ATLAS database application enhancements using Oracle 11g

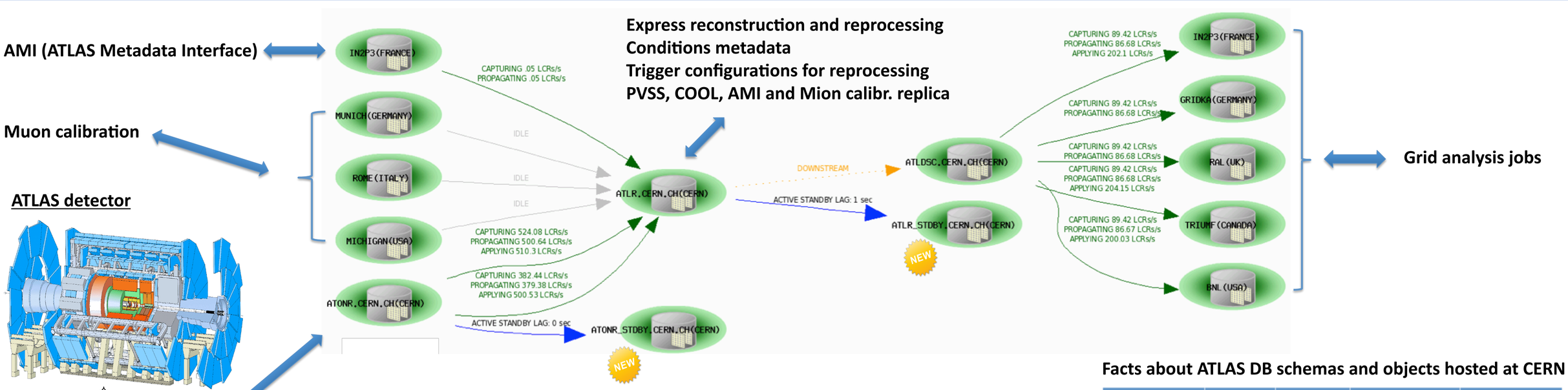
Gancho DIMITROV (CERN/ATLAS), Luca CANALI (CERN/IT),
Roman SOROKOLETOV (UTA/ATLAS), Marcin BLASZCZYK (CERN/IT)
On behalf of the ATLAS Collaboration

CHEP 2012
New York

I. Abstract

The ATLAS experiment at LHC relies on databases for detector online data-taking, storage and retrieval of configurations, post data-taking analysis, files management over the grid, job submission and management, data replication to other computing centers. The Oracle Relational Database Management System (RDBMS) has been addressing the ATLAS database requirements to a great extent for many years. Several database clusters were deployed for the needs of different applications. The ATLAS databases at CERN were upgraded to the newest Oracle version 11g Release 2 during the LHC technical stop in winter 2011-2012. Oracle 11gR2 comes with several new features and enhancement of existing features intended to improve performance, reduce maintenance effort and increase service availability. In this work we present our evaluation of the most relevant new features of 11g of interest for ATLAS applications and use cases.

II. ATLAS databases and main DB applications



Facts about ATLAS DB schemas and objects hosted at CERN

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 - Partitioned	37319 411	64755 826	498 25	13992 8091
Volume - Table segments - Index segments - LOB segments	2,4 TB 2,5 TB 0,5 TB	5.5 TB 7,5 TB 0,7 TB	8,3 TB 3,4 TB (incl. IOTs) 0,3 TB	4,5 TB 12,5 TB -
Average daily segments growth (Jan-Apr. 2012)	5.0 GB/day	9.2 GB/day	17 GB/day	24.2 GB/day

III. Database operational and application challenges

- Database clusters built on common base of servers and storages (for reducing const and maintenance efforts) serve different type of application workload.
- The nature of the grid generates dynamic workload. Cope with high loads is essential thus decoupling transactional and non-transactional workload is important.
- Inappropriate execution plans of the SQL statements by either wrong assumptions, lack of good testing, not accurate or not upto date statistics, inappropriate DB layout
- Design applications for scalability, long term data archiving and prepare them with good level of logging information for addressing eventual issues and incidents.

IV. Use of Oracle 11g features for performance, scalability and ease in adminitration

Result caching on the server side

It is useful in cases where data does not change often but is queried on a frequent basis. Now it suits well in:

PanDA server queries

For the first month of use > 12 million query executions were resolved from the cache (=95% of all executions, 17 distinct queries)

DDM catalog functions

Activated on certain PLSQL functions which are called with high rate (~ 3 billion calls per month) . The statistics shows that 99% of the executuons were resolved from this result set cache.

Automatic interval partitioning

PanDA database schema

A PanDA standard heap organized table changed to interval partitioning in transparent to the application way. The benefit is that the resource expensive row by row deletion for sustaining 90 days 'sliding data window' is removed and replaced by partition removal approach. New partition is created on a fly when a user transaction imposed a need for it instead of raising an error as in the 10g version .

TABLE_NAME	PARTITION_NAME	HIGH_VALUE
DATASETS	DATASETS_BEFORE_01122011	TO_DATE(' 2011-12-01 00:00:00',
DATASETS	SYS_P14395	TO_DATE(' 2012-01-01 00:00:00',
DATASETS	SYS_P14396	TO_DATE(' 2012-02-01 00:00:00',
DATASETS	SYS_P14397	TO_DATE(' 2012-03-01 00:00:00',
DATASETS	SYS_P14398	TO_DATE(' 2012-04-01 00:00:00',

Better disk space utilization

'Sliding window' on partitioned indices based on DATE or NUMBER column type can be achieved by setting unusable oldest index partitions (applicable for indices which are needed only for access to the most recent data). In that way space is freed.

Studying on the index importance

Studying and judging an index importance for the application can be done by setting the index **visible or invisible** to the Oracle Optimizer. This is of great help for the TAGs data access pattern studies where the data volume is in the order of terabytes per year and the number of indices is not negligible.

Customization in statistics gathering

Having up-to-date stats on the tables is essential for having optimal queries data access path

⇒ **Incremental stats 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 incremental way.

⇒ **Forbid histograms computation** on columns this can be appropriate

⇒ **Set user defined values for the stats stale percent threshold.**

TABLE_OWNER	TABLE_NAME	PARTITION_NAME	NUM_ROWS	LAST_ANALYZED
ATLAS_PANDA	JOBSARCHIVED4	PART_JOBSARCHIVED4_28032012	670646	28-MAR-12 11:30:43
ATLAS_PANDA	JOBSARCHIVED4	PART_JOBSARCHIVED4_29032012	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

Active Data Guard (ADG)

It is a standby copy of a production database that can be queried in read-only mode

Good for:

=> Scaling out database load

Offload main production cluster. Run reports against prod data on ADG

=> Replication

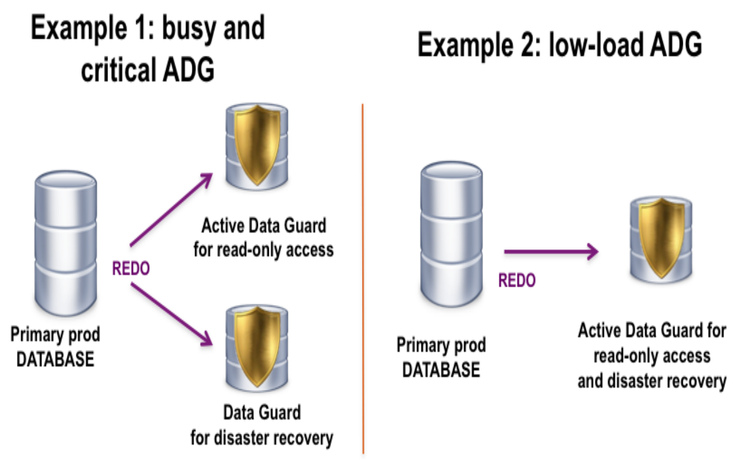
To be considered for online-offline replication

Lower latency and more robust than Streams as

replication is done at Oracle block-level and thus is less complex

=> Disaster recovery solution

Much faster than recovery from backup. Already available in 10g.



ATLAS use cases:

1) Currently ADCR ADG database (deployment setup as of example 1 is used for data dumps, exports, analytical queries on large data volumes.

2) ATONR ADG is considered for access to the PVSS ALERT data from clients on GPN as the ALERT tables cannot be replicated via the Streams.

Conclusions

⇒ Some of the new 11g features showed to be well suited for certain ATLAS use cases. Series of improvements and tunings were done using them in order to have smoother and easier operations plus better overall performance.

⇒ There are ongoing studies on using of Active Data Guards for offloading the primary databases.

⇒ Keep on a close work with the DB developers community on setting best practices and getting maximum from the current DB technologies.