

# Data Integration

## Skema Star dan Snowflake

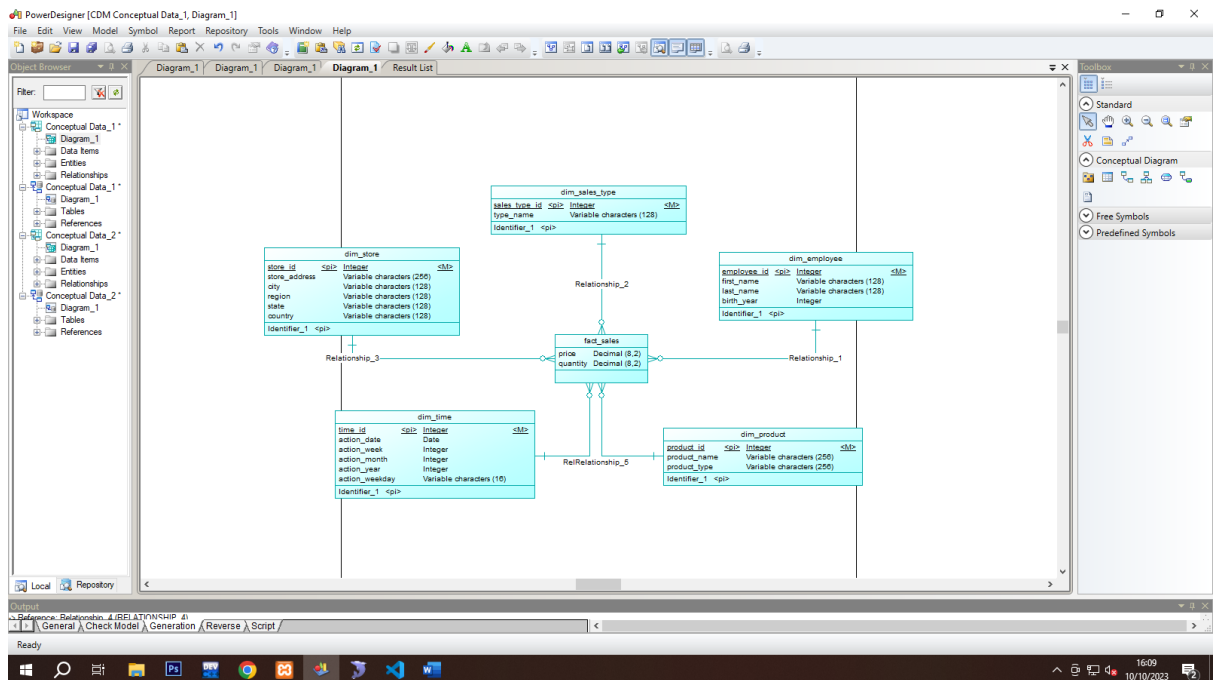
NAMA : Made Adhitya Dwipayana

NIM : 2201010366

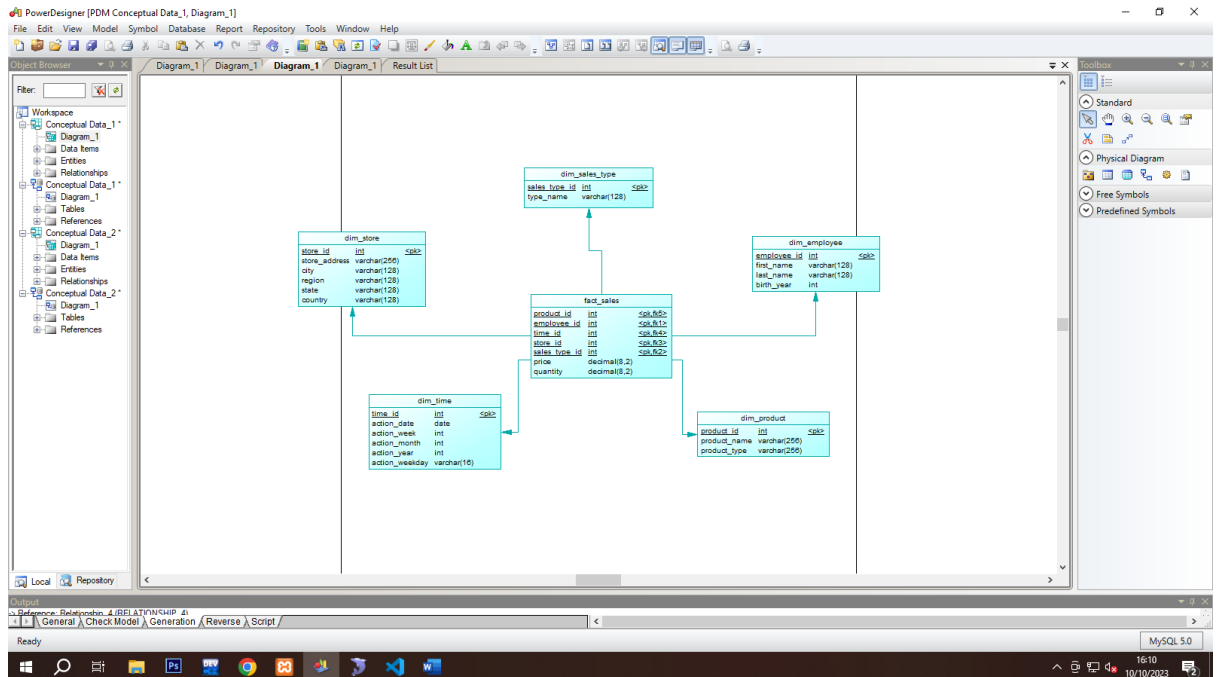
KELAS : U

## STAR

- CDM



- PDM



- SS Database

The screenshot shows the SQLyog Ultimate interface with a MySQL database schema. The schema includes the following tables and their attributes:

- dim\_employee**: employee\_id (PK), first\_name (varchar(128)), last\_name (varchar(128)), birth\_year (int).
- dim\_product**: product\_id (PK), product\_name (varchar(256)), product\_type (varchar(256)).
- dim\_sales\_type**: sales\_type\_id (PK), type\_name (varchar(128)).
- dim\_store**: store\_id (PK), store\_address (varchar(256)), city (varchar(128)), region (varchar(128)), state (varchar(128)), country (varchar(128)).
- dim\_time**: time\_id (PK), action\_date (date), action\_week (int), action\_month (int), action\_year (int), action\_weekday (varchar(16)).
- fact\_sales**: product\_id (FK), employee\_id (FK), time\_id (FK), store\_id (FK), sales\_type\_id (FK), price (decimal(8,2)), quantity (decimal(8,2)).

The screenshot also shows the SQLyog Ultimate interface with a query window displaying the following SQL code:

```

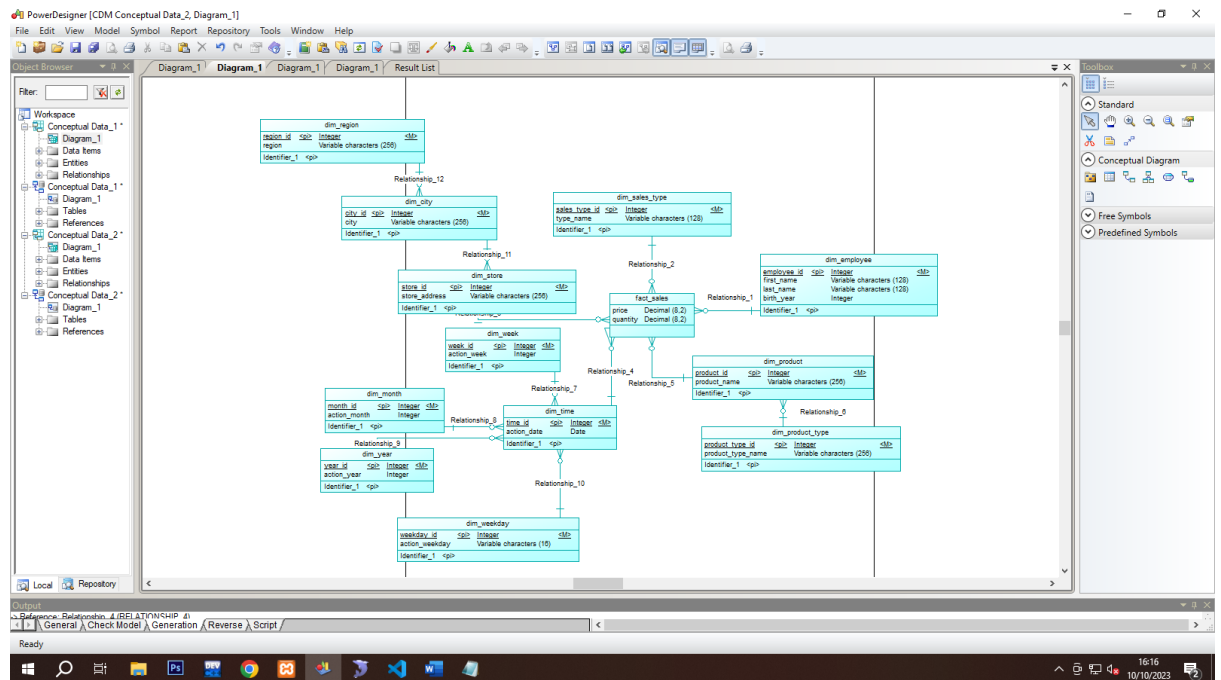
1  -- DEMO name: MySQL 5.0
2  -- Created on: 10/10/2023 15:43:02
3
4
5
6
7  DROP TABLE IF EXISTS DIM_EMPLOYEE;
8
9  DROP TABLE IF EXISTS DIM_PRODUCT;
10
11  DROP TABLE IF EXISTS DIM_SALES_TYPE;
12
13  DROP TABLE IF EXISTS DIM_STORE;
14
15  DROP TABLE IF EXISTS DIM_TIME;
16
17  DROP TABLE IF EXISTS FACT_SALES;
18
19  -- Table: DIM_EMPLOYEE
20  --
21  --
22  CREATE TABLE DIM_EMPLOYEE
23  (
24    EMPLOYEE_ID INT NOT NULL,
25    FIRST_NAME VARCHAR(128),
26    LAST_NAME VARCHAR(128),
27    BIRTH_YEAR INT,
28    PRIMARY KEY (EMPLOYEE_ID)
29  );
30
31  -- Table: DIM_PRODUCT
32  --
33  --
34  CREATE TABLE DIM_PRODUCT
35  (
36    PRODUCT_ID INT NOT NULL,
37    PRODUCT_NAME VARCHAR(256),
38    PRODUCT_TYPE VARCHAR(256),
39    PRIMARY KEY (PRODUCT_ID)
40  );

```

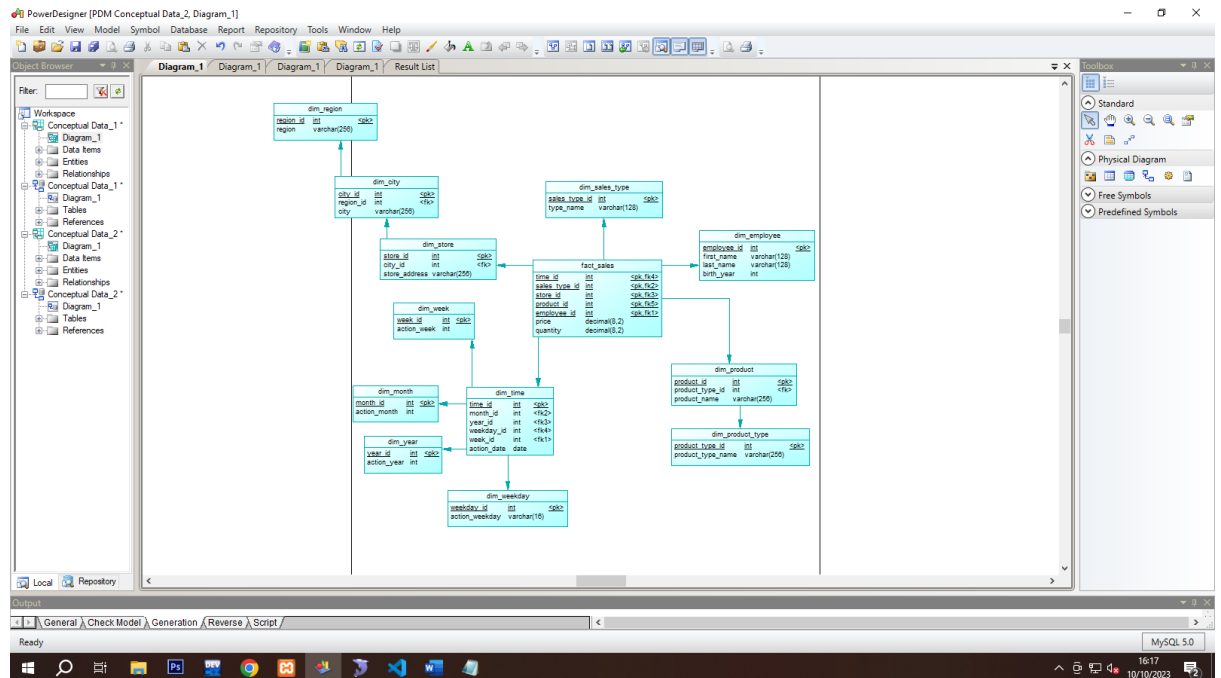
The bottom of the screenshot shows the execution results: 0 row(s) affected, 1 warning(s), Execution Time: 00:00:00.000.

# SNOWFLAKE

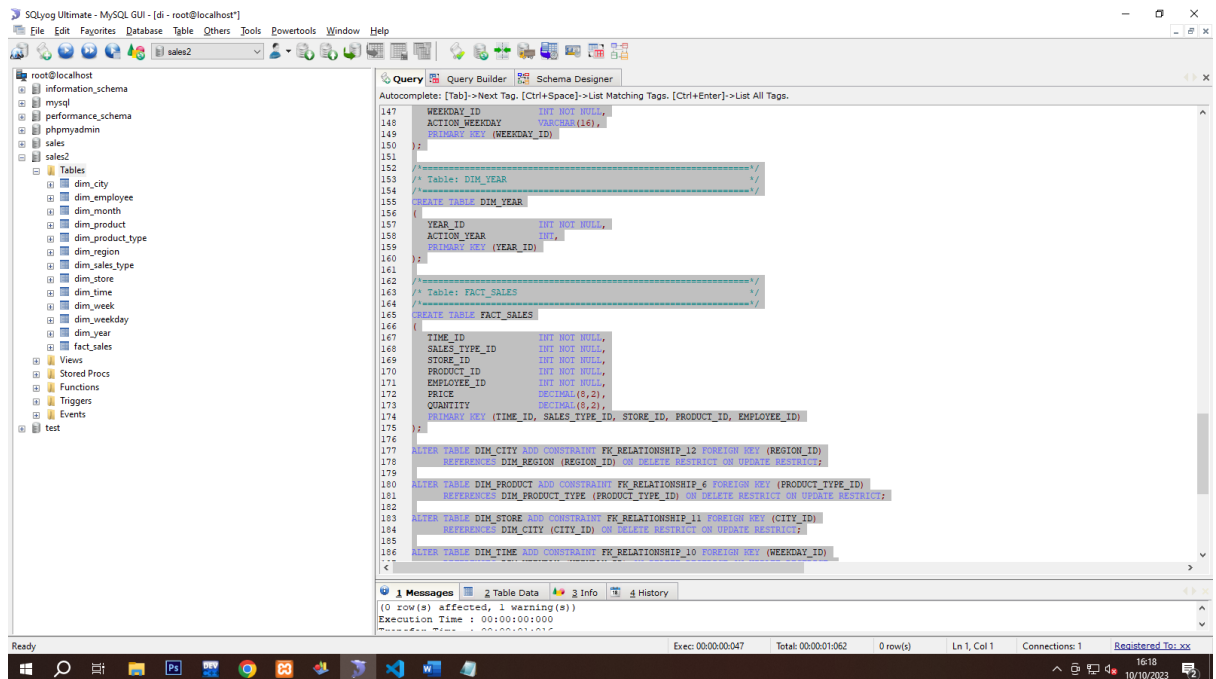
- CDM



- PDM



- SS Database



## PERBEDAAN SKEMA STAR DAN SNOWFLAKE

Skema bintang (star schema) dan skema salju (snowflake schema) adalah dua jenis struktur penyimpanan data yang digunakan dalam perancangan basis data berdimensi (data warehousing) untuk mendukung analisis bisnis dan pelaporan. Perbedaan utama antara keduanya:

### 1. Struktur Hierarki:

- Star Schema: Star schema memiliki struktur yang sederhana dan mudah dimengerti. Data disimpan dalam satu tabel fakta (fact table) yang terhubung langsung dengan tabel-tabel dimensi (dimension tables) tanpa ada tingkat hierarki atau normalisasi yang kuat. Ini membuatnya efisien dalam kueri dan pelaporan.

- Snowflake Schema: Snowflake schema, di sisi lain, memiliki struktur yang lebih kompleks. Tabel dimensi dalam snowflake schema dapat dinormalisasi lebih lanjut menjadi beberapa tabel terkait. Sebagai hasilnya, tingkat normalisasi yang lebih tinggi dapat mengurangi redundansi data, tetapi juga membuat skema ini lebih sulit untuk dipahami dan dapat mengakibatkan kueri yang lebih rumit.

### 2. Kinerja:

- Star Schema: Star schema cenderung memiliki kinerja yang lebih baik dalam hal kueri dan pelaporan karena tidak ada normalisasi berlebihan yang harus diatasi. Kueri seringkali dapat dieksekusi lebih cepat.

- Snowflake Schema: Snowflake schema dapat memerlukan lebih banyak operasi gabungan (joins) untuk menggabungkan data dari tabel yang ternormalisasi, yang dapat mengurangi kinerjanya dalam beberapa kasus.

### 3. Kesulitan dalam Perancangan dan Pemeliharaan:

- Star Schema: Star schema lebih mudah dirancang, dimengerti, dan dipelihara karena memiliki struktur yang lebih sederhana. Ini cocok untuk aplikasi dengan kebutuhan pelaporan dan analisis yang cukup sederhana.

- Snowflake Schema: Snowflake schema dapat menjadi lebih rumit dalam perancangan dan pemeliharaan karena tingkat normalisasi yang lebih tinggi dan jumlah tabel yang lebih besar. Ini sering digunakan ketika ada kebutuhan untuk menghindari redundansi data secara ekstrem atau ketika data harus terstruktur dalam hierarki yang mendalam.

### 4. Ruang Penyimpanan:

- Star Schema: Star schema dapat menghasilkan lebih sedikit ruang penyimpanan karena data cenderung memiliki redundansi yang lebih tinggi.

- Snowflake Schema: Snowflake schema dapat memerlukan lebih banyak ruang penyimpanan karena normalisasi yang lebih kuat dapat mengurangi redundansi data, tetapi pada saat yang sama, membuat struktur yang lebih kompleks.

Pemilihan antara star schema dan snowflake schema harus didasarkan pada kebutuhan spesifik dari proyek data warehousing dan analisis bisnis Anda. Jika kinerja dan keterbacaan kode kueri adalah prioritas utama, maka star schema mungkin menjadi pilihan yang lebih baik. Namun, jika efisiensi penyimpanan data atau normalisasi yang ketat diperlukan, maka snowflake schema dapat lebih sesuai.

## KELEBIHAN

- Skema Bintang (Star Schema)

1. Kinerja yang Baik: Skema bintang memiliki kinerja yang baik dalam hal kueri dan pelaporan. Koneksi langsung antara tabel fakta dan tabel dimensi memungkinkan kueri dieksekusi dengan cepat, terutama pada basis data yang besar.

2. Mudah Dipahami: Struktur bintang yang sederhana membuatnya mudah dipahami oleh pengembang dan pengguna. Ini memungkinkan pengembangan dan pemeliharaan yang lebih cepat.

3. Efisien untuk Data Warehousing: Skema bintang biasanya digunakan untuk lingkungan data warehousing di mana tujuannya adalah analisis bisnis dan pelaporan.

4. Pemahaman yang Baik: Skema bintang cocok untuk aplikasi dengan kebutuhan pelaporan dan analisis yang cukup sederhana. Ini menyediakan pandangan yang jelas tentang data yang diperlukan untuk analisis bisnis.

- Kelebihan Skema Salju (Snowflake Schema):

1. Normalisasi yang Kuat: Skema salju memiliki tingkat normalisasi yang lebih tinggi, mengurangi redundansi data. Ini menghemat ruang penyimpanan dan memungkinkan pemeliharaan data yang lebih efisien.

2. Fleksibilitas: Struktur snowflake lebih fleksibel dalam mengakomodasi perubahan kebutuhan bisnis karena dimensi atau atribut dapat diperbarui tanpa memengaruhi tabel fakta.

3. Integritas Data yang Baik: Normalisasi dalam skema salju dapat membantu menjaga integritas data dengan lebih baik karena setiap nilai hanya disimpan sekali.

## KEKURANGAN

- Skema Bintang (Star Schema):

1. Redundansi Data: Skema bintang dapat menghasilkan redundansi data karena beberapa informasi mungkin disimpan dalam tabel dimensi yang berbeda. Hal ini dapat menghabiskan ruang penyimpanan.

2. Keterbatasan Normalisasi: Skema bintang umumnya tidak memprioritaskan normalisasi data, yang dapat mengakibatkan potensi masalah integritas data.

3. Kurang Fleksibel: Struktur bintang mungkin kurang fleksibel dalam mengakomodasi perubahan kebutuhan bisnis karena perubahan dalam dimensi atau atribut yang diperlukan dapat memerlukan perubahan skema yang signifikan.

- Skema Salju (Snowflake Schema):

1. Kinerja yang Kurang Baik: Snowflake schema cenderung memiliki kinerja yang lebih rendah dalam hal kueri karena memerlukan lebih banyak operasi gabungan (joins) yang kompleks.

2. Kompleksitas Perancangan: Struktur snowflake yang lebih kompleks dapat membuat perancangan dan pemeliharaan lebih sulit dan membutuhkan lebih banyak usaha.

3. Kesulitan Pemahaman: Skema salju mungkin sulit dipahami oleh pengguna akhir yang tidak memiliki pengetahuan teknis yang mendalam tentang struktur data.