Configuration and Utilization of the OLAP Cache to Improve the Query Response Time

Applies to:

SAP NetWeaver BW 7.0

Summary

This paper outlines the steps to improve the Query response time by using the BW OLAP Cache. The response time will show tremendous improvement by accessing the Query result set data from the OLAP Cache than from the DataBase. This paper also includes both the recommended settings for the various parameters involved within each section and general recommendations for Cache performance.

Author: Gopalakrishnan Venkatachalam

Company: Global Softech Inc

Created on: 20 March 2010

Author Bio

Gopalakrishnan Venkatachalam is working as an SAP BW Consultant. He is a certified SAP BW Professional with experience in Data Warehousing and Reporting.

Table of Contents

Purpose

This paper outlines the steps to improve the Query response time by using the BW OLAP Cache. The response time will show tremendous improvement by accessing the Query result set data from the OLAP Cache than from the DataBase. This paper includes both the recommended settings for the various parameters involved within each section and general recommendations for Cache performance at the end. It also describes the settings that can only be modified by the BASIS and those that can be made at the InfoProvider and the query level by Developers and Analysts.

Prerequisites

a. OLAP Cache should have been activated. To check that this has been done, please see Global Cache Settings below.
b. BW 7.0 and above must be installed.

OLAP Cache Overview

The OLAP cache is architected to store query result sets as highly compressed cluster data and to give all users on an application server access to those result sets. If the same query (or a subset) is then executed by another user, the subsequent query request can be filled by accessing the result set already stored in the OLAP cache. In most cases, the OLAP cache offers significant performance gains over other techniques such as aggregates when it is configured correctly. Moreover, its benefits can be expanded if qualifying queries are enabled to fully access cached data. Storage options are available for query results and can be optimized to best meet the needs of the system as well as for the individual queries. It must be maintained properly along with the queries that use it for the OLAP cache to reach its fullest potential.

There are two types of OLAP cache. They are:

1. **Cross-transaction cache (aka global cache) -** Query results are stored on the application server instance.
2. **Local Cache -** Query results are stored in the roll area for each session.

Irrespective of this, the global cache also shares some properties of the local cache. For instance, data is similarly retained in the roll area as long as it is required by the OLAP processor. On the other hand, the local cache size also has significance in the global cache. With both types of cache, cache objects that are no longer needed are deleted from the roll area if the local cache size is exceeded. The focus of this article is primarily on the Global Cache.

OLAP Cache Settings

There are a number of places where the settings for the OLAP cache usage can be set. The following is a list of the various levels where OLAP cache performance can be enhanced. This will be followed by a detailed description on how each of these settings can be displayed and/or changed.

1. **Global:** *BASIS ONLY SETTING* – This setting ensures that all queries use OLAP cache by default, but the settings can be overwritten at both the InfoProvider and/or query level. Global Cache Settings shows how to verify this. In some cases, it may be necessary to switch OLAP cache off. Please refer Switching off OLAP Cache section to determine how to do this.
2. **InfoProvider:** This setting ensures that all queries using the InfoProvider will initially default to the settings of the InfoProvider. These settings can be overwritten at the Query level. Please refer InfoProvider Level Cache Settings section to determine how to do this.
3. **Query:** Any settings made at this level only apply to that specific query. One can, for example, choose not to use cache at this level which would only impact the given query. Please refer Query Level Cache Settings section to determine how to do this.

Global Cache Settings

The global activation of cache should be performed by your BASIS team and should not be changed by any other person (recommended). It is, however, a good idea to check that this has been done before going ahead with the settings at a query level. If a check is done and the green light is not turned on beside Cache Act., please contact your BASIS team.

Note: The pre-requisite for the OLAP Cache is that the SHM Memory should at least be equal to the Global Memory Cache (OSS Note: 656060). The parameter is **rsdb/esm/buffersize_kb** (Size of the ESM buffer) and can be set using Transaction code RZ11.

To check the OLAP Cache options, execute transaction code **RSRCACHE**. The green light next to Cache Act should be shown. Please note that the Local and Global Cache Sizes may be different from those below. The best values for these settings are determined by the BASIS team and should not be modified.

These settings are set using Transaction code RSRCACHE (Cache Parameter button), RSCUSTV14 and also using Transaction code SPRO (Menu Path: SAP NetWeaver → Business Intelligence → Performance Settings → Global Cache Settings).

The persistence mode in the above screenshot is set when Cache Mode 2, 3 or 4 is used.

InfoProvider Level Cache Settings

OLAP relevant settings can be changed at the InfoProvider level by those that have the relevant authorization levels. *Read Mode*, *Cache Mode, Persistence Mode* and *Delta Cache* can be set up at this level. Whenever a query is created, the initial settings will default to the InfoProvider level settings. These can be displayed or changed by executing the transaction code *RSDIPROP.* RSDIPROP can also be reached while in MultiProvider, InfoCube, DSO, or InfoSet maintenance by following the menu path: **Environment → InfoProvider Properties.** For InfoObjects, this can be accessed only by executing

transaction **RSDIPROP** directly. Note that even if the server has had cache globally activated, it can be deactivated at either the InfoProvider or Query level by setting the *Cache Mode* to 0.

The following screen shot shows the various cache relevant settings that can be made at the InfoProvider level by executing the transaction code RSDIPROP.

Read Mode

The read mode determines how the OLAP processor gets data during navigation. Three values are supported:

- **H - Query to be read when you navigate or expand hierarchies**

  The data across the hierarchy is aggregated and transferred to the OLAP processor on the hierarchy level that is the lowest in the start list. When expanding a hierarchy node, the children of this node are then read. The amount of data transferred from the database to the OLAP processor is the smallest in this mode; however, it has the highest number of read processes. This is the optimal setting recommended by SAP.

  Recommendation: Use it for Large Hierarchies

- **X - Query to read data during navigation**

  The OLAP processor only requests data that is needed for each navigational status of the query in the Business Explorer. The data that is needed is read for each step in the navigation. In contrast to the Query to be read when you navigate or expand hierarchies mode, presentation hierarchies are always imported completely on a leaf level here. The OLAP processor can read data from the main memory when the nodes are expanded. When accessing the database, the best aggregate table is used and, if possible, data is aggregated in the database.

  Recommendation: Use it for small hierarchies and for large result set

- **A - Query to read all data at once**

  There is only one read process in this mode. When you execute the query in the Business Explorer, all data in the main memory area of the OLAP processor that is needed for all possible navigational steps of this query is read. During navigation, all new navigational states are aggregated and calculated from the data from the main memory.

  Recommendation: Use it for small InfoCubes and for queries with few free characteristics

Cache Mode

The five different modes for *Cache Mode* are shown below. The values are the same whether they are at the InfoProvider or Query level:

- **0 Cache Is Inactive**

  This effectively switches the Global Cache off for the given object which means that the result set for that given object will not be cached in the global cache. However, the local cache will still be used, but the result set will only be available in the specific user session.

- **1 Main Memory Cache Without Swapping**

  When the cache reaches the maximum size, the cache can no longer be updated. The least recently used Cache entry is deleted from the Cache memory using the LRU Algorithm.

- **2 Main Memory Cache with Swapping**

  When the cache reaches the maximum size, the oldest cached data is swapped to a background store (which can be a Flat file, Cluster or BLOB Table) and new data can be added.

- **3 Persistent Cache per Application Server**

  Useful only if you have one application server or you have the users who are executing the same reports allocated to the same application server (User Groups or Load Balancing).

- **4 Persistent Cache Across Each Application Server**

  Data is stored on a central DB and shared with all application servers. This setting is recommended for a BW landscape with multiple application servers.

- **5 BLOB/Cluster Enhanced**

    Data is stored in an enhanced BLOB/Cluster table. Avoids a central Directory and the locks associated with the Enqueue concept

Persistence Mode

The four different values for *Persistence Mode* are shown below. The values are the same whether they are at the InfoProvider or Query level:

- **Inactive** - Global cache turned off (By default for Cache Mode 1)

- **Flat File** - Cache stored as flat file

    Recommendation: For small result result set when file is located as close to the Application server as possible

- **Cluster table** - stores data as a cluster table in the database

    Recommendation: For medium and small result sets (< 1MB)

- **Transparent table (BLOB)** - faster than cluster tables on large query result sets. OSS note 683194 gives further details on the advantages of using BLOB Tables.

    Recommendation: For large result sets (> 1 MB) due to more efficient DB Operations

Delta Cache

This setting is applicable for InfoCubes and MultiProviders containing at least one InfoCube. This can be activated by checking the available box. When checked, a query will read data from the OLAP Cache as far as possible. New requests that have been loaded will be read from the InfoCube's (F) fact table and will be updated into the OLAP cache. Since the Delta cache is based on Request ID, compression of the requests in the F Fact table will invalidate the OLAP Cache and the cache has to be built up completely. Note that the default is for this option to be checked.

Recommendation: Use it only for InfoCubes that are not compressed frequently.

Query Level Cache Settings

The Query properties can be changed at the Query Monitor tool which is accessed using the Transaction Code RSRT. When a Query is created, the settings of *Read Mode, Cache Mode,* and *Persistence Mode* will default to those set at the InfoProvider level. There are some additional settings which can be made at this level. To display or change the property of an individual query, execute transaction code **RSRT** and then follow the steps below:

1. Enter the Query name or select the query name from the drop down list which displays the InfoProvider name and Query Name.

2. Click Properties. The new pop-up Query Properties opens.

3. Change the settings if required.

4. Select ✔ or ✖ dependent on whether changes have been made.

The following screenshot illustrates the above steps and the other settings are explained below that.

Request Status (aka Data Integrity)

This setting only applies to InfoCubes and MultiProviders that contain InfoCubes. The values are:

- **0 All InfoProviders up to a securely accessible status (RQTS0)**

  All data that is rolled up into the aggregate is read. If there is no aggregate all data is used up to the first open request or request with errors. This is the default setting for queries.

- **1 Real-time enabled InfoCubes up to current status**

  In addition to the data for the mode *All InfoProviders up to securely accessible status (RQTS0)*, the data of real-time enabled InfoCubes up to and including the open request is read.

- **2 All InfoProviders up to current status (up to Qualok)**

  This includes the requests of an InfoCube with aggregates that have not yet been aggregated, but which come before the first open request or request with errors.

- **3 All InfoProviders up to the current status (all green requests)**

  All requests without errors (all green requests) are read, as well as those that come after a request with errors or request that is not yet complete.

- **9 All complete InfoProvider (including red requests)**

  All data available is read, in other words, all request with errors, as well as open requests for answering a query.

Recommendation: Setting the value to 0 has the best performance

Use Selection of Structure Elements

This option can be selected for any query, but is only really useful in a couple of cases. In queries, selections are frequently defined on one or more characteristics in the columns, or more precisely, in the structure elements. One or more of these structure elements are often filtered in the BEx Analyzer or in Web applications. If you do not select this option, these dynamic (structure element) selections are not transferred to the database which means that the data for the entire structure or for both structures will be read from the database and not from OLAP Cache.

If those are not as required, these can be modified as shown below. Note that even if the cache has been enabled globally or specific values have been assigned at the InfoProvider level, they can still be changed at the query level. For example, cache usage can be deactivated at the Query level by setting the ***Cache Mode*** to 0.

Calculate w/Packed Numbers

This option relates to the degree of accuracy and determines which number format is used internally for calculating key figures. The degree of accuracy is determined in the InfoObject maintenance for each key figure and in the query monitor for all query key figures. Calculating with packed numbers causes a high processing status overall, so choose this option for those application areas in which a high accuracy is required.

No Parallel Processing

You can turn off parallel processing for a single query. This can be beneficial because the query then uses fewer system resources if you use non-parallel processing. In the case of queries, with very fast response times, the effort required for parallel processing can be greater than the potential time gain. If you choose this option, the selected query is processed sequentially in the future.

Generation Log

When checked, this option allows you to display the corresponding generation log after the query is generated.

Optimization Mode
There are three available values for this. They are:

- **0 Query is Optimized after Generation**
  This is the default value and means that the size of the memory structure, in which data is held, will, be determined by the OLAP processor. It is done after the first execution and again after a period of 31 days since the last optimization.

- **1 Query Optimization with Individual Periods in Days**
  Here you can determine the optimization period yourself.

- **9 Query Optimization Inactive**
  No optimization done.

To change the Query properties for all or selected queries of an InfoProvider, we can use the Query Mass Maintenance feature from Query Monitor.

For this, use the menu path RSRT → Environment → Query Mass Maintenance
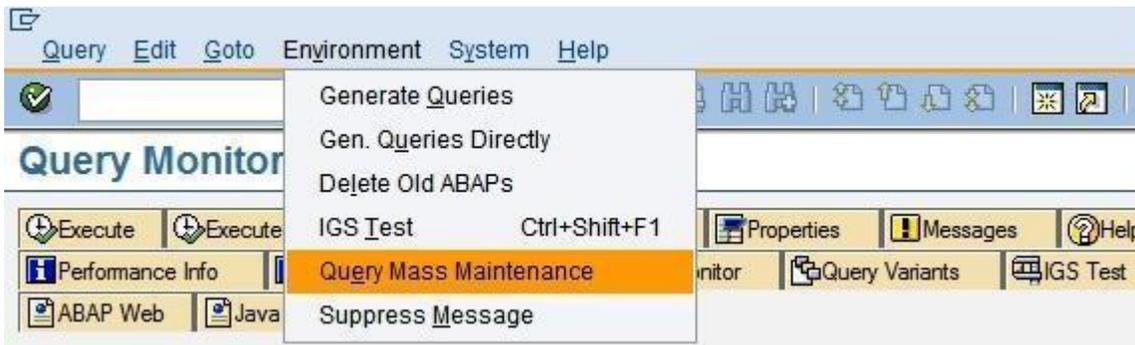
Once you have made the required changes to the settings, click on Generate report to regenerate the

Queries for the new settings to take effect. This will be confirmed by a message.

To view the cache settings for all or selected queries, the RSRREPDIR table can be viewed. The steps are shown below.

1. Execute transaction **SE16.**
2. Enter **RSRREPDIR** for the table name**.**
3. The **CACHEMODE** field below shows the settings of the individual queries. The numbers in this field correspond to the cache mode settings in Cache Mode Settings.

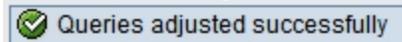| OBJVERS	INFOCUBE	COMPID (Query Technical Name)	READMODE	CACHEMODE	DELTACACHE
A	0IC_C03	0IC_C03_Q0001	H	1	X
A	0IC_C03	0IC_C03_Q0002	H	1	X
A	0IC_C03	0IC_C03_Q0003	H	1	X
A	0IC_C03	0IC_C03_Q0004	H	1	X
A	0IC_C03	0IC_C03_Q0005	H	1	X
A	0IC_C03	0IC_C03_Q0006	H	1	X
A	0IC_C03	0IC_C03_Q0007	H	1	X
A	0IC_C03	0IC_C03_Q0008	H	1	X
A	0IC_C03	0IC_C03_Q0009	H	1	X
A	0IC_C03	0IC_C03_Q0010	H	1	X
A	0IC_C03	0IC_C03_Q0011	H	1	X
A	0IC_C03	0IC_C03_Q0012	H	1	X
A	0IC_C03	0IC_C03_Q0013	H	1	X
A	0IC_C03	0IC_C03_Q0014	H	1	X
A	0IC_C03	0IC_C03_Q0015	H	1	X

Cache Monitoring

Cache monitor is the central monitoring tool for the OLAP Cache. This is accessed using the Transaction code RSRCACHE. This can be used to check if the query was read from the OLAP Cache. The following screenshot shows the layout of the Cache monitor with the various options:
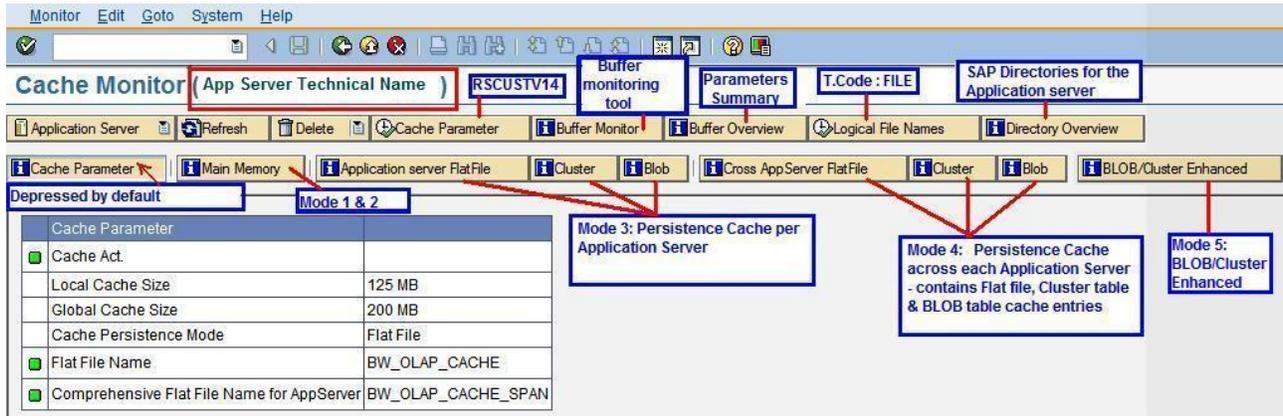
For Cache mode 1 and 2, the Read flag is checked for entries that were read from the Cache.
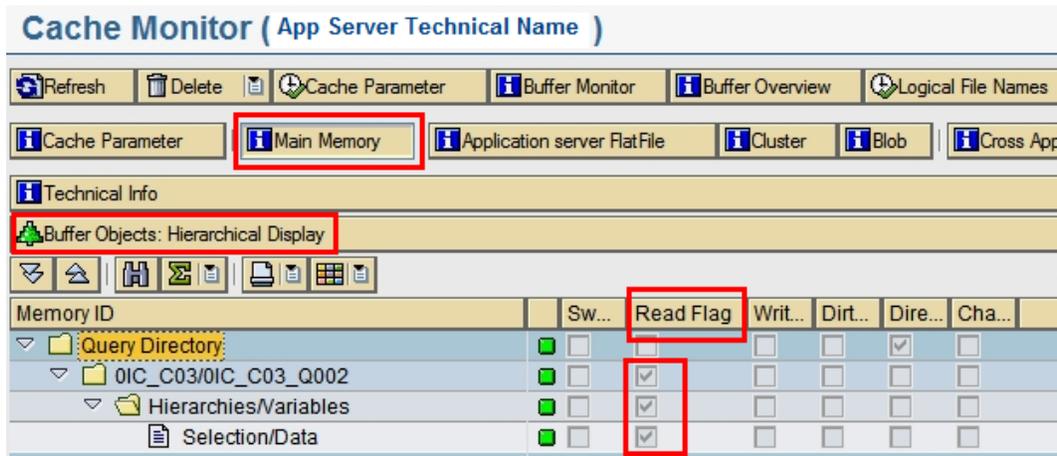
For Cache modes 3, 4 and 5, this can be checked from the context menu and by clicking Detail.
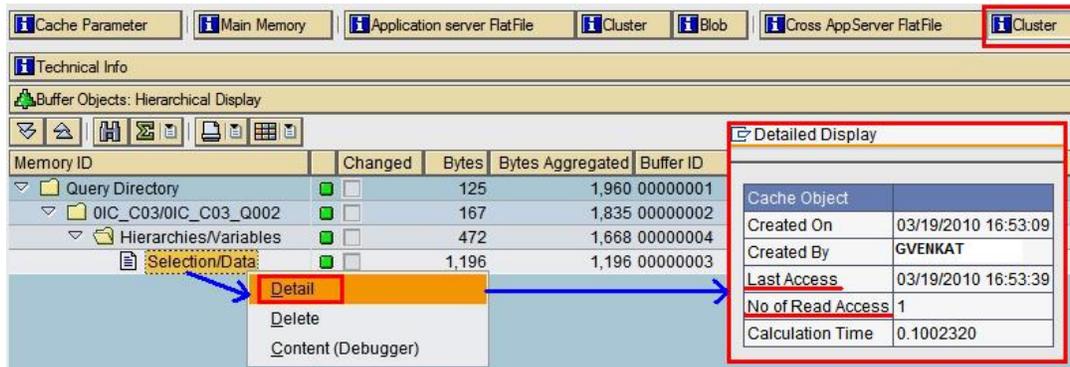
Cache usage can also be checked by selecting ⏵Execute + Debug from the Query Monitor (RSRT) and selecting ☑ Display Statistics Data . If Total DBTRANS and Total DBSEL are present in the events of the Statistics, cache is not being used. This is shown in the following screenshot:

Query Runtime

Query executed for the second time. Data read from the OLAP Cache as shown by the events:
Read Cache & Cache Read Access
The following events are not present as Query doesn't access the DataBase:
1. Total DBTRANS
2. Total DBSEL
3. Data Manager

Query executed for the first time. Data read from the DataBase as shown by the events:
1. Total DBTRANS
2. Total DBSEL
3. Data Manager

Aggregation Layer

..	User	Ha..	Han..	InfoProvid..	Object Name	..	Eve..	Event Text	Duration	Co...	Event C...
1	GVENKAT		DFLT			2		Not Assigned	0.001616	0	1
1	GVENKAT	1	BR...		RRX_SESSION_INITIALIZE	2	10...	RFC call	0.020504	0	1
1	GVENKAT	2	BR...		RRX_REPORT_OPEN	2	10...	RFC call	0.001159	0	1
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	3010	OLAP: Quer...	0.082813	0	1
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	3999	OLAP Other...	0.146976	0	18
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	19...	3.x Query Vi...	0.199496	0	2
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	19...	3.x Analyzer...	4.401644	31,...	2
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	4600	Authorizatio...	0.172061	0	2
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	3510	OLAP: EXIT...	0.013876	0	3
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	3500	OLAP Initial...	0.322915	0	13
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	4300	Value Auth...	0.004234	0	8
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	3000	OLAP: Setti...	1.995560	0	32
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	2505	Read Cache	0.036301	0	4
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	2525	Cache Rea...	0.000000	0	3
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	2500	Cache Gen...	0.023751	0	2
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	3100	OLAP: Rea...	0.018505	2,9...	1
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	3900	OLAP: Rea...	4.398014	0	9
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	3200	OLAP: Data...	1.715001	20,...	1
1	GVENKAT	1	OLAP	0SD_C02	0SD_C02_Q00021	2	19...	3.x Query Vi...	0.096518	0	1
1	GVENKAT	3	BR...		RRX_GRID_CMD_PROCESS	2	10...	RFC call	0.001729	0	1

Eve..	Event Text	Duration	Counter	Eve...
2	Not Assigned	0.001637	0	1
2 10...	RFC call	0.010362	0	1
2 10...	RFC call	0.001094	0	1
2 3010	OLAP: Query Gen.	0.022507	0	1
2 3999	OLAP Other Time	0.083358	0	18
2 19...	3.x Query View Open	0.171696	0	2
2 19...	3.x Analyzer Server	4.409187	31,177	2
2 4600	Authorization Buffer	0.099453	0	2
2 3510	OLAP: EXIT Variables	0.003543	0	3
2 3500	OLAP Initialization	0.097989	0	13
2 4300	Value Authorizations	0.004532	0	8
2 3000	OLAP: Settings	1.307288	0	32
2 2500	Cache Generation	0.001482	0	4
2 9010	Total DBTRANS	0.000000	2,977	1
2 9011	Total DBSEL	0.000000	2,977	1
2 9000	Data Manager	20.2949...	0	1
2 3110	OLAP: Data Selection	0.122775	0	1
2 2520	Cache Commit	0.000329	0	1
2 3100	OLAP: Read Data	0.023626	2,977	1
2 3900	OLAP: Read Texts	1.003281	0	9
2 3200	OLAP: Data Transfer	1.722614	20,839	1
2 19...	3.x Query View Data	0.094481	0	1
2 10...	RFC call	0.001707	0	1

The above statistics data is distributed across various database tables. An analysis using two predefined views (**RSDDSTAT_OLAP** and **RSDDSTAT_DM**) is therefore the easiest and can be checked using transaction code SE16.

- The **RSDDSTAT_OLAP** view contains the data from the events from the areas for the front end and calculation layer of the analytic engine (*Front End/Calculation Layer*).

- The **RSDDSTAT_DM** view contains the data from the events from the area for the aggregation layer and analytic engine (contains data from the Data Manager part of the Statistics).

This statistics data can also be displayed using the Queries of the Technical Content MultiProvider 0TCT_MC02.

Invalidation of Cache

The OLAP Cache data can be invalidated. For Basic InfoCubes, Direct Update DSO, and Master data InfoProviders, the system automatically sets a time stamp in table RSDINFOPROVDATA with each change. At Query execution, this time stamp is compared with the time stamp in the cache package. The cache package is ignored and the old entry is removed if the RSDINFOPROVDATA time stamp has a time after the time in the package. The OLAP engine loads the current query result from the InfoProviders, and writes back this new OLAP Cache entry. In case of an InfoSet and MultiProvider, the maximum of the time stamp of participating part providers is considered.

Cache invalidation also occurs when the query structure is changed or the query is regenerated. In addition, the OLAP Cache can be invalidated and deleted completely via the OLAP Cache monitor (transaction RSRCACHE). The OLAP Cache entry is invalidated in the following cases:

- Transaction Data Load to the InfoProvider

- Deletion of Transaction Data from the InfoProvider

- Master Data Load, Hierarchy load and subsequent change run of InfoObjects contained in the query definition

- Currency conversion rate change

Switching Off OLAP Cache

We may require the OLAP Cache to be switched off in some instances including but not limited to performance testing and during performance tuning. This would effectively remove cache overhead. Following is a list of the options for switching it off:

- Using transaction code RSRT to switch off at Query Level for existing ones

- Using transaction code RSDIPROP to switch off for all new Queries to be developed on an InfoProvider

- Using RSRCACHE to make cache Inactive globally

We would want to switch it off:

- for very fast Queries

- for an InfoProvider that gets loaded frequently in a Day

- for ad hoc queries that are not used frequently

- for one time query execution (using transaction RSRT->debug). Select "Do Not Use Cache" under Others as shown below:

Priming the Cache (Filling the Cache)

The OLAP cache can be filled with the Query result set before any user executes a Query. This can be done with the BEx Broadcaster and ensures that the first user of a query has a greater possibility of using OLAP cache. This can be automated using Process Chains (Process type: Trigger event data change for Broadcaster).

Restrictions and Considerations

- If the OLAP Cache is not active, a local cache is used automatically. This local cache stores query results only within one session for the same user and cannot be used by another user.
- The best caching options depend on several factors including frequency of Data load and Query usage and complexity of the Query.
- In general, queries that make good candidates for caching are complex and popular with users and run on a data set that is not often refreshed.
- Queries created and run by a small user group that never uses the same query variables twice do not execute faster from the cache.
- The cache is not automatically turned on or off by the system - so if you set a query to use cache it always will use the cache.
- The OLAP cache cannot be used for characteristics that are defined as InfoProviders (OSS 733296). This results in longer runtimes in executing queries. To check whether an InfoObject is defined as an InfoProvider see the screen shot below.

Under Characteristics Maintenance > Master Data/ Text

Referencing characteristics operate on the same datasets (master data) as the basic characteristic. It is not necessary to allow them for the query definition. The result of the query would be identical to the relevant query on the basic characteristic. SAP makes it clear not to use OLAP cache for these reference InfoObjects.

Since the system has no information about changes to the data store on which a remote provider is based, queries on a remote cube, on a MultiProvider with a remote part provider, or on an InfoSet that uses a Direct update DSO cannot generally use any data from the OLAP cache.

Recommendations

- Query design plays a significant part in the performance. Queries should be reviewed closely to ensure that the design lends itself to the efficiencies provided by OLAP Cache.

- Use standard BW statistics reports to monitor performance before and after priming the cache to determine its effect on each query run-time. This allows you to benchmark and analyze how your efforts affect end user experience.

- Best practices demand that you evaluate the most frequently run queries using the cache and create these as views for Web templates to be run in batch. For less frequently run queries - criticality and business priority should also be a component. Even if the query is not necessarily run frequently, it may be wise to use OLAP cache to ensure that all Users have the same end user experience.

- Aggregates and OLAP Cache can be used together to enhance the Query response time. The data is cached from the Aggregates which is a smaller data set compared to the entire InfoProvider thereby reducing the DB time.

- Regular audit of the Cache should be part of the OLAP Cache strategy. Program RSR_CACHE_RSRV_CHECK_ENTRIES can be scheduled to run on a regular basis to remove the unused cache entries.

Related content

http://help.sap.com/saphelp_nw70ehp1/helpdata/en/41/b987eb1443534ba78a793f4beed9d5/frameset.htm

http://www.sdn.sap.com/irj/scn/go/portal/prtroot/docs/library/uuid/9f4a452b-0301-0010-8ca6-ef25a095834a?QuickLink=index&overridelayout=true

Related OSS Notes:

1026944 - New cache mode for BI 7.0 without directory

1138864 - Delta Procedure of the OLAP Cache

751402 - Cache use: Can change queries w/ variables during navigation

733296 - Characteristic as InfoProvider: Time Stamp for OLAP Cache

807967 - Filling the Cache from the Reporting Agent

822302 - OLAP Cache for Remote Providers

962651 - OLAP Cache - Validation & Invalidation

683162 - Referencing InfoObjects as InfoProviders

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.