

ORACLE®

Top 10 Database Compression Insights: What You Need to Know

ORACLE
OPEN
WORLD



Gregg Christman
Product Manager
Database Performance
January 2019

ORACLE®

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

Program Agenda

- **1 Compression Tips and Insights**
- **2 More Information**
- **3 Questions**

Advanced Compression

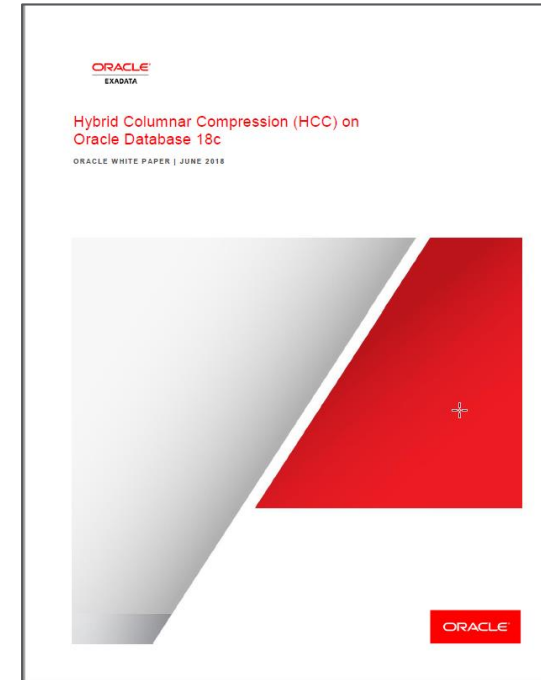
FEATURE:	SUITABLE FOR:
Advanced Row Compression	Data compression for active tables and partitions in OLTP and Data Warehouse environments
Advanced LOB Compression and Deduplication	Unstructured data compression and deduplication for SecureFiles managed data
Advanced Index Compression	Index compression
Advanced Network Compression and Data Guard Redo Transport Compression	Network and Data Guard Redo compression
RMAN/Data Pump Backup Compression	Backup compression
Flashback Data Archive Optimization	Compression of Flashback Data Archive history tables
Heat Map	Automatically track access and usage patterns at the block level (for modifications), and at the table or partition level (for reads and modifications)
Automatic Data Optimization	Declarative policy language to automate compression and movement of data
Online Move Table/Partition <i>to Any Compressed Format</i>	ONLINE Table/Partition Move to compressed format

[1] Updating HCC Compressed Data

Hybrid Columnar Compression

- Hybrid Columnar Compressed row that gets updated gets moved to new Compression Unit (CU)
 - ROWID of the row also changes
 - Moved row's compression level changed from Hybrid Columnar to lesser compressed level (typically 2x to 4x ratio)
- **INSIGHT:** Updated rows can “automatically” be returned to HCC compression levels using Automatic Data Optimization (ADO) policy
 - The row remains in its new Compression Unit
 - Can also *ALTER TABLE MOVE ONLINE* to return to HCC compression
- **IMPACT:** If you do lots of row updates, then HCC performance could be impacted as will the storage footprint for the rows being updated

See: <https://blogs.oracle.com/DBStorage/> (more information about HCC)



<https://www.oracle.com/technetwork/database/database-technologies/performance/hybridcolumnarcompression-oow2010-200703.pdf>

Related MOS notes....

How Hybrid Columnar Compression Works When Rows Are Updated/Inserted (Doc ID 1332853.1)

[2] Estimating LOB Compression

Advanced LOB Compression/SecureFiles LOB Compression

- Best practice suggestion is to use SecureFiles for LOBS over 4k in size, better compression than storing inline
- LOBs, such as documents or XML files, typically experience a compression ratio of 2x to 3x
 - Bitmap images are already compressed and are unlikely to compress any further
 - If LOB was already compressed unlikely Oracle will get much, if any, additional compression
- Starting with Oracle Database 12c users can estimate compression ratios for LOBS with Compression Advisor
 - Can also estimate LOB compression using something akin to gzip

See: <https://blogs.oracle.com/DBStorage/> (more information about LOB Compression)



<http://www.oracle.com/technetwork/database/features/secure-files/securefiles-whitepaper-2009-160970.pdf>

Related MOS notes....

How to Check Space Occupied by LOB Compression (11g)
(Doc ID 861344.1)

[3] Relation of PCTFREE to UPDATES

Advanced Row Compression

- Check PCTFREE if your application performs updates
 - PCTFREE defaults to 10
- If PCTFREE is 0 (or too low), and the block is full, then updates will cause row movement and/or row chaining
 - Row movement/chaining can result in performance degradations for the updates, and for subsequent queries which access the affected rows
 - This is true whether or not the rows in the block are compressed or not
- Before Oracle Database 12cR2, blocks containing many types of chained rows could not be compressed
 - This limitation has been removed in Oracle Database 12c Release 2 and above



<http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/3411538.pdf> (page 8)

Related MOS notes....

Questions on "pctfree" Value and Row Chaining
(Doc ID 2302720.1)

[4] Compression Wait Event in AWR

Advanced Row Compression

- **Question:** If a block is being compressed, and a SQL Insert/Update operation tries to access the block as it's being compressed, what type of *wait event* would you likely see in AWR?
- **Insight:** It would be a *buffer busy wait*
 - *Buffer Busy Waits occur when an Oracle session needs to access a block in the buffer cache, but cannot because the buffer copy of the data block is locked*
- *buffer busy wait* condition can happen because:
 - A session cannot pin the buffer in the buffer cache because another session has the buffer pinned
 - A session cannot pin the buffer in the buffer cache because another session is reading the buffer from disk

C.3.16 buffer busy waits

Wait until a buffer becomes available.

There are four reasons that a session cannot pin a buffer in the buffer cache, and a separate wait event exists for each reason:

1. "buffer busy waits": A session cannot pin the buffer in the buffer cache because another session has the buffer pinned.
2. "read by other session": A session cannot pin the buffer in the buffer cache because another session is reading the buffer from disk.
3. "gc buffer busy acquire": A session cannot pin the buffer in the buffer cache because another session is reading the buffer from the cache of another instance.
4. "gc buffer busy release": A session cannot pin the buffer in the buffer cache because another session on another instance is taking the buffer from this cache into its own cache so it can pin it.

Prior to release 10.1, all four reasons were covered by "buffer busy waits." In release 10.1, the "gc buffer busy" wait event covered both the "gc buffer busy acquire" and "gc buffer busy release" wait events.

Wait Time: Normal wait time is 1 second. If the session was waiting for a buffer during the last wait, then the next wait will be 3 seconds.

Parameter	Description
<i>file#</i>	See "file#"
<i>block#</i>	See "block#"
<i>class#</i>	See "class"

See Also:

[Oracle Database Performance Tuning Guide](#) for more information about the potential causes of the buffer busy waits wait event

<https://docs.oracle.com/en/database/oracle/oracle-database/18/refrn/descriptions-of-wait-events.html#GUID-DA269AB3-4230-4B37-8F33-D3088F6BC1DD>

Related MOS notes....

WAITEVENT "buffer busy waits" Reference Note
(Doc ID 34405.1)

[5] What About Redo Compress?

Advanced Compression

- Advanced Compression **does not** have a feature that compresses redo logs directly
- When a block is recompressed, the redo logs will store enough data to restore the block to its state prior to recompression
 - Some of the features of Advanced Compression will help reduce the amount of redo generated over time, by **reducing the size of database objects**. Including: tables, indexes, SecureFiles LOBs, etc...
- If *Data Guard Redo Transport Compression* is enabled, then the redo data will be compressed before it is transmitted from the primary to the standbys, and then decompressed and applied on the standbys

7 Redo Transport Services

An Oracle Data Guard configuration requires that Oracle redo transport services be configured and monitored.

See the following topics:

- Introduction to Redo Transport Services
- Configuring Redo Transport Services
- Cascaded Redo Transport Destinations
- Data Protection Considerations for Cascading Standbys
- Validating a Configuration
- Monitoring Redo Transport Services
- Tuning Redo Transport

The `COMPRESSION` attribute is used to specify that redo data is transmitted to a redo transport destination in compressed form. Redo transport compression can significantly improve redo transport performance on network links with low bandwidth and high latency.

Redo transport compression is a feature of the Oracle Advanced Compression option. You must purchase a license for this option before using the redo transport compression feature.

The following example uses all of the `LOG_ARCHIVE_DEST_n` attributes described in this section. A `DB_UNIQUE_NAME` has been specified for both destinations, as has the use of compression. If a redo transport fault occurs at either destination, then redo transport attempts to reconnect to that destination, but not more frequently than once every 60 seconds.

```
DB_UNIQUE_NAME=BOSTON
LOG_ARCHIVE_CONFIG='DG_CONFIG=(BOSTON,CHICAGO,HARTFORD) '
LOG_ARCHIVE_DEST_2='SERVICE=CHICAGO ASYNC NOAFFIRM VALID_FOR=(ONLINE_LOGFILE,
PRIMARY_ROLE) REOPEN=60 COMPRESSION=ENABLE DB_UNIQUE_NAME=CHICAGO'
LOG_ARCHIVE_DEST_STATE_2='ENABLE'
LOG_ARCHIVE_DEST_3='SERVICE=HARTFORD SYNC AFFIRM NET_TIMEOUT=30
VALID_FOR=(ONLINE_LOGFILE,PRIMARY_ROLE) REOPEN=60 COMPRESSION=ENABLE
DB_UNIQUE_NAME=HARTFORD'
LOG_ARCHIVE_DEST_STATE_3='ENABLE'
```

<https://docs.oracle.com/en/database/oracle/oracle-database/18/sbydb/oracle-data-guard-redo-transport-services.html#GUID-DC156D33-A034-4766-85BF-ADF414C19945>

Related MOS notes....

Redo Transport Compression in a Data Guard Environment
(Doc ID 729551.1)

[6] Can Users Improve the Compression Ratio?

Advanced Row Compression

- It is possible.... space usage reduction gives the best results where the most duplicate data is stored (low cardinality)
 - Sorting data** (on the columns with the most duplicates) may increase the compression ratio
 - Larger block sizes** may provide higher compression ratios – always test first
- Organizations must consider the cost of both in relation to the amount of increase in the compression ratio
 - A small increase in compression ratio may not be worth the added effort
 - A larger increase, especially with colder/historic data (few/no modifications), may be worth the extra effort

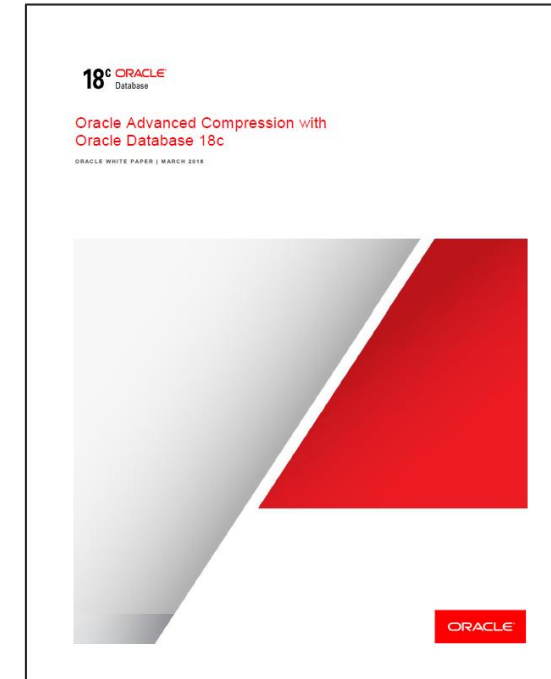


<http://www.oracle.com/ocom/groups/public/@otn/document/s/webcontent/3411538.pdf> (page 8)

[7] Data Pump Data Compression

Data Pump Backup Compression

- Data Pump data compression is completely independent of OLTP or Advanced Row Compression
- The Data Pump dumpfile is uncompressed inline during the import process
 - Data is then imported into the target table based on the compression characteristics of the table
- Data Pump data compression is licensed on the export side, not the import side
 - Can export, using data pump compression, to decrease dumpfile size then import (uncompressed) into databases (including Standard Edition)



<https://www.oracle.com/technetwork/database/options/compression/advanced-compression-wp-12c-1896128.pdf>

Related MOS notes....

Importing into Existing Tables with Oracle Data Pump
(Doc ID 261103.1)

See: <https://blogs.oracle.com/dbstorage/data-pump-and-advanced-compression-%e2%80%93-how-does-this-relationship-benefit-you>

[8] Buffer Cache Efficiency and Compression

Advanced Row Compression

- **Question:** Is there a way to see if compression is helping to improve my buffer cache efficiency?
- **Insight:** Use statistics from V\$SYSSTAT view to help determine if compression is making the buffer cache more efficient
- These numbers typically will be lower, for a compressed table, when compared to the same application using uncompressed data
 - **consistent gets from cache** – Number of times a consistent read was requested for a block from the buffer cache
 - **db block gets from cache** – Number of times a CURRENT block was requested from the buffer cache

E.2 Statistics Descriptions

This section describes some of the statistics stored in the V\$SESSTAT and V\$SYSSTAT views. The statistics are listed in alphabetical order.

The CLASS column contains a number representing one or more statistics classes. The following class numbers are additive:

- 1. User
- 2. Redo
- 4. Enqueue
- 8. Cache
- 16. OS
- 32. Real Application Clusters
- 64. SQL
- 128. Debug

For example, a class value of 72 represents a statistic that relates to SQL statements and caching.

Some statistics are populated only if the TIMED_STATISTICS initialization parameter is set to TRUE. Those statistics are flagged with a Y in the right-hand column.

Table E-1 Database Statistics Descriptions

Name	Class	Description	TIMED_STATISTICS
consistent gets from cache	8	Number of times a consistent read was requested for a block from buffer cache. This is a subset of "consistent gets" statistics value.	
db block gets from cache	8	Number of times a CURRENT block was requested from the buffer cache. This is a subset of "db block gets" statistics value.	

<https://docs.oracle.com/en/database/oracle/oracle-database/18/refrn/statistics-descriptions-2.html#GUID-2FBC1B7E-9123-41DD-8178-96176260A639>

[9] Bitmap Index Compression

Included with the Database

- Bitmap Index Compression is separate from table compression, and is a standard feature of the Oracle Database
 - Advanced Index compression and Prefix Compression doesn't apply to bitmap indexes
 - A bitmap index uses a different key from a B-tree index, but is stored in a B-tree structure -- using branch blocks and leaf blocks just as in a B-tree.
- Bitmap indexes are always stored in a compressed manner without the need of any user intervention

8.4 Bitmap Index Access Paths

Bitmap indexes combine the indexed data with a rowid range.

This section explains how bitmap indexes, and describes some of the more common bitmap index access paths:

About Bitmap Index Access

In a conventional B-tree index, one index entry points to a single row. In a bitmap index, the key is the combination of the indexed data and the rowid range.

Bitmap Conversion to Rowid

A bitmap conversion translates between an entry in the bitmap and a row in a table. The conversion can go from entry to row (`TO ROWID`), or from row to entry (`FROM ROWID`).

Bitmap Index Single Value

This type of access path uses a bitmap index to look up a single key value.

Bitmap Index Range Scans

This type of access path uses a bitmap index to look up a range of values.

Bitmap Merge

This access path merges multiple bitmaps, and returns a single bitmap as a result.

Parent topic: [Optimizer Access Paths](#)

8.4.1 About Bitmap Index Access

In a conventional B-tree index, one index entry points to a single row. In a bitmap index, the key is the combination of the indexed data and the rowid range.

The database stores at least one bitmap for each index key. Each value in the bitmap, which is a series of 1 and 0 values, points to a row within a rowid range. Thus, in a bitmap index, one index entry points to a set of rows rather than a single row.

<https://docs.oracle.com/en/database/oracle/oracle-database/18/tgsql/optimizer-access-paths.html#GUID-B5E41901-933E-4A31-B5D6-9771FD435EC0>

[10] Compression and the Optimizer

Advanced Row/Index Compression

- **Question:** Does compression have any impact on the optimizer -- not directly but indirectly yes
- The statistics for a compressed table, however, are different from the same table in non-compressed form and so execution plan differences can occur
 - If table is compressed, the size of table is smaller. This could make the optimizer prefer a Full-Table-Scan more than it would on the same uncompressed table

4 Query Optimizer Concepts

This chapter describes the most important concepts relating to the query optimizer, including its principal components.

This chapter contains the following topics:

Introduction to the Query Optimizer

The **query optimizer** (called simply the **optimizer**) is built-in database software that determines the most efficient method for a SQL statement to access requested data.

About Optimizer Components

The optimizer contains three components: the transformer, estimator, and plan generator.

About Automatic Tuning Optimizer

The optimizer performs different operations depending on how it is invoked.

About Adaptive Query Optimization

In Oracle Database, **adaptive query optimization** enables the optimizer to make run-time adjustments to execution plans and discover additional information that can lead to better statistics.

About Approximate Query Processing

Approximate query processing is a set of optimization techniques that speed analytic queries by calculating results within an acceptable range of error.

About SQL Plan Management

SQL plan management enables the optimizer to automatically manage execution plans, ensuring that the database uses only known or verified plans.

About the Expression Statistics Store (ESS)

The **Expression Statistics Store (ESS)** is a repository maintained by the optimizer to store statistics about expression evaluation.

Parent topic: Query Optimizer Fundamentals

<https://docs.oracle.com/en/database/oracle/oracle-database/18/tgsql/query-optimizer-concepts.html#GUID-06129ACE-36B2-4534-AE68-EDFCAEBB3B5D>

Related MOS notes....

Master Note for OLTP Compression

(Doc ID 1223705.1)

[11] Compression and Encryption

Advanced Row/Index Compression

▪ Tablespace Encryption

- *Data and index compression are done before encryption* -- this ensures the maximum space and performance benefits from compression, while also receiving the security of encryption at rest

▪ Column Encryption

- *Compression is done after encryption.* This means that compression will have minimal effectiveness on encrypted columns
- There is one notable exception: if the column is a Secure Files LOB, and the encryption is implemented with Secure Files LOB Encryption, and the compression (and possibly deduplication) are implemented with Secure Files LOB Compression & Deduplication, then compression will be done before encryption



<https://www.oracle.com/technetwork/database/options/advanced-security/overview/advanced-security-tde-faq-2995212.pdf>

[12] Index Compression

Advanced Index and Prefix Compression

- **Prefix Compression** (included with the database)
 - Eliminates duplicate copies of pre-defined number of index prefix columns at the index leaf block level
 - *Customers experience: 2x compression ratios*
 - Running ANALYZE INDEX will give advice on whether / how many columns to choose
 - When the leading columns are very selective, or if there are not many repeat for the prefix columns, it is possible to make indexes larger than their uncompressed equivalent due to the overhead to store the prefix entries
- **Advanced Index Compression** (Advanced Compression)
 - The optimal compression algorithms will be computed on a block by block bases
 - Two compression levels: **Advanced High** (4x-5x compression ratios) and **Advanced Low** (2x compression ratios)



<http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/3554923.pdf>




Program Agenda

- 1 Compression Tips and Insights
- 2 **More Information**
- 3 Questions

Additional Resources



Join the Conversation

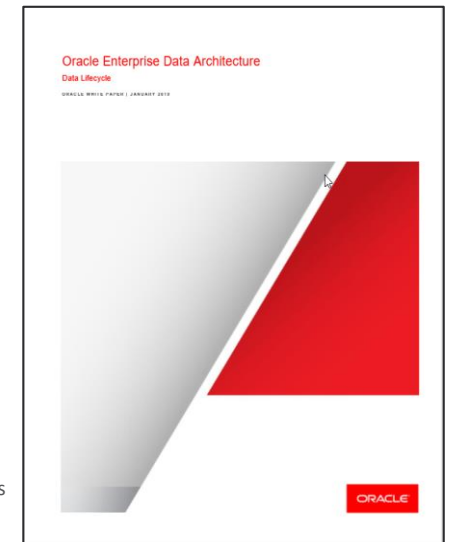
-  https://twitter.com/aco_gregg
-  <https://blogs.oracle.com/DBStorage/>
-  <http://www.oracle.com/database/advanced-compression/index.html>

<https://www.oracle.com/database/technologies/advanced-compression.html>

The screenshot shows the Oracle website header with the Oracle logo, search, sign in, and country/region options. The breadcrumb trail reads 'Products and Services / Database / Database Technologies / Advanced Compression'. The main heading is 'Oracle Advanced Compression'. Below it, a paragraph describes the technology: 'Provides a comprehensive set of compression capabilities to help improve database performance and reduce storage costs. It allows organizations to reduce their overall database storage footprint by enabling compression for all types of data: relational (table), unstructured (file), index, network, Data Guard Redo and backup data. While query performance, storage cost savings and data optimization are often seen as the most tangible benefits, additional innovative technologies can help reduce CapEx and OpEx costs for all components of an IT infrastructure, including memory and network bandwidth as well as heating, cooling and floor-space.' A 'Join the conversation' button with a right arrow is located at the bottom of the text block.

Advanced Row Compression
Advanced LOB Compression
Advanced Index Compression
Hybrid Columnar Compression
Automatic Data Optimization
Heat Map
SecureFiles

<https://www.oracle.com/databases/technologies/advanced-compression.html>



More Information....

MOS notes....

- Advanced Compression Master Note (Doc ID 1223705.1)
- How to compress a table that is online (Doc ID 1353967.1)
- **Advanced Compression critical patches** (Doc ID 1061366.1)
- Redo Transport compression with Data Guard (Doc ID 729551.1)
- **How to see if rows are compressed in a table** (Doc ID 1477918.1)
- A complete understanding of RMAN compression (Doc ID 563427.1)
- How to determine if Advanced Compression is used by DataPump (Doc ID 1993134.1)
- Index Organized Tables (IOTs) and compression (Doc ID 1555637.1)
- **How to find the optimal Index Key COMPRESS level for Indexes** (Doc ID 601690.1)

Questions



ORACLE®