NETEZZA TO SNOWFLAKE MIGRATION GUIDE

Migration strategies and best practices

CHAMPION GUIDES

TABLE OF CONTENTS

WHY MIGRATE?

Decades ago, Netezza identified the need to manage and analyze large volumes of data. But the volume, velocity, and variety of data have since changed, and the cloud has enabled what's possible today with modern data analytics. For example, by separating compute from storage, Snowflake has developed a modern cloud data platform that automatically and instantly scales storage and compute in a way not possible with Netezza, whether the current Netezza system is on-premises or in the cloud. Snowflake accomplishes this with its multi-cluster, shared data architecture.

YOUR MOTIVATION TO MIGRATE

Some of the key reasons customers migrate off of Netezza include the following:

1. **The legacy platform is inadequate:** Traditional technology fails to meet the needs of today's business users, such as unlimited concurrency and performance.
2. **The cloud offers a no-management solution:** Moving from on-premises to the cloud means moving away from traditional IT delivery models to a more on-demand, as-a-service model with minimal management intervention.
3. **New data sources and workloads are already in the cloud:** The cloud also allows for new types of analytics to be assessed and refined without a long-term commitment to infrastructure or specific tools.
4. **The cost is affordable and predictable:** Snowflake Cloud Data Platform allows true pay-as-you-go cloud storage and compute scalability without the need for complex reconfiguration as your data or workloads grow.

WHY SNOWFLAKE?

Snowflake's innovations break down the technology and architecture barriers that organizations still experience with other data warehouse vendors. Only Snowflake has achieved all of the defining qualities of an effective cloud data platform:

NEAR-ZERO MAINTENANCE

Snowflake reduces complexity with built-in performance so there's no infrastructure to tweak and no tuning required.

ALL OF YOUR DATA

With Snowflake, you can create a single source of truth to easily store, integrate, and extract critical insights from petabytes of structured and semi-structured data (JSON, XML, AVRO, ORC, or Parquet).

ALL OF YOUR USERS

Snowflake allows a virtually unlimited number of concurrent users and applications without performance degradation.

FASTER ANALYST ACCESS TO DATA

Snowflake's elastic, near-unlimited scale and speed means analysts have fast access to all current and historical data at any time to make quicker, more accurate decisions.

DATA SHARING

Snowflake enables direct, governed, and secure data sharing in near-real time, so enterprises can easily forge one-to-one, one-to-many, and many-to-many data sharing relationships.

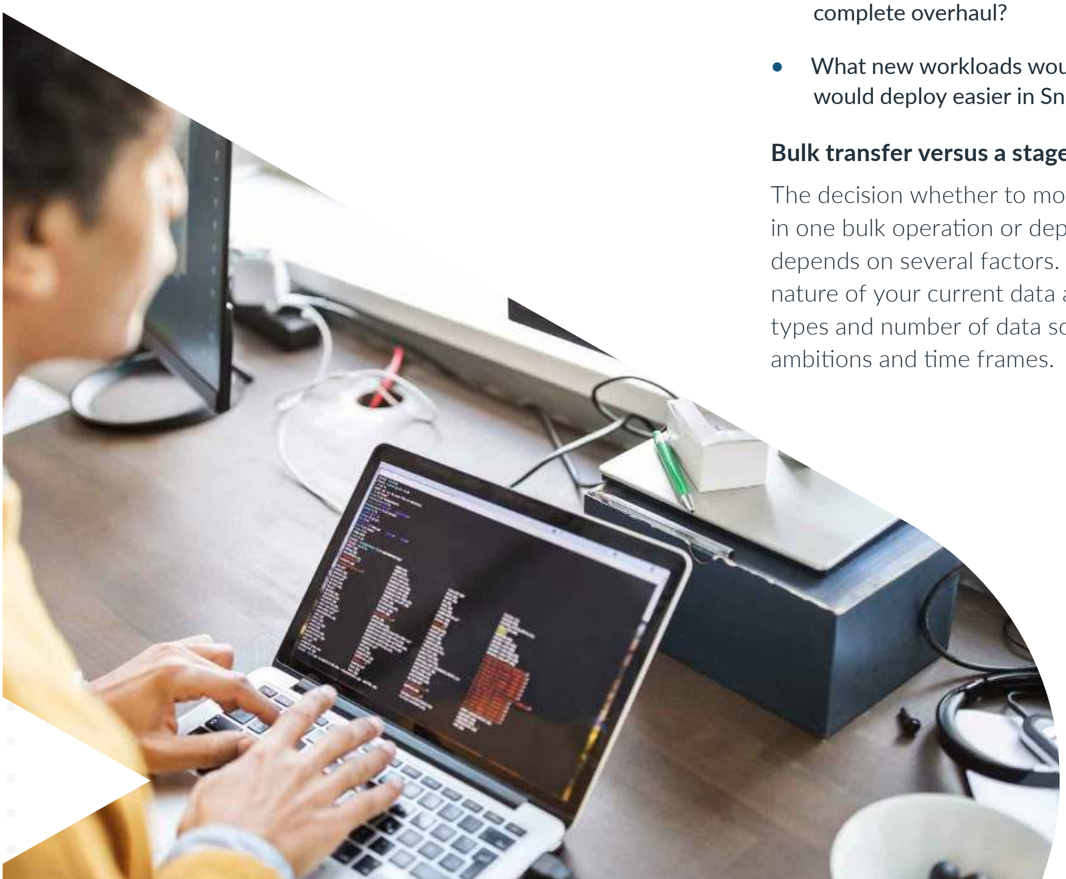
COMPLETE SQL DATABASE

Snowflake supports the tools millions of business users already know how to use today.

STRATEGY—THINKING ABOUT YOUR MIGRATION

WHAT SHOULD YOU CONSIDER?

There are several things to contemplate when choosing your migration path. Many organizations pilot the migration on a subset of the data and processes. Then they migrate in stages, reducing risk and showing value sooner. However, you must balance risk mitigation against the need to maintain momentum and minimize the period of running systems in parallel. In addition, your approach may be constrained by interrelationships within the data, such as data marts that rely on references to data populated via a separate process in another schema.



Questions to ask about your workloads and data:

- What workloads and processes can you migrate with minimal effort?
- Which processes have issues today and would benefit from reengineering?
- What workloads are outdated and require a complete overhaul?
- What new workloads would you like to add that would deploy easier in Snowflake?

Bulk transfer versus a staged migration

The decision whether to move data and processes in one bulk operation or deploy a staged approach depends on several factors. They include the nature of your current data analytics platform, the types and number of data sources, and your future ambitions and time frames.

Consider moving data in one bulk transfer if you have any of the following:

- Highly integrated data across the existing warehouse
- A single independent, standalone data mart
- Well-designed data and processes using standard ANSI SQL
- A need to move off legacy equipment quickly

Consider using a staged approach if you have any of the following:

- A warehouse platform with many independent data marts and other data applications that can be moved independently over time
- Critical data and processes within your data warehouse that no longer perform well and require reengineering
- New business requirements that can't be met by reworking legacy processes
- Changes to your data ecosystem, such as new data ingestion, BI, or visualization tools

WHAT YOU DON'T NEED TO WORRY ABOUT

When migrating to Snowflake from Netezza, you can ignore the following factors because they are no longer relevant:

Data distribution

In Snowflake, you don't have to worry about data distribution or data skewing. Because compute is separate from storage in Snowflake's architecture, the data is not predistributed to the MPP compute nodes. Since Snowflake's data is not predistributed, it can scale to more parallel compute nodes instantly. But with Netezza, customers would have two options:

- Purchasing a new appliance, followed by migration to the new system with everything that entails
- Requesting a capacity increase (available to Netezza customers who have purchased a Capacity on Demand system) and waiting for implementation assistance.

Either option requires significant planning and resources, which prevents organizations from accessing data. But Snowflake's instant scalability enables organizations to have consistent access to data in near-real time.

Workload management

Workload management is unnecessary in a Snowflake environment due to its multi-cluster architecture, which enables you to create separate compute clusters for your disparate workloads to avoid resource contention completely.

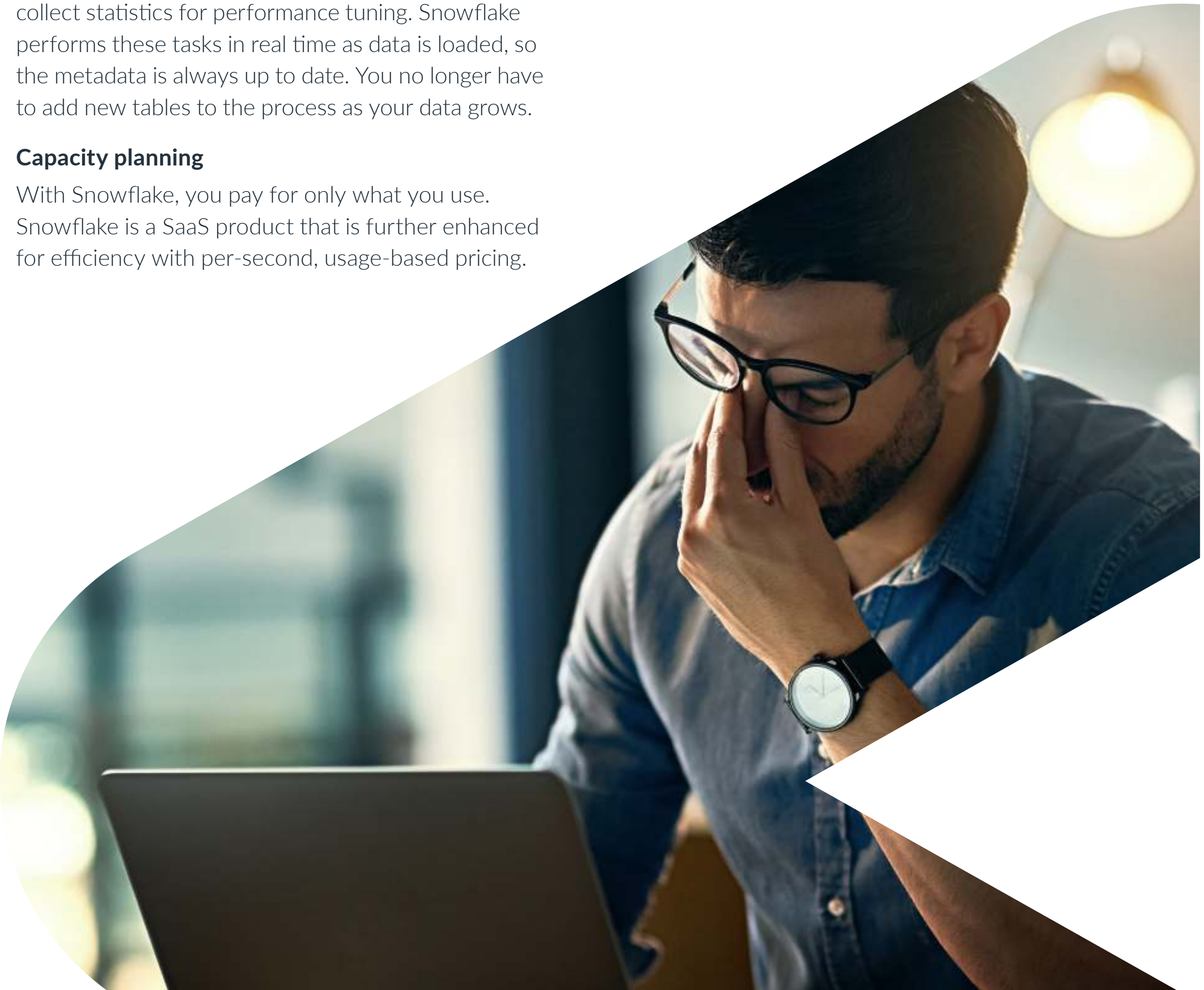
Statistics collection

Snowflake automatically captures statistics, relieving DBAs from the mundane tasks of setting up jobs to collect statistics for performance tuning. Snowflake performs these tasks in real time as data is loaded, so the metadata is always up to date. You no longer have to add new tables to the process as your data grows.

Capacity planning

With Snowflake, you pay for only what you use. Snowflake is a SaaS product that is further enhanced for efficiency with per-second, usage-based pricing.

Under this model, Snowflake also offers further cost reductions for customers who want to pre-purchase usage. But with capacity planning for an on-premises Netezza system, you run a risk of over- or under-configuring your system. If you need more capacity, you must buy in predefined increments. Snowflake's elastic storage and compute architecture eliminates this risk, so you can save money and avoid the time previously spent on extensive planning.



Disaster recovery

Netezza has several disaster recovery scenarios, but many of them require you to purchase another appliance. With Snowflake, this is unnecessary. Snowflake leverages many of the built-in features of the cloud, such as the automatic replication of data built into AWS. Snowflake is automatically synced across three AWS availability zones by design.

Separate dev/test environment

With Netezza, to do development and testing, you need additional appliances, which means an additional cash outlay for the hardware plus the configuration time. But with Snowflake, you can simply create another database in your account and set it up for any purpose you need, such as dev or test. In addition, with Snowflake's Zero-Copy Cloning, you can instantly populate those databases with complete copies of production data for no additional cost. With Netezza, you would have to endure the painstaking process of exporting and importing your production data between servers.

MIGRATING YOUR EXISTING NETEZZA WAREHOUSE

To successfully migrate your enterprise data warehouse to Snowflake, develop and follow a plan that includes the steps in this section.

MOVING YOUR DATA MODEL

As a starting point for your migration, you need to move your database objects, including databases, tables, views, and sequences from Netezza to Snowflake. In addition, you may want to include all of your user account names, roles, and objects grants. At a minimum, set up the user who owns the Netezza database on the target Snowflake system before migrating data. The scope of your initial migration dictates the specific objects you decide to move.

After deciding which objects to move, choose one of the following methods for moving your data model from Netezza to Snowflake:

Using a data modeling tool

If you have stored your data warehouse design in a data modeling tool, you can generate the DDL needed to rebuild your database objects. Because Snowflake uses standard SQL, you simply need to pick ANSI SQL instead of Netezza as the output scripting dialect. Keep in mind that Snowflake is self-tuning and has a unique architecture. You don't need to generate code for any distribution keys. You need only need basic DDL, such as CREATE TABLE, CREATE VIEW, and CREATE SEQUENCE. After you have these scripts, you can log into your Snowflake account to execute them.

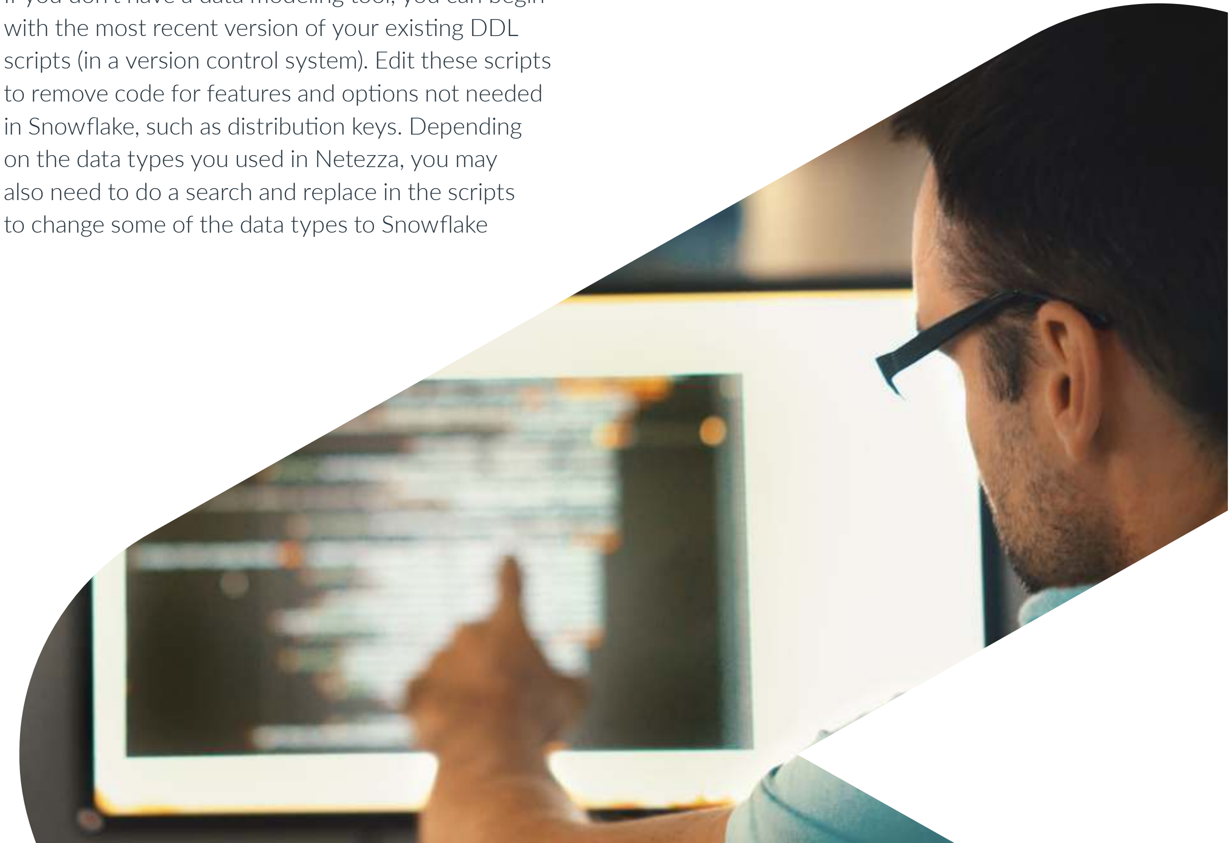
If you have a data modeling tool, but the model is not current, we recommend you reverse engineer the current design into your tool, then follow the approach outlined above.

Using existing DDL scripts

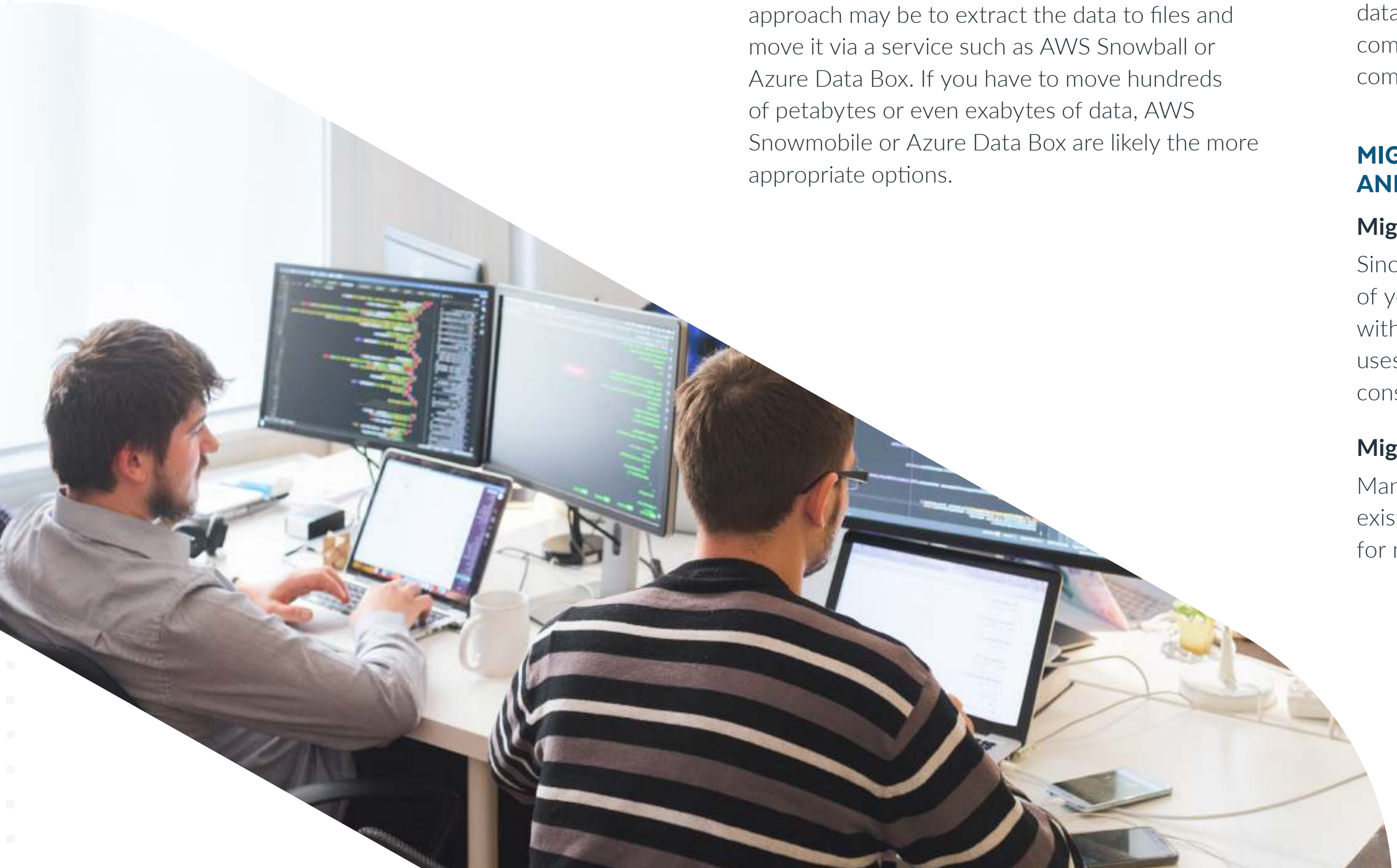
If you don't have a data modeling tool, you can begin with the most recent version of your existing DDL scripts (in a version control system). Edit these scripts to remove code for features and options not needed in Snowflake, such as distribution keys. Depending on the data types you used in Netezza, you may also need to do a search and replace in the scripts to change some of the data types to Snowflake optimized types. For a list of these data types, see Appendix C.

Creating new DDL scripts

If you don't have a data modeling tool or current DDL scripts, you will need to extract the Netezza metadata needed to generate these scripts using a utility such as nzdumpschema or NZ_DDL. But for Snowflake, this task is simpler since you won't need distribution keys.



As mentioned previously, depending on the data types in your Netezza design, you may also need to change some of the data types to Snowflake optimized types. You will likely need to write a SQL extract script to build the DDL scripts. Rather than do a search and replace after the script is generated, you can code these data type conversions directly into the metadata extract script which lets you automate the extract process and execute the move iteratively. Plus, you will save time editing the script later. Additionally, coding the conversions into the script is less error-prone than any manual cleanup process, especially if you are migrating hundreds or even thousands of tables.



MOVING YOUR EXISTING DATA SET

After building your objects in Snowflake, move the Netezza system's historical data to Snowflake. You can use a third-party migration tool (see Appendix A), an ETL tool, or a manual process. When choosing a process, consider how much data you have to move. For example, to move tens or hundreds of terabytes up to a few petabytes of data, a practical approach may be to extract the data to files and move it via a service such as AWS Snowball or Azure Data Box. If you have to move hundreds of petabytes or even exabytes of data, AWS Snowmobile or Azure Data Box are likely the more appropriate options.

If you choose to move your data manually, you will need to extract the data for each table to one or more delimited flat files in text format using either External Tables to get single files or NZ_UNLOAD to create several files in parallel. Then upload these files using the PUT command into a cloud storage staging bucket, either internal or external. We recommend that these files be between 100 MB and 1 GB to take advantage of Snowflake's parallel bulk loading.

After you have extracted the data and moved it to the storage service of your chosen cloud infrastructure provider, you can begin loading the data into your table in Snowflake using the COPY command. See more details about the COPY command in the Snowflake online documentation.

MIGRATING YOUR QUERIES AND WORKLOADS

Migrating data queries

Since Snowflake uses ANSI-compliant SQL, most of your existing queries will execute on Snowflake without requiring change. However, Netezza uses some specific extensions, so there are a few constructs to watch out for.

Migrating BI tools

Many of your queries and reports likely use an existing BI tool. Therefore, you'll need to account for migrating those connections from Netezza to Snowflake. You'll also have to test those queries and reports to be sure you're getting the expected results.

This should not be too difficult since Snowflake supports standard ODBC and JDBC connectivity, which most modern BI tools use. Many of the mainstream tools have native connectors to Snowflake. Check the Snowflake website to see if your tools are supported. Don't worry if your tool of choice is not listed. You should be able to establish a connection using either ODBC or JDBC. If you have questions about a specific tool, your Snowflake contact will be happy to help.

Handling workload management

As stated earlier, the workload management required in Netezza is unnecessary with Snowflake. The multi-cluster architecture of Snowflake enables you to create separate virtual warehouses (compute clusters) for your disparate workloads to avoid resource contention completely. Your workload management settings in Netezza (WLM) will give you a good idea of how to set up Snowflake virtual warehouses. However, you'll need to consider the optimal way to distribute these in Snowflake.

As a starting point, create a separate virtual warehouse for each workload and to size the virtual warehouse to meet the SLA for that workload. Consider the following:

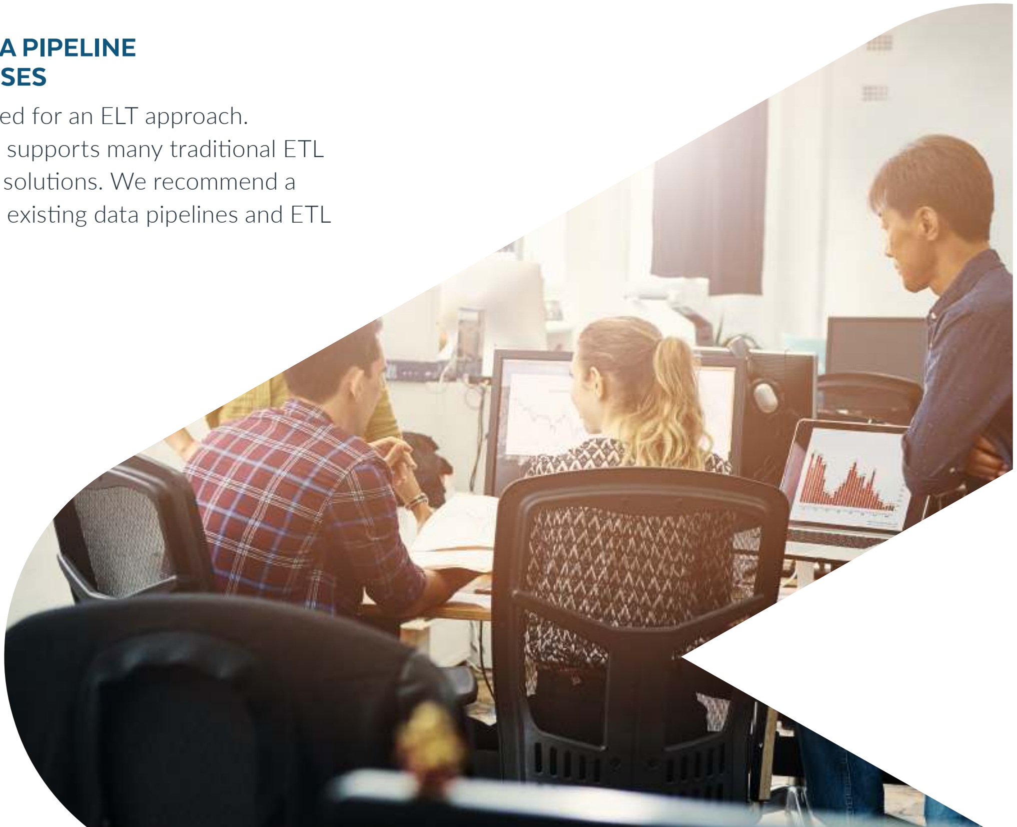
- Is there a specific time period in which this workload must complete? Between certain hours? You can easily schedule any Snowflake virtual warehouse to turn on and off or to automatically suspend and resume when needed.
- How much compute will you need to meet that window? Use that answer to determine the virtual warehouse size.
- How many concurrent connections will this workload need? If you normally experience bottlenecks, you may want to use the Snowflake multi-cluster resource to allow automatic scaling during peak workloads.
- Think about dedicating at least one large virtual warehouse for tactical, high-SLA workloads.
- If you discover a new workload, you can easily add it on demand with Snowflake's ability to instantly provision a new virtual warehouse.

MOVING THE DATA PIPELINE AND ETL PROCESSES

Snowflake is optimized for an ELT approach. However, Snowflake supports many traditional ETL and data integration solutions. We recommend a basic migration of all existing data pipelines and ETL processes to minimize the impact on your project unless you are planning to significantly enhance or modify them. Because testing and data validation are key elements of any changes to the data pipeline, maintaining these processes reduces the need for extensive validation.

Snowflake has worked diligently to ensure that the migration of processes running on traditional ETL platforms is as painless as possible. Native connectors for tools such as Talend and Informatica make the process quick and easy.

Run the data pipeline in both Snowflake and



Netezza during the initial migration to simplify the validation process by enabling a quick comparison of the results from the two systems. When you're sure queries running against Snowflake are producing identical results as queries from Netezza, you can be confident that the migration did not affect data quality and you should see a dramatic improvement in performance.

For data pipelines that require reengineering, you can leverage Snowflake's scalable compute and bulk-loading capabilities to modernize your processes and increase efficiency. You may consider taking advantage of Snowpipe for loading data continuously as it arrives to your cloud storage provider of choice, without any resource contention or performance degradation. Snowflake makes it easy to bring in large data sets and perform transformations at any scale.

CUT OVER

After you migrate your data model, your data, your loads, and your reporting to Snowflake, plan your switch from Netezza to Snowflake.

Here are the steps:

1. Execute a historic, one-time load to move all the existing data.
2. Set up ongoing, incremental loads to collect new data.
3. Communicate the cutover to all Netezza users, so they know what's changing and what to expect.
4. Ensure all development code is checked in and backed up, which is a good development practice.
5. Point production BI reports to pull data from Snowflake.
6. Run Snowflake and Netezza in parallel for a few days and perform verifications.
7. Turn off the data pipeline and access to Netezza for the affected users and BI tools.for the affected users and BI tools.

NEED HELP MIGRATING?

Snowflake is available to accelerate your migration, structure and optimize your planning and implementation activities, and apply customer best practices to meet your technology and business objectives. Snowflake's Engagement, Delivery, and Advisory Services Team deploys a powerful combination of data architecture expertise and advanced technical knowledge of the platform to deliver high performing data strategies, proofs of concept, and migration projects.

Our global and regional solution partners also have extensive experience performing proofs of concept and platform migrations. They offer services ranging from high-level architectural recommendations to manual code conversions. Many Snowflake partners have also built tools to automate and accelerate the migration process.

Whether your organization is fully staffed for a platform migration or you need additional expertise, Snowflake and our solution partners have the skills and tools to accelerate your journey to Snowflake Cloud Data Platform, so you can reap the full benefits of Snowflake quickly. To find out more, please contact the Snowflake sales team or visit Snowflake's Customer Community Lodge.

ient-01-info-customer.html"
="fa fa-address-card"></i>
</a>
</li>
<li>
<a href="#">
<i class="fa fa-sign-out"></i>
</a>
</li>
</ul>
</li>
</ul>

APPENDIX A: MIGRATION TOOLS

Snowflake ecosystem partners have offerings that may help with your migration from Netezza to Snowflake. For more information, or to engage these partners, contact your Snowflake representative. These are just a few:

WIPRO

Wipro has developed a tool to help migration. CDRS Self Service Cloud Migration Tool (patent pending) provides an end-to-end, self-service data migration solution for migrating your on-premises data warehouse to Snowflake. This includes Snowflake-specific optimizations.

WHERESCAPE

Existing WhereScape customers who have built their Netezza warehouse using WhereScape RED can leverage all the metadata in RED to replatform to Snowflake. Since WhereScape has native, optimized functionality for targeting a Snowflake database, it's relatively simple to redeploy the data warehouse objects and ELT code with these steps:

- Change the target platform from Netezza to Snowflake.
- Generate all the required objects (tables, views, sequences, and file formats).
- Deploy and execute the DDL in Snowflake.
- Regenerate all the ELT code for the new target.
- Deploy and execute to continue loading your data in Snowflake.

ANALYTIX DS MAPPING MANAGER

If you need to migrate from one ETL tool or method to a new one, consider Mapping Manager. This tool uses a metadata-driven approach to migrate ETL/ELT code from one tool vendor to another.

APPENDIX B: MIGRATION PREPARATION CHECKLIST

DOCUMENT THE EXISTING SOLUTION

- List of databases that need to be migrated
- List of database objects that need to be migrated
- List of processes and tools that populate and pull data from the existing solution
- List of security roles, users, and permissions
- List of Snowflake accounts that exist or need to be created
- Frequency of security provisioning processes
- Current architecture diagram

DETERMINE A MIGRATION APPROACH

- List of processes to migrate with no change
- List of processes that need reengineering
- List of processes that need to be fixed
- Draft of migration deliverables
- Planned architecture diagram

CAPTURE THE DEVELOPMENT AND DEPLOYMENT PROCESSES

- List of tools that will be introduced with the migration
- List of tools that will be deprecated after the migration
- List of development environments needed for the migration
- List of deployment processes used for the migration

PRIORITIZE DATA SETS FOR MIGRATION

- List of data sets to migrate first
- Method for identifying process dependencies for data sets
- Documentation of process dependencies for data sets

IDENTIFY THE MIGRATION TEAM

- List of migration team members and roles
- Contact information for all team members

DEFINE THE MIGRATION DEADLINES AND BUDGET

- List of business expectations for the migration deadline
- Documented budget allocated for the migration project
- Completed estimate template for Snowflake virtual warehouses

DETERMINE THE MIGRATION OUTCOMES

- List of high-level desired outcomes and assumptions at the completion of the migration
- Documented plan for communicating the migration project wins to stakeholders

APPENDIX C: DATA TYPE CONVERSION TABLE

NETEZZA DATA TYPE	NOTES	SNOWFLAKE DATA TYPE	NOTES
BOOLEAN		BOOLEAN	
CHAR	Up to 64K	CHAR	Up to 16MB
VARCHAR	Up to 64K	VARCHAR	Up to 16MB
NCHAR	Up to 64K	CHAR	Up to 16MB
NVARCHAR	Up to 64K	VARCHAR	Up to 16MB
DATE		DATE	
TIMESTAMP		TIMESTAMP	
TIME		TIME	
TIME WITH TIME ZONE		TIMESTAMP_TZ	
NUMERIC(p,s)		NUMERIC	
REAL		REAL (FLOAT)	
DOUBLE PRECISION		DOUBLE PRECISION (FLOAT)	
INTEGER		INTEGER	
BYTEINT		BYTEINT	
SMALLINT		SMALLINT	
BIGINT		BIGINT	

APPENDIX D: SQL CONSIDERATIONS

These tables contains a representative list of changes needed to ensure that Netezza SQL queries run correctly in Snowflake.

AGE

This function retrieves the interval between two timestamps/dates and provides precision to the same degree as the data. This functionality can be achieved in Snowflake via the DATEDIFF function.

NETEZZA	SNOWFLAKE
SELECT AGE('10-22-2003', '7-6-2002');	SELECT DATEDIFF(day, '2003-10-22', '2002-7-6');

NOW

This returns the current timestamp. This can be achieved via the CURRENT_TIMESTAMP function in Snowflake.

NETEZZA	SNOWFLAKE
SELECT NOW();	SELECT CURRENT_TIMESTAMP();

STRPOS(S,B)

Retrieves the starting position of string “b” in string “s”. Can be achieved via the POSITION function in Snowflake.

NETEZZA	SNOWFLAKE
SELECT STRPOS("fuzzy wuzzy","uzzy");	SELECT POSITION("uzzy","fuzzy wuzzy");

CURRENT_PATH

Retrieves the PATH session variables for the current Netezza session. Similar functionality can be achieved in Snowflake by taking substrings from CURRENT_SCHEMAS.

NETEZZA	SNOWFLAKE
SELECT CURRENT_PATH	SELECT substr(current_schemas(), 3, length(current_schemas()) - 4) as CURRENT_PATH;

ABOUT SNOWFLAKE

Snowflake Cloud Data Platform shatters the barriers that prevent organizations from unleashing the true value from their data. Thousands of customers deploy Snowflake to advance their businesses beyond what was once possible by deriving all the insights from all their data by all their business users. Snowflake equips organizations with a single, integrated platform that offers the only data warehouse built for any cloud; instant, secure, and governed access to their entire network of data; and a core architecture to enable many other types of data workloads, including a single platform for developing modern data applications. Snowflake: Data without limits. Find out more at **snowflake.com**.