**Tackling old data management issues with Data Warehouse in the Rhinic Hospital Organization**

Kyaw Thiha Naing

Department of Information Technology

HELP University

Metro Mall, Fraser Business Park, Kuala Lumpur

Malaysia

sam.kyaw.567@gmail.com

*Abstract: -* The management of old data needs to be done in the most efficient way to not affect data integrity and to also prevent unnecessary delays in attaining important data that is essential for proper management of patients and also payment records in hospitals.Rhinic Hospital Organization design and implement a data warehouse to tackle the old data management problem based on a correlated business scenario to track financial statuses of hospital branches. Data analysis has been made based on the implemented data warehouse and the findings indicate how the presence of old data can affect financial analysis. The report concludes with elaborations on why the data warehouse will still need to be improved in the future.

*Keywords: -*Rhinic, Data Warehouse, Star Schema, OLAP, OLTP

**1 System Description**

Rhinic Hospital Organization aims to provide quality healthcare by effectively managing data in each hospital branch with an integrated data warehouse. The establishment of a systematic data warehouse will remove a lot of complications in data filtration resulting in no further delays in transfer of data and immediate treatments can be made. The organization has identified that the presence of old data and how it is managed, has been a problem effecting the financial flow of the hospital branches. Due to the hospitals having to collect a lot of data, a substantial amount of data needs to be stored in storage systems. The detrimental effect of old data presence on the efficiency of database systems in hospitals can also not be overlooked. Therefore, it has become such an importance to identify and eliminate data no longer having any informative values. Furthermore, the inclusion of outdated data can also lead to wrong analyses being made. The identification of irrelevant data can be made with proper data analysis after the data warehouse aiming to tackle this main problem, has been successfully implemented.

**1.1 Requirements for the project**

**1.11 Staff composition**

**Staff:** Healthcare Finance Manager

**Responsibility:** The healthcare finance manager oversees the financial flow of the hospitals and carry out analyses that are helpful in decision-making procedures)

**Staff:** Finance Assistant

**Responsibility:** The finance assistants manage invoices, process payments, and handle financial records

**Staff:** Data Manager

**Responsibility:** The data manager handles and manages data related to patients, staff, and appointments

**Staff:** Porters

**Responsibility:** Porters are in charge of transportation of patients

**Staff:** Ward Clerks

**Responsibility:** Ward clerks work standby at the reception desks in wards

**Staff:** Service assistant

**Responsibility:** Service assistants are responsible for bringing food and drinks for patients

**1.12 Gathering requirements**

The requirement gathering process for this project will involve stakeholders like healthcare finance manager, data manager and investors. There will be several meetings between the stakeholders to discuss about ideas that will help identify possible solutions to problems as well as pointing out detailed opportunities that have been simplified for common understanding. Then, the procession of the requirements gathering will focus on analysis of documents at hand which are the data that have already been gathered to identify the gaps in the business flow so that those gaps can be filled in the implementation of the project. Furthermore, focus groups involving patients can be made to gain their feedback on particular things that they do not find appropriate from their point of view so that improvements can be made in the future that will positively impact the hospitals’ business flow. Furthermore, there are no data lakes created to be designated to any staff member of the Rhinic Hospital Organization, resulting in no possibility of data swamps occurring.

**1.2 Infrastructure Plan**

**1.21 Considerations for Hardware and Desktop**

The hardware and desktop involvement will be minimal for the data warehouse due to choosing hybrid cloud computing system as an incorporated infrastructure. Therefore, there will still be some on-premises devices but most of the computing and storage services are provided by the cloud platform. The hybrid cloud infrastructure is a flexible system in which prior hardware investments would not go to a waste by fully going cloud but at the same time gain all the benefits of cloud services due to it involving both on-premises hardware as well as a cloud environment.

**1.22 Considerations for Connectivity and Network**

The network implementation is based on the internet as cloud computing is partly involved and the hospital branches are also in different places. The deployment of the application to perform operations on the data warehouse is also through the internet using HTTP as a protocol to avoid any complications. That way the application is available to individuals in the Rhinic Hospital organization. This network deployment method is also cost-effective compared to the others.

**1.23 Data security (Physical Threats)**

There will still be some data stored in the on-premises hardware, so the structural protection of the premises is still taken into consideration. The rooms where the storage devices are located will have high walls and ceilings which are fireproof. Additionally, the rooms will all have small windows with locks on them. The building will also use a key card access system for all rooms so that potential data loss from theft is minimized.

**1.24 Security (information and software threats)**

Firewall will be used for the database server so that traffic access is limited, and the data warehouse will be protected from security threats. The only traffic that will be allowed through is from the specific application used to perform data warehouse operations to access the data. The user access to the data warehouse and the database will also be limited to a certain number of people. Furthermore, some of the staff will have limited privileges while accessing the data warehouse for extra security. Their accounts will also have strong passwords and will be locked automatically after four or five attempts.

**2 Fact table generation through problem statement, related business scenario and OLTP schema**

**Problem: Old data record management**

As described previously, it is necessary to constantly get rid of old data entirely so that the valuable storage space is not wasted, especially in the case of hospital database system. The data trends are also affected by the composition of old data. By effectively deleting irrelevant data, the maintenance efforts and measures can greatly be reduced due to having less volume of data to manage.

**Business scenario:**

- Ability to create a report on amount of payment for appointments and identify outdated payment records

In order to clear out very old appointment records and know whether hospitals are doing well in certain areas financially, the finance managers from the Rhinic Hospital Organization want to generate a summary of payment records for appointments, which hospitals the appointments took place and where/when the appointments have occurred.

The Rhinic Hospital Organization has a OLTP database system shown in the figure below in which associated entities and their relative data are stored.

Diagram, schematic

Description automatically generated

Fig.1: Rhinic Hospital Organization Class diagram (zoom in)

According to the business scenario that is based on the statement of the problem, the fact table below has been created by selecting appropriate entities and attributes from the OLTP database system.

|  |  |
| --- | --- |
| **Goal** | **History of appointment payment records broken down by hospitals in each city, country, and period of time** |
| **FACT** | Fact\_Appointment\_Payment |
| **Measurements** | Amount (SUM) |
| **Dimensions** | How: Payment (payment\_id, patient\_id, amount, payment\_date)  Appointment (appointment\_id, appointment\_date, patient\_id) |
| What: Hospital (hospital\_id, address\_id) |
| Where: City (city\_id, city\_name, country\_name) |
| When: Date (day, month, year) |

Fig.2: Fact table based on business scenario

**3 Dimensional Modelling**

**3.1 Representation of the dimensional model**

According to the fact table made previously, the star schema below has been created in order to represent the overall dimensional modelling showcasing the necessary attributes and tables. The dimension’s tables’ primary keys are foreign keys in the fact table which also has an extra attribute namely payment\_amount.

Diagram

Description automatically generated

Fig.3: Rhinic Hospital Organization Star Schema

**3.2 General strategy of transformation from OLTP to OLAP**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **OLAP** | | | | **OLTP** | |
| **Dimensional Table** | **Description** | **Column** | **Data Type** | **Table Source** | **Column Source** |
| Dim\_Hospital | Hospital related information relevant to the business scenario is stored in the dim\_hospital table such as hospital\_id to uniquely identify different hospital, city\_id to know the location of each hospital | dim\_hospital\_id | SMALLINT (5) | Hospital | hospital\_id |
| city\_id | SMALLINT (5) | Address | city\_id |
| hospitalsakila\_id | SMALLINT (5) | Hospital | hospital\_id |
| Dim\_City | City related information relevant to the business scenario that is associated with patients, staff and hospitals are stored in the dim\_city table such as dim\_city\_id to unique identify different cities, city\_name to describe the city’s names and country\_name to describe the countries in which the cities are located | dim\_city\_id | SMALLINT (5) | City | city\_id |
| city\_name | VARCHAR (50) | City | city |
| country\_name | VARCHAR (50) | Country | country |
| hospitalsakila\_id | SMALLINT (5) | City | city\_id |
| last\_update | TIMESTAMP | City | last\_update |
| Dim\_Appointment | Appointment information relevant to the business scenario is stored in the dim\_appointment table such as dim\_appointment\_id to unique identify different appointments, patient\_id to describe the patients who came for the appointment | dim\_appointment\_id | SMALLINT (5) | Appointment | Appointment\_id |
| patient\_id | SMALLINT (5) | Patient | patient\_id |
| hospitalsakila\_id | SMALLINT (5) | Appointment | Appointment\_id |
| Dim\_time | This table stores time-related information to be mapped to payment\_date to display when the payments happened | id | INT | - | - |
| dt | DATE | Payment | payment\_date |
| yr | SMALLINT | - | - |
| mth | SMALLINT | - | - |
| dy | SMALLINT | - | - |
| wk | SMALLINT | - | - |
| day\_name | SMALLINT | - | - |
| Fact\_appointment\_payment | Information regarding to payment for appointments along with the associated hospitals and cities are stored in this fact table | time\_id | SMALLINT (5) | Payment | payment\_date |
| hospital\_id | SMALLINT (5) | Hospital | hospital\_id |
| appointment\_id | SMALLINT (5) | Appointment | appointment\_id |
| city\_id | SMALLINT (5) | City | city\_id |
| payment\_amount | DECIMAL (20,5) | Payment | SUM (amount) |

Fig.4: Transformation strategy from OLTP tables to OLAP tables

**4 OLAP Design and Implementation**

**4.1 OLAP Design**

The figure below showcases how the OLTP tables are mapped to the dimension and the fact tables in the OLAP.

**Diagram

Description automatically generated**

Fig.5: Mapping of tables from OLTP to OLAP (zoom in)

The table below shows what columns, data types and constraints will be included in each OLAP table as well as the justification on choosing data types and constraints.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OLAP Table** | **Column** | **Data Type** | **Reason for choosing data type** | **Constraints and reasons for choosing them** |
| Dim\_Hospital | dim\_hospital\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type | **PRIMARY KEY** (dim\_hospital\_id)  Reason: To uniquely identify each hospital |
| city\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type |
| hospitalsakila\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type |
| Dim\_City | dim\_city\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type | **PRIMARY KEY** (dim\_city\_id)  Reason: To uniquely identify each city |
| city\_name | VARCHAR (50) | The attribute will be holding character values, making varchar an appropriate data type |
| country\_name | VARCHAR (50) | The attribute will be holding character values, making varchar an appropriate data type |
| hospitalsakila\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type |
| last\_update | TIMESTAMP | The attribute holds a combination of date and time, making timestamp a suitable data type |
| Dim\_Appointment | dim\_appointment\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type | **PRIMARY KEY** (dim\_appointment\_id)  Reason: To uniquely identify each appointment |
| patient\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type |
| hospitalsakila\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type |
| Dim\_time | id | INT | The attribute needs to hold a wide range of whole numbers, making int a suitable data type | **PRIMARY KEY** (id)  Reason: To uniquely identify each row storing time values |
| dt | DATE | The attribute holds date values in YYYY-MM-DD format, making date a suitable data type |
| yr | SMALLINT | The attribute only holds small whole numbers, making smallint a suitable data type |
| mth | SMALLINT | The attribute only holds small whole numbers, making smallint a suitable data type |
| dy | SMALLINT | The attribute only holds small whole numbers, making smallint a suitable data type |
| wk | SMALLINT | The attribute only holds small whole numbers, making smallint a suitable data type |
| day\_name | SMALLINT | The attribute only holds small whole numbers, making smallint a suitable data type |
| Fact\_appointment\_payment | time\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type | **FOREIGN KEY** (hospital\_id) REFERENCES dim\_hospital(dim\_hospital\_id)  Reason: The business scenario requires the fact table to divisively provide a summary report using several attributes including hospital, resulting in applying hospital\_id as a foreign key  **FOREIGN KEY** (appointment\_id) REFERENCES dim\_appointment(dim\_appointment\_id)  Reason: The business scenario requires the fact table to divisively provide a summary report using several attributes including appointments, resulting in applying appointment\_id as a foreign key  **FOREIGN KEY** (city\_id) REFERENCES dim\_city(dim\_city\_id)  Reason: The business scenario requires the fact table to divisively provide a summary report using several attributes including cities, resulting in applying city\_id as a foreign key  **FOREIGN KEY** (time\_id) REFERENCES dim\_time(id)  Reason: The business scenario makes time extremely relevant due to identifying old records. The time\_id is used as a foreign key, referencing the dim\_time table to describe when the payments for the appointments happened. |
| hospital\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type |
| appointment\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type |
| city\_id | SMALLINT (5) | The attribute only holds small whole numbers, making smallint a suitable data type |
| payment\_amount | DECIMAL (20,5) | The attribute should be able to store decimal values, making decimal a suitable datatype |

Fig.6: Justification for Data types and constraints in the OLAP tables

**4.2 OLAP Implementation**

OLAP implementation is done by creating the tables in a separate schema called “dw\_hospitalsakila”.

The following SQL queries are used to create the schema and the OLAP tables.

CREATE DATABASE DW\_Hospitalsakila;

USE DW\_Hospitalsakila;

CREATE table dim\_hospital (

dim\_hospital\_id smallint(5) NOT NULL,

city\_id smallint(5) NOT NULL,

hospitalsakila\_id smallint(5) NOT NULL,

PRIMARY KEY (dim\_hospital\_id)

);

CREATE table dim\_time (

id int NOT NULL AUTO\_INCREMENT,

dt date DEFAULT null,

yr smallint DEFAULT null,

mth smallint DEFAULT null,

dy smallint DEFAULT null,

wk smallint default null,

day\_name char(20) default null,

PRIMARY KEY (id)

);

CREATE table dim\_city (

dim\_city\_id smallint(5) NOT NULL,

city\_name VARCHAR(50) NOT NULL,

country\_name VARCHAR(50) NOT NULL,

hospitalsakila\_id smallint(5) NOT NULL,

last\_update TIMESTAMP NOT NULL,

PRIMARY KEY (dim\_city\_id)

);

CREATE table dim\_appointment (

dim\_appointment\_id smallint(5) NOT NULL,

patient\_id smallint(5) NOT NULL,

hospitalsakila\_id smallint(5) NOT NULL,

PRIMARY KEY (dim\_appointment\_id)

);

CREATE table Fact\_Appointment\_Payment (

time\_id int,

hospital\_id smallint(5),

appointment\_id smallint(5),

city\_id smallint(5),

payment\_amount decimal(20,5),

FOREIGN KEY (hospital\_id) REFERENCES dim\_hospital(dim\_hospital\_id),

FOREIGN KEY (appointment\_id) REFERENCES dim\_appointment(dim\_appointment\_id),

FOREIGN KEY (city\_id) REFERENCES dim\_city(dim\_city\_id),

FOREIGN KEY (time\_id) REFERENCES dim\_time(id)

);

The figure below shows all the dimension tables and the fact table being created in the “dw\_hospitalsakila” schema.

Graphical user interface, text, application

Description automatically generated

Fig.7: OLAP tables created

**5 Findings from the Data Warehouse**

**5.1 Visualization and Analysis**

**Chart, bar chart

Description automatically generated**

Fig.8: Total payments summated for each hospital throughout the years

Chart, bar chart

Description automatically generated

Fig.9: Number of appointments in each hospital throughout the years

Table

Description automatically generated with medium confidence

Fig.10: Data summary

The grand total payments for all the hospitals throughout the years are 9.95 in 2005, 72.9 in 2006 and 40.95 in 2007. As shown in figure 9 above, the number of total appointments held in 2005 for both hospitals is 4 which is more than the 3 appointments held in 2007 for both hospitals. Even though in 2007 there were fewer appointments, the grand total payment is 31 units higher than that of 2005. This is simply due to pricing policy changes that have been applied after 2005, making the individual appointment pricings increase. Due to the presence of the 2005 data, when the finance managers at the Rhinic Hospital Organization do analyses, they can get the wrong idea that the hospitals did very bad financially in 2005 but somehow drastically improved in 2006 and 2007 which is not the case at all. This finding shows how irrelevant data can be identified and removed so that financial analyses can be made systematically and consistently.

**5.2 Summary of findings**

|  |  |  |
| --- | --- | --- |
| **Problem Statement** | **Business Scenario** | **Findings** |
| Old data record management | - Ability to create a report on amount of payment for appointments and identify outdated payment records  In order to clear out very old appointment records and know whether hospitals are doing well in certain areas financially, the finance managers from the Rhinic Hospital Organization want to generate a summary of payment records for appointments, which hospitals the appointments took place and where/when the appointments have occurred. | Chart, bar chart  Description automatically generated  Fig.11 : Appointment instances for hospital branches throughout the years  Table  Description automatically generated with medium confidence  Fig.12 : Summary of Data  The data warehouse processes helped us identify outdated data that is no longer important due to pricing policy changes in the Rhinic Hospital Organization. Even though in 2007 there were fewer appointments, the grand total payment is 31 units higher than that of 2005 for both hospital branches. This finding points out how the presence of the 2005 data can result in wrong financial analyses being made by finance managers. Therefore, through this finding, the Rhinic Hospital Organization will be able to pinpoint and eradicate irrelevant but problematic data so that analyses of financial flows can be made accurately and systematically. |

Fig.13: How findings benefit towards the problem statement and business scenario

**6 Conclusion**

In conclusion, the analyses to be made by finance managers for the long term will no longer be affected as the appointment payment records in 2005 will not be considered. The establishment of a systematic data warehouse automatically tackles a lot of issues as they can be easily identified and solved by data managers as well as the finance managers by making different types of analysis. More attributes will have to be stored in the long term as the healthcare landscape is changing due to technological advancements and increasing demands from consumers. Therefore, the Rhinic Hospital Organization will still improve upon the established data warehouse to further tackle the old data management problem in a bigger scope as there will be more hospital branches in the future.