Infor LN Data Compression

Publication Information

Release: Infor LN 10.x

Publication date: May 2, 2014

Document code: B0050D

Contents

About this guide

From porting set 8.9a onwards, Infor LN supports table compression for IBM DB2, Microsoft SQL Server and Oracle database to reduce the size and growth of customer databases. Index compression is supported from porting set 9.0a. This document explains the data compression techniques and how they are implemented in Infor LN. Also the results of benchmark activities on compressed data are published in this document.

Intended audience

This document is intended for Infor LN database Administrators and Technical Consultants who are interested in reducing the size and growth of the database by using data compression.

Contacting Infor

If you have questions about Infor products, go to the Infor Xtreme Support portal at www.infor.com/inforxtreme.

If we update this document after the product release, we will post the new version on this Web site. We recommend that you check this Web site periodically for updated documentation.

If you have comments about Infor documentation, contact documentation@infor.com.

Chapter 1 General

1

Introduction

Some Infor LN customers are experiencing an enormous growth in the volume of data. Most database vendors do support data compression to reduce the size and growth of the database while application scalability and performance remains the same or even improves.

Data compression has these benefits:

- Significant savings in disk storage space
- Fewer page reads, because more rows can fit on a page
- Reduced disk I/O activity, because more compressed rows than uncompressed rows fit on a page
- Ability to compress older data that is not often accessed, such as archived tables, while leaving more recent data that is frequently accessed in uncompressed form
- Possibility to free space no longer required for a table
- Faster backup and restore

This document describes how Infor LN supports data compression for IBM DB2, Microsoft SQL Server, and Oracle database. Infor executed benchmarks on these databases measuring the impact of Infor LN data compression. The results of these benchmarks are published in this document.

Deployment

From porting set 8.9a Infor LN supports table compression in SQL Server, Oracle, and DB2 databases. From porting set 9.0a also index compression is supported.

Note: A database edition that supports compression is required. For more information see the latest porting set technical notes.

In case of a new Infor LN installation, you can enable table compression in the installation wizard. You can also enable table compression in an existing Infor LN installation. To achieve this, complete these steps:

1 Start the Storage Parameters Optimization (ttdba0132m000) session and switch table compression on. It is possible to enable table compression on specific tables only.

2 Convert the changes to runtime. The <database>_storage_param file is updated.

Then during the creation of a new table, or reconfig with dump of existing tables, compression will be used for these tables. To force compression on existing tables, run the Reorganize Tables (ttaad4225m000) session. Tables must be recreated to enable compression. Therefore you must select the **Data and Indices** check box. Note that this step incurs application downtime.

Support and Licensing

Data compression can be an optional or licensed feature by the database vendor. Ensure you have the required version, edition, or license installed, before using data compression with Infor LN. For more information see the latest porting set technical notes.

Benchmark

Infor executed a benchmark on DB2, Oracle, and Microsoft SQL Server to get more insight in the compression ratio, data growth, and performance of Infor LN running on a database:

- without compression,

- with table compression or

- with table and index compression.

The benchmark is executed with the standard Infor LN FP6 benchmark kit, running an online transaction processing (OLTP) and batch workload. The OLTP workload simulates sales, purchase, service, finance, and browsing business scenarios with high active concurrent users creating orders, entering delivery information, and performing other typical sales and manufacturing tasks. The dataset used for the benchmark contains almost 20,000 tables with data to simulate five representative customer business processes.

The results of this test are documented in the following chapters. Note that it does not make sense to compare compression figures between the three different databases, because there are differences in the used hardware and settings. Also, the figures will give an indication about the possible space savings. Your mileage may vary as the ratios highly depend on the dataset.

Note that the tests are done on a database with the varchar datatype. We recommend migration to varchar when still using the char datatype. Moving to varchar significantly reduces the database storage and can be done together with data compression.

Infor recommends testing these options in a test environment with your own specific data before implementing this in production.

Chapter 2 Oracle

2

General

Infor LN supports page compression on Oracle. To measure the performance impact of Oracle page compression, a benchmark has been executed on an Intel Xeon X5650, 2.67GHz, 12 core system with Oracle 11.2.0.4 database. The default database block size of 8k is used during the benchmark. The next sections describe the results on disk space including data growth, CPU usage, disk I/O, and response times.

Infor LN only supports **advanced** OLTP table compression on Oracle databases.

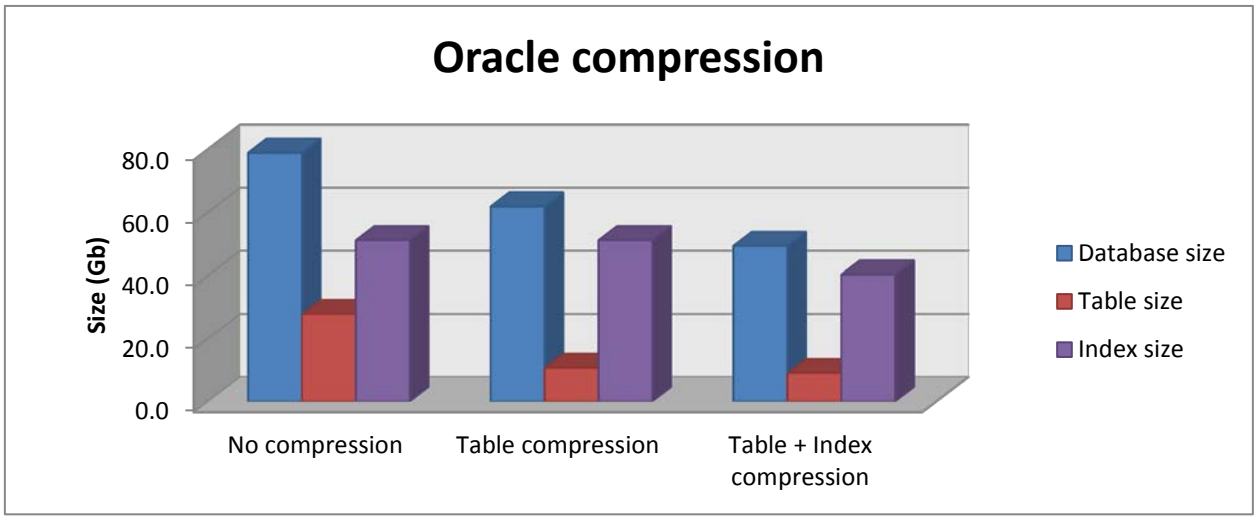
Note: Index compression is possible and supported by Infor LN porting set 9.0a and later with Oracle database. However, new inserted data will not be compressed in the index. Therefore, **indexes should be recompressed manually on a regular basis**. Also, unique single- and multi-column indexes cannot be compressed, as it would result in an index structure that increases rather than decreases in size due to the extra overheads associated with having a prefix index entry for every index row entry. Oracle will generate “ORA-25193: cannot use COMPRESS option for a single column key” when attempting to compress such an index. The Infor LN porting set will not compress unique single- and multi-column indexes.

Space usage and growth

The Infor LN database was initially populated with a representative dataset. This database has been compressed using table compression and also table + index compression. This table shows the space savings when using Oracle data compression:

	No compression	Table compression	Table + Index compression
Database size (Gb)	**86.8**	**69.7**	**56.9**
Table size (Gb)	27.7	10.6	10.6
Index size (Gb)	59.1	59.1	46.3
Compression ratio	-	1.25	1.53

This graph visualizes the Oracle compression benefits for the database size:



A 400 user benchmark has been executed on this database. This table shows the database size after the benchmark of 1 hour:

	No compression	Table compression	Table + Index compression
Database size (Gb)	**89.0**	**71.4**	**58.2**
Table size (Gb)	28.3	10.7	10.7
Index size (Gb)	60.7	60.7	47.5
Compression ratio	-	1.25	1.53
Data growth (Gb)	2.2	1.7	1.3

Benchmark results

The standard Infor LN OLTP benchmark running 400 users has been executed to measure the differences between the initial, table compressed, and table + index compressed database.

CPU usage

The measured CPU overhead during the benchmark is very low when using Oracle compression. The following table shows the average total CPU usage during a stable period of the benchmark for the three scenarios:

Scenario	Average CPU usage (%)
Baseline (No compression)	31.9
Table compression	29.5
Table + Index compression	32.2

DISK I/O

The following table shows the disk I/O during the benchmark. Depending on the data, Oracle compresses often data after the actual insert/update/delete statements, which can explain the low benefit on disk I/O during the benchmark when using Oracle compression.

Scenario	Avg Disk Read (KB/sec)	Benefit to base (%)	Avg Disk Write (KB/sec)	Benefit to base (%)	Avg IOPS	Benefit to base (%)
Baseline (No compression)	821	-	7968	-	458	-
Table compression	922	-12%	9392	-18%	488	-7%
Table + Index compression	721	12%	8903	-12%	445	3%

Notes

- For customers running Oracle 11g, at least patchset **11.2.0.4 must be installed**. Other critical patches required for Oracle 11g compression are listed in metalink note #1061366.1.
- Each time a block is compressed successfully, an image of the block prior to compression must be retained in undo for consistent read and transaction rollback. Therefore the amount of undo created when updating a table set for OLTP compression will be greater than the amount of undo created for a non-compressed table. Likewise, whatever is written to undo is also written to the redo logs, so there is a corresponding increase in redo when OLTP compression is used. Therefore we recommend the following:
  - Ensure to have enough redo log groups. A good starting point for 10-1000 users is six redo log groups with at least 1 file of 1 GB.
  - Ensure to have enough space in the UNDO tablespace when bulk loading data.
- Leave sufficient free space in the data and index blocks so updates that lengthen the row fit within the block and prevent row migrations. To do this, ensure PCTFREE is specified in lib/ora_storage_param with at least a value of 10. For example:

  ```
  *:*:*:COMPRESS=1;TABLESPACE data PCTFREE 10 INITRANS 3
  ```
- Oracle compression-specific white papers are accessible via metalink note #1329441.1.

Chapter 3 SQL Server

3

General

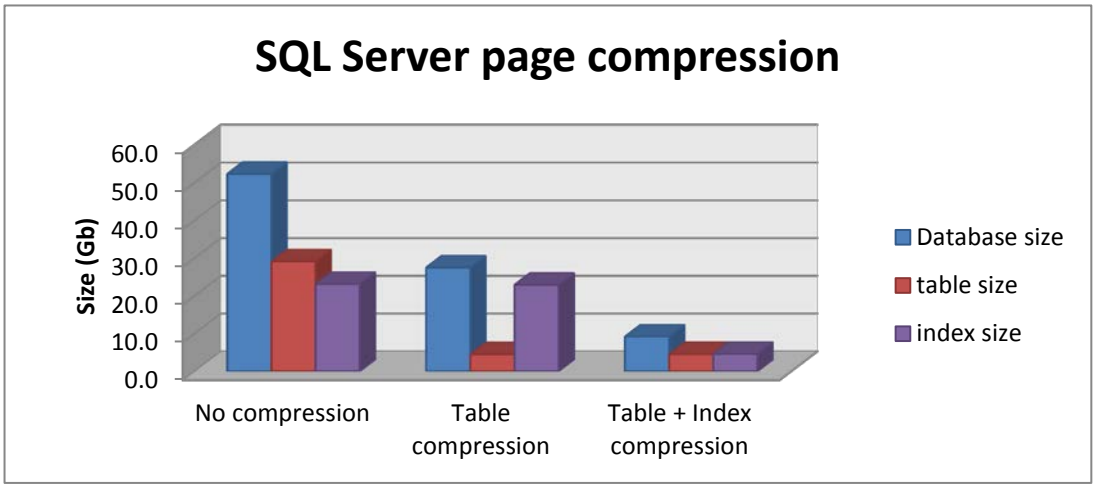
Infor LN supports page compression on SQL Server. To measure the performance impact of SQL Server page compression, a benchmark has been executed on an Intel Xeon X5650, 2.67GHz, 12 core system with SQL Server 2012 database. The default database block size of 8k is used during the benchmark. The next sections describe the results on disk space including data growth, CPU usage, disk I/O and response times.

Space usage and growth

The Infor LN database was initially populated with a representative dataset. This database has been compressed using table compression and also table + index compression. This table shows the space savings when using SQL Server data compression:

	No compression	Table compression	Table + Index compression
Database size (Gb)	**52.4**	**27.5**	**9.2**
Table size (Gb)	29.2	4.6	4.6
Index size (Gb)	23.2	22.9	4.6
Compression ratio	0	1.9	5.7

This graph visualizes the SQL Server compression benefits for the database size:



A 400 user benchmark has been executed on this database. This table shows the database size after the benchmark of 1 hour:

	No compression	Table compression	Table + Index compression
Database size (Gb)	**55.8**	**30.1**	**9.9**
Table size (Gb)	31.0	5.3	5.3
Index size (Gb)	24.8	24.8	4.6
Compression ratio	-	1.88	5.6
Data growth (Gb)	3.4	2.6	0.7

Compression time

On our benchmark server, it took approximately 50 minutes to add index compression on all 20,000 tables. The tables we loaded with table compression via Infor LN. This time may vary a lot, depending on the customer dataset and available hardware resources.

Benchmark results

The standard Infor LN OLTP benchmark running 400 users has been executed to measure the differences between the initial, table compressed and table + index compressed database.

CPU usage

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

Scenario	Average CPU usage (%)
Baseline (No compression)	32.7
Table compression	32.2
Table + Index compression	32.2

DISK I/O

Disk I/O decreases significantly when compression of tables and indexes is enabled.

Scenario	Avg Disk Read (KB/sec)	Benefit to base (%)	Avg Disk Write (KB/sec)	Benefit to base (%)	Avg IOPS	Benefit to base (%)
Baseline (No compression)	349	-	6894	-	610	-
Table compression	89	74%	5412	21%	500	18%
Table + Index compression	26	93%	4151	23%	391	36%

Chapter 4 DB2

4

General

Infor LN supports compression on IBM DB2. Note that indexes created afterwards on compressed tables will be compressed as well on DB2. To measure the performance impact of DB2 compression, a benchmark has been executed on an IBM Power7, 3.55 Ghz, quad core system with DB2 v10.1 database. The default database block size of 8k is used during the benchmark. The next sections describe the results on disk space including data growth, CPU usage, disk I/O and response times.

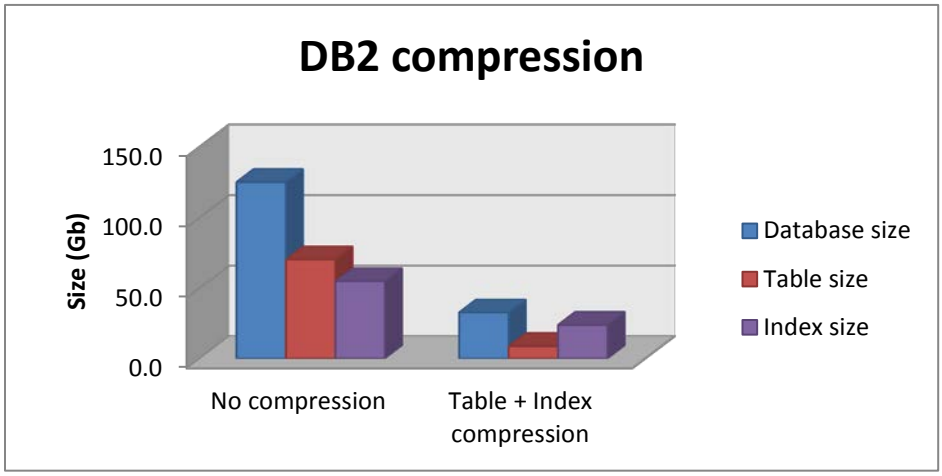
To enable compression on DB2 databases prior to version 10.5, an edition containing the Storage Optimization feature pack is required. Make sure you enabled the license for the Storage Optimization Package before you start the LN installation.

Space usage and growth

The Infor LN database was initially populated with a representative dataset. This database has been compressed using table + index compression. This table shows the space savings when using DB2 data compression:

	No compression	Table + Index compression
Database size (Gb)	**125.2**	**32.7**
Table size (Gb)	70.3	8.8
Index size (Gb)	54.9	23.9
Compression ratio	-	3.8

This graph visualizes the DB2 compression benefits for the database size:



A 400 user benchmark has been executed on this database. This table shows the database size after the benchmark of 1 hour:

	No compression	Table + Index compression
Database size (Gb)	**127.6**	**33.4**
Table size (Gb)	71.5	9.0
Index size (Gb)	56.1	24.4
Compression ratio	-	3.8
Data growth (Gb)	2.4	0.7

Benchmark results

The standard Infor LN OLTP benchmark running 400 users has been executed to measure the differences between the initial and table + index compressed database.

CPU usage

There is almost no difference in CPU usage during the benchmark. When using compression, the CPU usage is even a little bit lower.

Scenario	Average CPU usage (%)
Baseline (No compression)	87.5

Scenario	Average CPU usage (%)
Table + Index compression	85.9

DISK I/O

Disk I/O decreases significantly when compression of tables and indexes is enabled.

Scenario	Avg Disk Read (KB/sec)	Benefit to base (%)	Avg Disk Write (KB/sec)	Benefit to base (%)	Avg IOPS	Benefit to base (%)
Baseline (No compression)	2480	-	4033	-	794	-
Table + Index compression	142	94%	1851	54%	320	60%

Chapter 5 Conclusion

5

Data compression can be very beneficial to help customers reducing the growth in the volume of data. The research documented in this paper shows that data compression can deliver disk storage space savings up to 500%, reduced disk I/O activity up to 60% while the data growth decreases with 80%. Next to that, in most cases, there is no performance penalty, but even a performance increase.