Oracle Advanced Compression: Throw Away Half of Your Disks and Run Your Database Faster

Sushil Kumar, Vineet Marwah

ORACLE®

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, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Agenda

- Data Growth Challenges
- Advanced Compression Feature Overview
  - Table Compression
  - Unstructured Data Compression
  - Backup Compression
  - Data Guard Network Transport Compression
- Competitive Analysis
- Questions and Answers

Challenges

- Explosion in Data Volumes
  - Government Regulations (Sarbanes-Oxley, etc)
  - User Generated Content (Web 2.0)
  - Application Consolidation
- IT Managers Must Support Larger Volumes of Data with Limited Technology Budgets
  - Need to optimize storage consumption
  - Also maintain acceptable application performance
- Intelligent and Efficient Compression Technology can Help Address These Challenges
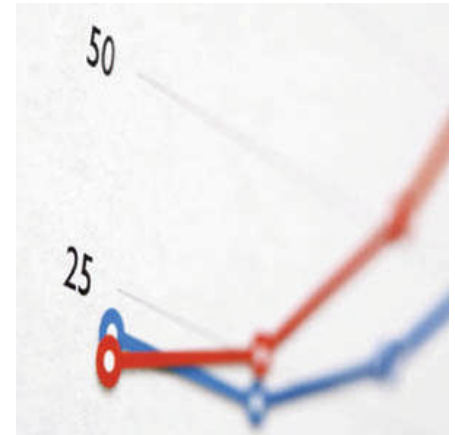
Introducing Advanced Compression Option

- Oracle Database 11g introduces a comprehensive set of compression capabilities
  - Structured/Relational data compression
  - Unstructured data compression
  - Compression for backup data
  - Network transport compression

- Reduces resource requirements and costs
  - Storage System
  - Network Bandwidth
  - Memory Usage

Table Compression

- Introduced in Oracle Database 9i Release 2
  - Compression during bulk load operations (Direct Load, CTAS)
  - Data modified using conventional DML not compressed
- Optimized compression algorithm for relational data
- Improved performance for queries accessing large amounts of data
  - Fewer IOs
  - Buffer Cache efficiency
- Data is compressed at the database block level
- Compression enabled at either the table or partition level
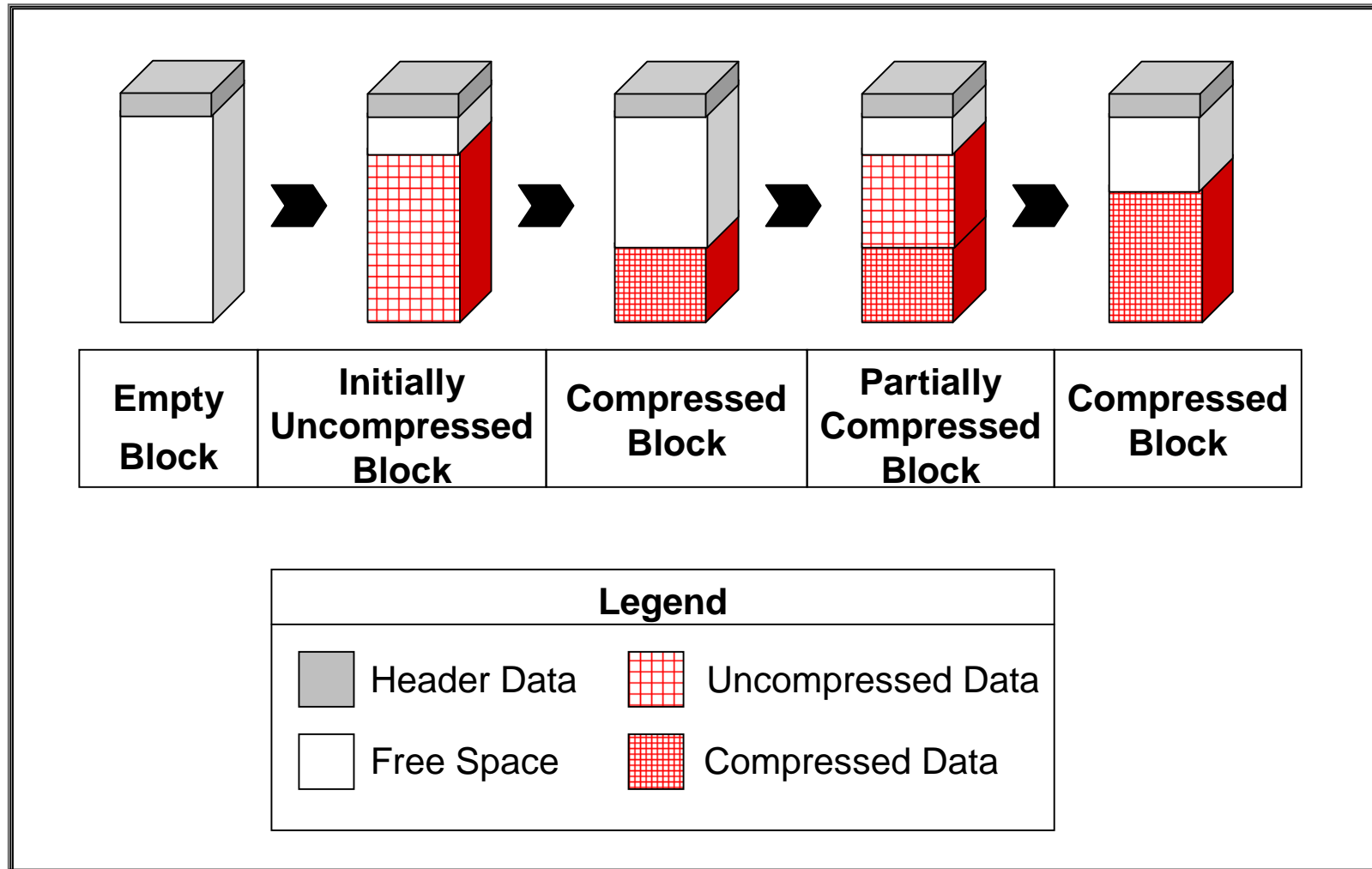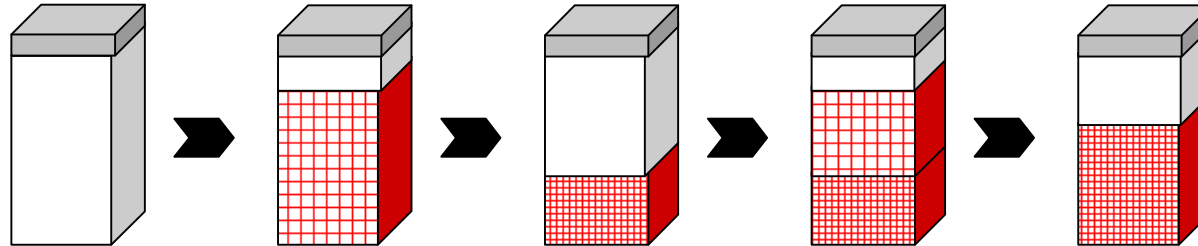- Completely transparent to applications

ORACLE®

OLTP Table Compression

- Oracle Database 11g extends table compression for OLTP data
  - Support for conventional DML Operations
    (INSERT, UPDATE)
- New algorithm significantly reduces write overhead
  - Batched compression ensures no impact for most OLTP transactions
- No impact on reads
  - Reads may actually see improved performance due to fewer IOs and enhanced memory efficiency

OLTP Table Compression Process

| Empty Block | Initially Uncompressed Block | Compressed Block | Partially Compressed Block | Compressed Block |

Legend

Header Data	Uncompressed Data
Free Space	Compressed Data

ORACLE

Block-Level *Batch* Compression

- Patent pending algorithm minimizes performance overhead and maximizes compression
- Individual INSERT and UPDATEs do not cause recompression
- Compression cost is amortized over several DML operations
- Block-level (Local) compression keeps up with frequent data changes in OLTP environments
  - Competitors use static, fixed size dictionary table thereby compromising compression benefits
- Extends industry standard compression algorithm to databases
  - Compression utilities such as GZIP and BZ2 use similar adaptive, block level compression

OLTP Table Compression

Employee Table

ID	FIRST_NAME	LAST_NAME
1	John	Doe
2	Jane	Doe
3	John	Smith
4	Jane	Doe

Initially Uncompressed Block

Header
1•John•Doe 2•Jane• Doe 3•John•Smith 4• Jane • Doe
Free Space

```
INSERT INTO EMPLOYEE
    VALUES (5, 'Jack', 'Smith');
COMMIT;
```

ORACLE

OLTP Table Compression

Employee Table

ID	FIRST_NAME	LAST_NAME
1	John	Doe
2	Jane	Doe
3	John	Smith
4	Jane	Doe
5	Jack	S

Compressed Block

Header

John=❶|Doe=❶|Jane=❷|Smith=❸

1•❶•❶ 2•❷•❶ 3•❶•❸ 4 • ❷
• ❶ 5•Jack•❸

Free Space

Local Symbol Table

ORACLE

OLTP Table Compression

Uncompressed Block

Header
1•John•Doe 2•Jane• Doe 3•John•Smith 4• Jane • Doe 5•Jack •Smith   Free Space

Compressed Block

Header
John=❶|Doe=❶|Jane=❷|Smith=❸
1•❶•❶ 2•❷•❶ 3•❶•❸ 4 • ❷ • ❶ 5•Jack•❸   Free Space

Local Symbol Table

More Data Per Block

Table Compression Syntax

OLTP Table Compression Syntax:

```
CREATE TABLE emp (
   emp_id NUMBER
 , first_name VARCHAR2(128)
 , last_name VARCHAR2(128)
 )COMPRESS FOR ALL OPERATIONS;
```

Direct Load Compression Syntax (default):

```
CREATE TABLE emp (
   emp_id NUMBER
 , first_name VARCHAR2(128)
 , last_name VARCHAR2(128)
 )COMPRESS [FOR DIRECT_LOAD OPERATIONS];
```

OLTP Table Compression
Best Practices

- Compress your 10 Largest Tables
  - 80/20 Rule – 20% of your Tables Consume 80% of your Space

- Better Compression with Bigger Block Sizes
  - Higher Probability of Duplicate Values

- B-Tree Index Compression
  - Validate Index and Review INDEX_STATS
    - INDEX_STATS.OPT_CMPR_COUNT
    - INDEX_STATS.OPT_CMPR_PCTSAVE

- Bitmap Indexes are Highly Compressed
  - Good for low and medium cardinality

ORACLE 11*g*
DATABASE

ORACLE

OLTP Table Compression
Getting Started with Compression

- Compression Advisor
  - Shows projected compression ratio for uncompressed tables
  - Reports actual compression ratio for compressed tables

OLTP Table Compression
Getting Started with Compression

- Monitoring Compression
  - DBA_TABLES
    - COMPRESSION (ENABLED/DISABLED)
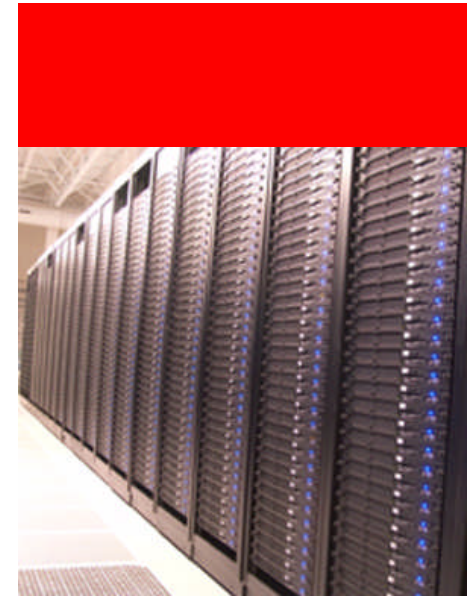   - COMPRESS_FOR (ALL OPERATIONS / DIRECT LOAD OPERATIONS)

```
oracle@obe11g:~/advisor
File   Edit   View   Terminal   Tabs   Help

SQL> SELECT table_name,compression,compress_for
  2      FROM user_tables
  3      WHERE compression = 'ENABLED';

TABLE_NAME                      COMPRESS COMPRESS_FOR
------------------------------- -------- ------------------
SALES_COMPRESS                  ENABLED  DIRECT LOAD ONLY
HZ_PARAM_TAB_C                  ENABLED  FOR ALL OPERATIONS

Elapsed: 00:00:00.29
SQL>
```
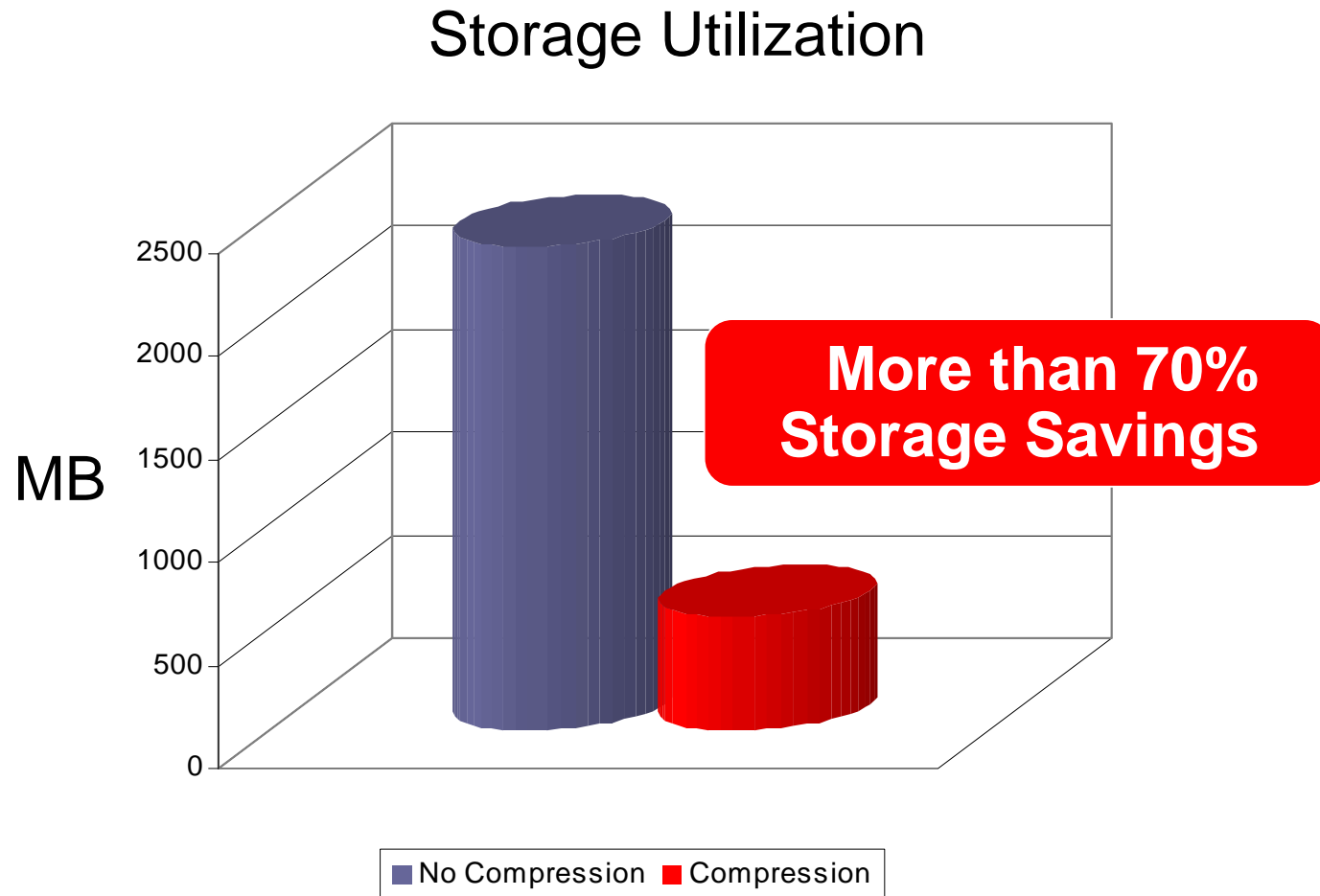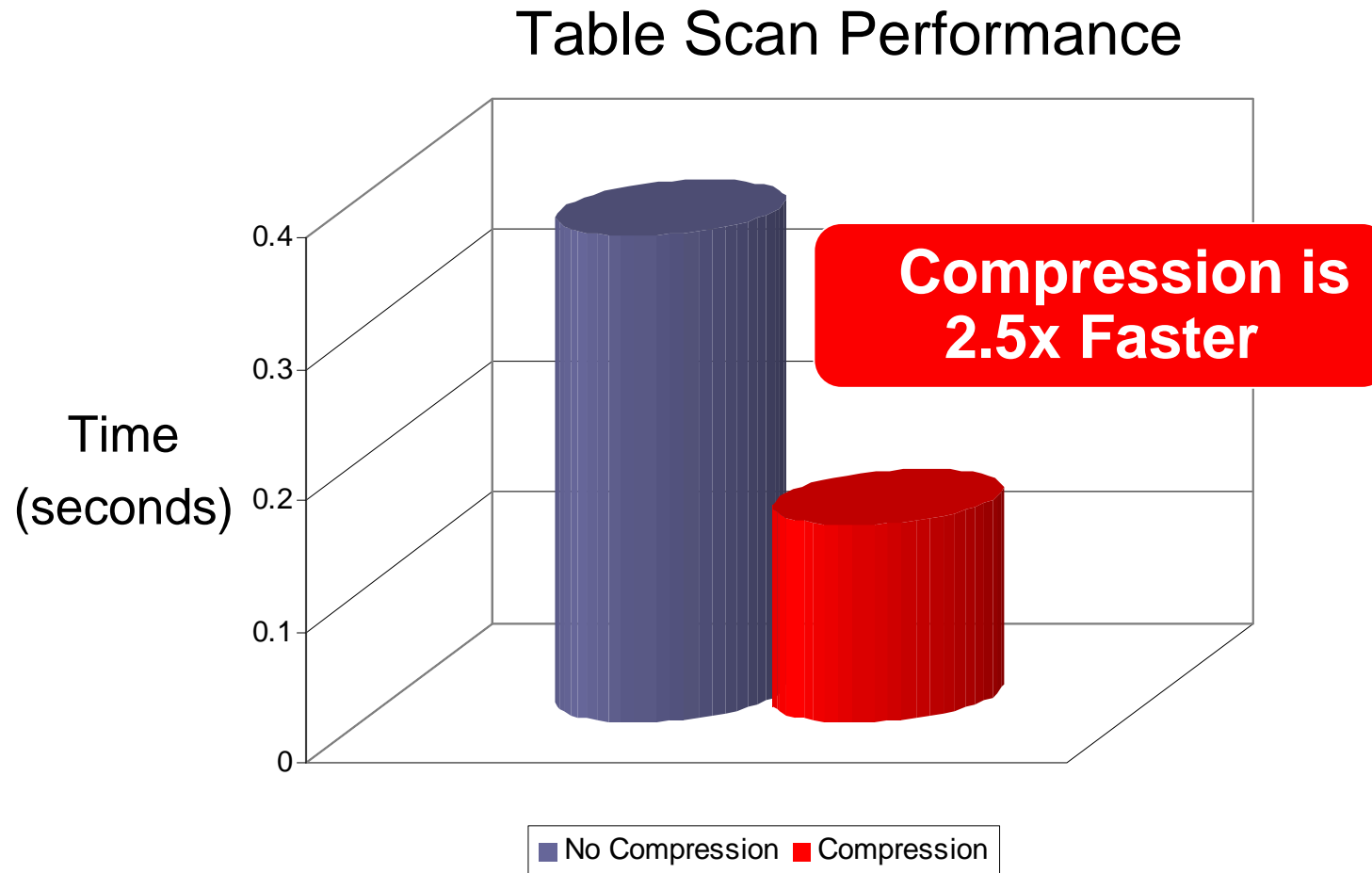
ORACLE

Table Compression Results

Real World Compression Overview

- Data from Oracle's implementation of Oracle Applications
  - Compressed the 10 Largest Tables
    - Two copies of each table: Compressed and Uncompressed
  - Oracle Enterprise Linux
  - Oracle Database 11g Release 1
- Test Queries
  - Full Table Scan
  - Index Range Scan
  - DML Operations (Insert, Update, Delete)

Real World Compression Results

Storage Utilization

MB

More than 70% Storage Savings

■ No Compression ■ Compression

ORACLE

Real World Compression Results

Table Scan Performance

Time (seconds)

Compression is 2.5x Faster

Legend: No Compression, Compression

ORACLE

Real World Compression Results

Table Scan Reads

Compression
performs
3.5x fewer reads

Physical Reads

14000
12000
10000
8000
6000
4000
2000
0

■ No Compression ■ Compression

ORACLE®

Real World Compression Results

Index Range Scan Performance

No Performance Impact on Index Scans

Time (seconds)

0.03
0.02
0.01
0

■ No Compression ■ Compression

ORACLE

Real World Compression Results

DML Performance

Time (seconds)

35
30
25
20
15
10
5
0

Compression has less than 3% overhead

Legend: ■ No Compression ■ Compression

ORACLE
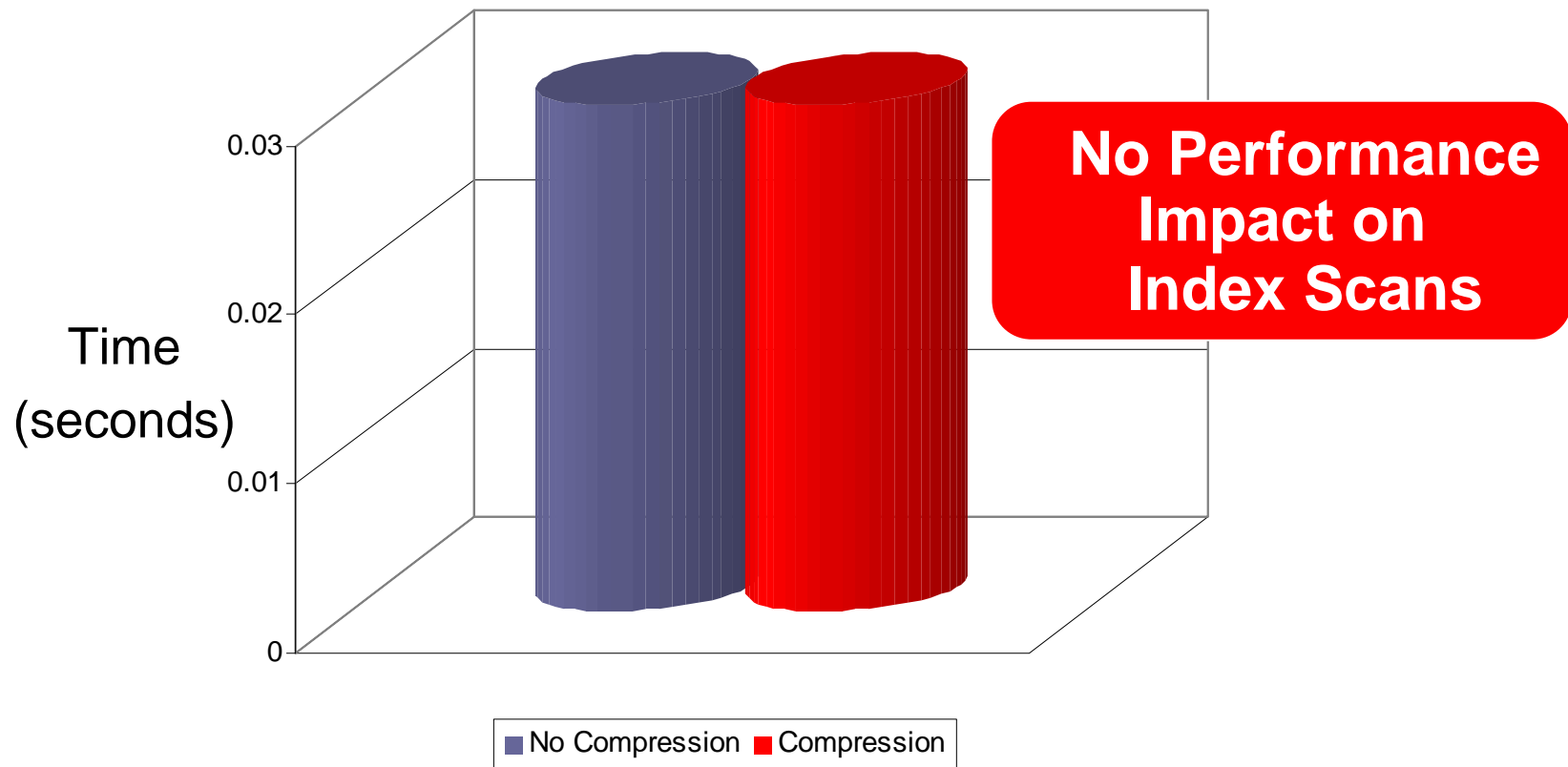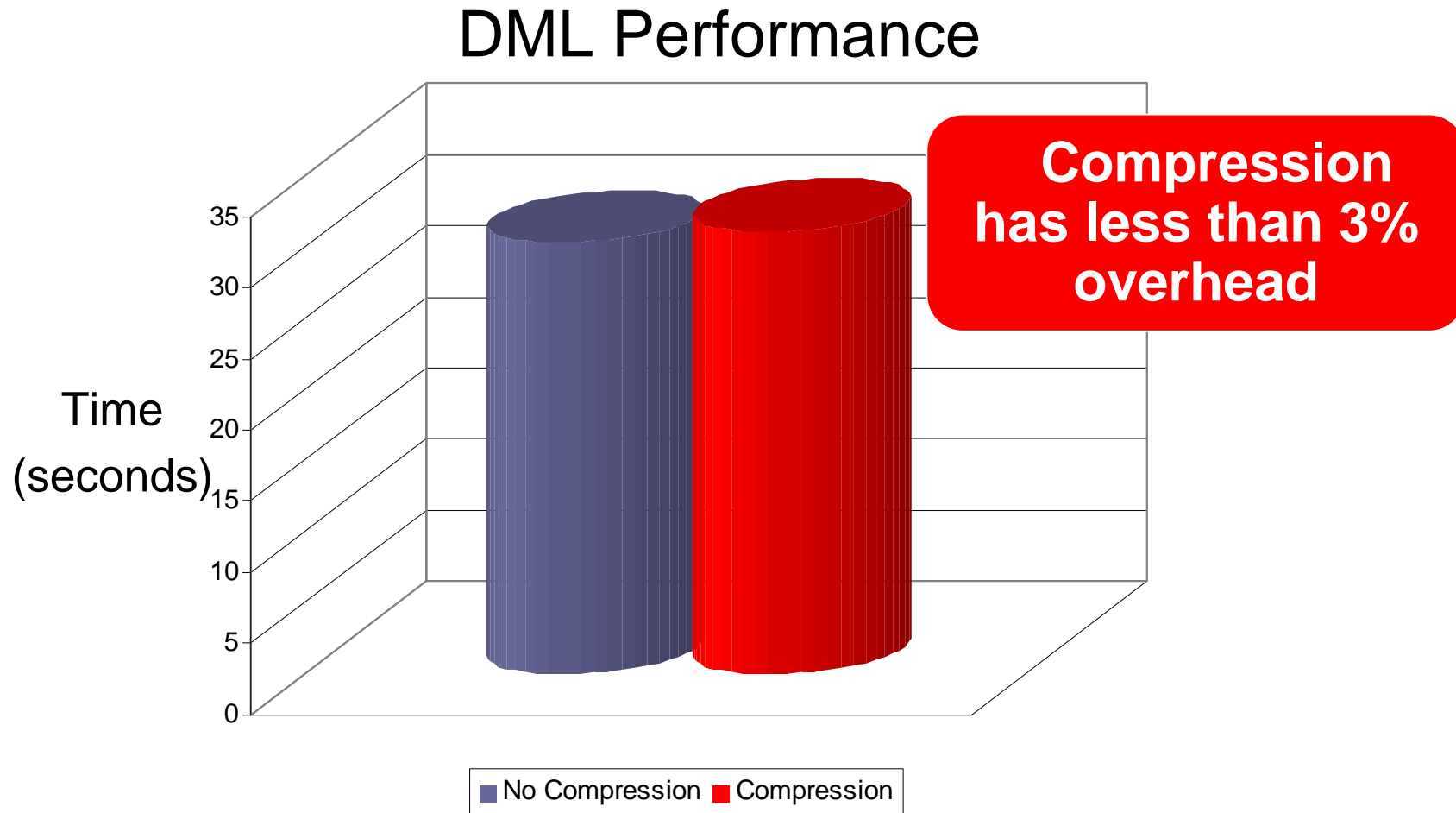
Introduction to SecureFiles

- SecureFiles is a new 11g feature designed to break the performance barrier keeping file data out of databases
- Next-generation LOB
  - Superset of LOB interfaces allows easy migration from LOBs
  - Transparent deduplication, compression, and encryption
  - Leverage the security, reliability, and scalability of database
- Enables consolidation of file data with associated relational data
  - Single security model
  - Single view of data
  - Single management of data
  - Scalable to any level using SMP scale-up or grid scale-out

SecureFiles Deduplication

Secure Hash
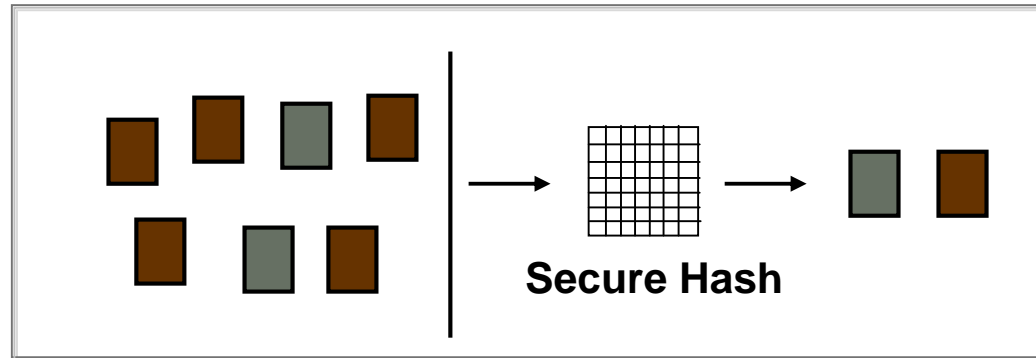
- Enables storage of a single physical image for duplicate data
- Significantly reduces space consumption
- Dramatically improves writes and copy operations
- No adverse impact on read operations
  - May actually improve read performance for cache data
- Duplicate detection happens within a table, partition or sub-partition
- Specially useful for content management, email applications and data archival applications

SecureFiles Compression

- Huge storage savings
  - Industry standard compression algorithms
  - 2-3x compression for typical files (combination of doc, pdf, xml)

- Allows for random reads and writes to Compressed SecureFile data

- Can be specified at a partition level

- Automatically detects if SecureFile data is compressible
  - Skips compression for already compressed data
  - Auto-turn off compression when space savings are minimal or zero

- Two levels of compression provide different compression ratios

- SecureFiles Compression is independent of table or index compression

SecureFiles Compression Syntax

Compression Syntax

```
CREATE TABLE docs (
doc_id NUMBER,
doc BLOB)
  LOB(image) STORE AS SECUREFILE
(TABLESPACE lob_tbs COMPRESS);
```

Deduplication Syntax

```
CREATE TABLE images (
image_id NUMBER,
image BLOB)
  LOB(image) STORE AS SECUREFILE
(TABLESPACE lob_tbs DEDUPLICATE);
```

Data Pump Compression

- Metadata compression available since Oracle Database 10g
- Oracle Database 11g extends compression to table data during exports
  - No need to decompress before import
- Single step compression of both data and metadata
  - Compressed data directly hits disk resulting in reduced disk space requirements
  - Internal tests reduced dump file size up to 75%
- Compression factor comparable to GNU gzip utility
- Application transparent
  - Complete Data Pump functionality available on compressed files

Data Pump Compression

- Performance cost: ~10% overhead
- Compression Ratio: comparable to gzip

Compression Method	OE/SH Schemas	Spatial Table	Spatial Schema
exdp compression=none	6.0 MB	26.6 MB	443 MB
expdp compression=all	1.5 MB (74.7%)	9.9 MB (62.7%)	140 MB (68.4%)
gzip –cv1	1.1 MB (82.7%)	11.4 MB (57.1%)	162 MB (63.4%)
gzip –cv6	835 KB (86.2%)	10.2 MB (61.7%)	142 MB (68.0%)
gzip –cv9	818 KB (86.5%)	10.1 MB (62.0%)	141 MB (68.2%)
compress	1.6 MB (74.2%)	13.8 MB (48.1%)	198 MB (55.3%)

Backup Compression

- Fast RMAN Compression
  - Compresses the backup set contents before writing them to disk or tape
  - No extra decompression steps are required during recovery when you use RMAN compression
  - High performance, industry standard compression algorithm
    - 40% faster backup compression versus Oracle Database 10g
  - Suitable for fast, incremental daily backups
  - Reduces network usage

Backup Compression Syntax

FAST RMAN Compression Configuration

```
RMAN> CONFIGURE COMPRESSION ALGORITHM 'zlib';
```
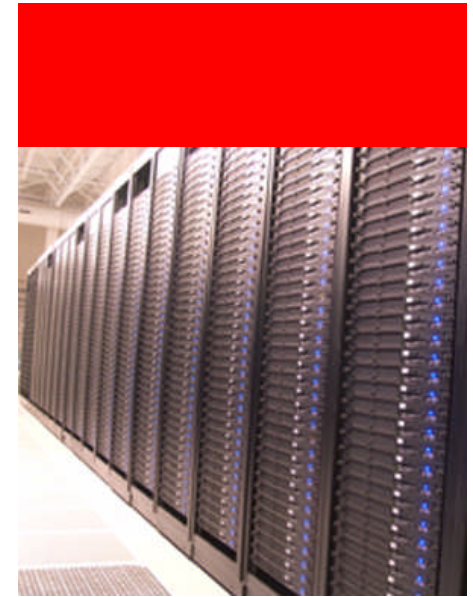
RMAN Compression Syntax

```
RMAN> backup as COMPRESSED BACKUPSET database
      archivelog all
```

DataPump Syntax

```
PROMPT> expdp hr FULL=y
      DUMPFILE=dpump_dir:full.dmp COMPRESS
```
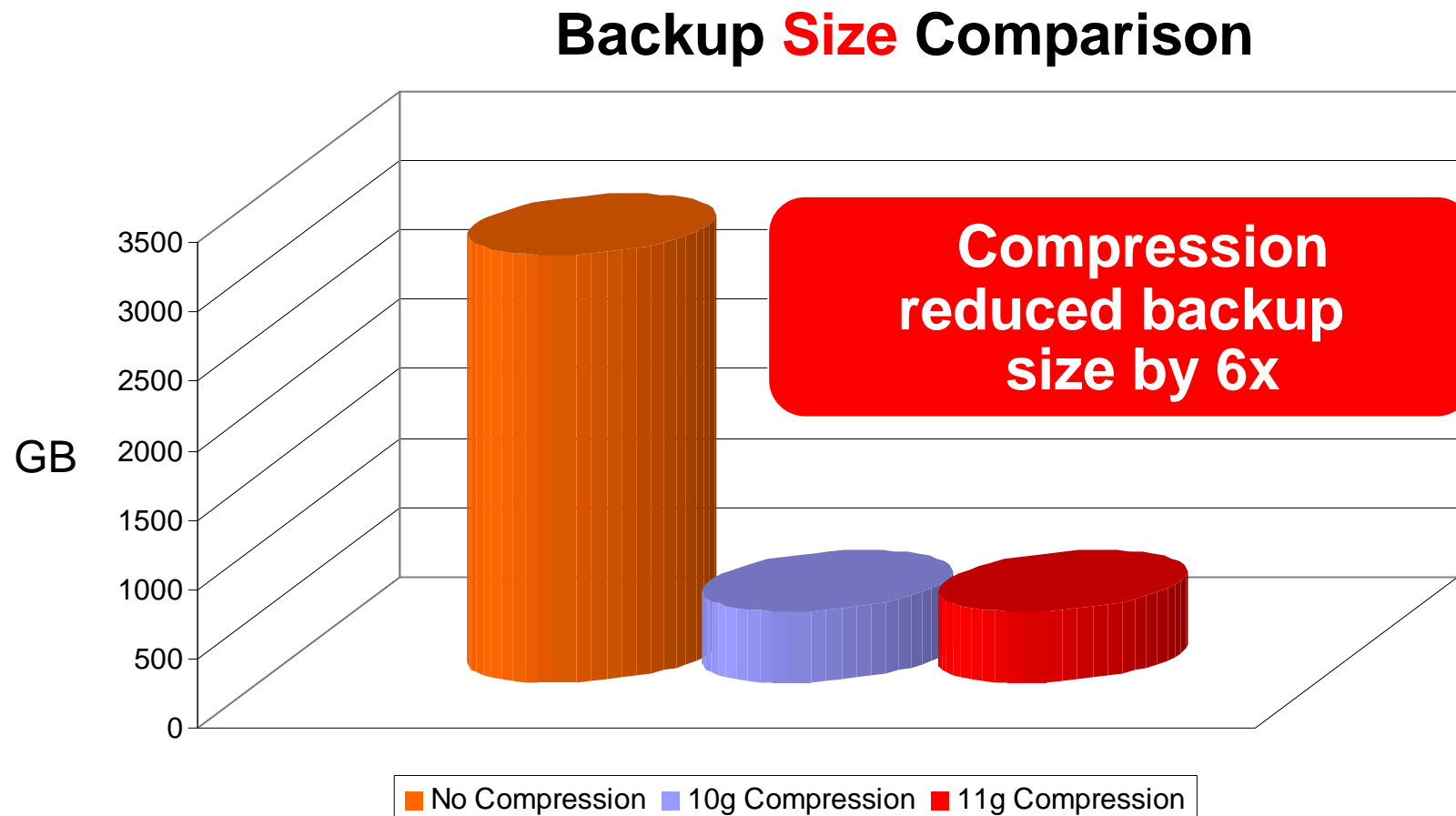
RMAN Compression Results
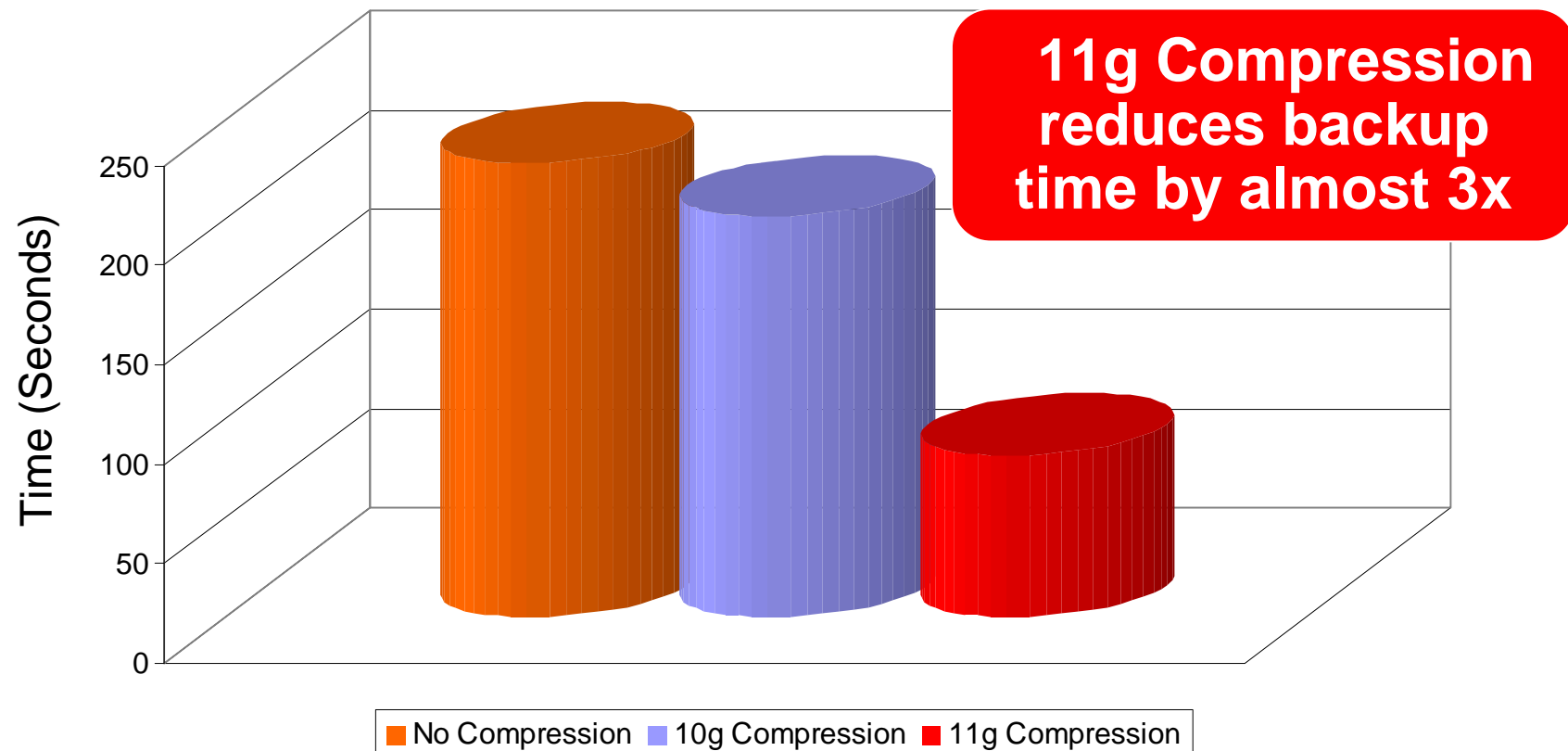
RMAN Compression Overview

- Oracle Applications Database
  - 3.5 GB Database
  - Oracle Enterprise Linux
  - Oracle Database 11g Release 1
- Test 1: Slow I/O (16 MB/s)
  - 11g RMAN without Compression
  - 10g RMAN with Compression
  - 11g RMAN with FAST Compression
- Test 2: Fast I/O (200 MB/s)
  - 11g RMAN without Compression
  - 10g RMAN with Compression
  - 11g RMAN with FAST Compression

Backup Compression Results

Backup **Size** Comparison

Chart axis label: GB

Y-axis values: 0, 500, 1000, 1500, 2000, 2500, 3000, 3500

Callout: **Compression reduced backup size by 6x**

Legend: No Compression — 10g Compression — 11g Compression

ORACLE

Backup Compression

**Backup Speed Comparison
Slow I/O (Tape)**

11g Compression reduces backup time by almost 3x

Time (Seconds)

250
200
150
100
50
0

■ No Compression ■ 10g Compression ■ 11g Compression

ORACLE

Backup Compression

Backup Speed Comparison
Slow I/O (Tape)

11g Compression is almost 2.5x faster than 10g Compression

Time (Seconds)

250

200

150

100

50

0

■ No Compression ■ 10g Compression ■ 11g Compression
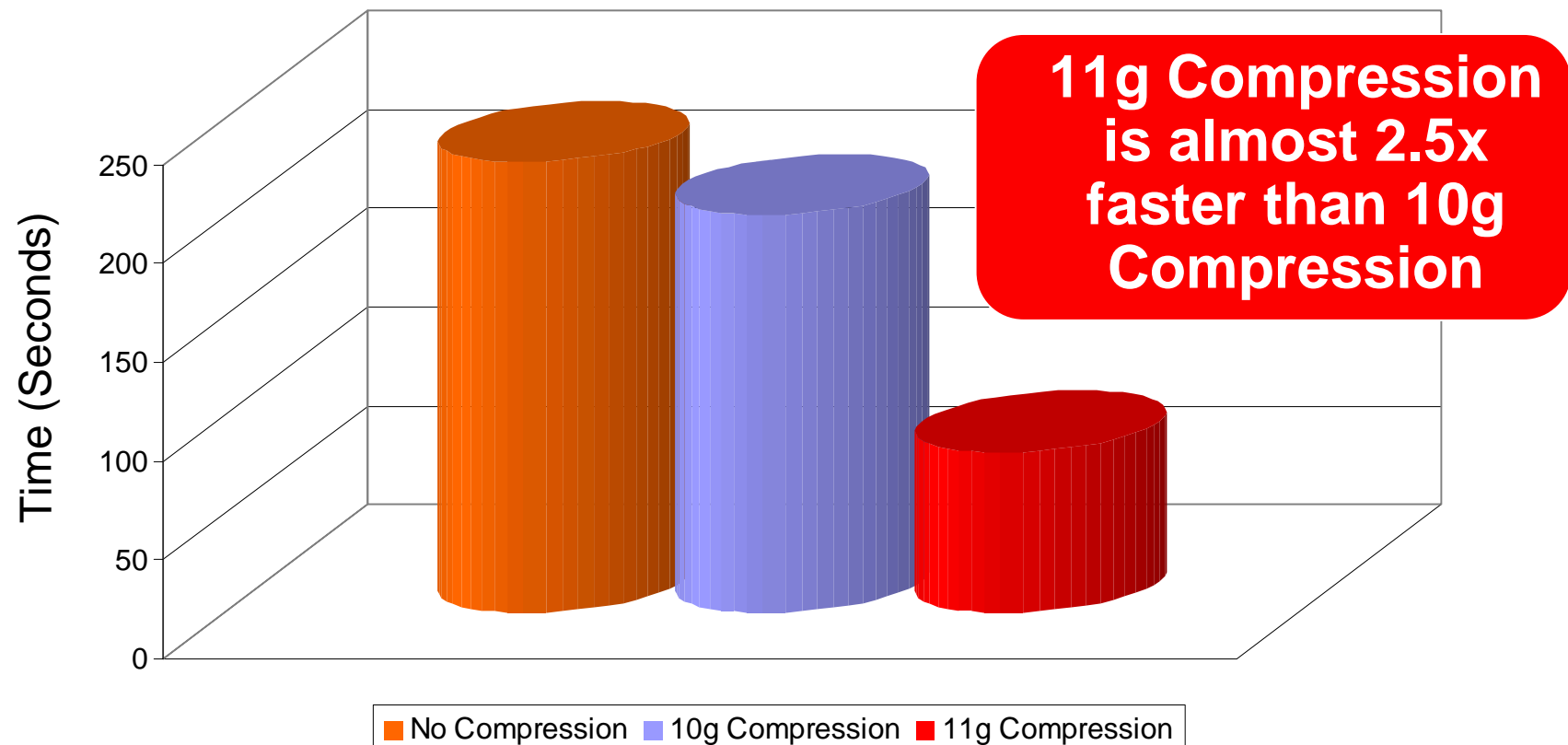
ORACLE®

Backup Compression

**Backup Speed Comparison
Fast I/O (Disk)**

11g Compression is almost 2.5x faster than 10g Compression

Time (Seconds)

250
200
150
100
50
0

■ No Compression ■ 10g Compression ■ 11g Compression

ORACLE

Oracle Data Guard

Primary Database   **Data Guard**   **Standby Database**
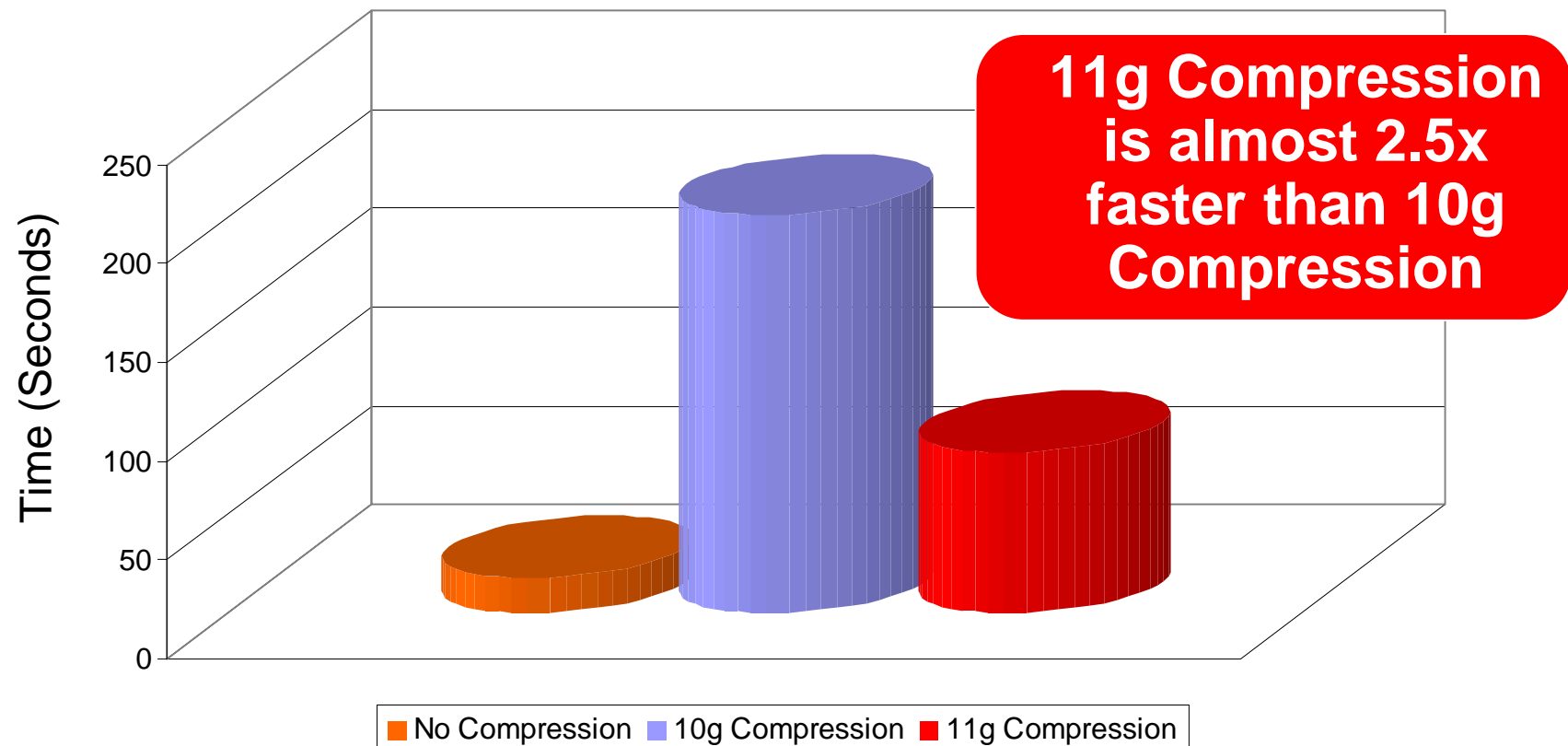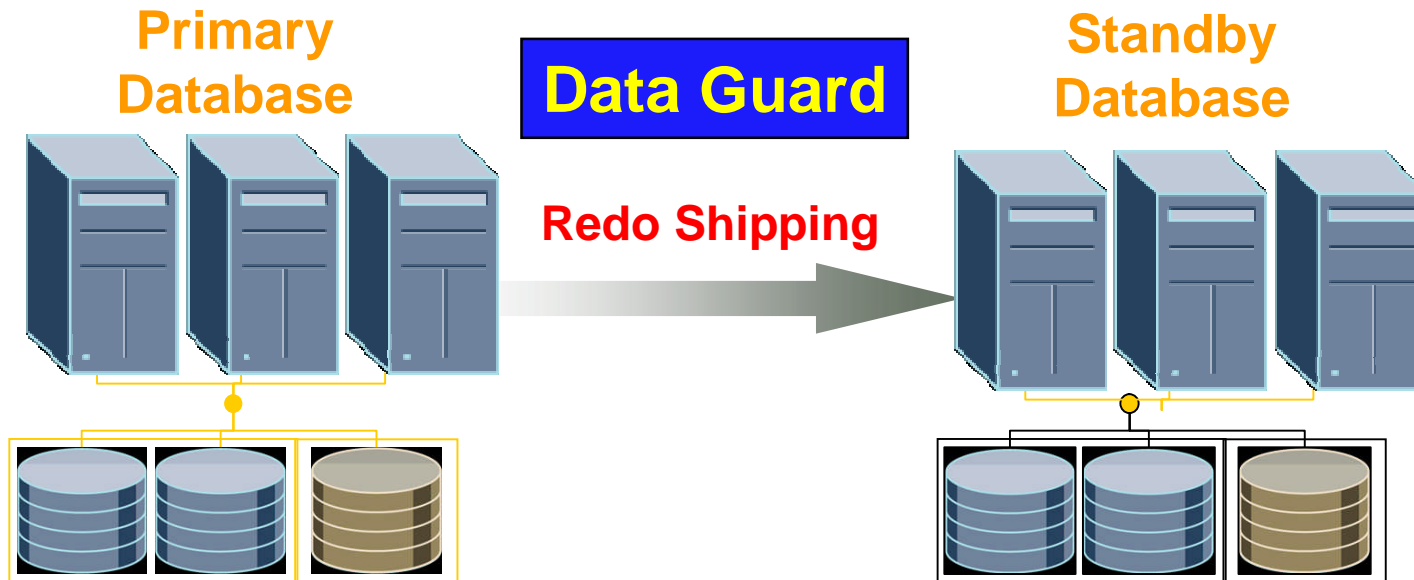
Redo Shipping

- Redo Shipping
  - Send redo data over network from primary to standby
  - Size of redo data typically small (transactional) and not network-bound
- Gap Resolution
  - After network outage – resynchronize standby
  - Size of redo data much larger

Network Compression
Oracle Data Guard Redo Transport Services

- Fast re-sync of standby database after network outages

- Lower bandwidth networks (<100Mbps)
  - 15-35% less time required to transmit 1 GB of data
  - Bandwidth consumption reduced up to 35%

- High bandwidth networks (>100 Mbps)
  - Compression will not reduce transmission time
  - But will reduce bandwidth consumption up to 35%

Competitive Analysis

Competitive Analysis – DB2

Oracle	IBM DB2
Block level compression	Table/partition level compression dictionary
- Adaptive / dynamic compression - Unlimited compression values	- New data may not be compressed / offline rebuild symbol table - Max 4K values per table
Zero decompression overhead	Some decompression overhead
- Minimal compression overhead - Batched compression	- More compression overhead - Transactional compression
Index compression	None
Unstructured data compression and de-duplication (SecureFiles)	None

ORACLE

Competitive Analysis – DB2

- Even without compression, Oracle needs less disk space than DB2:

    - Oracle uses variable length representation for numbers – DB2 uses fixed length

    - DB2 has more overhead for VARCHAR data

- SAP BW data needs 30-50% less disk space if stored in Oracle

Competitive Analysis – Teradata

Oracle	Teradata
Block level compression	Field level compression
- Dynamic compression values - Unlimited values	- Admin must define compression values - Max 255 compression values
All data types supported	Limited data type support
- Minimal compression overhead - Batched compression	- More compression overhead - Transactional compression
Backup compression	None
Unstructured data compression and de-duplication (SecureFiles)	None

Competitive Analysis – Hardware Based Compression

Oracle	Hardware Compression
Database specific compression algorithm	Generic compression algorithm - deficient for databases
Fine grained compression – compress at tablespace, table or partition level	No fine grained compression - compress at file level
Improved read performance in several cases – no decompression overhead	Poor read performance – significant decompression overhead
Improved memory efficiency - data stays compressed in memory	No memory benefits - data needs to be decompressed
Batched algorithm for minimal compression overhead	Data updates are very expensive
Benefits automatically cascade to all environments	Benefits limited to specific hardware environment

Competitive Analysis

Compression: Oracle vs Competition
Top 10 Tables in an ERP Application

ORACLE
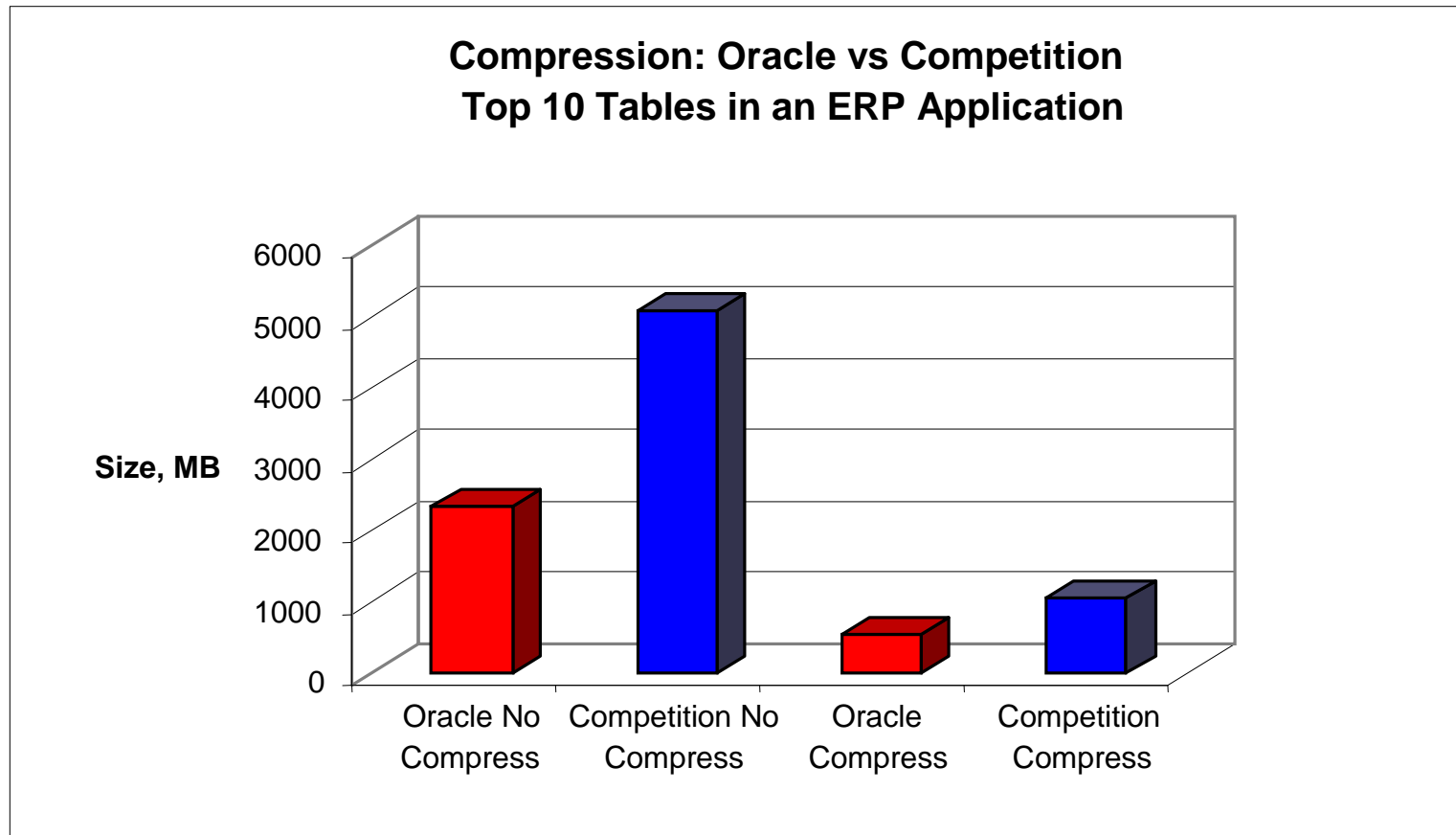
Summary

- Advanced Compression Option contains comprehensive data compression capabilities for all types of data
  - Structured, Unstructured, Backup, Network Transport
- Reduces storage consumption by 2 to 4 times
- Improves read performance
- Enhances memory, buffer cache utilization
- Complete application transparency
- Benefits diverse application workloads

ORACLE®