Qserv: Questions from in2p3 to SLAC and related answers

Emmiters   Emmanuel Medernach, computer scientist, developer and EGI TIER2 manager
                            Fabrice Jammes, computer scientist, developer and head of LPC/in2p3 IT department

Approval   Emmanuel Gangler, scientist, head of LSST team at LPC/in2p3

Diffusion

For action   SLAC (Douglas Smith, Daniel Wang, Jacek Becla)

For information   LSST-France

Change history

Date	Author	Description of the update
6Feb2013	LPC	Creation
13Feb2013	LPC	Answer proposal to questions 1 to 8 (2,3,5,6 7 closed), thanks to SLAC help.
28Feb2013	LPC	Questions 4, 8, 9, 10, 11 still opened or added. (#1 closed)

N°	Theme	Priority	Remark	Status
1	**Install** **Ticket/ #2511**	**High**	**IN2P3 Question (submitted on 1Feb2013) :** Is the code review for ticket #2511 (scons automated install) done ? If not, what is the timeline ? This is a priority item as further development like automated tests, are dependent on this ticket. **SLAC Answer (submitted on 9Feb2013) :** Daniel send this email to Douglas : Okay, please bring the branch up to date with the master, confirm that things still work, and then put it on master. See steps 4 and 5 in the example: https://dev.lsstcorp.org/trac/wiki/GitDemoAndTutorial#LSSTgitworkflowandbranchmanagementpolicy Thanks! –Daniel **IN2P3 Question (submitted on 13Feb2013) :** Could you do the merge please ? **IN2P3 Answer (submitted on 28Feb2013) :** We've done the merge.	**Closed 2**
2	**Data–loading**	**Med**	**IN2P3 Question (submitted on 1Feb2013) :** Could we have some information about PT1.1 'Source' loading, as it would allow to complete the data loading and to execute queries for Qserv validation. Is the next command ok : */opt/qserv-dev/bin/python /opt/qserv-dev/qserv/master/examples/partition.py --output-dir /data/lsst/pt11_partition --chunk-prefix Source --theta-column 33 --phi-column 34 /data/lsst/pt11/Source.txt --num-stripes 10 --num-sub-stripes 2* are these parameters correct : *--theta-column 33 --phi-column 34*	**Closed**

			SLAC Answer (meeting of 6Feb2013) : In tests/case01/data/Source.schema zero index based of raObject and declObject are 33 and 34. 'Source' table fields ' raObject' and 'declObject' are strictly equals with fields 'ra' and 'decl' of 'Object' table. Object moves aren't modelized in these parameters.	
3	**Automated tests**	**Low**	**IN2P3 Question (submitted on 6Feb2013) :** Is the development on tests/runTest.py inside the perimeter of ticket #2014 or will a new ticket be created ? At the moment development is done in branch "u/fjammes/automatedTest" **SLAC Answer (submitted on 6Feb2013) :** **DST** : Well, yes, that can be part of the ticket. I think the important point there is that the ticket should be for only running runTest.py from the installer. Other dev. of the use of runTest.py should be another ticket. Meeting (7Feb2013) : **DST** : remove test data after test execution not to conflict with other data (like PT1.1 for example). Indeed, 'LSST__Object' and 'LSST__Source' tables in 'qservMeta' database are the same and so conflict for each dataset.	**Closed**
4	**Automated tests / Data-loading**	**Low**	**IN2P3 Question (submitted on 6Feb2013) :** The automated tests with qserv mode needs to load partitioned data. In your tests/runTest.py comments, you propose this : *#use the loadPartitionedObjectTables.py script to generate loadO* but loadPartitionedObjectTables.py doesn't exist. Should we use instead *master/examples/loader.py*, or re-implement in Python the load() function of *admin/custom/bin/qserv-admin.pl* ? **SLAC Answer (submitted on 6Feb2013) :**	**Open**

			DST : Hmm... well good question. Sounds like we need to create a re-usable python loader class, maybe need a new ticket there ? **IN2P3 Question (submitted on 13Feb2013) :** What do you think of next script (see mail sent to qserv-l@slac.stanford.edu on 8Feb) : *python master/examples/loader.py --help* It seems to do the job, it may have been written by Jacek. **IN2P3 Question (submitted on 28Feb2013) :** We now use *master/examples/loader.py* in runTests.py with success, and we plan tu use it to load PT1.1 Sources. Do you think it is a good solution ?	
5	**Data-loading**	**Med**	**IN2P3 Question (submitted on 6Feb2013) :** Furthermore, in order to allow Qserv users to save expensive storage cost., we propose to develop a dynamic loader, which load data in DB directly at partitionning, without having to store partitioned data in intermediate files ? Do you think it is interesting ? **SLAC Answer (submitted on 6Feb2013) :** **DST** : Yeah, well, we've thought of this, and there was some talk with Serge about having a type of partitioner which is more of a streaming executable. Taking data from one input, and loading it into the database as output. There is a concern, since on large data sets, these processes take days, and are then quite fragile for errors, and you don't want the loading left in some unknown state on failure, such that you need to re-start everything from the beginning. It would be good to be able to batch the partitioning and loading into smaller when contained parts, and be able to re-start on the part left over on failure. But not sure that is the best thing to work on just yet, I think we might need more experience on handle large datasets with the partitioning and loading still.	**Closed**

				Meeting (7Feb2013) : DST : complicated and cost-effective : error on failure should be managed.	
6	Minutes	Med		**IN2P3 Question : Wiki page for Phone meeting minutes (submitted 6Feb2013)** We would like to have a centralized place to put phone meeting minutes, could it be possible to use your wiki pages for that ? **SLAC Answer (submitted on 6Feb2013) :** **DST** : Um, maybe, does trac scale enough to have a new page added for each meeting? Meeting (7Feb2013) : LPC team has now created a page on the wiki : https://dev.lsstcorp.org/trac/wiki/db/Qserv/IN2P3/Meetings	**Closed**
7	Benchmarking	Med		**IN2P3 Question : Standard performance tests for QServ (submitted 6Feb2013)** Thanks for giving us next link : https://dev.lsstcorp.org/trac/wiki/dbQserv/perTableKeyCache . Does the queries defined in https://dev.lsstcorp.org/trac/wiki/dbQueriesForPerfTest also fit for Qserv master and node performance and benchmarking tests ? Beyond individual queries, is there a document describing a typical queries workload to be used for benchmarking ? Could we define standard performance tests for QServ master and nodes ? Could you provide the ones you used before ? **SLAC Answer (submitted on 6Feb2013) :** **JBA** : Regarding #7, for sample queries related to PT1.1, I'd suggest to use: https://dev.lsstcorp.org/trac/wiki/db/Qserv/250NodeTestPlan However, PT1.1 is not good enough for testing this: https://dev.lsstcorp.org/trac/wiki/db/Qserv/PerTableKeyCache	**Closed**

| | | | because this test exposes the fact that we have one huge table (RunDeepForcedSource), and some small tables, such as the Science_Ccd_Exposure, or RefObject.

It'd make more sense to use the Winter13 data set for this test. That data set is on the large side... a little over 2 TB. I believe we will be moving more towards relying on that data set in the future, so it makes sense to get it. We are in the process of partitioning it right now on slac machines. We are happy to make it available to you, if you have space to take it, either as csv files, or partitioned data.

Meeting (7Feb2013) :
DST : test with PT1.1 data with MySQL tuning parameters

IN2P3 Answer (submitted on 13Feb2013) :
This point must be delayed as in2p3 priority is to run Qserv on 250 nodes soon. So in2p3 team must focus now on installation/configuration/administration and data-loading procedures. | |
| 8 | **Data-loading** | **Med** | **IN2P3 Question : Standard performance tests for QServ (submitted 7Feb2013)**
How could we generated Qserv meta LSST_Object and LSST_Source tables with good (chunk, subchunk) data.
Meeting (7Feb2013) :
DST : watch qserv-admin.pl and try to use it.

IN2P3 Question (submitted on 13Feb2013) :
We've changed some partitionning parameters (see tests/runTests.py in ticket #2014) so that chunk and subchunk data are added in Source_ChunkIds and | **Open** |

			Object_ChunkIds table during data-loading. Then meta are added in LSST_Object and LSST_Source via of bunk of mysql queries of this type :	
			# mysql -u\<u\> -p\<p\> qservTest_case01_q qservMeta -e "insert into LSST__Object SELECT objectId, chunkId, subChunkId from qservTest_case01_q.Object_100"	
			(inspired from initial Jacek comments in tests/runTests.py).	
			It is quite simple and efficient enough for the tiny test dataset.	
			Does it seems correct ?	
			IN2P3 Question (submitted on 28Feb2013) :	
			We now use this technique	
			in runTests.py with success, and we plan tu use it to load PT1.1 Sources. Do you think it is a good solution ?	
9	**Data-loading**	**Med**	**IN2P3 Question : W13 data transfert at CC-IN2P3 (submitted on 28Feb2013)**	**Open**
			Could you please transfer us W13 data at CC-IN2P3 in /sps/lsst/Qserv/data/Winter2013/ ?	
10	**Data-loading**	**Med**	**IN2P3 Question : W13 data transfert at CC-IN2P3 (submitted on 28Feb2013)**	**Open**
			How the chunk number (7200) is computed depending on stripes and substripes, it seems to be computed as 2*60*60, what is 60 ?	
11	**Data-loading**	**Med**	**IN2P3 Question : Design of runTests.py (submitted on 28Feb2013)**	**Open**
			Would you be you interested in generalizing our design used in tests/runtTests.py, so that we continue to improve it. It mainly consists in :	
			– read and analyze input data config in a separate class (type of files, extensions, use of a config file with meta data)	
			– Use of QservDataLoader which could be used for large tests, like for PT1.1 or W13 for example. This class could be generalized for loading data on worker.	