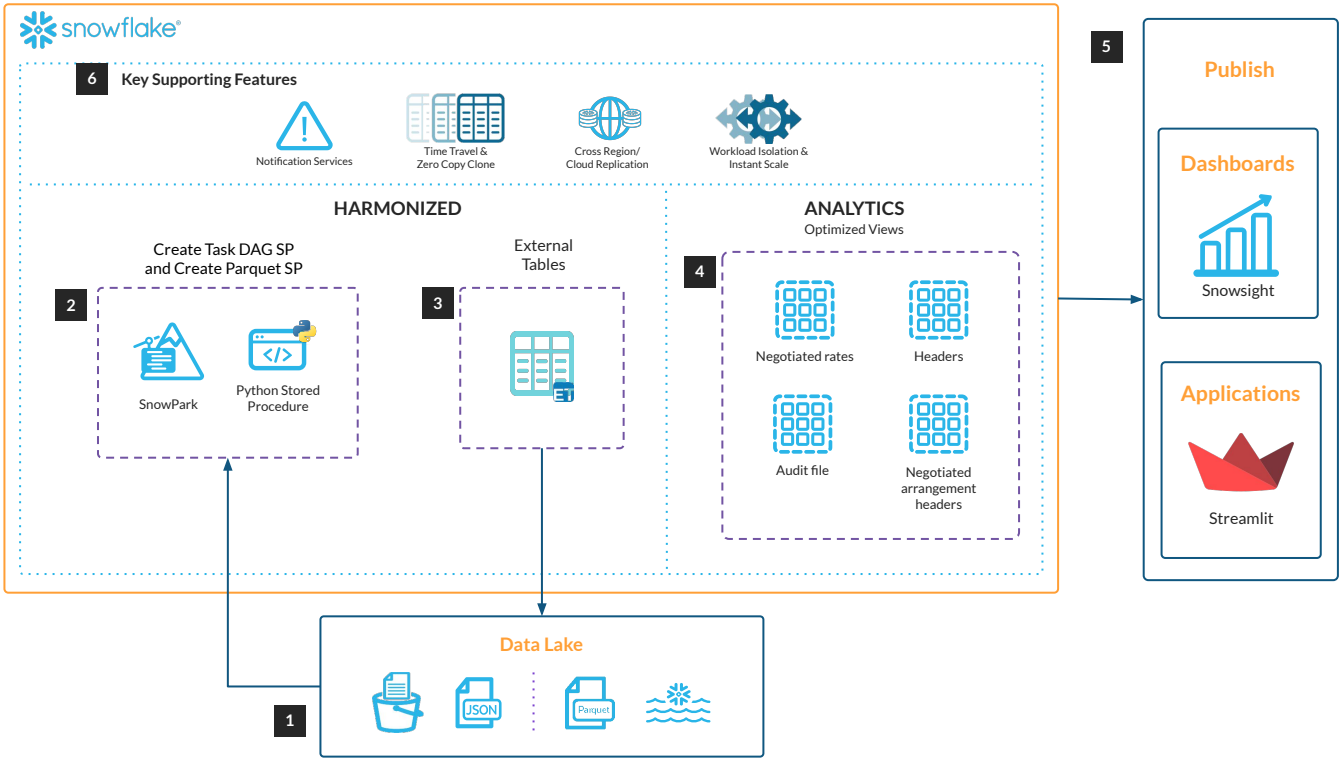
PRICING TRANSPARENCY
REFERENCE ARCHITECTURE (Snowflake Native)



OBJECTIVE

Processing complex CMS mandated JSON files every month requires large computing infrastructure that can be scaled up as the files are made available, then scaled down once the processing is done. Snowpark (Python) first assesses the size of the file then breaks in into chunks which can then be processed in parallel. In addition audit and summary tables are created and all conveniently facilitated by a Streamlit UI.

DESCRIPTION

1. CMS Json files are stored in your data lake environment.
2. Python dynamically reads the new files and determines optimal parallelization to construct a task DAG. Each task processes its portion in parallel storing the parsed data back into the data lake as parquet files.
3. Users may optionally choose to create materialized or dynamic views over the data lake for optimal performance.
4. Data now in a normalized format may be processed for pricing transparency analysis. This data may be combined with other data sources for more comprehensive insights.
5. This data is made available to downstream analytic and BI tools for reporting or the data can be further processed by Streamlit applications.
6. This solution is all facilitated by Snowflake's unique platform offering Notification Services as well as workload isolation and instant scale.

RELATED CONTENT

Blog: Ingestion of Healthcare Pricing Transparency Data Files Natively on Snowflake

POLARIS