Oracle Compression for Infor ERP LN

be enterprising

Publication Information

Document code: B0050B US

Release: Infor ERP LN

Publication date: September 08

Table of Contents

About this Guide

This document explains the usage of Oracle compression techniques for Infor ERP LN to reduce the size and growth of customer databases.

Send us your comments

We continually review and improve our documentation. Any remarks/requests for information concerning this document or topic are appreciated. Please e-mail your comments to documentation@infor.com.

In your e-mail, refer to the document code and title. More specific information will enable us to process feedback efficiently.

Chapter 1 Summary

1

Introduction

This document explains the usage of Oracle compression techniques for Infor ERP LN to reduce the size and growth of customer databases.

Oracle compression has been available since Oracle 9.2 and has been improved in the latest Oracle 11g release.

This document is based on the compression features available in Oracle 10g.

Licensing

Compression is available starting from the Enterprise Edition level and is not available in the Standard Edition.

The new ‘Advanced compression feature’ in Oracle 11g is licensed separately in addition to the Enterprise Edition.

Test environment

Infor has tested Oracle table and index compression in an Infor ERP LN FP3 environment using an Oracle 10.2.0.3 database on IBM AIX p550 (4 * 1.65Ghz Power 5+ CPU).

Key findings

All findings are only based on the compression of the following tables: tfgld481, tfgld482, tfgld495, and tfgld498.

Disk space reduction

There is a large reduction in disk space for a number of tables in the tfgld range. The actual percentage will differ per customer based on data stored in the tables.

Table	Gain in diskspace	
	Table compression	Table + Index compression
	%	%
tfgld481	88	88
tfgld482	29	68
tfgld495	43	72
tfgld498	96	96

Performance penalty

The benchmark shows no measurable difference in CPU usage. Disk IO improves with compression enabled by average 20%. Average response times in the benchmark are 2% slower.

Performance of inserts is hardly affected. Performance of updates is slower.

Table	Performance degradation %	
	Table compression	Table + Index compression
	%	%
tfgld482 – update	26	9
tfgld482 - insert	2	2

There can be a performance penalty for index (range) scans incase of compressed indexes, because Oracle needs to rebuild the index keys.

Disadvantages

Increased database management and downtime

Compression involves more database management and downtime due to index rebuilds.

In Oracle10 added rows are not automatically compressed. A compression job needs to be scheduled on regular a basis to compress added records.

After each table compression, Oracle marks all indexes with the status UNUSABLE. Therefore, all indexes of a compressed table must be rebuilt. During the index rebuild, queries on the table will be very slow because Oracle cannot use the indexes.

Our measurements show that table compression and index rebuild/compression are very fast in Oracle10. For more details refer to chapter 2.2.

Restrictions on compressed tables

There are two DDL (data definition language) restrictions on compressed tables:

- It is not possible to add a column with a default value
- It is not possible to remove a column

In case of a data definition update, e.g. by installing a new feature pack, Infor ERP LN will reconfigure tables and if necessary add or remove columns. In case of compressed tables the reconfigure will fail with Oracle error ORA-39726.

A solution for this is to temporary uncompress the table and apply the reconfiguration. Then compress the table again and rebuild the indexes.

There is a bug in Oracle10.2 which does not allow the adding or removing of columns on compressed tables even after they are uncompressed. Oracle patch 6512811 needs to be installed to solve this.

Chapter 3 contains an example of decompressing a table.

Recommendation

Infor recommends enabling table and indexing compression only for the largest tables in the tfgld range:

Table	Data access	Comments
Tfgld481	Only inserts	From FP3 onwards replaced by table tfgld465
Tfgld465	Only inserts	Only available in ERP LN version FP3 and newer
Tfgld482	Inserts and updates	
Tfgld495	Inserts and updates	
Tfgld498	Only inserts	When ‘log all elements’ is disabled, this table is no longer being used

The general recommendation is to enable table and index compression for large transaction logging and history tables which have none, or limited, updates and deletes.

Note that compression is not always very effective. In our test to compress tdsls451 there was 0% disk space improvement for table compression and only 9% for index compression.

Infor does not recommend enabling compression on other ERP LN tables.

Chapter 2
Detailed Results

2

Disk space gain

The following is the disk space gain measured in the test environment per table:

Table TFGLD481 Logged elements, 102720 records

Segment	Char Bytes	Table compress Bytes	Table+Index compress Bytes
TTFGLD481100	284164096	30408704	30408704
TTFGLD481100$IDX1	5242880	5242880	3145728
Total size (MB)	276	34	32
Gain %	0	88	88

Table TFGLD482 Integration transactions, 1104908 records

Segment	Initial Bytes	Table compress Bytes	Table+Index compress Bytes
TTFGLD482100	573571072	75497472	75497472
TTFGLD482100$IDX1	44040192	59768832	34603008
TTFGLD482100$IDX2	114294784	173015040	75497472
TTFGLD482100$IDX3	191889408	284164096	83886080

TTFGLD482100$IDX4	83886080	121634816	51380224
Total size (MB)	961	681	306
Gain %	0	29	68

Table TFGLD495 Reconciliation data, 1107850 records

Segment	Initial Bytes	Table compress Bytes	Table+Index compress Bytes
TTFGLD495100	912261120	99614720	99614720
TTFGLD495100$IDX1	45088768	45088768	34603008
TTFGLD495100$IDX10	40894464	40894464	15728640
TTFGLD495100$IDX2	150994944	150994944	117440512
TTFGLD495100$IDX3	332398592	332398592	55574528
TTFGLD495100$IDX4	83886080	83886080	52428800
TTFGLD495100$IDX5	39845888	39845888	13631488
TTFGLD495100$IDX6	41943040	41943040	14680064
TTFGLD495100$IDX7	108003328	108003328	55574528
TTFGLD495100$IDX8	32505856	32505856	13631488
TTFGLD495100$IDX9	99614720	99614720	51380224
Total size (MB)	1800	1025	500
Gain %	0	43	72

Table TFGLD498 Logged elements, 1279994 records

Segment	Initial Bytes	Table compress Bytes	Table+Index compress Bytes
TTFGLD498100	3507486720	90177536	90177536
TTFGLD498100$IDX1	55574528	55574528	40894464
Total size (MB)	3398	139	125
Gain %	0	96	96

Time to compress and rebuild indexes

To indicate the expected downtime, the table below shows the duration of the table compressions and index rebuild in our environment. The server is an IBM AIX p550 server with 4 * 1.65 GHz CPU.

Table or index	Rows	Duration of table compress	Duration of index rebuild + compress
Tfgld498	1279994	71 sec	-
Tfgld498101$IDX1		-	5 sec
Tfgld481	102720	37 sec	-
Tfgld481101$iDX1		-	1 sec
Tfgld495	1107850	23 sec	-
Tfgld495$IDX1		-	4 sec
Tfgld495$IDX2		-	13 sec
Tfgld495$IDX3		-	19 sec
Tfgld495$IDX4		-	7 sec
Tfgld495$IDX5		-	5 sec
Tfgld495$IDX6		-	5 sec
Tfgld495$IDX7		-	10 sec
Tfgld495$IDX8		-	5 sec
Tfgld495$IDX9		-	10 sec
Tfgld495$IDX10		-	4 sec

Benchmark results

The standard ERP LN benchmark has been carried out to measure the differences between the initial, table compressed and table+index compressed Oracle10 database. Tables compressed are tfgld481, tfgld482, tfgld495, and tfgld498.

CPU usage

There is no measurable difference in CPU usage during the benchmark.

Scenario	Average CPU usage during the benchmark in %
Initial	29.5
Table compression	29.4
Table+index compression	29.8

DISK IO

Disk IO decreases significantly when compression of tables and indexes is enabled.

Scenario	Avg Disk read	Gain	Avg Disk write	Gain	Avg	Gain
	KB/sec	%	KB/sec	%	IO/sec	%
Initial	264		1857		145	
Table compression	219	17	1662	11	136	6
Table+index compression	219	17	1454	22	132	9

Response times

On average the response times in the benchmark are 2% slower with table and index compression enabled.

Scenario	Avg response time difference
Table compression	0%
Table+index compression	+2%

With table and index compression enabled, the transaction response times are in the range of 10% faster and up to a maximum of 9% slower.

Chapter 3
Deployment

3

Introduction

This chapter contains a description of the steps to compress a table and its indexes. Please note that an additional step is included to determine the actual number of fields to compress per index. This is an important step because compressing the wrong number of index field's results in less diskspace gain. The number of fields to compress per index differs based on the actual data in the table, so it needs to be determined in each ERP LN company.

Steps to compress a table and indexes

Table tfgld482 is used in the following example:

1 Determine all indexes of a table:

Select table_name, index_name from user_indexes where table_name = 'TTFGLD482%'.

2 Determine optimal number of fields to compress per index:

Validate index ttfgld482101$idx1

Select opt_cmpr_count from index_stats

3 Compress table:

Alter table ttfgld482<compnr> move compress.

At this point, all indexes of this table are marked unusable.

4 Compress all indexes using the calculated number of fields from (2).

Alter index ttfgld482<compnr>$idx1 rebuild compress <number from (2)>

To determine the initial and resulting size of a table and its indexes, complete the following step:

Select segment_name, bytes from user_segments where segment_name like 'TTFGLD482<compnr>%'

Steps to uncompress a compressed table

Table tfgld482 is used in the following example:

1. Uncompress table

   Alter table ttfgld482<compnr> move nocompress;

2. At this point all indexes of this table are marked unusable. Use this query to find out all indexes of a table:

   select index_name, status from dba_indexes where table_name='TTFGLD482<compnr>';

3. Rebuild all indexes (online)

   Alter table ttfgld482<compnr>$idx1 rebuild;