



Student Number: 10374210

Student Name: Sanjay Kumar

Module Name and Code: B9DA102 Data Storage Solutions
for Data Analytics

Module Lecturer: John Honan

Assignment Topic: Implementing Data Warehouse and Data
Visualization for Northwind Database

Source Database Explanation

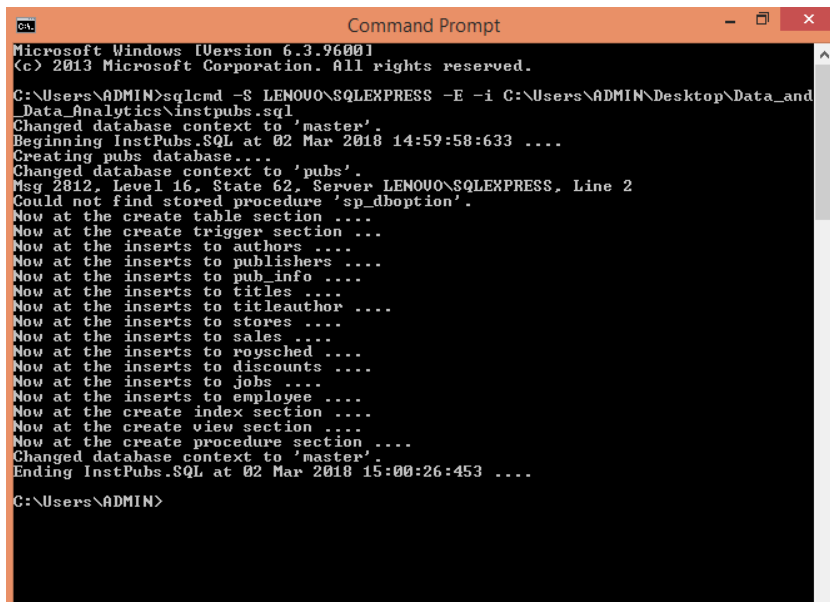
Northwind Database is a sample database about a company named "Northwind Traders". The database has stored all the sales transactions that occurred between 1996 to 1998, between the company and its customers and the purchase transactions between Northwind and its suppliers. (Microsoft Inc, 2018)

It contains the following detailed information:

- Suppliers/Vendors of Northwind – who supply to the company.
- Customers of Northwind – who buy from Northwind.
- Employee details of Northwind traders – who work for Northwind.
- The product information – the products that Northwind trades in.
- The shippers – details of the shippers who ship the products from the traders to the end-customers.
- Sales Order transaction – details of the transactions taking place between the customers & the company.

Exporting the source database to Microsoft SQL Server

- sqlcmd -S LENOVO\SQLEXPRESS -E -i C:\Users\ADMIN\Desktop\Data_and_Data_Analytics\instpubs.sql
- sqlcmd -S LENOVO\SQLEXPRESS -E -i C:\Users\ADMIN\Desktop\Data_and_Data_Analytics\instnwnd.sql



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\ADMIN>sqlcmd -S LENOVO\SQLEXPRESS -E -i C:\Users\ADMIN\Desktop\Data_and
Data_Analytics\instpubs.sql
Changed database context to 'master'.
Beginning InstPubs.SQL at 02 Mar 2018 14:59:58:633 ....
Creating pubs database....
Changed database context to 'pubs'.
Msg 2012, Level 16, State 62, Server LENOVO\SQLEXPRESS, Line 2
Could not find stored procedure 'sp_dboption'.
Now at the create table section ....
Now at the create trigger section ....
Now at the inserts to authors ....
Now at the inserts to publishers ....
Now at the inserts to pub_info ....
Now at the inserts to titles ....
Now at the inserts to titleauthor ....
Now at the inserts to stores ....
Now at the inserts to sales ....
Now at the inserts to roysched ....
Now at the inserts to discounts ....
Now at the inserts to jobs ....
Now at the inserts to employee ....
Now at the create index section ....
Now at the create view section ....
Now at the create procedure section ....
Changed database context to 'master'.
Ending InstPubs.SQL at 02 Mar 2018 15:00:26:453 ....
C:\Users\ADMIN>
```

Part 1 - Business Drivers

Subject Area

Internal process performance It is important for a company that buys and sells products that it understands its supply performance i.e. how well you fulfil orders and meet customer expectations. Delivery reliability is an important criterion for customers when choosing a supplier, particularly one that deals with perishable goods.

Northwind's logistics partners deliver products to customers. For the purposes of this assignment, the focus will be on logistics reliability. Delivery in full, on-time (DIFOT) is a key metric used to assess delivery reliability of an organization. DIFOT can be calculated using the following formula:

$$\text{DIFOT Rate} = \frac{\text{No. of shipments that arrived in full, on - time}}{\text{No. of shipments}}$$

As well as analyzing DIFOT rates, average freight costs for each logistics company will be reviewed in parallel. It is important when assessing reliability of Northwind's logistics providers that cost of service is also considered e.g. a more reliable service often incurs a higher expense. (The balance between maintaining customer satisfaction levels and generating sufficient revenue/profit can often be difficult.)

Key performance indicator (KPI): Delivery in full, on time, logistics reliability

Tables: Order Details, Orders, Customers, shippers, Employees

Case Study: Like one old Customers wants some new products which he never ordered so he ask the north wind employees to provide their logistics partner details who provide that product with good cost and delivery time.

Stack Holders

In my database we have used Shipper as a main stakeholder because the subject area is Shippers. Shippers Performance is based on how much time they take shipping. Delivered products to the Northwind of customer.

Part 2 - Data Modelling

ANALYSIS QUESTION

- ANALYSIS I – My Primary Analysis is the Analysis on the Key Performance Indicator for the Internal Process. Here the internal process is the Logistic Companies Efficiency. The Efficiency is determined by the Delivery Performance. Thus, my First Analysis is the Logistic Companies Delivery Performance.
- ANALYSIS II – Extending the Analysis I, the second analysis is Country-Wise Shipment Freight vs Average Shipper Price.
- ANALYSIS III – Third Analysis includes which Employee handle more shipping
- ANALYSIS IV – Forecast Sales Per Month and Year
- ANALYSIS V – City-Wise Shipping
- ANALYSIS VI – Country Wise Shipping

DATA WAREHOUSE REQUIREMENTS

According to the Analysis Topics, I have taken the Dimension tables as follows:

Employees: Contains the Employee information like Employee First and last Name, Title, Hire Date, and City etc.

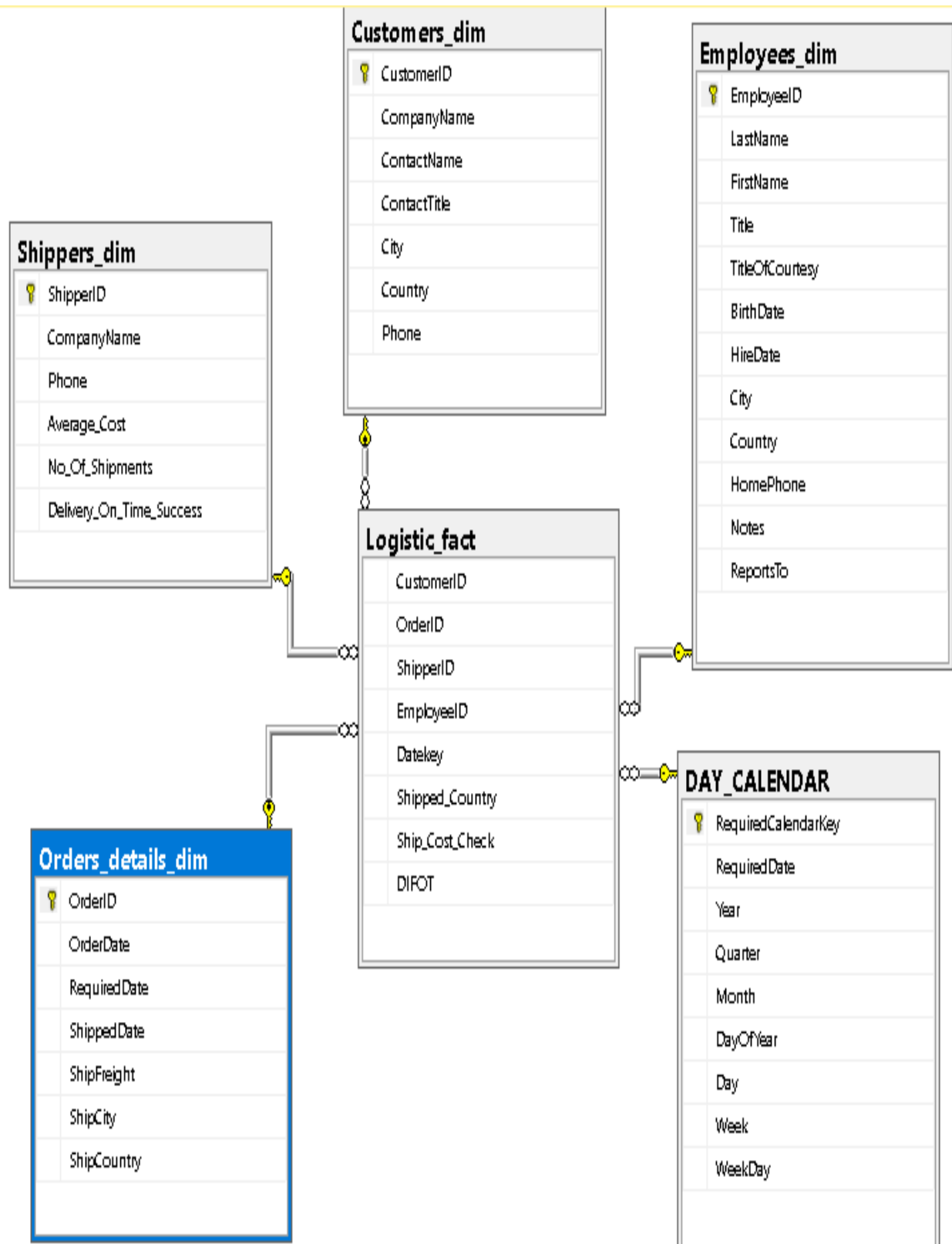
CUSTOMERS: Contains the Customer information like Customer Name, City, Region, Country, etc.

SHIPPER: Contains Shipper Details like Shipper Name, Shipper City, and Total No shipments and No of Successful Shipments etc.

ORDERCALENDAR: Contains Order Date, Year, Month, Quarter, Day of Month, Week, Week Day, etc.

And the fact table is Logistic table. It contains the primary keys of all the Dimension

Star Schema



Part 3 - Implement Tables and ETL Procedures

ETL PROCESS

Extraction – I have extracted the Source data from the Northwind Source database.

Transformation – The source data was then transformed to the requirement while loading the data to Dimension and fact tables in Data Warehouse. Few of the transformations are as follows:

- Delivery in full on Time Rate following calculation:

$$\text{DIFOT Rate} = \frac{\text{No. of shipments that arrived in full, on - time}}{\text{No. of shipments}}$$

- Ship Cost Check is a measure which check the cost of shipping is less than average or not
- Average Cost of each shipper this measure I put in the shippers Table
- Delivery on Time Successful this measure tells successful shipping out of total shipments
- Total no of shipments Done by Each Shipper
- Order Calendar Dimension – Extracting the year, month, quarter, week, weekday, etc. from the Order Date which is in datetime format.

Loading – The primary key is created for all the records pushed to Dimension tables in Data Warehouse from the Source. And the fact table is made with the primary keys of the dimension tables and other useful attributes from the source database tables.

Thus, the finished Data Warehouse is created using the ETL process where only useful data is pushed to Data Warehouse from the source.

Star Schema Development Queries

CREATE TABLE QUERIES



DatawarehouseCreateTables.sql

INSERT DATA QUERIES



DatawarehouseInsertQueries.sql

TABLEAU

Tableau is the most powerful, secure, and flexible end-to-end analytics platform for any meaningful data.

There is an exceptional analytics demand and hence, it's needed to quickly build powerful calculations from existing data, drag and drop reference lines and forecasts, and review statistical summaries. Tableau helps in every aspect of this and helps in trend analyses, regressions, and correlations and true statistical understanding.

Thus, helping to make data-driven decisions with confidence. (Tableau Inc, 2018)

TABLEAU ANALYSIS



TABLEAU
DATAWAREHOUSE /

Why I am using the Microsoft SQL Server DW solution

1. Security Features Are Good

SQL Server utilizes Policy-Based Management to distinguish security strategies that are rebellious. This component permits just approved work force access to the database. Security reviews and occasions can be composed consequently to log documents.

2. Improved Performance

The MS SQL server has worked in straightforward information pressure highlight alongside encryption. Clients don't have to adjust programs with a specific end goal to scramble the information. The MS SQL server approaches control combined with effective authorization administration instruments. Further, it offers an improved execution with regards to information gathering.

3. Free

SQL server incorporates successful information administration and information mining devices alongside plate parcelling. Your server's ideal support can be guaranteed by following viable information administration hones. These practices additionally enable you to guarantee the accessibility and recoverability of information.

Why SQL Server is better than any other RDBMS Applications

SQL Server resembles most RDMS frameworks, a database motor yet what improve it than different RDMS frameworks (particularly SQL 2008 onwards) are new highlights and different fancy odds and ends it accompanies.

Following are few comparisons with other RDBMS applications:

1. Easiest coordination with world's most basic database: Spreadsheet, Microsoft exceed expectations specifically and control turn has essentially improved its esteem.
2. Easy to use interface
3. Simple to make upkeep designs.
4. Incorporated Security (windows verification): This certainly help streamline server get to in view of Active registry approaches and gatherings.
5. Authorizing: The permitting structure of SQL Server is greatly improved when contrasted with different RDMS frameworks. Different RDMS frameworks have an exceptionally complex permitting structure which turns out to be much exorbitant than SQL Server.
6. SQL Server Management Studio (SSMS): As contrast with different RDMS frameworks devices, SSMS is the best device for an engineer or a DBA.
7. SQL Server Business Intelligence: Business Intelligence in SQL Server has progressed significantly and has advanced to such an extent. It is truly outstanding if not the best in the market right now.
8. Overseeing and Monitoring: SQL Server 2008 R2 has truly scored high in its new organization and checking instruments. It has made the life of a DBA (even unintentional DBA) a great deal less demanding

Part 4 TABLEAU ANALYSIS EXPLANATION

Analysis I:

My First Analysis is about the Delivery Performance of the Shipper. I have categorized the performance into three parts: Early Delivery, On time Delivery and Late Delivery.

There are three Types Shipping Companies whose Delivery Performance is plotted against the order number:

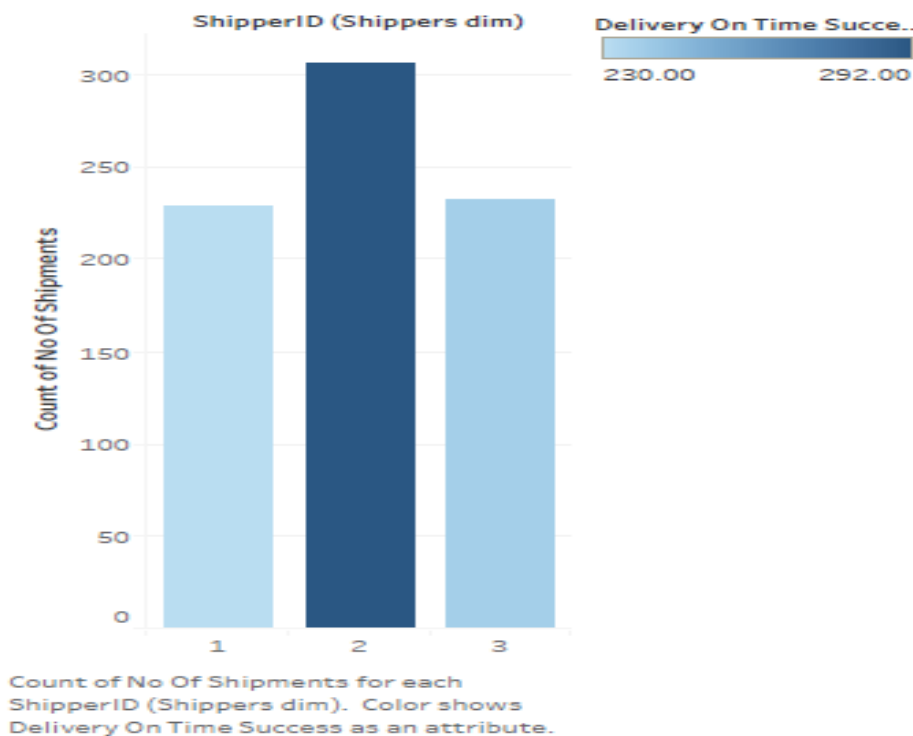
- Federal Shipping
- Speedy Express
- United Package

I have given the color code for the various bar graph like, Green color for On-Time Delivery, Blue color for an Early Delivery and Red color for Late Delivery.

Delivery Performance is calculated by following calculation:

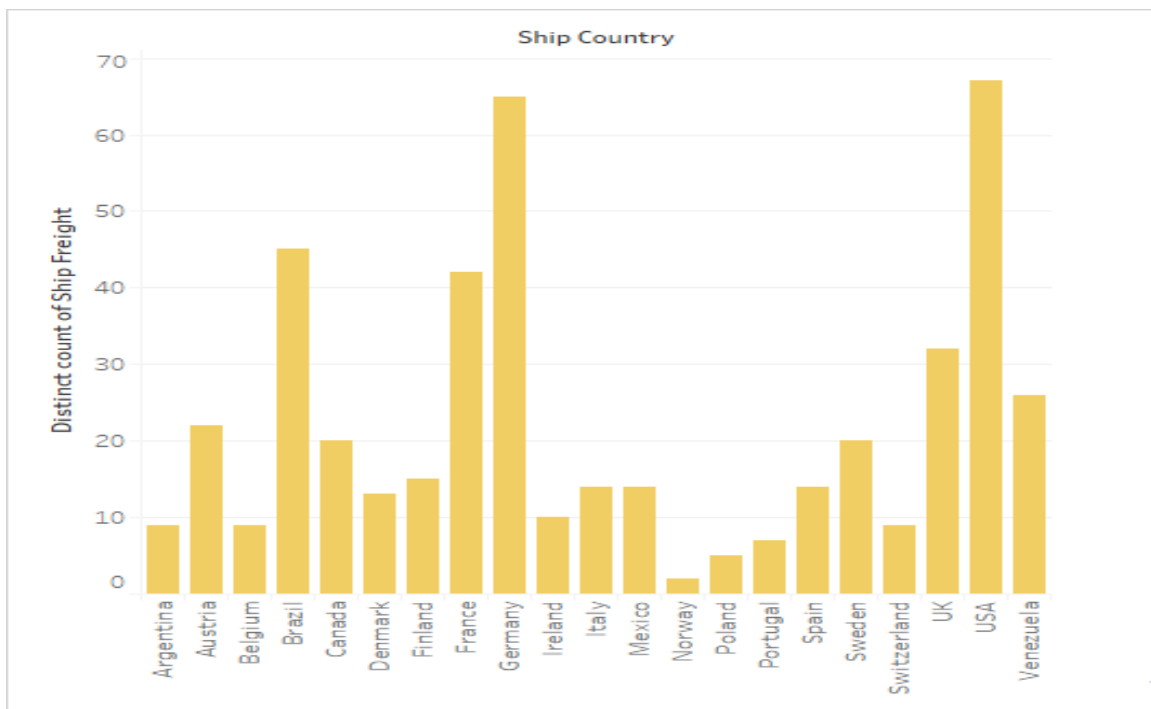
Successful Shipping (Required Date – Shipped Date)>0

Sheet 10



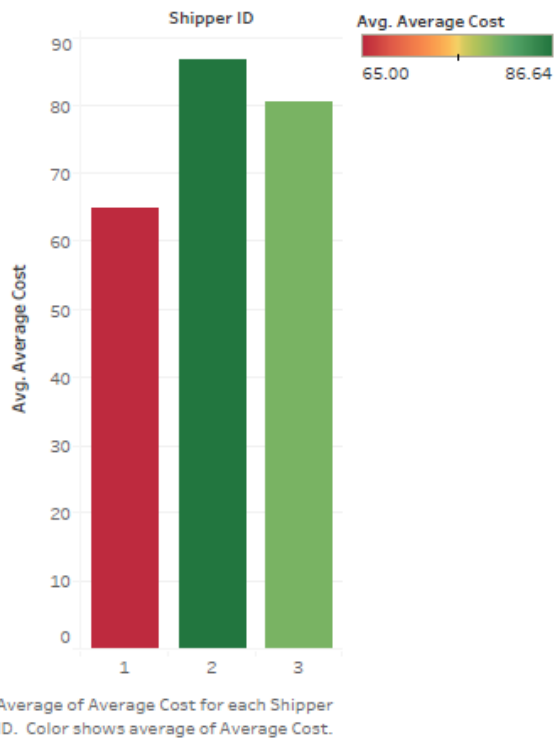
Ship Cost

Country Wise in which USA and Germany Have high ship Cost.



Shippers Average Cost per shipments in Dollars.

Sheet 4

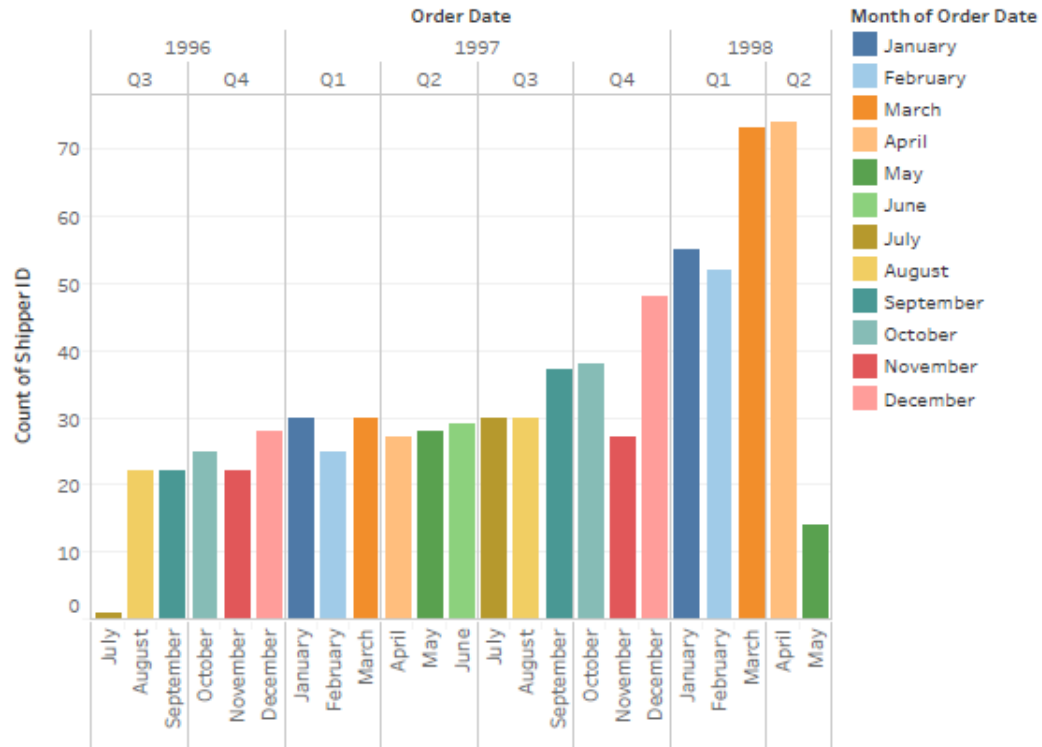


Employee who take more orders and passes more shipments.

Sheet 3

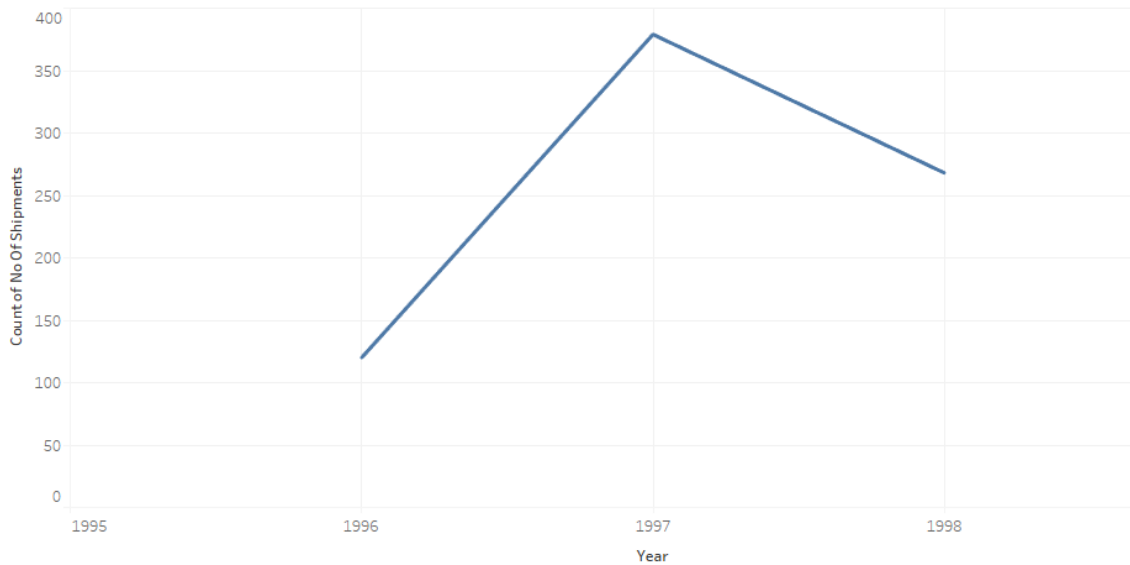


Month Wise and Year by Number of Orders



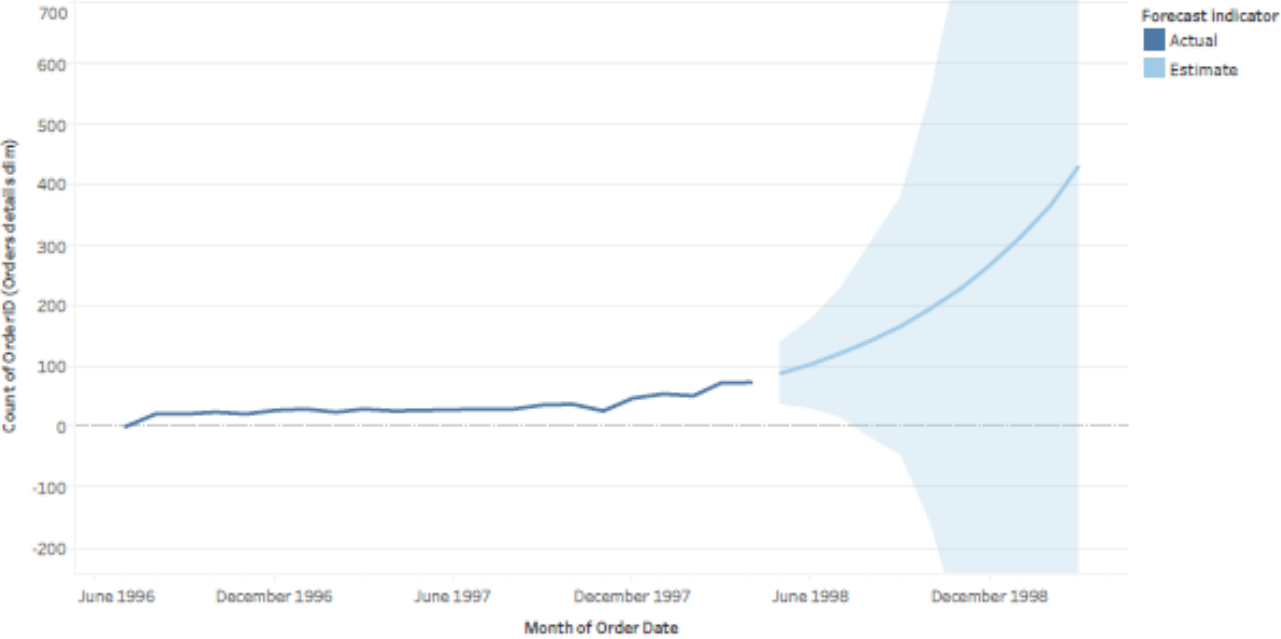
Count of Shipper ID for each Order Date Month broken down by Order Date Year and Order Date Quarter. Color shows details about Order Date Month.

Sheet 8



Forecast By 6 months of no of shipments

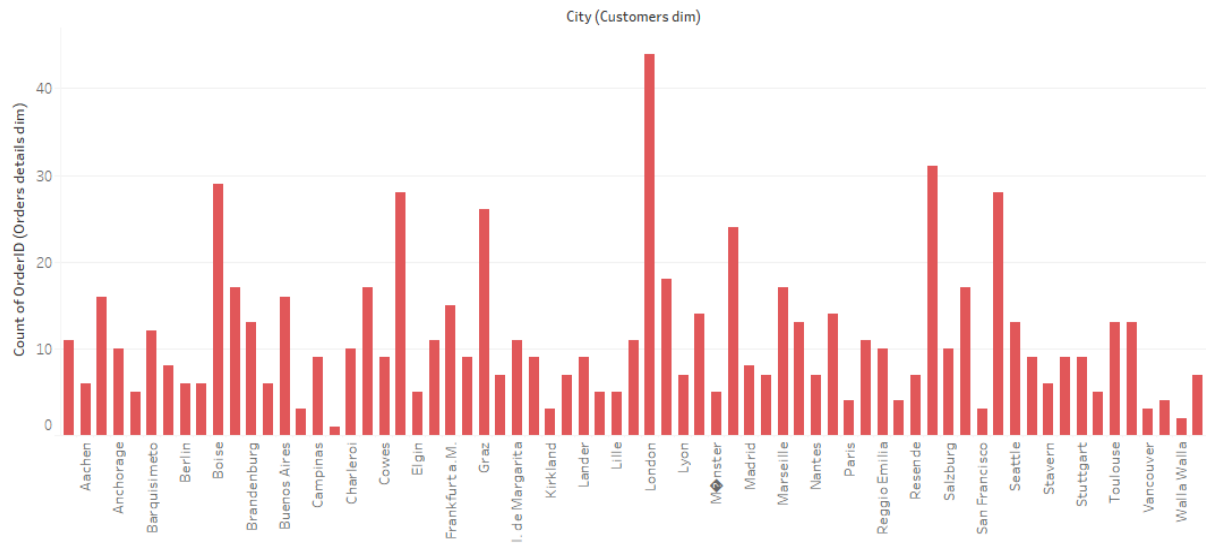
Sheet 6



The trend of count of OrderID (Orders details dim) (actual & forecast) for Order Date Month. Color shows details about Forecast Indicator.

City Wise Shipments in chart more shipments were delivered to city London

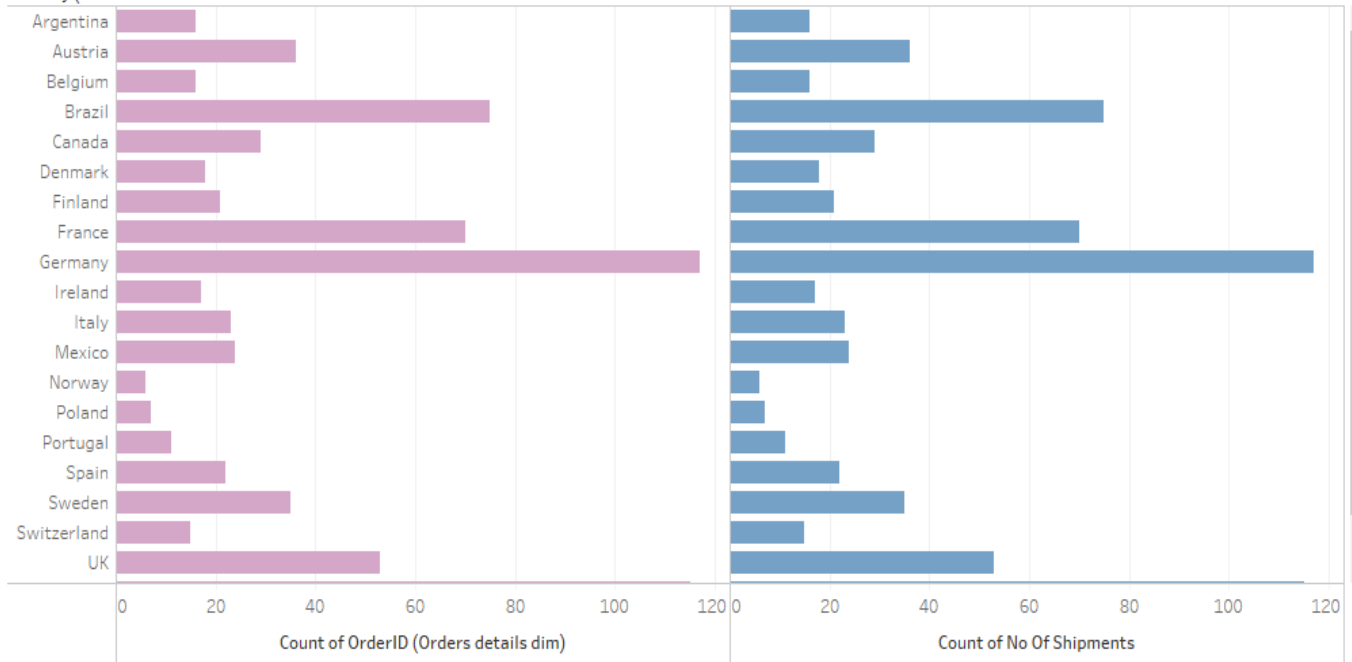
Sheet 5



Country Wise Shipments and Orders

Sheet 7

Country (Cu..



Part 5 - Modelling and Graph Data Models

5 Discuss the limitations of the relational ER and star schema dimensional models.
Implement the Northwind ER model as a graph using Neo4j.

ER Model:

There are 2 types of database designing from the system requirements:

- Top-Down approach or **Entity-Relationship Modelling**
- Bottom Up or Normalization approach

We are more interested in Entity-Relationship Modelling. It is a graphical strategy, which is utilized to change over the necessity of the framework to graphical portrayal, with the goal that it can turn out to be well justifiable. It likewise gives the structure to planning of database.

The Entity-Relationship (ER) show was initially proposed by Peter in 1976 as an approach to bring together the system and social database sees. Essentially expressed, the ER display is a reasonable information demonstrate that perspectives this present reality as elements and connections. An essential segment of the model is the Entity-Relationship graph, which is utilized to outwardly speak to information objects.

Limitations of Relational ER Dimensional Modelling:

- 1 Restricted validation constraints and specifications.
- 2 Loss of data content: Some data be lost or covered up in ER model.
- 3 Restricted relationship representation: ER model used to constrain relationship when contrasted with another information models like social model and so forth.
- 4 No representation of data manipulation: It is hard to demonstrate information control in ER model.
- 5 Mostly used for for high state design: ER model is exceptionally famous for outlining high level state of designs.

Star Schema Dimensional Modelling

Star schema model is the easiest and simplest dimensional model in terms of architecture. It is named as star schema because of the reason that it relates to a star. Which focus only to the centre table or data. In the middle of the star schema is the Fact Table which connects to other Dimension Tables.

Limitations of Star Schema: -

- 1) ER Model is 2-dimensional, Star Schema is Multi-Dimensional.
- 2) ER Model is Normalized, Star Schema is Denormalized.
- 3) No Reusability of Master Data: Since the Master Data table is going about as the Dimension table which is inside the cube, So the Master information table can't be reused.
- 4) Limited Analysis: Most extreme number of characteristics in a Fact table is: 16, Since every trademark column in the reality table associates with one Dimension table/Master information Table, So max no. of Dimension tables what we can have is restricted to 16.

Neo4j script for Northwind ER Model:

```
// Northwind Graph
```

```
: play northwind-graph
```

```
-----  
LOAD CSV WITH HEADERS FROM "http://data.neo4j.com/northwind/products.csv" AS  
row
```

```
CREATE (n:Product)
```

```
SET n = row,
```

```
    n.unitPrice = toFloat(row.unitPrice),
```

```
    n.unitsInStock = toInteger(row.unitsInStock), n.unitsOnOrder =  
toInteger(row.unitsOnOrder),
```

```
    n.reorderLevel = toInteger(row.reorderLevel), n.discontinued = (row.discontinued <>  
"o")  
-----
```



```
LOAD CSV WITH HEADERS FROM "http://data.neo4j.com/northwind/categories.csv" AS
row
```

```
CREATE (n:Category)
```

```
SET n = row
```

```
-----
LOAD CSV WITH HEADERS FROM "http://data.neo4j.com/northwind/suppliers.csv" AS
row
```

```
CREATE (n:Supplier)
```

```
SET n = row
```

```
-----
CREATE INDEX ON :Product(productID)
```

```
-----
CREATE INDEX ON :Category(categoryID)
```

```
-----
CREATE INDEX ON :Supplier(supplierID)
```

```
-----
1
```

```
Create data relationships:
```

```
MATCH (p:Product),(c:Category)
```

```
WHERE p.categoryID = c.categoryID
```

```
CREATE (p)-[:PART_OF]->(c)
```

```
-----
----
Query using patterns
```

```
1
```

```
MATCH (s:Supplier)-->(:Product)-->(c:Category)
```

```
RETURN s.companyName as Company, collect(distinct c.categoryName) as Categories
```

2

```
MATCH (c:Category {categoryName:"Produce"})<--(:Product)<--(s:Supplier)
```

```
RETURN DISTINCT s.companyName as ProduceSuppliers
```

Load and Index record

1

```
LOAD CSV WITH HEADERS FROM "http://data.neo4j.com/northwind/customers.csv" AS  
row
```

```
CREATE (n:Customer)
```

```
SET n = row
```

2

```
LOAD CSV WITH HEADERS FROM "http://data.neo4j.com/northwind/orders.csv" AS  
row
```

```
CREATE (n:Order)
```

```
SET n = row
```

3

```
CREATE INDEX ON :Customer(customerID)
```

```
CREATE INDEX ON :Order(orderID)
```

Data relationships:

```
MATCH (c:Customer),(o:Order)
```

```
WHERE c.customerID = o.customerID
```

```
CREATE (c)-[:PURCHASED]->(o)
```

Load and index records

```
LOAD CSV WITH HEADERS FROM "http://data.neo4j.com/northwind/order-details.csv"  
AS row
```

```
MATCH (p:Product), (o:Order)
```

```
WHERE p.productID = row.productID AND o.orderID = row.orderID
```

```
CREATE (o)-[details:ORDERS]->(p)
SET details = row,
  details.quantity = toInteger(row.quantity)
```

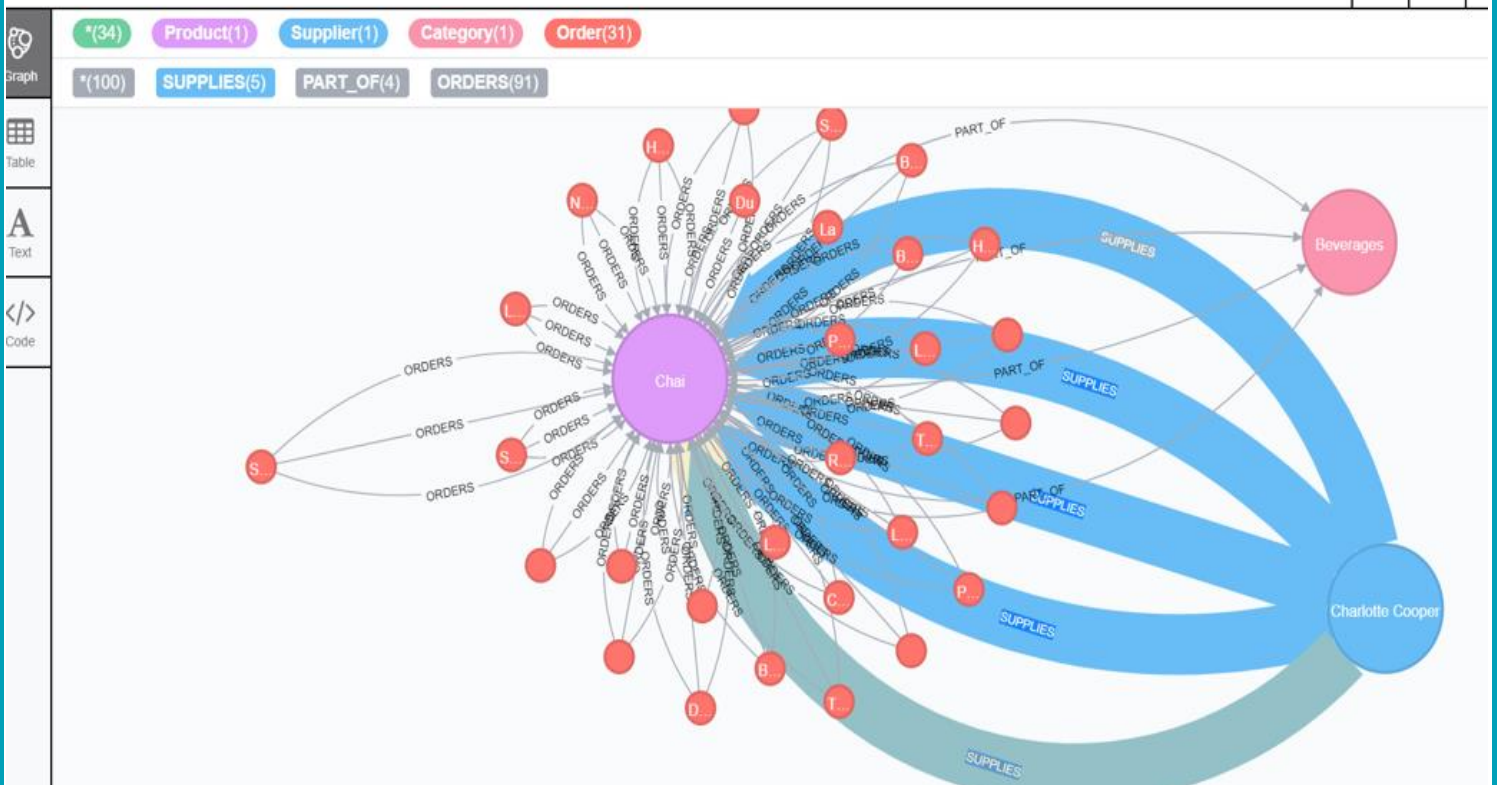
Query using patterns

```
MATCH (cust:Customer)-[:PURCHASED]->(:Order)-[o:ORDERS]->(p:Product),
  (p)-[:PART_OF]->(c:Category {categoryName:"Produce"})
RETURN DISTINCT cust.contactName as CustomerName, SUM(o.quantity) AS
TotalProductsPurchased
```

To see the graph:

```
MATCH (p:Product),(s:Supplier)
WHERE p.supplierID = s.supplierID
CREATE (s)-[:SUPPLIES]->(p) RETURN p
```

\$ MATCH (n) RETURN n LIMIT 1



Part 6 - Graph information retrieval and analysis

Evaluate the use of Neo4j and Cypher Query language for information retrieval of the Northwind data.

SOLD

```
MATCH (order:Order {orderId: row.OrderID})
MATCH (employee:Employee {employeeID: row.EmployeeID})
MERGE (employee)-[:SOLD]->(order);
```

REPORTS TO

```
MATCH (employee:Employee {employeeID: row.EmployeeID})
MATCH (manager:Employee {employeeID: row.ReportsTo})
MERGE (employee)-[:REPORTS_TO]->(manager);
```

How Many Orders were Made by Each Part of the Hierarchy?

```
MATCH (e:Employee)
OPTIONAL MATCH (e)-[:REPORTS_TO*0..]-(sub)-[:SOLD]->(order)
RETURN e.employeeID, [x IN COLLECT(DISTINCT sub.employeeID) WHERE
x <> e.employeeID] AS reports, COUNT(distinct order) AS
totalOrders
ORDER BY totalOrders DESC;
```

Which Employee had the Highest Cross-Selling Count of 'Chocolade' and Which Product?

```
MATCH (choc:Product {productName:'Chocolade'})<-[:PRODUCT]-
(:Order)<-[:SOLD]-(employee),
      (employee)-[:SOLD]->(o2)-[:PRODUCT]->(other:Product)
RETURN employee.employeeID, other.productName, count(distinct o2)
as count
ORDER BY count DESC
LIMIT 5;
```

Discuss the role of graph databases in the Hadoop-Spark Ecosystem.

First, I explain About HDFS and Spark

HDFS (Hadoop Distributed File System): HDFS allows you to store huge amounts of data in a distributed and a redundant manner. For example, a **1 GB** (i.e 1024 MB) text file can be split into **16 * 128MB** files and stored on 8 different nodes in a Hadoop cluster. Each split can be replicated **3** times for fault tolerance so that if 1 node goes down, you have backups. HDFS is good for sequential **write-once-and-read-many times** type access.

Spark Ecosystem Spark is a powerful open-source processing engine alternative to Hadoop. At first, it based on high speed, ease of use and increased developer productivity. Also, supports machine learning, real-time stream processing, and graph computations as well. Moreover, Spark provides in-memory computing capabilities. Also, supports a vast collection of applications. For ease of development, it also supports API's like Java, Python, R, and Scala.

About Graph Databases

A graph is composed of two elements: a node and a relationship. The data model for a graph database is also significantly simpler and more expressive than those of relational or other NoSQL databases. A graph database is a database that uses graph structures for semantic queries with nodes, edges and properties to represent and store data. Most graph databases are NoSQL in nature and store their data in a key-value store or document-oriented database.

Graph databases employ nodes, properties, and edges.

- Nodes represent entities such as people, businesses, accounts, or any other item you might want to keep track of.
- Properties are pertinent information that relate to nodes.
- Edges are the lines that connect nodes to nodes, or nodes to properties and they represent the relationship between the two. Most of the important information is stored in the edges. Meaningful patterns emerge when examining the connections and interconnections of nodes, properties, and edges.

Graph Database -

Graph databases effectively store data relationships; they're also flexible when expanding a data model or conforming to changing business needs.

- Flexibility,
- Performance
- Scalability

Graph Databases –

There are many graph databases available but Neo4j is very popular.

Use Cases –

- Real time recommendation
- Social Networking

Graph Databases in Hadoop and Spark Ecosystem

Hadoop is used everywhere to process large amounts of data. Graph Databases on the other hand are all combining highly connected, high quality data from a variety of sources. Using Hadoop to efficiently pre-process, filter and aggregate raw information to be suitable for Neo4j imports is a reasonable approach.

Real world, log-, sensor-, transaction- and event data is noisy. Most of the data frames don't add new information but are repetitive. For enriching a good graph model with variant information, you want to filter out the noise and project the raw data into a format that can be easily ingested in your graph