**Data Warehouse Design & Implementation Project**

**Introduction:**

This project aims to explore data warehouse requirements for Order Management, create a star schema, and deploy it on servers. Additionally, it involves building a data cube with specified measures and dimensions.

## Task1 Investigate the Facts and the dimensions of Order Management.

In the context of a data warehouse, fact tables and dimension tables are fundamental components that help organize and store data effectively.

### Fact tables

A fact table is a central table in a data warehouse schema that stores the "what happened" numerical data, regarding business processes or events like ordering, shipping, invoicing, etc.

#### Order transaction fact table

A screenshot of a computer

Description automatically generated

**Fact Table Description:**

1)**Fact table:** Order transaction fact table.

2)**Type of the fact table:** Accumulating Snapshot Grain because it captures various stages of an order transaction through multiple date foreign keys, each representing a significant milestone.

3)**Granularity:** The granularity for an order transaction fact table is one row for each line item on an order.

**Measures Analysis:**

1) Order Line Quantity:

Type: Additive. Quantity can be summed across different transactions to calculate the total quantity of products shipped.

Null Values: not null. Null values in Quantity are unlikely as it represents the number of units sold, which should always have a numerical value.

2) Extended Order Line Gross Dollar Amount:

Type: Additive. Represents the total gross dollar amount for each order line and can be summed across transactions.

Null Values: Null. This measure might have null values if the gross dollar amount for a specific order line is not recorded.

3) Extended Order Line Discount Dollar Amount:

Type: Additive. Represents the total discount amount applied to each order line and can be summed across transactions.

Null Values: Null. This measure might have null values if no discount was applied to a specific order line.

4) Extended Order Line Net Dollar Amount:

Type: Additive. Represents the total net dollar amount for each order line and can be summed across transactions.

Null Values: Null. This measure could have null values if the net dollar amount for a specific order line is not recorded.

#### Sales Rep-Customer Assignment Fact

A diagram of a sales report

Description automatically generated with medium confidence

**Fact Table Description:**

1)**Fact table:** Sales Rep-Customer Assignment Fact table

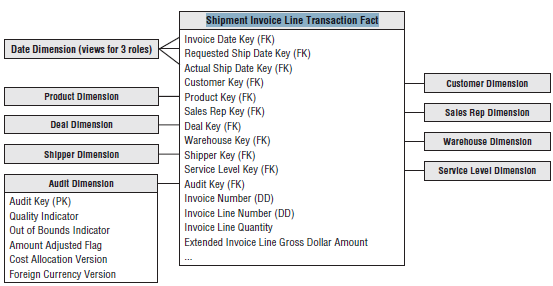
2)**Type of the fact table:** Accumulating Snapshot Grain because it captures different milestones in the sales process represented by multiple date foreign keys.

3)**Granularity:** Each row in the "Sales Rep-Customer Assignment" table represents a specific assignment of a sales representative to a customer, providing detailed information about the duration and specifics of the assignment.

**Measures Analysis:**

The Sales Rep-Customer Assignment Fact is a factless fact table that does not have measures.

#### Shipment Invoice Line Transaction Fact table



**Fact Table Description:**

1)**Fact table:** Shipment Invoice Line Transaction Fact table.

2)**Type of the fact table:** Accumulating Snapshot Grain due to multiple date foreign keys representing different milestones in the shipment and invoicing process.

3)**Granularity:** Each row in the "Shipment Invoice Line Transaction" table provides detailed information about a specific shipment and invoice line transaction.

**Measures Analysis:**

1) Invoice Line Quantity:

Type: Additive. Invoice Line Quantity can be summed across transactions to get a total quantity of products invoiced and shipped.

Null Values: Not null. Quantity should always have a numerical value as it represents the number of units invoiced and shipped.

2) Extended Invoice Line Gross Dollar Amount:

Type: Additive. Extended Invoice Line Gross Dollar Amount can be summed across transactions to calculate the total gross amount of the invoiced line items.

Null Values: Null. Null values might occur if the gross dollar amount for a specific invoice line is not recorded.

3) Extended Invoice Line Allowance Dollar Amount:

Type: Additive. Extended Invoice Line Allowance Dollar Amount can be summed across transactions to get the total allowance amount applied to the invoiced line items.

Null Values: Null. Null values might occur if no allowance was applied to a specific invoice line.

4) Extended Invoice Line Discount Dollar Amount:

Type: Additive. Extended Invoice Line Discount Dollar Amount represents the total discount amount applied to the invoiced line items and can be summed across transactions.

Null Values: Null. Null values might occur if no discount was applied to a specific invoice line.

5) Extended Invoice Line Net Dollar Amount:

Type: Additive. Extended Invoice Line Net Dollar Amount can be summed across transactions to get the total net amount for the invoiced line items.

Null Values: Null. Null values might occur if the net dollar amount for a specific invoice line is not recorded.

6)Extended Invoice Line Fixed Mfg Cost Dollar Amount:

Type: Additive. Extended Invoice Line Fixed Mfg Cost Dollar Amount can be summed across transactions .

Null Values: Null. Null values might occur if the net dollar amount for a specific invoice line is not recorded.

7)Extended Invoice Line Variable Mfg Cost Dollar Amount:

Type: Additive. Extended Invoice Line Variable Mfg Cost Dollar Amount can be summed across transactions .

Null Values: Null. Null values might occur if the Variable Mfg Cost Dollar Amount for a specific invoice line is not recorded.

8)Extended Invoice Line Storage Cost Dollar Amount:

Type: Additive. Extended Invoice Line Storage Cost Dollar Amount can be summed across transactions .

Null Values: Null. Null values might occur if the Storage Cost Dollar Amount for a specific invoice line is not recorded.

9)Extended Invoice Line Distribution Cost Dollar Amount:

Type: Additive. Extended Invoice Line Distribution Cost Dollar Amount can be summed across transactions .

Null Values: Null. Null values might occur if the Distribution Cost Dollar Amount for a specific invoice line is not recorded.

10) Extended Invoice Line Contribution Dollar Amount:

Type: Additive. Extended Invoice Line Contribution Dollar Amount represents the contribution margin for the invoiced line items and can be summed across transactions.

Null Values: Null. Null values might occur if the contribution dollar amount for a specific invoice line is not recorded.

11)Shipment On-Time Counter:

Type: Additive. Shipment On-Time Counter counts the number of shipments that were delivered on time.

Null Values: Not Null. This measure should always have a numerical value representing the count of on-time shipments.

12)Requested to Actual Ship Lag:

Type: Semi-additive. Requested to Actual Ship Lag represents the difference in time between the requested and actual ship dates. It measures the delay in shipment.

Null Values: Null. Null values might occur if the requested or actual ship dates are not recorded for a specific shipment.

#### Order Fulfillment Accumulating Fact

A diagram of a data flow

Description automatically generated with medium confidence

**Fact Table Description:**

1)**Fact table:** Order Fulfillment Accumulating Fact.

2)**Type of the fact table:** Accumulating Snapshot Grain because it captures different milestones in the order fulfillment process represented by multiple date foreign keys.

3)**Granularity:** Each row in the " Order Fulfillment Accumulating Fact " table provides detailed information about a specific invoice line transaction.

**Measures Analysis:**

1. Order Quantity:

Type: Additive. Order Quantity can be summed across different transactions to get a total quantity of products ordered.

Null Values: Not null. Order Quantity is a fundamental measure and is unlikely to have null values.

1. Extended Order Line Dollar Amount:

Type: Additive. Extended Order Line Dollar Amount represents the total dollar value of the order line.

Null Values: Null. This measure might have null values if the price information for a particular product is not available at the time of order creation.

1. Release to Manufacturing Quantity:

Type: Additive. Release to Manufacturing Quantity can be summed across different transactions.

Null Values: Null. This measure might have null values if the product is not released for manufacturing.

1. Manufacturing Pass Inspection Quantity:

Type: Additive. Manufacturing Pass Inspection Quantity can be summed across different transactions.

Null Values: Null. This measure might have null values if the products do not pass quality inspections, indicating potential quality control issues.

1. Manufacturing Fail Inspection Quantity:

Type: Additive. Manufacturing Fail Inspection Quantity can be summed across different transactions.

Null Values: Null. This measure might have null values if all products pass quality inspections.

1. Finished Goods Inventory Quantity:

Type: Additive. Finished Goods Inventory Quantity can be summed across different transactions.

Null Values: Null. This measure might have null values if products are directly shipped after manufacturing without being stored in inventory.

1. Authorized to Sell Quantity:

Type: Additive. Authorized to Sell Quantity can be summed across different transactions.

Null Values: Null. This measure might have null values if products are not authorized for sale.

1. Shipment Quantity:

Type: Additive. Shipment Quantity can be summed across different transactions.

Null Values: Null. This measure might have null values if the products are not shipped.

1. Shipment Damage Quantity:

Type: Additive. Shipment Damage Quantity can be summed across different transactions.

Null Values: Null. This measure might have null values if no products are damaged during shipment.

1. Customer Return Quantity:

Type: Additive. Customer Return Quantity can be summed across different transactions.

Null Values: Null. This measure might have null values if no products are returned.

1. Invoice Quantity:

Type: Additive. Invoice Quantity can be summed across different transactions.

Null Values: Null. This measure might have null values if no products are invoiced.

1. Extended Invoice Dollar Amount:

Type: Additive. Extended Invoice Dollar Amount represents the total dollar value of products listed on invoices.

Null Values: Null. This measure might have null values if the pricing information for products is not available during the invoicing process.

1. Order to Manufacturing Release Lag:

Type: Semi-additive. Represents the time taken from order creation to the release of products for manufacturing.

Null Values: Null. This measure might have null values if the time lag information is not recorded.

1. Manufacturing Release to Inventory Lag:

Type: Semi-additive. Represents the time taken from product release from manufacturing to placement in inventory.

Null Values: Null. This measure might have null values if the time lag information is not recorded.

1. Inventory to Shipment Lag:

Type: Semi-additive. Represents the time taken from product selection from inventory to shipment.

Null Values: Null. This measure might have null values if the time lag information is not recorded.

1. Order to Shipment Lag:

Type: Semi-additive. Represents the total time taken from order creation to product shipment.

Null Values: Null. This measure might have null values if the time lag information is not recorded.

### Dimension tables

A dimension table is a descriptive table in a data warehouse that provides the "who, what, where, when, why" context, enabling comprehensive analysis in data warehousing scenarios.

|  |  |  |  |
| --- | --- | --- | --- |
| **Dimension Name** | **Dimension’s fact table** | **type** | **Hierarchical Attributes** |
| Date Dimension | Order transaction & Shipment Invoice Line Transaction & Order Fulfillment Accumulating | Date Dimension&  Role Playing Dimension | Hierarchies are always applied, and transactions typically roll up according to multiple hierarchies such as day, month, year, etc. |
| Sales Rep Dimension | Order transaction & Shipment Invoice Line Transaction & Order Fulfillment Accumulating | regular dimension |  |
| Customer Dimension | Order transaction & Shipment Invoice Line Transaction & Order Fulfillment Accumulating | regular dimension |  |
| Product Dimension | Order transaction & Shipment Invoice Line Transaction & Order Fulfillment Accumulating | regular dimension | Hierarchies are often applied, leading transactions to roll up according to multiple product hierarchies. |
| Deal Dimension | Order transaction & Shipment Invoice Line Transaction & Order Fulfillment Accumulating | regular dimension |  |
| Shipper Dimension | Shipment Invoice Line Transaction & Order Fulfillment Accumulating | regular dimension |  |
| Warehouse Dimension | Shipment Invoice Line Transaction & Order Fulfillment Accumulating | regular dimension |  |
| Service Level Dimension | Shipment Invoice Line Transaction | regular dimension |  |
| Audit Dimension | Shipment Invoice Line Transaction | regular dimension |  |
| Manufacturing Facility Dimension | Order Fulfillment Accumulating | regular dimension |  |

## Task2 Create a relational database schema

Here is the UML Class Diagram for the Order Star Schema. All tables will be created next.A screenshot of a diagram

Description automatically generated

## Task3 Create tables on the servers and create materialized Views.

### Use database A2

--A2 Database is for Data Warehouse Assignment 2

USE A2

### Create the A2DimDate Dimension table and create clustered index for its PK

--Create a dimension table A2DimDate by copying [WideWorldImportersDW].Dimension.Date

SELECT \* INTO A2DimDate

FROM [WideWorldImportersDW].Dimension.Date;

-- Add the primary key constraint to the A2DimDate table

ALTER TABLE A2DimDate

ADD CONSTRAINT PK\_A2DimDate PRIMARY KEY (Date);

--Create a clustered index on the "Date" column of A2DimDate

CREATE CLUSTERED INDEX IX\_Clustered\_Date ON A2DimDate (Date);

### Create the A2DimShipper Dimension table with a clustered index for its PK

--Create the A2DimShipper Dimension Table with a clustered index for its PK

CREATE TABLE A2DimShipper (

ShipperKey INT PRIMARY KEY NOT NULL,

CompanyName NVARCHAR(255) NOT NULL

);

-- Inserting data into A2DimShipper table

INSERT INTO A2DimShipper (ShipperKey, CompanyName)

VALUES

(1, 'Vins et alcools Chevalier'),

(2, 'Toms Spezialitäten');

### Create the A2DimCustomer Dimension Table with a clustered index for its PK

-- Create the A2DimCustomer Dimension Table with clustered index for its PK

CREATE TABLE A2DimCustomer (

CustomerKey INT IDENTITY(1,1) PRIMARY KEY,

CompanyName NVARCHAR(100) NOT NULL,

Address NVARCHAR(100),

PostalCode NVARCHAR(10),

City NVARCHAR(15),

Region NVARCHAR(15),

Country NVARCHAR(15),

Phone NVARCHAR(25)

);

-- Inserting data into A2DimCustomer table

INSERT INTO A2DimCustomer (CompanyName, Address, PostalCode, City, Region, Country, Phone)

VALUES

('Bottom-Dollar Markets', '23 Tsawassen Blvd.', 'T2F 8M4', 'Tsawassen', 'BC', 'Canada', '(604) 555-4729'),

('Comércio Mineiro', 'Av. dos Lusíadas, 23', '05432-043', 'Sao Paulo', 'SP', 'Brazil', '(11) 555-7647');

### Create the A2DimProduct Dimension Table with a clustered index for its PK

-- CreatE the A2DimProduct Dimension Table

CREATE TABLE A2DimProduct (

ProductKey INT IDENTITY(1,1) PRIMARY KEY,

ProductName NVARCHAR(100) NOT NULL,

QuantityPerUnit NVARCHAR(50),

UnitPrice MONEY ,

CategoryName NVARCHAR(20) ,

CategoryDescription NVARCHAR(100)

);

-- Inserting data into A2DimProduct table

INSERT INTO A2DimProduct (ProductName, QuantityPerUnit, UnitPrice, CategoryName, CategoryDescription)

VALUES ('Air cushion machine (Blue)', '1', 1899.00, 'Machinery', 'Machinery and Equipment for Industrial Use'),

('Air cushion film 200mmx200mm 325m', '1', 90.00, 'Packaging Supplies', 'Materials Used for Packaging and Shipping'),

('Bubblewrap dispenser (Red)', '1.5m', 240.00, 'Packaging Supplies', 'Materials Used for Packaging and Shipping'),

('Ride on vintage American toy coupe (Black)', '1/12 scale', 285.00, 'Toys', 'Children Toys and Games');

### Create the A2FactOrder Fact Table with non-clustered index for its PK and FK

-- Creating A2FactOrder Fact Table with Non-Clustered Indexes

CREATE TABLE A2FactOrder (

OrderKey INT IDENTITY(1,1) PRIMARY KEY NONCLUSTERED,

CustomerKey INT NOT NULL,

ProductKey INT NOT NULL,

OrderDateKey date NOT NULL, -- Assuming this is a reference to a Date dimension

DueDateKey date NOT NULL, -- Assuming this is a reference to a Date dimension

ShippedDateKey date NOT NULL, -- Assuming this is a reference to a Date dimension

ShipperKey INT NOT NULL,

Quantity INT,

UnitPrice DECIMAL(18,2),

TaxRate DECIMAL(18,3),

FOREIGN KEY (CustomerKey) REFERENCES A2DimCustomer(CustomerKey), -- Reference your Customer dimension

FOREIGN KEY (ProductKey) REFERENCES A2DimProduct(ProductKey), -- Reference your Product dimension

FOREIGN KEY (OrderDateKey) REFERENCES A2DimDate(Date), -- Reference your Date dimension

FOREIGN KEY (DueDateKey) REFERENCES A2DimDate(Date), -- Reference your Date dimension

FOREIGN KEY (ShippedDateKey) REFERENCES A2DimDate(Date), -- Reference your Date dimension

FOREIGN KEY (ShipperKey) REFERENCES A2DimShipper(ShipperKey) -- Reference your Shipper dimension

);

-- Creating Non-Clustered Indexes

CREATE NONCLUSTERED INDEX IX\_CustomerKey ON A2FactOrder(CustomerKey);

CREATE NONCLUSTERED INDEX IX\_ProductKey ON A2FactOrder(ProductKey);

CREATE NONCLUSTERED INDEX IX\_OrderDateKey ON A2FactOrder(OrderDateKey);

CREATE NONCLUSTERED INDEX IX\_DueDateKey ON A2FactOrder(DueDateKey);

CREATE NONCLUSTERED INDEX IX\_ShippedDateKey ON A2FactOrder(ShippedDateKey);

CREATE NONCLUSTERED INDEX IX\_ShipperKey ON A2FactOrder(ShipperKey);

-- Inserting data into A2FactOrder Fact Table

INSERT INTO A2FactOrder (CustomerKey, ProductKey, OrderDateKey, DueDateKey, ShippedDateKey, ShipperKey, Quantity, UnitPrice, TaxRate)

VALUES

-- Inserting first set of data

(1, 3, '2013-01-05', '2013-04-05', '2013-03-26', 1, 50, 240.00, 13.000),

(2, 1, '2013-01-06', '2013-04-06', '2013-03-27', 2, 50, 1899.00, 20.000),

(2, 2, '2013-01-06', '2013-04-06', '2013-03-27', 2, 500, 90.00, 14.000),

(2, 3, '2013-01-07', '2013-04-07', '2013-03-28', 2, 200, 240.00, 13.000),

(1, 1, '2013-01-08', '2013-04-08', '2013-03-29', 1, 10, 1899.00, 20.000),

(1, 2, '2013-01-08', '2013-04-08', '2013-03-29', 1, 100, 90.00, 14.000),

(1, 4, '2013-01-16', '2013-04-16', '2013-04-06', 1, 80, 285.00, 14.000),

(2, 4, '2013-01-16', '2013-04-16', '2013-04-06', 2, 100, 285.00, 14.000),

(1, 1, '2013-02-01', '2013-05-02', '2013-04-22', 1, 20, 1899.00, 20.000),

(1, 2, '2013-02-01', '2013-05-02', '2013-04-22', 1, 200, 90.00, 14.000),

(2, 1, '2013-02-05', '2013-05-06', '2013-04-26', 2, 40, 1899.00, 20.000),

(2, 2, '2013-02-05', '2013-05-06', '2013-04-26', 2, 400, 90.00, 14.000),

(2, 3, '2013-02-09', '2013-05-10', '2013-04-30', 2, 150, 240.00, 13.000),

(1, 3, '2013-02-10', '2013-05-11', '2013-05-01', 1, 60, 240.00, 13.000),

(1, 4, '2013-02-17', '2013-05-18', '2013-05-08', 1, 100, 285.00, 14.000),

(2, 4, '2013-02-17', '2013-05-18', '2013-05-08', 2, 200, 285.00, 14.000),

(2, 1, '2013-03-08', '2013-06-06', '2013-05-27', 2, 30, 1899.00, 20.000),

(2, 2, '2013-03-08', '2013-06-06', '2013-05-27', 2, 300, 90.00, 14.000),

(2, 3, '2013-03-11', '2013-06-09', '2013-05-30', 2, 180, 240.00, 13.000),

(1, 1, '2013-03-15', '2013-06-13', '2013-06-03', 1, 30, 1899.00, 20.000),

(1, 2, '2013-03-15', '2013-06-13', '2013-06-03', 1, 300, 90.00, 14.000),

(1, 4, '2013-03-18', '2013-06-16', '2013-06-06', 1, 80, 285.00, 14.000),

(2, 4, '2013-03-18', '2013-06-16', '2013-06-06', 2, 170, 285.00, 14.000),

(1, 3, '2013-03-25', '2013-06-23', '2013-06-13', 1, 80, 240.00, 13.000),

(2, 1, '2013-04-09', '2013-06-23', '2013-06-13', 2, 20, 1899.00, 20.000),

(2, 2, '2013-04-09', '2013-06-23', '2013-06-13', 2, 200, 90.00, 14.000),

(2, 3, '2013-04-13', '2013-06-27', '2013-06-17', 2, 120, 240.00, 13.000),

(1, 3, '2013-04-17', '2013-07-01', '2013-06-21', 1, 90, 240.00, 13.000),

(1, 1, '2013-04-18', '2013-07-02', '2013-06-22', 1, 40, 1899.00, 20.000),

(1, 2, '2013-04-18', '2013-07-02', '2013-06-22', 1, 400, 90.00, 14.000),

(1, 4, '2013-04-19', '2013-07-03', '2013-06-23', 1, 70, 285.00, 14.000),

(2, 4, '2013-04-19', '2013-07-03', '2013-06-23', 2, 110, 285.00, 14.000),

(2, 1, '2013-05-02', '2013-06-21', '2013-06-16', 2, 10, 1899.00, 20.000),

(2, 2, '2013-05-02', '2013-06-21', '2013-06-16', 2, 100, 90.00, 14.000),

(1, 4, '2013-05-20', '2013-07-09', '2013-07-04', 1, 60, 285.00, 14.000),

(2, 4, '2013-05-20', '2013-07-09', '2013-07-04', 2, 90, 285.00, 14.000),

(2, 3, '2013-05-22', '2013-07-11', '2013-07-06', 2, 80, 240.00, 13.000),

(1, 3, '2013-05-25', '2013-07-14', '2013-07-09', 1, 100, 240.00, 13.000),

(1, 1, '2013-05-28', '2013-07-17', '2013-07-12', 1, 50, 1899.00, 20.000),

(1, 2, '2013-05-28', '2013-07-17', '2013-07-12', 1, 500, 90.00, 14.000);

### Create Materialized View

-- Create View A2FactOrderTotalValuePerCustomer

CREATE VIEW dbo.A2FactOrderTotalValuePerCustomer

WITH SCHEMABINDING

AS

SELECT

CustomerKey,

SUM(ISNULL(Quantity, 0) \* ISNULL(UnitPrice, 0) \* (1 + ISNULL(TaxRate, 0)/100)) AS OrderTotalValue,

COUNT\_BIG(\*) AS CountBig

FROM dbo.A2FactOrder

GROUP BY CustomerKey;

-- Create Unique Clustered Index on the Indexed View

CREATE UNIQUE CLUSTERED INDEX IDX\_V1

ON dbo.A2FactOrderTotalValuePerCustomer(CustomerKey);

## Task4 Create a data cube

### Data Set – A2

A screenshot of a computer

Description automatically generated

### Cube in Visual Studio 2022

A screenshot of a computer

Description automatically generated

### Cube in SQL Server Management Studio

A screenshot of a computer

Description automatically generated