Service Incident Report

Type of Incident: AFS latency
Location: IN2P3-CC
Duration: 6 months
Date: July the 8th 2010 to January the 7th 2011
Authors: Luisa Arrabito, Pierre Girard
Report status: Final
Report logs:
- 2011-01-20: Creation by Luisa Arrabito
- 2011-02-06: Internal review
- 2011-02-11: Rolf Rumler corrections
- 2011-02-14: Final version sent to wlcg-scod@cern.ch

Description

A fraction of LHCb jobs, ranging from 1% to 6%, was systematically failing because they exceeded the timeout set for the environment setup of the LHCb application (600 s). This indicates a too slow response of the site's shared file system AFS, which contains the software installation.

Timeline

- **08/07/2010:** ticket opened against CC-IN2P3 because a fraction of LHCb jobs was failing during the software setup (see the description section). Also specific LHCb SAM jobs indicated some shared area slowness: https://gus.fzk.de/ws/ticket_info.php?ticket=59880

  IN2P3 people don't see any overload on the AFS server, which is generally the cause of such slowness. The investigations then focused on the client side.

- **13/07/2010:** the error is reproduced locally by IN2P3 people by running the LHCb code which executes the software setup. The failure rate obtained is close to the one observed by LHCb jobs.

  A first analysis shows that at least one of the causes of such slowness is the SL5 tuning applied to WNs, whose aim was to reduce the generalized problem of SL5 WNs crashes. IN2P3 people propose to temporary increase the timeout set in the LHCb application, while they keep working on the SL5 tuning. LHCb prefers to keep the timeout as it is until a solution is found.

- **From 13/07/2010 until 04/10/2010:** different kernel parameters are tested, but no meaningful improvement is observed.
- **04/10/2010:** deployment of an LHCb test infrastructure. Dedicated sets of WNs are configured with different combinations of tunings. The tunings applied cover:
  - SL5 tuning: customized kernel parameters and improved network driver;
  - AFS cache tuning: cache parameters and cache size;

- WNs isolation: exclusion of atlas jobs, which have a similar intensive access to the shared area, and LHCb isolation from all other users;
- AFS DB updated and dedicated to a set of WNs.

Jobs running the LHCb software setup are systematically submitted by IN2P3 people on different sets of WNs and the execution time is monitored in order to identify the best tuning.

- **25/11/2010:** a higher failure rate (~30%), due to timeout, affects LHCb jobs of some reprocessing productions. It turns out that most of the failures occurred on 24 cores WNs, thus confirming the idea that a tuning of the number of concurrent jobs is necessary on these machines.
- **26/11/2010:** LHCb agrees to increase the timeout while investigations go on. LHCb is however asked to keep unchanged the timeout for the LHCb SAM test in order to keep a visible trace of the original problem.
- **28/12/2010:** a development version of the AFS client is tested in combination with the ongoing tests.
- **04/01/2011:** a best tuning is found with 21 concurrent jobs on the 24 cores machines and the new tested version of the AFS client.
- **07/01/2011:** all 24 cores WNs are configured to allow 21 concurrent jobs and are put back in production. The new AFS client is also progressively deployed on the whole cluster. End of the incident.

Analysis

Analysis of the LHCb environment setup

In order to better understand why the problem particularly appeared with LHCb environment setup, its execution was traced with the “strace” utility. The result shows that this setup is particularly consuming in terms of file access operations.

- 110765 stat operations
- 17868 open operations

So many file accesses during the environment setup of each LHCb job could potentially overload the AFS servers, but that has not really been observed on CCIN2P3 AFS servers while LHCb timeout occurred.

It was also considered the possibility that the overload was put on the AFS DB servers which are in charge of the AFS meta-data. As there is no easy way to check if there is or not any overload on this kind of servers, an AFS DB server was dedicated to a small subset of WNs. If there was an overload problem with the AFS DB servers, then this dedicated configuration should have shown improvement. But no real progress was noticed.

That finally led CCIN2P3 experts to focus on the client side and then to test different AFS client configurations.

Analysis of first tests campaign results

The new tests consisted in probing different WN configurations by changing the number of AFS daemons, the size of the AFS cache, etc.

In the Annex A, the Figure 1 is an historical view (for sake of clarity only the last 2 months are displayed) of the mean time (over the jobs of the day) needed to execute the LHCb

environment setup on the different sets of WNs, while in the Figure 2, the same view is given for the timeout rate (with a timeout fixed at 600s) of each set of WNs. None of these tunings has shown good performances, except on WNs with atlas jobs excluded. On the contrary, the standard configuration of our cluster (blue curve) has better performances than dedicated WNs.

This last observation means that the problem is particularly concentrated on the machines used for the tests. One common feature of those WNs is to be the most recent hardware, the only ones with 24 virtual cores machines (2×6 cores with hyper-threading). Consequently, such WNs are configured to allow a greater number of concurrent jobs (28) than the other ones of the cluster.

The problem was then reinterpreted as an AFS client performance degradation correlated with the number of simultaneous running jobs.

Analysis of second tests campaign results

New tests were run to stress AFS to better understand the effect of the increase of simultaneous jobs on a WN. The test job type is very similar to LHCb environment setup by being very AFS access intensive:

- 100000 stat operations
- 7000 open operations

The Figure 3 shows the results of the tests run by using AFS, NFS or local FS. The growth of execution time seems to be linear for the local FS, but is slightly curved for both AFS and NFS with an important upwards slope.

Analysis of third tests campaign results

Given the previous result, the investigations then focused to find the number of concurrent jobs giving the best compromise between the CPU efficiency of the jobs and the overall computing power of our cluster. Also the influence of hyper-threading deactivation has been studied.

In Annex C, the Figure 4 is an historical view of the mean time needed to execute the software setup on machines having a varying number of concurrent jobs and hyper-threading switched on or off. For comparison one machine is kept with standard settings (blue curve). The same historical view is given in Figure 5 for the timeout rate.

Conclusion

The environment setup phase of LHCb jobs has shown to be very intensive in terms of AFS access operations. Moreover, at the WN level, AFS client becomes a critical resource which can significantly increase the execution time of a job according to the number of concurrent AFS-consuming jobs.

The timeout problem of LHCb jobs is a direct consequence of AFS latency on WNs running too many jobs at the same time. Those WNs were identified as being the latest CCIN2P3 WNs, with 24 logical CPUs, which were initially configured according to HEPSpec2006 benchmark results. Obviously, this CPU benchmark doesn’t take into account the AFS latency effect on job efficiency.

Impact

A fraction of LHCb jobs, ranging from 1% to 6%, was systematically failing because they exceeded the timeout set by the LHCb application on the software setup phase.

On November the 26th 2010, as requested by CCIN2P3, the job timeout was increased by LHCb. Since, the job failure problem has moved to a job efficiency problem which drastically reduced the impact on operations from both VO side and site side.

Corrective actions

The number of concurrent jobs on all 24 core machines was reduced from 28 (24 grid jobs) to 21 (18 grid jobs). Also a development release (1.5.78) of the AFS client is now deployed on the whole cluster because it has been proven to be much more efficient than the latest public release (1.4.12), and reliable enough according to our local tests.

Annex A: AFS configurations tests

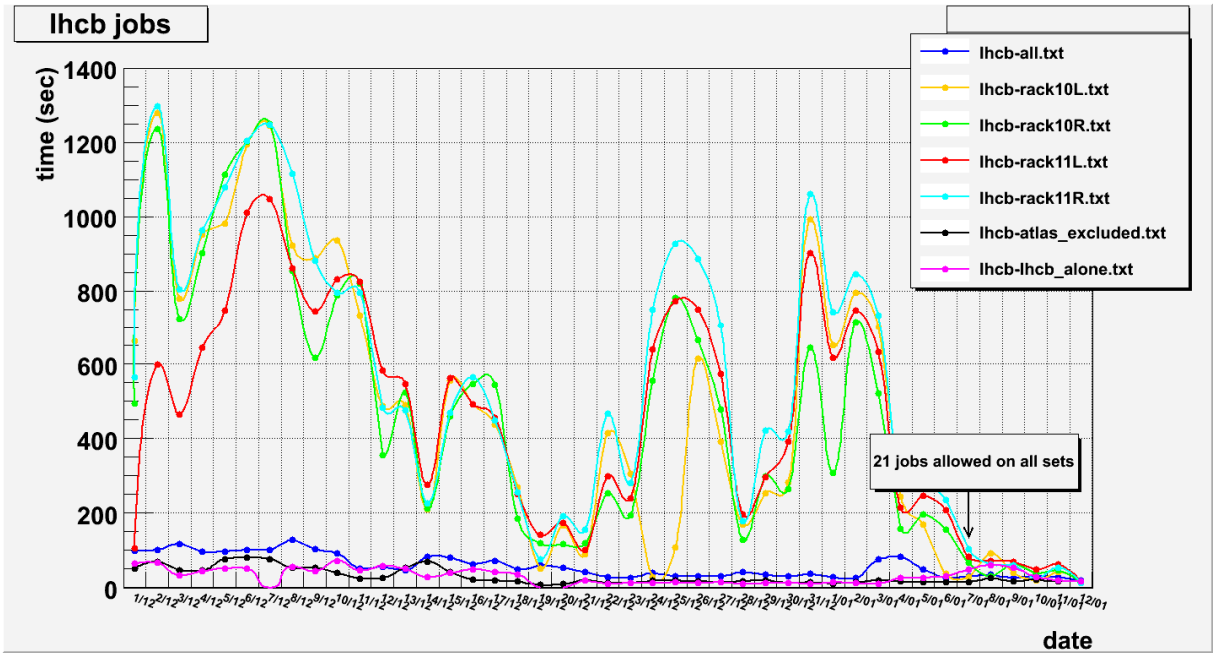

Figure 1: 2-monthes view of the time mean according to different AFS configurations

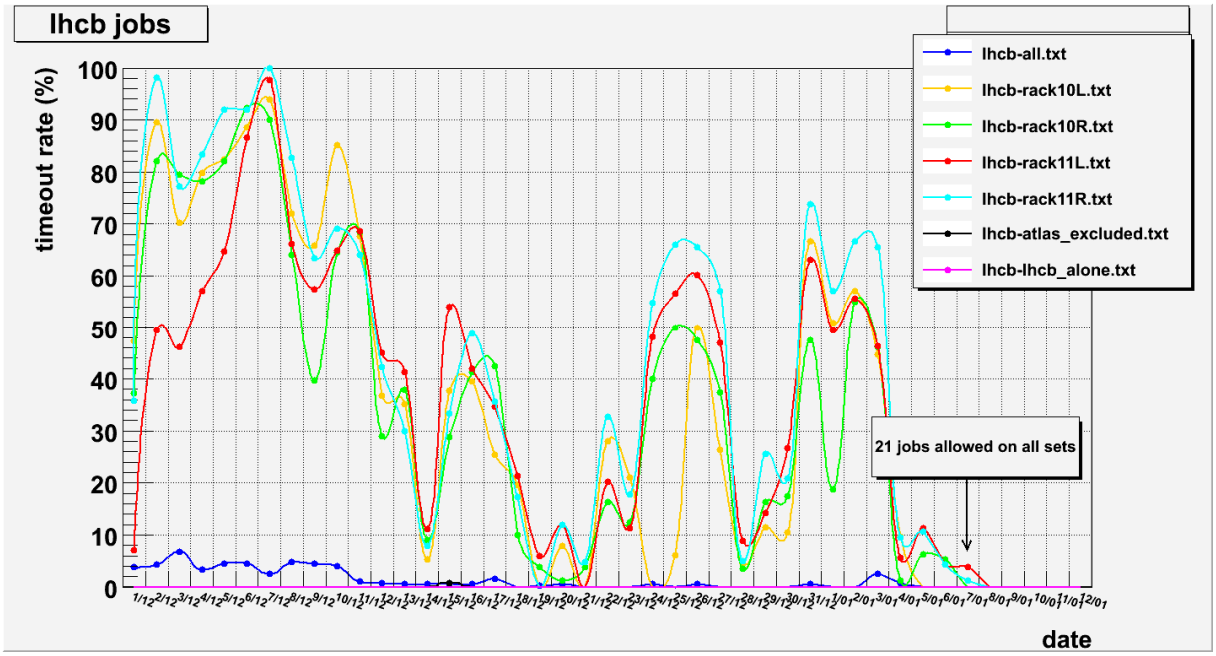

Figure 2: 2-monthes view of timeout rate according to different AFS configurations

Annex B: AFS stress tests

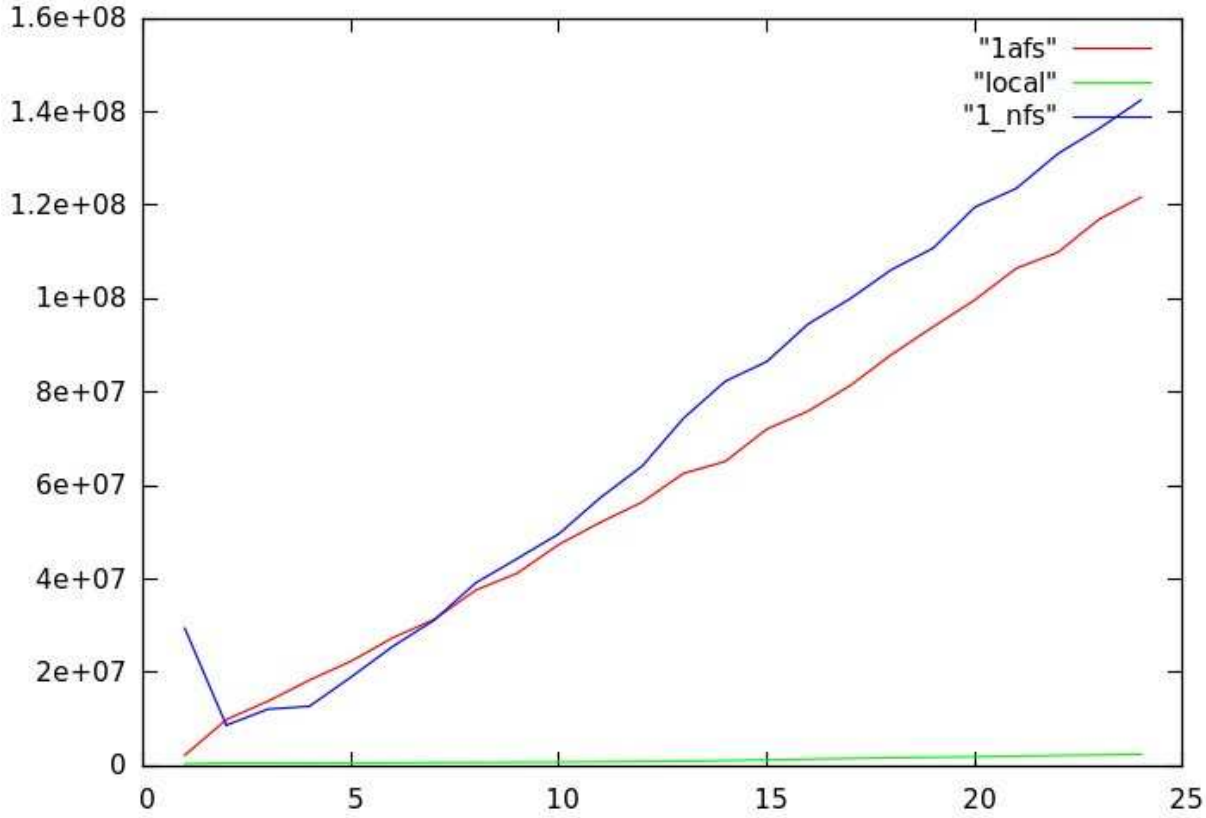

Figure 3: Execution times (in µs) by number of simultaneous test jobs on one WN for both AFS, NFS4.0 and local FS.

Annex C: WN Job slots configurations tests

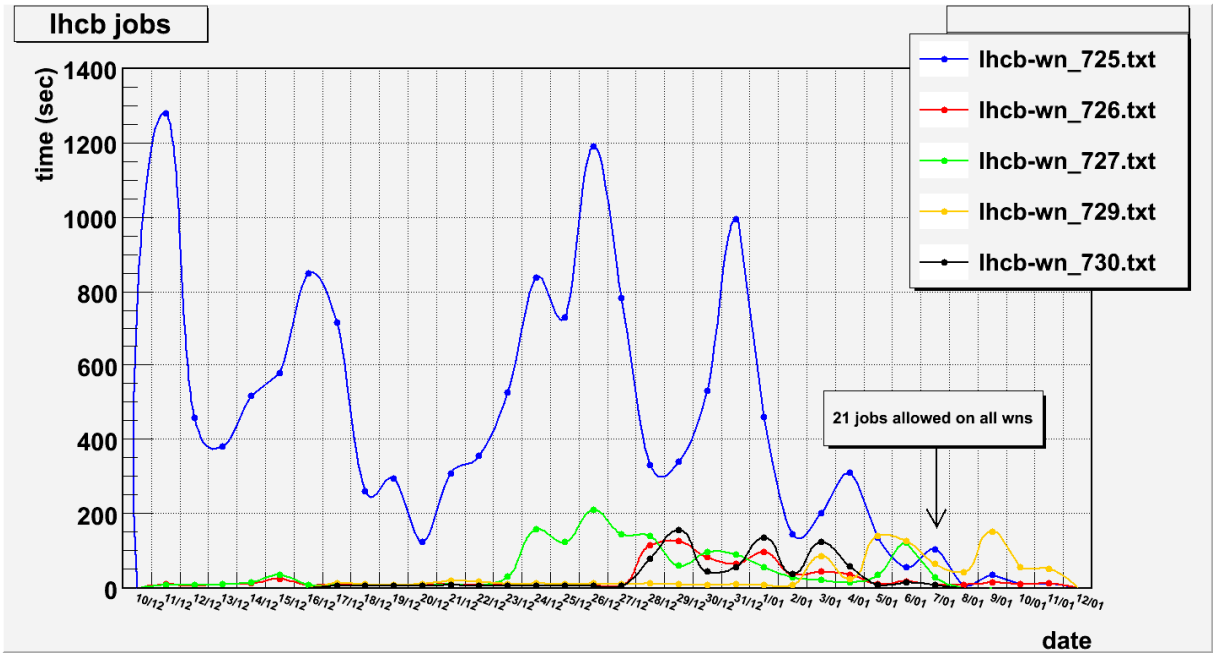

Figure 4: Historical view of the time mean according to different job slots configurations

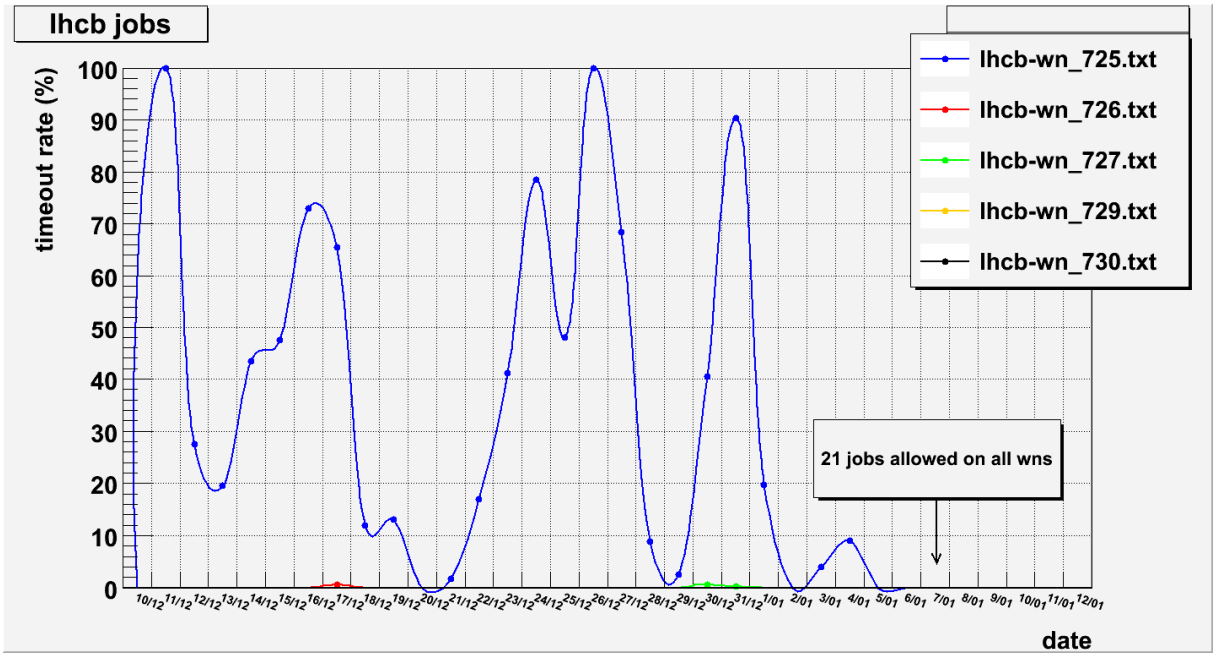

Figure 5: Historical view of timeout rate according to different job slots configurations