Real-Time Retail Demand Forecasting - Comprehensive Blueprint

Objective

The goal of this project is to predict product demand in real-time using sales transactions, seasonality, and promotions. This helps retailers manage inventory better, reduce stockouts, and optimize pricing strategies.

Workflow Overview

The system works like a pipeline:

1. POS systems send sales data.

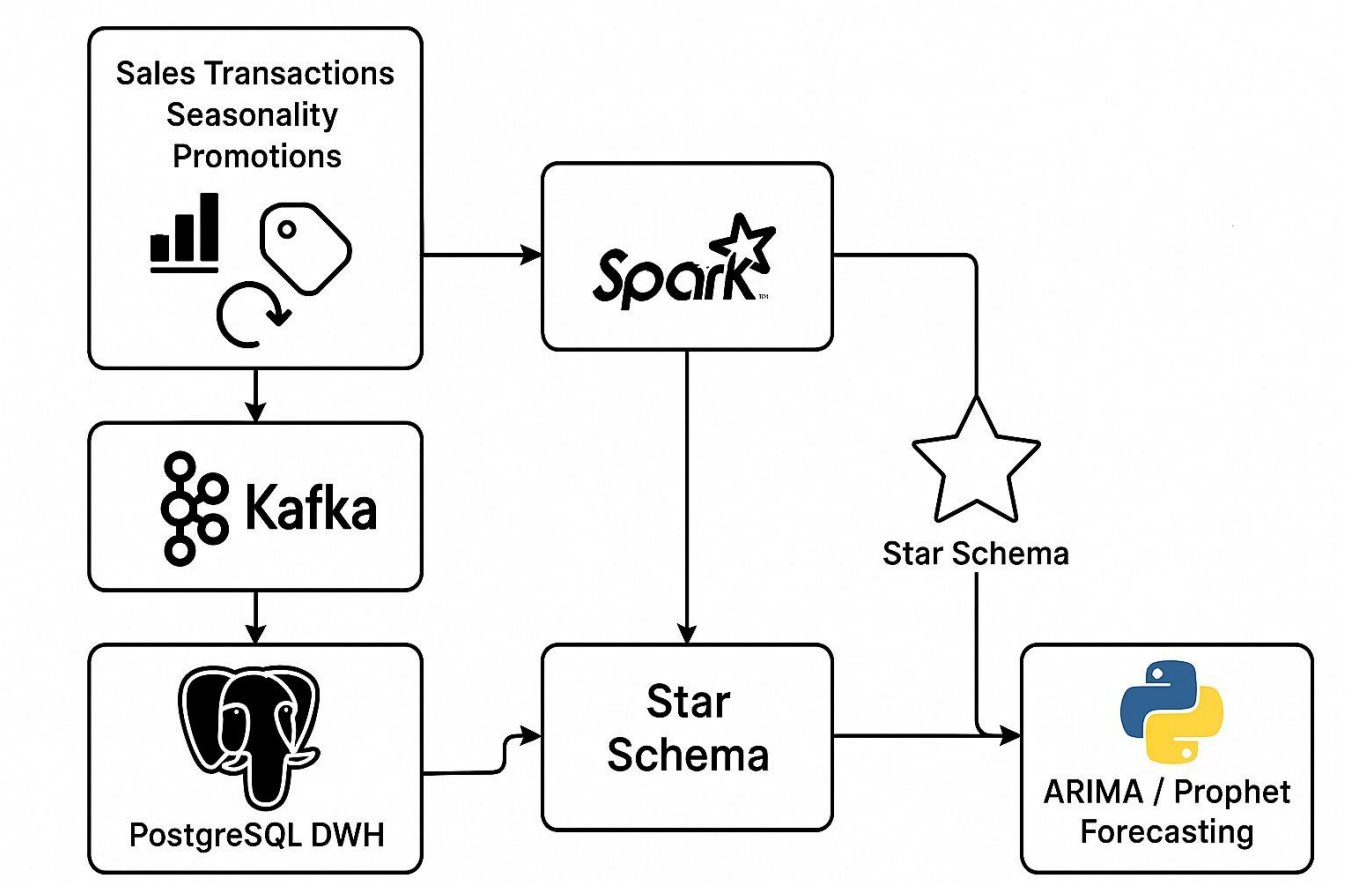
2. Kafka streams the data in real-time.

3. Spark Streaming processes and cleans the data.

4. PostgreSQL stores the structured data using a star schema.

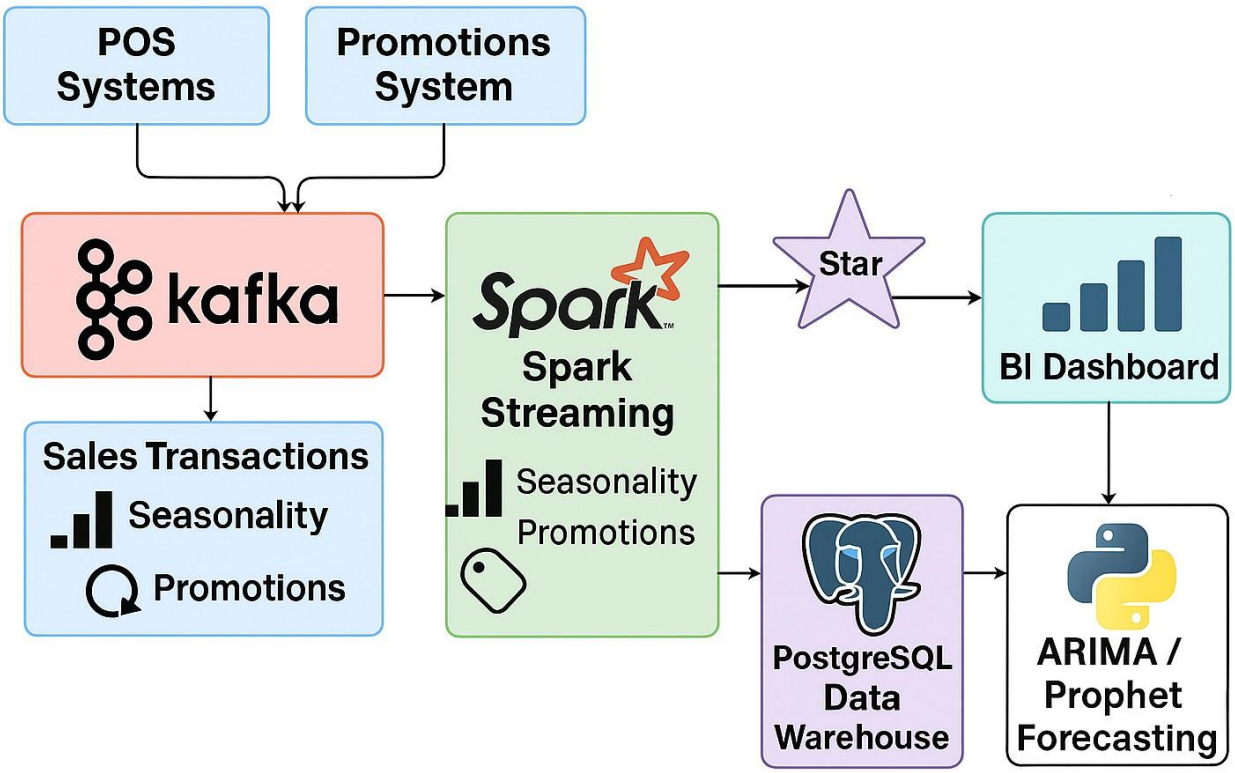
5. Python ML models forecast future demand.

6. Dashboards visualize actual and predicted demand.



System Architecture

This diagram shows the actual working of the system, including data ingestion, processing, storage, forecasting, and visualization.



Dataset Used

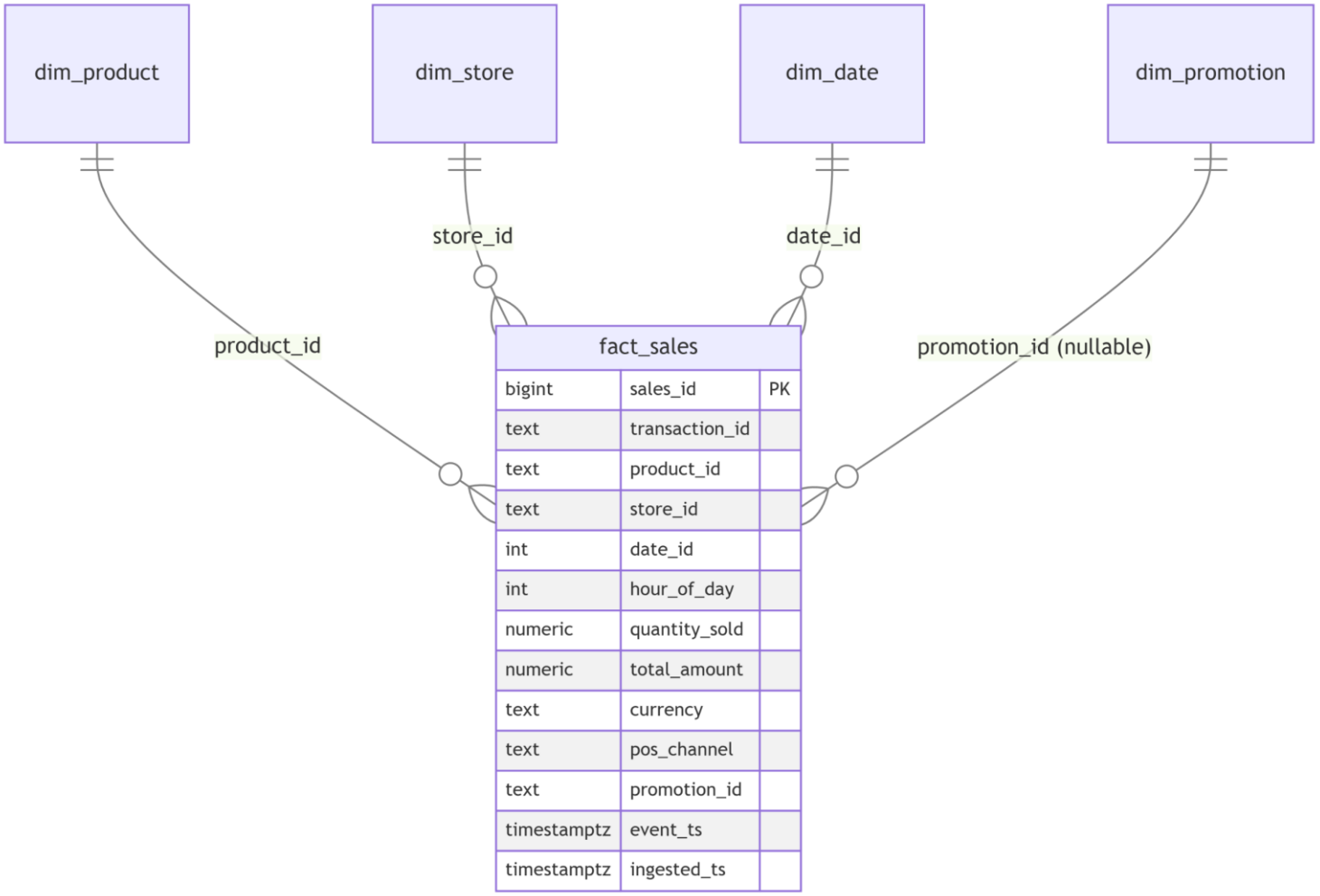
We use the Retail Store Inventory Forecasting Dataset from Kaggle:

Link: https://www.kaggle.com/datasets/anirudhchauhan/retail-store-inventory-forecasting-dataset

This dataset contains over 73,000 rows of daily sales data across multiple stores and products. It includes features like store ID, product ID, category, region, inventory level, units sold, price, weather, and promotion flags. It is ideal for time-series forecasting and inventory optimization.

Star Schema Design

The star schema consists of a central fact table 'fact_sales' and four dimension tables: 'dim_product', 'dim_store', 'dim_date', and 'dim_promotion'. Each dimension provides descriptive attributes for analysis. Below is a UML-style diagram illustrating the schema.



Sample Dataset Records

Below is a sample of 10 records from the dataset. Each row represents daily sales data for a product in a store, including inventory levels, units sold, price, weather conditions, and promotion status.

Date	Store ID	Product ID	Category	Region	Inventory	Units Sold	Price	Weather	Promotion
2025-08-01	S101	P1001	Electronic	North	50	30	299.99	Sunny	True
2025-08-02	S101	P1001	Electronic	North	45	35	299.99	Rainy	False
2025-08-03	S101	P1002	Clothing	North	60	40	99.99	Cloudy	True
2025-08-04	S102	P1003	Groceries	South	80	70	19.99	Sunny	False
2025-08-05	S102	P1003	Groceries	South	75	65	19.99	Rainy	True
2025-08-06	S103	P1004	Electronic	East	40	25	499.99	Sunny	False
2025-08-07	S103	P1004	Electronic	East	35	30	499.99	Cloudy	True
2025-08-08	S104	P1005	Clothing	West	55	45	149.99	Rainy	False
2025-08-09	S104	P1005	Clothing	West	50	50	149.99	Sunny	True

Column Explanation

- Date: The day of the sales record
- Store ID: Unique identifier for each store
- Product ID: Unique identifier for each product
- Category: Type of product (e.g., Electronics, Clothing)
- Region: Geographic location of the store
- Inventory Level: Stock available at the start of the day
- Units Sold: Number of units sold that day
- Price: Selling price per unit
- Weather: Weather condition on that day
- Promotion: Whether a promotion was active (True/False)

Sample SQL Schema

-- Dimension Tables

CREATE TABLE dim_product (product_id TEXT PRIMARY KEY, product_name TEXT, category TEXT, brand TEXT);

Machine Learning Approach

We use time-series models like ARIMA and Prophet to forecast future demand. These models analyze historical sales data and consider seasonality, holidays, and promotions to make predictions.

Tools Used

- Kafka: Real-time data streaming
- Spark Streaming: Data processing
- PostgreSQL: Data storage
- Python: Forecasting models
- Power BI/Tableau: Dashboards