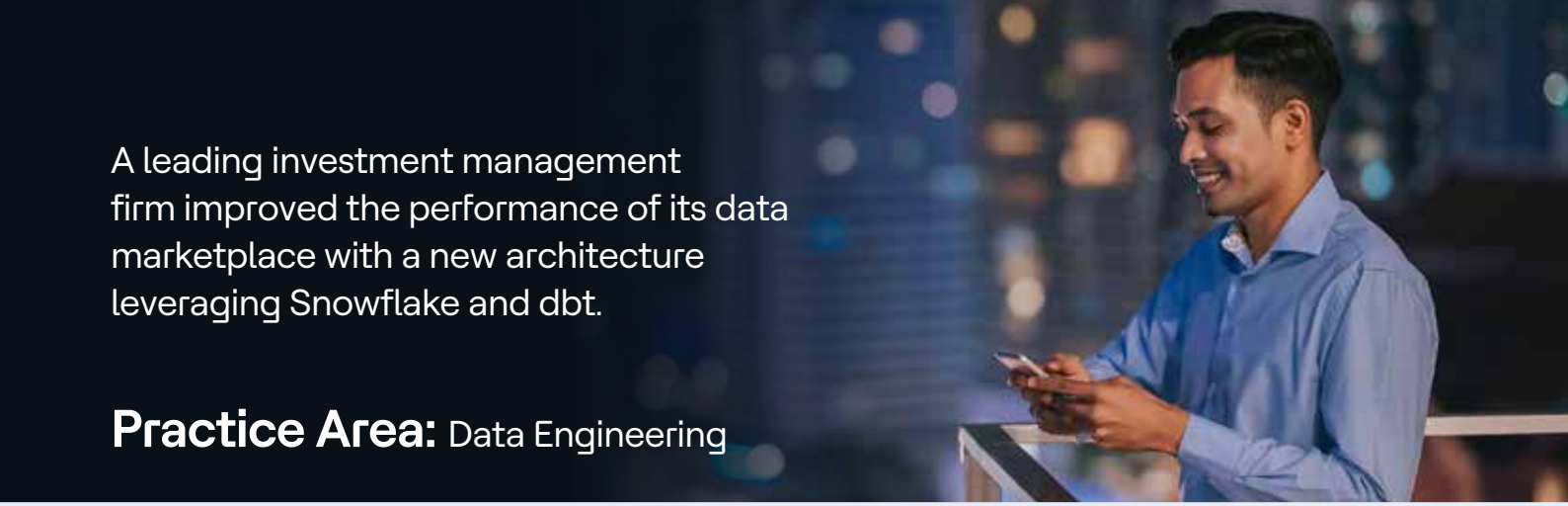
HCLTech | Supercharging Progress™

Data Marketplace Architecture



A leading investment management firm improved the performance of its data marketplace with a new architecture leveraging Snowflake and dbt.

Practice Area: Data Engineering

Objectives

01 Move from legacy architecture and batch processing to a modern stream-based architecture

02 Ensure the system could handle the huge increase of data due to stream-based architecture

03 Design and deploy a metadata-driven ingestion system to reduce the technical knowledge required for data stewards to onboard datasets

04 Build data pipelines in highly volatile infrastructure

05 Accommodate a wide range of data types

Business Impact

Faster insights via reduced data load times

Improved customer satisfaction through better product performance

Technologies

- Snowflake
- DBT
- Python
- Apache Airflow
- DataHub
- Kubernetes

Background

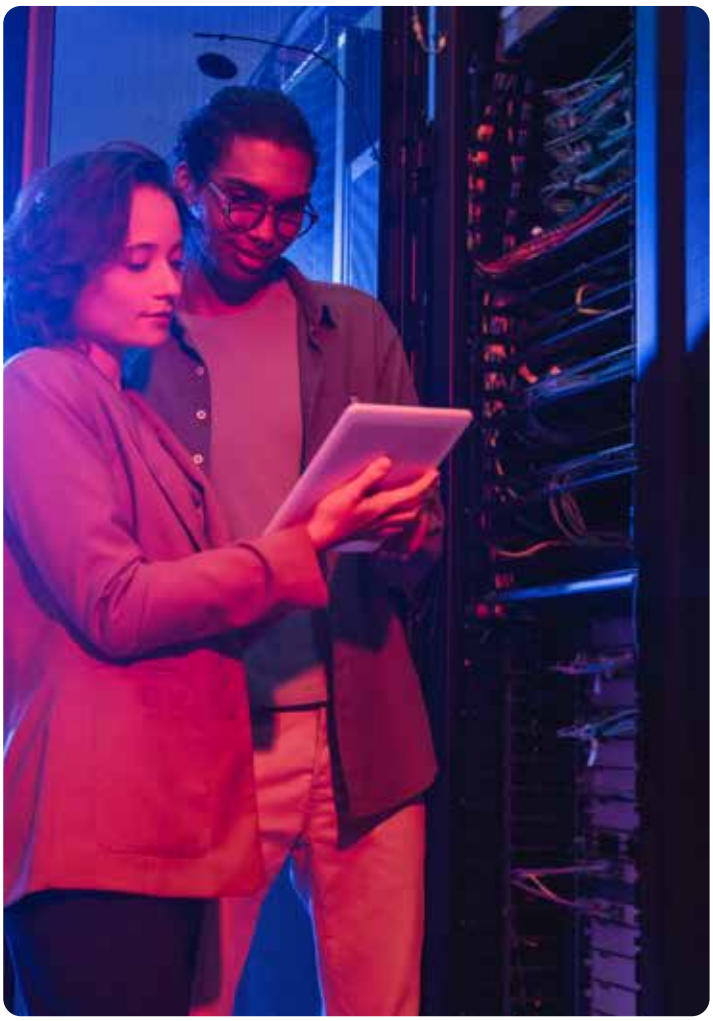
One of the world's largest investment, advisory and risk management firms provides portfolio management software that helps customers visualize and act on investment data. To build on the software's success, the company set out to create a marketplace for financial data sets that leverages the technology behind the software. However, during development, they were unsatisfied with the time it took to onboard new datasets and make them usable on the platform — the underlying issue resulted in month-long delays before a dataset could be shared on the data marketplace. The company wanted to improve the platform's performance to ensure customer satisfaction upon the product's launch and reached out to HCLTech due to the team's Snowflake and dbt expertise to design and implement a solution.

Challenge

At the start of the project, the HCLTech team conducted an analysis of the data pipeline of the client's product. It revealed that the inadequate data set onboarding speeds were the result of two primary issues: the lack a standardized and streamlined integration strategy to ingest data from third-party datasets and the technical limitations of the legacy Sybase database.

To enable faster data throughput, the client needed to shift from batch-based processing to a stream-based architecture. This change, however, would result in a substantial increase in the volume of data that the system would need to accommodate. The system would need to parse 5-10GB XML files into relational tables and quickly process 300-400,000 small files.

The legacy Sybase database was not scalable enough to accommodate such an increase in required data throughput. In addition, the data stewards who onboard the data sets needed to be proficient in Perl, Java, Python and Informatica PowerCenter to be able to onboard new data sets, which further resulted in delays.



Solution

The HCLTech team designed the data architecture and setup, managed the ingestion and transformation setup, and created a data sharing strategy to improve the platform's performance.

The team replaced the outdated Sybase database with a system relying primarily on Snowflake and dbt. In the new system, the data loads into the S3API-compatible object store, where the file formats received from vendors were standardized and schematized. The data is then loaded to Snowflake from the object store to the stage 0 raw layer to maximize the performance and leverage the capabilities of dbt to transform the data and the perform quality check operations. The team chose Apache Parquet as the standard format for its capabilities to store nested data.

To lower the technical proficiency requirements of data stewardship, the team implemented a metadata-driven autoingestion platform. With this platform, data stewards need only to fill out metadata templates for the data to get picked up automatically by the framework. This strategy and system eliminated the need to interact with a database, ETL or orchestration tool, as the new framework automatically generates everything based on the metadata.

To manage the diverse array of data that the platform needs to onboard, the team constructed a data lake that could accommodate structured, unstructured or semi-structured data. This enabled the HCLTech team to rapidly and efficiently access all data assets regardless of format. The team achieved this using a Python-based backend, where the Python code was written to parse configuration and metadata files to automatically generate pipelines with stages for extraction, loading, transformation and data quality checks. The business logics are also configuration-driven, and analysts can write their own configuration and create a curated data set, which can then be shared with clients using Snowflake's zero-copy cloning.

Outcome

HCLTech delivered a highly performant, robust and modular solution for onboarding data sets after three months of planning and three months of development. To enable the fastest possible data onboarding speeds, the new framework accommodates all data input forms while maintaining a high level of data quality control. This way, the platform can serve as a credible source of timely insights for internal and external customers. The client has already used it to onboard new data sources and plans to add over 160 more in the near future.

The new, metadata-driven and configuration-based onboarding tool supports a wide range of connectors to source data feeds created by both external vendors and the client's internal data producers. It sources the data with supported protocols at the desired frequency and delivers it to the object store and the Snowflake database. Thanks to the solution, the client can now onboard new datasets in days, rather than months. The new architecture enables easier data consumption for the client's customers and grants them access to an isolated replica using Snowflake Data Exchange. While the client's legacy architecture had an average latency of three days with larger data sets, the new solution can load close to 4TB of semi-structured XML data in a couple of hours.