**NORTH SOUTH UNIVERSITY**



Department of Electrical and Computer Engineering

Project Report

Course: CSE411

Section: 2

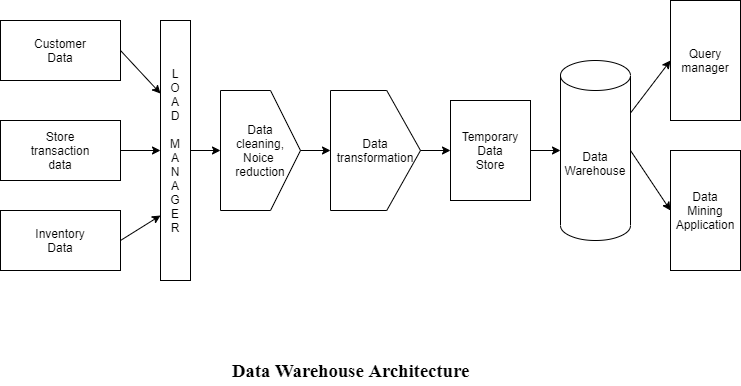
Name: Dipta Voumick

ID: 1632178042

Submission date: 26th January, 2021

**Task 1:** Data warehouse architecture

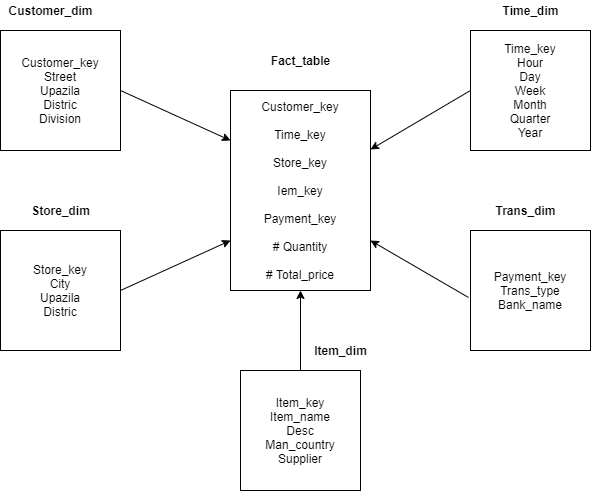
The sources in this architecture are the individual store transactions/sales data, customer data and inventory data from the head office of the company. These sets of data are loaded onto the staging area. Data cleaning and noise reduction then takes place where some missing data is replaced with statistical analysis. An example of this would be if a customer missed a portion of their location when filling a form. That information would be inferred and replaced. The data is then transformed to fit the schema and format of the data warehouse. The transformed data is then stored in the data warehouse. From there, queries can be executed using the query manager and data mining through applications for knowledge discovery.



**Task 2:** Star schema for data warehouse

The fact table for this data warehouse has the dimensional attributes payment\_key, customer\_key, item\_key, time\_key and store\_key, which all connect to dimension tables. The measure attributes are quantity and total\_price. The dimension tables for this data warehouse are store\_dim, item\_dim, customer\_dim, trans\_dim and time\_dim.

Some of the fields in the dimension tables have been dropped as they were no needed for decision making. These include customer name, NID and address.



**Task 3:** Cross Tabulation

The cross tabulations I made for this project are:

1. City-wise sales from different transaction types – transaction dimension and store dimension

SQL for total sales of transcation types: SELECT trans\_type, SUM(total\_price) FROM fact\_table a, trans\_dim b

WHERE a.payment\_key = b.payment\_key GROUP BY trans\_type;

SQL for total sales by city:

SELECT city, SUM(total\_price) FROM fact\_table a, store\_dim b WHERE a.store\_key = b.store\_key GROUP BY city;

SQL for other cells:

SELECT city, trans\_type, SUM(total\_price) FROM fact\_table a, trans\_dim b, store\_dim c

WHERE a.payment\_key = b.payment\_key AND a.store\_key = c.store\_key GROUP BY city, trans\_type;

**Resultant table:**

# cash online total



**City-wise Transaction Types Used**

90000000

80000000

70000000

60000000

50000000

40000000

30000000

20000000

10000000

0

cash

online

**District**

|  |  |  |  |
| --- | --- | --- | --- |
|  | 103263.25 | 3515629 | 3618892.25 |
| Barishal | 99295.25 | 3523494.75 | 3622790 |
| Chadpur | 91822 | 3513211.25 | 3605033.25 |
| Comilla | 100221.25 | 3540093.5 | 3640314.75 |
| Dhaka | 2225888.5 | 77609120.25 | 79835008.75 |
| Gazipur | 98072.75 | 3517646.25 | 3615719 |
| Narsingdi | 104997.5 | 3562064.5 | 3667062 |
| Sylhet | 97454.25 | 3515957.75 | 3613412 |
| total | 2921014.8 | 102297217.3 | 105218232 |

**Graph 1**: City-wise transaction types used

**Sales**

**Sales**

**Graph 2:** Total sales for transaction types

**Total Sales for Transaction Types**

120000000

100000000

80000000

60000000

40000000

total

20000000

0

cash

onlinee

**Transaction Types**



**Total Sales for Cities**

90000000

80000000

70000000

60000000

50000000

40000000

30000000

20000000

10000000

0

total

**Cities**

**Sales**

**Graph 3:** Total sales for cities

1. Quantity sold of item type by district – store dimension and item dimension SQL for total quantity of item type:

SELECT desc, SUM(quantity) FROM fact\_table a, item\_dim b WHERE a.item\_key = b.item\_key GROUP BY desc;

SQL for total quantity sold by district: SELECT district, SUM(quantity) FROM fact\_table a, store\_dim b WHERE a.store\_key = b.store\_key GROUP BY district;

SQL for other cells:

SELECT desc, district, SUM(quantity) FROM fact\_table a, item\_dim b, store\_dim c

WHERE a.item\_key = b.item\_key AND a.store\_key = c.store\_key GROUP BY desc, district;

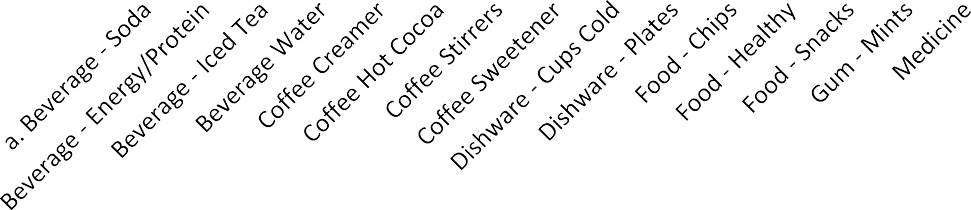
**Resultant table:**

# B.Baria Barishal Chadpur Comilla Dhaka Gazipur Narsingdi Sylhet Total

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| a. Beverage - Soda | 23384 | 23439 | 22944 | 22168 | 496781 | 23621 | 23283 | 23415 | 635620 |
| a. Beverage  Sparkling Water | 16287 | 17077 | 16921 | 15890 | 364745 | 16890 | 16333 | 17072 | 464143 |
| Beverage -  Energy/Protein | 10833 | 10550 | 10651 | 10885 | 242166 | 10762 | 11191 | 11022 | 307038 |
| Beverage - Gatorade | 7789 | 7719 | 7521 | 7418 | 170536 | 7919 | 7842 | 7817 | 216744 |
| Beverage - Iced Tea | 5672 | 5054 | 5484 | 5430 | 121681 | 5680 | 5848 | 5842 | 154849 |
| Beverage - Juice | 5861 | 5638 | 5793 | 5527 | 118024 | 5418 | 5577 | 5255 | 151838 |
| Beverage Water | 6862 | 6711 | 6750 | 7214 | 153797 | 6903 | 7249 | 7154 | 195486 |
| Coffee Cream | 4876 | 4403 | 4885 | 4589 | 102171 | 4593 | 4700 | 4441 | 130217 |
| Coffee Creamer | 1660 | 1699 | 1555 | 1488 | 35056 | 1462 | 1789 | 1575 | 44709 |
| Coffee Ground | 3729 | 4088 | 3745 | 4370 | 85449 | 3992 | 4022 | 3974 | 109395 |
| Coffee Hot Cocoa | 1558 | 1637 | 1811 | 1470 | 34578 | 1461 | 1692 | 1582 | 44207 |
| Coffee K-Cups | 5276 | 5312 | 5425 | 5561 | 120125 | 5601 | 5508 | 5180 | 152808 |
| Coffee Stirrers | 729 | 817 | 784 | 764 | 17275 | 880 | 931 | 832 | 22180 |
| Coffee Stirrers | 727 | 872 | 872 | 873 | 17386 | 819 | 743 | 811 | 22292 |
| Coffee Sweetener | 6546 | 5983 | 6097 | 6103 | 139528 | 5858 | 6259 | 6495 | 176374 |
| Dishware - Bowls | 1556 | 1478 | 1594 | 1515 | 33380 | 1393 | 1656 | 1505 | 42572 |
| Dishware - Cups Cold | 2191 | 2357 | 2326 | 2380 | 51804 | 2379 | 2437 | 2255 | 65874 |
| Dishware - Cups Hot | 4506 | 4201 | 4607 | 4875 | 103612 | 4497 | 4648 | 4647 | 130946 |
| Dishware - Plates | 4535 | 4840 | 4467 | 4642 | 103953 | 4626 | 4633 | 4735 | 131696 |
| Dishware - Utensils | 2225 | 2508 | 2283 | 2540 | 51651 | 2284 | 2619 | 2172 | 66110 |
| Food - Chips | 16595 | 16909 | 16536 | 16421 | 361888 | 16093 | 16534 | 16616 | 460976 |
| Food - Chocolate | 10752 | 11100 | 10761 | 10834 | 239472 | 11226 | 11448 | 10922 | 305593 |
| Food - Healthy | 22830 | 22671 | 22373 | 23397 | 497262 | 22534 | 22234 | 22409 | 633301 |
| Food - Nuts | 4479 | 4582 | 4490 | 4554 | 103677 | 4361 | 4622 | 4642 | 130765 |
| Food - Snacks | 4118 | 3794 | 3989 | 3872 | 85676 | 4101 | 3984 | 3986 | 109534 |
| Food - Sweets | 8507 | 8714 | 8427 | 8692 | 192213 | 8487 | 8245 | 8186 | 243285 |
| Gum - Mints | 6330 | 6108 | 6200 | 6296 | 138151 | 5977 | 6411 | 6048 | 175473 |
| Kitchen Supplies | 15179 | 14803 | 14862 | 15476 | 329028 | 14624 | 14820 | 14251 | 418792 |
| Medicine | 1507 | 1661 | 1609 | 1658 | 34872 | 1616 | 1553 | 1725 | 44476 |
| Total | 207099 | 206725 | 205762 | 206902 | 4545937 | 206057 | 208811 | 206566 | 5787293 |

**Quantity**

**Graph 4**: Quantity of each item type sold by district



**Quantity Item Type Sold by District**

600000

500000

400000

300000

200000

100000

0

B.Baria

Barishal Chadpur Comilla Dhaka Gazipur Narsingdi

Sylhet

**Item Types**

**Quantity Sold by Item Type**

700000

600000

500000

400000

300000

200000

100000

0

Total

**Item Type**

**Quantity**

**Graph 5:** Total quantity sold for each item type

a. Beverage -…

Beverage -… Beverage - Iced… Beverage Water Coffee Creamer Coffee Hot Cocoa

Coffee Stirrers

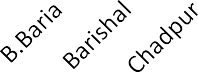
Coffee Sweetener Dishware - Cups… Dishware - Plates

Food - Chips Food - Healthy Food - Snacks Gum - Mints

Medicine

**Quantity**

**Graph 6:** Total goods sold by district



**Total Goods Sold by District**

7000000

6000000

5000000

4000000

3000000

2000000

1000000

0

Total

**District**

1. Year wise sales of districts – store dimension and time dimension SQL for total sales for every year:

SELECT year, SUM(total\_price) FROM fact\_table a, time\_dim b WHERE a.time\_key = b.time\_key GROUP BY year;

SQL for total sales of districts:

SELECT district, SUM(total\_price) FROM fact\_table a, store\_dim b WHERE a.store\_key = b.store\_key GROUP BY district;

SQL for other cells:

SELECT district, year, SUM(total\_price) FROM fact\_table a, time\_dim b, store\_dim c

WHERE a.time\_key = b.time\_key AND a.store\_key = c.store\_key GROUP BY district, year;

**Task 4:** 4-Dimensional data cube

In a 4-D data cube, the different levels correspond to the number of dimensions used to group queries. For example, at 1-D only one dimension is used for group by on aggregated queries in the fact table, like store or customer. In 2-D, two dimensions are used for group by. Like store, item or item, customer. At 4-D, four dimensions are used for group by and if there are only four dimensions used with the fact table, it will return the whole fact table. With 0-D, the operation is executed on the entire fact table with no group by.

