ATLAS database application enhancements using Oracle 11g

G Dimitrov\(^1\), L Canali\(^1\), M Błaszczyk\(^1\), R Sorokoletov\(^2\), on behalf of the ATLAS Collaboration

\(^1\)European Organization for Nuclear Research (CERN), 1211 Geneva, Switzerland

\(^2\)University of Texas at Arlington (UTA), 701 South Nedderman Drive, Arlington, USA

E-mail: gancho.dimitrov@cern.ch

Abstract. The ATLAS experiment at LHC relies on databases for detector online data-taking, storage and retrieval of configurations, calibrations and alignments, post data-taking analysis, file management over the grid, job submission and management, condition data replication to remote sites. Oracle Relational Database Management System (RDBMS) has been addressing the ATLAS database requirements to a great extent for many years. Ten database clusters are currently deployed for the needs of the different applications, divided in production, integration and standby databases. The data volume, complexity and demands from the users are increasing steadily with time. Nowadays more than 20 TB of data are stored in the ATLAS production Oracle databases at CERN (not including the index overhead), but the most impressive number is the hosted 260 database schemes (for the most common case each schema is related to a dedicated client application with its own requirements). At the beginning of 2012 all ATLAS databases at CERN have been upgraded to the newest Oracle version at the time: Oracle 11g Release 2. Oracle 11g come with several key improvements compared to previous database engine versions. In this work we present our evaluation of the most relevant new features of Oracle 11g of interest for ATLAS applications and use cases. Notably we report on the performance and scalability enhancements obtained in production since the Oracle 11g deployment during Q1 2012 and we outline plans for future work in this area.

1. Introduction

The ATLAS experiment [1] is the biggest multi-purpose particle detectors in the history of particle physics. It is one of the four big experiments built at the Large Hadron Collider (LHC) at CERN and is used to record and study proton-proton collisions at a final center-of-mass energy of up to 14 TeV. The ATLAS detector is located in Geneva, Switzerland at the European Organization for Nuclear Research (CERN). The data that are recorded by the detector are transferred to globally distributed computing centers for analysis. Additionally, the computing centers produce simulated particle collisions for comparison with detector data. The aggregate data accumulate to 20 Petabytes per year not including temporary and transient data. This rate is likely to grow in correspondence with the energy and luminosity increases of the LHC in the coming years.

Apart from the above impressive volumes of event data, there are also impressive numbers for the detector condition and management data stored in the relational databases not only in terms of volume but also in number of versatile applications each with its own requirements. Relational databases are the most popular form of database scheme and have been commonly used since 1970. In relational
databases, data are organized in the form of related tables: each table can have many rows and these rows can have many columns. The software interface used to build and access the data structures is the Structured Query Language (SQL) [2]. It is the most widely known query language used today.

The ATLAS experiment relies on Oracle RDBMS (Relational Database Management System) [3] for detector online data-taking, storage and retrieval of configurations, calibrations and alignments, post data-taking analysis, file management over the grid, job submission and management, data replications to other computing centers and many others. Currently the ATLAS data stored in Oracle is larger than 20 TB, not including the indices overhead, in more than 260 database schemas. Supporting so many schemes (DB applications) as shown in the Table 1 is a real challenge.

| Database / Metric | ATONR | ATLR | ADCR | ATLARC |
|-------------------|-------|------|------|--------|
| Main database role | Online data taking | Post data taking analysis | Grid jobs and file management | Event-level metadata (TAGs) |
| # Cluster nodes | 3 | 4 | 4 | 2 |
| # DB schemas | 70 | 154 | 12 | 50 (incl. 23 arch) |
| # Tables | | | | |
| - Non partitioned | 37319 | 64755 | 498 | 13992 |
| - Partitioned | 411 | 826 | 25 | 8091 |
| Volume | | | | |
| - Table segments | 2.4 TB | 5.5 TB | 8.3 TB | 4.5 TB |
| - Index segments | 2.5 TB | 7.5 TB | 3.4 TB (incl. IOTs) | 12.5 TB |
| - LOB segments | 0.5 TB | 0.7 TB | 0.3 TB | - |
| Average daily segments growth (Jan-Apr. 2012) | 5.0 GB/day | 9.2 GB/day | 17 GB/day | 24.2 GB/day |

Table 1. Facts about the ATLAS Oracle databases at CERN (as of April 20th 2012)

2. ATLAS database deployment and the main DB applications

ATLAS has four production database clusters at CERN, supported by IT-DB group, each having its own specific role. In addition, some of them are source, others are destination of data replication via Oracle Streams technology. This data replication is of great importance for many applications.

2.1. The ATLAS online database ATONR is strongly related to the data-taking operations. This is a critical part of the data taking operation and as such cannot be accessed by any offline activities that would eventually compromise its availability and performance. ATONR holds three major classes of data:

- Detector, trigger and data acquisition configuration
- Detector calibrations and alignments for high-level trigger algorithms
- Detector, trigger and data acquisition. The commercial product PVSS (Process Visualization and Steering System) [4] is used by all experiments at LHC to store and monitor the detector parameters, such as temperatures, voltages, currents, gas flows etc.
2.2. The ATLAS database for offline analysis ATLR is available for Tier-0 express reconstruction and reprocessing, conditions metadata, trigger configurations for reprocessing, logging of critical operations and many others. ATLR is a destination of the data replication from the online database ATONR and also is a source for further replication of the condition and trigger data to all ATLAS Tier-1s using the Streams technology.

2.3. ADCR database (ATLAS Distributed Computing RAC (Oracle Real Application Cluster)) hosts data and application logic for large and complex systems like:
   - Distributed Data Management over the grid (DDM), previously called Don Quijote 2 (DQ2), now called Rucio [5]
   - Production and Distributed Analysis (PanDA) [6]
   - LCG File catalog (LFC) [7]
   - ATLAS Grid Information System (AGIS) [8] which stores information about services, resources, configuration parameters and topology of the whole ATLAS computing grid

2.4. ATLARC database stores data from the Event-level Metadata (TAGs) system [9]. It supports efficient identification and selection of events of interest for a given analysis. This database is also used for long term data archiving from the other systems.

3. Database operational and application challenges

   The scalability, reliability and increase in the performance have been always in the top of the list with requirements for the data management and processing. Databases are a core component of the ATLAS computing infrastructure and are expected to be highly available and reliable. Over the first two years of LHC data taking this has been largely the case. Many lessons have been learned on the way, however as one could expect, the interaction of Oracle software with the applications has shown to be one of the critical points in the application implementation and operation of database services.
Here we mention some of the challenges that we are most focused on:

- **Database clusters built on common base of servers and storages for reducing cost and maintenance efforts serve different type of application workload.**

  The Oracle services for WLCG [10] follow the idea of providing high availability and scalability while using the same commodity/low-cost components. The architecture and deployment model has been uniform across online, offline and Tier-1 installations (however specific implementation details differ across sites).

- **The nature of the grid generates dynamic workload. Coping with high loads on the database side is essential.**

  Over the years there has been a growth in resource usage on some the ATLAS database clusters that naturally comes from the new user demands.

- **Stabilize the performance of SQL statements. Avoid pitfalls generated by wrong assumptions on data size and/or access patterns, lack of relevant testing, inaccurate or stale database objects statistics, errors in DB layout.**

  Change management and application testing life cycle play an important role for ensuring smooth daily operations. In particular all new applications are developed on a separated development DB service. Before going to production they are tested on integration services potentially with the help of a database administrator for tuning.

- **Design applications for scalability and design for long term data archiving**

  Data archiving requires creating a strategy for long-term data retention. This activity is not only about saving disk space. Instead, it is also about achieving maximum performance of applications through a reduced amount of data to process. Not only will applications run faster, but also backups, recoveries and reorganizations will execute more efficiently.

4. **Use of the new Oracle 11g features for performance, scalability and ease of administration**

In order to address the above-mentioned challenges and to make the ATLAS database applications more reliable, we explored and evaluated some of new Oracle 11g features. The ones described below we deployed in production in March 2012 (all ATLAS databases were upgraded to Oracle version 11.2.0.3 in Jan-Feb 2012)

- **Active Data Guard (ADG)**

  This is one of the most game-changing features available in Oracle 11g as it provides improvements in the areas of disaster recovery, scalability and replication. An Active Data Guard database is a standby copy of a production database that can be queried in read-only mode. This opens the possibility of scaling out the database load by offloading the main production cluster and at the same time of having a disaster recovery solution.

  In addition it seems very much suitable for online to offline replication because of lower latency and more robustness than Streams as replication is done at Oracle block-level. The ADG technology is not new to database services for LHC experiments as it is based on an enhancement of the physical standby (aka Data Guard) that has been used at CERN for several years already.

  Currently in ATLAS the ADCR ADG database (deployment setup as of example 1 in figure 2) is used for data dumps, exports and analytical queries on large data volumes. ATONR ADG is considered for access to the PVSS alert data from clients on the general-purpose network, as the alert specific tables cannot be replicated via the Streams.
Figure 2. Active Data Guard deployment models

- **Result caching on the server side**
  This is a performance-enhancing feature. It is useful in cases where data do not change often, but is queried on a frequent basis. Oracle sends back to the client a cached result if the result has not been changed meanwhile by any transaction, thus improving the performance and scalability (we have zero table or index block reads). Result caching can be enabled on SQL queries, sub-queries and also on PL/SQL functions. In ATLAS this new feature was activated for specific statements in the PanDA and the DDM systems. The statistics shows that 95% of the executions of the PanDA server queries (17 distinct queries with this option on) were resolved from the result set cache and 99% of the calls to the DDM stored PL/SQL functions as well (having a rate of 3 billion calls per month).

- **Interval partitioning**
  This is a manageability feature for large databases. Range based partitioning has always been requiring maintenance efforts from the developer or from a database administrator for pre-creation of partitions relevant to future time. Now with the interval partitioning new partition is created on the fly when a user transaction imposes a need for it instead of raising an error as in the Oracle 10g version. We take advantage from that and changed a particular PanDA table in transparently for the application. The second benefit is that the resource expensive row-by-row deletion for sustaining 90 days ‘sliding data window’ (application specific decision) is removed and replaced by partition removal approach.

- **Better disk space utilization**
  This is a feature to ease administration. Disk space can be freed by implementing a ‘sliding window’ on partitioned indices based on DATE or NUMBER column type by setting unusable oldest index partitions. This approach is applicable for indices which are needed only for access to the most recent data. For example, recently 130 GB of space were freed by applying this method on a DDM table that keeps tracking grid information.

- **Justification for existence of an index**
  This is a feature to enhance performance and ease database administration. Indices are often beneficial for performance of queries in transactional database. Although in some cases they can cause additional IO and/or storage usage that are not justified. This feature helps database administrators and developers to address that. Studying and judging an index importance for the application can be done by setting the index ‘visible’ or
‘invisible’ to the Oracle Optimizer. In ATLAS this feature is of great help for the TAGs data access pattern studies where the data volume is in the order of few TB per year and the number of indices is not negligible (space used from the index segments is three times larger than the table ones).

- **Customization in statistics gathering**

  This is a performance feature. Having up-to-date statistics for the user tables is essential for having optimal queries’ data access path. Today we take advantage of a new approach in statistics collection on partitioned tables called incremental statistics gathering. Oracle spends time and resources on collecting statistics only on partitions which are transactional active and computes the global table statistics using the previously ones in an incremental way. This is of great importance for ATLAS as we have plenty of large volume tables (Figure 3)

| TABLE_OWNER | TABLE_NAME | PARTITION_NAME | NUM_ROWS | LAST_ANALYZED |
|-------------|------------|----------------|----------|----------------|
| ATLAS_PANDA | JOBSARCHIVED4 PART_JOBSARCHIVED4_28012012 | | 670646 | 28-Mar-12 11:30:43 |
| ATLAS_PANDA | JOBSARCHIVED4 PART_JOBSARCHIVED4_29012012 | | 655132 | 29-Mar-12 11:30:36 |
| ATLAS_PANDA | JOBSARCHIVED4 PART_JOBSARCHIVED4_30032012 | | 735349 | 30-Mar-12 11:31:08 |
| ATLAS_PANDA | JOBSARCHIVED4 PART_JOBSARCHIVED4_31032012 | | 257352 | 30-Mar-12 11:30:38 |

**Figure 3.** Incremental statistics gathering on a partitioned table

Apart from setting the incremental statistics gathering on tables of interest, we also set stats stale threshold and have specific setup for letting or avoiding histogram collection instead of relying on the Oracle default settings because we know much better the nature of our data and the behavior of our applications.

5. **Conclusions**

We took advantage and activated in production some of the new Oracle 11g features shortly after this major version upgrade in order to be in a better state during the current important data-taking and analysis year. The availability, scalability and performance are improved and we keep on studying and exploring new solutions for addressing the ever-increasing user demands. In this area of work an important factor for success is the effective communication between database administrators and developers. This has been the case in ATLAS many years and has given good results in the development, maintenance and operation phases.

**References**

[1] The ATLAS Collaboration et al 2008 JINST 3 S08003 doi:10.1088/1748-0221/3/08/S08003
[2] Structured Query Language, http://en.wikipedia.org/wiki/SQL
[3] Oracle RDBMS, http://www.oracle.com/us/products/database/index.html
[4] PVSS service at CERN, http://j2eeps.cern.ch/wikis/display/EN/PVSS+Service
[5] Garonne V for the ATLAS Collaboration 2012 The ATLAS Distributed Data Management project: Past and Future These proceedings (CHEP2012)
[6] T Maeno et al 2011 J. Phys.: Conf. Ser. 331 072024 doi:10.1088/1742-6596/331/7/072024
[7] LFC, https://twiki.cern.ch/twiki/bin/view/LCG/LfCwlcg
[8] Alexey Anisenkov et al 2012 AGIS: The ATLAS Grid Information System These proceedings (CHEP2012)
[9] Elisabeth Vinek et al 2011 J. Phys.: Conf. Ser. 331 042027
[10] WLCG, http://lcg.web.cern.ch/lcg/