

Introduction to Snowflake

Tahun demi tahun kita melihat semakin banyak data tools yang dipasarkan, selalu dengan fitur-fitur baru atau pendekatan baru untuk memecahkan masalah yang sama yang sudah kita kenal. Dan sebagai konsekuensinya, bisnis cenderung berubah dari satu tools ke tools lainnya, contohnya seperti dulu ketika kita ingin memproses data yang cukup besar, kita perlu melakukan parallel computing dimana antar komputer kita hubungkan melalui kabel lan agar terkoneksi satu sama lain, tapi sekarang sudah ada cloud computing dimana kita tidak perlu berurusan dengan hardware lagi, dll.

Apa itu Snowflake?

Dikembangkan pada tahun 2012, Snowflake adalah SaaS (*Software As A Service*) yang dikelola sepenuhnya yang menyediakan platform tunggal untuk *data warehousing*, *data lake*, *data engineering*, *data science*, data application development, dan berbagi serta consume secure data real-time/shared.



You will be able to:

- Access a World of Data and Services
- Gain Modern Data Governance and Security
- Build and Drive Your Business Forward with Data
- Connect locally and globally with Snowflake's platform

Snowflake is a true SaaS offering

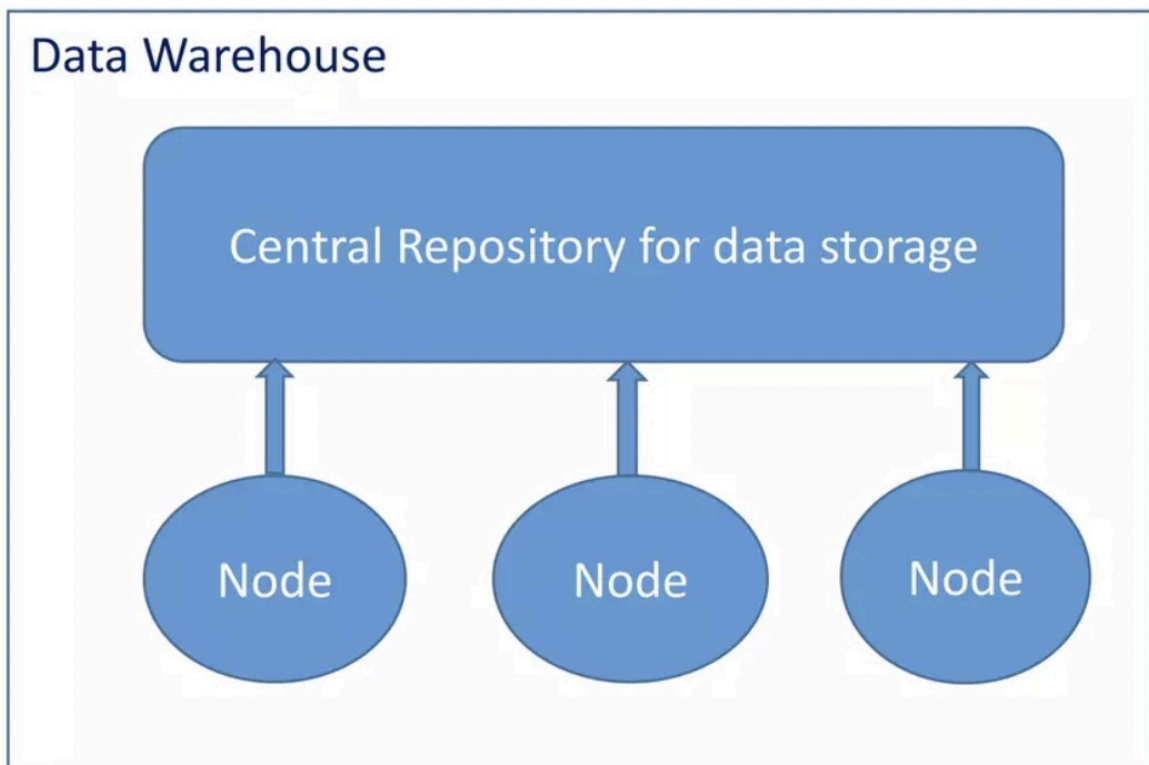
- There is no hardware (virtual or physical) to select, install, configure, or manage.
- There is virtually no software to install, configure, or manage.
- Ongoing maintenance, management, upgrades, and tuning are handled by Snowflake.

Snowflake Architecture

Arsitektur Snowflake adalah gabungan dari arsitektur shared-disk tradisional dan arsitektur database shared-nothing. Sama hal nya dengan arsitektur shared-disk, Snowflake menggunakan repositori data pusat untuk data tersimpan yang dapat diakses dari semua node komputasi di platform. Namun mirip dengan arsitektur shared-nothing, Snowflake memproses query menggunakan cluster komputasi MPP (massively parallel processing) di mana setiap node dalam kluster menyimpan sebagian dari seluruh kumpulan data secara lokal. Pendekatan ini menawarkan kesederhanaan manajemen data dari arsitektur shared-disk, namun dengan performa dan manfaat yang lebih besar dari arsitektur shared-nothing.

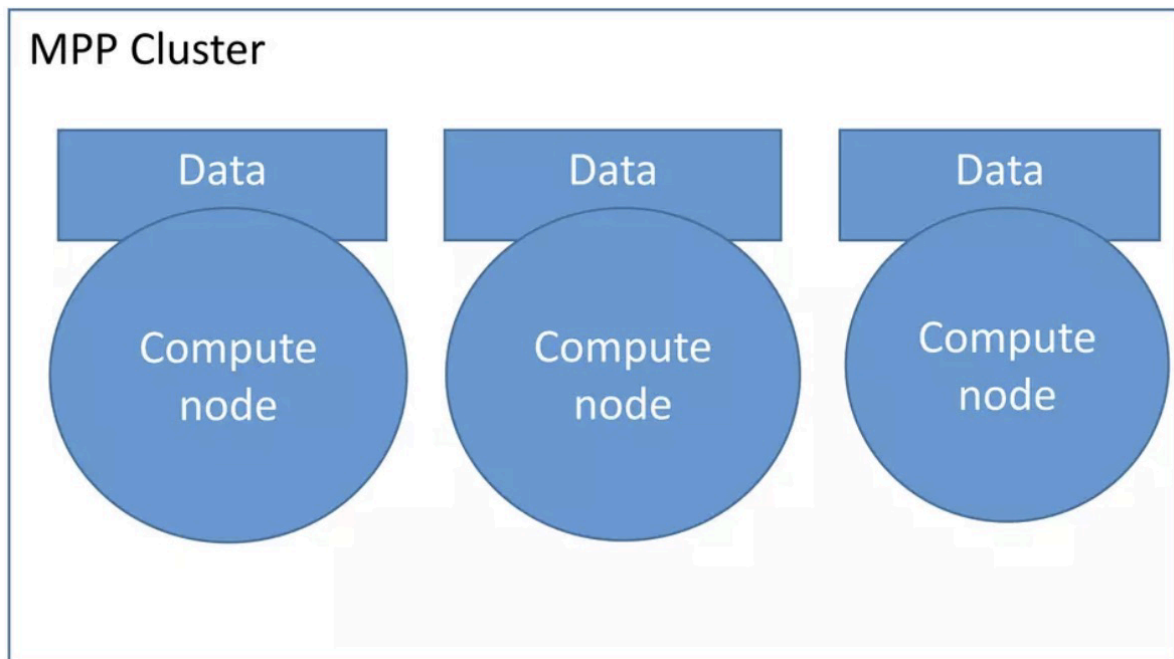
Apabila masih bingung, berikut gambaran nya

Shared Disk Architecture



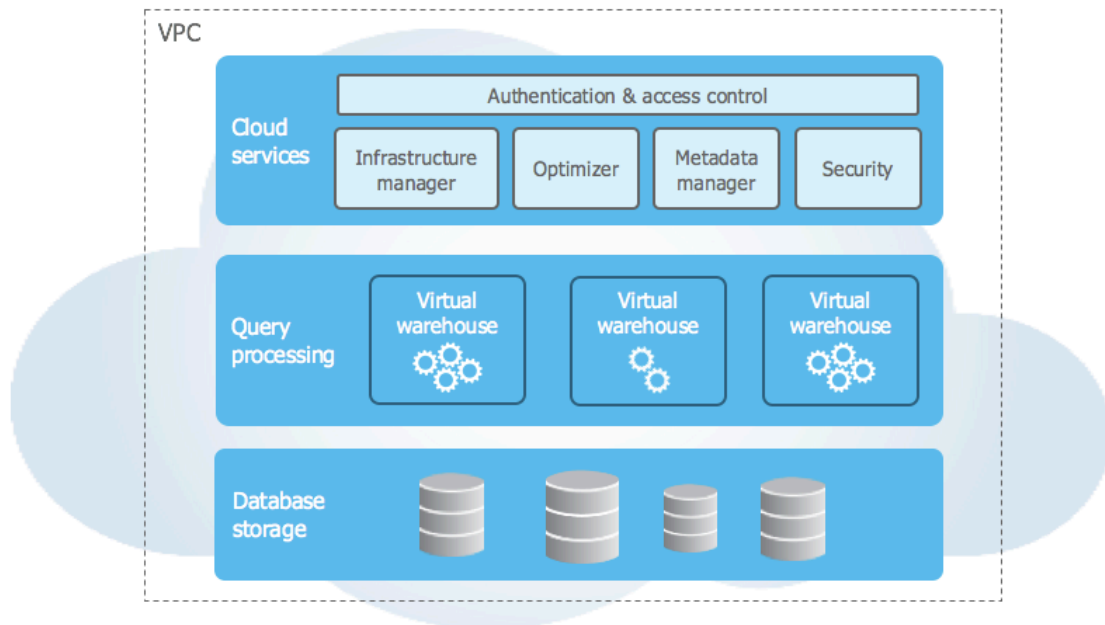
Arsitektur shared-disk adalah arsitektur yang digunakan dalam komputasi terdistribusi dimana node berbagi perangkat disk yang sama tetapi setiap node memiliki memori pribadinya sendiri. Disk memiliki node aktif yang berbagi memori jika terjadi kegagalan. Dalam arsitektur ini, disk dapat diakses dari semua node cluster. Arsitektur ini memiliki kemampuan beradaptasi yang cepat terhadap perubahan beban kerja dan menggunakan robust optimization techniques.

Shared Nothing Architecture



Arsitektur Shared Nothing adalah arsitektur yang digunakan dalam komputasi terdistribusi dimana setiap node bersifat independen dan node yang berbeda saling terhubung oleh suatu jaringan. Setiap node terbuat dari prosesor, memori utama, dan disk. Motif utama arsitektur ini adalah untuk menghilangkan pertikaian antar node. Di sini Node tidak berbagi memori atau penyimpanan. Disk memiliki node individual yang tidak dapat dibagikan. Ia bekerja secara efektif dalam lingkungan volume tinggi dan baca-tulis.

Snowflake Architecture Layer



Database Storage

Saat data di load ke dalam Snowflake, Snowflake mengatur ulang data tersebut ke dalam format kolom internal yang dioptimalkan dan dikompresi. Snowflake menyimpan data yang dioptimalkan ini di penyimpanan cloud.

Snowflake mengelola semua aspek tentang bagaimana data ini disimpan — organisasi, ukuran file, struktur, kompresi, metadata, statistik, dan aspek penyimpanan data lainnya ditangani oleh Snowflake. Objek data yang disimpan oleh Snowflake **tidak terlihat atau dapat diakses secara langsung oleh pengguna**, mereka hanya dapat diakses melalui operasi query SQL yang dijalankan menggunakan Snowflake.

Query Processing

Eksekusi query dilakukan di layer pemrosesan. Snowflake memproses query menggunakan “virtual warehouse”. Setiap virtual warehouse adalah cluster komputasi MPP yang terdiri dari beberapa node komputasi yang dialokasikan oleh Snowflake dari penyedia cloud.

Setiap virtual warehouse adalah cluster komputasi independen yang tidak berbagi sumber daya komputasi dengan virtual warehouse lainnya. Hasilnya, setiap virtual warehouse tidak berdampak pada kinerja virtual warehouse lainnya.

Cloud Services

Layer Cloud Services adalah sekumpulan layanan yang mengkoordinasikan aktivitas di seluruh Snowflake. Layanan ini menyatukan semua komponen Snowflake yang berbeda untuk memproses permintaan pengguna, mulai dari login hingga pengiriman query. Layer cloud services juga berjalan pada compute instances yang disediakan oleh Snowflake dari penyedia cloud.

Services managed in this layer include:

- Authentication
- Infrastructure management
- Metadata management
- Query parsing and optimization
- Access control

Connecting to Snowflake

Snowflake supports multiple ways of connecting to the service:

- A web-based user interface from which all aspects of managing and using Snowflake can be accessed.
- Command line clients (e.g. SnowSQL) which can also access all aspects of managing and using Snowflake.
- ODBC and JDBC drivers that can be used by other applications (e.g. Tableau) to connect to Snowflake.
- Native connectors (e.g. Python, Spark) that can be used to develop applications for connecting to Snowflake.
- Third-party connectors that can be used to connect applications such as ETL tools (e.g. Informatica) and BI tools (e.g. ThoughtSpot) to Snowflake.

How to create Free Account Snowflake

Snowflake menawarkan free trial selama 30 hari bagi pengguna untuk meng explore dan menguji platform data cloud-nya. Berikut langkah untuk membuat akun uji coba gratis di Snowflake:

1. **Visit the Snowflake Website:** Go to the Snowflake website at <https://www.snowflake.com/>.
2. **Navigate to the Pricing Page:** Look for a "Pricing" or "Get Started" section on the website. This might vary, so explore the site to find information about free trials.
3. **Start the Free Trial:** Click on the "Free Trial" or similar button to initiate the sign-up process.
4. **Provide Information:** Fill out the required information to create your account. This may include your name, email address, company information, and other details.
5. **Choose a region for your Snowflake account:** Choose the region basically it's the same like when you choose a region on cloud platform, you can choose the region that is cheapest or has the most low latency with your own region.
6. **Verify Your Email:** After providing your information, you might need to verify your email address. Check your email inbox for a verification message from Snowflake and follow the provided instructions.
7. **Set Up Your Account:** Once your email is verified, you may be prompted to set up your Snowflake account. This could involve creating a username, password, and other account details.
8. **Access the Snowflake Platform:** Once your account is set up, you should be able to log in to the Snowflake platform and start exploring its features.

How to add data on Snowflake

To add data to Snowflake, you generally follow these steps:

1. **Create a Database:** Log in to your Snowflake account, navigate to the "Databases" tab, and create a new database where you want to store your data.
2. **Create a Schema:** Within your database, create a schema to organize your tables.
3. **Create a Table:** Design your table structure and create a table within your chosen schema. You can define columns, data types, and constraints.
4. **Load Data into the Table:** You can load data into your Snowflake table using various methods:
 - **Snowflake Web Interface:** You can use the Snowflake web interface to upload data from local files.
 - **Snowflake Copy Command:** Use the `COPY INTO` command to load data from external storage like AWS S3 or Azure Blob Storage.
 - **Snowpipe:** Set up Snowpipe to automatically load streaming data into your Snowflake tables.

Here's a simplified example using SQL commands:

```
sql
-- Create a Database
CREATE DATABASE my_database;

-- Use the Database
USE DATABASE my_database;

-- Create a Schema
CREATE SCHEMA my_schema;

-- Use the Schema
USE SCHEMA my_schema;

-- Create a Table
CREATE TABLE my_table (
    column1 INT,
    column2 STRING,
    column3 DATE
);

-- Load Data into the Table (assuming a CSV file)
COPY INTO my_table
FROM 's3://your-s3-bucket/your-data.csv'
```

```
CREDENTIALS=(AWS_KEY_ID='<your-access-key-id>'
AWS_SECRET_KEY='<your-secret-access-key>')
FILE_FORMAT=(TYPE='CSV' FIELD_OPTIONALLY_ENCLOSED_BY='"'
SKIP_HEADER=1);

-- Query the Data
SELECT * FROM my_table;
```

Make sure to replace placeholders like '[your-s3-bucket/your-data.csv](#)' and the AWS credentials with your actual values.

In addition to the methods mentioned earlier, Snowflake supports various data loading options. Here are a few more alternatives:

1. **Snowflake Data Loading Tools:**
 - SnowSQL: Snowflake's command-line client that allows you to run SQL commands, including loading data.
 - Snowflake Data Pump: A tool for bulk data loading into Snowflake using parallel processing.
2. **Third-Party ETL Tools:**
 - You can use popular ETL (Extract, Transform, Load) tools like Talend, Informatica, or Apache NiFi to integrate and load data into Snowflake.
3. **Snowflake Connectors:**
 - Snowflake provides connectors for various programming languages and frameworks, such as Python, JDBC, ODBC, and more. You can use these connectors to programmatically load data.
4. **REST API:**
 - Snowflake has a REST API that allows you to perform various operations, including loading data. You can use this API to automate data loading tasks.

Each of these methods has its own use case and advantages. Choose the one that best fits your requirements and the tools you are comfortable working with.