Snowflake Migration and Testing

STEP BY STEP GUIDE

Contents

1. Introduction

iCEDQ is a DataOps automation platform for Data Testing and Monitoring and is used extensively by our customers for data warehouse testing, cloud data migration testing, and production monitoring.

Enterprises are choosing Snowflake as their data warehouse platform. As with any new data platform, the customer must migrate data from the legacy database to Snowflake, which is a risky endeavor.

83% OF DATA MIGRATIONS FAIL OR EXCEED THEIR BUDGETS AND SCHEDULES
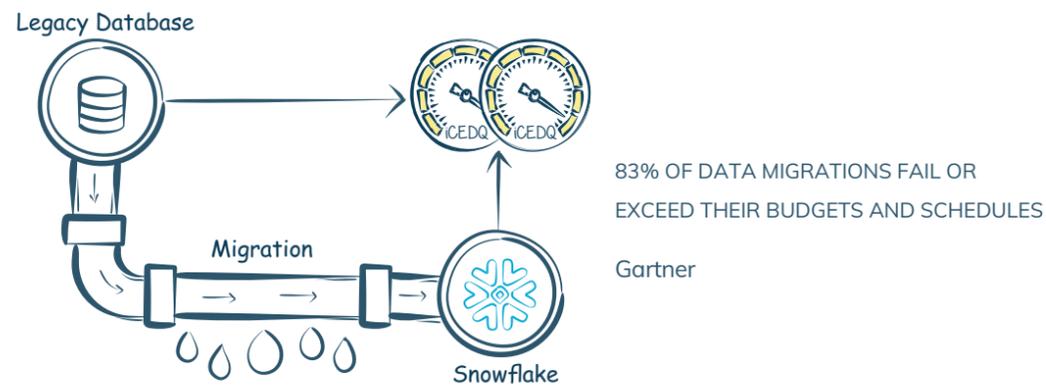
Gartner

iCEDQ has partnered with Snowflake **(SNOWFLAKE TECHNOLOGY PARTNER)** to mitigate the data migration risk. iCEDQ has a data reconciliation engine to certify data between the legacy database and the Snowflake database.

Migrating to Snowflake involves a progression of tasks, some of which are serial, and some may run in parallel. This handbook provides a step-by-step implementation guide for the execution of Snowflake migration and testing.

Intended Audience:
Professionals working on data migration, especially those involved in Snowflake migration.

What will you learn?
This document provides a comprehensive framework with planning, scoping, and testing for migration from legacy databases such as Oracle, DB2, Netezza to the Snowflake database.

KEY STEPS:

Pre-migration Analysis and Test Planning

Schema Migration and Testing with iCEDQ

Auto-Generate Migration Test Rules with ICEDQ's Rule X Software

Initial Load and Testing with iCEDQ

Delta Load and Testing with iCEDQ

*Parallel Run and Testing with iCEDQ*

Cutover and Release Certification with iCEDQ

Production Data Monitoring with iCEDQ

---

2. The Steps: Snowflake Data Migration

The diagram below describes the steps the team needs to perform to migrate data from a legacy database to Snowflake.

① Pre-migration Analysis and Test Planning

② Schema Migration and Testing with iCEDQ

③ Use iCEDQ's RuleX Utility to Automatically Generate Snowflake Migration Testing Rules

④ Initial Load and Testing with iCEDQ

⑤ Delta Load and Testing with iCEDQ

⑥ Parallel Run and Testing with iCEDQ

⑦ Cutover and Release Certification with iCEDQ

⑧ Production Data Monitoring with iCEDQ
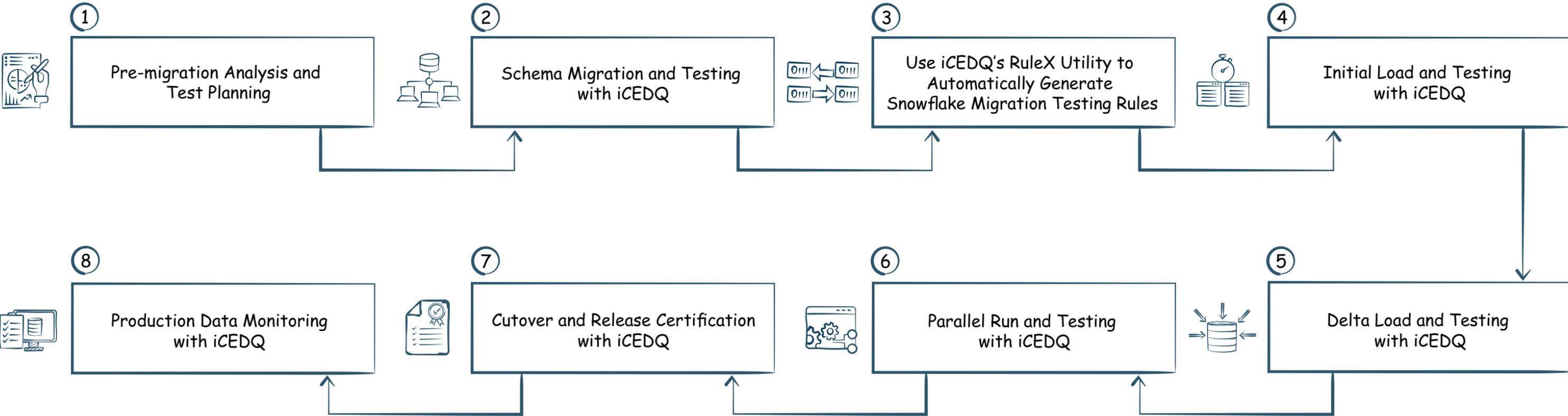
2.1 . Pre-Migration Analysis & Test Planning
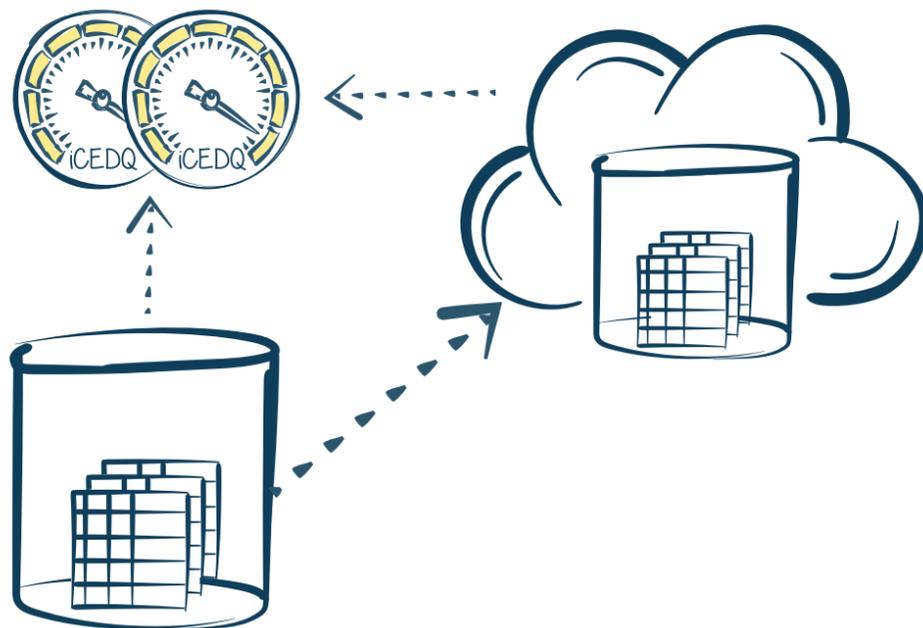

Test Plan

The preparation of migration begins with the analysis of differencesbetween the legacy database and Snowflake. This analysis willprovide the inputs for the overall migration strategy and planning.

Before starting the Snowflake data migration project, it's essential to conduct an as-is analysis of the data ecosystems and identify the changes required for Snowflake database implementation. Below are some of the areas where the migration team needs to pay attention before initiating actual migration.

1. Identify the upstream systems
2. Identify the downstream systems
3. Identify the SMEs and business users for respective upstream and downstream systems
4. Identify the user ids and application ids that login to the legacy database. Separate the read-only users from users with write access
5. Setup a team that will be responsible for the actual data migration and testing
6. Setup a PMO to coordinate changes with the impacted downstream and upstream systems
7. Draw the Architecture Diagram for as-is (Current data warehouse) and to-be (Snowflake)
8. Define a data migration strategy (Lift and Shift/ Staged)
9. Document the physical differences between the legacy database and Snowflake
   a. Datatype differences
   b. Precision issues
   c. Differences between stored procedures
   d. SQL syntax differences
   e. Non-enforcement of index, primary keys, and foreign keys
   f. Differences in security provisioning
   g. Connectivity differences

2.2 Schema Migration and Testing with iCEDQ

The migration commences with the movement of database objects, including databases, schemas, tables, views, and sequences from the legacy database to Snowflake. Movement of database objects is done in tandem with the creation of user account names, roles, and object grants within Snowflake. After deciding which objects to move, the next step is to choose a method for the movement of the data model from the legacy database to Snowflake. The list below outlines the steps enterprises need to perform for migrating the schema into Snowflake and testing it with iCEDQ.

1. Reverse engineer the existing data warehouse data model.
2. Create a new data model for Snowflake.
3. Set the proper physical properties for datatypes, schema name, and other objects
4. Recreate the equivalent of stored procedures, views, synonyms in Snowflake.
5. Setup appropriate, read-only, and write access to users.
6. Capture all the DMLs (aka SELECT, INSERT, UPDATE, DELETE) from both the upstream and downstream applications. If necessary, use logging features of the legacy database to capture the statements.
7. Test the new Snowflake schema with iCEDQ's schema Reconciliation rules.

   a. Create rules to compare legacy database schema with Snowflake schema.
   b. Ensure during schema reconciliation, iCEDQ rules have equivalent datatypes or names correctly translated.
   c. Schedule iCEDQ's schema reconciliation to run periodically to capture any intended or unauthorized changes in the snowflake schema.
   d. Based on the schema exceptions report generated by iCEDQ, take corrective actions.
   e. Get business users to sign off on the schema and data models.
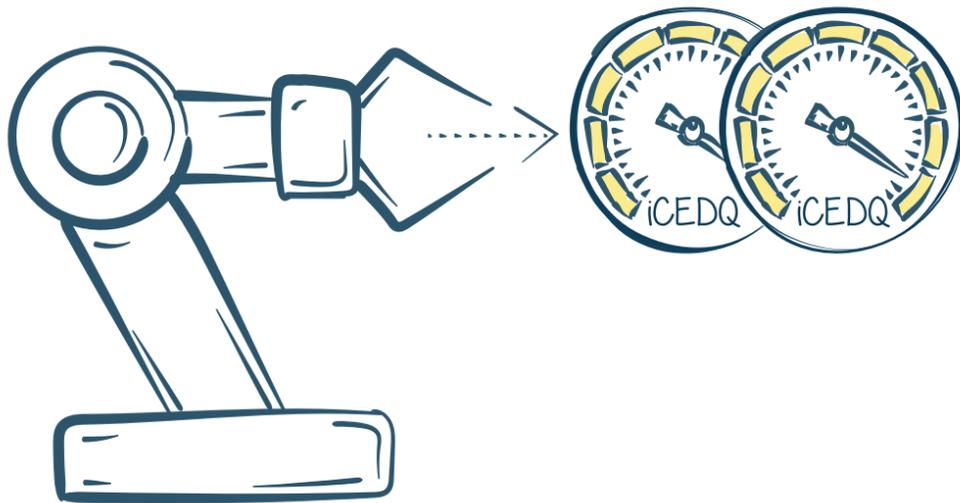
During database migration, business rules usually do not change. The changes are limited to the technical and structural aspects of the database. Mostly The logical data model is the same, but there will be changes in the physical data model, depending on the features that are not supported or the equivalent available in Snowflake.

# 2.3 Auto-Generate Migration Test Rules with iCEDQ's RuleX Software

iCEDQ's **RuleX** *utility smartly analyzes legacy database and Snowflake metadata and automatically creates rules inside the iCEDQ platform. These rules are ready to use for test executions.*

The migration from a legacy database to Snowflake involves minimal changes to the data model. While the stored procedures and triggers might have to be re-written, the tables, columns, and data types remain similar. If rules are created in iCEDQ using its UI (User Interface), it takes a few minutes to develop and implement the rules, but the task is repetitive and probably boring. A Snowflake migration project with thousands of tables will require thousands of rules. Manual rule creation in iCEDQ's UI, while easy, still requires many person-hours

iCEDQ's RuleX is a smart software utility that automatically creates multiple data testing rules in iCEDQ that are ready for execution. Below are the steps to configure and run RuleX auto rule generation utility:

1. Create a metadata document with table and view list for the legacy and Snowflake data model (datatype, precision)
2. Provide the metadata document to the RuleX software
3. RuleX will automatically create data migration testing rules based on the pre-configured metadata
4. Fine-tune the iCEDQ rules if required
5. Run/schedule the rules

2.4 Initial Data Load and Testing with iCEDQ
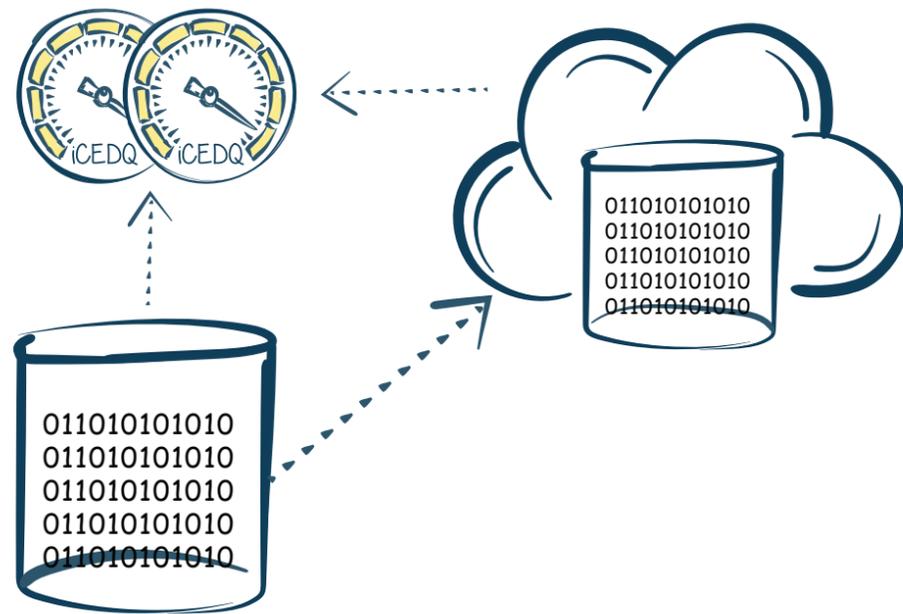
Once the new schema is certified, the next step is to do an initialload of the Snowflake database. Movement of historical data into the cloud instance of Snowflake is a lengthy process, the team needs to perform different tests like Row Count, Data Aggregation, and Data Compare to ensure the accuracy of the migration.

To begin the migration of data from legacy systems to Snowflake, the team will need to extract historical data from the legacy system and transfer the data to Snowflake. The below list captures the steps that enterprises need to perform for initialization of Snowflake.

1. Decide how much historical data is needed to seed the Snowflake data warehouse. The migration project could be an excellent time to purge unnecessary or archived data.
2. Even if all the data is eventually needed, we can bring some data in at a later stage.
3. Straight ETL migration processes will be slow for the migration of large tables, so export-import utilities and scripts must be used. Snowflake suggests the usage of their standard methods (PUT and COPY) for importing data into Snowflake .
4. Execute, extract, and load the data from the legacy database to Snowflake.
5. Test the initial data load with iCEDQ.
6. Since the initial load could have billions of records, different tests are required to test the accuracy of data and migration:

   a. **Row Count Test**: This is the initial test to compare row counts between the source database and Snowflake. The success of the Row Count test does not imply that the data has no issues, and further detailed checks are required to confirm the accuracy of the data.
   b. **Data Aggregation Test**: This test evaluates quantities such as the sum of numeric columns by month or year between the legacy database and Snowflake database. If the data aggregation test fails for a specific group, then row by row comparison for that group may be performed.
   c. **Data Compare Test**: This test compares data between the legacy and Snowflake databases. This test will recognize the exact data issues that the team needs to fix. Some of the data issues identified by this test are
      i. Data Truncation
      ii. Floating-point issues
      iii. Incorrect Date Time
      iv. Special Character conversion
   d. Get business sign off on initial data load.

2.5 Delta Load and Testing with iCEDQ

Incremental Load

Source
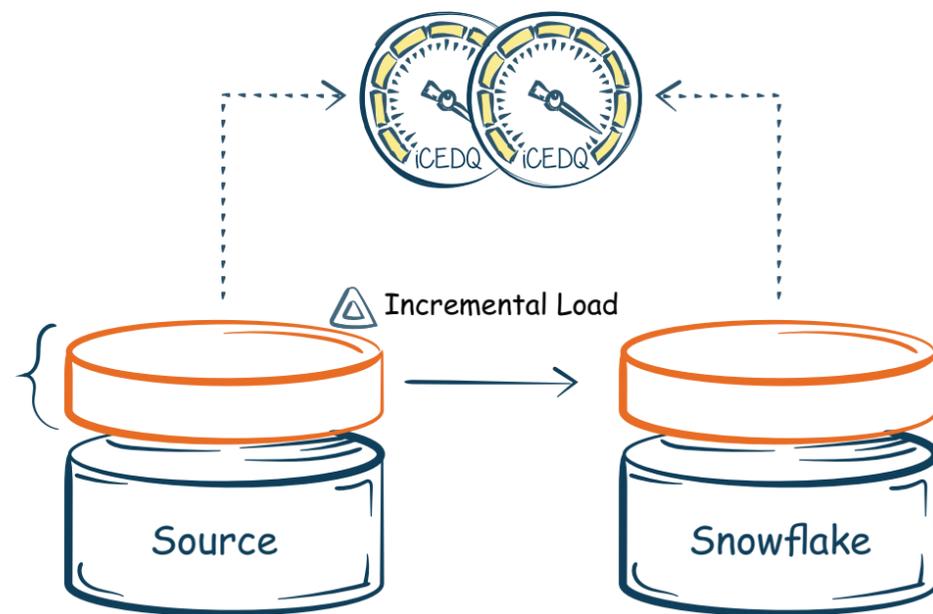
Snowflake

On completion of historical data migration into Snowflake. It's necessary to load the delta(data) in Snowflake by running the delta load process for keeping data in Snowflake up to date.

The upstream systems will keep on generating data after the migration of the initial data load. It's essential to load the incremental data into Snowflake for keeping it up to date. Different tests are required to ensure the accuracy of this data. The following list describes the steps to be performed for loading incremental data from the legacy database to the Snowflake database:

1. Use export-import utilities or scripts to load the delta (data) from the legacy database to Snowflake.
2. Execute the scripts to extract and load the delta (data) from the legacy data warehouse to Snowflake.
3. Test the delta load with iCEDQ.
4. Below are the different tests required to ensure the accuracy of data during incremental load:

    a. **Delta Row Count Test**: This test compares row counts between the legacy database and Snowflake. The success of the Delta Row Count test does not imply that the data has no issues, and further detailed checks are required to ensure the accuracy of data

    b. **Delta Data Aggregation Test**: This test evaluates quantities such as the sum of numeric columns by month or year between the legacy and Snowflake database. If the data aggregation test fails for a specific group, then row by row comparison for that group may be performed

    c. **Delta Data Compare Test**: This test compares daily/delta loads to identify issues between the legacy and Snowflake databases. This test is required to check for duplicates in tables and identify which upstream process is responsible for data quality issues.

5. Get business sign off on delta load.

2.6 Parallel Run and Testing with iCEDQ

Parallel deployment and operation
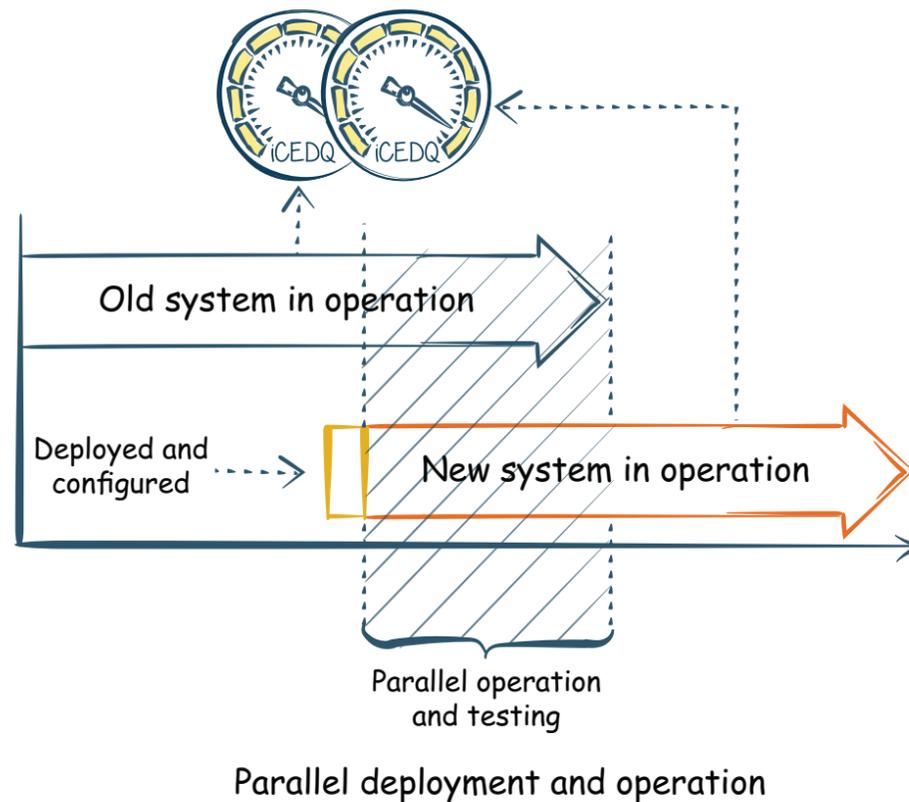
After the movement of initial and delta load into Snowflake, it's necessary to run the legacy system and Snowflake in parallel before officially migration is signed off

Before the final cutover to Snowflake, the legacy system and Snowflake should run in parallel, which means that the upstream systems will populate both the legacy system and Snowflake at the same time. When the upstream system feeds the same value to the legacy as well as the snowflake data warehouse, the final data state should be the same. iCEDQ will reconcile the new data values in both the system to certify the parallel run.

1. Both systems must run parallelly for some time. This step is crucial to validate whether the migration was completed successfully before shutting down the legacy system.
2. Stop the data migration processes and configure the upstream systems to populate both the legacy system and Snowflake.
3. Execute the below tests to ensure that the data is loaded correctly in Snowflake:
   a. **Row Count Test**: Compare row count between the legacy database and Snowflake.
   b. **Data Aggregation Test**: Evaluate quantities such as the sum of numeric columns by month or year between the legacy system and Snowflake.
   c. **Data Compare using MD5 hash**: This test compares the contents of the row based on the MD5 hash, rather than comparing each column individually.
      I. Add an MD5 hash to the data extracted from the legacy system for columns that will not change when loaded into Snowflake.
      II. After loading data into Snowflake, generate another MD5 hash across the same set of columns and compare it with the MD5 hash from the legacy system.
   d. Get business users to sign off on the parallel run.

2.7 Cutover and Release Certification with iCEDQ
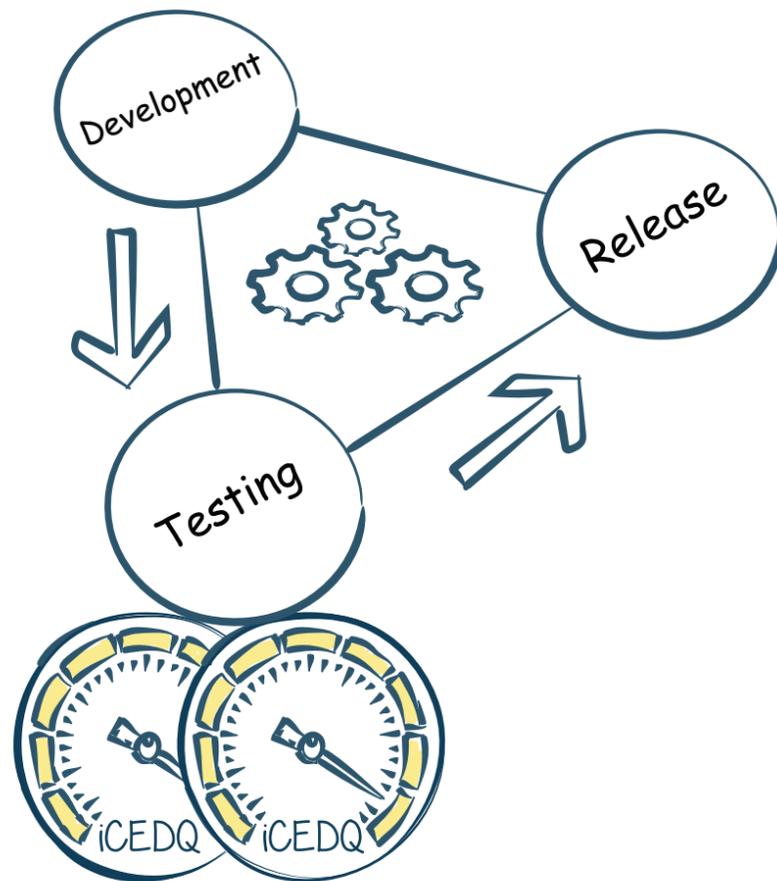
Before cutover, the final certification is the final milestone during the lifecycle of the Snowflake migration. Once the complete migration of the legacy data model, data loads, and reporting in Snowflake is certified, the next step is to switch off the legacy database. This event is called Cutover to Snowflake.

Before going live on Snowflake, a certification from a trusted external party like iCEDQ will instill the confidence of business users in the migration project. Below are the steps the team needs to perform before switching off the legacy database:

1. Define the qualification criteria for the passing of each migration step
2. Configure these criteria as rules in iCEDQ
3. Get a certification from iCEDQ that the previous steps (Schema Migration, Initial Load, Delta Load, Parallel Run) have passed the qualification criteria and inform the business users
4. Communicate the cutover date to legacy system users
5. Ensure legacy system users can log in to Snowflake
6. Ensure all development code is checked in and backed up
7. Point production BI reports to Snowflake
8. Turn off the data pipeline and access to legacy database
9. Cutover to Snowflake
10. Get business users to sign off on cutover

2.8 Production Data Monitoring with iCEDQ

Data issues are inevitable even after the migration as the continuous movement and transformation of data is fraught with possibilities for data corruption and data loss. Therefore, It makes sense to continuously monitor the production data so that business users can make correct decisions

iCEDQ can perform production data monitoring on a Snowflake Data warehouse leveraging its ability to audit, validate, and reconcile data within/across heterogeneous data sets. The below list describes the production data monitoring capabilities of iCEDQ:

1. **Audit Rules Engine:** These rules monitor data daily; they generate exception reports highlighting data issues for the technology team to fix so that business users can have correct data for making critical decisions.

2. **Compliance Reporting**: These Rules ensure that the data is compliant with SOX, FINRA, BCBS 239, and other regulations. The iCEDQ repository stores the test results after test execution, allowing users to create custom reports for compliance purposes.

3. **Workflow Integrations:** iCEDQ enables enterprises to integrate/embed audit rules in their current workflow using the command line or web service interface. Based on the results returned by the rules, these workflows can be stopped to take corrective action. This integration allows you to monitor the system for data issues proactively.

3. ROI and iCEDQ's Role in Data Migration

There are many legacy data-warehouses such as Teradata, Netezza Snowflake. The amount of data they are warehousing is in multiple terabytes. Manual testing can involve data sampling and manual scripting, making it impossible and unrealistic to test such a volume of data. On the other side, iCEDQ has helped our clients verify 100% of their data migration to Snowflake with ICEDQ's rules-based approach and RuleX utility; it enables CI/CD pipelines, Jenkins integration, and provides a robust solution for end-to-end Snowflake data migration testing.

"We have standardized iCEDQ for all our cloud migration"

SENIOR DIRECTOR OF ADVANCED ANALYTICS, **ALBERTSON**

"iCEDQ has enabled testers to keep up with the pace of developers. and reduced the testing time by half"

DIRECTOR OF QUALITY ASSURANCE, **HEALTHFIRST**

"BMC was able to achieve 100% test coverage after iCEDQ implementation."

DIRECTOR OF BUSINESS ANALYTICS, **BMC SOFTWARE**

"RuleX utility helped BMC reduce the duration of IT testing from 24 months to 2 months"

DIRECTOR OF BUSINESS ANALYTICS, **BMC SOFTWARE**

50% reduction in the number of resources required for data migration testing

100% automation in test execution

Zero production incidents

100% test coverage

4 Conclusion

Data Migration is risky, however with the right approach to testing, mitigating the risks is possible. The iCEDQ DataOps automation platform, along with techniques, procedures, and our expertise in data testing, can not only accelerate but ensure the success of your data migration to Snowflake.

As a Snowflake Technology Partner, iCEDQ has built utilities to not only automate testing but also automate rule creation. iCEDQ blends seamlessly into your migration project as a dependable and time-tested suite of migration testing tools - whether your organization is fully staffed for platform migration or you need additional capabilities to help you achieve a smooth migration.

About iCEDQ

iCEDQ is an industry leading DataOps platform for Data Test Automation and Production Data Monitoring. iCEDQ is a certified Snowflake technology partner. Our clients, such as Albertsons, Fidelity, and BMC are extensively using iCEDQ and RuleX software for their Snowflake data migration testing. Also, consider iCEDQ DataOps Platform for:

Legacy Database

Migration

Snowflake

ETL / Data Warehouse

Cloud Data Migration

BI Report Testing

Big Data Testing

Production Data

Contact Us: 60 Long Ridge Road, Suite 303, Stamford CT 06902 | contact@iCEDQ.com | +1 (203) 666-4442