Future PostgreSQL work tasks for the CTA project

This document lists and describes the future CTA work tasks that are either directly or indirectly related to running CTA on top of a PostgreSQL backend as opposed to an Oracle one. Before describing each individual task, this document first describes the choices for using either PostgreSQL or Oracle.

The preferred choice of database backend technology to use in a CTA deployment is very much different for the Tier-0 at CERN and the Tier-1s in the LHC Grid.

- Tier-0: Initial deployment with Oracle and migrate to PostgreSQL later.
- Tier-1: Deploy with PostgreSQL.

The choice of Tier-0 to deploy with Oracle first has been made to reduce the risks of unknown operational problems with PostgreSQL. The CERN Tier-0 knows how to fully operate, tune, backup and recover an Oracle backend.

It is essential that operational experience is gained with PostgreSQL before the very first deployment of CTA at a Tier-1 site. This experience will be gained through two separate activities:

- The stress tests of the CTA Continuous Integration (CI) will run regular multi-million file tests against a PostgreSQL database provided by the IT database group.
- The CTA pre-production instance currently being assembled for the ALICE experiment will use a PostgreSQL database provided by the IT database group, as opposed to the ATLAS preproduction instance which will use Oracle.

Task 1. Finalise and optimise CTA database schema for performance and functionalities

This task depends on no other tasks being completed first. The task should take 4 person-weeks. The impact of the task is future proofing the use of PostgreSQL and completing the database side of critical CTA functionalities.

Database schema modifications need to be made in order to support critical functionalities such as:

- Logical deletes that allow CTA to recover potential petabytes of data accidentally deleted by users but still physically stored on append-only tapes.
- Track the operational states of tapes and files such as which tapes were imported from CASTOR in order to enforce the policy of not writing to them so they could easily be returned to CASTOR in case of an unexpected migration issue.

Modifications need to be made to both the database schema and access code in order to ensure performance on both PostgreSQL and Oracle databases, for some examples:

- Reduce the memory footprint of database rows. There are currently half a billion files in the CASTOR namespace and it therefore not too big a leap to imagine that 1 billion files will be quickly achieved in LHC Run 3. A single additional byte stored for each file would equate to 1 Gigabyte of additional database space. The size of a database row for a CTA file is therefore very important when trying to get as many rows as possible cached in the RAM of the CTA database server(s).
- Reduce the number of network round trips to the database. Contrary to CASTOR, the CTA project separates business logic from database functionality and therefore purposely avoids usage of PL/SQL and its variants on the database server. This is required in order to keep compatibility with “any” database backend technology. The business logic of CTA is implemented within the application daemons of the project. Performance is gained with such an architecture by favouring the querying and modifying of batches of database rows rather than one network round trip for each individual CTA file.

Task 2. Split the CTA catalogue by LHC experiment

This task is dependent on task 1 being completed first. This task should take person-4 weeks. The impact of this task is to reduce the operational impact of future upgrades and repairs to CTA deployments.

The hierarchical namespace information of each file in CTA is naturally split across one EOS instance per LHC experiment. This is currently not the case for the flat tape file listing information stored within the CTA catalogue. Like its CASTOR predecessor, there is currently one CTA catalogue database for the whole of CERN. The CTA catalogue should be split by experiment instances in order to ease interventions without requiring to agree with all experiments and users at CERN to a global downtime.

Task 3. Develop the injection tools / code modules required to insert CASTOR metadata into the CTA catalogue

This task is dependent of tasks 1 and 2 being completed first. This task should take person-4 weeks. The impact of this task is enabling the LHC experiments to migrate out of CASTOR and into CTA.

The context of this task is within the implementation of the tools necessary to migrate the LHC experiments from CASTOR to CTA. At a high level of abstraction there will be two flows of metadata from CASTOR to CTA. Unlike the single namespace of CASTOR which stores both the hierarchical namespace metadata and the flat tape file listings, CTA distributes this information between the EOS namespace and the CTA catalogue. The EOS namespace is responsible for storing hierarchical namespace metadata such as directories, permissions and attributes. The CTA catalogue is responsible for the flat tape file listings. This split enables tasks such as repack to run on the CTA catalogue without having to access and degrade the performance of the EOS namespace. Tools and/or code modules need to be implemented that can efficiently insert the tape file listings of CASTOR's half a billion files into both PostgreSQL and Oracle versions of the CTA catalogue. These tools and/or code modules will also handle inserting non-file metadata such as tapes, tape libraries, archive routes and mount policies.

Task 4. Develop the reconciliation tools that ensure the CTA EOS namespace and CTA catalogue are consistent with each other

This task is dependent of tasks 1 and 2 being completed first. This task should take 4 person-weeks. The impact of this task is the long-term prevention of metadata inconsistencies within CTA.

CTA splits its file metadata between the hierarchical namespace of EOS and the flat tape file listings of the CTA catalogue. These two metadata structures must be periodically proven to be consistent with each other and corrected where necessary. Tools therefore need to be developed to detect inconsistencies and facilitate reconciliation. These tools must be performant with both PostgreSQL and Oracle databases.

Task 5. Develop the tools required to migrate a deployed Oracle CTA instance to a PostgreSQL one.

This task is dependent on tasks 1 and 2 being completed first. This task should take 4 person-weeks. The impact of this task is the complete migration of an Oracle-based CTA instance to PostgreSQL.

The beginning of this document explains why CERN Tier-0 currently plans to deploy CTA on top of an Oracle backend followed by a later migration to a PostgreSQL solution. Tools needs to be developed to efficiently perform such a migration and to help facilitate the operational procedures that will need to be carried out.