From dbt to SQLMesh:
Enhancing OMOP CDM Data Conversion Efficiency

Siriraj Informatics and Data Innovation Center

Presenter: Nongnaphat Wongpiyachai
Email: nongnaphat.won@mahidol.edu
Chinapat Onprasert, Sornchai Manosorn, Natthawut Adulyanukosol

1. Background

- Converting to the OMOP Common Data Model (CDM) requires a robust ETL pipeline to standardize diverse data sources.
- While widely used for OMOP CDM conversions, also showcased by Siriraj Hospital at the OHDSI Global Symposium 2022 [1], *dbt faces challenges* with data consistency and collaboration in large-scale transformations.
- **"SQLMesh"**, an open-source tool developed by Tobiko Data, Inc., offers a more efficient and reliable solution for managing the OMOP CDM conversion pipeline, addressing key limitations of dbt.

2. Methods

Siriraj Hospital has transitioned to SQLMesh, enhancing the efficiency and reliability of its data transformation processes (see Figure 1).

1. Pipeline Overview: Relevant data is pulled from hospital's diverse databases, incorporating OHDSI standardized vocabularies and DDLs necessary for transformation.

2. Mapping and Transformation: SQLMesh simplifies this process by allowing the separation of plans within the pipeline to target specific tasks.

3. Quality Assurance and Validation: Rigorous checks ensure accuracy and consistency, version control and tracking enhance reproducibility and traceability.

4. Deployment: Once data passes quality checks, it is integrated into the production environment for research and analysis. SQLMesh orchestrates the entire process efficiently.

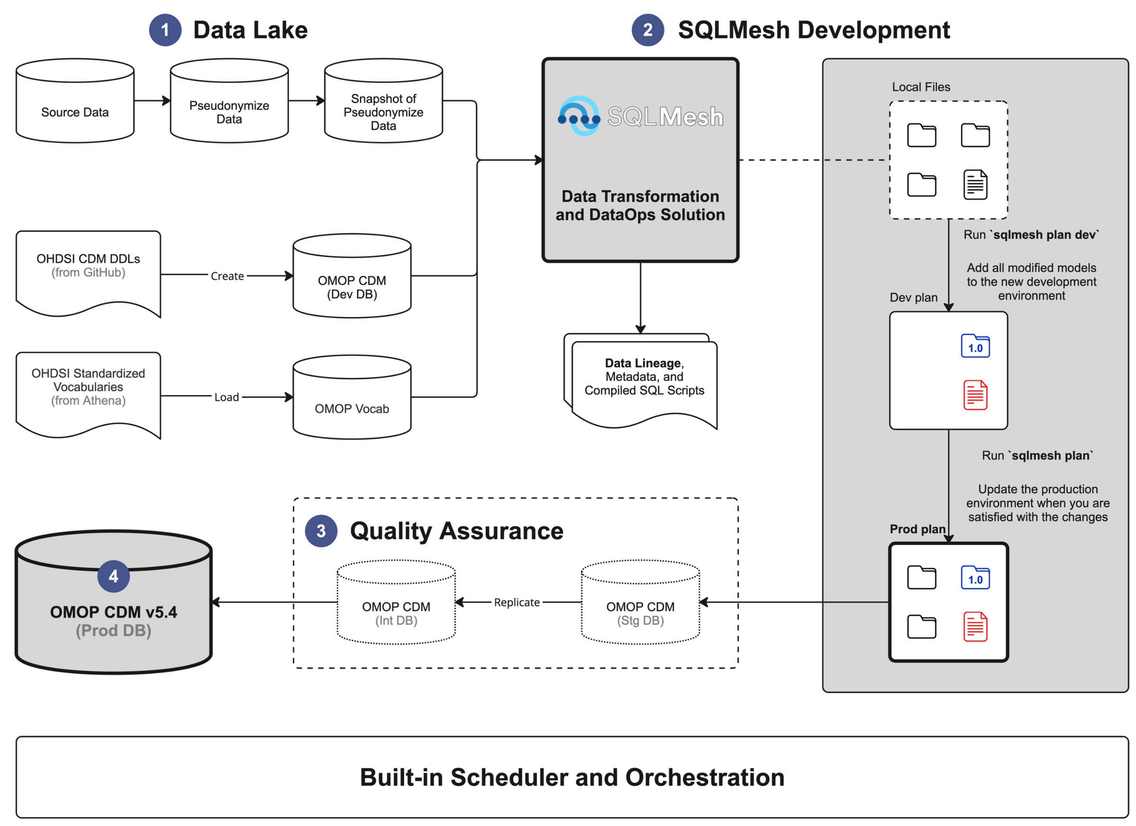

Figure 1: Overview of the OMOP CDM Conversion Pipeline at Siriraj Hospital

3. Results

1. SQL Transpilation: SQLMesh uses SQLGlot [6] to parse SQL queries, detect syntax errors at compile time, and optimize performance. It enables query reuse across multiple database dialects.

2. Column-Level Lineage: Tracks data flow and transformations at the column level, enhancing traceability.

3. Environment Management: Generates isolated schemas for data transformations, ensuring consistency before deployment. Supports multiple testing and development environments.

4. Incremental Loading: Loads only new or updated data, reducing processing time and cost for large datasets.

5. Data Validation: Offers tools for audits and unit tests to ensure accuracy and block flawed data from production.

6. Versioning: Tracks data changes, supports rollbacks, and ensures traceability alongside source code version control.

Feature	Custom scripts SQL/Python	dbt	SQLMesh
Language	SQL, Python	SQL, Jinja	SQL, Python, Jinja
Platform Support	Any	Multiple databases	Multiple databases with SQL transpilation
Data Lineage	Manual tracking	Table-level	Column-level
Change Management	Manual	Schema contracts	Automatic schema and data contracts
Testing Framework	Custom	Built-in	Built-in (including unit tests)
Scalability	Depends on implementation	Can be costly for large datasets	Efficient for large datasets
Cost	No licensing costs	Open-source (paid options available)	Open-source (paid options available)
Community & Ecosystem	N/A	Large, active community	Smaller, growing community

Table 1: Side-by-side comparison of data transformation tools

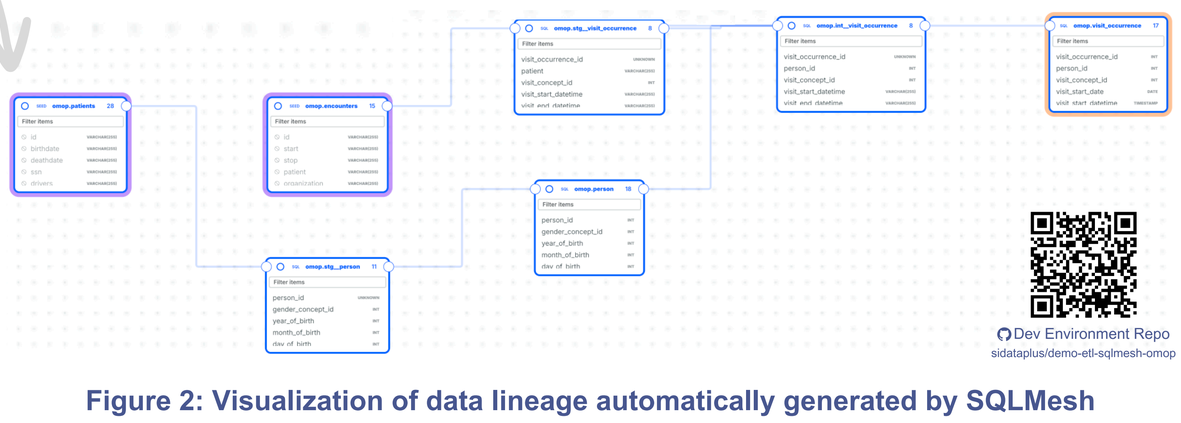

Figure 2: Visualization of data lineage automatically generated by SQLMesh

SQLMesh: Streamlining OMOP CDM Conversion

SQLMesh has optimized data transformation at Siriraj Hospital, boosting efficiency and reliability. This transition standardizes data for research while offering developers enhanced tools.

```sql
MODEL (
  name omop.stg__person,  -- Name of the model
  kind VIEW,              -- Specify the model type
  grain (
    person_source_value   -- Define the grain (unique identifier)
  ),
  audits (UNIQUE_VALUES(
    columns = (person_source_value)   -- Column to check for uniqueness
  ))
);

SELECT *
FROM omop.patients;   -- Source table containing patient data
```

Figure 3: Illustration of SQLMesh Model Configuration

References

[1] Pitchayarat T, Pinyo G, Tanchotsrinon W, et al. Using dbt—a free and open-source software framework—to transform data into OMOP CDM in the ETL process. In: OHDSI Global Symposium 2022. Observational Health Data Sciences and Informatics; 2022.
[2] Carlson R, Phad M, Martin S. Moving OMOP to the cloud with DBT and snowflake (all of us research program). In: OHDSI Global Symposium 2022. Observational Health Data Sciences and Informatics; 2022.
[3] Ashcroft Q, Kirkwood D, Howcroft T, Knight J, Dobson S, Chandrabalan VV. Implementing the OMOP common data model using dbt. In: OHDSI Global Symposium 2023. Observational Health Data Sciences and Informatics; 2023.
[4] Bracons Cucó G, Gil Rojas J, Peñafiel Macias P, et al. OntoBridge Versus Traditional ETL: Enhancing Data Standardization into CDM Formats Using Ontologies Within the DATOS-CAT Project. In: Mantas J, Hasman A, Demiris G, et al., eds. Studies in Health Technology and Informatics. IOS Press; 2024. doi:10.3233/SHTI240681
[5] Tobiko Data, Inc. SQLMesh. 2024. Accessed October 5, 2024. https://sqlmesh.com/
[6] Mao T. SQLGlot. 2024. Accessed October 5, 2024. https://sqlglot.com