Muhammad Shaheer (14944)
Haseeb Feroz (11496)

Project Report DW:
Sales Optimization through Online Ads:

Step 1 (Acquiring Business Knowledge for the Process):

The First Step was the building block for the Project, and the most essential one as well since this was where most of the information was gathered for the Database and the Dimensional Modeling Workbook. For the First Step, we contacted our seniors to get details on how large scale Ecommerce Sites like Daraz operate and manage their databases.

What we got to know was that these ecommerce sites did not maintain a separate database for their Ad campaigns, but rather the same one with different changes such as different tables in the schema. From the different Contacts and sites that were scouted, we were able to find different terms and different facts that these companies/whole industry used to keep in track and take measure of the multiple changes and important info.

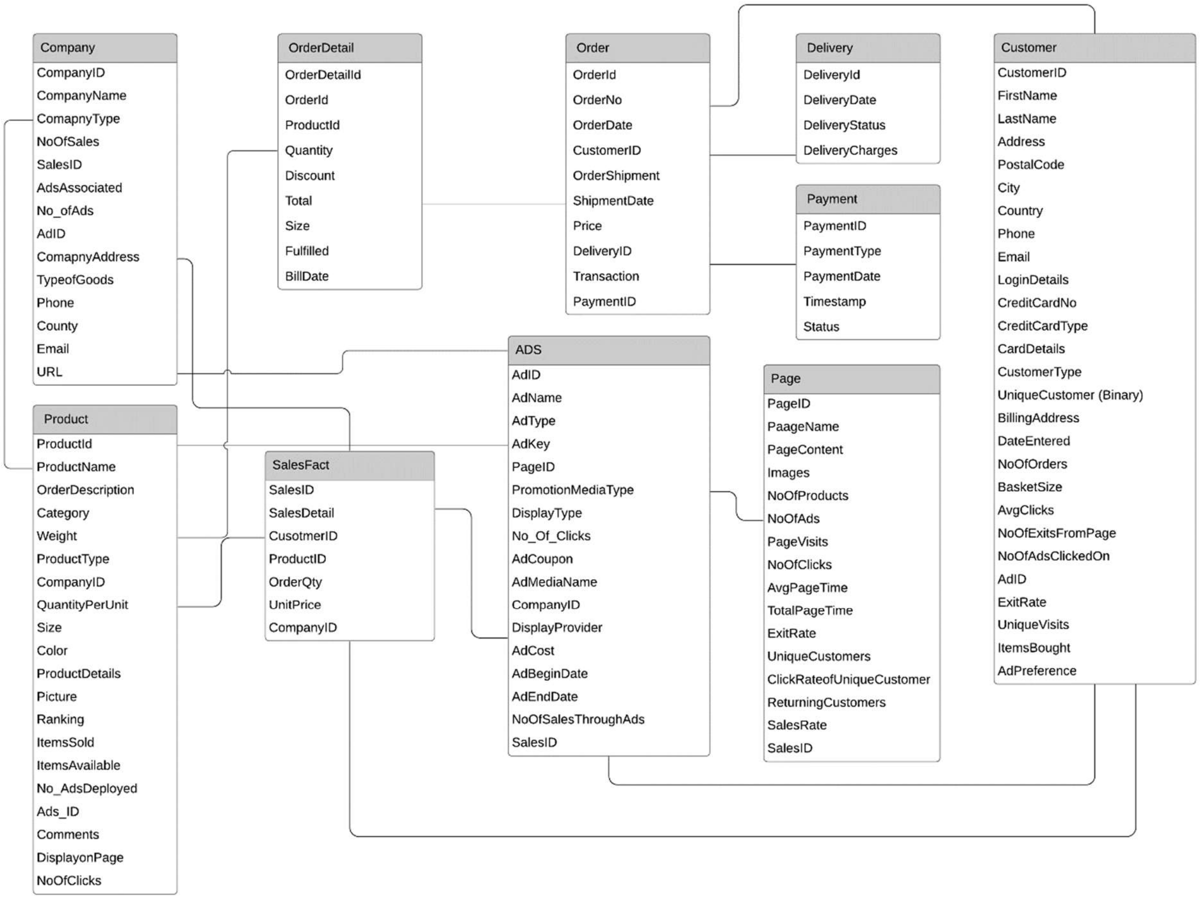
Comments:

The knowledge which was acquired from the different companies was very interesting and enlightening, as it showed how different companies operated and how they would deduce different information from the simplest of tasks/things we do on the internet.

Muhammad Shaheer (14944)
Haseeb Feroz (11496)

Step 2 (Generating OLTP Database):

In this step, we made an OLTP Database diagram from our research through our contacts indifferent companies that were working as data analysts and ELT technicians:



Step 3 (Generating Dimensional Model):

After generating the OLTP database, we worked on the dimensional modelling excel file, and filled it best to our ability. We came up with 2 fact tables: Namely Sales_Fact and Ad_Analysis_Fact.

The Sales_Fact table has a transactional grain type, where one row of the table represents one table. The Facts of the Fact table are as follows:

- **Order_ID:** Unique Identifier of Orders
- **Order_Date:** The date on which the order was placed
- **Shipping_Price:** The shipping price applied to the order
- **Item_Discount:** Discount applied to items in the order
- **Order_GrandTotal:** The grand total of the order, after discounts and deductions, etc.

- **Product_ID:** Identifier of products in the order
- **Customer_ID:** Identifier of the customer that placed the order
- **Store_ID:** Identifier of the store where the order was placed

The Ad_Analysis_Fact table has an operational grain type, where one row represents one Ad. The facts of the fact table are as follows:

- **Ad_ID:** Unique Identifier of the Ad
- **Ad_Name:** Name of the Ad
- **Ad_Type:** Type of Ad, such as image or video etc.
- **Ad_Duration:** Time period that the Ad was up
- **Page_ID:** Identifier of the page on which the Ad was displayed
- **Product_ID:** Identifier of the product being advertised in the Ad
- **Clicks_Per_View:** No. of clicks on the Ad
- **Unique_Visitors:** Number of unique visitors viewing the Ad
- **Successful_Sales:** Number of Ad clicks that led to sale of the product advertised in the Ad
- **Returning_Visitor:** Boolean value that tells whether the visitor is unique or not
- **Average_Screen_Time:** Average time the Ad was on the screen of a visitor

We can use these Facts tables to analyze the interactions of users/viewers of the Ads and whether they led to any successful sales or not. This can be by cross analyzing the orders from the Sales_Fact table with the Successful_Sales Fact in the Ad_Analysis_Table Fact table.

Step 4 (Data Lakes):

In this step we looked into what Data Lakes are and compared the different Data Lake solutions offered by different providers such as Microsoft Azure, Amazon AWS, Apache Hudi as well as a few open source options.

A data lake is a system of storage where you can store data in any form, whether it be structured (in the form of a relational database), semi-structured (in the form of excel or .CSV file formats), unstructured (in the form of raw files such as emails, PDFS etc.), or binary (in the form of videos and images). It has a flat architecture, where every data element is given a unique identifier and associated metadata information.

Key Components of Data Lakes:

Data Ingestion

A data lake should have the ability to handle data coming from different sources in different formats, where it cleans and transforms the data from the different formats so that analyzing it all and extracting information from all the data is easier.

Muhammad Shaheer (14944)
Haseeb Feroz (11496)

Data Storage

A data lake should provide significant data storage capacity since the data is coming from many different sources. It should also be highly scalable and be able to store raw and in-process data, supporting and allowing for compression and encryption on the various data formats that will be stored on the data lake. It is also imperative that data access be fast.

Data Governance

This involves managing the availability, security, usability, and integrity of the data stored in the data lake. Effective data governance ensures that the data is consistent and is not prone to misuse. The importance of data governance is increasing due to new data privacy regulations and also increased reliance on analytics to optimize organization operations as well as business decision making.

Data Discovery

After ingesting the data, that is after the data has been imported from different sources and stored into the data lake, understanding the data is of paramount importance before the data is prepared or analyzed. The stored data needs to be tagged, using unique identifiers, so that the raw data can be organized and interpreted to extract information.

Data Exploration and Visualization

Data exploration is the initial step in data analysis, where users explore large data sets to uncover initial patterns and points of interest. The aim of data exploration is to help the user create a broad picture of the data. Users can use data visualization tools to get a better sense of the data, so the users can pick out the relevant data, according to the analysis they want to do. Data exploration reduces work time and helps find useful actionable insights from the start, so users can perform better analysis

Different Data Lake services

Muhammad Shaheer (14944)
Haseeb Feroz (11496)

AWS

AWS offers a service called AWS Lake formation where it helps make it easy for users to make their own data lake. When making a data lake, a lot of manual work needs to be done, such as going to the different sources you want to take data from, import that data and clean it, as well as setting up security for the stored data, such as compression and encryption. With AWS Lake formation, you just define the data sources and the data access and security rules that you want to apply and AWS Lake formation does all the menial tasks for you.

Azure

Azure offers different services related to Data Lakes. It has differently named services for each of the categories of Analysis and Storage. For storage, there is Azure Data Lake Storage. This service only provides Storage capabilities, where Azure offers a cost effective, limitless storage capacity, as well as security features such as encryption and advanced threat protection. Azure Data Lake is built on Azure Blob storage, which is an object storage solution by Microsoft for the cloud.

For Analysis, there is Data Lake Analytics service. In this service, Azure offers a pay per job service, where you pay for only the processing power you use. The execution environment of Azure Data Lake Analytics also actively analyzes programs as they run and gives suggestions to improve performance and reduce cost. This feature coupled with the pay per job feature can make the use of this service very cost effective, since You might be requesting more processing power than you need to run, and Azure will recommend the amount you should, which will most likely be less than what you requested, significantly driving down your cost. Azure also offers processing programs in different languages, such as R, Python and .NET over petabytes of data, promising processing on demand and instant scaling, since there is no infrastructure to manage.

Apache Hudi

Apache Hudi Data Lake services offer real-time Ingestion as well as real-time Analytics, which would be perfect for lower scale data, that needs quick query responses such as for system monitoring and interactive real-time analysis applications. Hudi does not have any external dependencies, and so it enables faster analytics, without increasing operational overhead. One fundamental ability that Hudi provides is to build a chain of tables derived from each other, expressed as workflows. Often times the data arrives late, which means that the correctness of data can not be guaranteed. However, what Hudi does, is that it takes the incoming new data, and applies processing logic and efficiently updates the late data. This means that cases where data may come in late, such as data coming in from mobile devices, where the connection may be going on and off, the data is processed and updated as it is being ingested.

IBM

IBM provides a variety of different services regarding data lakes with respect to Storage, governance, and access and analysis. It provides high quality storage, with petabytes of storage, as well as efficient moving of data to and from the data lake. With regards to governance, IBM provides time-tested solutions that improve the quality of data, as well as efficient integration of data and security. IBM also provides a service called IBM Watson Studio, where you can use IBM Watson, IBM’s premiere AI for business analytics, to build and train AI and machine learning models to prepare and analyze data from your data lake.

Kylo

Kylo is an open-source data lake management software platform with self-service data ingestion and data preparation, while providing metadata management, governance, and security. What sets Kylo apart from the mainstream data lake management tools such as AWS and Azure is that it also provides monitoring of feeds and services in the data lake. This means that Kylo provides health indicators, which gives users visibility on service issues impacting availability, while also tracking the level of service from the source where the data is arriving from.