James Lyou, Dai Tan, Dae Woong Choi

December 12, 2018

Info 330

Group Project 2

Rudimentary ETL:

```
CREATE TABLE Shipment_DIMENSION (
    Package_ID INT PRIMARY KEY,
    Shipping_Address VARCHAR (100),
    [Weight] INT,
    Volume INT,
    OrderedDate DATE,
    Arrival_Date DATE,
    Branch_Location VARCHAR (100)
);

CREATE TABLE Customer_DIMENSION (
    Customer_ID INT PRIMARY KEY,
    First_Name VARCHAR (100),
    Last_Name VARCHAR (100),
    [Address] VARCHAR (100)
);

CREATE TABLE Employee_DIMENSION (
    Employee_ID INT PRIMARY KEY,
    First_Name VARCHAR (100),
    Last_Name VARCHAR (100),
    Branch_Name VARCHAR (100),
    Salary INT
);

CREATE TABLE Transaction_FACT (
    Transaction_ID INT,
    Customer_ID INT,
    Employee_ID INT,
    Package_ID INT,
    Cost INT
);

ALTER TABLE Transaction_FACT
ADD FOREIGN KEY (Customer_ID) REFERENCES Customer_DIMENSION(Customer_ID);

ALTER TABLE Transaction_FACT
ADD FOREIGN KEY (Employee_ID) REFERENCES Employee_DIMENSION(Employee_ID);
```

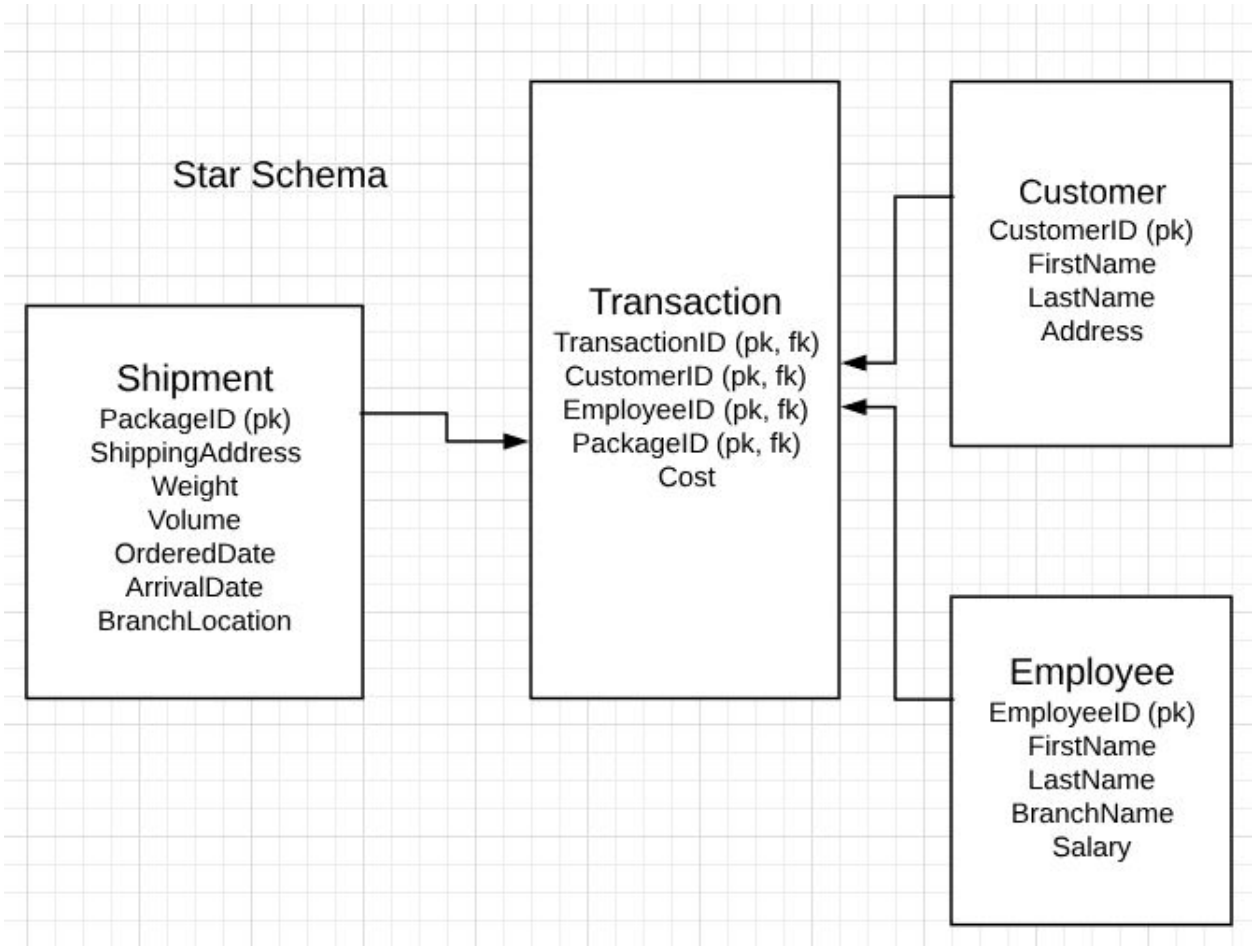
```sql
ALTER TABLE Transaction_FACT
ADD FOREIGN KEY (Package_ID) REFERENCES Shipment_DIMENSION(Package_ID);


--ShipmentDimension
INSERT INTO Shipment_DIMENSION (Package_ID, Shipping_Address, [Weight], Volume,
OrderedDate, Arrival_Date, Branch_Location)
SELECT p.PackageID, Shipping_Address = CAST(p.ShippingAddress AS VARCHAR(100)),
p.[Weight] AS StandardWeight, p.Volume AS StandardVolume, t.[Date] AS OrderedDate,
         p.ArrivalDate AS Arrival_Date, Branch_Location = CAST(p.LastBranchLocation AS
VARCHAR(100))
FROM group9.dbo.Package AS p
JOIN group9.dbo.[Transaction] AS t
ON p.PackageID = t.PackageID;


--CustomerDimension
INSERT INTO Customer_DIMENSION (Customer_ID, First_Name, Last_Name, [Address])
SELECT c.CustomerID, First_Name = CAST(c.FirstName AS VARCHAR(100)), Last_Name =
CAST(c.LastName AS VARCHAR(100)), c.[Address]
FROM group9.dbo.Customer as c;


--EmployeeDimension
INSERT INTO Employee_DIMENSION (Employee_ID, First_Name, Last_Name, Branch_Name, Salary)
SELECT e.EmployeeID, First_Name = CAST(e.FirstName AS VARCHAR(100)), Last_Name =
CAST(e.LastName AS VARCHAR(100)), e.BranchName AS Branch_Name, e.Salary
FROM group9.dbo.Employee as e;


--TransactionFact
INSERT INTO Transaction_FACT (Transaction_ID, Employee_ID, Customer_ID, Package_ID, Cost)
SELECT t.TransactionID, e.EmployeeID, c.CustomerID, p.PackageID, t.Cost
FROM group9.dbo.[Transaction] as t
JOIN group9.dbo.Employee as e ON t.EmployeeID = e.EmployeeID
JOIN group9.dbo.Customer as c ON t.CustomerID = c.CustomerID
JOIN group9.dbo.Package as p ON t.PackageID = p.PackageID;
```



Justification of star schema:

The reason why a star schema is used here is because the specific columns of data taken from the original database is not all of them. Rather the columns taken are chosen specifically to describe a certain action being done – in this case a transaction. The use of a star schema here also runs the risk of having redundant data as it denormalizes the data. The extent of it has the dimension tables being referred directly to the fact table. The advantage for the schema however is that querying this data warehouse is much faster. When it comes to a more real world application, getting queries about transactions when customer order errors occur quickly is important.

Business questions:

--Question 1: What is the monthly average transaction cost?

```
SELECT AVG(Cost) AS Monthly_average_transaction_cost
FROM Transaction_FACT;
```

--Question 2: What is the average payment of employee per year?

```
SELECT AVG(Salary) AS Annual_average_payment
FROM Employee_DIMENSION;
```

--Question 3: What is the monthly average customer?

```
SELECT COUNT(c.Customer_ID) AS Monthly_average_customer
FROM Customer_DIMENSION c, Transaction_FACT t
WHERE c.Customer_ID = t.Customer_ID AND t.Package_ID IN (
        SELECT s.Package_ID FROM Shipment_DIMENSION s WHERE s.Package_ID
        = t.Package_ID);
```

--Question 4: What is the average volume of packages in Houston that was arrived before 2003?

```
SELECT AVG(Volume) AS Average_volume_package_in_Houston
FROM Shipment_DIMENSION
WHERE Branch_Location = 'Houston' AND
        DATEDIFF(DAY, '2003-01-01', Arrival_Date) > 0;
```

--Question 5: What is the monthly transaction total?

```
SELECT SUM(Cost) AS Monthly_transaction_total
FROM Transaction_FACT;
```

--Question 6: What is the total cost of transaction on December?

```
SELECT sum(Cost)
FROM Transaction_FACT t, Shipment_DIMENSION s
WHERE t.Package_ID = s.Package_ID and s.OrderedDate LIKE '%12%';
```

--Question 7: Find the employee who is also a customer. (this one does not involve aggregate over time)

```
SELECT c.First_Name, c.Last_Name
FROM Customer_DIMENSION c, Employee_DIMENSION e
WHERE c.First_Name = e.First_Name AND
        c.Last_Name = e.Last_Name;
```