19c ORACLE Database

Oracle Advanced Compression Frequently Asked Questions

FEBRUARY 2019

ORACLE

1.0 About Compression: The Features and more…

1.1 What are the features of Oracle Advanced Compression?

Advanced Compression includes a comprehensive set of compression features designed to maximize resource utilization and reduce costs by enabling compression for structured data (Advanced Row Compression), unstructured data (Advanced LOB Deduplication and Compression), database backups (RMAN and Data Pump compression) and network compression (Data Guard Redo Log transport and Advanced Network Compression).

Advanced Network Compression compresses data transferred between Oracle Database and client applications. It is transparent to client applications and can improve SQL response times while saving network bandwidth. Advanced Compression also includes an optimization for Flashback Data Archive (FDA) history tables, which reduces the storage requirements when using FDA to track changes to tables.

Several features of Advanced Compression enhance the Information Lifecycle Management capabilities of Oracle Database. Heat Map automatically tracks data modification times at the row and segment level, and access times at the segment level, providing unprecedented insights into how data is being accessed. Automatic Data Optimization (ADO) provides declarative syntax to automatically move and compress data based on usage statistics collected by Heat Map. Together these capabilities help implement Information Lifecycle Management (ILM) strategies.

1.2 With cost of disk storage falling, why do I still need to care about compression?

Enterprises are experiencing an explosion in the volume of data required to effectively run their businesses. This trend in data growth can be attributed to several key factors. Recent changes in the regulatory landscape, such as Sarbanes-Oxley and HIPAA, are contributing to this trend by mandating that enterprises retain large amounts of information for long periods of time.

Mass distribution of rich and multimedia content over the Internet, made possible through advancements in broadband technologies, also contributes to the growth in overall data volume. Various estimates indicate that data volume is almost doubling every 2-3 years.

This sudden explosion in data volume presents a daunting management challenge for IT administrators. First and foremost are the spiraling storage costs: even though the cost per MB of storage has been declining dramatically in the last few years, the enormous growth in

the volume of data that needs to be retained online makes storage one of the biggest cost elements of most IT budgets. In addition, application scalability and performance must continue to meet the demands of the business – even as data volumes explode.

Advanced Compression helps organizations cope with these challenges. Innovations in Oracle compression technologies help organizations reduce the resources and costs of managing large data volumes. Another important benefit is in database performance. A major bottleneck for many systems is I/O bandwidth. Advanced Compression can help alleviate that bottleneck in several cases by reducing the amount of data that needs to be transferred across I/O channel and also further boost performance through improved memory efficiencies.

1.3 Who is the typical Advanced Compression customer?

Oracle compression is used by financial, government, education, healthcare, utilities, insurance, retail, manufacturing and more….

1.4 My storage has compression, why not just my storage-based compression?

Oracle compression is built into the database, this allows Oracle to keep data and indexes compressed in memory – this isn't possible with storage-based compression

2.0 Enabling Compression

2.1 How do I compress an existing table online to enable Advanced Row Compression?

There are three recommended approaches to enabling Advanced Row Compression on existing tables:

Online Redefinition (DBMS_REDEFINITION)

This approach will enable Advanced Row Compression for future DML 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 interruption 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 -- any global indexes are invalidated and need to be rebuilt after the online redefinition.

ALTER TABLE … MOVE ROW STORE COMPRESS ADVANCED

This approach will enable Advanced Row Compression for future DML 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.

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.

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 ROW STORE COMPRESS ADVANCED 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 data file, 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 data file.

ALTER TABLE … MOVE TABLE/PARTITION/SUBPARTITION … ONLINE

This approach will enable Advanced Row Compression for future DML and will compress existing data. ALTER TABLE ... MOVE TABLE/PARTITION/SUBPARTITION … ONLINE allows DML operations to continue to run uninterrupted on the table/partition/subpartition being moved. Indexes are maintained during the move operation, so a manual index rebuild is not required.

2.2 How much compression can I expect by using Advanced Row Compression?

The compression ratio achieved in a given environment depends on the nature of the data being compressed; specifically the cardinality of the data. In general, organizations can expect to reduce their storage space consumption by a factor of 2x to 4x when using Advanced Row Compression. A 2x compression ratio represents approximately a 50% reduction in the storage footprint.

2.3 Can I compress data at a partition level?

Yes. Compression can be done at a tablespace, table or partition level.

2.4 Do I have to compress all the tables/partitions in my database?

No – You can pick-and-choose which tables/partitions you want to compress and you do not have to compress all tables/partitions at one time

2.5 Should I enable compression at the Tablespace level?

Regarding whether or not to compress at the Tablespace level: For custom applications, we recommend compressing at the Tablespace level, but users should consider turning off compression on very high traffic tables, such as tables used as queues. For commercial packaged applications, where typically the number of objects can be very large, the recommended approach is object selection instead of exclusion. Often the largest tables and indexes consume the majority of the database space. Compressing those objects, while excluding high traffic objects like tables used as queues, will give the majority of the compression benefits. Other objects can be compressed over time as needed.

2.6 After I enable compression, is there any other admin work we need to do to maintain compression?

No. Once compression is enabled there isn't any maintenance required to maintain the compression of your tables and partitions

2.7 How much storage will I save using Advanced Compression?

Advanced Compression Advisor is a PL/SQL package used to estimate potential compression ratios, for Advanced Row Compression, based on analysis of a sample of data. It provides a good estimate of the actual compression ratio that may be obtained after implementing Advanced Row Compression.

A version of Advanced Compression Advisor, which supports Oracle Database 9*i* Release 2 through 11g Release 1, is available (free) on the Oracle Technology Network Advanced Compression website. The Advanced Compression Advisor (DBMS_COMPRESSION) is included with Oracle Database 11g Release 2 and above

2.8 Can SecureFiles LOBs be compressed.

Yes. The Advanced LOB Compression and Deduplication features of Advanced Compression intended to reduce the amount of storage required for SecureFiles LOBs.

2.9 Can Index-Organized Tables be compressed with Advanced Row Compression or Advanced Index Compression?

No. Index-Organized Tables (IOT's) are essentially indexes, so they can't be compressed with Advanced Row or Basic Compression. IOT's can be compressed with Prefix Compression but not with Advanced Index Compression.

2.10 Should I compress all tables?

The general recommendation is to compress all the tables in the database with one exception: if the table is used as a queue, i.e. rows are inserted into the table, then later most or all of the rows are deleted, then more rows are inserted then deleted, then you shouldn't compress the table.

2.11 Should I create some test data and apps to test Advanced Compression?

No. The best test environment for each Advanced Compression capability is where you can most closely duplicate the production environment (using your actual data and applications) – this will provide the most realistic (pre- and post- compression) performance and functionality comparisons.

2.12 My system doesn't have any spare CPU resources, is that going to be an issue?

Although CPU overhead is typically minimal (3% to 5% typically), implementing Advanced Row Compression is ideal on systems with available CPU cycles, as compression will have additional, although minor overhead for some DML operations.

2.13 Are there any data types not supported by Advanced Row Compression?

Advanced Row Compression is NOT supported for use with tables that have LONG data types.

2.14 Will a larger block size provide better compression ratios?

Larger blocks don't always ensure higher Advanced Row Compression ratios. Test with your own data to determine if larger/smaller block sizes will have an impact on your Advanced Row Compression ratio.

3.0 Compression Benefits

3.1 What are the benefits of using Advanced Row Compression?

Advanced Row Compression uses a unique compression algorithm specifically designed to work with OLTP (and data warehouse) applications. The algorithm works by eliminating duplicate values within a database block, even across multiple columns. Compressed blocks contain a structure called a symbol table that maintains compression metadata. When a block is compressed, duplicate values are eliminated by first adding a single copy of the duplicate value to the symbol table. Each duplicate value is then replaced by a short reference to the appropriate entry in the symbol table.

Through this innovative design, compressed data is self-contained within the database block, as the metadata used to translate compressed data into its original state is stored in the block header. When compared with competing compression algorithms that maintain a global database symbol table, Oracle's approach offers significant performance benefits by not introducing additional I/O when accessing compressed data.

The compression ratio achieved in a given environment depends on the data being compressed, specifically the cardinality of the data. In general, organizations can expect to reduce their storage space consumption by a factor of 2x to 4x by using Advanced Row Compression. That is, the amount of space consumed by uncompressed data will be two to four times larger than that of the compressed data.

The benefits of Advanced Row Compression go beyond just on-disk storage savings. One significant advantage is Oracle's ability to read compressed blocks directly without uncompressing the blocks. This helps improve performance due to the reduction in I/O, and the reduction in system calls related to the I/O operations. Further, the buffer cache becomes more efficient by storing more data without having to add memory.

3.2 I am already using the Oracle Basic Table Compression feature introduced in Oracle 9i. What additional benefits do I get with Oracle Advanced Compression?

Oracle Database 9i introduced Basic Table Compression which only compressed data that was loaded using bulk load operations. Advanced Row Compression, a feature of Advanced

Compression, allows data to be compressed during all types of data manipulation operations, including conventional DML such as INSERT and UPDATE. In addition, Advanced Row Compression reduces the associated compression overhead of write operations making it suitable for transactional/OLTP environments. Advanced Row Compression, therefore, extends the benefits of compression to all application workloads.

Although storage cost savings and optimization across servers (production, development, QA, Test, Backup and etc...) are often seen as the most tangible benefits, additional innovative technologies included in Advanced Compression are designed improve performance and to reduce CapEx and OpEx costs for all components of your IT infrastructure, including memory and network bandwidth as well as heating, cooling and floor-space costs.

3.3 I am already using the Basic RMAN Backup Compression feature -- in what way is the RMAN Backup Compression feature of Advanced Compression different from this?

Advanced Compression provides three levels of RMAN Compression: LOW, MEDIUM, and HIGH. The amount of storage savings increases from LOW to HIGH, while potentially consuming more CPU resources. Compression Level LOW provides the fastest compression algorithm and is best suited when backup is constrained by CPU. Compression Level MEDIUM provides a balance between CPU usage and compression ratio and finally, Compression LEVEL HIGH provides the best compression ratio and highest CPU utilization and is best suited when backup is constrained by network or I/O.

3.4 What are SecureFiles? What is the relationship between Advanced LOB Compression and SecureFiles?

SecureFiles, a feature in Oracle Database, offers a 'best-of-both-worlds' architecture for storing unstructured content, such as documents, spreadsheets and XML files. SecureFiles is specifically engineered to deliver high performance for file data comparable to that of traditional file systems while retaining the advantages of the Oracle database.

SecureFiles is designed as a superset of the ANSI standard LOB data type and offers easy migration from existing BasicFile LOBs, the precursor to SecureFiles. With SecureFiles, organizations can now manage all relational data and associated file data in Oracle using a single security/audit model, a unified backup & recovery process, and perform seamless retrievals across all information.

Advanced Compression has two storage optimization features that can be leveraged with

SecureFiles. The first feature, Advanced LOB Deduplication, is an intelligent technology that eliminates duplicate copies of SecureFiles data. The second feature, Advanced LOB Compression, utilizes industry standard compression algorithms to further minimize the storage requirements of SecureFiles data.

There are three levels of Advanced LOB compression available: LOW, MEDIUM, and HIGH. By default, Advanced LOB Compression uses the MEDIUM level, which typically provides good compression with a modest CPU overhead of 3%-5%. SecureFiles Compression LOW is optimized for high performance and maintains about 80% of the compression achieved through MEDIUM, while utilizing 3x less CPU. Finally, Advanced LOB Compression HIGH achieves the highest storage savings but incurs the most CPU overhead.

4.0 Compression Algorithm and Optimizations

4.1 What kind of technology is used to compress data?

Advanced Row Compression uses a unique compression algorithm specifically designed to work with OLTP/DW applications. The algorithm works by eliminating duplicate values within a database block, even across multiple columns.

Compressed blocks contain a structure called a symbol table that maintains compression metadata. When a block is compressed, duplicate values are eliminated by first adding a single copy of the duplicate value to the symbol table. Each duplicate value is then replaced by a short reference to the appropriate entry in the symbol table. Through this innovative design, compressed data is self-contained within the database block as the metadata used to translate compressed data into its original state is stored in the block.

When compared with competing compression algorithms that maintain a global database symbol table, Oracle's unique approach offers significant performance benefits by not introducing additional I/O when accessing compressed data.

4.2 What optimizations has Oracle done to minimize compression overhead?

Advanced Row Compression has no adverse impact on read operations. There is additional work performed while writing data, making it impossible to eliminate performance overhead for write operations. However, Oracle has put in a significant amount of work to minimize this overhead for Advanced Row Compression.

Oracle compresses blocks in batch mode rather than compressing data every time a write

operation takes place. A newly initialized block remains uncompressed until data in the block reaches an internally controlled threshold. When a transaction causes the data in the block to reach this threshold, all contents of the block are compressed. Subsequently, as more data is added to the block and the threshold is again reached, the entire block is recompressed to achieve the highest level of compression.

This process repeats until Oracle determines that the block can no longer benefit from further compression. Only transactions that trigger the compression of the block will experience the slight compression overhead. Therefore, a majority of transactions on compressed blocks will have the exact same performance as they would with uncompressed blocks.

4.3 Does table data get decompressed before it is read?

No. Oracle has the ability to read compressed blocks directly without having to first uncompress the block. Therefore, there is no measurable performance degradation for accessing compressed data. In fact, in many cases performance may improve due to the reduction in I/O since Oracle will have to access fewer blocks. Further, the buffer cache will become more efficient by storing more data without having to add memory.

4.4 What is the performance impact of using Advanced Row Compression?

For DML operations on a compressed table, Advanced Row Compression's specialized batch algorithm keeps the performance overhead to a minimum. Internal tests at Oracle showed a minimal overhead of less than 5% (CPU) for a DML workload.

It is important to note that Oracle compresses blocks in batch mode rather than compressing data every time a write operation takes place. When a transaction causes the data in the block to reach an internal threshold, all contents of the block are compressed. Subsequently, as more data is added to the block and the threshold is again reached, the entire block is recompressed to achieve the highest level of compression.

This process repeats until Oracle determines that the block can no longer benefit from further compression. Only transactions that trigger the compression of the block will experience the slight compression overhead. Therefore, a majority of transactions on compressed blocks will have the exact same performance as they would with uncompressed blocks.

4.5 Do chained rows get compressed?

Before Oracle Database 12*c*, blocks containing many types of chained rows could not be compressed. This limitation was removed starting with Oracle Database 12*c* Release 2.

5.0 Information Lifecycle Management

5.1 What does Advanced Compression provide in terms of Information Lifecycle Management?

Information Lifecycle Management (ILM) is the practice of applying policies for the effective management of information throughout its useful life. ILM includes every phase of a "row" from its beginning to its end, and consists of the policies, processes, practices, and tools used to align the business value of information with the most appropriate and cost effective IT infrastructure from the time information is created through its final disposition.

Automatic Data Optimization (ADO) can create policies, and automate actions (compression tiering and storage tiering) based on those policies, to implement your ILM strategy. ADO utilizes the usage statistics collected by Heat Map, and depending on your ILM requirements, may also require use of Partitioning, Advanced Row Compression, and Hybrid Columnar Compression.

More information about ADO is available on the Oracle Technology Network (OTN) page for Automatic Data Optimization.

6.0 Compression Overhead

6.1 What is the overhead associated with Advanced Row Compression?

Approximately 3% to 5% CPU is typically reported by customers. CPU overhead offset partially by reduced IO

7.0 Compression and Data Pump

7.1 Does Data Pump use Advanced Row Compression to compress backups?

No**.** Data Pump compression is completely independent of Advanced Row Compression. The Data Pump dump file is uncompressed inline during the import process, and the data is then imported into the target table based on the compression characteristics of the table.

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US

blogs.oracle.com/oracle
facebook.com/oracle
twitter.com/oracle
oracle.com

Hardware and Software, Engineered to Work Together