ORACLE

EXADATA

Hybrid Columnar Compression (HCC) on Exadata

ORACLE

Introduction

Hybrid Columnar Compression on Exadata enables the highest levels of data compression and provides enterprises with tremendous cost-savings and performance improvements due to reduced I/O. HCC is optimized to use both database and storage capabilities on Exadata to deliver tremendous space savings AND revolutionary performance. Average storage savings can range from 10x to 15x depending on which Hybrid Columnar Compression level is implemented – real world customer benchmarks have resulted in storage savings of up to 204x.

With average storage savings of 10x from HCC, IT managers can drastically reduce and often eliminate their need to purchase new storage for several years. For example, a 100 terabyte database achieving 10x storage savings would utilize only 10 terabytes of physical storage. With 90 terabytes of storage now available, IT organizations could potentially delay storage purchases for a significant amount of time. The 90 terabytes of storage could even be used to store up to 9 more databases with 100 terabytes of data compressed to 10 terabytes of actual disk space each.

Hybrid Columnar Compression is an enabling technology for both Warehouse Compression and Archive Compression. We will discuss each of these capabilities in detail later in this paper, but first let's explore the implementation/benefits of Hybrid Columnar Compression on Exadata – the next generation in compression technology.

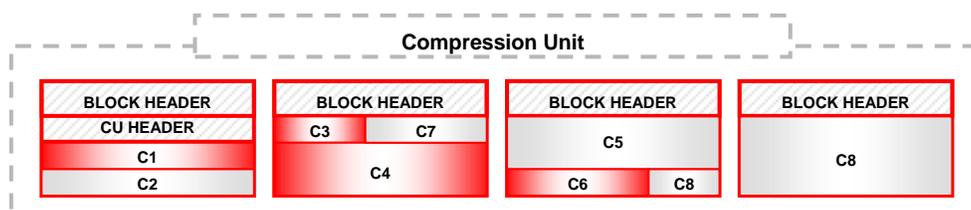# Hybrid Columnar Compression: Technology Overview

Traditionally, data has been organized within a database block in a 'row' format, where all column data for a particular row is stored sequentially within a single database block. Having data from columns with different data types stored close together limits the amount of storage savings achievable with compression technology. An alternative approach is to store data in a 'columnar' format, where data is organized and stored by column.

Storing column data together, with the same data type and similar characteristics, dramatically increases the storage savings achieved from compression. However, storing data in this manner can negatively impact database performance when application queries access more than one or two columns, perform even a modest number of updates, or insert small numbers of rows per transaction.

Oracle's Hybrid Columnar Compression technology is a new method for organizing data within a database block. As the name implies, this technology utilizes a combination of both row and columnar methods for storing data. This hybrid approach achieves the compression benefits of columnar storage, while avoiding the performance shortfalls of a pure columnar format.

A logical construct called the compression unit is used to store a set of hybrid columnar-compressed rows. When data is loaded, column values for a set of rows are grouped together and compressed. After the column data for a set of rows has been compressed, it is stored in a compression unit.

Conceptual Illustration of a Logical Compression Unit

To maximize storage savings with Hybrid Columnar Compression, data must be loaded using data warehouse bulk loading techniques. Examples of bulk load operations commonly used in data warehouse environments are:

- Insert statements with the APPEND hint

- Parallel DML

- Direct Path SQL*LDR

- Create Table as Select (CTAS)

Queries on hybrid columnar compressed data often run in the Exadata storage cells with Smart Scans, using a high performance query engine that utilizes special columnar processing techniques. Data sent back to the database server(s) is usually compressed (and is typically much less data than is read from disk) and the compressed data is subsequently processed by the database server(s). Note that data remains compressed not only on disk, but also remains compressed in the Exadata Smart Flash Cache, on Infiniband, in the database server buffer cache, as well as when doing back-ups or log shipping to Data Guard.

One of the key benefits of the hybrid columnar approach is that it provides both the compression and performance benefits of columnar storage without sacrificing the robust feature set of the Oracle Database. For example, while optimized for scan-level access, Oracle is still able to provide efficient row-level access, with entire rows typically being retrieved with a single I/O, because row data is self-contained within compression units.

In contrast, pure columnar formats require at least one I/O per column for row-level access. With data warehousing tables generally having hundreds of columns, it is easy to see the performance benefits of Hybrid Columnar Compression on Exadata. Further, tables using Hybrid Columnar Compression on Exadata still benefit from all of the high availability, performance, and security features of the Oracle Database.

Note that while data in Hybrid Columnar compressed tables can be modified using conventional Data Manipulation Language (DML) operations - INSERT, UPDATE, DELETE - performing such operations could result in a reduction of the HCC compression ratio. It is recommended that HCC be enabled on tables or partitions with no or infrequent DML operations. If frequent DML operations are planned on a table or partition, then the Oracle Advanced Compression Option is better suited for such data.

Warehouse Compression

Data warehouses have become increasingly important in the day-to-day operations of enterprises. They are responsible for storing significant amounts of data, transforming that data into strategic information, and providing management with the necessary intelligence to run the enterprise. As the importance of data warehouses has increased, so too has the amount of data managed by data warehouses. With data volumes often doubling every two years, IT Managers are experiencing significant challenges both in storage costs and application query performance. Warehouse

Compression is the next-generation compression feature dedicated to solving both of these challenges.

Warehouse Compression provides significant storage savings by leveraging Hybrid Columnar Compression technology. Warehouse Compression typically provides a 10:1 (10x) compression ratio, delivering roughly five times the industry average savings. For example, enabling Warehouse Compression on an uncompressed 100 terabyte data warehouse would reduce the storage requirements to only 10 terabytes. Warehouse Compression would return 90 terabytes of storage back to the enterprise for other uses. In fact, the enterprise could use this reclaimed storage to support the growth of its data warehouse without purchasing additional storage for over 4 years, assuming the database doubled in size every two years. Clearly, storage savings of this magnitude dramatically reduce costs as enterprises can significantly delay storage purchases for many years.

Many data warehouse applications are hitting a performance bottleneck due to growth in data volumes. Analytical queries are scanning hundreds of gigabytes, if not terabytes, of data making the storage system the limiting factor to performance and scalability. With Oracle Exadata Storage Server, traditional and costly remedies such as purchasing additional disks to improve scan performance and I/O throughput are no longer necessary. With Exadata's massively parallel storage grid, Smart Scan capabilities, and Warehouse Compression, IT administrators are no longer forced to increase the number of disk drives in their storage arrays simply to increase performance.

While Warehouse Compression is a storage saving feature, the implementation of Hybrid Columnar Compression on Exadata is optimized to improve I/O scan performance during typical Data Warehouse queries. The I/O required to scan a Warehouse-compressed table typically decreases by the compression ratio achieved. Therefore, scan-oriented queries that access a table that has a compression ratio of 10:1 will likely have a reduction in I/O of up to 10x. Total query performance will also likely improve, however it will depend on the available CPU resources.

Further improving performance is Exadata's Smart Scan technology, which greatly reduces the amount of data sent from storage to the database server by offloading much of the scan activities to the Exadata storage. Exadata Smart Scan works directly on hybrid columnar-compressed data on Exadata. With this level of improvement in I/O scan performance, Warehouse Compression reduces costs by both decreasing the amount of storage required and by eliminating the need to increase the number of disk drives and related hardware to meet performance objectives.

Warehouse Compression provides two levels of compression: LOW and HIGH. Warehouse Compression HIGH typically provides a 10x reduction in storage, while Warehouse Compression LOW typically provides a 6x reduction. Both levels have been optimized on Exadata to increase scan query performance by taking advantage of the fewer number of blocks on disk.

To maximize the storage savings and query performance benefits of Warehouse Compression, the default level is HIGH. The increased storage savings may cause data load times to increase modestly. Therefore, Warehouse Compression LOW should be chosen for environments where load time service levels are more critical than query performance.

Archive Compression

One of the biggest challenges facing IT administrators today is the cost and complexity of managing historical data. IT managers are being forced to reduce costs, yet conflicting business requirements dictate that data be kept available for significantly longer periods of time, often indefinitely.

Organizations have developed Information Lifecycle Management (ILM) strategies to help mitigate the costs of storing this data. As data ages, the typical ILM strategy involves moving data to less expensive storage, including less expensive disk drives and often archiving this data to tape. As a result, the more expensive and higher performing disk drives are used exclusively for the most recent and thus most accessed data. Hybrid Columnar Compression's Archive Compression is a new approach to reducing the storage requirements and costs of storing this historical data.

Archive Compression provides significant storage savings by leveraging Hybrid Columnar Compression technology. Archive Compression is optimized to maximize storage savings, typically achieving a compression ratio of 15:1 (15x). That is, an uncompressed table or partition would require 15x more storage than a table or partition using Archive Compression.

In contrast to Warehouse Compression, Archive Compression is a pure storage saving technology. Tables or partitions utilizing Archive Compression will typically experience a decrease in performance - a factor of the compression algorithm being optimized for maximum storage savings. Therefore, Archive Compression is intended for tables or partitions that store data that is rarely accessed.

Databases supporting any application workload, including OLTP and Data Warehouses, can use Archive Compression to reduce the storage requirements of historical data. Oracle supports

enabling any type of table compression at the partition or sub-partition level. An OLTP application, therefore, can store historical data in partitions with Archive Compression, while active data remains in partitions with Oracle's OLTP Table Compression. OLTP Table compression, a feature of the Advanced Compression Option, is a compression technology that is optimized for active transactional databases. OLTP Table compression typically provides storage savings of 2x - 4x, delivering significant savings to OLTP databases. Data Warehouses on Exadata will typically store frequently queried data in partitions with Warehouse Compression (for performance), while historical data is stored in partitions with Archive Compression (for storage savings).

In many applications, historical data is responsible for consuming up to 80% of the allocated storage. It is no wonder that IT administrators are implementing ILM strategies that archive much of this historical data to tape. However, this approach has several inherent flaws. Once data is archived to tape, the application can no longer access this data directly. In order to access the archived data, IT administrators must first restore the data from tape and load it back into the database. This can take a tremendous amount of time and doesn't meet the requirements of today's fast-paced businesses.

Complicating things further, data archived to tape becomes out of sync with structural changes to the database schema, such as the addition of columns and constraints. Therefore, restoring this data back into the database requires not only a significant amount of time but also a significant amount of resources to correctly restore the data and make it accessible by the application. Of course, requests for this data are usually extremely urgent and any delays affect management's ability to make critical business decisions. As you can see, this approach to storing historical data can actually be quite costly to the business.

Archive Compression provides the storage savings benefits of archiving data to tape while keeping this data online for immediate access and modification. Further, as the application evolves, all the historical data will evolve with the database schema modifications, such as new columns, constraints, etc. Therefore, when an application user needs to access historical data, the application will be able to seamlessly service queries without any need to involve IT administrators or application developers.

While query performance against tables or partitions with Archive Compression is slower than tables or partitions with Warehouse Compression or OLTP Table Compression, they are orders of magnitude faster than queries against data that is archived to tape.

Migration and Best Practices

For new tables and partitions, enabling Hybrid Columnar Compression is as easy as simply CREATEing the table or partition and specifying a compression level, such as "COMPRESS FOR QUERY HIGH". See the example below:

CREATE TABLE emp (emp_id NUMBER, first_name VARCHAR2(128), last_name VARCHAR2(128))
COMPRESS FOR QUERY HIGH;

The **COMPRESS FOR QUERY HIGH** compression level is used as an example in this document, the additional compression levels, available with Hybrid Columnar Compression, include:

COMPRESS FOR QUERY LOW

COMPRESS FOR ARCHIVE LOW

COMPRESS FOR ARCHIVE HIGH

For existing tables and partitions, there are two recommended approaches to enabling Hybrid Columnar Compression:

1. **Online Redefinition (DBMS_REDEFINITION)**

   - This approach will enable Hybrid Columnar Compression for future bulk loads and also compress existing data. Using DBMS_REDEFINITION keeps the table online for both read/write activity during the migration. Run DBMS_REDEFINITION in parallel for best performance.

   - Online redefinition will clone the indexes to the interim table during the operation. All the cloned indexes are incrementally maintained during the sync (refresh) operation so there is no interrupt in the use of the indexes during, or after, the online redefinition. The only exception is when online redefinition is used for redefining a partition -- the global index is invalidated and needs to be rebuilt after the online redefinition.

2. **ALTER TABLE … MOVE COMPRESS FOR QUERY HIGH**

   - This approach will enable Hybrid Columnar Compression for future bulk loads and also compress existing data. While the table is being moved it is online for

read activity but has an exclusive (X) lock – so all DML will be blocked until the move command completes. Run ALTER TABLE…MOVE in parallel for best performance.

- The ALTER TABLE...MOVE statement allows you to relocate data of a non-partitioned table, or of a partition of a partitioned table, into a new segment, and optionally into a different tablespace. ALTER TABLE…MOVE COMPRESS compresses the data by creating new extents for the compressed data in the tablespace being moved to -- it is important to note that the positioning of the new segment can be anywhere within the datafile, not necessarily at the tail of the file or head of the file. When the original segment is released, depending on the location of the extents, it may or may not be possible to shrink the datafile.

- ALTER TABLE… MOVE will invalidate any indexes on the partition or table; those indexes will need to be rebuilt after the ALTER TABLE… MOVE. For partition moves, the use of ALTER TABLE… MOVE PARTITION with the UPDATE INDEXES clause will maintain indexes (it places an exclusive (X) lock so all DML will be blocked until the move command completes) – not available for non-partitioned tables.

Below are some best practices and considerations when using Hybrid Columnar Compression:

- The best test environment for Hybrid Columnar Compression is where you can most closely duplicate the production environment– this will provide the most realistic (pre- and post- compression) performance comparisons.

- Oracle Advanced Compression Advisor (DBMS_COMPRESSION) is a PL/SQL package, included with Oracle Database 11g Release 2, that is used to estimate potential storage savings for Hybrid Columnar Compression based on analysis of a sample of data. It provides a good estimate of the actual results that may be obtained after implementing Hybrid Columnar Compression. It also provides compression ratio estimates for table compression using OLTP Table Compression and Basic Table Compression. Compression Advisor requires a minimum of 1 million records in a table/partition in order to estimate Hybrid Columnar Compression ratios.

- The support of Hybrid Columnar Compression on Pillar and ZFSSA enables Oracle Database users to utilize Oracle's Hybrid Columnar Compression on Pillar Axiom and

Sun ZFS Storage Appliance (ZFSSA) storage hardware. This provides the storage benefits of Oracle's Hybrid Columnar Compression, which had previously been exclusive to the Exadata platform, to Oracle Database users who use Pillar Axiom or Sun ZFSSA storage (or both), enabling compression ratios of 6x to 15x, depending on the data and the compression level chosen by the user.

- Hybrid Columnar Compression is NOT supported for use with the LONG data types and the use of UNIFORM EXTENTS is not recommended with Hybrid Columnar Compression

- There are no restrictions with Hybrid Columnar Compression in regards to the minimal amount of data needed with HCC. HCC can be very effective even with only a few MB's of data per segment/partition. However, when using smaller amounts of data (MB's per segment) and Parallel Loads it is important to note that Parallel Loads sometime use temp segment merge, where each loader process creates a separate segment, in this scenario Oracle recommends having a couple of hundred MB's per segment/partition.

- Hybrid Columnar Compression is designed for relational data, not for unstructured data in BLOBs (or CLOBs). LOBs are best stored in the Oracle Database as SecureFiles LOBs, and if the customer has licensed the Advanced Compression Option, they can use SecureFiles Compression and Deduplication to potentially reduce the amount of storage required for LOBs.

- Hybrid Columnar Compression is primarily intended for QUERY intensive data as well as archive/historic data. DML operations (INSERT/UPDATE) against a Hybrid Columnar Compressed table/partition can reduce the overall compression savings over time since data INSERTED/UPDATED via DML operations will not be compressed to the same ratio as data that is bulk loaded.

Conclusion

With exploding data volume growth in both data warehouses and OLTP applications, IT Managers need tools to efficiently manage their IT infrastructure while controlling costs and maintaining or improving performance.

Hybrid Columnar Compression on Exadata provides the IT Manager with those exact tools - a robust set of compression features that drastically reduce infrastructure costs while improving application performance.

ORACLE®

Hybrid Columnar Compression on Exadata
November 2012
Author: Gregg Christman
Contributing Authors: Kevin Jernigan

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Oracle is committed to developing practices and products that help protect the environment

Hardware and Software, Engineered to Work Together