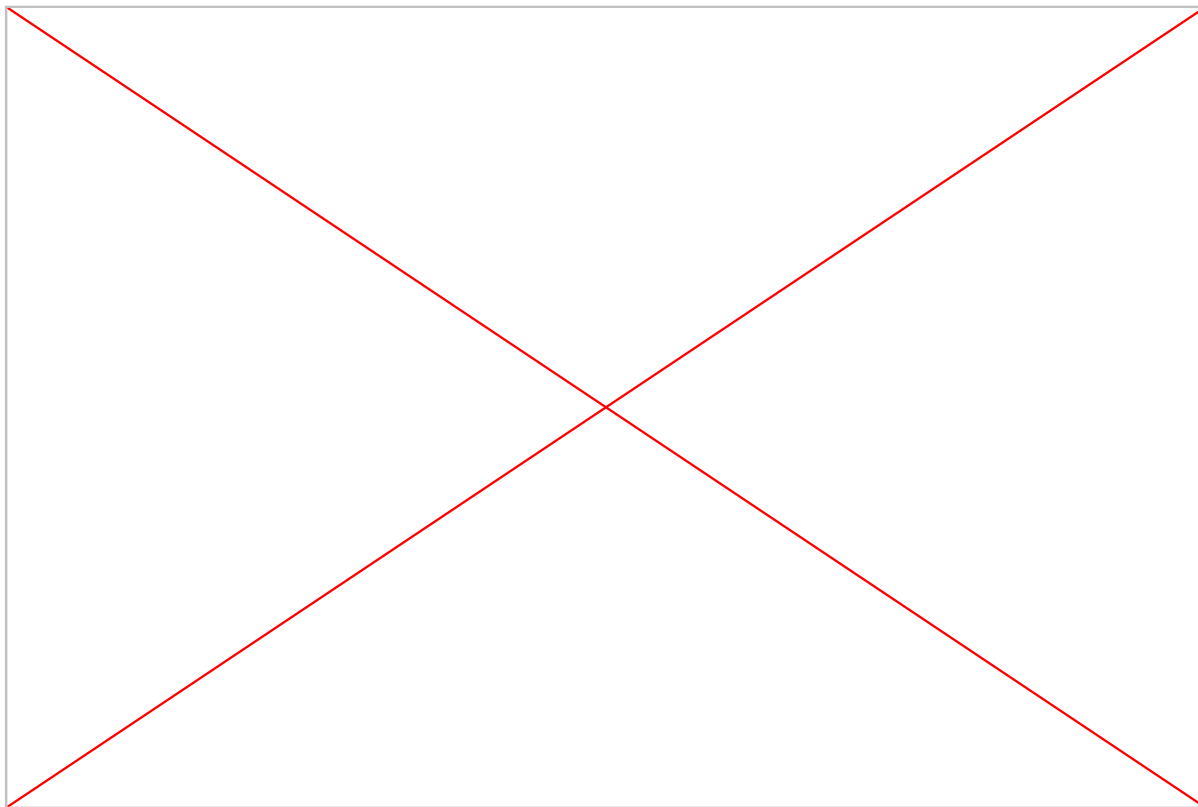
- [Documentation: Semantic View](#)
- [Documentation: Snowflake Intelligence](#)
- [Blog: Snowflake's Native Semantic Views](#)
- [Blog: How Semantic Views Avoid Subtle Mistakes in Complex Calculations](#)



Demo



Demo



Thank you!

