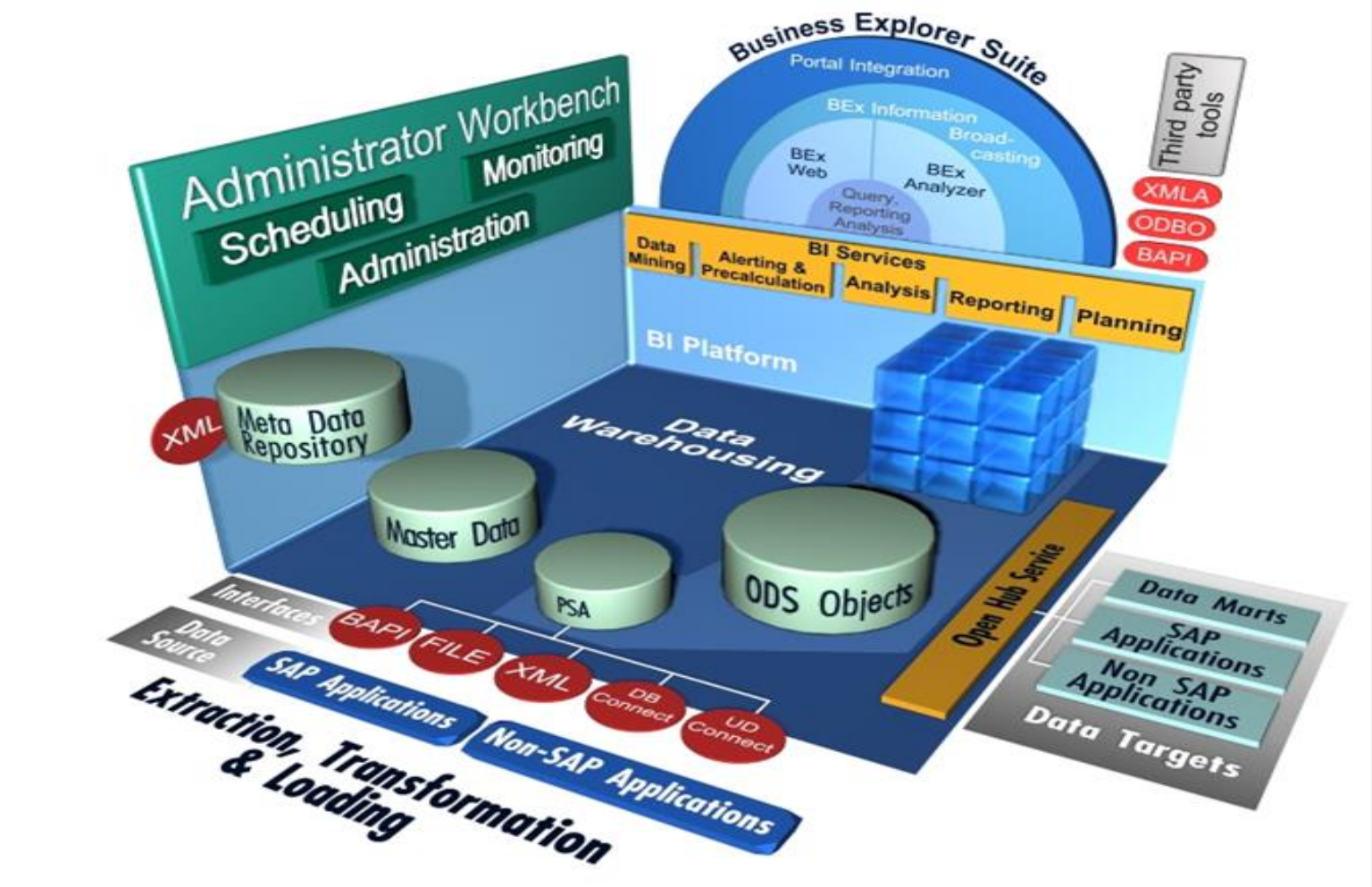
SAP BW

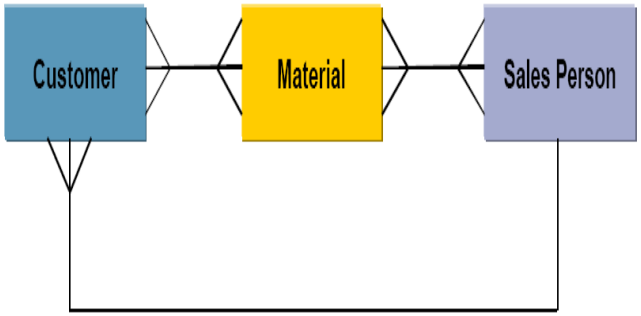
Lesson 03 : Modeling Part 2 - Infoproviders

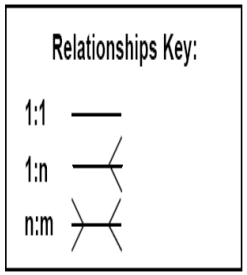
Capgemini

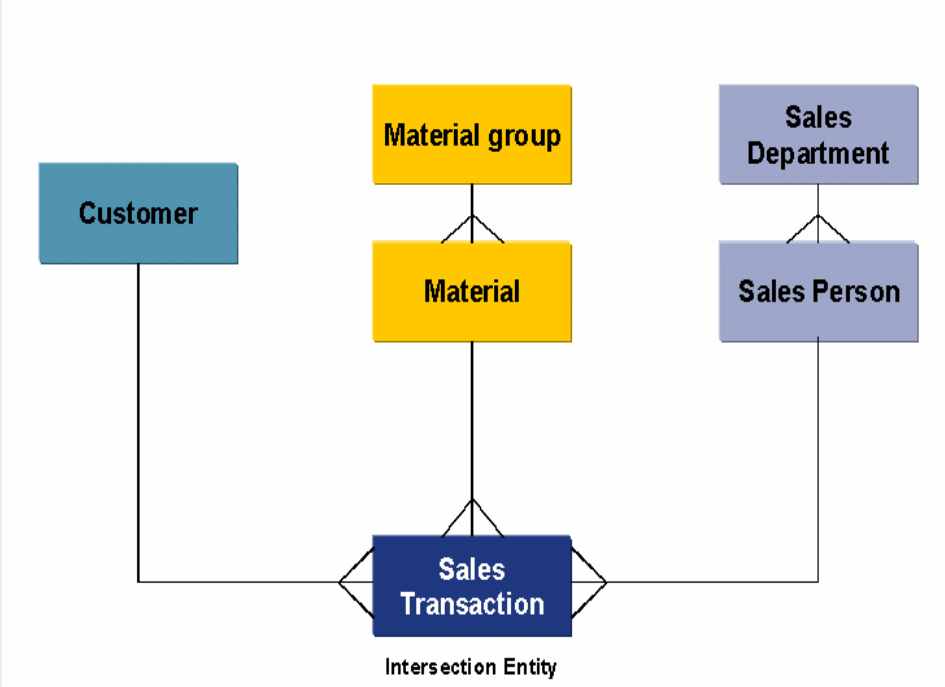
SAP BI Architecture



Review - ER Model

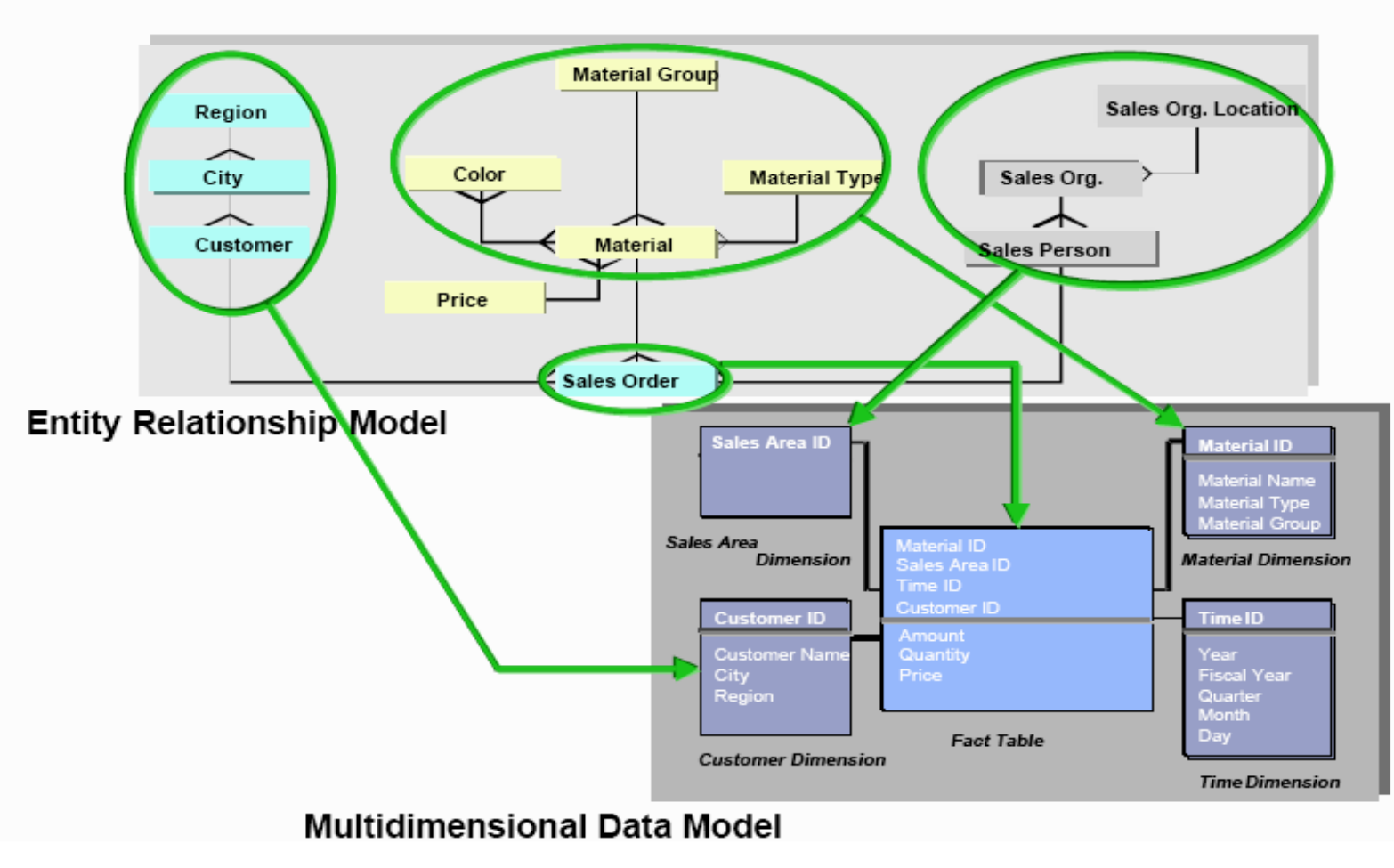




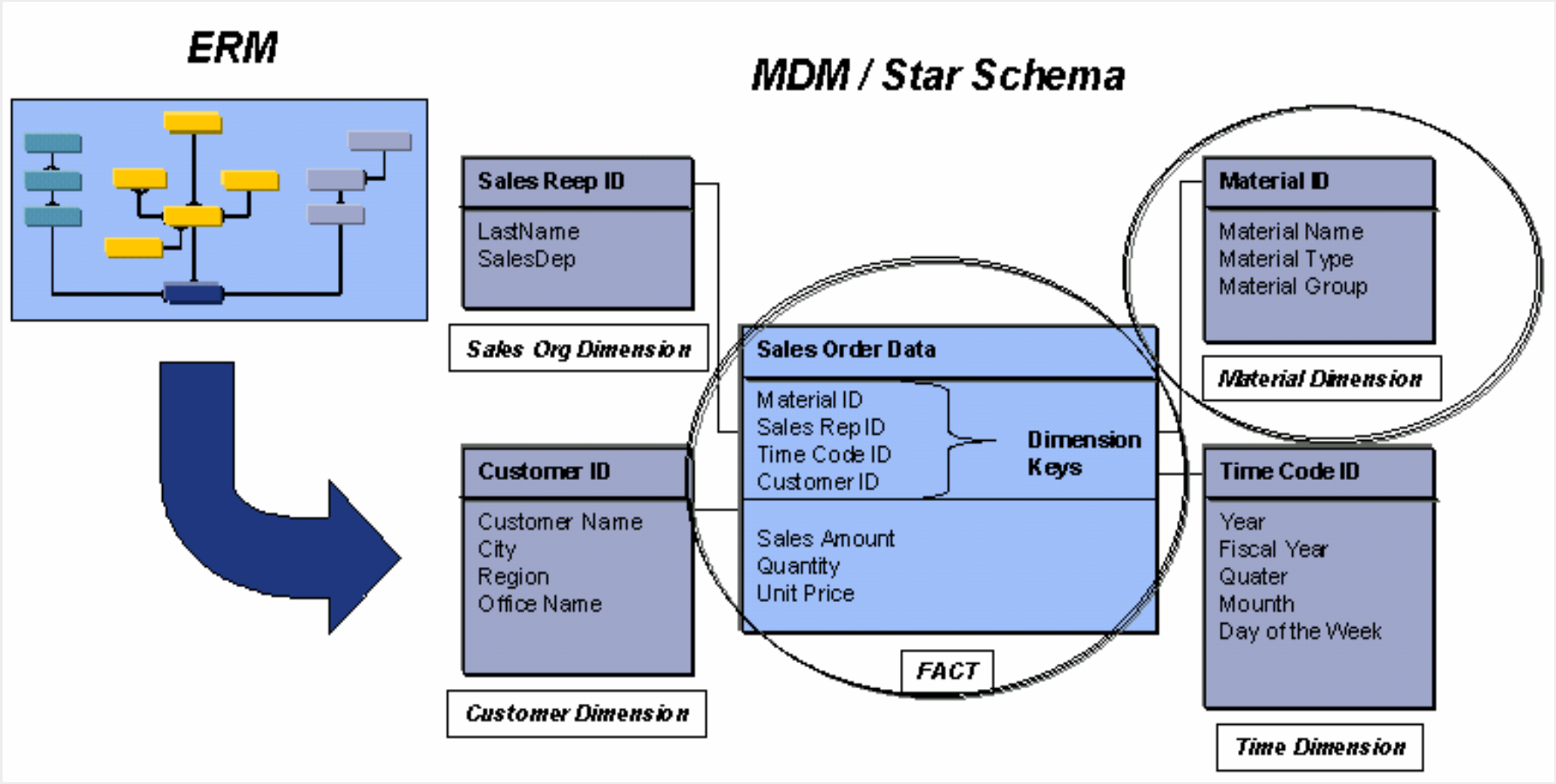


The strong entities cover the whole model. All other entities are dependent on these strong entities.

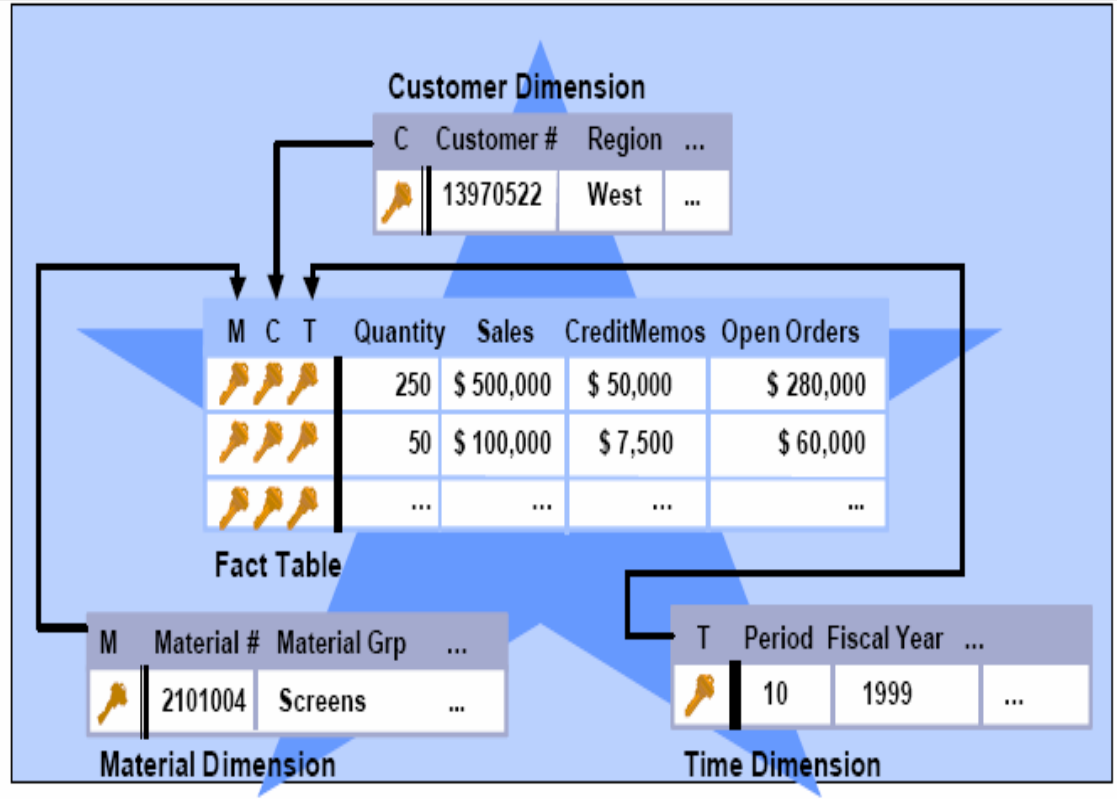
Review - Conversion From ERM to MDM

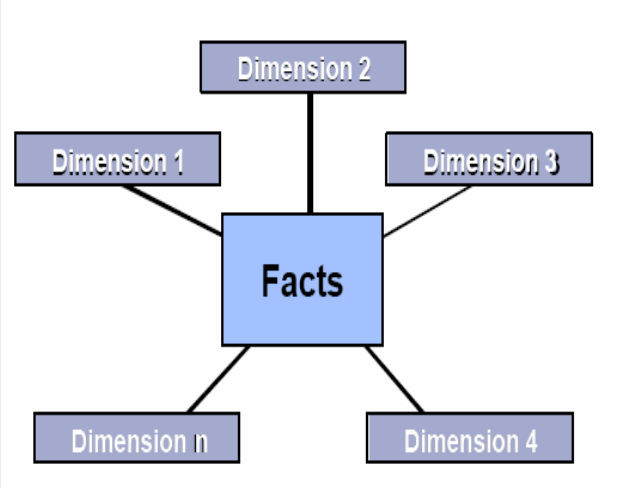


Multi-Dimensional Model

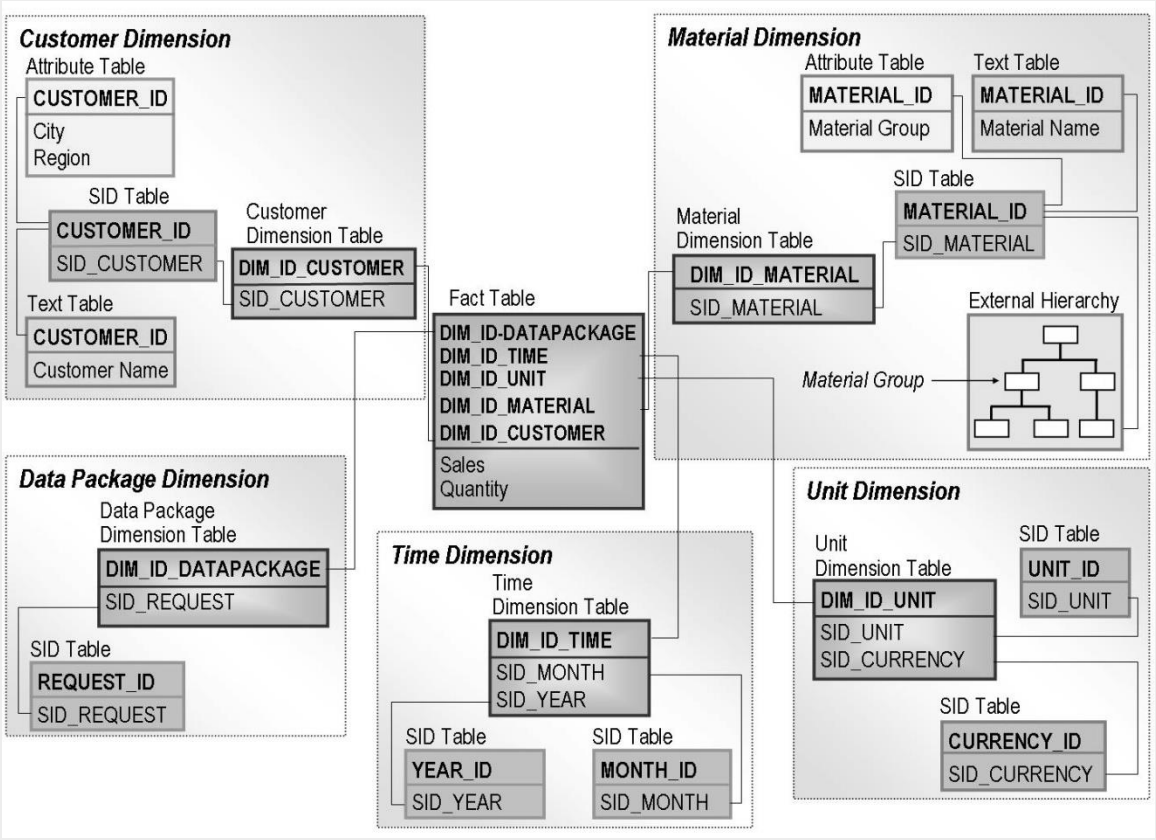


Review: Star Schema





Review – Extended Star Schema Example



Differences

Classic Star Schema	BW Star schema
Fact	Key Figure
Dimension Attribute	Characteristic
Described Attribute	Attribute Text
———	External Hierarchies
Dimension Tables (contain master data)	Dimension Tables (do not contain master data)
Dimension = Dimension Table	Dimension = Dimension Table (optional), SID Tables, Master Data Tables (optional)

Advantages

Advantages of BI star Schema :

- Thanks to the SIDs,the link to master data from the dimensions tables, the following possibilities exist:

- Easy modelling of slow dimensions (time dependant Master Data)
- Multilingual Capability
- Cross Cube of master Data (similar to shared dimensions)
- Ability to handle NULL values of Characteristics

- The use of automatically generated INT4 keys (SID keys and DIM ID keys) enables faster access to data than via long alphanumeric keys(all of our big tables are 100% numbers)

InfoProviders

Info Providers – Business Purpose

➢An InfoProvider is an object for which queries can be created or executed in BEx. InfoProviders are physical objects or sometimes logical “views” that are relevant for reporting.

➢An InfoProvider can be either physical storage of data in real database tables, or a virtual collection of data (like a view) that only collects data temporarily to feed it to a query, but does not permanently store it.

➢**2 Basic InfoProviders:**

➢InfoCube

➢Data Store Objects (DSO)

➢InfoProviders can be displayed, created, and maintained in transaction RSA1, the Data Warehousing Workbench. They are accessed via the path Modeling → InfoProvider.

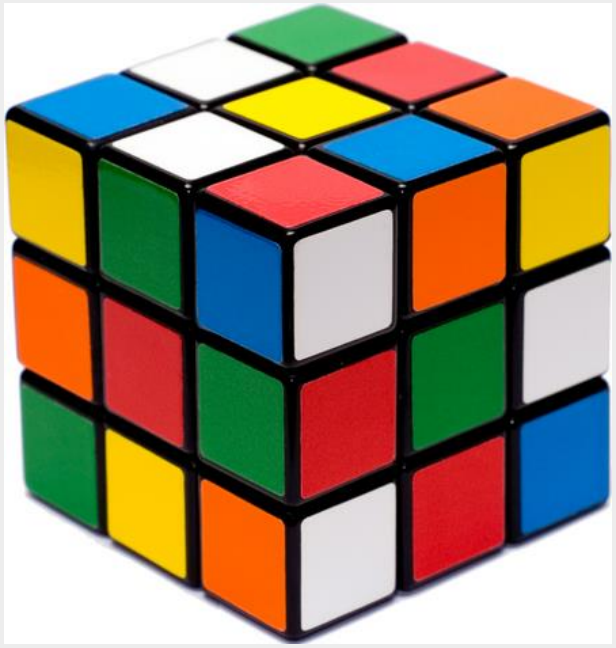
InfoCube

InfoCubes

➢InfoCubes are central objects of multi-dimensional model in SAP BI. Reports and analysis are based on InfoCubes.

➢An InfoCube describes an self-enclosed dataset for business-area from a reporting point of view.

➢An InfoCube consists of several InfoObjects and is structured according to the star schema.



InfoCubes

➢An InfoCube is additive in nature, it aggregates data on the basis of characteristics combinations.

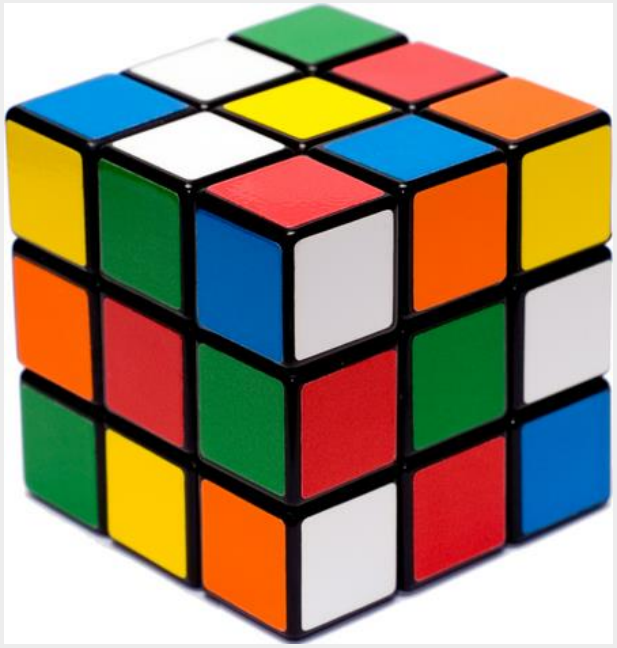
➢**Types of InfoCubes**

➢Basic

- Standard InfoCube
- Real Time InfoCube

➢**Virtual Providers**

- Based on DTP
- Based on BAPI
- Based on FM

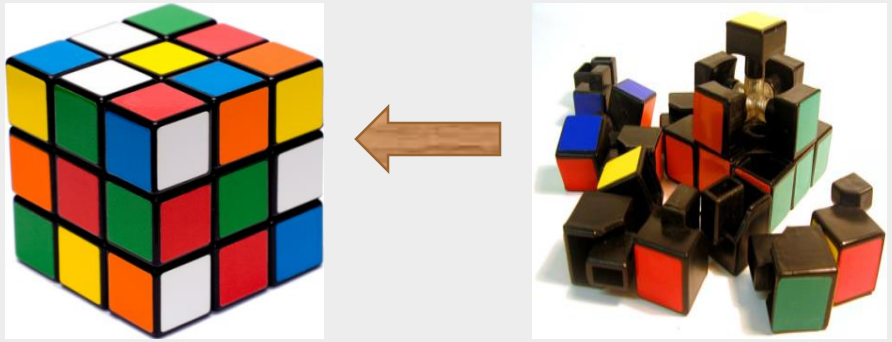


InfoCubes – Designing the Dimensions

➢The characteristics determine the **granularity** (the degree of detail) at which the key figures are kept in the InfoCube.

➢Characteristics that logically belong together (district and area, for example, belong to the regional dimension) are **grouped** together in a dimension.

➢An InfoCube can have **maximum of 16 dimensions** (3 dimensions are provided by SAP i.e. Time, Data Package and Unit). A maximum of 248 characteristics can be included in each dimension.

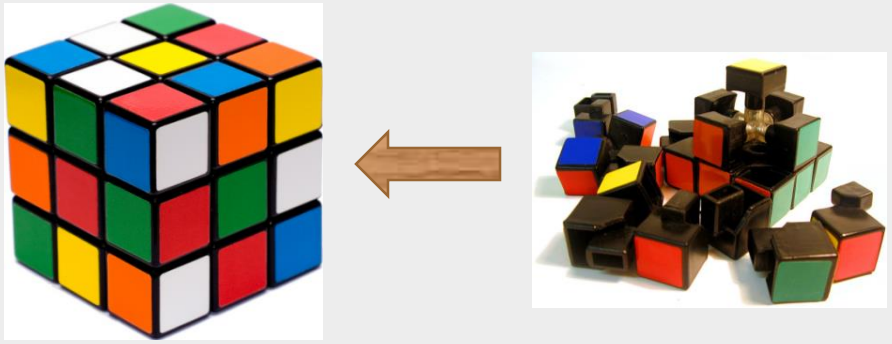


InfoCubes – Designing the Dimensions

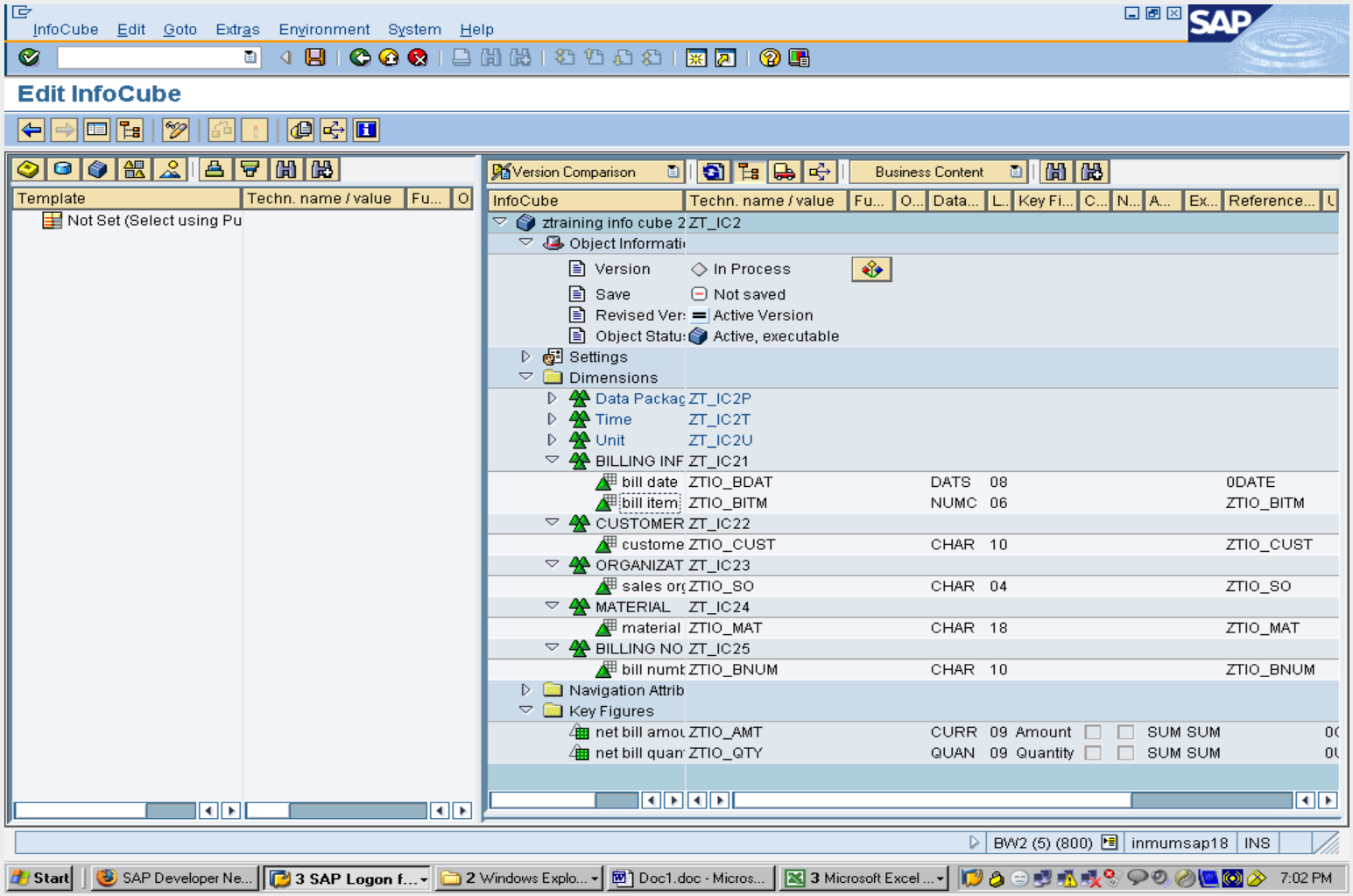
- Dimension tables should be **small** with regards to data volume, which is desirable for reasons of performance.

- In case data volume is high convert dimension to **line item dimension**.

- A cube has 2 fact tables - **E and F**. When the requests in the cube are not compressed the data exists in the F fact table and when the requests are compressed the data lies in the E fact table.



InfoCube


InfoCube Edit Goto Extras Environment System Help
SAP
Edit InfoCube
Template
Techn. name / value
Not Set (Select using Pu
Version Comparison
Business Content
InfoCube
Techn. name / value
ztraining info cube 2 ZT_IC2
Object Informati
Version In Process
Save Not saved
Revised Ver: Active Version
Object Statu: Active, executable
Settings
Dimensions
Data Packac ZT_IC2P
Time ZT_IC2T
Unit ZT_IC2U
BILLING INF ZT_IC21
bill date ZTIO_BDAT DATS 08 0DATE
bill item ZTIO_BITM NUMC 06 ZTIO_BITM
CUSTOMER ZT_IC22
custome ZTIO_CUST CHAR 10 ZTIO_CUST
ORGANIZAT ZT_IC23
sales or(ZTIO_SO CHAR 04 ZTIO_SO
MATERIAL ZT_IC24
material ZTIO_MAT CHAR 18 ZTIO_MAT
BILLING NO ZT_IC25
bill numt ZTIO_BNUM CHAR 10 ZTIO_BNUM
Navigation Attrib
Key Figures
net bill amou ZTIO_AMT CURR 09 Amount SUM SUM
net bill quan ZTIO_QTY QUAN 09 Quantity SUM SUM
BW2 (5) (800)
inmumsap18
INS
Start
SAP Developer Ne...
3 SAP Logon f...
2 Windows Explo...
Doc1.doc - Micros...
3 Microsoft Excel ...
7:02 PM

Real Time InfoCube

➢A real time InfoCube is a special Basic InfoCube, especially developed for planning applications.

➢The system accesses data in such a InfoCube is transactional, in other words, data is written to the InfoCube and instantly read again when required.

- e.g. In BW BPS/IP, users enter plan data via planning layouts which is in turn stored in Real time InfoCubes.

➢A standard Basic InfoCube is optimized for pure read access and hence not suitable here.

Virtual InfoCube

- Virtual InfoCubes are special InfoCubes. A virtual InfoCube represents a logical view. Unlike Basic InfoCube, no data is physically stored in BW. The data is taken from the source system only after a query has been executed.

- SAP virtual cube allows one to define queries with direct access to transaction data in other SAP systems.

- Virtual cube allows reporting using data from non-SAP systems. External system transfers the requested data to the OLAP processor via the BAPI.

- Virtual InfoCube with Services requires a user-defined function module as a Data Source.

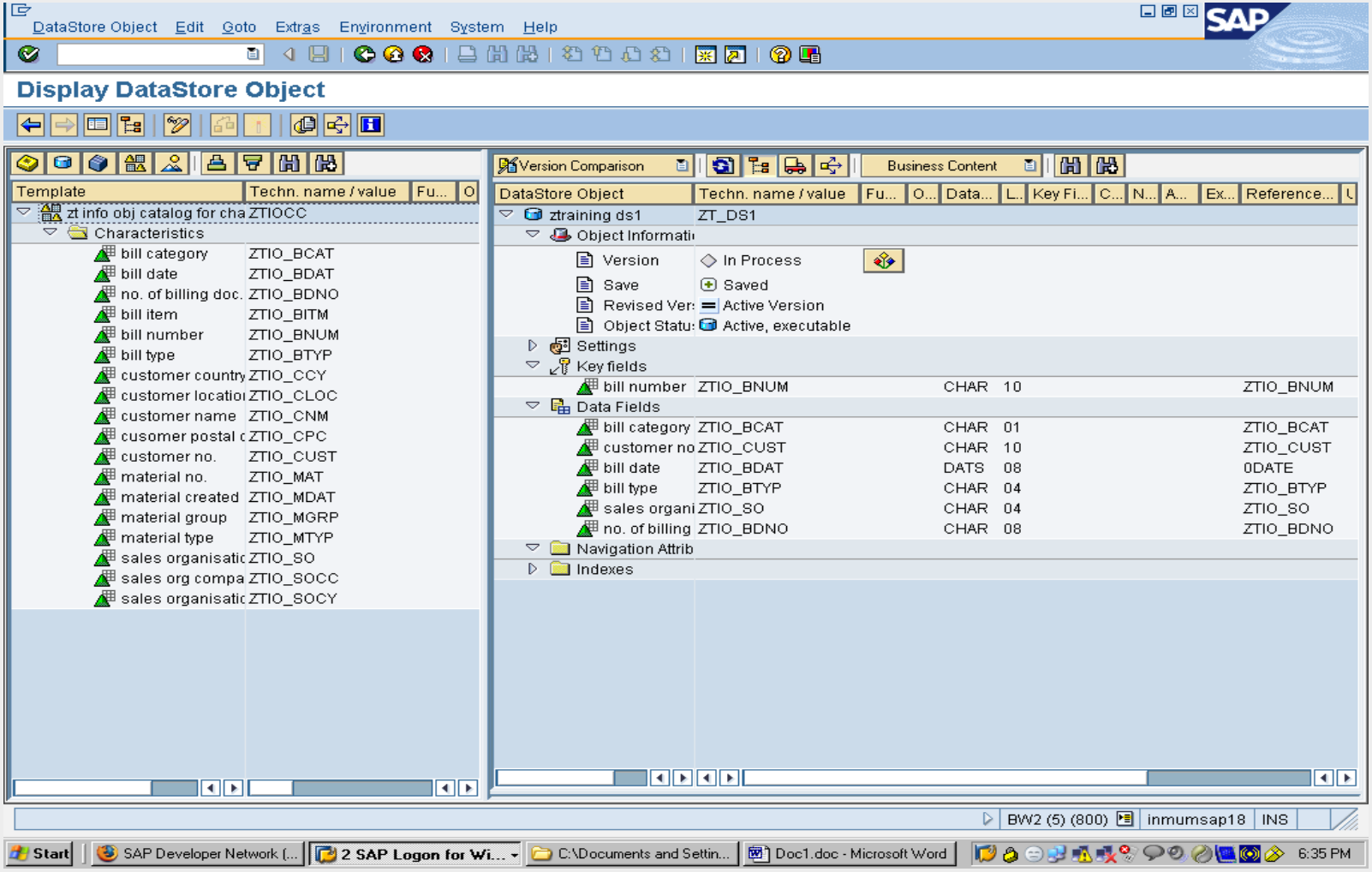
Data Store Object (DSO)

Data Store Object (DSO)

- An DSO object (Data Store Object) acts as a storage location for consolidated and cleaned-up transaction data (transaction data or master data, for example)..

- An DSO object contains key fields (for example, customer order number / item) and data fields that can also contain character fields (for example, order status, customer) and key figures.

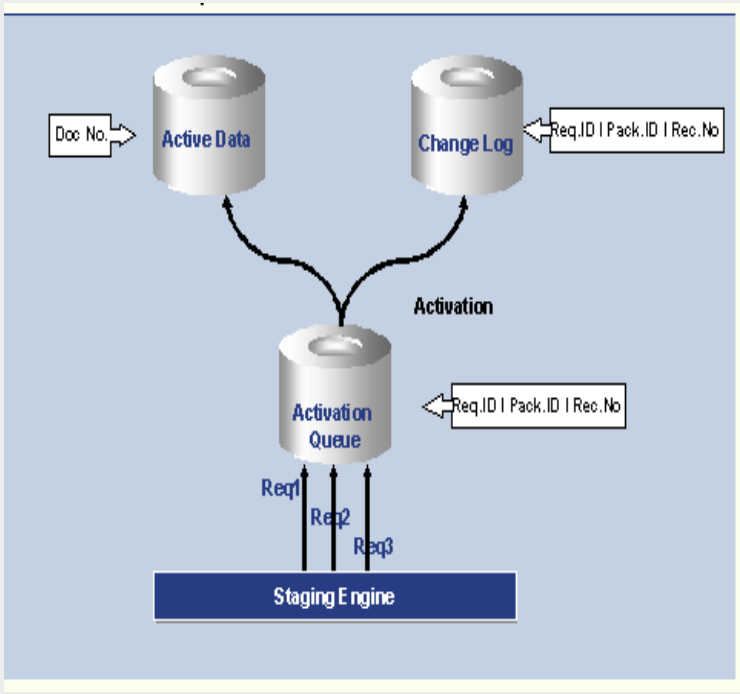
Data Store Object (DSO)



Data Store Object (DSO)

➢Every DSO object is represented on the database by three transparent tables:

- **Active data**: A table containing current status of data
- **Activation queue**: New data is saved in this table before it is activated. The data in this table is deleted after activation.
- **Change log**: Contains the complete (activation) history of the changes to DSO data



Data Store Objects

➢There are three different types of DSO's in BI 7.0

- Standard DSO
- Direct Update DSO
- Write Optimized DSO

➢DSO for Direct Update is same as Transactional ODS in 3.x

➢Write Optimized DSO is a completely new type of DSO.

Write Optimized DSO

- Consists of only one table of active data

- The system does not generate SIDs for write-optimized DataStore objects and you do not need to activate data. This means that you can save and further process data quickly.

- Reporting is possible on the basis of these DataStore objects. However, it is recommend to use them as a consolidation layer, and update the data to additional InfoProviders, standard DataStore objects, or InfoCubes.

- If two data records with the same logical key are extracted from the source, both records are saved in the DataStore object.

Scenarios for using Write Optimized DSO

➢You use write-optimized DataStore objects in the following scenarios:

- Write-optimized DataStore object can be used **as a temporary storage area for large sets of data if you are executing complex transformations** for this data before it is written to the DataStore object. Subsequently, the data can be updated to further (smaller) InfoProviders. You only have to create the complex transformations once for all data.

- It also can be used **as EDW layer for saving data**. Business rules are only applied when the data is updated to additional InfoProviders.

Direct Update DSO

- A direct update DSO object differs from a standard DSO object in the way it prepares data.

- DataStore object type is **filled using APIs** and can be **read via a BAPI**.

- Transactional ODS objects **allow data to be available quickly**. The data from this kind of ODS object is accessed transactionally and read again, as soon as possible.

Differences between Data Store Object Types

Type	Structure	Data Supply	SID Generation	Are BEx Queries Possible?
Standard	Consists of three tables: activation queue, table of active data, change log	From data transfer process	Yes	Yes
For direct update	Consists of the table of active data only	From APIs	No	Yes
Write-optimized	Consists of the table of active data only	From data transfer process	No	Yes

MultiProviders

MultiProvider

➢A MultiProviders is a type of InfoProvider that combines data from a number of InfoProviders and makes it available for analysis purposes.

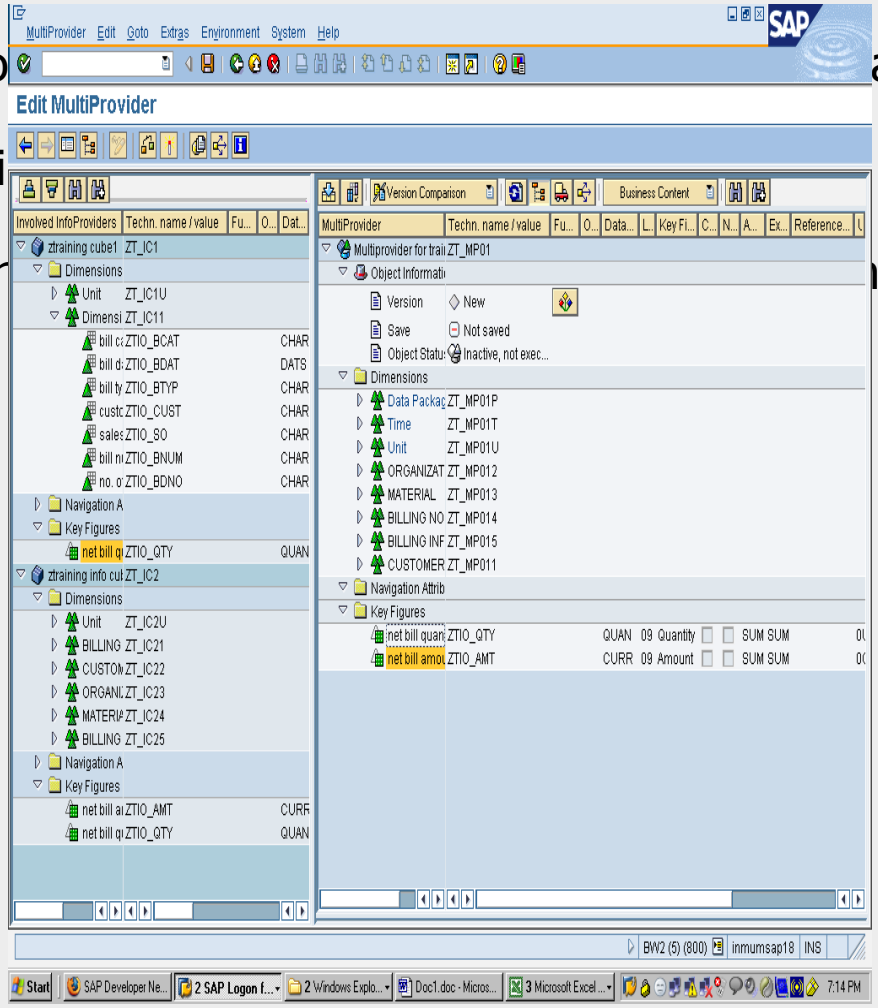
➢The MultiProviders itself does not contain any data. Its data comes entirely from the InfoProviders on which it is based. MultiProviders only exist as a logical definition. These InfoProviders are connected to one another by a **union** operation.

➢A query based on a MultiProvider is divided internally into subqueries. There is a subquery for each Info Provider included in the MultiProvider. These **subqueries are usually processed in parallel**.

- InfoProvider A contains Sales actuals. InfoProvider B contains Sales plan data. One can combine the two InfoProviders into one MultiProvider to compare actual vs. plan Sales.

MultiProvider

- One can combine InfoCube, DSO o a MultiProvider.
- A union operation is used to combi a MultiProvider.
- Here, the system constructs the ur other words, all values of these data



InfoSet

InfoSet

➢InfoSets allows reporting on several InfoProviders by using combinations of InfoObjects (master data-bearing characteristics) and DSO objects via a join condition.

➢If one of the InfoObjects contained in the join is time dependent the join is known as temporal join. (Not available in 3.x)

NOTE: For performance reasons it is not possible to define an InfoCube as a right operand of a InfoSet

➢An InfoSet can have following join conditions

- Inner Join
- Left Outer Join
- Temporal Join

➢Inner join and left outer join are only different in the situation where one of the involved tables does not contain any suitable record which meets the join conditions.

InfoSet Joins

Left Operand (Customer)	Keyfigure	Right Operand (Customer)	Keyfigure
C1	10	**C1**	5
C2	20	**C2**	7
C3	30	**C5**	3
C4	40		

Inner Join:

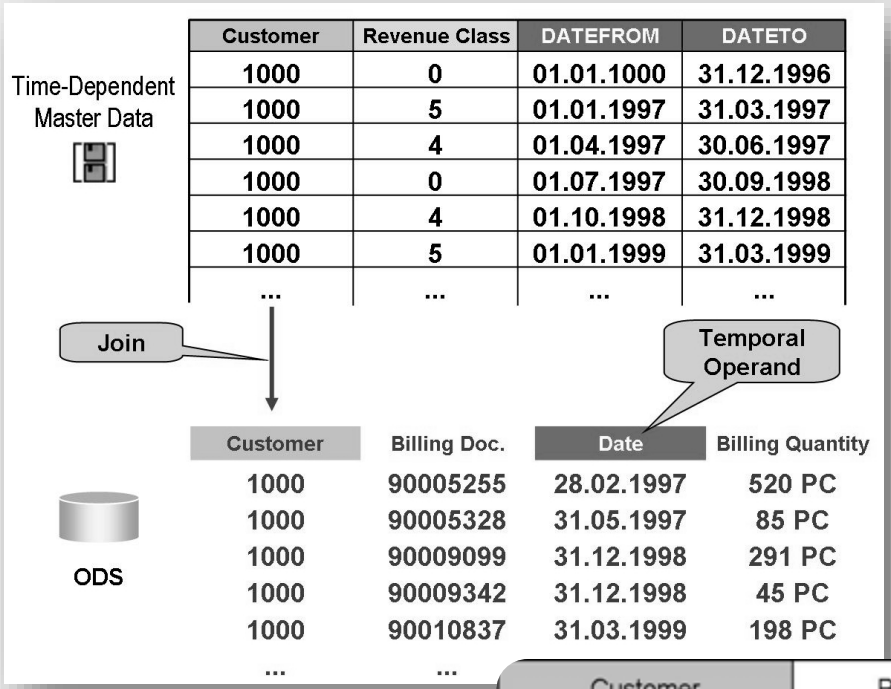
Output	C1	15
	C2	27

Left Outer Join:

Output	C1	15
	C2	27
	C3	33
	C4	40

Note that **C5** is not there in this list.
Unlike MultiProvider Union.

Infoset Joins



Temporal Join:

Customer	Billing Doc.	Date	Revenue Class	Billing Qty.
1000	90005255	28.02.1997	5	520 PC
1000	90005328	31.05.1997	4	85 PC
1000	90009099	31.12.1998	4	291 PC
1000	90009342	31.12.1998	4	45 PC
1000	90010837	31.03.1999	5	198 PC

Data Targets

Data Targets

➢A data target is an object into which data is loaded. Data targets are the **physical objects** that are relevant during data modeling and when loading the data.

➢Data targets are:

- InfoObjects (characteristics with attributes or texts)
- InfoCube
- DSO Objects

InfoProviders

➢An InfoProvider is an object for which queries can be created or executed in BEx. InfoProviders are the objects or views that are relevant for reporting.

➢ Types of InfoProviders:

- MultiProviders
- InfoSets
- Virtual Providers
- DSO
- InfoCube
- InfoObject

Data Targets vs InfoProvider

- Data targets physically store data in the underlying database tables.

- InfoProviders not necessarily store data. Data can be populated on the fly via InfoProviders by the reports.

- All Data Targets are InfoProviders but vice versa is not always true.