SAP BW Columnstore Optimized Flat Cube on Microsoft SQL Server

Applies to:
SAP Business Warehouse 7.4 and higher running on Microsoft SQL Server 2014 and higher

Summary
The Columnstore Optimized Flat Cube is an alternative cube design that is available as of SAP BW 7.4 SP 8 on Microsoft SQL Server. It uses the same simplified database structures as the HANA Optimized Cube, such as:

- One fact table only (no separate f-fact and e-fact tables)
- No dimension tables, which results in simplified BW queries and reduced disk space usage
- Simplified administration and improved performance for Inventory Cubes

Since the Columnstore Optimized Flat Cube requires a writeable columnstore, it is only available as of SQL Server 2014.

Author: Martin Merdes
Company: Microsoft
Created on: March 2015

Author Bio
Martin Merdes is a Senior Software Engineer at Microsoft. Since 1996 he has been working in different technical functions for Microsoft at SAP Germany: Support Engineer, Senior Consultant, Technical Account Manager, Premier Field Engineer and Software Development Engineer. Since 2009 he is responsible for the porting of SAP BW on Microsoft SQL Server.
# Table of Contents

Overview ....................................................................................................................... 3

Prerequisites .................................................................................................................. 4
- SQL Server ................................................................................................................ 4
- SAP BW ...................................................................................................................... 4

Using the Flat Cube ....................................................................................................... 4

Benefits .......................................................................................................................... 5

Creating a Flat Cube ...................................................................................................... 6

Checking cube type ....................................................................................................... 6

Converting a Flat Cube ................................................................................................. 7

Related Content ........................................................................................................... 9

Copyright ..................................................................................................................... 10
Overview

The Columnstore Optimized Flat Cube is an alternative cube design, which is available as of SAP BW 7.4 SP 8 on Microsoft SQL Server. It uses the same simplified database structures as the HANA Optimized Cube.

Data Warehouse systems typically store data in a Star Schema. This has always been the case also on non-SAP systems. A Star Schema consists of dimension tables grouped around a central fact table (resulting in a star):

**Classic Star Schema (Non-SAP)**

![Classic Star Schema Diagram]

For performance reasons, there is typically a compound index on all dimensions of the fact table. This restricts the maximum number of dimensions of a cube to the maximum number of columns in an index. For rowstore indexes, the SAP Data Dictionary allows a maximum of 16 columns. Therefore, SAP did not use this simple star schema in SAP BW.

**SAP BW Extended Star Schema (rowstore optimized)**

![SAP BW Extended Star Schema Diagram]

A dimension in a data warehouse, is called a characteristic in SAP BW. Since 16 characteristics are typically not sufficient for an SAP BW Cube, an additional layer was introduced: SAP BW Dimensions. By grouping characteristics in dimensions, an SAP BW cube can include hundreds of characteristics. However, the additional dimension layer results in additional joins for SAP BW queries. A dimension having only one characteristic can be defined as a Line Item Dimension in SAP BW. In this case SAP BW directly joins the characteristic with the fact tables (as for characteristic C5 in the example below). For improving query performance in the rowstore, SAP BW contains a second adaption of the Star Schema: two fact tables. While the f-fact table is optimized for data load, the e-fact table is optimized for reporting. The Star Schema adopted in SAP BW is often called SAP BW Extended Star Schema.

**Flat Cube (columnstore optimized)**

![Flat Cube Diagram]

The optimizations of the SAP BW Extended Star Schema were introduced for optimizing the rowstore. When using the columnstore they are no longer necessary. There is no limitation on the number of index columns for a columnstore index. The clustered columnstore index contains all columns of the fact table. Therefore, SAP retired the Extended Star Schema starting with HANA. The resulting schema looks like a typical Star Schema, but SAP found another name for it: HANA Optimized Flat Cube. For other database platforms the new schema is simply called a Flat Cube. Actually, the Flat Cube still contains a single dimension: the packet dimension. It is used for fast (logical) request deletion. However, all other dimensions are gone with the Flat Cube.

Unfortunately there is no consistent naming convention in the SAP community for cube schemas:

- The newest schema is called Columnstore Optimized Flat Cube, HANA Optimized Cube or simply Flat Cube. Indeed it is identical with the classical Star Schema outside SAP. The only difference is the naming convention: What is called a dimension outside SAP is called a characteristic in SAP BW.
- Since the SAP BW Extended Star Schema has been used in SAP BW for more than a decade, it is often also called Classical Star Schema or simply Star Schema. To distinguish from a Flat Cube, it is sometimes also called Non-Flat Cube
Prerequisites

The Columnstore Optimized Flat Cube requires additional SAP BW code. SAP Note 2114876 always contains the up-to-date prerequisites. The documentation and known issues are listed in SAP Note 2116639. Additional code corrections are maintained in SAP Note 2091520.

As at March 2015, you have to install the following software components:

SQL Server

The minimum required SQL Server version is Cumulative Update 5 (CU 5) of SQL Server 2014. However, we recommend installing the newest CU and service pack, once available. You can download the latest SQL Server patches from http://technet.microsoft.com/en-us/sqlserver/ff803383.aspx.

SAP BW

The Flat Cube is fully supported as of SAP BW 7.4 SP 10. It also runs on SAP BW 7.4 SP 8 and SP 9 if the correction instructions of SAP Note 2029797 have been applied. In this case you can convert Non-Flat Cubes to Flat Cubes and vice versa. However, you cannot directly create a Flat Cube. The checkbox Flat Infocube in the dialog box Create Infocube is only available as of SAP BW 7.4 SP10.

Using the Flat Cube

Once you have created a Flat Cube, you can use it just as a Non-Flat Cube. There is no need to change existing BW queries or BW process chains. However, there are a few restrictions:

- Aggregates are not possible for a Flat Cube
- A BIA index is not possible for a Flat Cube
- BW partitioning (of the e-fact table) is not possible
- BW Real-Time Cubes are not supported for Flat Cubes on Microsoft SQL Server

You can decide cube-by-cube which cube type to use. You can convert a Non-Flat Cue to a Flat Cube and vice versa. If you see the need for aggregates on a particular Flat Cube, you can convert this cube back to a Non-Flat Cube.

Although there is no e-fact table any more, you can still use BW cube compression. This also reduces the number of rows for a Flat Cube. The Reference Points of Inventory Cubes are also only available, if you perform BW cube compression.

Since the Flat Cube is using the writeable columnstore index, you should follow the same recommendations as for Non-Flat Cubes with writeable columnstore index, in particular:

- Keep database index statistics up-to-date (by including the process chain type “Construct Database Statistics” in the BW process chains containing a Data Transfer Process)
- Perform columnstore rowgroup compression (by including the process chain type “Generate Index” in the BW process chains containing a DTP)
Benefits

The absence of the dimension tables in a Flat Cube has many advantages

- **Reduced disk space requirements**
  In SAP BW there are several indexes on the dimension tables. On Microsoft SQL Server, the dimension tables always use b-tree indexes, even for columnstore cubes. Therefore the space usage of the dimension tables is significant. When using the Flat Cube you can save the disk space for the dimension tables.

- **Improved query performance**
  With the Flat Cube an SAP BW query might still contain some joins (for attributes and hierarchies), but the joins for all dimension tables (except the packet dimension) are gone. Therefore, queries are simpler and faster, compared with Non-Flat columnstore cubes.

- **Improved data load performance**
  Since there is no need to create DIMIDs, the data load into a Flat Cube is typically faster compared with a Non-Flat Columnstore Cube. However, this is not always the case. When re-loading the same data into a Non-Flat Cube, there is no need to create DIMIDs. Furthermore, the fact table of a Flat Cube contains more columns: one column per characteristic instead of one column per dimension.

The disk space usage of an SAP BW cube depends on many factors: The used cube type, number of dimensions, characteristics, and key figures. The following diagram shows the total disk space usage (fact table and dimension tables) of a sample cube using different cube types:

1. BW Cube using rowstore with no database compression
2. BW Cube using rowstore with PAGE compression
3. BW Cube using read-only columnstore on e-fact table
4. BW Cube using columnstore on f-fact and e-fact table
5. Flat Cube

The sample cube contains 100,000,000 rows (with 90% in the e-fact table). The cube has 38 BW Characteristics, 11 BW Dimensions and 10 BW Key Figures
Creating a Flat Cube

As of SAP BW 7.4 SP 10 you can choose the Flat Cube schema when creating a new cube simply by selecting the check box *Flat InfoCube*:

If you want to create a Real-Time Cube, you have to select the check box *Real Time* in the dialog window above. However, it is not recommended to create a Real-Time Flat Cube: **Do not select the check boxes Real Time and Flat InfoCube at the same time.**

In a Real-Time Cube, data is not loaded by means of BW process chains. Therefore there is no way to include the columnstore rowgroup compression during data load. This results in open columnstore rowgroups and poor query performance. You should convert any accidentally created Real-Time Flat Cube to a Non-Flat Cube as described below.

Checking Cube Type

In transaction RSA1 you can easily figure out whether an existing cube is a Flat Cube or not. A Flat Cube has the cube subtype *Flat InfoCube*:
Converting a Flat Cube

You can use the BW repartitioning framework for converting a Non-Flat Cube to a Flat Cube and vice versa: In SAP transaction RSA1, right-click the cube and choose Additional Functions → Repartitioning in the context menu.

The conversion is not simply an index creation. DIMGIDs have to be converted to SIDs. Depending on the hardware resources, this can take several hours for cubes having billions of rows. On our test system running on Microsoft Azure it took less than an hour for a cube having 100,000,000 rows.

The Flat Cube conversion always runs as a batch job. This job calls dialog work processes using RFCs. By default, only 3 dialog processes are used. You can decrease the time needed for the conversion by configuring (many) more dialog processes using RSADMIN parameter QUERY_MAX_WP_DIAG. The same parameter also impacts BW repartition runs. However, it generally does not help to increase QUERY_MAX_WP_DIAG if there are not enough free dialog work processes.

Before you start the conversion to an Inventory Flat Cube, you have to confirm that all Historical Movements are already compressed:

The reason for this check is the following: For Inventory Cubes you have to load Historical Movements in a separate request. The data transfer process (DTP) is identical for Historical Movements and normal transactional data. You only have to make sure that requests containing Historical Movements need a special option for BW cube compression: No Marker Update.
This has changed for Flat Cubes. There is no longer a special option in BW cube compression. Instead, requests with Historical Movements are already marked during DTP:

Due to this functional change, it is not allowed to convert a Non-Flat Cube to a Flat Cube as long as there are uncompressed Historical Movements in the cube.

A Flat cube uses fixed request IDs 0, 1 and 2. Therefore, you cannot convert a Non-Flat Cube to a Flat Cube, if request ID 2 is already used. In this case the cube conversion asks you to compress the first request (with ID 2) and restart the conversion again.

The conversion to Flat Cube runs in separate steps. The original cube is not touched before the step SET_READ_LOCK is reached. If there is any failure in any step before, you can either restart the conversion request or simply delete it. Be careful: This is not the case for any step after SET_READ_LOCK or for any step in the conversion to Non-Flat Cube. Deleting such a request could result in data loss!
Related Content


SAP Note 2116639 SQL Server 2014 Columnstore documentation

SAP Note 2114876 Release Planning SAP BW for SQL Server Columnstore

SAP Note 2029797 Support for Flat Cube on Microsoft SQL Server

SAP Note 1949486 SQL Server 2014 column-store support for SAP BW


