Monitoring system for the Belle II distributed computing

Kiyoshi Hayasaka for the Belle II computing group
Kobayashi-Maskawa Institute for the Origin of Particles and the Universe, Nagoya University
Chikusa-ku Furo-cho, Nagoya, Japan
E-mail: hayasaka@hepl.phys.nagoya-u.ac.jp

Abstract. Belle II experiment is a next-generation B-factory at KEK in Japan, which will collect 50 ab\(^{-1}\) data sample for 10 years, that corresponds to about \(5 \times 10^{10}\) \(B\bar{B}\)-pair events. To handle such a huge data sample, Belle II has adopted the distributed computing. A monitoring system is necessary to operate the computing system stably and we have been developing the monitoring system for Belle II computing based on our experience we have gained through the mass production test of the Monte Carlo simulation events. In this paper, we introduce our monitoring system, especially, the one we call “active-way” monitoring.

1. Introduction

Belle II experiment is a next-generation B-factory at KEK in Japan, which will start for physics run without (with) vertex detector in 2017 (2018). For 10 years, Belle II will collect 50 ab\(^{-1}\) data sample, which corresponds to about \(5 \times 10^{10}\) \(B\bar{B}\)-pair events. To analyze this huge number of events, we roughly need to handle 1 MHS06 CPU resources, 100 PB storage for one set of raw data and 100 PB one for Monte Carlo (MC) simulation and analysis-data events, finally. Thus, we adopt distributed computing technique. As a distributed computing software framework, DIRAC (Distributed Infrastructure with Remote Agent Control) has been adopted, which is originally developed for LHCb and can handle grid, cloud and local traditional cluster resources. \[1\] At the present, around 40 sites participate in Belle II computing, where 25,000 concurrent jobs are handled at peak, as shown in figure 1. To understand the system well, for example, to find a bottle-neck, to check the scalability and the stability of the central servers, we have performed the mass production tests of the MC simulation events, called MC campaigns, where the MC events same as that Belle II collaborators actually analyze are generated using Belle II software. On our distributed computing system, Belle II software is provided through CernVM File System (CVMFS). \[2\]

2. Experiences through the operation

We have performed MC campaigns four times so far and experienced many things through the operation. In particular, we have faced several kinds of typical troubles such as problems on Computing Element (CE), on Worker Node (WN), on the central servers which execute DIRAC components and so on. Let me introduce several examples here: When some problem happens on CE, the pilot job, which makes the Belle II job-executable environment beforehand, cannot be accepted by CE, fails to transfer payload files or stays idle for a long period on CE. In the
Figure 1. (Left) History for the number of the concurrent jobs on Belle II computing system from 7th Oct. to 6th Nov. in 2014. (Right) Location of sites joining Belle II computing. Green and red indicate enabled and disabled to submit a job, respectively.

first case, CE is down. In the second case, CE’s certificate or proxy on the pilot job is expired. In the last case, CE is alive but batch job system is down. Of course, they are the most probable causes based on our experience through the operation and there are many kinds of exception. For example, merely when the network is so busy, the payload transfer is failed.

When a WN is problematic, the pilot job can start, but, it cannot make the environment for the Belle II jobs and finishes immediately without running a payload job. In many cases, the WN has hardware trouble such as HDD failure. Sometimes, disk space on some WN is insufficient due to bad-behavior jobs which remain many files. Newly joined site or newly added WN tends to forget to install some required rpm packages. Occasionally, clock adjustment is out of work. Since Belle II software is provided via CVMFS on each WN, it is expected that CVMFS works well stably. Here we list up the problems relating to CVMFS, that we have faced: (i) CVMFS is not mounted. (ii) CVMFS is mounted but some files on CVMFS can not be read with I/O error. (iii) Files on CVMFS can be read but they are not up-to-date. In the last case, the Belle II job can run but it may make a wrong result.

Since these problems make the efficiency of our resources low, we like to detect and fix them as quickly as possible.

3. Belle II monitoring system

Although DIRAC already has its own monitoring system, some of the problems mentioned in the previous section are hard to detect with the existing monitors. Therefore, we have been developing our own monitoring system. We classify the monitoring ways into two categories, i.e., the passive and active ways. Monitoring in the former utilizes information collected by DIRAC, makes statistics and visualizes them. While it is discussed in [3], Monitoring in the active way needs several tools outside DIRAC, which probe problems on WNs, CEs and so on, and we newly have developed them.

3.1. SiteCrawler

SiteCrawler is a system to collect information on each WN such as CPU type, no. of cores, amount of the equipped memory, disk free space, OS type and version, mounted CVMFS revision, and diagnoses the WN concerning installed rpm packages, open ports, http proxy status, clock adjustment and Belle II software environment. To collect the above information, SiteCrawler submits a test job to each site via DIRAC once a hour periodically. The result is stored in DB and summarized in the web page as shown in figure 2. SiteCrawler can detect the change of the
settings such as opening ports and so on, for example, after downtime. In addition, when a new site jobs or some WNs are added, SiteCrawler is helpful.

| Site status summary |
|---------------------|
| Site | worker node | DIRAC | Name | Host | Status | Date |
|---------------------|
| Site1 | worker node1 | DIRAC | Name1 | Host1 | Status1 | Date1 |
| Site2 | worker node2 | DIRAC | Name2 | Host2 | Status2 | Date2 |
| Site3 | worker node3 | DIRAC | Name3 | Host3 | Status3 | Date3 |

Figure 2. Summary of the result obtained by SiteCrawler. First column shows the site name defined inside DIRAC and the second shows the latest investigated worker node's name. From the third column, the collected information and the diagnosed results on the worker node. Shaded rows indicate the site on which SiteCrawler does not execute more than 2 days. By clicking the site name, the previously investigated results on other worker nodes can be seen.

3.2. CE test job submitter

In order to check CE’s health, we have developed CE test job submitter. When a failure rate for DIRAC’s pilot job submission on some CE is higher than 70% or SiteCrawler result is not updated more than 6 hours, this tool submits a test job to the CE and stores the submission result and its status history after the submission if the submission is succeeded. In figure 3, the web interface of the CE test job submitter is shown. When the submission is failed, by the failure reason from the output of the submission command, we can know what happens on the CE. In the case where the job submission is succeeded but the job status does not turn to ‘running’, CE works but something is wrong inside the site, for example, its batch queue system is down, CRL or CA is not up-to-date. We can guess it from the status history of the job or the reason why the job is aborted. If everything goes well, it turns out that the problem is outside the CE anyway. At the present, only LCG sites are supported and the job submission test is performed by “glite-ce-job-submit”. In the future, OSG and traditional cluster sites will be supported.

![CE Job Submission test result](image)

Figure 3. (Left) Sites on which some job submission recently is failed. (Right) History of the job submission test and its result. By clicking “log”, the output from “glite-ce-job-submission” or “glite-ce-job-status” can be checked.

3.3. DIRAC port checker

When a central DIRAC server receives more requests for some DIRAC service such as providing the configuration than it can manage, the server will be down and the unexpected crash of the
server may make some damage to the server, for example, HDD physical failure, inconsistent inode, unflushed files. Besides, when some service is not supplied for the running pilot jobs, they stops. To avoid this kind of trouble, we make a load distribution of the crowded DIRAC services and monitor of ports used by the services on the DIRAC servers. When a DIRAC service is crowded, the corresponding port gets busy. Therefore, we check if the port can be open or not using Berkeley sockets. When some port can not be open, the operators are reminded by the e-mail message. The history is visualized by MUNIN [4] as shown in figure 4. Also, this helps to check if the load distribution works well or not.

Figure 4. History of a response of ports used by central DIRAC servers, where 1 (0) means that the port can (not) response. This figure shows, around 8:00 on Monday, the port 9135 on dirac3.cc.kek.jp was busy while around 14:00 on Monday, the port 9135 on dirac4.cc.kek.jp was busy within a very short period.

4. Summary
When a huge computing resources is handled, it is important to monitor the whole system for the effective operation. Belle II computing utilizes 1 Million HS06 CPU resources and $O(100)$ PB storage and is handled by DIRAC to realize the distributed computing. For our operation, DIRAC’s own monitoring system is not so sufficient. Therefore, we need to develop our monitoring system based on the experience through the several MC campaigns. In this paper, we have introduced our monitoring tools, which are called the active way. They can detect several kinds of the problems on the WNs, CEs and the central servers. In the near future, we make them more automated and develop an automatic notification system to the site maintainers to operate Belle II computing more efficiently.

Acknowledgements
We are grateful for the support and the provision of computing resources by CoEPP in Australia, HEPHY in Austria, McGill HPC in Canada, CESNET in the Czech Republic, DESY, GridKa, LRZ/RZG in Germany, INFN-CNAF, INFN-LFN, INFN-LNL, INFN Pisa, INFN Torino, ReCaS (Univ. & INFN) Napoli in Italy, KEK-CRC, KMI in Japan, KISTI GSDC in Korea, Cyfronet, CC1 in Poland, NUSC, SSCC in Russia, SiGNET in Slovenia, ULAKBIM in Turkey, UA-ISMA in Ukraine, and OSG, PNNL in USA. We acknowledge the service provided by CANARIE, Dante, ESnet, GARR, GEANT, and NII. We thank the DIRAC and AMGA teams for their assistance and CERN for the operation of a CVMFS server for Belle II.

References
[1] http://diracgrid.org/, A. Tsaregorodtsev et al., J. Phys.: Conf. Ser. 119 (2008) 062048
[2] http://cernvm.cern.ch/portal/filesystem
[3] “Job monitoring on DIRAC for Belle II distributed computing”, Y. Kato et al., Proceedings of CHEP2015
[4] http://munin-monitoring.org/