Database meeting 2022-07-20

Date

20 Jul 2022

Attendees

- Igor Gaponenko Fabrice Jammes Colin Slater Andy Hanushevsky Salman Habib Fritz Mueller Joanne Bogart

Notes from the previous meeting

- Database meeting 2022-07-13

Discussion items

Discussed	Item	Who	Notes
✓	Project news	Fritz Mueller	No news

✓	Status of DP02	Fritz Mueller Colin Slater Igor Gaponenko Fabrice Jammes	Fabrice Jammes: - DP02 ingested at IN2P3 - minor problems with the implementation of the contributions queue in `qserv-ingest` when ingesting the catalog in IDF were addressed Fritz Mueller: there is a number of fix-ups - `ObsCore` tables are missing in IN2P3 - `ForcedSourceOnDIaObject` is incorrect, and there is an ongoing effort on generating the correct input data for the table - truth tables need to be re-ingested into IN2P2 - we should be heading towards a single-step ingest of the catalog using `qserv-ngest` rather than the experimental step-by-step approach taken so far in `qserv-prod` and `qserv-int` . This will be available after the required data corrections will be finalized. Fritz Mueller worked with the RSP team on the performance and optimization of Qserv - generating a lot of load (hundreds of parallel queries) on the `qserv-int` - using `Object` and `Source` tables only - considering switching the tests to `qsderv-dev` where the next/final version of the catalog s going to be ingested and retained for the purpose of the performance testing - was wondering about the possibility to use (as a whole or in parts) `Kraken` - Andy Salnikov parts of `Kraken` (including the query generator) could be reused. Though, the processing engine was designed to be NCSA-specific (relying on `SLURM`). The next problem with the idea (of reuse) is that `Kraken` talks to Qserv via the MySQL protocol (`mysql-proxy`), meanwhile the `Monkey` interacts with Qserv indirectly via the TAP service. Igor Gaponenko : - Fabrice Jammes we need to improve `qserv-ingest` to allow incremental ingest of tables (one table after another in separate runs of the workflow) as the *fix-ups* to the catalog by replacing (deleting and re-ingesting) existing tables - `qserv-dev` has 10 workers. Is this okay for the load testing? - Fritz Mueller it's good since it would allow testing the scalability of Qserv Treating the table `MatchesTruth` as the RefMatch table: - Fritz Mueller any progress on the truth tables? - ⚠ DM-35578 - Jira project doesn't exist or you don't have permission to view it. Status of `ForcedSourceOnDiaObject`: - Colin Slater: the test version of one tract (split into 49 patch `Parquet` files) is available for testing. The correct `YAML` schema is already in the `main` branch. Links to the Butler collection were posted in the group Slack channel. - Igor Gaponenko will work ASAP on ingesting this slice into the separate table `ForcedSourceOnDiaObject_v1` and report results to unblock processing of the rest of the collection Andy Salnikov on the status of extracting data for the table `ObsCore` (in the long run, not specifically for `DP02`): - have investigated various options considering the performance and complexity of the solutions - working on the tech note to summarize those - the "quick-and-dirty" solution is available, and it's relatively fast as it only takes 1.5 hours to extract the required data for Qserv. Though, it's not correct. - the correct approach for extracting the data had problems with the performance - it runs "forever" (is taking the amount of time that's beyond any practical limits) compared with the quick and dirty ○ because of that, zero chance to implement the extractor as a view in `PopstgreSQL` ○ a potential option is to have the materialized view (with the drawback of updating the view each time an update to the underlying tables is made. Other issues with this idea may be seen as well. ○ another option - generate the table at a time when the ingest is happening, which will require extra steps for integrating this process with Butler. This has its own implications that would need to be further studied. Fritz Mueller: the last missing capability in `DP02` will be to add ~12 magnitude columns to the table `Object`: - first by using the R-I system's REST API to `ALTER TABLE ...` that's already available - In the next step, set the right values on the new columns by running `UPDATE ...` using a mechanism that may need to be devised (via the R-I system's REST API or manually) - Igor Gaponenko will work on this after ingesting the new version of the table `ForcedfSourceOnDiaObject` and after Fritz Mueller will be done with the ongoing investigation of the `RefMatch` option for the table `TruthSummary`.

✓	Versioning of the Replication/Ingest system's REST API	team	Context: • ⚠ DM-35456 - Jira project doesn't exist or you don't have permission to view it. Discussed possible options to address the problem: • Fritz Mueller : it's okay to introduce the version into JSON files • Fritz Mueller proposed the capability-based versioning by grouping services into functional groups and • introduce different version numbers for each capability to be reported by the service `/meta/version` of the REST API • let the workflow check the versions against the expected ones • per-end-point version check (as an extreme case of the one mentioned above) • Fabrice Jammes want to fail early rather than along the road after ingesting • Igor Gaponenko will make another iteration in investigating these options and prepare a plan for Fritz Mueller • Igor Gaponenko mentioned obstacles in migrating and testing `qserv-ingest` against new versions of the API.
✓	Merging three Git packages `qserv` , `qserv-operator` and `qserv-ingest` into one	Fritz Mueller Fabrice Jammes	Discussed obstacles in testing new features in the triplet of 3 packages `qserv`, `qserv-operator`, and `qserv-ingest`: • it's a three-way problem that needs to be (presently) addressed sequential, one package at a time • there is a recipe on how to tweak the CI for testing each package. However, it's complicated. • the right solution would be to merge both packages into `qserv` Fritz Mueller will start working on prototyping the idea next week by merging/cloning `qserv-operator` into `qserv`. The updated CI will need to be implemented for that. After that, a similar effort will be made for `qserv-ingest`.

Action items

□