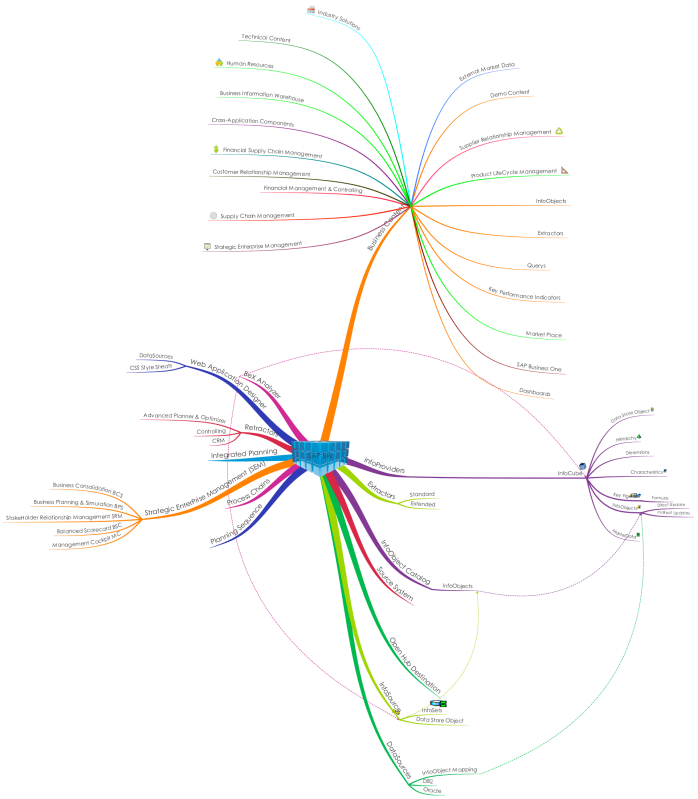
MODELING AGGREGATES



A DEEP DIVE INTO MODELING AGGREGATES

Authors: Lonnie Ayers, Doug Ayers, Victor Ayers

Table of Contents

Model Aggregates

Overview

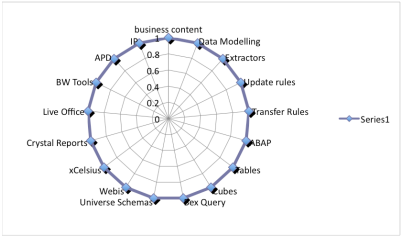
As part of our series on SAP BW Data Modeling, we next cover the subject of how to Model Aggregates. In this guide we cover:

- Granularity
- Business Warehouse Aggregates
- Partitioning
- Performance in General

In order to get the greatest value from this short guide, you need knowledge of a specific industry, such as the aircraft industry, knowledge of datawarehouse concepts, and a variety of SAP Business Warehouse Skillsets.

Aircraft Industry-Common BW User

Knowledge



BW Consulting Skillsets

About SAP BW Consulting, Inc.

SAP BW Consulting, Inc. is one of the fastest growing Business Intelligence consultancies. Our focus is on delivering value to our customers and providing a challenging set of projects for consultants. Our approach is based upon continuous education of our consultants, ourselves, and our customers.

Industry Focus

We bring more than 70 years of combined Industry Experience spanning the Military Logistics arena, Automotive Real Time embedded system, High-Tech Manufacturing, NASA and other government organizations, Rail, Airlines, Manufacturing, Consumer Packaged Goods (CPG), Airports and pharmaceuticals.

Introduction to Aggregates

The single most important decision the SAP BW data modeler makes that drives performance is the level of granularity.

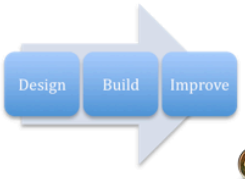
In order to manage large fact tables, SAP BW uses aggregates.

Other performance techniques are available as well, including partitioning of the tables, for example, into years.

When you finish this guide, you should be able to do the following:

- Explain what an aggregate is and what it does.
- Create aggregates using both the automated tools available from SAP and manually based on your analysis results
- Use aggregates to make reporting faster
- Understand how to maintain table spaces of aggregates in order to ensure faster reporting
- Decide on other performance enhancement techniques

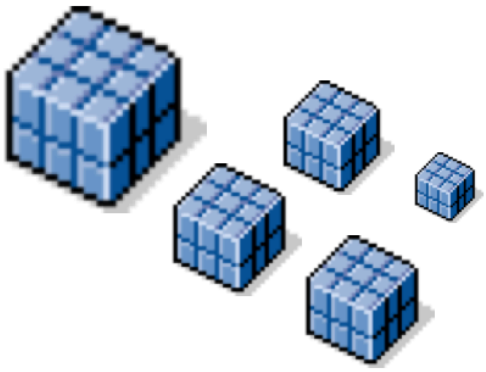
Performance Aspects: Business Scenario



- Aggregates are used to make reporting faster.
- On the other hand maintenance of aggregates is itself a performance issue.
- Enhancements improving performance of aggregate usage in queries as well as aggregate maintenance are crucial.

Steps in the Design Process

- Choose a business process to model
- Choose the grain of the process
- Choose the dimensions that apply to each fact table record
- Choose the measured facts that will populate each fact table record



Credit: Ralph Kimball, **The Data Warehouse Toolkit.**

- Depending upon the amount history required, ODS (Operational Data Stores in SAP BW 3.X and DSO (Data Store Objects in 7.X)) must be considered. A typical use case would be to use a DSO to capture the last 3 months of line item detail but use an InfoCube to store the aggregated 5 year view.
- The first step is to model a business process such as cost center accounting, sales, Human Resources or purchase orders. Typically, several InfoCubes would be used for each process.
- Then, decide on what system users should to go for line item detail vs. summarized data such as BW or R/3 (ECC). Archiving strategies can and should be a factor in this decision.
- For each InfoCube, decide on the 13 freely defined dimensions and which characteristics to put in each one. Keep in mind that any InfoObject in a dimension will then be updated via transaction data loads. That data will reflect characteristic relationships that exist in that data or are generated in the update rules.
- The measured facts are the KPI's (Key Performance Indicators) that are relevant for the business process. For example, 0QUANTITY and 0AMOUNT for the sales order process. These are also referred to as key figures and statistics.

Granularity of Data

- The granularity of data is the level of detail on the database-the characteristics that describe our key figures.
  - Most detailed level of data displayed
  - Broken down by (according to) - for example, sales by customer or sales by material
- The granularity of the data determines how far you can drilldown on the data
- Example: Granularity of time
  - Day versus month
  - A customer buys the same product 2 to 3 times a month
  - A time granularity of day results in 2 or 3 entries in the fact table
  - A time granularity of month results in 1 record being added to the fact table, but also in a loss of information (namely the number of orders for each day of the week)
  - Flights per month is less granular than flights per day per flight segment

Granularity

- Granularity is a term that describes how detailed a database is in a data warehousing context.
- Data that is "highly granular" or has "high granularity" is very detailed data, meaning that there are a large number of characteristics describing the key figures.
- A typical example would be that a "by customer" level of granularity is less detailed than "by customer, by material", because a customer may have bought many materials but is still just one customer.
- Granularity is the fundamental criteria that determines the extent to which you are able to drill down on the data.
- Granularity also affects the size of the database. Data that is stored "by Passenger, by month" is much more summarized than "by Passenger, by Class, by Route". The quantity of data that is generated over the course of a year for the first case is much less than for the second case.

- We recommend that you store your data at the "Atomic" level of granularity in the InfoCube. In other words, the transactional level data. This gives the customer the longest useful data warehouse life and maximum flexibility in the drill-downs, drillups and reporting details.

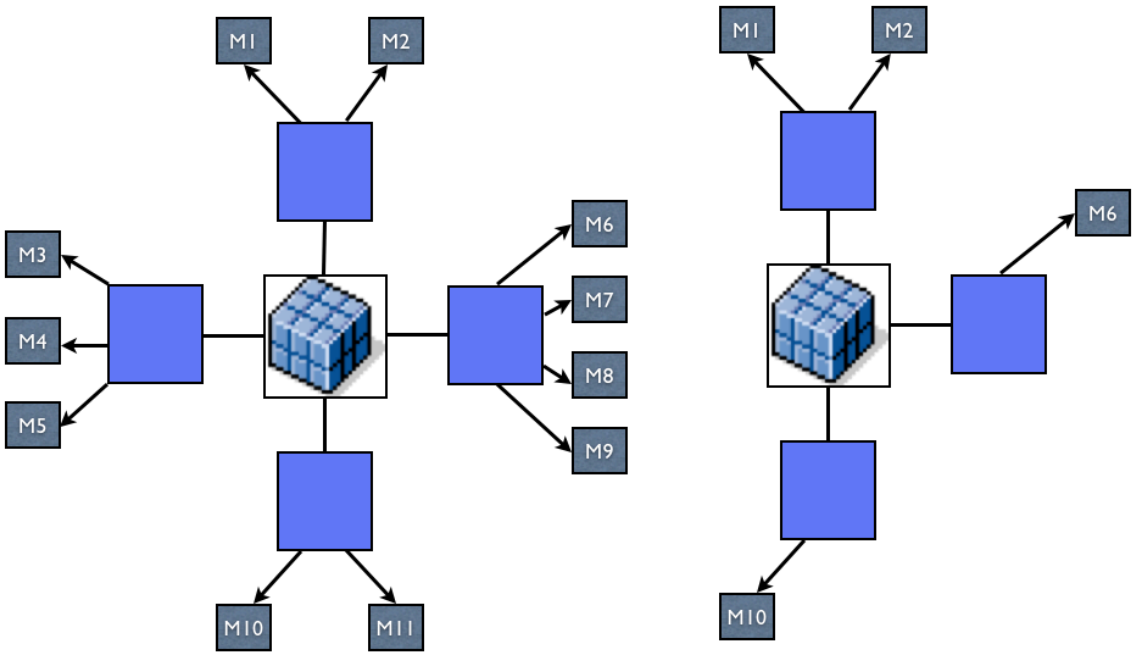
Aggregates ...

- Are like InfoCubes,
- Summarize ("aggregate") data of the originating InfoCube
- Contain redundant information,
- Accelerate the access to the information
- Are to InfoCubes as DB Indexes are to DB Tables
- Are performance-enhancing features.

- Much like a SAP BW Query, an aggregate constitutes a subset of the star schema of the related InfoCube. However, it uses its own private fact table and possibly its **own dimension tables**
- In our example, aggregates can discard certain levels of details, such as "day" and "city" or the sales organization keep data on a summarized level.
- Because an aggregate does not contain all the detailed information of the original InfoCube and as a such cannot replace an InfoCube. However, a small number of well-defined aggregates can substantially improve the performance of the standard queries that users will be executing.

- Aggregate functions happen in the background. They are not visible to the end-user. The system automatically uses an aggregate for the InfoCube that the query is written against.
- In the event exception aggregation is used, reference characteristics are added automatically to every aggregate.
- If a time characteristic delivered by SAP is the reference characteristic, all time characteristics that can be derived from it are added automatically.

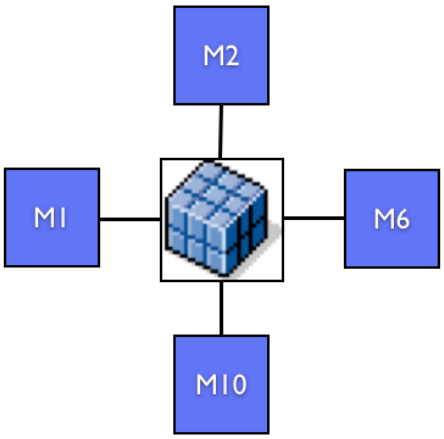
InfoCube / Aggregate Structure



- The above diagram shows the structure of an InfoCube and an Aggregate built on combination of characteristic M1, M2, M6 and M10
- In our example, the aggregate is built with a smaller multidimensional structure than the InfoCube.

Flat Aggregates

- If an aggregate has less than 15 components, each component is put into a separate dimension ("Flat Aggregates")
- The dimensions (except the package and unit) are marked as "Line Item" dimensions.



(Package and Unit Dimension not Displayed)

Aggregates - Properties

Aggregates Can Be Created:	Aggregates Other Comments:
For Basic Cubes	Stored as a new, separate, transparent InfoCube which holds the aggregated data
On Dimension Characteristics	Used to reduce the volume of data read while querying.
On Navigational Attributes	Data for one query-step may be read out of several aggregates
On Hierarchy Levels	
On Hierarchy Levels Where the Structure is Time-Dependent.	

- Aggregates cannot be created for MultiProviders, RemoteCubes or ODS-Objects (because they do not contain data, among other reasons).
- From a technical perspective, an aggregate is a separate InfoCube with its own fact table and dimension tables. When an aggregate is created, it is given a 6-digit number <1NNNNN> that starts with a "1". The table name for an aggregate is derived in the same way from this number as InfoCube table names.
- For example, if an aggregate has the technical name 1000001, its fact tables are called **/BIC/E100001** and **/BIC/F100001**. Its dimensions have the table names **/BIC/D100001P**,**/BIC/D100001T**, and so on.
- Dimension tables **can be shared** between an InfoCube and an aggregate. In this example, dimension 2 (the country dimension), is shared between the InfoCube and the aggregate. It is not necessary to create a new dimension table. A link to this dimension table is created in the aggregate fact table.
- Dimensions are only shared if **all** characteristics of the **InfoCube-dimension** are also used in the aggregate. Otherwise, a **new dimension table** is created for the aggregate.

- There is no longer a **link** from the new aggregate to the dimension customer (dimension 1), since the aggregate does not contain any information about the customer.

Creating Aggregates with Time-Dependent Components

- Aggregates with a **time-dependent** component (navigational attribute or hierarchy) are **calculated for a keydate** (corresponding to the keydate of a query)
- The keydate can be determined by:
  - **A BEx variable which is filled via an SAP or User Exit.**
  - **A fixed date.**
- This date is retrieved when the aggregate is filled.
- Only for this keydate are the aggregate values calculated.

The aggregate with a time-dependent component only contains data for a snapshot of the InfoCube/Master Data. This snapshot is determined by the key-date.

Using Time-Dependent Aggregates in Queries

- If **Time-Dependent** components are used, queries can only use aggregates with the same keydate.
- Important for the use of aggregates is not the variable but the processed keydate because only data for this keydate is available in the aggregate.

- The variables used in aggregates can be the same variables used in queries for the Key-Date.
- **For example:** A query uses **time-dependent attributes** and if the Key-date is the variable **"Current Date" (0DAT)** then **the aggregate** with time-dependent attributes can also be defined with the variable **"Current Date" (0DAT)**.

Adjustment of Time-Dependent Aggregates

- Since the keydates are subject to change, the **time-dependent** aggregates need to be updated regularly
- Process "Adjustment of Time-Dependent Aggregates" adjusts data of all aggregates with variables for the keydate to the changes of the keydate.
- This process is available in the process chains.
- Only aggregates are recalculated for which the keydate changes.
- This adjustment can be an expensive process.

- Changes in the master data means changes of navigation attributes or hierarchies, too. It is therefore recommended that you adjust the data in the aggregates after you load the master data. In order for reporting to deliver consistent results, the master data and hierarchies are in two versions.
  - ☑ The active version, where you can see the query
  - ☑ A modified version, which at some point, becomes the active version
- The change run (also called the hierarchy-attribute realignment run) adjusts the data in the aggregates and turns the modified version of the navigation attributes and hierarchies into an active version. In almost every phase of the change run, you can carry out reporting on old master data and hierarchies.
- If there are any changes to master data, they are not available for reporting until the change run is executed and finished.
- During a change run, no rollup at all is possible. Even aggregates that are not affected by the change run are locked.

Aggregates - Example Using Master Data

Master Data Table: Country

Country	Valid From	Valid to	Sales Person
Austria	1/1/2000	3/31/2000	Huber
Austria	**4/1/2001**	**12/31/2001**	**Meyer**
Germany	1/1/2000	03/31/2000	Meyer
Germany	**4/1/2000**	**12/31/2001**	**Huber**
USA	**1/1/2000**	**12/31/9999**	**Smith**

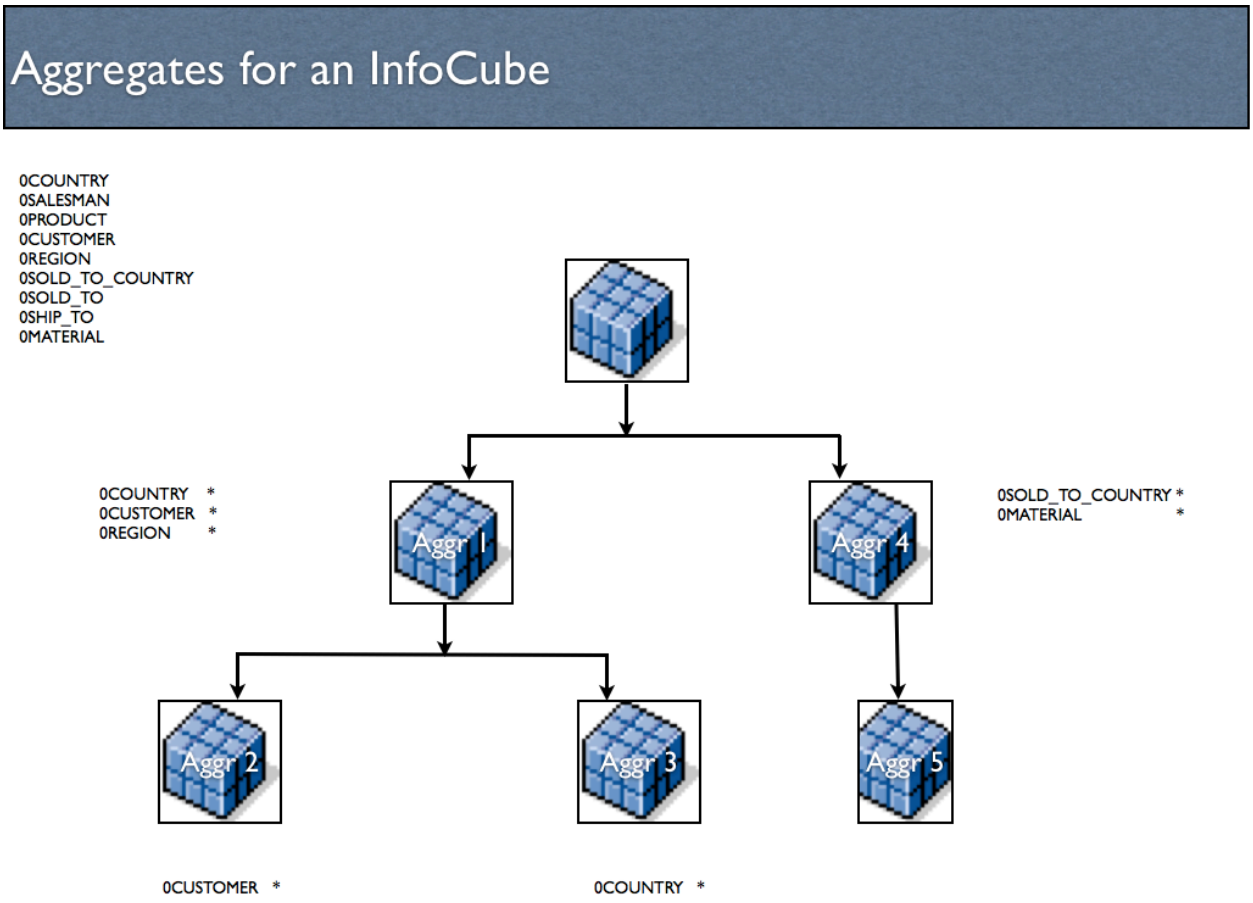
Fact Table: Sales Data

Country	Customer	Revenue
USA	Very Soft Inc.	10
Germany	Ocean Networks	15
USA	Funny Duds Inc.	5
Austria	Ocean Networks	10
Austria	Thor Industries	10
Germany	Funny Duds Inc.	20
USA	Very Soft Inc.	25

Aggregate Tables: Sales Data

Sales Person Keydate	*(all) 9/1/2001
Sale Person	**Revenue**
Huber	35
Meyer	20
Smith	40

Data for queries like "revenue of sales person 'Huber' for keydate 9/1/2012, can be read out of this aggregate.



- The graphic shows an existing set of aggregates for an InfoCube.
- A child aggregate can be built or rolled up from its parent aggregate.
- Example: Aggregate 1 can be used to roll up aggregate 3, or to recreate aggregate 3 during a change run.

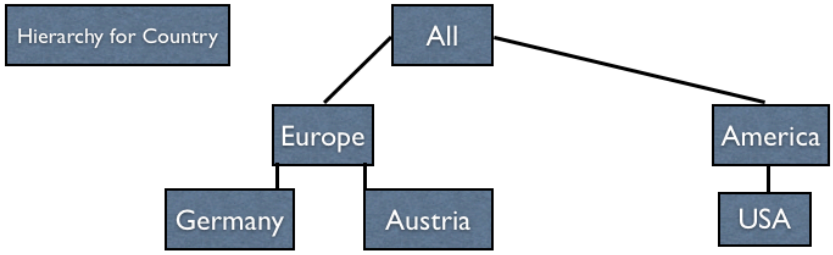
Rollup Hierarchy

The screen displays the hierarchy of all existing aggregates created for InfoCube 0SD_C03

The aggregate Basis Aggregate was created last, but BW has dynamically mapped it as the parent to all child aggregates. From now on, the child *Aggregate for the Leaf* rolls up data from *Basis Aggregate* instead of from the **InfoCube**.

Although a third aggregate was created which is in the hierarchy located on top of both aggregates, the value for summarized records is still the same. This value is always the value at the time when the aggregate was created or recreated.

Aggregates - Example Using Hierarchies



Fact Table: Sales Data

Country	Customer	Revenue
USA	Very Soft Inc.	10
Germany	Ocean Networks	15
USA	Funny Duds Inc.	5
Austria	Ocean Networks	10
Austria	Thor Industries	10
Germany	Funny Duds Inc.	20
USA	Very Soft Inc.	25

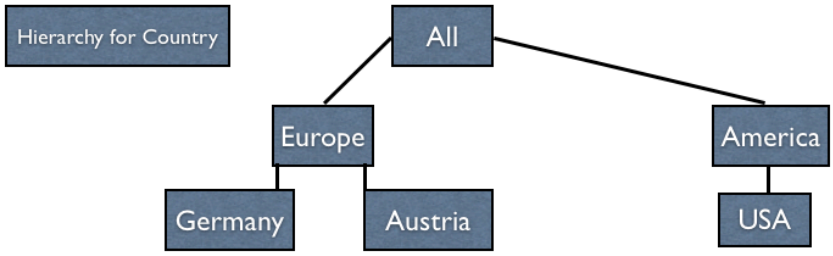
Aggregate Tables: Sales Data

Country H, Level 2

Country	Sales
America	40
Europe	55

- Queries like “sales for Europe”, “sales for ALL”, “overall sales”, or “sales for all countries ordered by the country hierarchy up to level 1 or 2” may use the aggregate (country H Level 2).
- Aggregates with a hierarchy are useful for queries which use nodes of the hierarchy as a filter or which use the hierarchy as a presentation hierarchy. (Refer to SAP note 198568 for exceptions).
- The level of the desired nodes must be less than or equal to the level in the aggregate.

Aggregates - Example Using Hierarchies



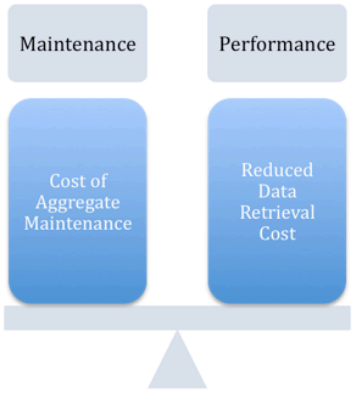
Fact Table:		Sales Data
Country	**Customer**	**Revenue**
USA	Very Soft Inc.	10
Germany	Ocean Networks	15
USA	Funny Duds Inc.	5
Austria	Ocean Networks	10
Austria	Thor Industries	10
Germany	Funny Duds Inc.	20
USA	Very Soft Inc.	25

Aggregate Tables: Sales Data	
Country	H, Level 2
Country	**Sales**
America	40
Europe	55

- Menu Path: ***Admin-Workbench>InfoProviders>right mouse click on InfoCube>Maintain aggregates***, or use transaction **RSDDV**.
- When adding the first time-dependent component (attribute or hierarchy) to an aggregate, the user is asked for a keydate.
- A fixed data (<**Calendar**>) or a variable can be chosen
- The date/variable can be changed via the context menu.
- The field "Keydate" is filled only for filled aggregates.

Query Performance vs. Aggregate Maintenance

- Tradeoff: Improved query performance vs. aggregate maintenance
- BW provides tools to find that tradeoff -> BW Statistics InfoCube



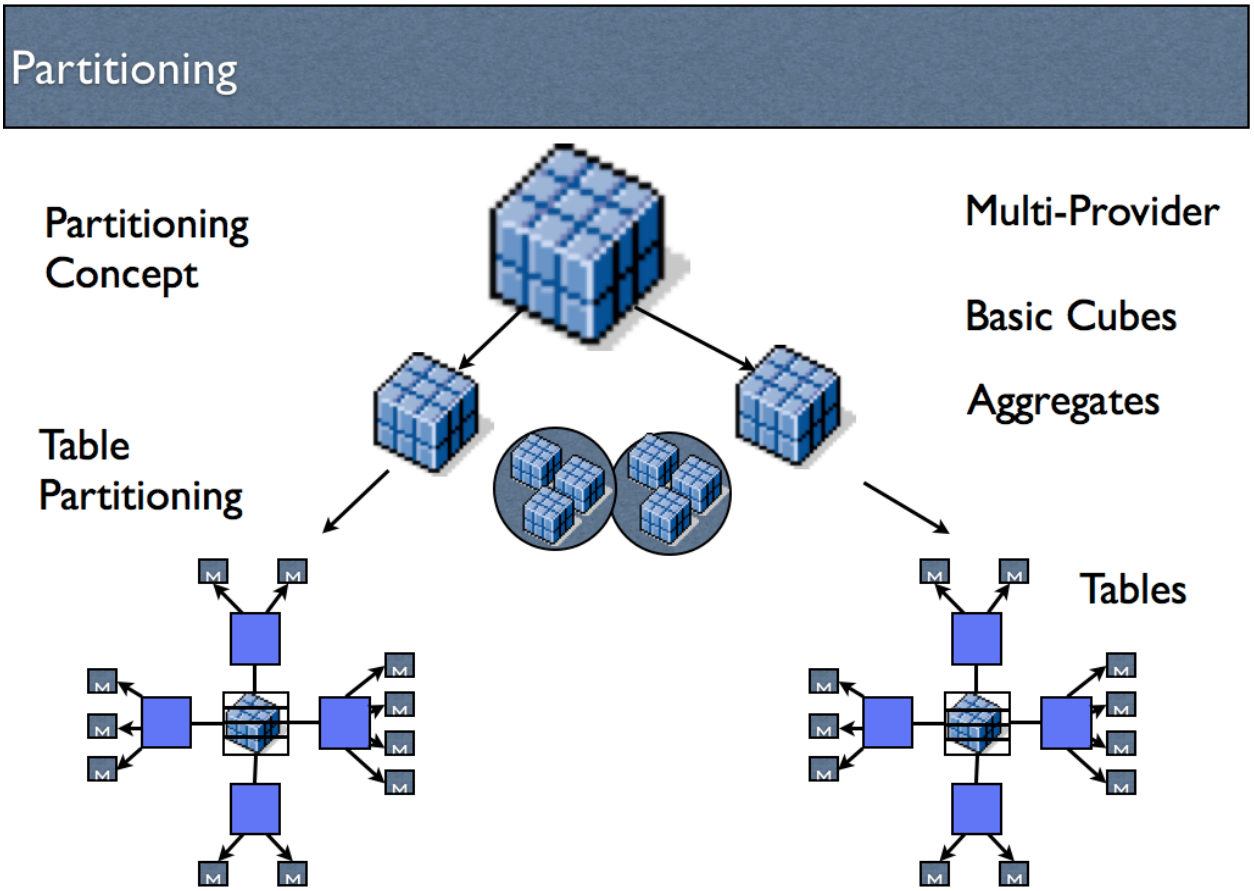
- Whenever data is loaded, the InfoCube's aggregates have to be updated as well in order to keep them in sync with the InfoCube
- A significant overhead in updating aggregates is generated. When new data is loaded, this results in an aggregate rollup needing to take place. Changes to master data and hierarchies require that all dependent aggregates be recalculated by calculating the differences/delta or by rebuilding.
- **Factors involved:**
    1. Frequency of changes that will cause recalculation.
    2. Availability of time to run the recalculation: no rollup, no master data updates, no hierarchy updates can take place during re-calculation.
- Changed aggregate data is also not available via query until recalculation is complete. Reporting on the old master data and hierarchies is possible.

- BW provides tools to determine aggregates required for improving your specific queries, and analyzing existing queries in order to identify those aggregates that are rarely used or not used at all.

Query Analysis(1)

Tools for analyzing queries: Query monitor (Transaction **RSRT**>execute & debug)

With this monitor you can analyze the first selection of the query. There is no possibility of analyzing the navigational steps. You can use the transaction **RSRTRACE** for that.



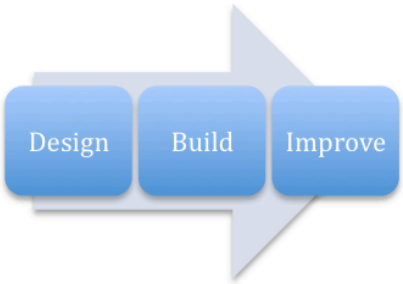
- This is simple example of how a reporting scenario can be partitioned using both partitioning concepts.
- **Multi-Provider Partitioning:** You might want to report your sales data using one InfoCube. That InfoCube could be built as a MultiProvider which is based on two identical basic InfoCubes. The latter contain disjointed sets of data, for example, one from southern sales regions and another from northern sales regions (as shown

here on this slide). This scenario results in more efficient, dense InfoCubes as opposed to combining the southern and northern region into one sparse InfoCube.

- **Table Partitioning:** Each of the two basic InfoCubes could be partitioned on the database level. That means that the fact tables inside the respective **star schema** (which physically represents an InfoCube) are partitioned. This is indicated by those horizontal lines splitting the fact tables into various partitions/fragments.

Other Performance Improvement Techniques

- Compression
- Secondary Indexes
- Load Master Data before Transaction Data
- Run Selective Queries
- Set Queries to 'Read on Demand'
- Line Item Dimensions



- Compressing the **'F'** into the **'E'** table packs records from multiple **request IDs** and results in more efficient storage and retrieval of data. This database function should be carried out when the **request ID** is not needed for data deletions.
- **Secondary indexes** are based on **database statistics** and result in more efficient read performance.
- **Master data is normally loaded first** so that the more time consuming transaction data load is slowed by having to create **SID IDs**.

- Running queries that are appropriately filtered is essential. In addition, your reporting strategy should be to read summarized data first and detail second.
- The recommended default **RSRT** setting is **'read on demand'**.
- **Line Item** dimensions are appropriate for InfoCubes with **line item detail** such as order number.

Aggregate Relevant Tables

TABLE	MEANING
RSDDAGGRDIR	Directory of Aggregates
RSDDAGGRCOMP	Description of Aggregates
RSDDAGGRT	Texts on Aggregates

SAP BW Transaction Codes

TRANSACTION CODE	MEANING
BAPI	BAPI Explorer
CMOD	Project Management of SAP Extensions
FILE	Maintenance of Logical File Paths
LISTCUBE	List Viewer for Data Targets such as InfoCubes, ODS Objects, Characteristic InfoObjects
LISTSCHEMA	Schema Viewer for BasicCubes (including Aggregates)
PFCG	Role Maintenance
RRC1, RRC2, and RRC3	Create, Modify and Display Definitions for Currency Conversion
RRMX	Start BEx Analyzer
RS12	Display and Delete Locked Entries (of Tables)
RSA1	Administrator Workbench (->Modeling)
RSA11	Administrator Workbench (->InfoProvider)
RSA12	Administrator Workbench (->InfoSources)
RSA13	Administrator Workbench (->Source Systems)
RSA14	Administrator Workbench (->InfoObjects)
RSA3	Extractor Checker from SAPI 3.0
RSA5	Transfer DataSources from Business Content

TRANSACTION CODE	MEANING
RSA6	DataSource and Application Component Hierarchy Maintenance
RSA7	Maintenance of Delta Queue
RSA9	Transfer Application Components from Business Content
RSBBS	Maintain Blocked Lines for the Report-to-Report (RRI)
RSCUSTV1	Change Settings for Flat Files. (->Thousands, Decimal, and Field Separators; Field Delimiters)
RSCUSTV6	Change Threshold Values for Data Loading (Package Size, Size of a PSA Partition, and IDOC Frequency Status)
RSCUSTV8	Change Settings for Aggregate Change Run (Threshold Value for Restructuring and Block Size)
RSD1, RSD2, and RSD3	Maintenance of InfoObjects: Characteristic/Flag/Unit Types
RSD4 and RSD5	Editing of Technical Characteristics and Time Characteristics
RSDBC	DB Connect: Select Tables and Views (Differs slightly between SAP BW 3.X and SAP BW 7.X systems in the view you see.
RSDDV	Maintenance of Aggregates
RSRT	Query Monitor
ST03	SAP BW Statistics
RSDIOBC	Editing of InfoObject Catalogs
RSDMD	Maintenance of Master Data (for one Characteristic)
RSDMPROM	Editing MultiProviders
RSDODS	Editing Objects
RSDV	Maintenance of the Validity Slice (BasicCubes with Flag Type of Non-Cumulative Value)
RSFH	Test Tool for Extraction of Transaction Data

TRANSACTION CODE	MEANING
RSIMG	SAP BW Customizing Implementation Guide
RSISET	Maintenance of InfoSets
RSKC	Maintenance of Permitted Additional Characters in SAP BW (special characters from other languages)
RSMD	Test Tool for Extraction of Master Data
RSMO	Monitor
RSMON	Administrator Workbench (->Monitor)
RSMONCOLOR	Valuation of Requests
RSO2	Maintenance of Generic DataSources
RSO3	Setup of Delta Extraction for Attributes and Texts
RSOR	Administrator Workbench (->Metadata Repository)
RSORBCT	Administrator Workbench (->Business Contenent)
RSPC	Maintenance of Process Chains
RSRTRACE	Query Trace
RSRV	Analysis and Repair of SAP BW Objects
RSSM	Maintenance of Reporting Authorization Objects
RSU1/RSU2/RSU3	Create, Modify and Display Update Rules (BasicCubes and ODS Objects)
SBIW (You must know this by heart)	Display of Implementation Guide (Customizing for Extractors)
SE03	Transport Organizer Tools
SE09	Transport Organizer
SE11	ABAP Dictionary
SE16	ABAP Data Browser
SE37	Function Builder (Maintenance of Function Modules)
SE38	ABAP Editor (Maintenance of ABAP Programs)
SE80	Object Navigator

TRANSACTION CODE	MEANING
SICF	Maintenance of System Internet Communication Framework
SM04	User List
SM12	Selection of Blocked Entries
SM21	Online Analysis of System Log
SM37	Job Overview
SM38	Queue (Job)-Definition
SM50	Process Overview
SM59	Maintenance of RFC Connections
SM62	Maintenance of Events
SM66	Global Work Process Overview
SMX	System->See Your Own Jobs
SPRO	Customizing Guidelines
SQ02	Maintenance of SAP Query / InfoSets
SQ10	Assignment of Query / InfoSets to User and Role
ST05	Performance Analysis (SQL Trace)
ST22	ABAP Dump Analysis
SU01	Maintenance of Users (You create them here)
SU24	Maintenance of Role Templates (Use Business Content Delivered Roles to Start This Process)
SU53	Resolve Error Codes (at the Authorization Level)
TRSA	Service API Test Tool

Some Useful R/3 (ECC) Transaction Code for BW Consultants

TRANSACTION	MEANING
LBWE	Customizing Cockpit for Logistics Extract Structures (LIS)

TRANSACTION	MEANING
KEBO	Create, Display and Delete CO-PA DataSource
RSA3	Extractor Checker for Service API 3.0 and Up
RSA5	Transfer DataSources from Business Content
RSA6	Perform Follow-Up Work on DataSources and Application Component Hierarchy
RSA7	Maintenance of Delta Queue
RSA9	Transfer Application Components from Business Content
RSO2	Maintenance of Generic DataSources
RSO3	Setup of Delta Extraction for Attributes and Texts
SBIW	Display of Implementation Guide (Customizing of Extractors)
SMQ1	qRFC Monitor (Output Queue)
TRSA	Test Tool for Service API

Glossary

A) Aggregates

Are like mini-infocubes and summarize data from an InfoCube. They provide faster query response times.

B) Calculated Key Figure

A Key Figure which is calculated or derived other Key Figures.

C) Change Run

Known as Hierarchy-attribute realignment run. Adjust the data in aggregates and turns the modified version of the navigation attributes and hierarchies into an active version.

D) Cumulative Values

Cumulative Value Key Figures are those Key Figures that are cumulated using all characteristics, thus also using time.

E) Granularity

The level of detail of data within your data model. Customer is less granular than Customer + Order Details.

F) Flat Aggregates

When an aggregate has less than 15 components, each component is put into a separate dimension.

G) Fact Table

Where the facts of an InfoCube are held.

H) Factless Key Figures

A key figure that is the intersection value of two tables. For instance, you can count the number of occurrences of a value that is present in the two tables.

I) InfoCube

An InfoCube is the central data storage object in SAP Business Warehouse. Its structure is set up to allow optimized query performance. It uses the SAP Extended Star Schema. There are several types of InfoCubes, some contain data, and some do not.

J) InfoSets

Different from the SAP query / InfoSet tool in that they are accessed via the SAP BW BEx.

K) Internal Business Volume

When two or more transactions occur between units, typically profit or cost centers, within a company, you need to 'net them out' in order to avoid double counting them.

L) InfoProvider

An element that is visible via BEx Query designer and can thus be reported on.

M) Key Figures

The answer you are trying to find when performing analysis. Examples include: Sales Totals, Sales by Customer, Profit and Loss, and many others.

N) Inner Join

Result contains all records that are common to both InfoProviders (with respect to the join condition).

O) Left Outer Join

A join condition that will return all the records contained in the first table, and any matching records in the second table that forms part of the join.

P) Master Data

Master Data is data that does not change very often (with some exceptions depending on the Industry), and includes, for example, Customer Names, Product Codes, or Material Safety Data Sheets.

Q) MultiProvider

A MultiProvider is a special InfoProvider that combines data from several InfoProviders. It does not contain any data.

R) Non-Cumulative Key Figure Values

Non-Cumulative Key Figure Values are those key figures that are measured in relation to a period of time; they cannot be meaningfully cumulated over time. Non-cumulative values are summarized over time using exception aggregation.

S) OLAP

An On-Line Analytical Processing or OLAP system is a system such as SAP Business Warehouse, which is, as it's title implies, optimized for analysis, and is not intended to perform business transactions, such as execute Sales Orders.

T) OLTP

An On-Line Transaction Processing or OLTP system is a system such as SAP R/3 (ECC) that performs business transactions, such as issue and process purchase orders.

U) Transitive Attributes

Transitive attributes are attributes at the secondary level. Suppose, for example, you have an InfoObject called Customer that has an attribute of Region, and that attribute, Region, has an attribute of Country. You can set up a process that you can report on Country via Customer.

V) Temporal Join

Used to show time dependent records.

W) Unions

Whereas a Join is used to find the intersection two groups of items have in common, a Union is used when creating a MultiProvider, and allows you join information from various InfoProvider