Advanced Data Engineering: Assignment 2

NGUYEN T. Hoang - SID: 15M54097

Fall 2015, W831 Tue. Period 5-6
Due date: 2015/10/27

Problem

Consider a data warehouse of a large delivery company (such as FedEx, UPS, DHL). The information generated for each delivery order is stated in the assignment document. For such situation, I consider these following aspect of the desired system: Business model, possible users and actions, and system requirements.

Business model The operation of the delivery company is assumed and stated simply as follow:

- Delivery company receives delivery orders and delivery objects from customers.
- Delivery company's branches process the delivery orders and execute them.
- During delivery, customers can track the delivery status of the package using an web application or mobile application.
- Various information about the delivery order will be recorded or queried through out the delivery process.
- For each order, there can be only one type of objects, but there can be multiple object in one order. Customers have to place multiple order in case they want to send multiple type of objects.
- Delivery company's employees can access the Data Warehouse to do their work.

Possible users and actions The term "users" is referred to the company's employees, who has access to the Data Warehouse system only. Their identities and possible actions are:

- **Statisticians** are professionals who write complex query *directly* to the Data Warehouse. They perform data analyzing to produce meaningful information about the data to help improving the company's operation. These users are professionals themself, so in the scope of this assignment there will be no consideration for these users. Statisticians have full direct access to the Data Warehouse.
- **Knowledge users** perform queries to quantify each subject area and create reports for other users. They have direct access to some certain parts of the Data Warehouse.
- **Excutives** are managers who perform queries for business administration and decision making. However, they do not need to perform complex queries. Aggregation is the most common query among excutives, and the queries is performed on the reports from knowledge users, not from the Data Warehouse. They use these information to make strategic business decisions. This type of user does not have direct access to the Data Warehouse, but they might have access to the graphic data visulization tool provied by the system.

- **Employees** are deliverypeople or office worker in each branch. Their queries are also performed on the report from knowledge users. They have limited access to the database, and they can only perform basic queries such as queries about a particular package's deliver address.
- **Customers** can also be a part of the user group. Customers can perform some queries about their orders status (e.g. expected delivery date, location, etc.) through a smartphone application or a web application. This type of user has the lowest right of access to the system.

System Requirements Based on the information given in the assignment document and the consideration above, I propose some of the requirements for the Data Warehouse:

- The system should be responsive for every users' query.
- Process power should be able to handle more than 1,200 order per second. (currently, it's 1000 order per second)
- The system should be scalable and robust.
- Address each business problem correctly and efficiently with each fact table of the star schema.

Question 1

Consider other information related to the deliver orders (e.g. zip, telephone number . . .)

- *They should be different order by order.*
- *Assume also appropriate data size for the information.*

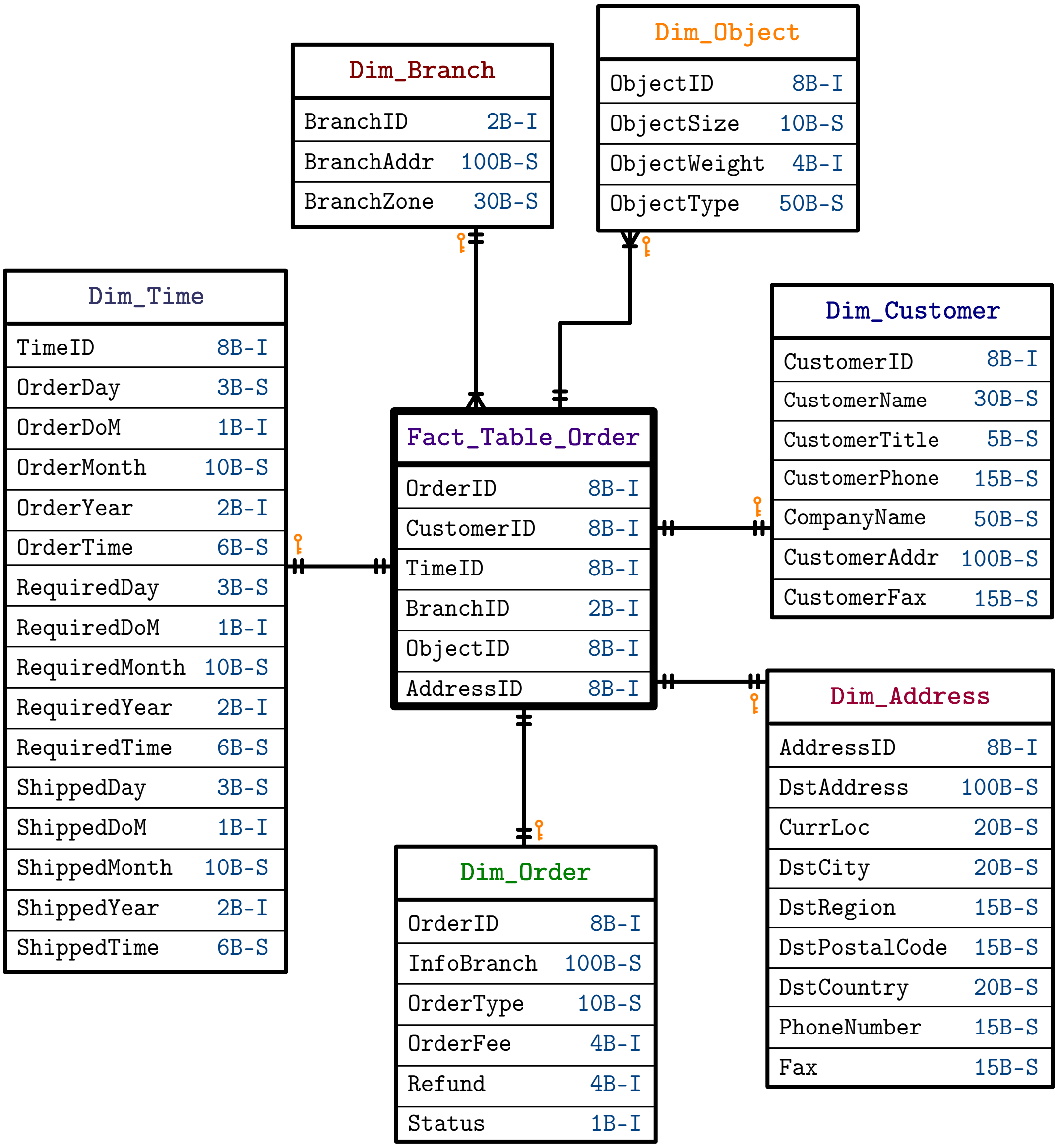
Answer: Extra information should be added to the delivery orders could be specific time requirements, or regional information for the order.

- OrderDay 3 BYTE STRING
- OrderDayOfMonth 1 BYTE INT
- OrderMonth 10 BYTE STRING
- OrderYear 2 BYTE INT
- RequiredDay 3 BYTE STRING
- RequiredDayOfMonth 1 BYTE INT
- RequiredMonth 10 BYTE STRING
- RequiredYear 2 BYTE INT
- ShippedDay 3 BYTE STRING
- ShippedDayOfMonth 1 BYTE INT
- ShippedMonth 10 BYTE STRING
- ShippedYear 2 BYTE INT
- PhoneNumber 15 BYTE STRING
- Fax 15 BYTE STRING
- CustomerPhone 15 BYTE STRING
- CustomerTitle 5 BYTE STRING
- CustomerAddr 100 BYTE STRING
- CompanyName 50 BYTE STRING
- DstCity 20 BYTE STRING
- DstRegion 15 BYTE STRING
- DstPostalCode 15 BYTE STRING
- DstCountry 20 BYTE STRING
- CurrLoc 20 BYTE STRING
- OrderStatus 1 BYTE INT

Question 2

Depict your star schema to store the deliver order information including your own assumptions for **Question 1**.

Answer: I propose a star schema as in figgure 1 below. In my design I use a factless fact table contains only keys to the dimention tables. The reason for me to choose this design is that a factless fact table can give better description about the delivery busines and the flexibility of the design. For example, using a factless fact table, the system can answer queries about both number of packages delivered in a period of time and the time it takes for a package to be delivered or even aggregation queries about customers. Also, I made design decision for data size of each attributes in the table. I considered the number of operations and performance (such as sorting) that the system should handle frequently.



Dim_Branch	
BranchID	2B-I
BranchAddr	100B-S
BranchZone	30B-S

Dim_Object	
ObjectID	8B-I
ObjectSize	10B-S
ObjectWeight	4B-I
ObjectType	50B-S

Dim_Time	
TimeID	8B-I
OrderDay	3B-S
OrderDoM	1B-I
OrderMonth	10B-S
OrderYear	2B-I
OrderTime	6B-S
RequiredDay	3B-S
RequiredDoM	1B-I
RequiredMonth	10B-S
RequiredYear	2B-I
RequiredTime	6B-S
ShippedDay	3B-S
ShippedDoM	1B-I
ShippedMonth	10B-S
ShippedYear	2B-I
ShippedTime	6B-S

Fact_Table_Order	
OrderID	8B-I
CustomerID	8B-I
TimeID	8B-I
BranchID	2B-I
ObjectID	8B-I
AddressID	8B-I

Dim_Customer	
CustomerID	8B-I
CustomerName	30B-S
CustomerTitle	5B-S
CustomerPhone	15B-S
CompanyName	50B-S
CustomerAddr	100B-S
CustomerFax	15B-S

Dim_Address	
AddressID	8B-I
DstAddress	100B-S
CurrLoc	20B-S
DstCity	20B-S
DstRegion	15B-S
DstPostalCode	15B-S
DstCountry	20B-S
PhoneNumber	15B-S
Fax	15B-S

Dim_Order	
OrderID	8B-I
InfoBranch	100B-S
OrderType	10B-S
OrderFee	4B-I
Refund	4B-I
Status	1B-I

Figure 1: Star Schema for Deliver Orders

Question 3

Calculate the amount of data stored in the data warehouse using the star schema of **Question 2** *for a day.*

Answer: Base on the proposed star schema in Figure 1, the data size calculation for one day is done as follow:

- The size of Dim_Branch is ignored since Dim_Branch contains total of 10,000 records, independent with the number of orders.
- Data size for each order is the sum of all size of attributes in the fact table and dimension table.
- Total data size for one day is the size of each order times with number of order per day.

Number of order per day:

$$N_{orders} = R_{orders\_per\_sec} \times Seconds\ In\ A\ Day$$

$$N_{orders} = 1000 \times 86400 = 86.4 \times 10^6\ orders$$

Data size for each order:

$$Size_{deliver\_order} = Size_{Fact} + Size_{Time} + Size_{Order} + Size_{Address} + Size_{Customer} + Size_{Object}$$

$$Size_{deliver\_order} = 42 + 74 + 127 + 228 + 223 + 72 = 766\ Bytes$$

Data stored in the data warehouse using the proposed star schema for a day:

$$DataSize_{1\_day} = N_{orders} \times Size_{deliver\ order}$$

$$DataSize_{1\_day} = 86.4 \times 10^6 \times 766 = 66,182,400,000\ Bytes$$

$$DataSize_{1\_day} \approx\ 66\ GB$$

Question 4

Calculate the increase of data amount for a year.

Answer: Not taking leap year into account, a year has 365 days. The amount of data will be increased over a year with the proposed star schema:

$$DataSize_{1\_year} = DataSize_{1\_day} \times 365 = 66,182,400,000 \times 365$$

$$DataSize_{1\_year} = 24,156,576,000,000\ Bytes \approx 24\ TB$$

Question 5

Consider the situation without using star schema; you use a big table containing all attributes, and compare with the star scheme you defined for **Question 2***.*

Answer: Assume that the big table containing all attributes omits all redundancy ID attributes to make a big table. The size of each order with such assumption is calculated by subtracting twice the size of fact table (each ID appear twice in the star schema), and adding back the size of the OrderID (8) along with the size of BranchID (2).

$$SizeTable_{deliver\_order} = Size_{deliver_order} - 2 \times Size_{Fact} + 8 + 2 = 692\ Bytes$$

$$SizeTable_{1\_year} = SizeTable_{1\_day} \times 365 = 59,788,800 \times 365$$

$$SizeTable_{1\_year} = 21,822,912,000\ Bytes \approx 22\ TB$$

By considering the data size increase over a year and architecture of both “big table” and proposed star schema, I can conclude the advantages and disadvantages of both scheme as follow:

1. Normalized big table
   - **Advantages** of this scheme is that it saves storage space (21 TB over a year versus 24 TB over a year). This scheme is also simpler to understand.
   - **Disadvantages** of this scheme is that it may handle join operations and changes poorly due to the nature of the big table.
2. Proposed Star Schema
   - **Advantages** of this scheme is that it has simple design. Moreover, star scheme can handle faster queries, join operation, and better compatible with OLAP model than the aforementioned big table schema.
   - **Disadvantages** of this scheme is that it has higher storage because of data denormalization

Question 6

Consider an SQL sentence to derive the total weight of the objects of which type is fresh foods, shipped by branches in Kansai zone for this year from the star schema of **Question 2**.

Answer:

Listing 1: SQL sentence for total weight of ‘Fresh foods’ delivered in ‘Kansai’ this year (2015)

```
SELECT SUM(ObjectWeight)
FROM Dim_Object
WHERE
    ObjectID IN (
        SELECT ObjectID
        FROM Fact_Table_Order
        WHERE BranchID IN (
            SELECT BranchID
            FROM Dim_Branch
            WHERE BranchZone = 'Kansai')
        AND TimeID IN (
            SELECT TimeID
            FROM Dim_Time
            WHERE ShippedYear = 2015))
    AND ObjectType = 'Fresh foods';
```