Appendix 3: Decision Support Views, Summary Tables, and Data Understanding

Data Understanding Approach

|  |
| --- |
| **[dbo].[T\_Agent\_Sales]** |
| ORDER\_ID (int, null) |
| AGENT\_ID (smallint, null) |

|  |
| --- |
| **[dbo].[T\_Categories]** |
| PROD\_CATEGORY\_ID (int, not null) |
| PROD\_CATEGORY (varchar(50), not null) |
| PROD\_CATEGORY\_DESC (varchar(2000), not null) |

* PROD\_CATEGORY has low cardinality (Peripherals and Accessories, Photo, ELECTRONICS, Hardware, Electronics, Software/Other)
* PROD\_CATEGORY & PROD\_CATEGORY\_DESC are the same with the exception of ELECTRONICS which has an incorrect description of photo.

|  |
| --- |
| **[dbo].[T\_Channels]** |
| CHANNEL\_ID (int, not null) |
| CHANNEL\_DESC (varchar(20), not null) |
| CHANNEL\_CLASS (varchar(20), null) |
| CHANNEL\_TOTAL (varchar(13), null) |

* CHANNEL\_ID has random values (3, 9, 5, 4, 2)
* CHANNEL\_CLASS has low cardinality (Direct, Indirect, Others)
* CHANNEL\_TOTAL only has one value. (Channel total)

|  |
| --- |
| **[dbo].[T\_Countries]** |
| COUNTRY\_ID (int, not null) |
| COUNTRY\_NAME (varchar(50), null) |
| COUNTRY\_SUBREGION (varchar(30), null) |
| COUNTRY\_REGION (varchar(20), null) |
| COUNTRY\_TOTAL (varchar(11), null) |

* COUNTRY\_REGION has low cardinality (Americas, Europe, Africa, Asia, Oceania)
* COUNTRY\_TOTAL only has one value. (World total)

|  |
| --- |
| **[dbo].[T\_Customers\_Int]** |
| CUST\_ID (int, not null) |
| CUST\_NAME (varchar(60), not null) |
| CUST\_GENDER (char(1), null) |
| CUST\_MAIN\_PHONE\_NUMBER (varchar(25), null) |
| CUST\_EMAIL (varchar(30), null) |
| CUST\_STREET\_ADDRESS (varchar(40), not null) |
| CUST\_POSTAL\_CODE (int, not null) |
| CUST\_CITY (varchar(30), not null) |
| CUST\_STATE\_PROVINCE (varchar(40), null) |
| COUNTRY\_ID (char(5), null) |
| CUST\_TOTAL (varchar(14), null) |

* Duplicate customers with different ID numbers
* CUST\_GENDER has low cardinality. (F, M)
* CUST\_TOTAL only has one value. (Customer total)

|  |
| --- |
| **[dbo].[T\_Customers\_Ext]** |
| CUST\_ID (int, not null) |
| CUST\_YEAR\_OF\_BIRTH (smallint, null) |
| CUST\_EDUCATION (int, null) |
| CUST\_MARITAL\_STATUS (varchar(20), null) |
| NO\_OF\_CHILDREN (int, null) |
| RACE (int, null) |
| INCOME (float, null) |
| CHECKING\_BAL (int, null) |
| ASSETS (int, null) |
| HOUSES (int, null) |
| STOCKS (int, null) |
| BONDS (int, null) |
| HOME\_EQUITY (int, null) |
| EQUITY (int, null) |
| NETWORTH (int, null) |

* CUST\_EDUCATION has low cardinality. (1, 2, 3, 4)
* CUST\_MARITAL\_STATUS has low cardinality. (married, single)
* RACE has low cardinality. (1, 2)
* INCOME should be two decimals

|  |
| --- |
| **[dbo].[T\_Order\_Items\_12\_13]** |
| SALESTRANS\_ID (int, not null) |
| ORDER\_ID (int, null) |
| PROD\_ID (int, null) |
| SHIPPING\_DATE (datetime, null) |
| QUANTITY\_SOLD (int, null) |
| AMOUNT\_SOLD (decimal(8,2), null) |
| UNIT\_PRICE (decimal(8,2), null) |

|  |
| --- |
| **[dbo].[T\_Order\_Items\_14]** |
| SALESTRANS\_ID (int, not null) |
| ORDER\_ID (int, null) |
| PROD\_ID (int, null) |
| SHIPPING\_DATE (datetime, null) |
| QUANTITY\_SOLD (int, null) |
| AMOUNT\_SOLD (decimal(8,2), null) |
| UNIT\_PRICE (decimal(8,2), null) |

|  |
| --- |
| **[dbo].[T\_Orders\_12\_13]** |
| ORDER\_ID (int, not null) |
| CUST\_ID (int, null) |
| CHANNEL\_ID (int, null) |
| PROMO\_ID (int, null) |
| SALE\_DATE (datetime, null) |
| PAYMENT\_DATE (datetime, null) |

|  |
| --- |
| **[dbo].[T\_Orders\_14]** |
| ORDER\_ID (int, not null) |
| CUST\_ID (int, null) |
| CHANNEL\_ID (int, null) |
| PROMO\_ID (int, null) |
| SALE\_DATE (datetime, null) |
| PAYMENT\_DATE (datetime, null) |

|  |
| --- |
| **[dbo].[T\_Products]** |
| PROD\_ID (int, not null) |
| PROD\_NAME (varchar(50), not null) |
| PROD\_DESC (varchar(4000), not null) |
| PROD\_WEIGHT\_CLASS (smallint, null) |
| PROD\_UNIT\_OF\_MEASURE (char(1), null) |
| PROD\_PACK\_SIZE (char(1), null) |
| SUPPLIER\_ID (int, null) |
| PROD\_STATUS (varchar(20), not null) |
| PROD\_LIST\_PRICE (decimal(8,2), not null) |
| PROD\_MIN\_PRICE (decimal(8,2), not null) |
| PROD\_TOTAL (varchar(13), null) |
| PROD\_SUBCATEGORY\_ID (smallint, not null) |

* PROD\_NAME & PROD\_DESC are the same values
* PROD\_WEIGHT\_CLASS has low cardinality. (1, 4)
* PROD\_UNIT\_OF\_MEASURE only has one value. (U)
* PROD\_PACK\_SIZE only has one value. (P)
* SUPPLIER\_ID only has one value. (1)
* PROD\_STATUS only has one value. (STATUS)
* PROD\_TOTAL only has one value. (TOTAL)

|  |
| --- |
| **[dbo].[T\_Promotions]** |
| PROMO\_ID (int, not null) |
| PROMO\_NAME (varchar(30), not null) |
| PROMO\_SUBCATEGORY (varchar(30), not null) |
| PROMO\_CATEGORY (varchar(30), not null) |
| PROMO\_COST (decimal(8,2), not null) |
| PROMO\_BEGIN\_DATE (datetime, not null) |
| PROMO\_END\_DATE (datetime, not null) |
| PROMO\_TOTAL (varchar(15), null) |

* PROMO\_TOTAL only has one value. (Promotion total)

|  |
| --- |
| **[dbo].[T\_Subcategories]** |
| PROD\_SUBCATEGORY\_ID (int, not null) |
| PROD\_SUBCATEGORY (varchar(50), not null) |
| PROD\_SUBCATEGORY\_DESC (varchar(2000), not null) |
| PROD\_CATEGORY\_ID (int, not null) |

PROD\_CATEGORY & PROD\_SUBCAT\_DESC is the same value.

Decision Support Views:

1. Provide SQL Scripts for creating VIEWS & Summary tables

|  |  |
| --- | --- |
| Decision Support Views | Queries |
| --- Yearly promotional analysis --- | CREATE VIEW YEARLY\_PROMOTIONAL\_ANALYSIS AS  SELECT P.PROMO\_ID, P.PROMO\_NAME, P.PROMO\_CATEGORY, P.PROMO\_SUBCATEGORY, P.PROMO\_BEGIN\_DATE, P.PROMO\_END\_DATE, P.PROMO\_COST, SUM(F.AMOUNT\_SOLD) AS TOTAL\_SALES  FROM GROUP1\_DWH.FACT\_ORDERS F  INNER JOIN GROUP1\_DWH.DIM\_PROMOTIONS P  ON F.PROMO\_KEY = P.PROMO\_KEY  GROUP BY P.PROMO\_ID, P.PROMO\_NAME, P.PROMO\_CATEGORY, P.PROMO\_SUBCATEGORY, P.PROMO\_BEGIN\_DATE, P.PROMO\_END\_DATE, P.PROMO\_COST  ------------------ |
| --- Post-promotional event analysis --- | CREATE VIEW POST\_PROMOTIONAL\_ANALYSIS AS  SELECT P.PROMO\_ID, P.PROMO\_NAME, SALESTRANS\_ID, PAYMENT\_DATE\_KEY, SALE\_DATE\_KEY, QUANTITY\_SOLD, AMOUNT\_SOLD  FROM GROUP1\_DWH.FACT\_ORDERS F  INNER JOIN GROUP1\_DWH.DIM\_PROMOTIONS P  ON F.PROMO\_KEY = P.PROMO\_KEY  ------------------ |
| --- Customer retention data --- | CREATE VIEW CUSTOMER\_RETENTION\_DATA AS  SELECT C.CUST\_ID, C.CUST\_NAME, C.CUST\_GENDER, C.CUST\_MAIN\_PHONE\_NUMBER, C.CUST\_EMAIL, C.CUST\_STREET\_ADDRESS, C.CUST\_CITY,  C.CUST\_STATE\_PROVINCE, C.CUST\_POSTAL\_CODE, C.CUST\_MARITAL\_STATUS, C. CUST\_NO\_OF\_CHILDREN, SUM(F.AMOUNT\_SOLD) AS TOTAL\_SALES, MAX(F.SALE\_DATE\_KEY) AS LAST\_PURCHASE  FROM GROUP1\_DWH.DIM\_CUSTOMERS C  JOIN GROUP1\_DWH.FACT\_ORDERS F  ON C.CUST\_KEY = F.CUST\_KEY  GROUP BY C.CUST\_ID, C.CUST\_NAME, C.CUST\_GENDER, C.CUST\_MAIN\_PHONE\_NUMBER, C.CUST\_EMAIL, C.CUST\_STREET\_ADDRESS, C.CUST\_CITY,  C.CUST\_STATE\_PROVINCE, C.CUST\_POSTAL\_CODE, C.CUST\_MARITAL\_STATUS, C. CUST\_NO\_OF\_CHILDREN |
| --- Shipping Turnaround Time --- | CREATE VIEW QUALITY\_ASSURANCE\_AUDIT AS  SELECT F.SALESTRANS\_ID, F.SALE\_DATE\_KEY, F.PAYMENT\_DATE\_KEY, F.SHIPPING\_DATE\_KEY, F.AMOUNT\_SOLD, F.QUANTITY\_SOLD, DATEDIFF(DAY, F.SALE\_DATE\_KEY, F.SHIPPING\_DATE\_KEY) AS DAYS\_TO\_SHIP, DATEDIFF(DAY, F.SALE\_DATE\_KEY, F.PAYMENT\_DATE\_KEY) AS DAYS\_TO\_PAY  FROM GROUP1\_DWH.FACT\_ORDERS F  WHERE DATEDIFF(DAY, F.SALE\_DATE\_KEY, F.SHIPPING\_DATE\_KEY) > 5  AND DATEDIFF(DAY, F.SALE\_DATE\_KEY, F.PAYMENT\_DATE\_KEY) > 3 |
| --- Sales per agent, agent experience, sale Transaction, sale Amount----- | CREATE VIEW AGENT\_PERFORMANCE\_ AS  SELECT D.YEAR4, D.YEAR\_MONTH\_NUMBER, A.AGENT\_EXPERIENCE\_LEVEL, SUM(O.AMOUNT\_SOLD) AS TOTAL\_AMOUNT\_SOLD, SUM(O.QUANTITY\_SOLD) AS TOTAL\_QUANTITY\_SOLD  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_AGENT A  ON O.AGENT\_KEY = A.AGENT\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.PAYMENT\_DATE\_KEY = D.DATE\_KEY  GROUP BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, A.AGENT\_EXPERIENCE\_LEVEL  GO |
| --- Sales per product category, product sub category, product name, sale transaction, sale amount --- | CREATE VIEW PROD\_SALES\_VIEW AS    SELECT D.YEAR4, D.YEAR\_MONTH\_NUMBER, P.PROD\_CATEGORY, P.PROD\_SUBCATEGORY, P.PROD\_NAME, SUM(O.AMOUNT\_SOLD) AS TOTAL\_SALES, SUM(O.QUANTITY\_SOLD) AS TOTAL\_SOLD  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_PRODUCTS P  ON O.PROD\_KEY = P.PROD\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.PAYMENT\_DATE\_KEY = D.DATE\_KEY  GROUP BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, P.PROD\_CATEGORY, P.PROD\_SUBCATEGORY, P.PROD\_NAME    GO |
| --- Sales per customer region, customer sub region, customer name, sale transaction, sale amount | CREATE VIEW CUSTOMER\_SALES\_VIEW AS  SELECT D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME, SUM(O.AMOUNT\_SOLD) AS TOTAL\_SALES, SUM(O.QUANTITY\_SOLD) AS TOTAL\_SOLD  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_CUSTOMERS C  ON O.CUST\_KEY = C.CUST\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.PAYMENT\_DATE\_KEY = D.DATE\_KEY  GROUP BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME  GO |
| --- Sales per channel class, channel, sale transaction, sale amount | CREATE VIEW CHANNEL\_v AS  SELECT D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CHANNEL\_CLASS, SUM(O.AMOUNT\_SOLD) AS TOTAL\_SALES, SUM(O.QUANTITY\_SOLD) AS TOTAL\_SOLD  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_CHANNELS C  ON O.CHANNEL\_KEY = C.CHANNEL\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.PAYMENT\_DATE\_KEY = D.DATE\_KEY  GROUP BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CHANNEL\_CLASS  GO |
| -- Promotional Detail View | CREATE VIEW PROMOTION\_DETAIL\_VIEW AS  SELECT F.[SALESTRANS\_ID]  , PM.[PROMO\_NAME]  , P.[PROD\_NAME]  , F.[AMOUNT\_SOLD]  , F.[QUANTITY\_SOLD]  , CAST (F.[AMOUNT\_SOLD]/ F.[QUANTITY\_SOLD] AS DECIMAL (8,2)) AS UNIT\_PRICE  , P.[PROD\_LIST\_PRICE]  , CAST (P.[PROD\_LIST\_PRICE] - F.[AMOUNT\_SOLD]/F.[QUANTITY\_SOLD] AS DECIMAL (8,2)) AS DISCOUNT\_RATE  FROM [GROUP1\_DWH].[FACT\_ORDERS] F JOIN [GROUP1\_DWH].[DIM\_PRODUCTS] P  ON F.[PROD\_KEY] = P.[PROD\_KEY]  JOIN [GROUP1\_DWH].[DIM\_DATE] D  ON F.[SALE\_DATE\_KEY] = D.[DATE\_KEY]  JOIN [GROUP1\_DWH].[DIM\_PROMOTIONS] PM  ON F.[PROMO\_KEY] = PM.[PROMO\_KEY]  WHERE D.[DATE\_VALUE] >= PM.[PROMO\_BEGIN\_DATE] AND D.[DATE\_VALUE] <= PM.[PROMO\_BEGIN\_DATE]  GO |
| -- Promotion Summary | CREATE VIEW PROMOTION\_SUMMARY\_VIEW AS  SELECT [PROMO\_NAME], SUM([AMOUNT\_SOLD]) AS TOTAL\_AMOUNT\_SOLD, SUM([QUANTITY\_SOLD]) AS TOTAL\_QUANTITY\_SOLD  FROM PROMOTION\_DETAIL\_VIEW  GROUP BY [PROMO\_NAME]  GO |
| GROUP1\_DWH.CUSTOMER\_SALES\_SUMMARY | SELECT \* INTO GROUP1\_DWH.CUSTOMER\_SALES\_SUMMARY FROM dbo.CUSTOMER\_SALES\_VIEW  SELECT D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME, SUM(O.AMOUNT\_SOLD) AMOUNT\_SOLD, SUM(O.QUANTITY\_SOLD) QUATITY\_SOLD  INTO GROUP1\_DWH.CUSTOMER\_PAYMENT\_SUMMARY  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_CUSTOMERS C  ON O.CUST\_KEY = C.CUST\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.PAYMENT\_DATE\_KEY = D.DATE\_KEY  GROUP BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME  ORDER BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME    GO |
| GROUP1\_DW.CUSTOMER\_PAYMENT\_SUMMARY | SELECT D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME, SUM(O.AMOUNT\_SOLD) AMOUT\_SOLD, SUM(O.QUANTITY\_SOLD) QUATITY\_SOLD  INTO GROUP1\_DW.CUSTOMER\_PAYMENT\_SUMMARY  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_CUSTOMERS C  ON O.CUST\_KEY = C.CUST\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.PAYMENT\_DATE\_KEY = D.DATE\_KEY  GROUP BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME  ORDER BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME  GO |