

Tests of PoD with ATLAS Prodsys2 and first experience with HTTP Federation

Running PROOF-on-Demand in the ATLAS Prodsys2

The LHC Run-2 will present new challenges for distributed computing due to higher trigger rate and luminosities. To address these challenges, the ATLAS computing system evolved to Prodsys2 [1]. The major modifications have been the introduction, in the tasks workflow, of the following components:

- *DEFT: Database Engine For Tasks* is responsible for the Meta tasks and tasks definition;
- *JEDI: Job Execution and Definition Interface* defines the jobs to submit to PanDA optimizing the workload.

Moreover Rucio has been introduced in replacement of DQ2 for the data management.

In order to be able to continue to run PROOF-based analysis with PoD on PanDa [2], the PoD/PanDA plug-in must be changed; in particular the JEDI options to scout jobs and to automatic retry jobs must be disabled.

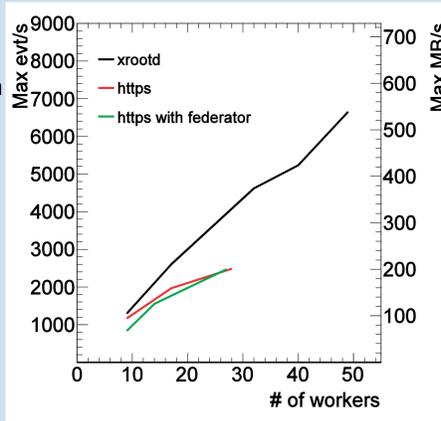
XROOTD and HTTP protocol with HTTP Federation

PoD cluster nodes in the Frascati Tier-2, input dataset on DPM at Frascati and Naples ATLAS Tier-2s.

- XROOTD protocol on DPM and FAX for XROOTD storage Federation fully in production. Both tested with PoD in a previous work [2]
 - root://atlas.inf.infn.it/dpm/Inf.infn.it/home/atlas...
- HTTP protocol with secure access, HTTPS, in production with DPM:
 - https://atlas.inf.infn.it/dpm/Inf.infn.it/home/atlas/....
- HTTP federation test-bed, [3], with federator located at Desy. Like FAX, it brings together the storage elements with the HTTP protocol under a unique namespace. Redirects clients requests to the closest available SRM:
 - http://federation.desy.de/fed/atlas/rucio/....

The overhead due to the federator is small.

XROOTD protocol seems to be more performing than HTTP.



HTTP protocol and HTTP Federation over LAN and WAN

Test performed with an analysis task running in Frascati and accessing, through the federator, a dataset available both in Frascati and Naples DPM/HTTP SEs.

To check that the federation worked, the task was launched while both SEs were up; the federator effectively redirected the workers to the Frascati SE, as expected.

To test the federator responsiveness, the task was launched a second time after bringing down the Frascati SE; the federator then redirected the workers to the Naples SE. After two minutes of task execution, the Frascati SE was enabled again: in about 5' the workers stopped accessing data from Naples and resumed local access (the observed delay is expected from the HTTP federation specs). The transition, with the related boost of performances in terms of the rate of processed events, is visible in the plot below.

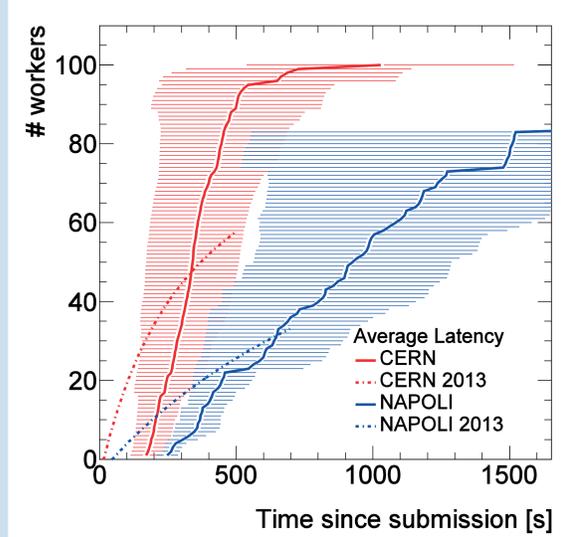
Proof-on-Demand and Prodsys2

As explained in [2], the worker availability curve as function of the time from submission can be used to measure the *startup latency* and *ramp-up slope*, two parameters affecting the *time-to-result* for an analysis.

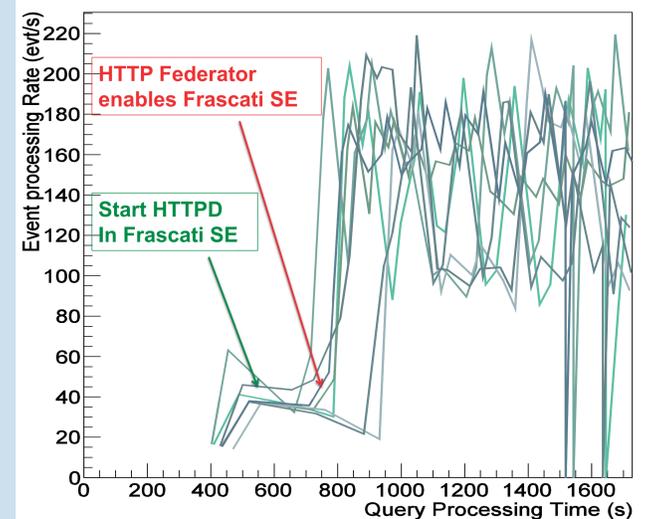
The measurements presented in [2] were repeated with an example of large site (CERN, red results) and an example of small site (Naples, blue results). Average (full line) and RMS (hashed area) have been calculated.

With respect to the previous measurements done in 2013, we find that

- The startup latency has significantly increased; this can be explained by the additional layers introduced in ProdSys2
- The ramp-up slope has also increased, indicating an increase in the amount of resources available in the sites



Second analysis task



Latency tests with dynamic worker addition

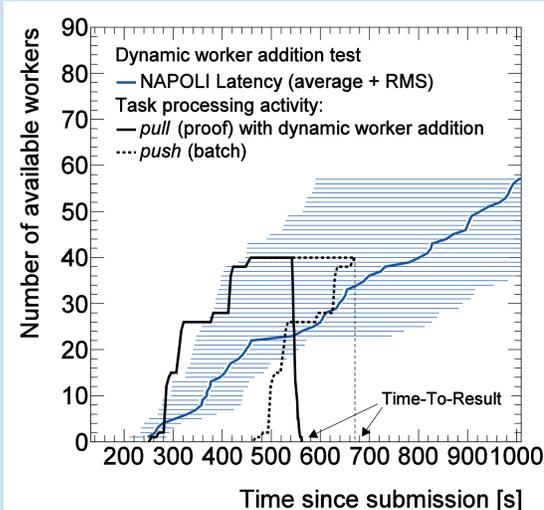
Pull model and dynamic worker addition

- Address cases of limited resources with inherent delays in worker availability
- *Pull* architecture for work assignment expected to be more efficient than *push* one [4]
- Effect larger for sites with less resources

Recent PROOF developments [5]

- Support for worker addition during data processing
- Possibility to start processing upon availability of first worker

The measurement was performed on a relatively small site, Naples Tier-2. The analysis task required about 150' of serialized time, reading a dataset stored locally. The model in [4] predicts a processing time of ~700s in pull mode and a ratio *push-vs-pull* time-to-result ratio of 0.73 or a *pull-vs-push* average speed-up of ~27%.



The analysis task was executed in PROOF query with dynamic worker addition enabled.

A PoD request for 40 workers is submitted to PanDA.

With a pull model (PROOF, solid black line) processing starts as soon as the first worker is available. The first worker processes for a much longer time than the last one.

The dashed black line shows what worker activity would look like in push-mode, assuming the same worker availability curve and the same amount of work for each worker. The completion time of the last starting sub-job determines the time-to-result.

In the example chosen the *pull-vs-push* speed-up in time-to-result is ~20%. This is in good agreement with the prediction, considering that the observed ramp-up rate is on the upper side of the distribution observed for the site.

Conclusions

- Initial tests with PoD/PanDa with ProdSys2 positive; an increase startup latencies is observed
- Tests of data access via the HTTP protocol seems to indicate that XROOTD is more performing
- First functional tests of the HTTP federation are positive; intensive tests are however required for an evaluation of the full functionality

References

- [1] S. Campana, A. Di Girolamo, "ATLAS Distributed Computing in LHC Run2", talk at CHEP2015 .
- [2] E Vilucchi et al, "PROOF-based analysis on the ATLAS Grid facilities: first experience with the PoD/PanDa plugin", J.Phys.Conf.Ser. 513 (2014) 032102 .
- [3] F. Furano, "Dynamic Federation", white paper 2015, http://svnweb.cern.ch/world/wsvn/lcgdm/ugr/trunk/doc/whitepaper/Doc_DynaFeds.pdf .
- [4] D Berzano et al, "PROOF as a Service on the Cloud: a Virtual Analysis Facility based on the CernVM ecosystem", J.Phys.Conf.Ser. 513 (2014) 032007 .
- [5] Full functionality available in ROOT 5.34/28 and in the forthcoming ROOT 6.04 .

Acknowledgments

- Thanks to F. Furano for his contribution on HTTP federation .
- This work was developed in the framework of the PRIN Project "STOA-LHC 20108T4XTM", CUP: I11J1200080001, and partly supported by it .