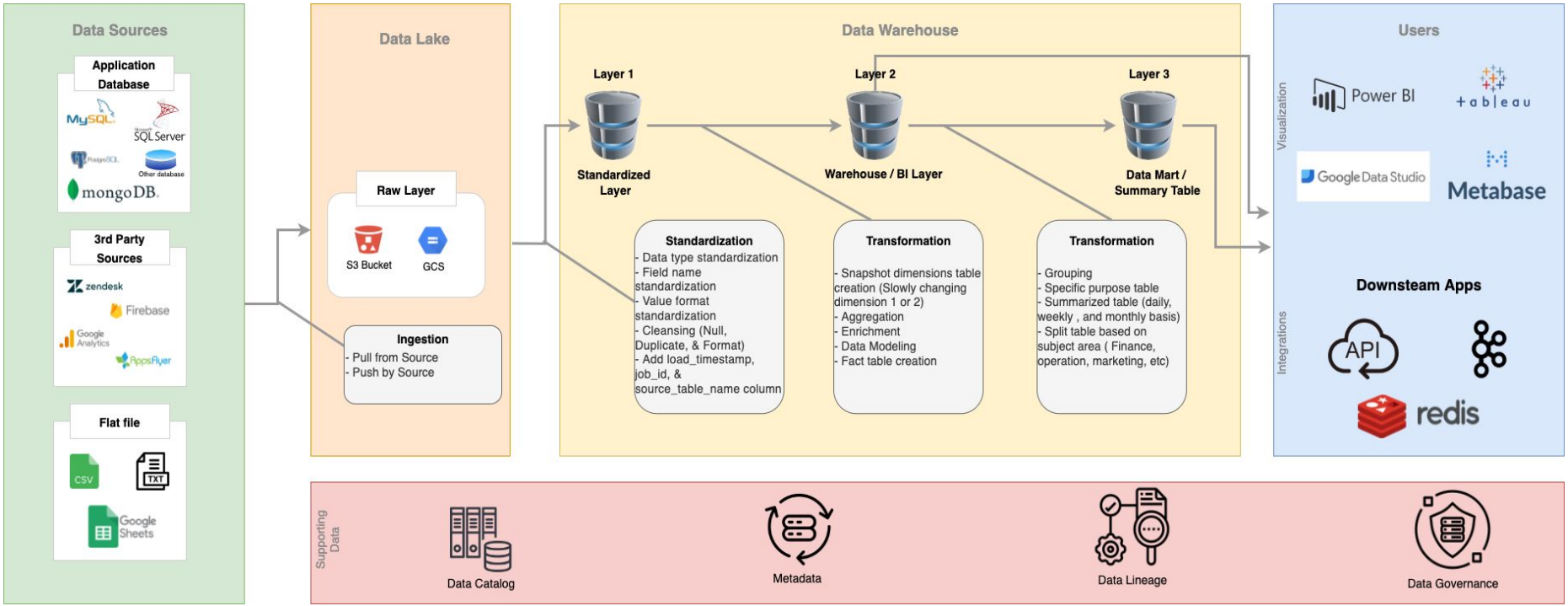
High Level Design Data Warehouse

Noverdy Safrizal
Linkedin

Diagram Arsitektur Data Warehouse



Detail Layering Description

Layer Name	Description	Data Model	Accessible For
Raw Layer	Transactional Database or other source As-is	3NF	DE & DWH Team
Layer 1	Standardized Layer	3NF	Data Team
Layer 2	Modeling & Aggregation Layer	Star/Snowflake Schema*	Data Team
Layer 3	Summarized / Specific Purpose Layer	-	Business Team / Table per department

Raw Layer

- Proses awal penarikan data ke dalam data lake di storage tertentu
- Raw layer dirancang untuk menjaga data sumber, dan berfungsi sebagai data lake dari database transaksional sistem atau sumber data lainnya
- Penarikan data dilakukan sesuai kebutuhan *freshness* data melalui proses *batching* atau *live streaming real-time data*
- Penambahan kolom `load_timestamp` di setiap tabel yang dimasukan ke data lake sesuai `timestamp` saat proses penyerapan data berjalan

Layer 1 / Standardize Layer

- Setelah data source ke dalam data lake maka dilakukan proses standarisasi data
- Proses standarisasi data meliputi beberapa proses sehingga bertujuan dapat *self-explanatory & business-friendly way*
    1. Proses *cleansing* atau *data quality checking* (*renamed field with naming conventions*, *change data type*, *NULLs handling*, *Check uniqueness*, *format value*, *add metadata like ingestion timestamp*, *duplication handling*)
    2. Penambahan partition dan cluster di tiap table, sehingga user ketika menggunakan table di layer lebih cepat & efisien
    3. Penambahan *column descriptions* sehingga ketika user tim data lain lebih mudah dalam melakukan explorasi table di layer 1
- Penamaan *table* selalu sama dengan raw layer sehingga memudahkan pelacakan data source

Layer 2 / Warehouse Layer

- Setiap table di data warehouse memiliki *surrogate keys* sebagai *primary & foreign key*
- Implementasi SCD (*Slowly Changing Dimension*) type 2 di setiap *dimension table*
- Modelling menggunakan star/snowflake schema tergantung kebutuhan dan kompleksitas table
- *Fact table creation* sesuai kebutuhan
- *Data quality checking* di setiap dimension dan fact table (Pengecekan *accuration*, *completeness*, *timeliness*, *consistency*)
- Standarnya proses batching dilakukan dengan data update D-1 atau sesuai kebutuhan dari *end-user* terhadap *freshness data*
- *Column Naming convention* (Setiap kolom harus mengikuti conventions seperti di layer 1)
- Setiap column memiliki *column descriptions* sehingga user mendapatkan informasi terhadap measure atau dimension yang akan digunakan

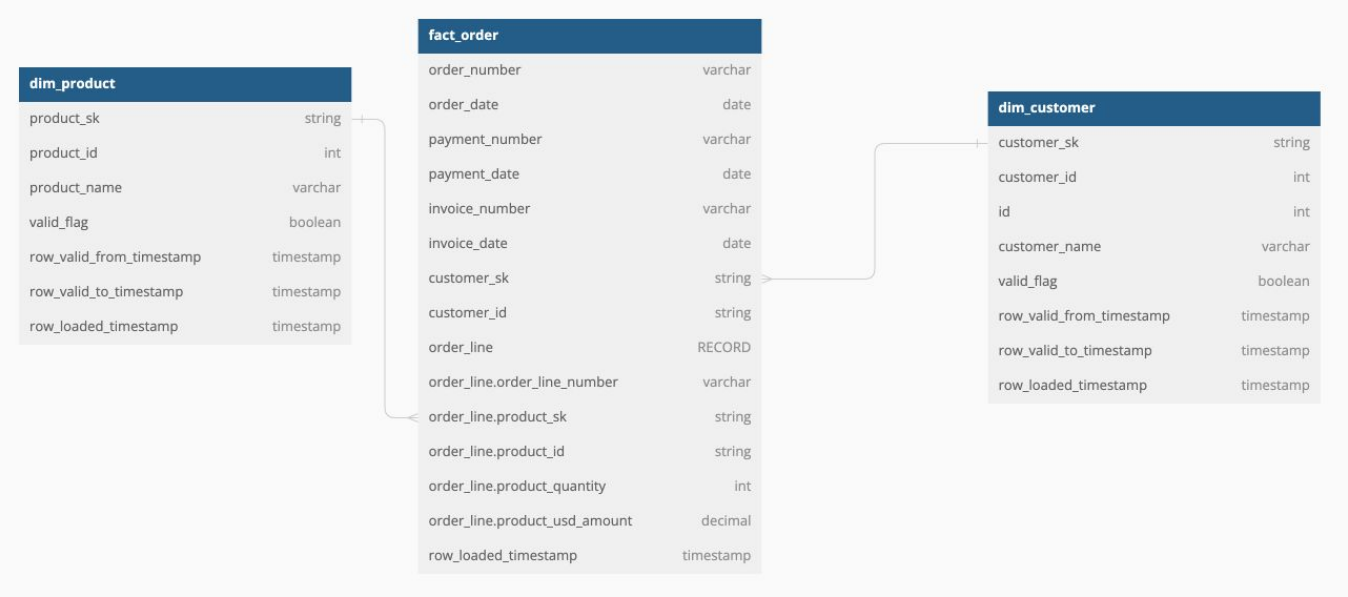
Layer 3 / Mart Layer

- Proses *ingestion* dari mart layer berasal dari table warehouse yang di agregasi ke *daily*, *weekly*, atau *monthly basis* sesuai kebutuhan dari end-user dalam penggunaan table
- Tiap mart layer dipisahkan ke beberapa dataset sesuai dengan department/ topic / subject area
- Setiap mart layer juga memiliki *same conventions* dengan layer sebelumnya
- *Access data level* mulai diterapkan di layer ini sesuai dengan user tiap *business unit*
- Setiap metrics atau formula akan di tuliskan di *column descriptions* guna memberikan informasi pelengkap *measure*
- *Self-service* akan menjadi salah satu *goal* dalam penggunaan table di data mart di tiap *business unit*

Supporting Component of Data

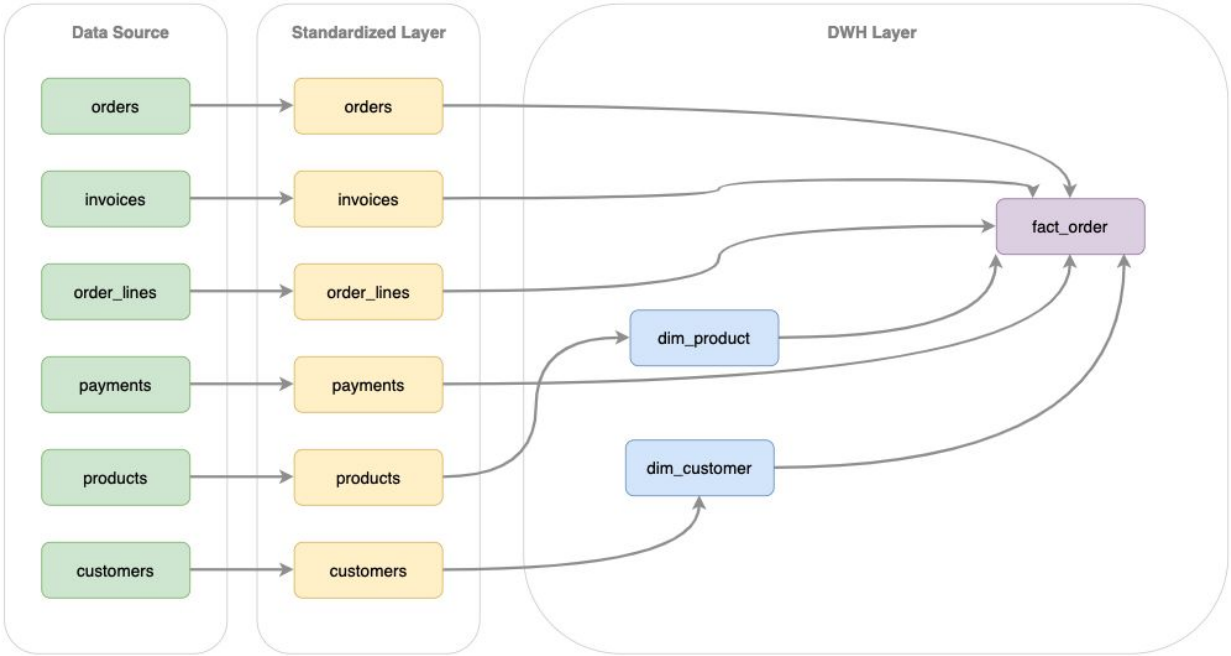
Feature	Description	Implemented for
Data Catalog	Dictionary each table that implemented on data warehouse	each table from standardized layer - mart layer
Metadata	Additional Column related to information about that table. And table related to audit & profiling	each table from raw layer - mart layer for every ingest new record.
Data Lineage	Diagram Flow of source coming from for each tables	each table from raw layer - mart layer, that will describe lineage of data
Data Governance	Access Data level, data quality checking, data security	each table from standardized layer - mart layer.

ERD Data warehouse
*star schema



Check on [this link]

Data Lineage



Dimension Table

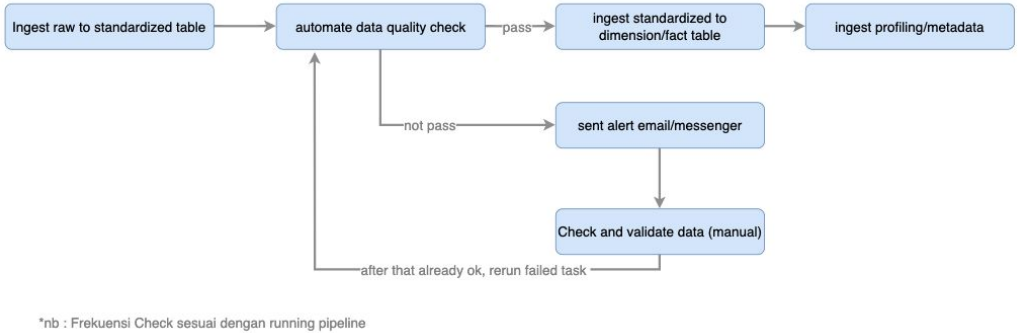
Dalam structure dimensional table disitu saya menggunakan SCD type 2. Dimana metode ini menyimpan tambahan baris baru dengan nilai yang baru. Dan *historical* disimpan dan dapat digunakan kapanpun diperlukan. User dapat menggunakan table dengan menggunakan **row_valid_flag = TRUE** untuk mendapatkan latest information terhadap unique id yang ada. Jika ada kebutuhan analisa dengan information sesuai dengan historical tanggal bergeraknya data maka user dapat menggunakan data dengan **in range** menggunakan **row_valid_from_timestamp** & **row_valid_to_timestamp.**

Use case dapat digunakan sesuai dengan kebutuhan analisa data seperti data historical maupun latest value di dimensional model tersebut dengan menerapkan SCD type 2 ini.

Data Quality Check

List data quality check - accuracy

- PK *uniqueness*
- *NULL values check*
- *Duplicate value check*
- *Accepted value check*
- *Data type check*
- Jumlah record antar source dan dwh tidak ada *discrepancies*
- *Value* measure check



*nb : Frekuensi Check sesuai dengan running pipeline

Pengecekan data quality ini dapat di setiap process ingestion berlangsung ketika data pipeline *running*. Dengan menambahkan process check ketika didalam workflow pipeline.

Komponen pendukung

Komponen pendukung tidak hanya sebagai pelengkap akan tetapi dapat berfungsi dengan baik jika diterapkan bersamaan dengan development data warehouse.

1. Metadata. Ex: Setiap table di masing-masing layer ditambahkan metadata seperti job_id, load_timestamp & data_source_name. Akan memudahkan ketika proses pengecekan, explorasi data dan validasi data ketika ada kebutuhan reconciliation.
2. Data Lineage. Ex: Setiap pembuatan data warehouse dilakukan joining table dalam proses modelling dimensional dan fact table, dengan adanya data lineage sangat membantu user khususnya tim BI untuk explorasi jalur data source yang mana ketika proses pembuatan *insight report* atau kebutuhan *ad-hoc analysis*
3. *Data Catalog*. Ex: Dengan adanya data catalog yang lengkap maka akan memudahkan user dalam proses explorasi terkait definisi kolom, data type, PII tertentu tiap kolom, lebih efisiensi dan baik dalam *customer experience*
4. Data Governance. Ex: Process data quality check termasuk bagian dari data governance yang sangat penting, dan juga data access level akan berguna ketika mulai *sharing access* di level mart table untuk tiap *business unit team*

Resiko komponen pendukung tidak diterapkan

- Data Catalog. Ketika tidak menerapkan adanya fitur data catalog maka, tiap user akan kesulitan dalam eksplorasi data, dan membuang waktu dalam eksplorasi data. Dan butuh sharing knowledge detail antara DWH developer & user
- Data Lineage. User akan sulit mencari tahu sumber data tiap DWH layer yang dibuat, karena sudah dalam bentuk agregasi dan membuang waktu untuk eksplorasi jalur data pada DWH table berasal dibandingkan proses data analis
- Data governance. Data yang tidak akurat maka dapat menyebabkan *discrepancy* di tiap value ataupun measure. Akan menambah proses *effort* dalam rekonsiliasi manual antar source dan target DWH table.

Regulasi

- Implementasi data catalog yang baik dan detail untuk tiap layer di data warehouse
- Implementasi dan knowledge sharing terhadap data lineage atau penerapan fitur open source framework (dbt doc, talent, dll)
- Data quality checking setidaknya diterapkan dari ingest ke *standardization layer* sampai *mart layer*. Adanya alert reminder ketika *error* atau *failure* sehingga bisa cepat di *acknowledge issue* tersebut