Applies to:
SAP BW 3.x & SAP BI Net Weaver 2004s. For more information, visit the EDW homepage.

Summary

A semantically partitioned object is an InfoProvider that consists of several InfoCubes or DataStore objects with the same structure. Semantic partitioning is a property of the InfoProvider. We specify this property when creating the InfoProvider. Semantic partitioning divides the InfoProvider into several small, equally sized units. This article describes a step by step approach to create a semantic partition.

Author: Vikram Srivastava
Company: Infosys Technologies Limited
Created on: 09 December 2010

Author Bio

Vikram Srivastava is working as a Technology Analyst with Infosys Technologies Limited. He has got a rich experience on various BW Implementation/Support Projects in both SAP BW 3.5 and SAP BW 7.0.
Table of Contents

Introduction to Semantic Partitioning .................................................................................................................. 3
Use of Semantic Partitioning .......................................................................................................................... 3
Step by Step Approach ........................................................................................................................................... 4
  Create an InfoCube with the property Semantically Partitioned ................................................................. 4
  Step by Step Demonstration ....................................................................................................................... 5
  Create a transformation ............................................................................................................................. 8
  Step by Step Demonstration ....................................................................................................................... 8
  Create a data transfer process .................................................................................................................... 9
  Step by Step Demonstration ....................................................................................................................... 10
  Create a process chain ................................................................................................................................ 15
  Step by Step Demonstration ....................................................................................................................... 15
Conclusion ............................................................................................................................................................... 19
Related Content ...................................................................................................................................................... 20
Disclaimer and Liability Notice .......................................................................................................................... 21
Introduction to Semantic Partitioning

A semantically partitioned object is an InfoProvider that consists of several InfoCubes or DataStore objects with the same structure.

Semantic partitioning is a property of the InfoProvider. We specify this property when creating the InfoProvider. Semantic partitioning divides the InfoProvider into several small, equally sized units (partitions).

Use of Semantic Partitioning

A semantically partitioned object offers the following advantages compared to standard InfoCubes or standard DataStore objects:

- Better performance with mass data:

  For DataStore objects, SAP recommends that we should not save more than 20 million data records. Otherwise the activation process takes considerably longer. For InfoCubes, SAP recommends not to save more than 200-400 million data records. Otherwise the compression and reconstruction of aggregates takes too long. The semantically partitioned object does not have these implicit restrictions, as the data quantity is distributed between several data containers.

- Close data connection:

  Error handling is better. If a request for a region ends with an error, for example, the entire InfoProvider is unavailable for analysis and reporting. With a semantically partitioned object, the separation of the regions into different partitions means that only the region that caused the error is unavailable for data analysis.

- Working with different time zones:

  EDW scenarios usually involve time zones. With a semantically partitioned object, the time zones can be separated by the partitions. Data loading and administrative tasks can therefore be scheduled independently of the time zone.

- Analysis and reporting

  We can use the semantically partitioned object for reporting and analysis, as we do with any other InfoProvider. We can also choose to only update selected partitions in an InfoCube, for example, or include selected partitions in a MultiProvider and use them for analysis, as shown in the following graphic:
Step by Step Approach

Create an InfoCube with the property Semantically Partitioned.

The semantically partitioned object consists of known objects, an InfoCube or a DataStore object. The semantic partitioning creates several objects with the same structure - the partitions. Here a wizard is available to help us minimize the required effort. We define the template for the partitions (the reference structure). The partitions are identical and are derived from the structure. We can only create and change the reference structure. The partitions are write-protected to make sure that they remain identical.

To keep the process of creating a semantically partitioned object as simple as possible, different objects are generated when the object is activated. The following graphic shows the objects that are generated (upper area) and the objects that we need to create ourselves (lower area).
Part of data flow is generated by data flowing out of the partitions. Here an InfoSource is created as well as simple dummy transformations that are not executed by the DTP. Part of the data flow is also created by data flowing into the partitions: An InfoSource is also generated with simple dummy transformations. This InfoSource represents a data entry layer for all partitions and makes it easier for us to connect sources.

A semantically partitioned object can only be transported as a whole object. The generated objects are not transported. Instead they are generated in the target system.

Step by Step Demonstration

1. Go to RSA1. In the InfoArea, select Create InfoCube or Create DataStore Object.

2. Make the required entries. We can only select Standard as the type for InfoCubes.

3. Choose the property Semantically Partitioned. The system now automatically creates an "envelope" (for the semantically partitioned object) in which the different objects are merged. A screen appears where we can define the semantically partitioned object. The right area of the screen displays the
InfoProvider definition. The left area of the screen contains a wizard that helps us to create the required objects.

We define the InfoCube or DataStore object. In doing so, we define the reference structure for the partitions. (This reference structure is a cube with name ZXXXX00 where ZXXXX is the technical name of the cube. The technical name of the cube should have length should be between 3 and 6)

4. Choose Maintain Partitions in the wizard. The reference structure is automatically saved.

5. Select the partition criteria. All characteristics are allowed for an InfoCube but only key fields are allowed for a DataStore object. (We can select a maximum of five partition criteria.)
6. Define the partitions. A maximum of 99 partitions can be created. Here we can define individual values, intervals and conditions. Choose Add Partition to create more partitions. The partitions are automatically given a name which we can change if required.

7. Check and save the partitions. When the check is performed, the system makes sure that the partitions do not overlap.
8. In the wizard, choose Start Activation. The objects are generated and a log is displayed.
9. Check in RSA1, the flow appears as shown below.

Create a transformation

Step by Step Demonstration

1. Click on Create transformation

2. Enter the source for the transformation and click on OK.
3. Specify the mappings for the transformation and activate

4. Transformation icon becomes green

Create a data transfer process.

The data transfer process is not part of the semantically partitioned object but the wizard still helps us to create the data transfer processes.
Templates make it easier to create several DTPs of the same type. A DTP template contains default parameters that can be used to create DTPs for the partitions. We can create DTP templates so that we can quickly and easily create a large number of DTPs with the same default settings.

**Step by Step Demonstration**

1. In the wizard, choose *Create Data Transfer Process*. If we want to use our own template: Create a folder in the *DTP Templates* area.
2. We create our own template in the folder. Choose [Template (Create New DTP Template)]

3. Enter a description for the template and choose [Next].
4. In the *Extraction* area, we can select the extraction mode, package size and key date for master data.

In addition, we can specify the tables from which the data is extracted.

In the area *DTP Filter*, we can choose between *Create Automatically* or *User-Defined*: 
If we choose the option *Create Automatically*, the system generates a filter, based on the partition properties, when the DTPs are generated. The filter ensures that only data relevant for the partition is loaded.

If we choose the option *User-Defined*, no filter is created. In this case, we have to make sure that the data is filtered in the source or in the transformation. We should expect longer loading times here. It may also be the case that the system chooses *User-Defined* automatically, if the filter cannot be generated.

5. Choose (Save DTP Template).

6. We select the partitions that we want to assign the DTP and assign it.

7. Drag and drop the template onto the partitions.

8. Choose Generate Start (DTP Generation). The DTP is now generated from the template.
9. These DTP’s have automatically got filters assigned to them
Create a process chain.

The process chains are not part of the semantically partitioned object but the wizard still helps us to create the process chains.

Step by Step Demonstration

1. In the wizard, choose Create Process Chain. The screen is now divided into three areas: The left screen area contains an overview of our DTPs. The upper right area shows the process chains that have been generated and in the lower right area, we can create the process chains in the detailed view.
2. In the overview, we select the DTPs that we want to run together in a process chain. Choose Add in the detailed view.

3. The system suggests a path and a sequence, which we can change if necessary. If we assign several DTPs to a path, the DTPs are executed in the specified sequence. If we select a separate path for each DTP, these DTPs are executed at the same time.

4. Save and generate the process chain. If we choose Generate, we create a start process.

5. Select a scheduling option and choose Create.
6. We save our entries and choose Back. The process chains are now displayed in the *Generated Process Chains* screen area. In the process chain maintenance screen (transaction RSPC), these process chains appear under the node *Created by semantically partitioned object*. 
7. To execute the process chains, we need to open the process chain maintenance screen. To do this, choose 📄 Process Chain Maintenance in the detailed view.

8. Choose 🕒 Schedule. The process chain is executed according to the settings.
9. All the icons turn green as shown below.
Conclusion

We have now created a semantically partitioned object with a data flow. In the editing screen of the semantically partitioned object, choose Display Monitor to see an overview of the status of the partitions and the DTPs. The displayed requests status tells us whether the requests are active and whether all the data is up to date. This last point means that the system checks whether the last request to be retrieved in every partition is the same, or whether one of the partitions contains a newer request than in the other partitions.
Related Content
www.help.sap.com
www.sdn.sap.com
www.service.sap.com
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.