Database Meeting 2022-10-12

Date

12 Oct 2022

Attendees

- Igor Gaponenko Fritz Mueller Andy Salnikov Colin Slater Kian-Tat Lim Fabrice Jammes Andy Hanushevsky Joanne Bogart

Notes from the previous meeting

- Database Meeting 2022-09-21

Discussion items

Discussed	Item	Who	Notes
✅	Project news	Fritz Mueller Colin Slater	Fritz Mueller on the project news: - DMLT F2F next week. It's open to everyone - Support for user-generated data products is to be discussed among the topics. This is going to be one of the hot topics for the upcoming year. Office spaces in Building 50. We've been contacted by the SLAC IT/Admin folks. Shared group space in ROB is an option for up to 4 people in one room. Staying on B50 would require being present on site for at least 3+ days a week. Anyone interested in a situation with the prospective shared office space should contact Christine Soldahl directly.
✅	User-generated data products in Qserv	Fritz Mueller	Context: - It's been on the DM's task list for years - Fritz Mueller has started discussing the topic with GPDF, Colin Slater, and Igor Gaponenko to figure out what needs to be done at Qserv and what would be done at the TAP layer. - the following discussion is meant to provide Fritz Mueller with feedback to help him prepare a presentation for the planned DMLT F2F meeting next week. Fritz Mueller will share the presentation with the team before the meeting. Fritz Mueller: - started working on figuring out the requirements and expectations - TAP seems to have some support for the user data - use case: a user has a table and he/she wants to get it ingested into Qserv and be referenced - long-term data management issues with maintaining these data within Qserv - an open question on the data rights and implementing the mechanics witing Qserv or TAP - an open question on managing the TAP schemas for these products. The schema has a global namespace design. No concept for per-user partitioning of the schema. - a question on maintaining the metadata on the products - etc. Colin Slater: A big question is on a possible user interface for this. It has to be interactive because users are not aware of the specific requirements of Qserv. So, the interface would need to ask them additional questions before proceeding with the ingest. Discussed an option of setting up the browser-based ingest "wizard" for requesting and managing the data ingest interactively: - a user uses the interface to formulate a request by providing as much information as he/she knows (wishes us to know) about the data by providing a location of the data in an object store (a URL or a collection of URLs), or uploading a file into the Web server, answering a few mandatory questions on the name of database/table - the service would analyze the data and respond with more questions to clarify the details of the request - the processing begins - the ongoing status and problems would be seen on the Web page - Fritz Muellerin most cases the ingests should be done within seconds or minutes Colin Slater on a minimal set of the formats supported here: - VO tables - CSV - Parquet - Binary tables in the FITS files Igor Gaponenko mentioned a use case of ingesting results of the `SELECT` queries sent to Qserv, like in: ``` SELECT ... FROM ... INTO <user-db>.<user-table> ```

✅	Technical discussion on scaling ingests in the Replication/Ingest system	Fabrice Jammes Igor Gaponenko	Context: • In the current version of the Ingest system, the concurrency of the data loading activities is limited by the corresponding worker configuration parameters specified at the start-up time of the worker services. In effect. these parameters specify the size of the thread pool at the worker ingest service. These threads are allocated for pulling data from the object stores (or the local file systems) and ingesting those into MySQL. • Our colleagues from IN2P3 may experience too much load on the internal Object store with the default values o the parameters. • Should we also allow the per-catalog limits on top of that? Igor Gaponenko proposed the following solution: • consider the limits configured at the worker server as the "hard" limit that can't be exceeded. The limit is presently based on the number of hardware threads available at the worker nodes, the type (HDD vs SSD) of the disk storage, etc. This limit doesn't take into account the I/O capacity of the remote object stores. Qserv usage by users and the status of Qserv are also not taken into consideration by this limit. • let the workflow set the soft limit when ingesting a specific catalog based on knowing characteristics (I/O capacity and supported parallelism level) of the remote worker store or the current use of Qserv. • the JIRA ticket will be made shortly to add support to the Replication/Ingest system for that. The following JIRA ticket has been registered to address this case: • ⚠️ DM-36627 - Jira project doesn't exist or you don't have permission to view it.
✅	Extending qserv-ingest to support the Parquet-to-CSV and partitioning phases	team	Fabrice Jammes will discuss it with the FrDF colleagues first. After that, someone would begin working on this project. Fritz Mueller thinks Igor Gaponenko should begin using `qserv-ingest` it to see how it may need to be improved.
✅	On building the table statistics (row counters) for optimizing certain classes of Qserv queries	Fabrice Jammes Igor Gaponenko	Context: • Qserv at FrDF doesn't seem to have these stats deployed in Qserv. As a result of this the simple (unconstrained) "`COUNT *`" queries are much slower in there compared with what's observed at IDF. • A reason for this is that the current version of `qserv-ingest` doesn't implement an extra step to be made after finishing ingesting a catalog into Qserv. Igor Gaponenko: On collecting the table statistics. It's been documented in the new version of the documentation on the Ingest system: Managing statistics for the row counters optimizations. This step can be done as a part of the catalog ingest campaign or *a posteriori*.
✅	Moving documentation on the Ingest system into Git	team	Fabrice Jammesmentioned issues with finding the Confluence-based documentation on the Ingest system. Suggested moving it to GitHub. In the meantime, it would be nice to have a link to Confluence from the Git-based documentation on Qserv. Fritz Mueller eventually, we would have to move this document to Git. We have an example of how the documentation on Qserv would look like using the customized styles at: https://documenteer.lsst.io/guides/index.html

Action items

- [ ] Igor Gaponenko owes collections of URLs for the DP02 Parquet files to our Google collaborators.