Top-level DB design for Big Data in ATLAS Experiment at CERN

Gancho Dimitrov
CERN
CH - 1211 Geneva 23
Switzerland

Keywords:

Database design, Big data, Performance tuning, Oracle, CERN

Introduction

The ATLAS EventIndex system accumulates a set of key quantities for a very large number of particle collision events (aka “events”) recorded by the ATLAS experiment (http://atlas.cern/) at the LHC (Large Hadron Collider) at CERN (https://home.cern/). The main project requirements are handling of tens of billions of rows per year with minimal DB resources, and providing outstanding performance for the fundamental use cases.
When designing the system we asked ourself questions as "how could we build a system to catalog tens of billions of objects which can retrieve information about any object in a fraction of a second? Also, how could we make our catalog dynamic (in content), extensible, and accessible?"

Technical challenges

While our relational schema is simple (just a few tables), challenges include:

- the need to manage significant number of rows to store in our “Events” table (tens of billions of particle collision events) and its associated data volume.
- the need to insert, update or delete potentially large sets of events (up to hundreds million rows) in atomic database transactions
- the need to proficiently handle duplicated “events” in the source data
- the need to retrieve data of any given list of “events” in a fraction of a second
- the need to compute “events” overlap among any dataset with common LHC run number.

Used DB techniques

Currently the Oracle-based EventIndex system hosts about 120 billion rows (“events”) as the data ingestion rate has gone ways beyond the initially foreseen 30 billion rows per year.
Some of the database optimization techniques deployed in this system, which further minimize storage volume beyond relational normalization, transaction volume and database load, and optimize query performance for the use cases, that strongly deserve mention are:

• Up to three GUID (Globally Unique Identifier) columns per Event record in the source data are 36 character strings (e.g. "21EC2020-3AEA-4069-A2DD-08002B30309D"). In our "Events" table, we store these columns using the "RAW" data type, reducing the 36 bytes of storage per GUID to 16 bytes. This considerably decreases Event table per row volume without loss of functionality: when the GUID columns are queried, Oracle easily converts them back to the original CHAR type as needed with the use of virtual columns.

• For the "Events" table we use Oracle's "basic" compression (de-duplication) for table data and compression on its primary key index. Moreover for data loading we use Oracle's direct data load interface. In combination, storage utilization is reduced by a factor of about 3.5 which has the added advantage of reducing similarly the I/O footprint for writing data, undo and redo to the storage subsystem.

• The "Events" table is List-type partitioned by dataset ID. The main advantage is that sets of events can be deleted by simply dropping the associated partition. This operation is needed more often than we initially expected because datasets are sometimes re-imported.

After optimization, storage volume is about 20 bytes/row, which is a factor of 10 reduction from the initial 210 bytes/row. This reduction, however, is only for the table segments. Overall, including indexes, storing 25 billion rows requires about 1TB of space (rather than 5TB). The savings of 4TB of disk space, in itself, is not the foremost point but has a knock-on effect which is particularly beneficial for query performance: It enables the caching of a larger fraction of the database rows into the database data cache (buffer pool) which yields real performance gains in query response.

Conclusion

In summary, using a relational model and a number of carefully chosen techniques available in Oracle RDBMS results in an impressive minimization of resources while exceeding performance goals.

Contact address:

Gancho Dimitrov
CERN
CH - 1211 Geneva 23
Switzerland

Phone: +41 22 7673310
Fax: +41 22 7678350
Email gancho.dimitrov@cern.ch
Internet: https://home.cern/