HCLTech | Supercharging Progress™

Data quality framework in Snowflake

Summary of the solution framework

For some crucial context, let us first summarize how data quality is ensured through a particular solution framework.

In a traditional ELT or data warehouse solution, you first need to ingest data into your staging area from various source systems and cleanse them before they can be processed further by downstream applications. If data quality is overlooked, data warehouse users will have inaccurate and incomplete data on their hands. This translates directly into erroneous results produced on running analytical queries on the dataset.

This data quality framework is based on configurable DQ rules applied to a specific column or a set of columns of a Snowflake (staging) table, thus curating the dataset by eliminating the bad records.

Business benefits

Customers can get the following benefits by using this framework:

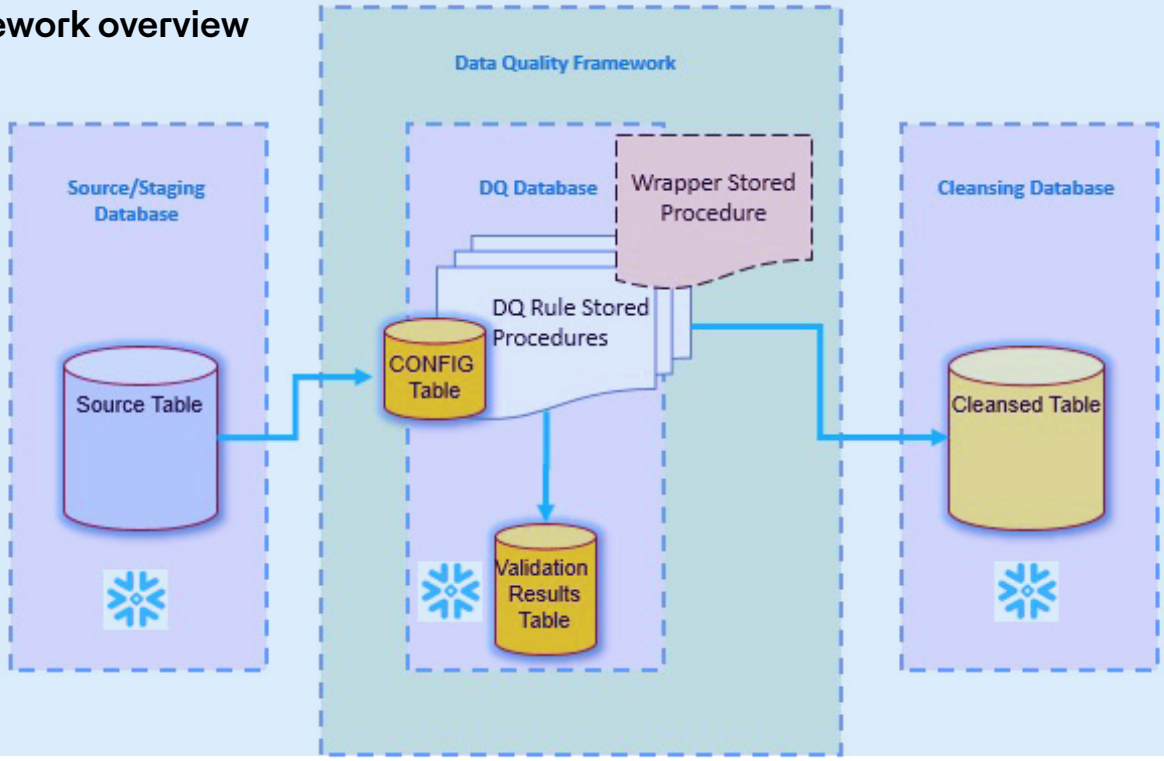
- This framework can be used to curate any Snowflake table by placing the rules as configurations. Hence, the time-to-market would be short as the developers wouldn't need to build any code. Additionally, this framework can give them a jump start to quickly customize and shorten the build phase.

- As the developers have to only put the DQ rule details to the CONFIG table, no code change is involved to cleanse any new data source.

- This framework supports schema evolution. Any change in the structure of any existing table doesn't have any impact on the solution framework, thus eliminating the need of any code change.

- Developers/users don't need to have expertise in Snowflake to use this framework. Basic SQL knowledge is sufficient to use it.

Solution framework overview



A brief technical overview

A JavaScript-stored procedure is created for each DQ rule mentioned below. When applied to a column(s) of a table, the procedure inserts the erroneous records of that table which don't satisfy the concerned DQ rule for the said column(s), along with some other metadata. This includes TABLE_NAME, COL_NAME, INVALID_VALUE, DQ_RULE and ERR_MSG into the DQ_RULE_VALIDATION_RESULTS table.

The Following DQ rules have been created:

RULE_DATE: Used to check the date value conforming to the pattern supplied
RULE_DECIMAL: Used to check a decimal value
RULE_INTEGER: Used to check an integer value
RULE_LENGTH: Used to check whether the length of a field is within the supplied value
RULE_NOT_NULL: Used to check whether a field contains NULL value
RULE_REGEX: Used to check whether a field conforms to the supplied regex pattern
RULE_SQL_FILTER: Used to check whether a record satisfies a SQL predicate
RULE_UNIQUE: Used to validate whether a field contains unique values
RULE_VALID_VALUES: Used to check whether a field contains values specified in the supplied value array

A Wrapper-stored procedure, DQ_RULE_VALIDATION, is created to call the RULE SPs mentioned above based upon the entries made in a configuration table named DQ_RULE_CONFIG for a concerned SOURCE TABLE where APPLY_RULE flag is set to TRUE.

DQ framework features

1

All the validated records can optionally be loaded into a CLEANSED table for downstream processing. The PARAM_CLEANSE_RECORD input parameter of the Wrapper procedure is used to determine the same.

2

If any DQ rule for a table is to be skipped, only APPLY_RULE flag should be set to FALSE for that entry.

3

Adding or removing rules on a dataset doesn't require any code changes. Only CONFIG table entries are required to be inserted/updated

4

A wrapper-stored procedure is created to call the DQ rule procedures based upon the entries made in the DQ_RULE_CONFIG table for a table to be validated.

The DQ_RULE_CONFIG table will hold the rule mapping for a table including "rule name", "rule parameter" and "apply rule flag".

5

Conclusion

This framework can be extended to include more complex cleansing rules as per the requirement and the same architecture can still be seamlessly used. The DQ_RULE_VALIDATION_RESULTS table can be used to create dashboards in Snowsight or any other BI tool to capture error record summary at the table level, DQ rule level, record level or at any other suitable granularity and to capture other KPIs as well.

The Wrapper SP can be scheduled in task for a full-blown Snowflake solution or the framework can be integrated with any ETL/ELT tools like Talend, Informatica, dbt, etc. The objective of this write-up is to help create a DQ framework so that the same can be leveraged to cleanse any source system feed with minimal/no code changes, thereby reducing the time-to-market.

HCLTech | Supercharging Progress™

HCLTech is a global technology company, home to 219,000+ people across 54 countries, delivering industry-leading capabilities centered around digital, engineering and cloud, powered by a broad portfolio of technology services and products. We work with clients across all major verticals, providing industry solutions for Financial Services, Manufacturing, Life Sciences and Healthcare, Technology and Services, Telecom and Media, Retail and CPG, and Public Services. Consolidated revenues as of 12 months ending September 2022 totaled $12.1 billion. To learn how we can supercharge progress for you, visit hcltech.com.

hcltech.com