

Introduction to Data loading using Virtual Providers in SAP® BI 7.0



Applies to:

SAP BI 7.0. For SAP 4.6C with plug-in 2004.1, this option has been available since Service pack 10.

For more information, visit the [Business Intelligence Homepage](#).

Summary

Virtual Info Cubes are Info Providers with transaction data. It is not stored in the objects themselves, but is read directly for analysis and reporting purposes. Virtual Providers using DTP is used for Up-to-date information from SAP Source system provided very small quantity of data exists. In this article, we'll deal with such a situation as a demo example, wherein data is loaded by using DTP. There are also other ways, like using BAPI. In case of VPs based on BAPIs, the transaction data is read from a different source which is called BAPI (Business Application Program Interface) and not specifically from an SAP source. Similarly, we have VPs using FM also, incase data needs to be loaded from non-BI Data Sources in BI without having to copy the dataset into the BI structures

Author: Arun Bala Ganesan

Company: Wipro Technologies

Created on: 22 February 2010

Author Bio



The author is presently working in Wipro Technologies with 1+ experience as Associate SAP BI Developer. His expertise includes SAP BI 7.0 and ABAP.

Table of Contents

Introduction	3
Creating a Virtual Info cube	3
Direct Access	3
3 types of Virtual Providers	3
Virtual Providers with Staging Connect	3
Virtual Providers with BAPI	3
Virtual Providers using Function Module	4
Step-by-Step Guide	4
Implementation Results	14
Related Content	15
Disclaimer and Liability Notice	16

Introduction

Virtual info provider with direct access is really slow, and must be threaded wisely. If we have a lot of records transmitting between the remote function calls. There might be a situation where data for the Info provider need not be stored physically. One such model is an Virtual Info provider. This acts very similar to an Info cube, but is not physical data storage, with dimensions, characteristics and key figures like normal Info cubes. But the actual generation of data for this cube, depends on one's making.

We can choose to create a virtual provider with a 'Direct access' to R/3. This means that upon viewing the data in this virtual provider, a remote function call is called direct to R/3 to get the values on the fly.

Creating a Virtual Info cube

In SAP BI 7.0, just right click on any info area, and choose create 'virtual provider'. For direct access, you choose Virtual Provider with 'direct access'. There is one more method by using the old R/3 info source.

Direct Access

Direct Access is suitable only for data sources with especially small data volume and for a small number of accesses. Direct Access is relevant for both transactional data and for master data, and is mapped by Virtual Info Providers, that are defined like regular Cubes & regular IOs. But exist only in the meta data of the BI system.

3 types of Virtual Providers

- VPs with Staging Connect (*Simple DTP*)
- VPs using BAPI (for transactional data only)
- VPs using Function Module

Virtual Providers with Staging Connect

VPs with Staging Connect offer an option to access the data of a source system through Data Sources of the Source system & use transformation procedure for the supply.

VPs for transactional data are modeled as Info Cubes, but we must choose the type which only selects Meta data. Virtual Providers for transactional data access the data source in the source system for characteristics & key figure. However, it always reads master data of characteristics from the existing master data tables in BI system.

Note: Attributes like IOs can be defined only as display & not as navigational data.

Each Virtual Provider can access a random number of Data Sources. The precondition is the ability of respective data source to provide data to BI system via direct access. The connection between data sources & a virtual provider is defined with regular staging by a transformation & DTP.

How many Data sources supply data to a virtual provider depends on the defined transformations & DTP.

With definition of DTP, we must select the option '*Do not extract from PSA but Access Data Source*' and the adapter for synchronous extraction. The data sources which are actually to be read with direct access need to be redefined explicitly. This is done in the context menu of the respective target structure (master data attributes or texts or cube) and is activated via '*Activate Direct Access*'

Virtual Providers with BAPI

There are also other ways, like using BAPI. In case of VPs based on BAPIs, the transaction data is read from a different source which is called BAPI (Business Application Program Interface) and not specifically from an SAP source.

Using VPs with BAPI, the systems from the third party suppliers can be used to retrieve data from these systems. The principle is similar to the virtual provider with staging connect, but the obtained data is not even processed in a transformation.

Data requirement is not directed to the data sources of the source system, but requires a specifically prepared 3rd party source system that provides a corresponding BAPI to exchange data.

It needs to be already supplied in the form required for analysis.

The creation of a VP with BAPI is similar to creation of VP with staging connect, where the type based on a BAPI is selected and the RFC destination to respective 3rd party source system is indicated.

Virtual providers with BAPI are used in special cases if data is to be obtained from a market data supplier. Up to release of 3.X of BW, BAPI was referred as general Remote Cube.

Virtual Providers using Function Module

In comparison to other Virtual Providers, this one is more generic. It offers more flexibility, but also requires a higher implementation effort. The data can be local or remote. We can also use your own logic to change data before it is passed to OLAP processor.

We use this incase data needs to be loaded from non-BI Data Sources in BI without having to copy the dataset into the BI structures

Step-by-Step Guide

Go to RSA1 (*Data Warehousing Workbench*) & follow the steps as below,

- Creation of Info area: In RSA1, under 'Info objects' we create an Info area. We enter description & its technical name & Save it. Creation of Characteristic Info Object Catalog: Here, we now right click on our 'Info Area' & choose the option 'Create Info object Catalog' to create Char. IOC.
- Editing Char. Info object Catalog: In this screen, we enter Info object catalog name & its technical description. We need to create a characteristic info object, hence we choose that option. After we key in the details, press F5.
- Creation of IO: We now right click the newly created Info object & choose option 'Create Info object'. While creating the characteristic Info object, we need to enter the char. Technical name & its long description.

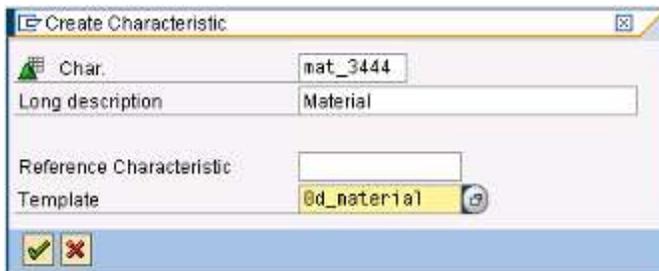


Fig.1.1 - *Creating Characteristic:*

Note: We use a reference characteristic, which is a template of type '0D_MATERIAL' (Material)

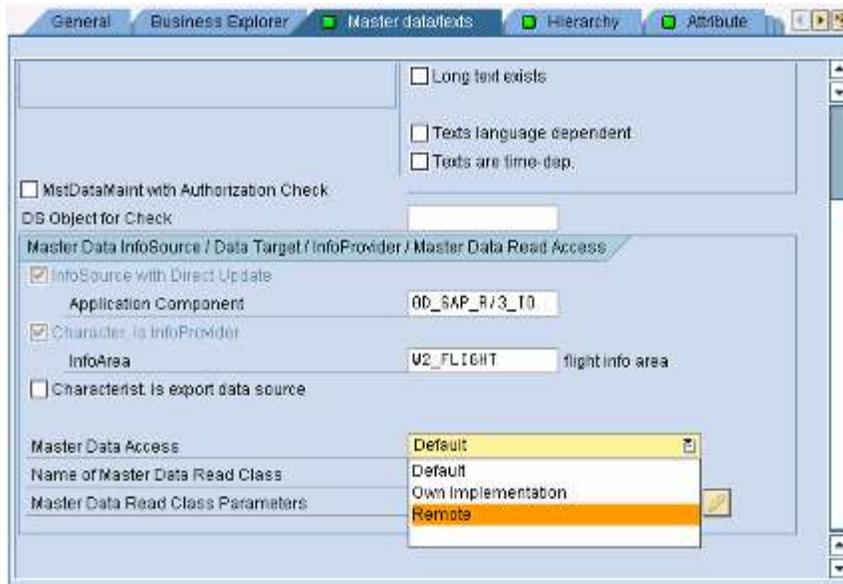


Fig.1.2 - Choice of Master Data Access:

- Now, under the tab 'Master Data Objects', we need to choose options for the 'Master Data Access'.
- We have several options like - 'Default', 'Own Implementation' & 'Remote'. Here, as we are trying to create a Virtual Info provider, we need to specify the access for this IO as 'Remote'.
- Once chosen, we must save & activate.

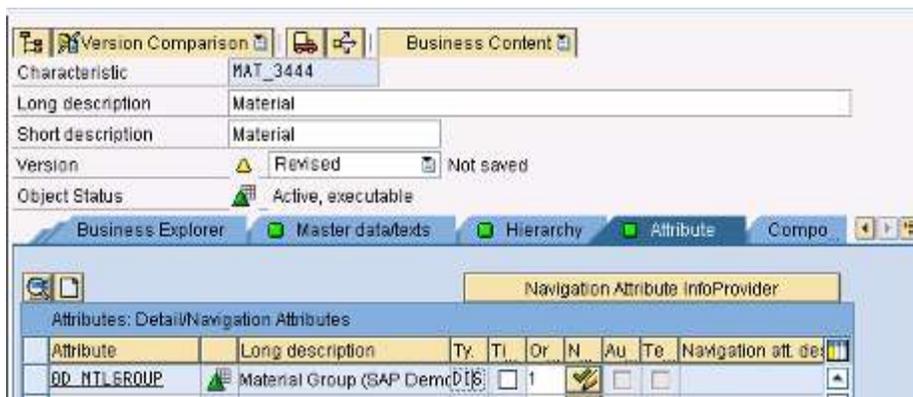


Fig.1.3 - Change of Attribute Type: Make NAV → DIS

- Here, we need to change the attribute type of OD_MATERIAL to 'Display' from 'Navigational'.
- Creating of KF IO Catalog: Here, once the change has been introduced. We need to create a key figure IO, by choosing 'Create Info Object Catalog' option upon right click.
- Edit Info object Catalog: Here, we need enter the Info object catalog name & enter name of the IA. We need to create a key figure info object, hence we choose that option.
- After we key in the details, press F5.
- Creation of IO: We now right click newly created Info object & choose option 'Create Info object'

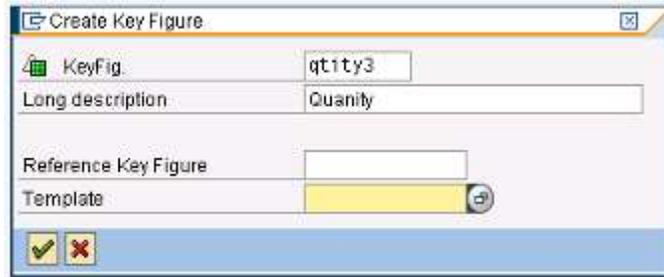


Fig.1.4 - Creating Key Figure

- While creating the characteristic Info object, we need to enter KF's technical name & its long description (Quantity).

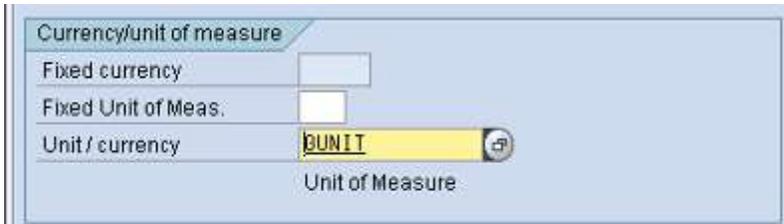


Fig.1.5 - Unit of Measure

- Now, we must enter unit of measure for 'Quantity'. Its type is Quantity & Unit of measure is 0UNIT



- Activation: Once done, we now activate the Key figure IO.

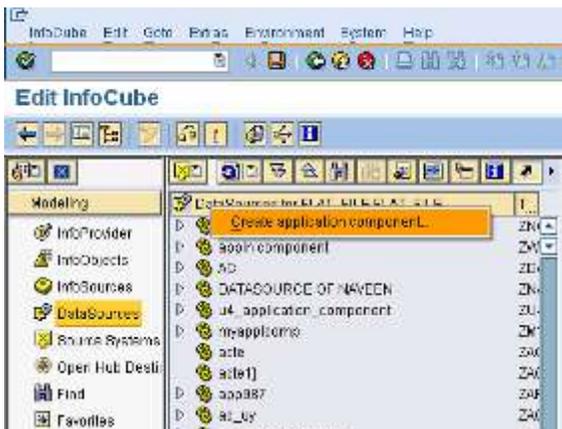


Fig.1.6 - Creation of Application Comp. Area:

- We now go to 'Data sources' tab in RSA1 and right click at the top and choose option 'Create Application Component'. Enter the name & save.

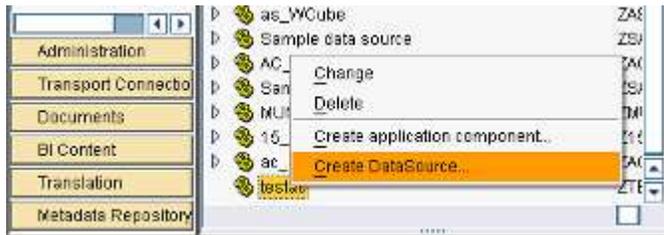


Fig.1.7 - Creation of Data source: Here, we need to create a data source.

So right click the particular 'Application Comp Area' & choose 'Create Data Source'



Fig.1.8 - Create Data source: Now, we must enter the Data source details like Name, Source System which is a Flat file, and Data type DS which is Transactional Data.

Once done press 'Enter' to transfer the details.

 A screenshot of a Microsoft Excel spreadsheet titled 'TEXT1'. The spreadsheet has columns A through G and rows 1 through 5. The data is as follows:

	A	B	C	D	E	F	G
1	MATID	MATGROUP	QUNIT	QQUANTITY			
2	MX100	TV	KG	300			
3	MX200	TV	KG	320			
4							
5							

Fig.1.9 - Flat File Creation: Once done we must create a flat file like CSV/XLS. A structure as this must be created [MATID, MATGROUP, QUNIT, QQUANTITY].

For the sake of transactions, we enter certain records (2 records as in example)

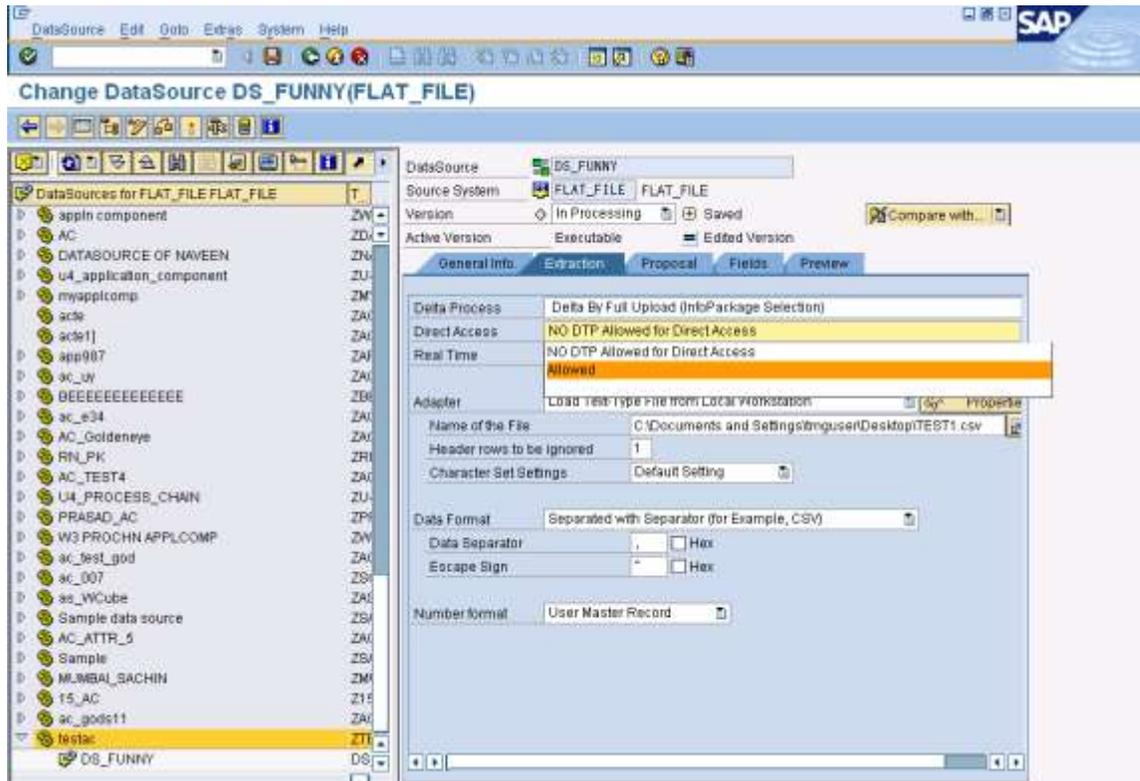


Fig. 1.10 - Extraction of Flat File Data to DS: At this step, under the 'Extraction' tab, we choose the option Direct Access: 'Allowed'.

Recall that in case of RDA, we used the 'Real time access' option. Here as this is a case of Virtual Info provider, we have chosen the Direct Access as 'Allowed'

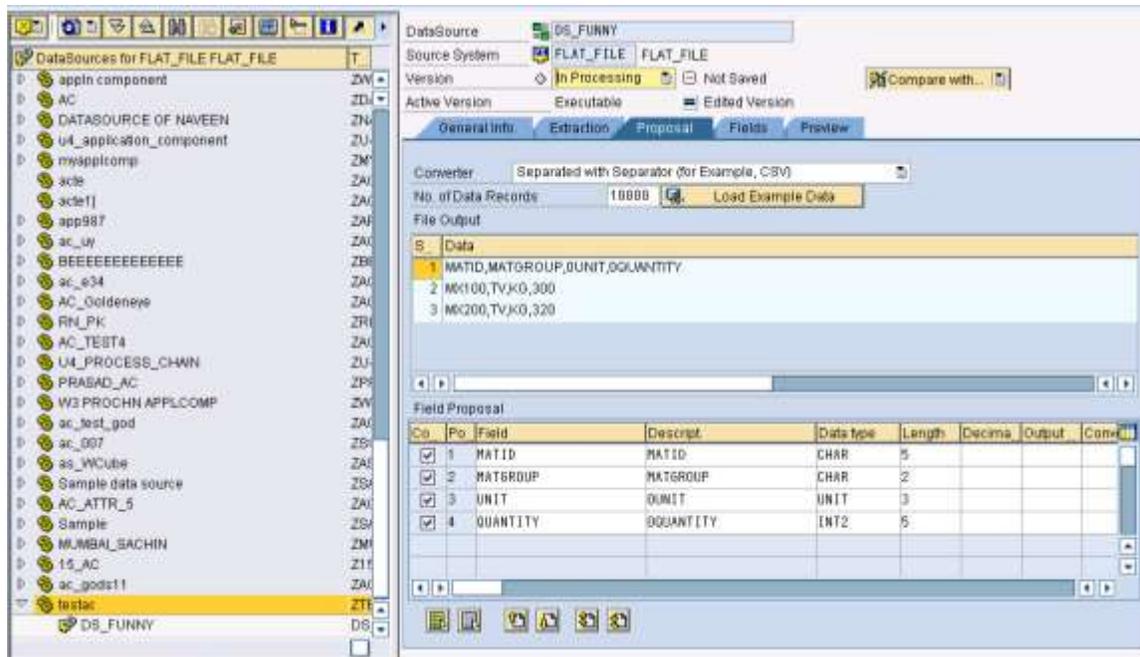


Fig. 1.11 - Field Proposal: We need to define the field proposal here.

The technical names of the fields are written in order & example data is loaded using 'Load Example Data'

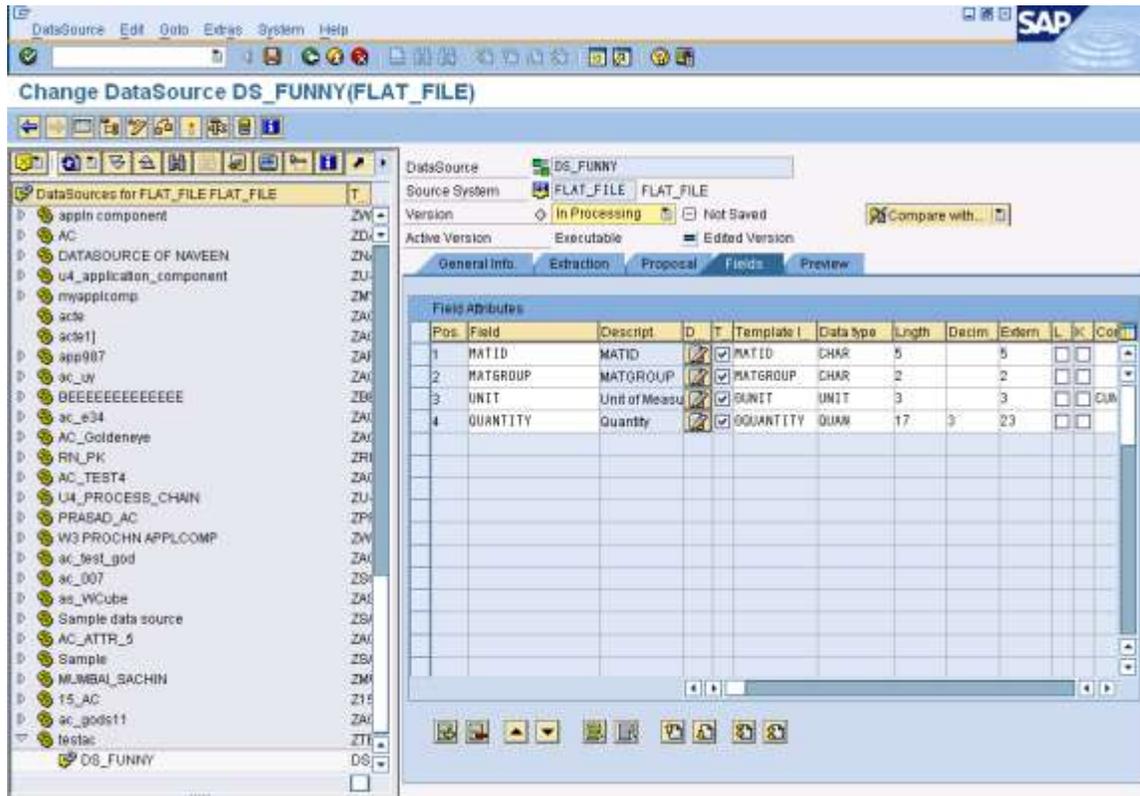


Fig.1.10 - Fields Tab: Under 'Fields' tab, we can now insert any template as necessary.

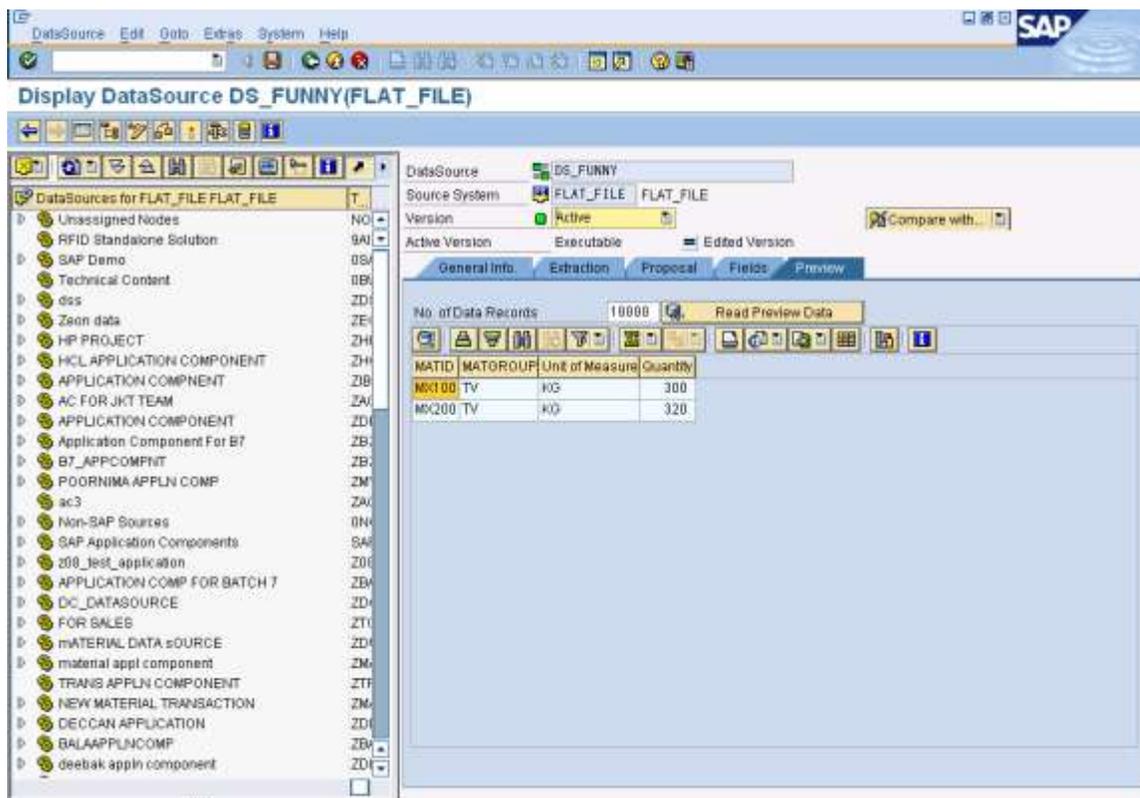


Fig.1.11 - Preview of Data: The sample data is now displayed as we click this tab & choose the option 'Read Preview Data'



Fig.1.12 - Changing the Fields: Here as we want to create our own fields, we choose not to use a template. So, once changes have been done, we save & activate so that DS changes from 'New' to 'Active' state.



Fig.1.13 - Creation of Virtual Provider: Go to RSA1 and choose the Info Area. Right click this Info Area & choose the option 'Create Virtual Provider'

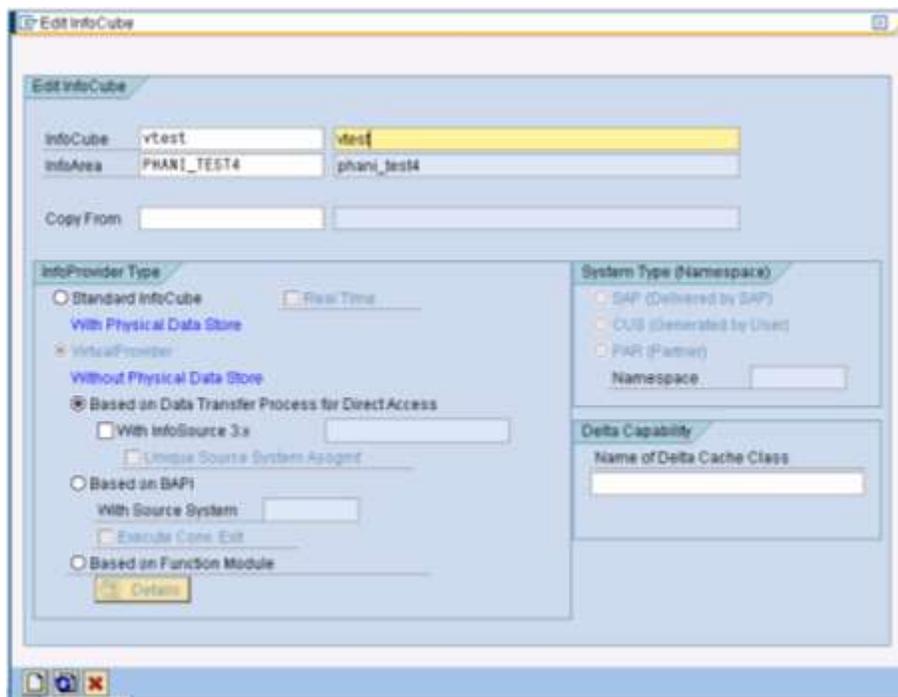


Fig.1.14 -Editing Info Cube: At this step, just enter the details of the info cube, its technical name. Now click Save to create the Virtual Provider.

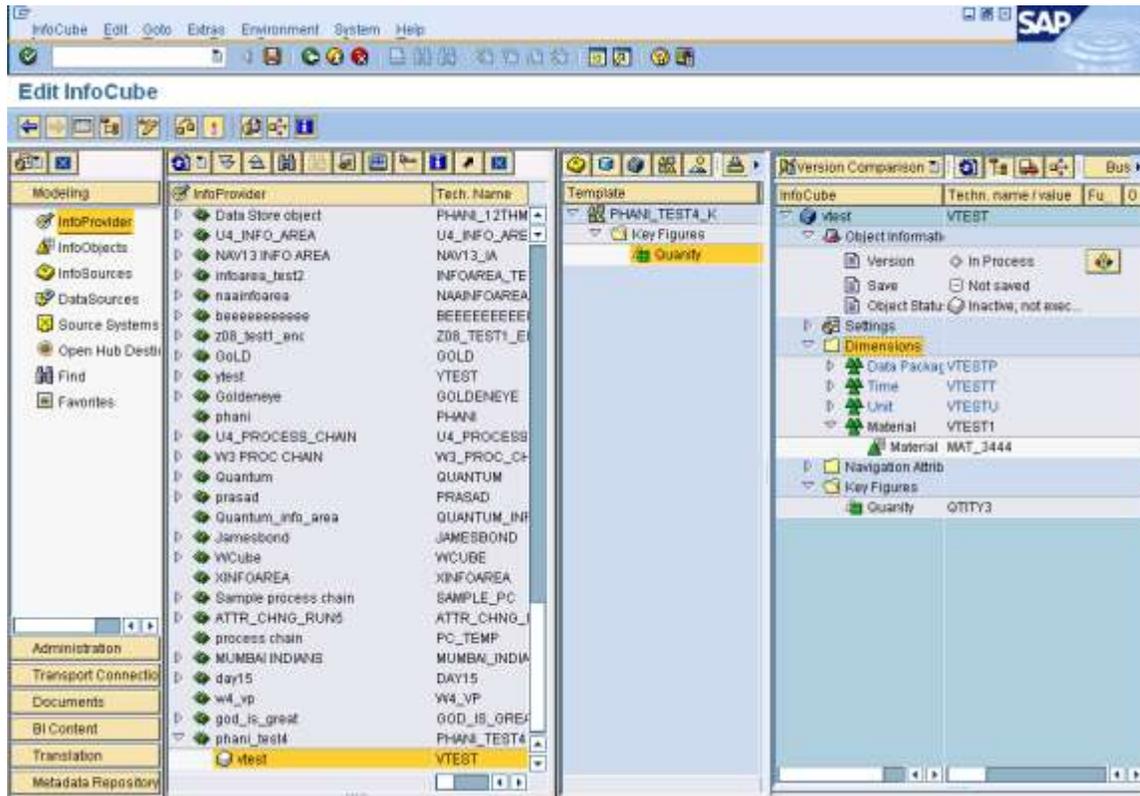


Fig.1.15 - Creating Dimensions: We create 'Material' dimension, insert the 'Material' Info object & use the key figure 'Quantity'

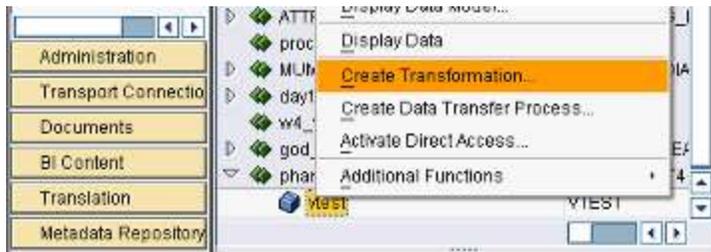


Fig.1.16 - Creating Transformations: We now need to create transformations
So, just right click the Info cube & choose option 'Create Transformation'

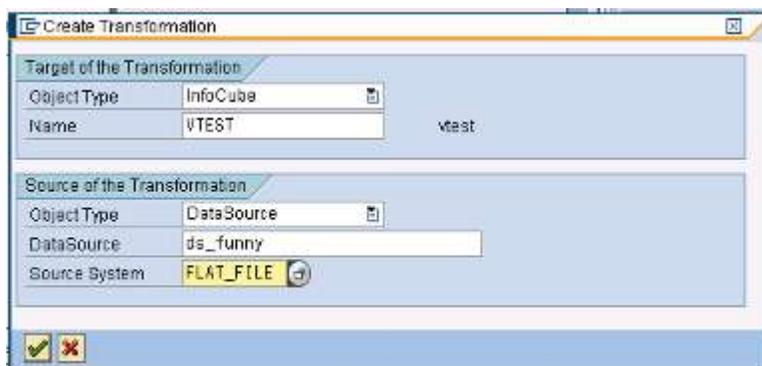


Fig.1.17 - Create Transformation: Specify the source & targets in this dialog box.

The Info cube is the target & source is DS which are a flat file system loaded from desktop etc. via direct access. Once done, press *Enter* to save.

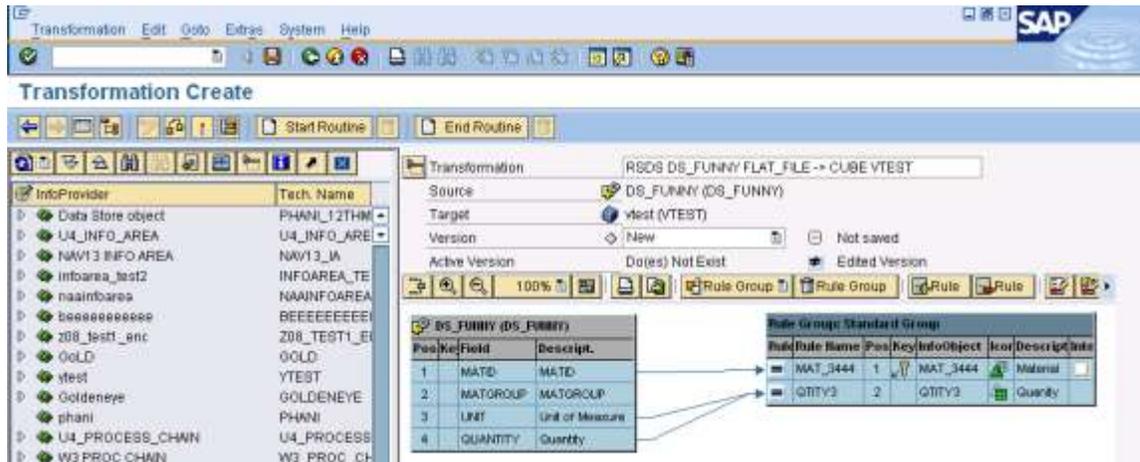


Fig.1.18 - Transformation: Here we can view transformation details & mapping between source and targets.

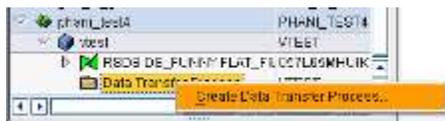


Fig.1.19 - Creation of DTP: At this level, we also need to create DTP.

So, right click on & choose the option 'Create Data Transfer Process'

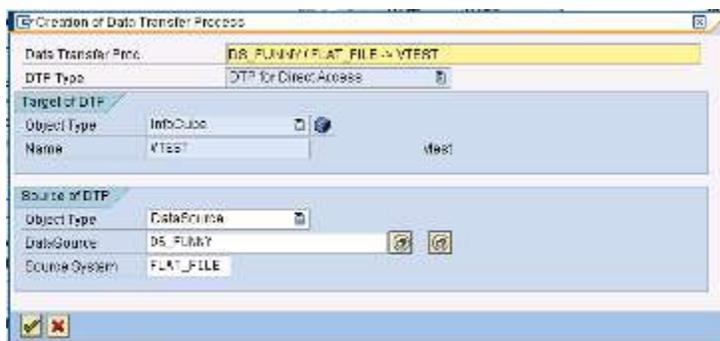


Fig.1.20 - DTP Mapping: We need to specify the source & targets here.

The target being the Info cube & source is flat file as created for transformations earlier.



Fig.1.21 - Activation: Once done, we need to activate the direct update.

So we choose the option 'Activate Direct Access' as we right click the Info Provider

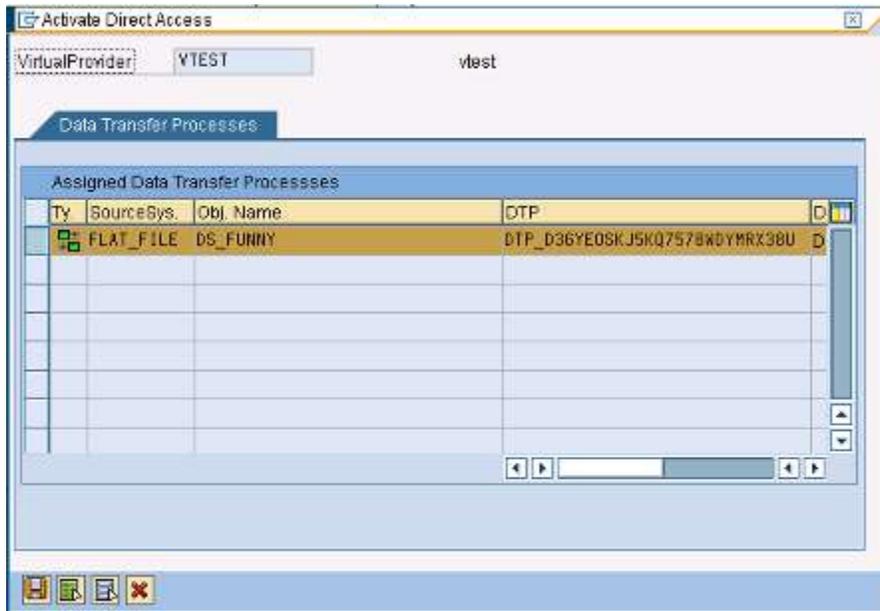


Fig.1.22 - Activation of Direct Access: Click on save to activate the direct access.

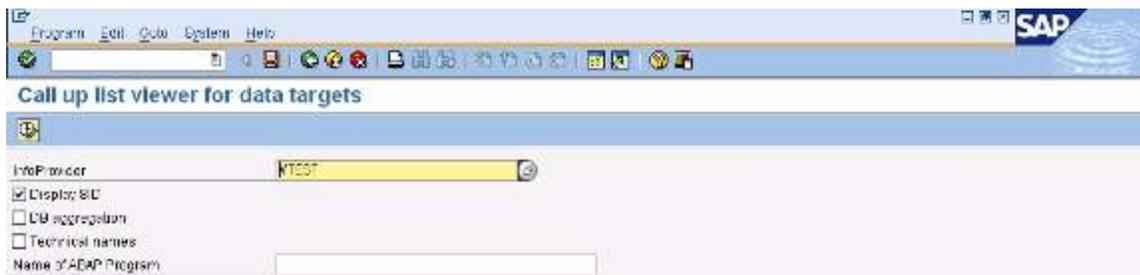


Fig.1.23 - LISTCUBE: Go to Transaction LISTCUBE. Enter the name of the Info Provider (VTEST). Use Field Selection Output: Choose the relevant fields- Material, Unit of Measure, Quantity

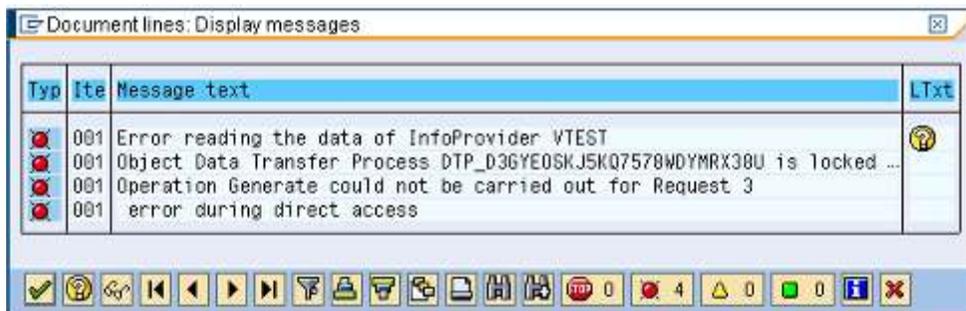


Fig.1.24 - Trouble Shooting: There exists some error reading the Info provider VTEST. We resolve the issue by activating all the objects again & restarting as in Fig 1.30.

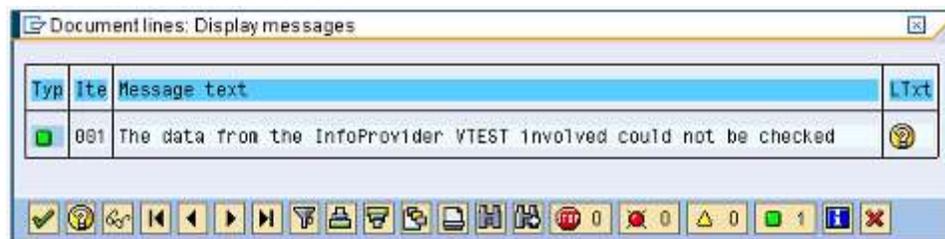
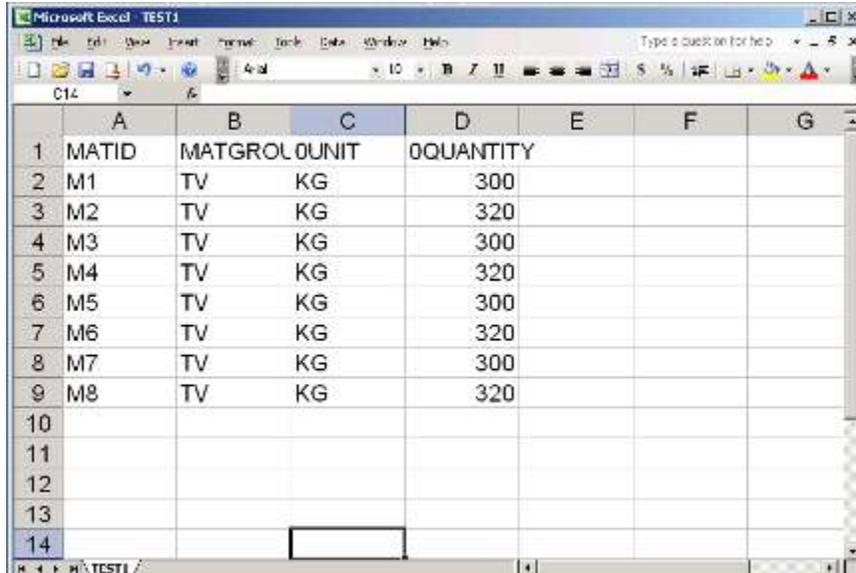


Fig.1.25 - Now, we have a Green to proceed, except the data from the info provider could not be checked.



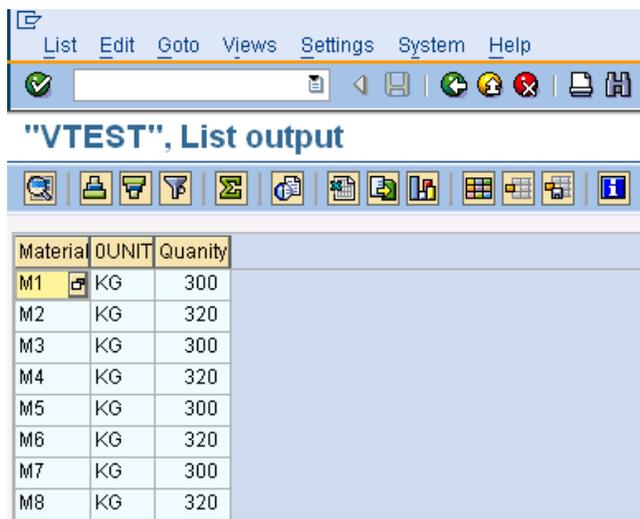
Material	UNIT	Quantity
MX100	KG	300
MX200	KG	320

Fig.1.26 - The records have been successfully extracted from file & loaded to BI System via Direct Access.



	A	B	C	D	E	F	G
1	MATID	MATGRO	UNIT	QUANTITY			
2	M1	TV	KG	300			
3	M2	TV	KG	320			
4	M3	TV	KG	300			
5	M4	TV	KG	320			
6	M5	TV	KG	300			
7	M6	TV	KG	320			
8	M7	TV	KG	300			
9	M8	TV	KG	320			
10							
11							
12							
13							
14							

Fig.1.27 - Let's test really if VIPs work or not, we introduce a change in the CSV file. After doing the changes on the CSV we go to LIST CUBE. Re-visit transaction.



Material	UNIT	Quantity
M1	KG	300
M2	KG	320
M3	KG	300
M4	KG	320
M5	KG	300
M6	KG	320
M7	KG	300
M8	KG	320

The output has been successful.

Implementation Results

Thus, the implementation of the Virtual Providers has been successfully completed in SAP BI 7.0.

Related Content

[Application of SAP BI Info providers](#)

[Multi providers: Technical Details](#)

[BI Reporting - NLS using SAND](#)

For more information, visit the [Business Intelligence homepage](#).

Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.