

OLTP Compression Approach  
Oracle FLEXCUBE Universal Banking  
Release 14.3.0.0.0  
[May] [2019]



---

# Table of Contents

<b>1. INTRODUCTION .....</b>	<b>1-1</b>
1.1 INTRODUCTION TO ORACLE 11G ADVANCED OLTP TABLE COMPRESSION .....	1-1
1.2 SCOPE .....	1-1
1.3 ADVANTAGES .....	1-1
<b>2. REQUIREMENT OR PROBLEM STATEMENT .....</b>	<b>2-1</b>
<b>3. PREREQUISITES .....</b>	<b>3-1</b>
3.1 SOFTWARE REQUIRED .....	3-1
<b>4. ORACLE ADVANCED OLTP COMPRESSION DESCRIPTION.....</b>	<b>4-1</b>
4.1 CORE FEATURES .....	4-1
4.2 THRESHOLD BASED BLOCK COMPRESSION .....	4-2
4.3 HOW TO ENABLE TABLE COMPRESSION .....	4-3
4.4 LICENSING .....	4-3
4.5 APPROACH.....	4-3
4.5.1 <i>Data Preparation</i> .....	4-3
4.5.2 <i>Test Cases</i> .....	4-3
4.6 ENVIRONMENT USED.....	4-4
4.7 COMPARISON MEASUREMENT RESULTS .....	4-4
4.7.1 <i>Storage Consumed</i> .....	4-4
4.7.2 <i>Performance with Inserts</i> .....	4-4
4.7.3 <i>Performance with Updates</i> .....	4-4
4.7.4 <i>Performance with Deletes</i> .....	4-4
4.7.5 <i>Performance with Selects</i> .....	4-4
4.7.6 <i>Performance with Statistics Computation</i> .....	4-4
4.7.7 <i>Performance with Index Creation</i> .....	4-4
4.7.8 <i>Disk Reads</i> .....	4-4

---

# 1. Introduction

## 1.1 Introduction to Oracle 11g Advanced OLTP Table Compression

Oracle Advanced OLTP Table compression feature was introduced in version 11g and this feature allows data to be compressed during all DML operations. Oracle 11g uses a unique algorithm that eliminates duplicate values within a single block even across multiple columns. Similar to Oracle 9i, each block contains a symbol table that maintains compression metadata and this allows better performance compared to the global symbol table based algorithms.

To reduce the overhead during compression Oracle 11g uses a threshold based compression, which only compresses the data within a block once the threshold is reached, and thus eliminates the overhead to compress the data every time it is written.

## 1.2 Scope

This document brings out the feasibility of performing a table level compression using Oracle 11g Advanced OLTP table compression feature. This document also presents results of a test done using sample data with various operations and thus quantifies the benefits achieved in some of these operations.

## 1.3 Advantages

The benefits of using Oracle Advanced OLTP compression feature is much beyond just disk storage savings. One significant advantage is its ability to read compressed blocks directly without having to uncompress it. Therefore there is no measurable performance degradation during read operations on compressed tables.

---

## 2. Requirement or Problem Statement

The requirement is to display the capability of the Oracle OLTP table compression feature and to measure the performance with various operations such as Read, DML, Statistics collection etc.

---

## 3. Prerequisites

### 3.1 Software Required

Oracle Database Server 11g Release 1 ( 11.1.0.6 )

Download and install the Oracle Database server 11g version 11.1.0.6 from the below link

<http://www.oracle.com/technology/software/products/database/index.html>

## 4. Oracle Advanced OLTP Compression Description

### 4.1 Core Features

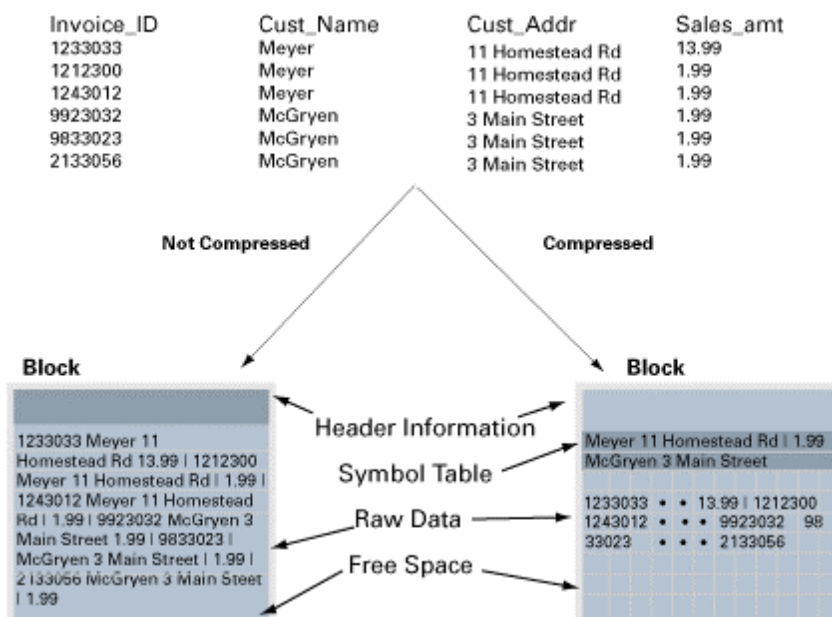
Oracle has been one of the pioneers in introducing database compression technology. Oracle database 9i introduced Table Compression several years ago to allow data to be compressed during bulk load operations which was ideally suited for data warehouse environments.

Oracle database 11g introduced a new feature called OLTP table compression. This allows data to be compressed during all types of data manipulation operations including conventional DML such as INSERT or UPDATE.

Oracle database 11g uses a unique compression algorithm specifically designed to work with relational data. 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 contained within 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.

Compressed Block vs. Noncompressed Block

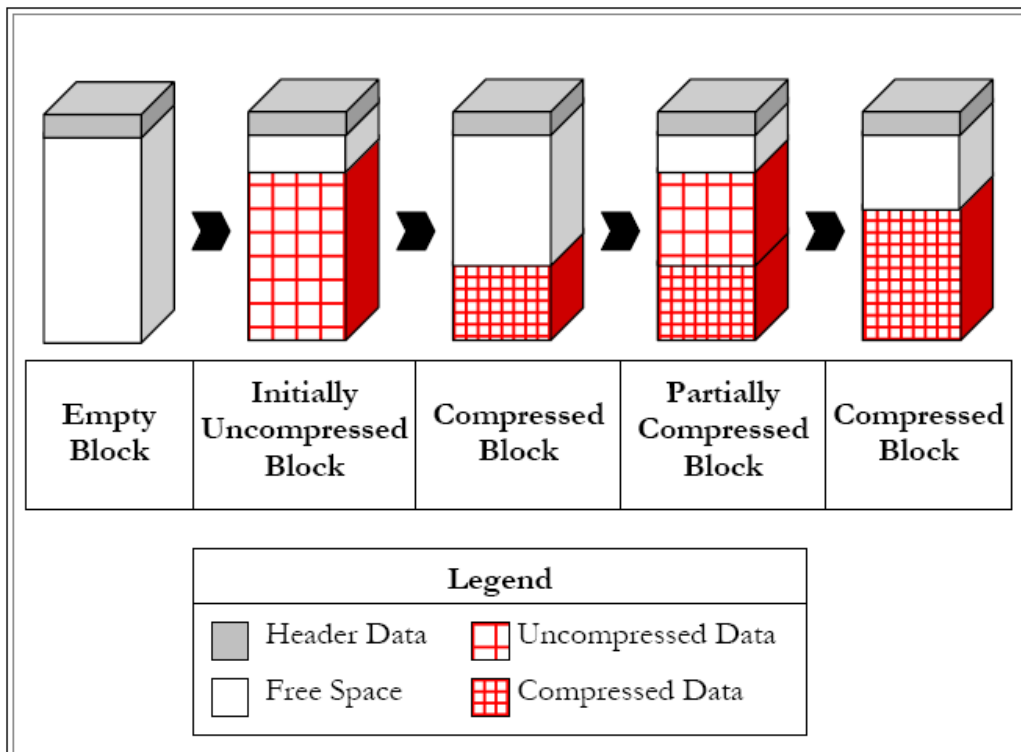


## 4.2 Threshold Based Block Compression

With Oracle's capability to read compressed blocks directly, there is no significant overhead in performance during read operations. However, compression requires additional work to be performed while writing the data making it unavoidable to eliminate performance overhead for write operations. To minimize this performance overhead, Oracle compresses a block in batch mode rather than compressing data every single 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 compression overhead. Therefore, a majority of OLTP transactions on compressed blocks will have the exact same performance as they would with uncompressed blocks.




## 4.3 How to Enable Table Compression

The following example shows how OLTP Table compression can be enabled while table creation:-

```
CREATE TABLE emp (  
Emp_id number,  
First_name varchar2 (20),  
Last_name varchar2 (20)  
) COMPRESS FOR ALL OPERATIONS;
```

The following example shows how OLTP Table compression can be enabled for an existing table:-

```
ALTER TABLE emp COMPRESS FOR ALL OPERATIONS;
```

 Issuing the above statement will not compress the existing records. Only the new records written to this table will get compressed.

## 4.4 Licensing

Oracle Table compression feature introduced in Oracle Database 9i is a base feature of Enterprise Edition (EE) and continues to be so even in Oracle Database 11g. The new OLTP Table Compression feature, however, is a part of the Oracle Advanced Compression option that **needs to be licensed in addition to the Enterprise Edition**. Performance Measurement on OLTP Compressed Table – a sample

A Sample test was done for various operations on Compressed and Uncompressed tables and a performance comparison was done between the two.

## 4.5 Approach

### 4.5.1 Data Preparation

- A Compressed table created with structure similar to dba\_objects.
- An Uncompressed table created with structure similar to dba\_objects.
- An Uncompressed table created and populated with 10 million rows out of which 40 thousand rows were distinct. This table was used to populate the above 2 tables.

### 4.5.2 Test Cases

Comparison between the Compressed and Uncompressed table has been done for the following cases:-

- Storage consumed
- Performance with Inserts
- Performance with Updates
- Performance with Deletes
- Performance with Selects
- Performance with Statistics computation



- Performance with Index creation
- Disk Reads

## 4.6 Environment Used

- Operating System - Linux x86\_64 2.6.18-53.el5
- Number of CPU's - 8
- Physical Memory - 16GB

## 4.7 Comparison Measurement Results

### 4.7.1 Storage Consumed

Storage of compressed table was **48% lesser** than uncompressed table.

### 4.7.2 Performance with Inserts

Inserts on compressed table were **3 times slower** than uncompressed table.

### 4.7.3 Performance with Updates

Updates on compressed table were **8 times slower** than uncompressed table.

### 4.7.4 Performance with Deletes

Deletes on compressed table were **15 percent faster** than uncompressed table.

### 4.7.5 Performance with Selects

Select on compressed table **took same time** as uncompressed table.

### 4.7.6 Performance with Statistics Computation

Computation of Statistics on compressed table was **5 percent slower** than uncompressed table.

### 4.7.7 Performance with Index Creation

Index creation on compressed table was **14 percent slower** than uncompressed table.

### 4.7.8 Disk Reads

Disk reads on compressed table were **48 percent lesser** than uncompressed table.



OLTP Compression Approach  
[May] [2019]  
Version 14.3.0.0.0

Oracle Financial Services Software Limited  
Oracle Park  
Off Western Express Highway  
Goregaon (East)  
Mumbai, Maharashtra 400 063  
India

Worldwide Inquiries:  
Phone: +91 22 6718 3000  
Fax:+91 22 6718 3001  
<https://www.oracle.com/industries/financial-services/index.html>

Copyright © [2007], [2019], Oracle and/or its affiliates. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

**U.S. GOVERNMENT END USERS:** Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate failsafe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

This software or hardware and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.