Databricks Nowoczena Platforma Danych

Databricks to ujednolicona platforma do zarządzania danymi i analityki, która łączy najlepsze cechy hurtowni danych (data warehouse) z elastycznością jeziora danych (data lake), tworząc architekturę lakehouse. Umożliwia efektywne przetwarzanie danych na dużą skalę, analitykę i rozwój sztucznej inteligencji.

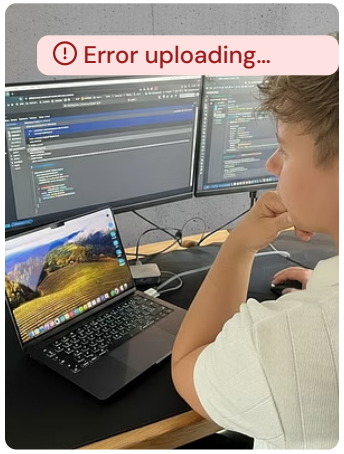
Krzysztof Burejza

Azure Data Engineer

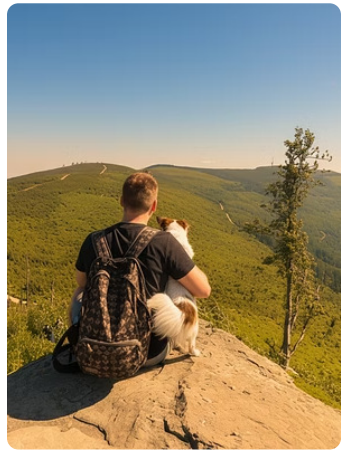
Jako Azure Data Engineer specjalizuję się w pełnym cyklu życia danych: od analizy, ekstrakcji, przetwarzania i integracji, po budowanie solidnych data warehosue, lakehouse i rozwiązań opartych na danych.

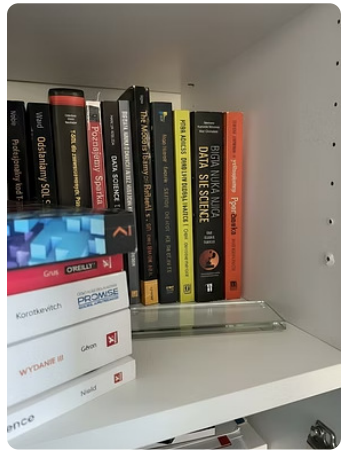
Moje Pasje poza Pracą

Poza światem danych, znajduję równowagę i inspirację w różnorodnych aktywnościach.









Agenda

Fundamentals & Exploration

Wprowadzenie do Databricks Lakehouse

Architektura Lakehouse, Databricks, Unity Catalog (różnice z Hive Metastore), praca z notebookami i klastrami.

Import i eksploracja danych

Wczytywanie z CSV, JSON, Parquet, Delta. Konfiguracja readerów, podstawowa eksploracja danych i typów.

Podstawowe transformacje (SQL i PySpark)

Wybór kolumn, tworzenie nowych pól, filtrowanie, sortowanie, agregacje i ich odpowiedniki w SQL.

Czyszczenie i jakość danych

Obsługa nulli, duplikatów, castowanie typów, standaryzacja dat i tekstu, wykrywanie problemów jakościowych.

Widoki i przepływy pracy

Różnice między VIEW, TABLE, DELTA TABLE. Widoki tymczasowe/trwałe. Rejestracja tabel w Unity Catalog. Prosty workflow notebooków i Jobs.

Lakehouse & Delta Lake

Fundamenty Delta Lake

ACID, log transakcji, zarządzanie schematem, operacje CRUD i podstawy optymalizacji.

Architektura Medallion

Warstwy Bronze/Silver/Gold, zasady projektowania i aspekty jakości danych.

Ładowanie Danych

Metody COPY INTO, Auto Loader, Structured Streaming i obsługa ewolucji schematu.

Pipeline B/S/G

Projektowanie i implementacja potoków danych od warstwy raw do Gold.

Praktyki Optymalizacji

Pushdown predykatów, przycinanie danych i efektywne zarządzanie plikami.

Transformation, Governance & Integrations

Zaawansowane transformacje w PySpark

Nauka window functions, agregacji kroczących, pracy ze strukturami złożonymi (arrays, structs, explode) oraz operacji na JSON i funkcjach datowych/czasowych.

Wprowadzenie do Lakeflow

Poznaj deklaratywne pipeline'y SQL, materialized views, streaming tables, expectations (warn/drop/fail), event log i lineage na poziomie tabel.

Orkiestracja w Databricks Jobs

Praca z multitask jobs, typami tasków (notebook, SQL, DLT, dbt), zależnościami między zadaniami, parametryzacją, monitoringiem i alertowaniem.

Governance z Unity Catalog

Omówienie katalogów, schematów, tabel, Volumes, modelu uprawnień (GRANT/REVOKE), maskowania, row-level security, lineage i audytu aktywności.

Integracje z BI i ML

Wykorzystanie Power BI (Direct Lake/Direct Query), warstwy Gold jako źródła raportów i datasetów ML, a także podstawy MLflow i Feature Store.

Architektura Analityczna Sprzedaży Detalicznej

Od surowych danych transakcyjnych do modelu gwiazdy w Delta Lake

Problem biznesowy: co chcemy policzyć?

Kontekst biznesowy – Sieć handlowa "MegaMarket"

Nasza sieć detaliczna, operująca w ponad **50 sklepach w całej Polsce**, generuje ogromne ilości danych transakcyjnych, danych o klientach i produktach. Codziennie przetwarzamy **miliony rekordów** pochodzących z różnorodnych systemów operacyjnych.

Brak jednolitego, zintegrowanego modelu analitycznego sprawia, że kluczowe pytania biznesowe pozostają bez odpowiedzi lub ich uzyskanie zajmuje dni, a nawet tygodnie, angażując cenne zasoby IT i analityków. To bezpośrednio wpływa na naszą konkurencyjność.

Konkretne wyzwania:

- **Zarządzanie zapasami:** Nie jesteśmy w stanie efektywnie zarządzać stanami magazynowymi w czasie rzeczywistym, co prowadzi do nadmiernych zapasów w jednych sklepach i braków produktów w innych (tzw. "stock-outs"). Jak optymalizować dostawy, aby zminimalizować straty i zmaksymalizować sprzedaż?
- **Segmentacja klientów:** Trudno jest nam precyzyjnie segmentować klientów i personalizować oferty, ponieważ brakuje nam spójnego widoku na ich historię zakupów we wszystkich kanałach (online i offline). Jakie grupy klientów są najbardziej dochodowe i jak zwiększyć ich lojalność?
- **Prognozowanie sprzedaży:** Obecne prognozy sprzedaży są często niedokładne ze względu na rozproszone dane i brak zaawansowanych modeli. Jak możemy dokładnie przewidywać sprzedaż dla poszczególnych produktów i lokalizacji, biorąc pod uwagę czynniki sezonowe i promocyjne?

Kluczowe pytania analityczne

Sprzedaż dzienna

Jaka jest dzienna sprzedaż w podziale na sklepy, segmenty klientów i kategorie produktów?

Metody płatności

Które metody płatności dominują i jak wpływają na wartość koszyka zakupowego?

Najlepsi klienci

Którzy klienci generują największą sprzedaż i w jakich lokalizacjach dokonują zakupów?

Główne wyzwanie

Dane w osobnych plikach bez spójnego modelu

Dane klientów, produktów i transakcji znajdują się w oddzielnych plikach CSV, Parquet i JSON, co uniemożliwia kompleksową analizę bez wcześniejszego zintegrowania i transformacji.

Źródła danych: przegląd plików

customers.csv

Dane klientów z systemu CRM zawierające profile, segmentację i informacje kontaktowe.

products.parquet

Katalog produktów z systemu ERP z informacjami o cenach, kosztach i kategoriach.

orders_*.json

Transakcje sprzedaży z systemów POS w trybie strumieniowym, zawierające szczegóły zamówień.

Wszystkie dane lądują do storage i są przetwarzane w Databricks Lakehouse z wykorzystaniem Delta Live Tables.

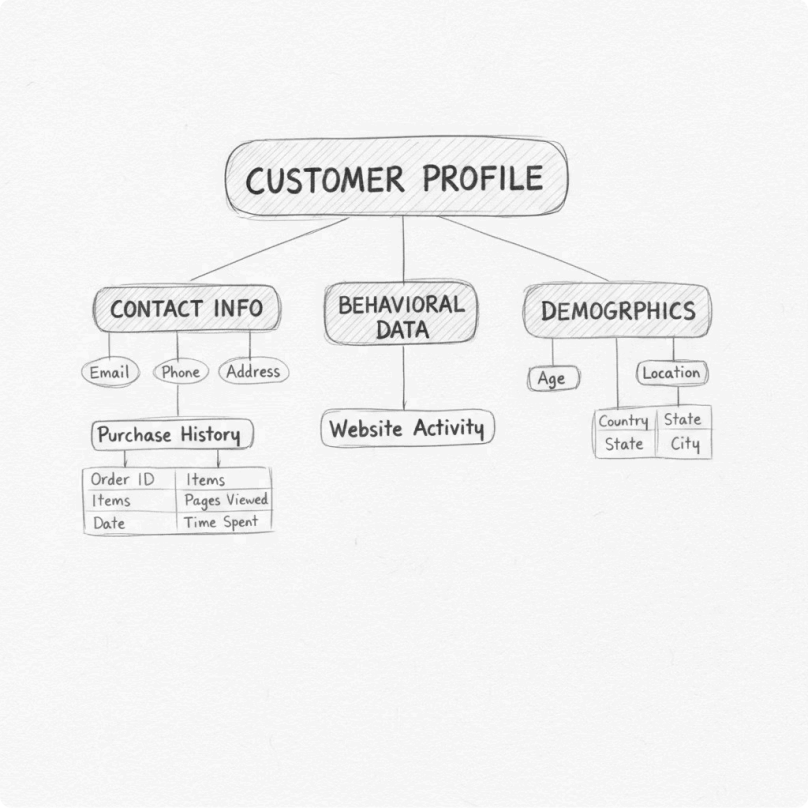
customers.csv: struktura danych

Główne pola biznesowe

- **Identyfikacja:** customer_id, first_name, last_name
- **Kontakt:** email, phone
- **Lokalizacja:** city, state, country
- **Timeline:** registration_date
- **Segmentacja:** customer_segment (Basic, Premium, VIP)

Znaczenie biznesowe

Dane klientów stanowią fundament analizy profilu zakupowego, segmentacji, wskaźników LTV (lifetime value), predykcji churn oraz strategii cross-sell i up-sell.



customers.csv: problemy jakościowe

Niespójne formaty

Różne formaty numerów telefonów i potencjalne braki w adresach email utrudniają walidację i komunikację.

Typy danych

Daty rejestracji przechowywane jako tekst wymagają konwersji do typu DATE dla analiz temporalnych.

Segmentacja

Segmenty klientów mogą być niespójne, zawierać wartości NULL lub nieaktualne klasyfikacje.

Brak historii

Nadpisywanie atrybutów bez śledzenia zmian uniemożliwia analizę ewolucji profilu klienta w czasie.

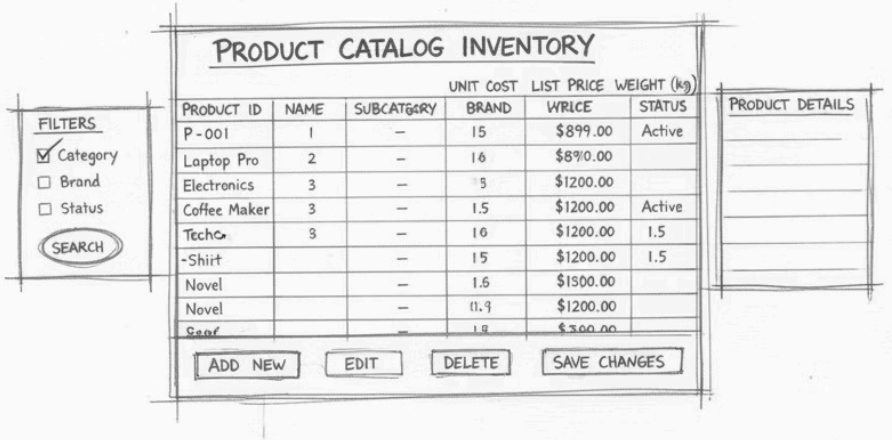
products.parquet: struktura kolumnowa

Atrybuty produktu

- product_id, product_name
- subcategory_code, brand
- unit_cost, list_price
- weight_kg
- status (active/inactive)

Zastosowania biznesowe

Katalog produktów umożliwia analizę marżowości, wydajności kategorii, strategii cenowej oraz optymalizacji portfolio produktowego.



products_updated.csv: wyzwania techniczne

Typy jako tekst

Pola liczbowe (ceny, koszty, wagi) przechowywane jako VARCHAR wymagają konwersji na DOUBLE dla obliczeń matematycznych.

Niespójne statusy

Różne warianty statusu produktu (ACTIVE, Available, active) wymagają normalizacji do jednolitej flagi binarnej.

Produkty nieznane

Brak jawnej identyfikacji produktów technicznych lub zastępczych wymaga utworzenia dedykowanego rekordu UNKNOWN.

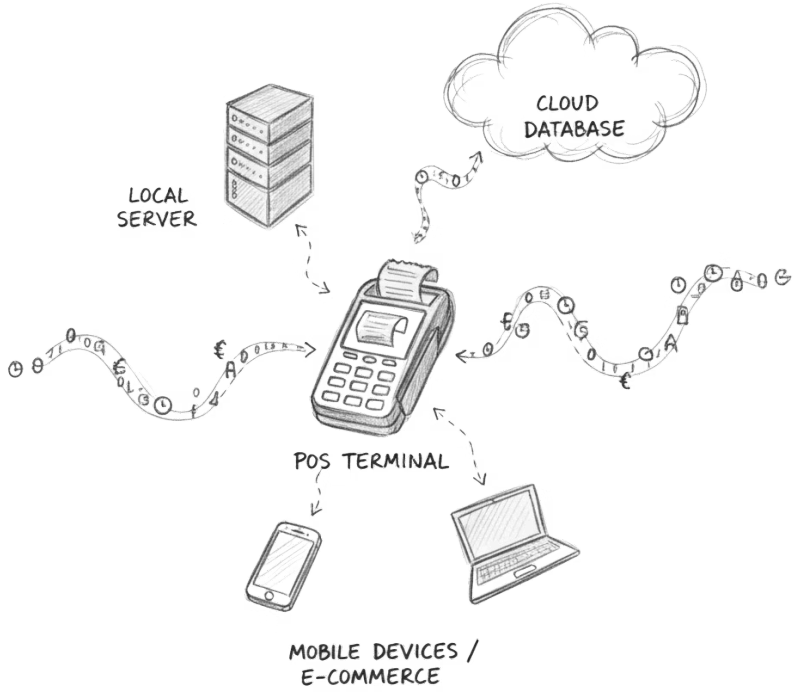
orders_stream_*.json: transakcje sprzedażowe

Struktura zamówienia

- **Klucze:** order_id, customer_id, product_id, store_id
- **Czas:** order_datetime
- **Ilości:** quantity, unit_price
- **Promocje:** discount_percent
- **Wartości:** total_amount
- **Płatność:** payment_method

Dane transakcyjne generują ponad 75 000 rekordów dziennie, z pikami przekraczającymi 150 000 podczas wyprzedaży sezonowych i w weekendy. Pochodzą one z wielu kanałów, w tym z terminali POS (np. Clover) w 50 sklepach stacjonarnych, platformy e-commerce (np. Shopify) oraz dedykowanej aplikacji mobilnej.

Dane są przesyłane strumieniowo w miksro-batchach co 30-60 sekund, w plikach JSON o rozmiarze około 5-10 MB, co wymaga dedykowanego potoku dla bieżącego streamu oraz efektywnego mechanizmu backfill dla danych historycznych. Kluczowe wyzwania obejmują utrzymanie spójności danych w czasie rzeczywistym, obsługę zmiennej przepustowości strumienia oraz integrację statusów płatności z zewnętrznych systemów aby zapewnić pełny obraz transakcji i minimalizować ryzyko niespójności.



orders_stream_*.json: problemy danych transakcyjnych

Timestampy

Wartości NULL lub nieprawidłowe formaty order_datetime uniemożliwiają analizę temporalną sprzedaży.

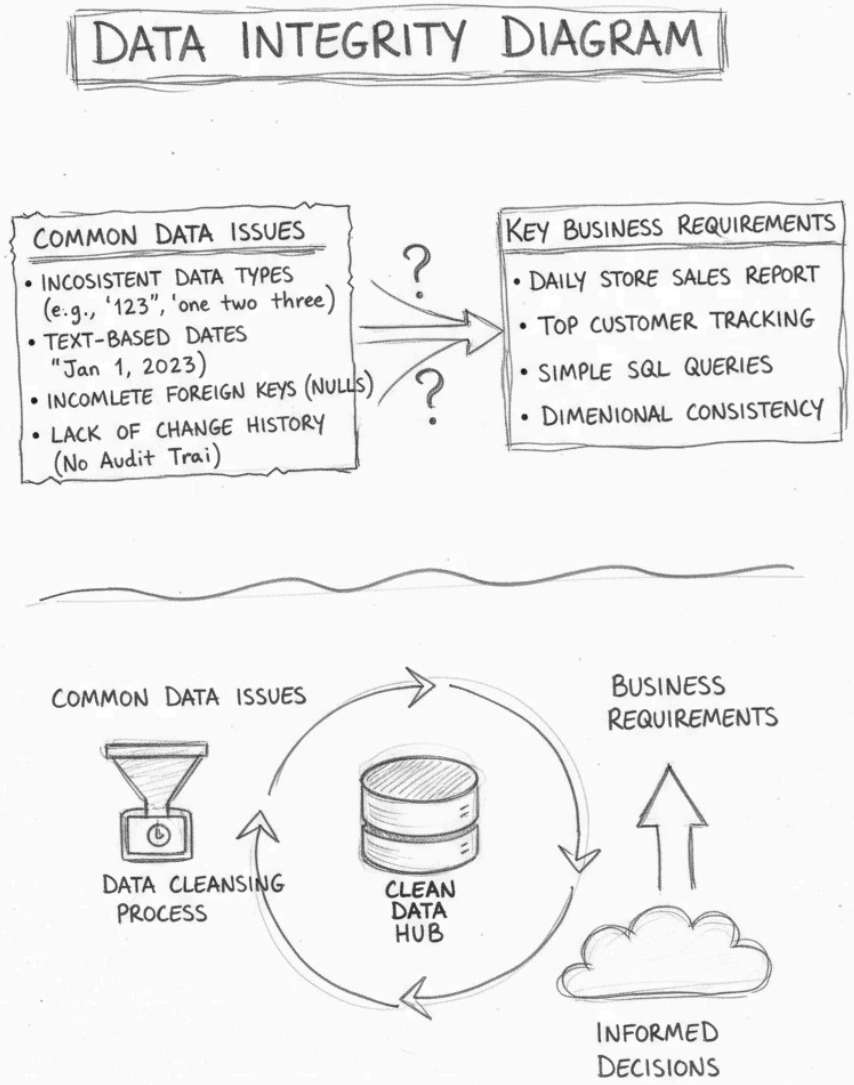
Spójność wartości

Niezgodność total_amount z obliczonym quantity × unit_price wskazuje na błędy integracji lub zaokrągleń.

Brakujące klucze

Transakcje bez customer_id lub product_id wymagają obsługi przez wzorzec unknown member.

Dodatkowo brak spójnej definicji miar (brutto, rabat, netto) komplikuje obliczenia marży i przychodów.



Dlaczego Medallion Architecture?

Problemy źródłowe

- Niespójne typy danych (string vs number)
- Daty jako tekst
- Niekompletne klucze obce
- Brak historii zmian

Wymagania biznesowe

- Sprzedaż dzienna po sklepie i płatności
- TOP N klientów w kwartale
- Proste zapytania SQL
- Zgodność wymiarów

Rozwiązanie: Medallion + Star Schema

Bronze → Silver → Gold

Bronze: Surowe dane z metadanymi

Minimalna transformacja, zachowanie lineage, pełna historia źródłowa.

Silver: Oczyszczenie i wzbogacenie

Walidacja, typy danych, miary, SCD2, reguły biznesowe.

Gold: Model semantyczny

Star schema z wymiarami i faktami gotowymi do raportowania BI.

Architektura przetwarzania: przepływ end-to-end

1

Źródła

CSV, Parquet, JSON z systemów CRM, ERP, POS

2

Bronze

bronze_customers, bronze_products, bronze_orders

3

Silver

silver_customers, silver_products, silver_orders

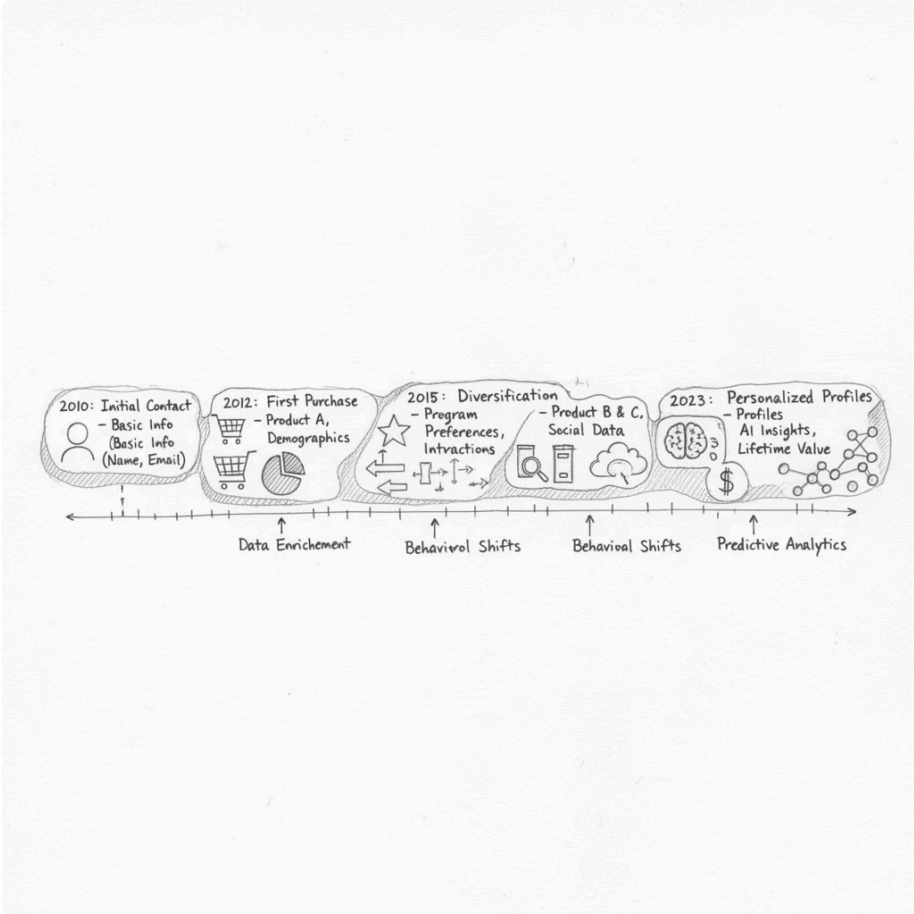
4

Gold

dim_*, fact_sales

Wszystko zaimplementowane w Delta Live Tables jako STREAMING TABLE, MATERIALIZED VIEW i FLOW z obsługą CDC, backfill oraz streaming ingest.

Silver Customers: SCD Type 2 z Delta Live Tables



Implementacja AUTO CDC

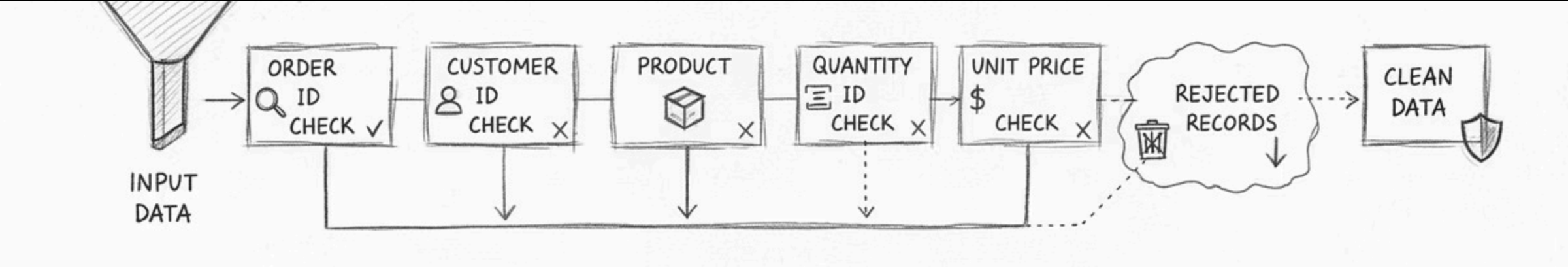
silver_customers – STREAMING TABLE

```
CREATE FLOW
  silver_customers_scd2_flow
AS AUTO CDC INTO silver_customers
FROM STREAM bronze_customers
KEYS (customer_id)
SEQUENCE BY ingestion_ts
STORED AS SCD TYPE 2
```

Kolumny techniczne

- __START_AT – początek ważności rekordu
- __END_AT – koniec ważności (NULL dla aktywnego)

Wartość biznesowa: Pełna historia zmian atrybutów klienta na osi czasu, umożliwiająca analizę ewolucji segmentacji i profilu zakupowego.



Silver Orders: constraints jakości danych

EXPECT Constraints z ON VIOLATION DROP ROW

- **valid_order_id** – order_id IS NOT NULL
- **valid_customer** – customer_id IS NOT NULL
- **valid_product** – product_id IS NOT NULL
- **valid_quantity** – quantity IS NOT NULL AND quantity <> 0
- **valid_unit_price** – unit_price IS NOT NULL AND unit_price >= 0

Rekordy nieprzechodzące walidacji są automatycznie odrzucane, co gwarantuje, że warstwa Gold i fact_sales nie będą zawierać rekordów z technicznymi błędami danych.

Silver Orders: obliczone miary biznesowe

Timestampy i klucze

- order_ts = CAST(order_datetime AS TIMESTAMP)
- order_date (implicit z order_ts)
- order_date_key = CAST(date_format(order_date,'yyyyMMdd') AS INT)

Wartości sprzedaży

- gross_amount = quantity × unit_price
- discount_amount = quantity × unit_price × discount_percent / 100
- net_amount = gross_amount - discount_amount

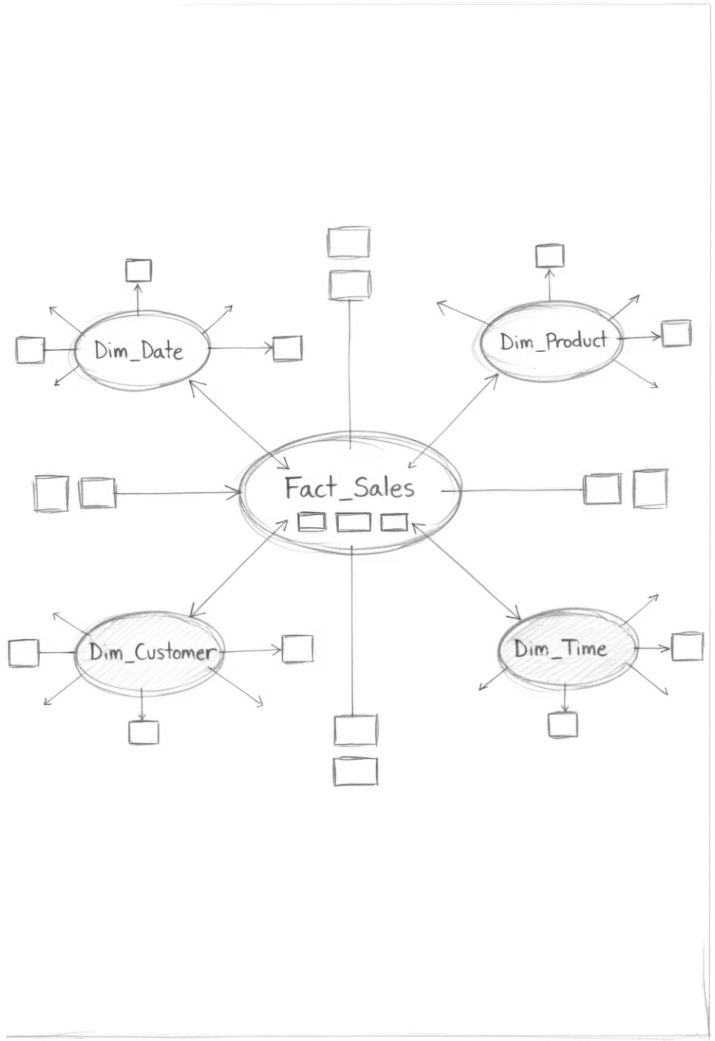
Normalizacja płatności

payment_method_code = COALESCE(payment_method, 'Unknown')

Wszystkie miary są policzone raz w Silver i propagowane do Gold, zapewniając spójność obliczeń w całym modelu.

GOLD

Model semantyczny Star Schema



Koncepcja modelu gwiazdy dla sprzedaży

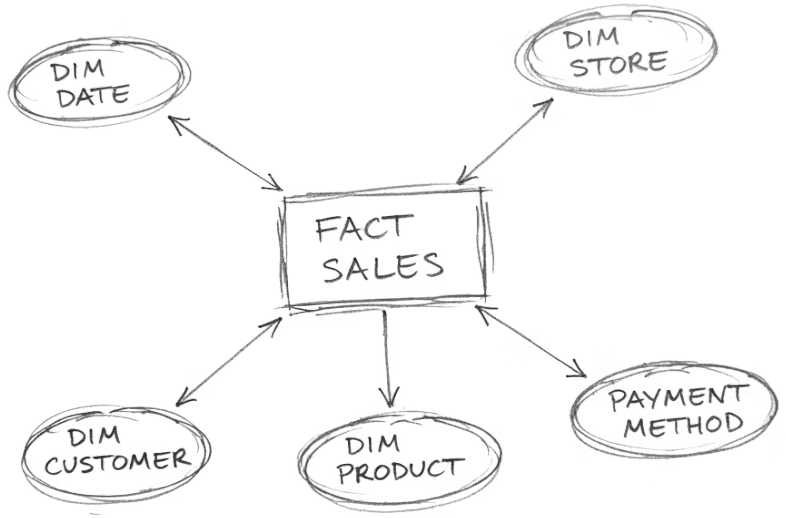
Centrum: Fact Sales

Tabela faktów zawierająca miary sprzedaży per transakcja z kluczami obcymi do wymiarów.

Wymiary analityczne

- **dim_date** – kalendarz sprzedażowy
- **dim_customer** – profile klientów
- **dim_product** – katalog produktów
- **dim_payment_method** – metody płatności
- **dim_store** – sklepy/kanały

Połączenia realizowane poprzez natural keys z warstwy Silver, bez wprowadzania surrogate keys na poziomie SQL.



Lakeflow: mapowanie artefaktów na warstwy

Bronze

- bronze_customers
  STREAMING TABLE
- bronze_products
  MATERIALIZED VIEW
- bronze_orders
  STREAMING TABLE + 2 FLOW

Silver

- silver_customers
  STREAMING TABLE + AUTO CDC SCD2
- silver_products
  MATERIALIZED VIEW
- silver_orders
  STREAMING TABLE + EXPECT

Gold

- dim_* (5 wymiarów)
  MATERIALIZED VIEW
- fact_sales
  STREAMING TABLE

Model semantyczny jest zintegrowany w Databricks Lakehouse, nie tylko w zewnętrznym narzędziu BI – zapewnia to spójność, wydajność i governance na poziomie platformy danych.