

New Features, Design, & Tuning Tips for Teradata 15.10

TERADATA®

XTIVIA TERADATA TEAM

A Publication of
XTIVIA

TABLE OF CONTENTS

1. Introduction to some new features.....	03
- Teradata JSON Support (v 15.0)	03
- Teradata Directory Manager (v 15.0)	03
- Auto Provisioning (v 15.10)	03
- Intelligent Memory Improvements (v 15.10)	03
- Partition-Level Locking (v 15.10)	04
- Teradata Secure Zones (v 15.10)	04
- New PI on Access Rights Table (v 15.10).....	04
2. Some Design and Tuning Tips.....	05
- Reduce Lock Contention	05
- The Primary Index (PI) Choice	05
- Partition Primary Index (PPI / MLPPI)	05
- Other Indexing Choices	06
- It's all in the Statistics	06
- Query Rewriting	07
3. XTIVIA Tools & Frameworks	08
- MVC Tool	08
- Table Skew Analyzer	08
- Object DDL Comparison Tool.....	08
- Tuning Environment Creator	08

1. Introduction to some new features (Teradata v 15.0, v 15.10)

Teradata JSON Support (v 15.0):

Teradata added JSON support from v15.0 allowing the storage and processing of JSON objects in Teradata database. This allows schemaless reading and processing of semi-structured data. We can also publish the results of SQL queries in JSON format.

Teradata Directory Manager (v 15.0):

Teradata provides a GUI tool TDM (Teradata Directory Manager) allowing easy mapping of Teradata objects to LDAP directory users. To use TDM, one needs to have a dedicated directory exclusively to be used by Teradata database. This reduces the impact on enterprise LDAP. TDM is available for free and supports all Teradata versions that can use Teradata 14 JDBC drivers.

Auto Provisioning (v 15.10):

Effective v15.10, a database user is automatically created for external users (directory users not mapped to Teradata database user) upon their first login. In previous releases, a generic Teradata pseudo-user EXTUSER was used to identify external users which means individual user activity could not be logged and tracked. EXTUSER could not own database objects. With the Auto Provisioning feature, authentic LDAP users get quick database logon without Administrator's intervention. This feature is turned off by default and it can be turned on by setting the DBSControl AutoProvision field to true.

Intelligent Memory Improvements (v 15.10):

Teradata Intelligent Memory (optional feature) was introduced from v 14.10 and it helps automatically keep the hottest data in memory (VERYHOT cache) for faster data processing improving Operational Intelligence and tactical query workloads. This requires enough RAM on the node to support very large FSG cache. Changes to VERYHOT cache required tpareset but from v 15.10, DBA can adjust the FSG dedicated to VERYHOT cache without needing tpareset.

1. Introduction to some new features (Teradata v 15.0, v 15.10) (continued)

Partition-Level Locking (v 15.10):

In earlier versions when DML operations were run on multiple rows within partitions of row-partitioned table, the lock is escalated to table level though the rows are being processed from fewer partitions. But now, Teradata locks only required partitions reducing the lock contention and improving concurrency.

Teradata Secure Zones (v 15.10):

This optional feature allows organizations to create “zones” or partitions for separate (and may be unrelated) divisions on a single Teradata database. Users needing to access database objects within a particular “secure zone” need to have required access and privileges. Typical use cases include multi-tenant database services, regulatory compliance, large conglomerates needing access and control on multiple databases owned by unrelated subsidiaries.

New PI on Access Rights Table (v 15.10):

Earlier versions of Teradata used to place table-level write lock on DBC.AccessRights table causing blocks and deadlocks when dropping and deleting objects. From v 15.10, Teradata has changed index (row partitioning) on the DBC.AccessRights table to reduce lock contention by placing partition-level or row-key level locks leading to improved performance on dropping and deleting objects.

2. Some Design and Tuning Tips

Reduce Lock Contention:

Teradata automatically chooses the lock level and type when nothing is specified. Using Access locks wherever possible for reading uncommitted data is one of the ways to reduce lock contention. From Teradata 15.10 we can define tables as Load Isolated allowing concurrent reads of most recent committed data while the table is being loaded.

The Primary Index (PI) Choice:

Write your queries to use Primary Index as much as possible for use in joins. Being able to join on the complete set of Primary Index columns is the least expensive join. The join condition can include additional columns, which are not part of the Primary Index (they will be applied as residual conditions) but missing Primary Index columns eliminates the possibility of a RowHash Match Scan.

If different Primary Index is needed for some SQL statements and changing Primary Index on a table is not an option (for whatever other good reasons), then use a volatile table or a real temporary table with the same structure and data like the original table but with a different Primary Index to be used in the query.

Partition Primary Index (PPI / MLPPI):

If range based selection is predominantly used on a table (example: Where Sales_Year Between 2012 and 2014), it is advised to create partitioned tables (PPI). At times, we may have to further partition (MLPPI) the data to directly access the required partition. Make sure that partition elimination occurs by checking the explain plan.

2. Some Design and Tuning Tips (continued)

Other Indexing Choices:

Teradata has variety of index choices for query performance. Teradata selects Secondary Indexes if the selectivity is more than 85%. Check the explain plan to ensure the index is being used.

Different types of Join Indexes can also be used to avoid large data redistribution. Proper impact analysis needs to be performed and testing needs to be done to validate the usage of indexes as they use storage space and they impact table maintenance. DBQL tables can also be used to validate index usage.

We can create Join Index on single table with data filter (Sparse Join Index) or only required columns in the Join Index. We can change the Primary Index of Join Index which is different from base table's Primary Index to allow targeted queries to hit the Join Index to avoid data redistribution. If we have a table with many columns but very few columns are being used in the query, we can create a Join Index with only required columns which reduces the resource usage significantly. We can use Aggregate Join Indexes as well depending on the granularity of required data.

It's all in the Statistics:

Statistics allow the optimizer to estimate the cost effective execution plan, if no stats available then latest optimizer dynamically collects/sample the stats. No statistics is better than having stale statistics as they may mislead the optimizer to choose incorrect join paths, duplicate big tables, etc. causing inefficient use of system resources. It is always recommended to collect statistics, always collect stats on Index columns, join columns and columns participating in where predicates.

Teradata DIAGNOSTIC HELPSTATS provides recommendation to collect stats on required column/index. Teradata 14.10 features USECOUNT helps get more insights into statistics usage. We can do the audit trail and remove the unwanted stats. From version 14 onwards Teradata recommends to combine multiple statistics collections in one statement to share the spool instead of individual collect stats. It is always good to run SUMMARY stats to provide hints to optimizer for extrapolating the numbers.

2. Some Design and Tuning Tips (continued)

Query Rewriting:

Over the years, Teradata optimizer has grown to automatically select the best execution plan for optimum performance. However, query performance can be improved sometimes by rewriting the query to use proper joins, indexes, etc. This is typically the case for large and complex queries.

Query rewriting requires good understanding of the business logic. Rewriting may involve using subqueries, materializing some of the query logic offline with ETL process, using volatile tables, using NOT NULL or any other conditions to avoid unnecessary data from being processed.

3. XTIVIA Tools and Framework

MVC Tool:

An automated DBA tool for performing Multi-Value Compression to save space and improve performance. Teradata doesn't provide any tool out of the box to define / create list of values to be compressed for tables. Manual effort is involved in collecting candidate column values for compression. Not all data types can be compressed and there is a limit for number of values that can be compressed. Our tool analyzes given tables and columns for possibility of MVC and performs the MVC upon confirmation.

Table Skew Analyzer:

An interactive command line tool to verify the skew and create modified DDL at runtime without changing the Primary Index physically on the table. This tool identifies table skew and allows the DBA to choose new columns at runtime and creates the DDL for deployment. Skew can be analyzed for all tables in a database in one go, for particular table or at a given skew percentage. The tool does not change PI on the analyzed table but it sends out the new DDL (with changed PI) in an email.

Object DDL Comparison Tool:

This is an automated tool to identify the DDL changes between different environments. Often DBA identifies the performance issues related to PI/PPI/Statistics being out of sync between environments. ETL developers also find the quality issues due to objects being out of sync. Our tool is designed to identify differences related to DDL, Primary Index, Partition Primary Index, Secondary Index and Statistics. The tool generates new DDL and a detailed report helping to sync up the environments.

Tuning Environment Creator

When approached for SQL performance tuning, DBAs usually spend lot of time in setting up the objects/statistics/data in order to not impact existing data and ongoing development and testing. Our tool automates this entire process and saves DBA time by automatically creating the required objects related to offending queries and copying data in a separate environment. Using this tool, DBA can focus on actual tuning exercise instead of spending time on copying objects and setting up environment for tuning.

Got questions?

We have the answers!

XTIVIA has been helping customers leverage their Teradata application for years. Our deep technical understanding, as well as our ability to solve business problems, has resulted in successful projects for our customers...every time. Let's talk so we can discuss how to improve your company.

Contact us today:
info@xtivia.com
Call 1-888-685-3101 ext. 2

