

DBT ASSIGNMENT-5

ASHWIN KRISHNA P

PES1201801465

SEM: 5 , SECTION : F

DESIGN,IMPLEMENTATION AND ANALYSIS OF MULTI-DIMENSIONAL DATABASE MODEL

The multidimensional model involves two types of tables:

Dimension tables:

A dimension table consists of tuples of attributes of the dimension.

Fact tables:

A fact table can be thought of as having tuples, one per a recorded fact. This fact contains some measured or observed variables and identifies them with pointers to dimension tables.

Here We are designing the database for ABC MOBILE SERVICES.

So I created the database called [ABCMOBILES]. Initially I designed relational model and then multidimensional model by creating Dimension tables and Fact tables.

Creating Dimensions and Fact tables:

```
CREATE TABLE DIM_CALL_ORIGIN_TERMINATION
(DIM_C_O_T_KEY NUMERIC(10) NOT NULL PRIMARY KEY,
TOWER_ID NUMERIC(10), CITY_CODE NUMERIC(10), COUNTRY_CODE NUMERIC(10), LOCATION VARCHAR(30), CITY VARCHAR(20), COUNTRY VARCHAR(20))

CREATE TABLE DIM_PLAN_COST
(
DIM_PLAN_COST_KEY NUMERIC(10) NOT NULL PRIMARY KEY,
PLAN_ID NUMERIC(10),
PLAN_NAME VARCHAR(20),
PLAN_TYPE VARCHAR(20),
MONTHLY_CHARGE NUMERIC(5),
ROAMING_CHARGE NUMERIC(5),
FREE_CALLS INT,
FREE_SMS INT,
)

CREATE TABLE DIM_CALL_DATE
(
DIM_CALL_DATE_KEY INT NOT NULL PRIMARY KEY,
CALL_MONTH VARCHAR(10),
CALL_YEAR INT,
BILLING_START_DATE DATETIME,
BILLING_END_DATE DATETIME
)
```

```

DIM_CALL_DATE_KEY INT NOT NULL PRIMARY KEY,
CALL_MONTH VARCHAR(10),
CALL_YEAR INT,
BILLING_START_DATE DATETIME,
BILLING_END_DATE DATETIME
)

CREATE TABLE DIM_CUSTOMER
(
DIM_CUSTOMER_KEY NUMERIC(10) NOT NULL PRIMARY KEY,
PHONE_NO NUMERIC(12),
ACCOUNT_ID INT,
ACCOUNT_NAME VARCHAR(20),
ACCOUNT_TYPE VARCHAR(20),
CUST_NAME VARCHAR(20),
CUST_ADDRESS VARCHAR(100),
CUST_CITY VARCHAR(20),
CUST_STATE VARCHAR(20),
CUST_COUNTRY VARCHAR(20),
)

CREATE TABLE DIM_DEVICE_DETAILS
(
DIM_DEVICE_KEY INT NOT NULL PRIMARY KEY,
DEVICE_DESCRIPTION VARCHAR(50),
MANUFACTURE_COMPANY VARCHAR(20),
MANUFACTURE_DATE DATE,
MODEL_NUMBER INT,
MODEL_NAME VARCHAR(20)
)

```

Fact table:

```

CREATE TABLE FACT_CALLS
(
DIM_C_O_T_KEY NUMERIC(10) NOT NULL REFERENCES DIM_CALL_ORIGIN_TERMINATION(DIM_C_O_T_KEY),
DIM_PLAN_COST_KEY NUMERIC(10) NOT NULL REFERENCES DIM_PLAN_COST(DIM_PLAN_COST_KEY),
DIM_CALL_DATE_KEY INT NOT NULL REFERENCES DIM_CALL_DATE(DIM_CALL_DATE_KEY),
DIM_CUSTOMER_KEY NUMERIC(10) NOT NULL REFERENCES DIM_CUSTOMER(DIM_CUSTOMER_KEY),
DIM_DEVICE_KEY INT NOT NULL REFERENCES DIM_DEVICE_DETAILS(DIM_DEVICE_KEY),
LOCAL_CALLS INT,
LOCAL_CALLS_BILLABLE_DURATION NUMERIC(10),
LOCAL_CHARGES NUMERIC(10),
STD_CALLS INT,
STD_CALLS_BILLABLE_DURATION NUMERIC(10),
STD_CHARGES NUMERIC(10),
ISD_CALLS INT,
ISD_CALLS_BILLABLE_DURATION NUMERIC(10),
ISD_CHARGES NUMERIC(10),
SMS_CHARGES NUMERIC(10),
DISCOUNT NUMERIC(10),
NET_CHARGES NUMERIC(10),
CONSTRAINT [PK_FACT_CALLS] PRIMARY KEY([DIM_C_O_T_KEY],[DIM_PLAN_COST_KEY],
[DIM_CALL_DATE_KEY],[DIM_CUSTOMER_KEY],[DIM_DEVICE_KEY])
)

```

After creating the tables I insert values into it.

After inserting a few values, the tables look like:

```
select * from DIM_CALL_ORIGIN_TERMINATION
SELECT * FROM DIM_PLAN_COST
SELECT * FROM DIM_CALL_DATE
select * from DIM_CUSTOMER
```

DIM_C_O_T_KEY	TOWER_ID	CITY_CODE	COUNTRY_CODE	LOCATION	CITY	COUNTRY
1001	2001	560018	91	CHAMARAJAPETE	BANGALORE	INDIA
1002	2002	560019	91	CHIKPETE	BANGALORE	INDIA
1003	2003	560028	91	WHITEFIELD	BANGALORE	INDIA
1004	2004	560038	91	KR PURAM	BANGALORE	INDIA
1005	2005	560085	91	AVALAHALLI	BANGALORE	INDIA
1006	2006	560095	91	MAJESTIC	BANGALORE	INDIA

DIM_PLAN_COST_KEY	PLAN_ID	PLAN_NAME	PLAN_TYPE	MONTHLY_CHARGE	ROAMING_CHARGE	FREE_CALLS	FREE_SMS
101	10	MITHRA	PREPAID	399	0	100	150
102	12	NEW1	PREPAID	199	15	50	50
103	13	NEW2	PREPAID	299	15	60	120
104	14	NEW3	PREPAID	349	25	75	170
105	15	NEW4	POSTPAID	499	0	75	250
106	16	WFH	PREPAID	250	25	75	110

DIM_CALL_DATE_KEY	CALL_MONTH	CALL_YEAR	BILLING_START_DATE	BILLING_END_DATE
20200101	JAN	2020	2020-01-01 23:23:40.000	2020-02-01 23:23:40.000
20200201	FEB	2020	2020-02-01 22:22:44.000	2020-03-01 23:59:59.000
20200301	MAR	2020	2020-03-01 10:23:20.000	2020-04-01 23:59:59.000
20200401	APR	2020	2020-04-01 12:23:50.000	2020-05-01 23:59:59.000
20200501	MAY	2020	2020-05-01 14:30:48.000	2020-06-01 23:59:59.000
20200601	JUN	2020	2020-06-01 22:30:59.000	2020-07-01 23:59:59.000

DIM_CUSTOMER_KEY	PHONE_NO	ACCOUNT_ID	ACCOUNT_NAME	ACCOUNT_TYPE	CUST_NAME	CUST_ADDRESS	CUST_CITY	CUST_STATE	CUST_COUNTRY
30001	9487878785	40001	ABC	PERMANENT	abc1	1ST MAIN,1ST CROSS,ABC STREET	BANGALORE	KARNATAKA	INDIA
30002	9487878786	40002	BCD	PERMANENT	bcd1	2ST MAIN,2ST CROSS,BCD STREET	BANGALORE	KARNATAKA	INDIA
30003	9487878787	40003	CDE	PERMANENT	cde1	3RD MAIN,3RD CROSS,CDE STREET	BANGALORE	KARNATAKA	INDIA
30004	9487878788	40004	DEF	PERMANENT	def1	4TH MAIN,4TH CROSS,DEF STREET	BANGALORE	KARNATAKA	INDIA
30005	9487878789	40005	EFG	TEMP	efg1	5TH MAIN,5TH CROSS,EFG STREET	BANGALORE	KARNATAKA	INDIA
30006	9487878790	40006	FGH	PERMANENT	fgh1	6TH MAIN,6TH CROSS,FGH STREET	BANGALORE	KARNATAKA	INDIA

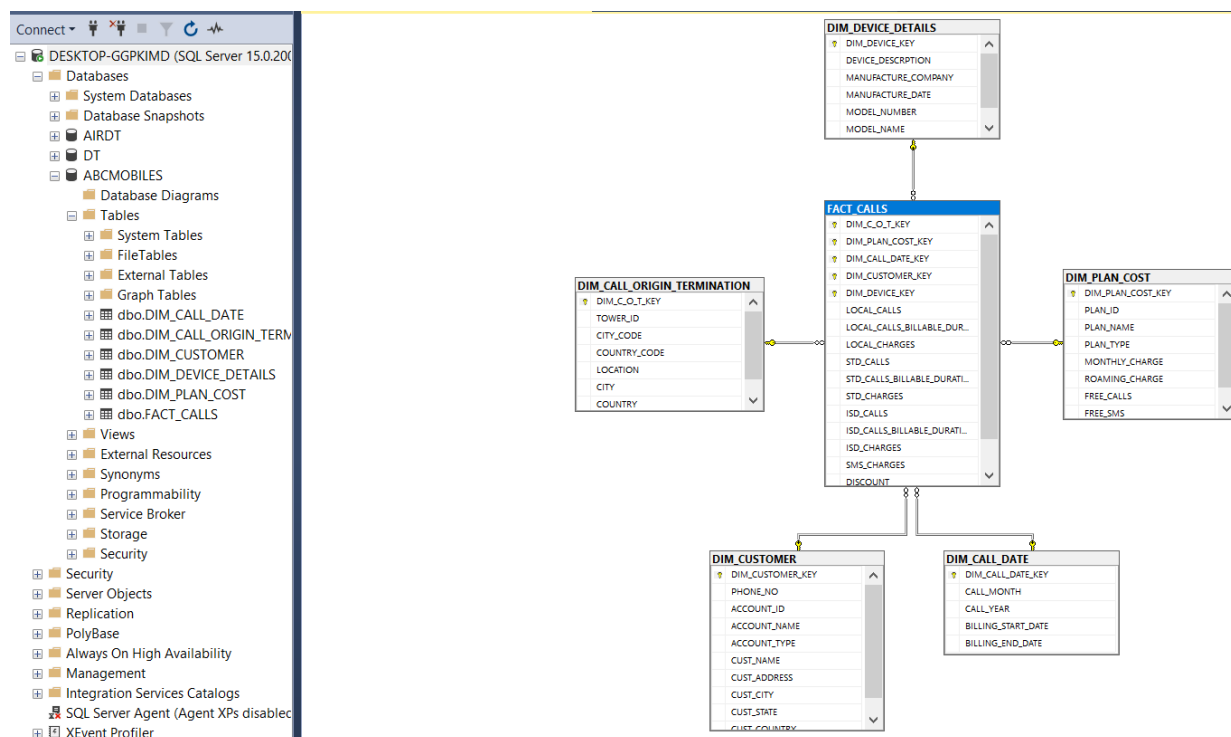
```
SELECT * FROM DIM_DEVICE_DETAILS
select * from FACT_CALLS
```

DIM_DEVICE_KEY	DEVICE_DESCRIPTION	MANUFACTURE_COMPANY	MANUFACTURE_DATE	MODEL_NUMBER	MODEL_NAME
801	SMARTPHONE.ANDROID	SAMSUNG	2018-12-15	21	GALAXY A1
802	SMARTPHONE.ANDROID	SAMSUNG	2020-10-15	55	GALAXY M21
803	ROUTER.JIO-FI	JIO	2019-12-15	34	VERSION-3
804	SMARTPHONE.IOS	APPLE	2018-11-15	28	IPHONE-X
805	SMARTPHONE.IOS	APPLE	2019-06-14	48	IPHONE SE
806	SMARTPHONE.IOS	APPLE	2019-07-13	52	IPHONE XS

DIM_C_O_T_KEY	DIM_PLAN_COST_KEY	DIM_CALL_DATE_KEY	DIM_CUSTOMER_KEY	DIM_DEVICE_KEY	LOCAL_CALLS	LOCAL_CALLS_BILLABLE_DURATION	LOCAL_CHARGES	STD_CALLS	STD_CALLS_BILLABLE_DURATION	STD_CHARGE
1001	101	20200101	30001	801	500	650	700	100	150	200
1002	102	20200201	30002	802	510	660	710	110	151	210
1003	103	20200301	30003	803	520	670	720	120	152	220
1004	104	20200401	30004	804	530	680	730	130	153	230
1005	105	20200501	30005	805	540	690	740	140	154	240
1006	106	20200601	30006	806	550	695	750	150	155	250

While designing the multi-dimensional model we can model it as a star schema or a snowflake schema. We prefer star schema because snowflake model makes the join operation expensive.

Star Schema diagram of the multi-dimensional model:



We can also notice all the tables and objects that are created under ABCMOBILES database, in object explorer on the left side.

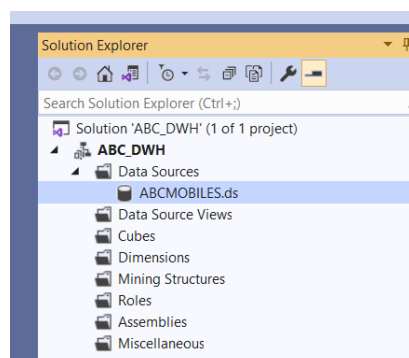
Now I use Visual Studio to convert this star schema into a cube and then proceed.

SQL SERVER DATA TOOLS and Microsoft Analysis Project Service tools in visual studio allows us to do many operations and analysis regarding a multidimensional data model.

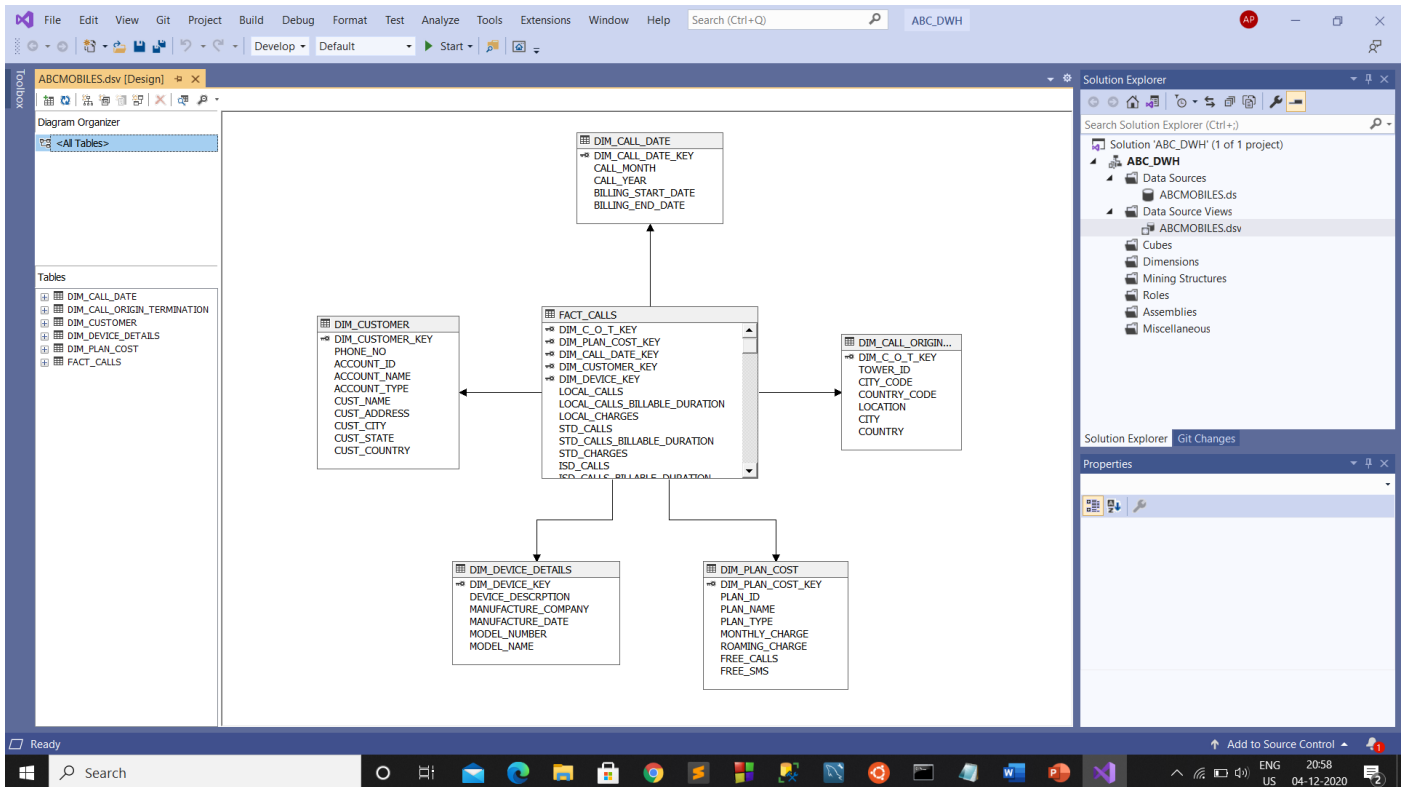
Initially we use the database ABCMOBILES that we created on SQL Server as the data source, and we create data source view for it.

ABC_DWH is the analysis project that I created:

ABCMOBILES is the data source.



Data source view:

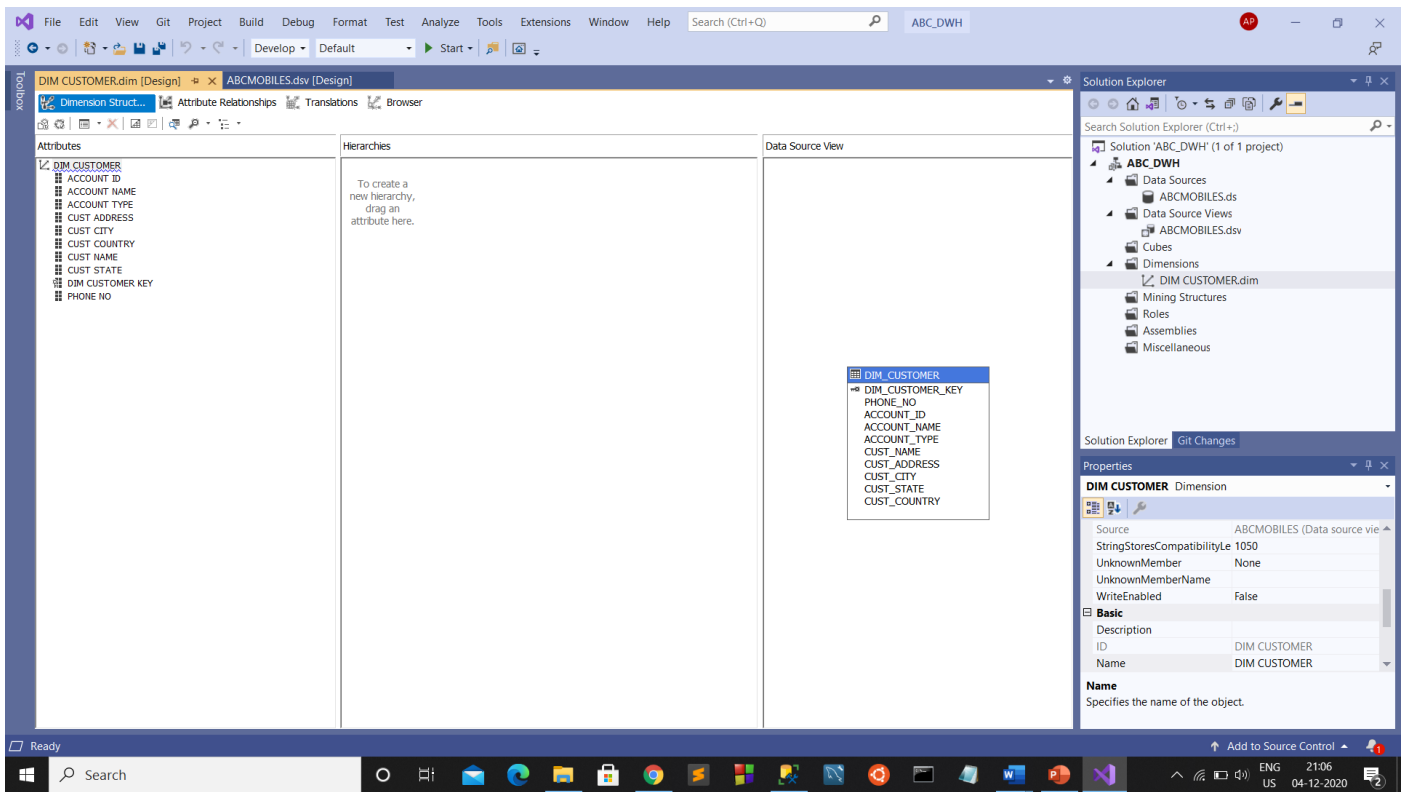


Then I created different dimensions as designed in the model.

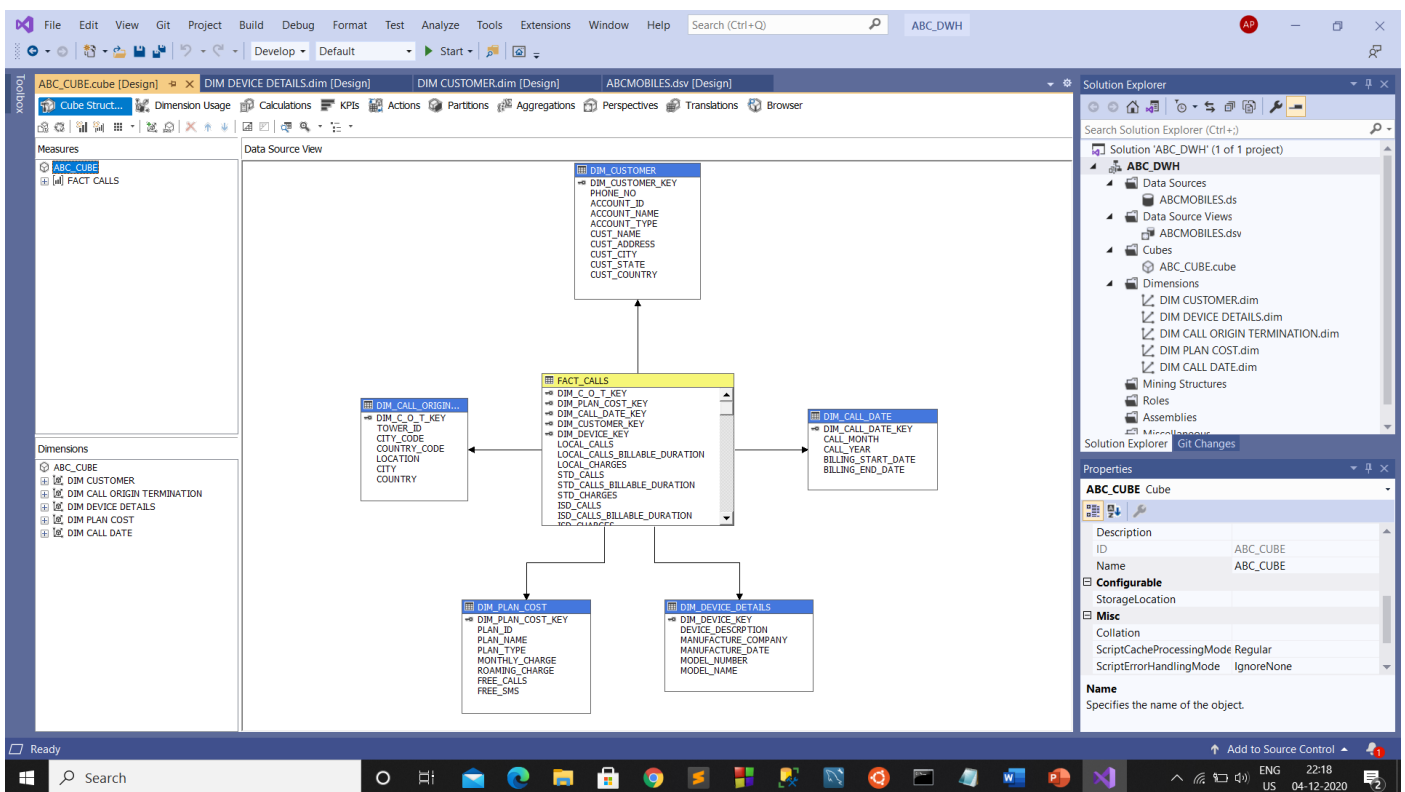
Creating dimension for CUSTOMER:

The screenshot shows the Dimension Wizard dialog box in SSDT. The 'Specify Source Information' tab is active, showing the data source view 'ABCMOBILES', the main table 'DIM_CUSTOMER', and the key columns 'DIM_CUSTOMER_KEY' and 'DIM_CUSTOMER_KEY'. The 'Select Dimension Attributes' tab is also visible, showing a list of attributes with checkboxes for 'Attribute Name', 'Enable Browsing', and 'Attribute Type'. The 'Attribute Name' column lists: DIM CUSTOMER KEY, PHONE NO, ACCOUNT ID, ACCOUNT NAME, ACCOUNT TYPE, CUST NAME, CUST ADDRESS, CUST CITY, CUST STATE, and CUST COUNTRY. The 'Enable Browsing' column has checkboxes for all attributes. The 'Attribute Type' column lists: Regular, Regular, Regular, Regular, Regular, Regular, Regular, Regular, Regular, and Regular. The 'Next >' button is highlighted.

Attribute Name	Enable Browsing	Attribute Type
<input checked="" type="checkbox"/> DIM CUSTOMER KEY	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> PHONE NO	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> ACCOUNT ID	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> ACCOUNT NAME	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> ACCOUNT TYPE	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> CUST NAME	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> CUST ADDRESS	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> CUST CITY	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> CUST STATE	<input checked="" type="checkbox"/>	Regular
<input checked="" type="checkbox"/> CUST COUNTRY	<input checked="" type="checkbox"/>	Regular



Create all dimensions and Using these dimensions we can build the cube. We build the cube:



Build the cube:

The screenshot shows the SSDT interface with the 'ABC_CUBE.cube [Design]' tab active. The 'Measures' pane on the left lists 'ABC_CUBE' and 'FACT CALLS'. The 'Dimensions' pane lists 'ABC_CUBE', 'DIM_CUSTOMER', 'DIM CALL ORIGIN TERMINATION', 'DIM DEVICE DETAILS', 'DIM PLAN COST', and 'DIM CALL DATE'. The 'Data Source View' shows a central fact table 'FACT_CALLS' connected to several dimension tables: 'DIM_CUSTOMER', 'DIM_DEVICE_DETAILS', 'DIM_CALL_DATE', and 'DIM_CALL_ORIGIN'. The 'Output' pane displays the build process logs, indicating successful completion. The 'Solution Explorer' on the right shows the project structure for 'ABC_DWH'.

```
Build
-----
Build started: Project: ABC_DWH, Configuration: Development
Started Building Analysis Services project: Incremental ....
Dimension [DIM_CUSTOMER] : Create hierarchies in non-parent child dimensions.
Dimension [DIM_DEVICE DETAILS] : Create hierarchies in non-parent child dimensions.
Database [ABC_DWH] : The database has no Time dimension. Consider creating one.
Build complete -- 0 errors, 3 warnings
-----
Build: 1 succeeded or up-to-date, 0 failed, 0 skipped -----
```

Deploy the cube:

The screenshot shows the SSDT interface with the 'ABC_DWH' project. The 'Deployment Progress' pane on the left shows the deployment status, including a green checkmark indicating successful completion. The 'Output' pane displays the deployment process logs, including sending the deployment script to the server. The 'Solution Explorer' on the right shows the project structure for 'ABC_DWH'.

```
Deployment Progress - ABC_DWH
-----
Server: localhost
Database: ABC_DWH
Processing Cube 'ABC_CUBE' completed.
Start time: 04-12-2020 23:53:57; End time: 04-12-2020 23:54:00

Status:
-----
Deployment Completed Successfully

Dimensions
-----
ABC_CUBE
DIM_CUSTOMER
DIM CALL ORIGIN TERMINATION
DIM DEVICE DETAILS
DIM PLAN COST
DIM CALL DATE

Output
-----
Build
-----
Build started: Project: ABC_DWH, Configuration: Development
Started Building Analysis Services project: Incremental ....
Dimension [DIM_CUSTOMER] : Create hierarchies in non-parent child dimensions.
Dimension [DIM_DEVICE DETAILS] : Create hierarchies in non-parent child dimensions.
Database [ABC_DWH] : The database has no Time dimension. Consider creating one.
Build complete -- 0 errors, 3 warnings
-----
Deploy started: Project: ABC_DWH, Configuration: Development
Performing an incremental deployment of the 'ABC_DWH' database to the 'localhost' server.
Generating deployment script...
Add RelationalDataSource ABCMOBILES
Process Database ABC_DWH
Done
Sending deployment script to the server...
Done
Deploy complete -- 0 errors, 0 warnings
-----
Build: 1 succeeded or up-to-date, 0 failed, 0 skipped -----
Deploy: 1 succeeded, 0 failed, 0 skipped -----
```

After deploying the cube, we can view and query the cube using SQL Server Analysis Services:

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left displays the hierarchy: DESKTOP-GGPKIMD (Microsoft Analysis) > Databases > ABC_DWH > Data Sources > ABCMOBILES > Data Source Views > ABCMOBILES > Cubes > ABC_CUBE. The central pane shows the cube's metadata, including measures like DISCOUNT, FACT CALLS Count, ISD CALLS, ISD CALLS BILLABLE DURATION, ISD CHARGES, LOCAL CALLS, LOCAL CALLS BILLABLE DURATION, LOCAL CHARGES, NET CHARGES, SMS CHARGES, STD CALLS, STD CALLS BILLABLE DURATION, and STD CHARGES. The right pane shows a query: `--to project total ISD CALLS and ISD CHARGES--`
`SELECT {[Measures].[ISD CALLS],[Measures].[ISD CHARGES]} ON COLUMNS FROM ABC_CUBE`. The Results pane at the bottom shows the query output:

ISD CALLS	ISD CHARGES
75	161

The screenshot shows the Microsoft SQL Server Management Studio interface with a filtered query. The Object Explorer on the left shows the same hierarchy as the previous screenshot. The central pane shows the cube's metadata. The right pane shows a query: `SELECT {[Measures].[ISD CALLS],[Measures].[ISD CHARGES]} ON COLUMNS,`
`[DIM CUSTOMER].[PHONE NO] ON ROWS`
`FROM ABC_CUBE`
`WHERE [ACCOUNT ID].[40002]`. The Results pane at the bottom shows the query output:

ISD CALLS	ISD CHARGES
11	30

The status bar at the bottom indicates: Query executed successfully. DESKTOP-GGPKIMD DESKTOP-GGPKIMD\ashwi ABC_DWH 00:00:01

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left displays the database structure for 'DESKTOP-GGPKIMD (Microsoft Analysis)', including 'ABC_DWH', 'Data Sources', 'Data Source Views', 'Cubes', 'Dimensions', 'Mining Structures', 'Roles', 'Assemblies', and 'Management'. The 'ABC_CUBE' cube is selected, showing its measures and dimensions. The query window on the right contains the following SQL query:

```
--ISD CALLS, CHARGES AND PHONE NUMBER OF CUSTOMER WHOSE ACCOUNT ID=40002--
SELECT {[Measures].[ISD CALLS],[Measures].[ISD CHARGES]} ON COLUMNS,
[Dim Customer].[Phone No].MEMBERS ON ROWS
FROM ABC_CUBE
WHERE [ACCOUNT ID].[40002]
```

The Results pane shows the following data:

	ISD CALLS	ISD CHARGES
All	11	30
9487878786	11	30

The status bar at the bottom indicates 'Query executed successfully.' and the execution time is 00:00:01.

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left displays the database structure for 'DESKTOP-GGPKIMD (Microsoft Analysis)', including 'ABC_DWH', 'Data Sources', 'Data Source Views', 'Cubes', 'Dimensions', 'Mining Structures', 'Roles', 'Assemblies', and 'Management'. The 'ABC_CUBE' cube is selected, showing its measures and dimensions. The query window on the right contains the following SQL query:

```
--ISD CALLS, CHARGES AND NAME OF CUSTOMER WHOSE ACCOUNT ID=40004--
SELECT {[Measures].[ISD CALLS],[Measures].[ISD CHARGES]} ON COLUMNS,
[Dim Customer].[Cust Name].MEMBERS ON ROWS
FROM ABC_CUBE
WHERE [ACCOUNT ID].[40004]
```

The Results pane shows the following data:

	ISD CALLS	ISD CHARGES
All	13	25
def1	13	25

The status bar at the bottom indicates 'Query executed successfully.' and the execution time is 00:00:01.

Here we can query for any data or details that is in the cube and we can see and observe necessary actions to be taken on the members of different dimensions on the left side explorer window.

We can project data row wise and column wise according to the hierarchy.

Further we can do deep analysis, reporting and can get any result of the numeric values in the tables of the database and we can get results to many problems and calculations by using this data model in **Microsoft Excel** by the help of **pivot table**. We can specify the attributes which should be represented in rows and which objects should be in columns. Also we can perform complex mathematical operations and can apply filters so that we could get preferred output.

To print bill by specifying CALL DATE KEY:

The screenshot shows Microsoft Excel with a PivotTable named 'PivotTable1' and the 'PivotTable Fields' task pane. The PivotTable is set to show 'Row Labels' and 'Values'.

PivotTable Data:

Row Labels	ISD CALLS	ISD CALLS BILLABLE DURATION	ISD CHARGES	LOCAL CHARGES	NET CHARGES	SMS CHARGES
20200101	10		15	20	700	749
20200201	11		16	30	710	759
20200301	12		17	22	720	769
20200401	13		18	25	730	779
20200501	14		19	24	740	819
20200601	15		15	40	750	849
Grand Total	75	100	161	4350	4724	204

PivotTable Fields Task Pane:

- Choose fields to add to report:**
 - ☒ DIM CALL DATE
 - ☒ DIM CALL DATE KEY
 - ☐ DIM CALL ORIGIN TERMINATION
 - ☒ DIM CUSTOMER
 - ☐ ACCOUNT ID
 - ☐ ACCOUNT NAME
 - ☐ ACCOUNT TYPE
- Drag fields between areas below:**
 - Filters:** (Empty)
 - Columns:** (Empty)
 - Rows:** DIM CALL DATE KEY
 - Values:**
 - ISD CALLS
 - ISD CALLS BILLABLE...
 - ISD CHARGES
 - LOCAL CHARGES

To print bill by specifying CUSTMER NAME, ACCOUNT ID and CITY of users of any organization(a set of people) in rows and other charges/amounts in columns and grand total is also shown :

PivotTable Fields

Choose fields to add to report:

Search

☐ LOCAL CALLS BILLABLE DURATION
☒ **LOCAL CHARGES**
☒ **NET CHARGES**
☒ **SMS CHARGES**
☐ STD CALLS
☐ STD CALLS BILLABLE DURATION
☒ **STD CHARGES**

Drag fields between areas below:

Filters

Columns

Rows

Values

☐ Defer Layout Update Update

Row Labels	ISD CHARGES	LOCAL CHARGES	STD CHARGES	SMS CHARGES	NET CHARGES
abc1					
40001					
BANGALORE	20	700	200	30	749
bcd1					
40002					
BANGALORE	30	710	210	30	759
cde1					
40003					
BANGALORE	22	720	220	34	769
def1					
40004					
BANGALORE	25	730	230	30	779
efg1					
40005					
BANGALORE	24	740	240	50	819
fgh1					
40006					
BANGALORE	40	750	250	30	849
Grand Total	161	4350	1350	204	4724

To print bill by specifying CUST NAME,MODEL NAME of device, MANUFACTURE DATE of device and PHONE NO of users in rows and other charges/costs in columns and grand total:

PivotTable Fields

Choose fields to add to report:

Search

☒ **NET CHARGES**
☒ **SMS CHARGES**
☐ STD CALLS
☐ STD CALLS BILLABLE DURATION
☒ **STD CHARGES**
☒ DIM CALL DATE
☐ DIM CALL DATE KEY

Drag fields between areas below:

Filters

Columns

Rows

Values

☐ Defer Layout Update Update

Row Labels	SMS CHARGES	STD CHARGES	ISD CHARGES	LOCAL CHARGES	DISCOUNT	NET CHARGES
abc1						
GALAXY A1						
2018-12-15						
9487878785	30	200	20	700	50	749
bcd1						
GALAXY M21						
2020-10-15						
9487878786	30	210	30	710	50	759
cde1						
VERSION-3						
2019-12-15						
9487878787	34	220	22	720	50	769
def1						
IPHONE-X						
2018-11-15						
9487878788	30	230	25	730	50	779
efg1						
IPHONE SE						
2019-06-14						
9487878789	50	240	24	740	80	819
fgh1						
IPHONE XS						
2019-07-13						
9487878790	30	250	40	750	50	849
Grand Total	204	1350	161	4350	330	4724

THANK YOU