

GROUP PROJECT

Team Name: Cluster Crafters (Mobile Phone Supply Chain Management)

PART 1: Design and Implementation of a Distributed Database System

This report presents the successful completion of Part 1 of our project, Mobile Phone Supply Chain Management. The system's design aims to streamline the supply chain process by integrating data across suppliers, manufacturers, warehouses, and order management.

System Architecture and Design

Entity-Relationship Diagram

The entity-relationship (ER) diagram has been created to represent the database schema in a structured format. The ER diagram, as presented in Figure 1, includes key entities such as Supplier, Manufacturer, MoobilePhone, Inventory, Warehouse, Order, and OrderDetails. The relationships between these entities are explicitly defined, with primary (PK) and foreign keys (FK) ensuring referential integrity.

Distributed Database Layout

The distributed database architecture is designed to ensure data availability, fault tolerance, and scalability. It employs a decentralized approach wherein data is replicated across multiple nodes, which are strategically located to optimize access speed and reliability.

Nodes Selection and Location

The nodes have been selected based on geographic distribution, demand centers for mobile phones, and network latency considerations to enhance the system's performance and reliability.

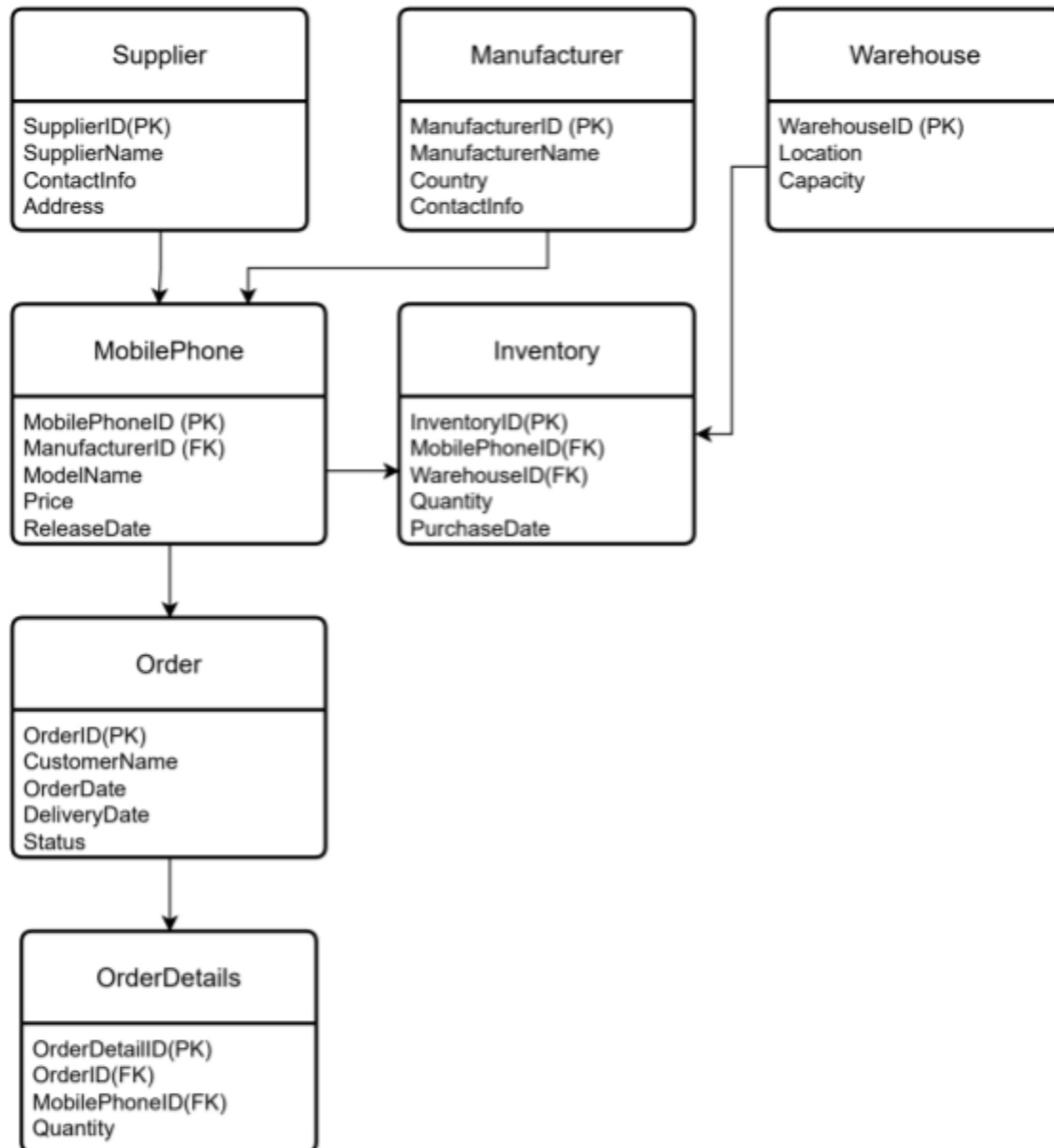


Figure 1: ER Diagram

Implementation Using CockroachDB

CockroachDB, a cloud-native SQL database, has been utilized for its strong consistency, distributed nature, and resilience. It provides an ideal platform for implementing a distributed database schema, ensuring that data remains consistent across multiple nodes.

Opting for CockroachDB Cloud offers scalability and high availability without the need for physical infrastructure. It ensures resilience through data replication and is cost-effective by eliminating hardware investments. The managed service handles maintenance, security, and updates, simplifying operations. Additionally, it enables geo-distributed data, reducing latency, and includes performance monitoring tools, while supporting distributed ACID transactions crucial for consistency across a global system.

Database Schema Deployment

The database schema, as defined in the ER diagram, has been successfully deployed across the selected CockroachDB nodes. The schema includes tables, constraints, indexes, and relationships that are vital for the management of the supply chain data.

```
create_tables_sql = [
    """
    CREATE TABLE Supplier (
        SupplierID SERIAL PRIMARY KEY,
        SupplierName VARCHAR(255),
        ContactInfo TEXT,
        Address TEXT
    )
    """,
    """
    CREATE TABLE Manufacturer (
        ManufacturerID SERIAL PRIMARY KEY,
        ManufacturerName VARCHAR(255),
        Country VARCHAR(255),
        ContactInfo TEXT
    )
    """,
    """
    CREATE TABLE MobilePhone (
        MobilePhoneID SERIAL PRIMARY KEY,
        ManufacturerID INT,
        ModelName VARCHAR(255),
        Price DECIMAL(10, 2),
        ReleaseDate DATE,
        FOREIGN KEY (ManufacturerID) REFERENCES Manufacturer(ManufacturerID)
    )
    """,
    """
    CREATE TABLE Warehouse (
        WarehouseID SERIAL PRIMARY KEY,
        Location VARCHAR(255),
        Capacity INT
    )
    """,
    """
    CREATE TABLE Inventory (
        InventoryID SERIAL,
        MobilePhoneID INT,
        WarehouseID INT,
        Quantity INT,
        PurchaseDate DATE,
        FOREIGN KEY (MobilePhoneID) REFERENCES MobilePhone(MobilePhoneID),
        FOREIGN KEY (WarehouseID) REFERENCES Warehouse(WarehouseID),
        PRIMARY KEY (InventoryID)
    )
    """,
    """
    ALTER TABLE Inventory PARTITION BY RANGE (InventoryID) (
        PARTITION Inventory_P1 VALUES FROM (MINVALUE) TO (100),
        PARTITION Inventory_P2 VALUES FROM (101) TO (200)
    );
    """,
    """
    CREATE TABLE Orders (
        OrderID SERIAL PRIMARY KEY,
        CustomerName VARCHAR(255),
        OrderDate DATE,
        DeliveryDate DATE,
        Status VARCHAR(50)
    )
    """,
    """
    CREATE TABLE OrderDetails (
        OrderDetailID SERIAL PRIMARY KEY,
        OrderID INT,
        MobilePhoneID INT,
        Quantity INT,
        FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),
        FOREIGN KEY (MobilePhoneID) REFERENCES MobilePhone(MobilePhoneID)
    )
    """
]
```

Dataset

We utilized Mockaroo as a tool to generate and customize our dataset.

Data Upload and Verification

Screenshots of sample data uploaded into the CockroachDB cloud will be provided as evidence of the successful implementation. These screenshots will showcase the populated tables with actual data reflecting the operations of the supply chain management.

```
(base) balajirajagururajakumar@Balajis-MacBook-Air Distrib
[dropped all tables
Tables created successfully!
Data inserted into the Supplier table successfully!
[Data inserted into the Manufacturer table successfully!
Data inserted into the MobilePhone table successfully!
Data inserted into the Warehouse table successfully!
Data inserted into the Inventory table successfully!
Data inserted into the Orders table successfully!
Data inserted into the OrderDetails table successfully!
```

The following figure shows that data has been inserted successfully. We have queried the database to provide 2 rows of data from each table.

```
(base) balajirajagururajakumar@Balajis-MacBook-Air DistributedPhoneSupplyChain % python cockroach.py
Data from Supplier: [(1, 'Apple', '652-522-7230', '02833 Prairie Rose Drive'), (2, 'Samsung', '403-664-2259', '3 Crowley Junction')]
Data from Manufacturer: [(1, 'Foxconn', 'France', '616-722-9433'), (2, 'Pegatron', 'Nigeria', '504-352-5255')]
Data from MobilePhone: [(1, 10, 'A', Decimal('3644.14'), datetime.date(2021, 1, 1)), (2, 8, 'B', Decimal('4693.24'), datetime.date(2013, 1, 1))]
Data from Warehouse: [(1, 'South Africa', 169701), (2, 'Indonesia', 300090)]
Data from Inventory: [(1, 42, 80, 14788, datetime.date(2020, 4, 22)), (2, 11, 21, 87024, datetime.date(2021, 12, 16))]
Data from Orders: [(1, 'Ronnie', datetime.date(2023, 1, 9), datetime.date(2022, 5, 5), 'Shipped'), (2, 'Lamar', datetime.date(2023, 2, 8), datetime.date(2023, 6, 12), 'Returned')]
Data from OrderDetails: [(1, 277, 68, 1), (2, 896, 75, 4)]
(base) balajirajagururajakumar@Balajis-MacBook-Air DistributedPhoneSupplyChain % █
```

Conclusion

The design and implementation of the distributed database system for Mobile Phone Supply Chain Management using CockroachDB have been completed. The system is now ready for the next parts of the project, which will include fragmentation and performance optimization.