Journal of Physics: Conference Series **664** (2015) 052028 doi:10.1088/1742-6596/664/5/052028

Belle II production system

Hideki Miyake[1], Rafal Grzymkowski[2], Radek Ludacka[3], and Malachi Schram[4], on behalf of the Belle II computing group[5]

[1]High Energy Accelerator Research Organization (KEK), Japan
[2]Institute of Nuclear Physics, Krakow, Poland
[3]Charles University, Prague, the Czech Republic
[4]Pacific Northwest National Laboratory, Richland, Washington, USA

[5]https://belle2.cc.kek.jp/ twiki/pub/Public/ComputingPublic/AuthorList4Belle2Computing.tex

E-mail: hideki.miyake@kek.jp

Abstract. The Belle II experiment will record a similar quantity of data to LHC experiments and will acquire it at similar rates. This requires considerable computing, storage and network resources to handle not only data created by the experiment but also considerable amounts of simulated data. Consequently Belle II employs a distributed computing system to provide the resources coordinated by the the DIRAC interware. DIRAC is a general software framework that provides a unified interface among heterogeneous computing resources. In addition to the well proven DIRAC software stack, Belle II is developing its own extension called BelleDIRAC. BelleDIRAC provides a transparent user experience for the Belle II analysis framework (basf2) on various environments and gives access to file information managed by LFC and AMGA metadata catalog. By unifying DIRAC and BelleDIRAC functionalities, Belle II plans to operate an automated mass data processing framework named a "production system". The Belle II production system enables large-scale raw data transfer from experimental site to raw data centers, followed by massive data processing, and smart data delivery to each remote site. The production system is also utilized for simulated data production and data analysis. Although development of the production system is still on-going, recently Belle II has prepared prototype version and evaluated it with a large scale simulated data production. In this presentation we will report the evaluation of the prototype system and future development plans.

1. Introduction

SuperKEKB/Belle II experiment is designed to provide 40 times larger instantaneous luminosity ($8 \times 10^{35}\mathrm{cm}^{-2}\mathrm{s}^{-1}$) compared with its predecessor, KEKB/Belle experiment. The accumulated raw data is expected to reach at 200 PB by the end of the experiment. Disk storage for Monte Carlo (MC) production, data processing, and user analysis will reach at 150 PB. The computing system must handle this amount of data and requires the integration of a variety of computing resources. The Belle II computing model (figure 1) enables, various types and sizes of computing resources to work together to deliver the required work packages. A detailed description of the Belle II computing model can be found from Ref. [1].

We employ DIRAC [2] interware to orchestrate the use of the resources via a unified interface. Consequently the Belle II distributed computing system is built upon DIRAC.

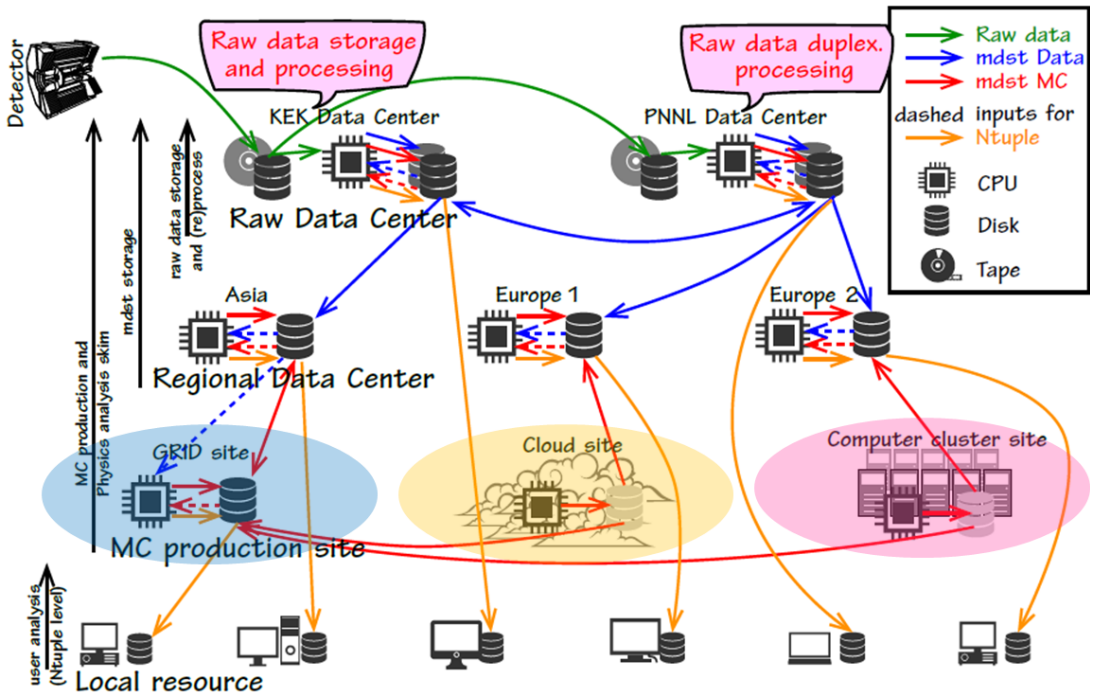

Figure 1. The Belle II computing model.

2. BelleDIRAC

We are developing own analysis software framework named basf2. Basf2 is widely utilized from central data acquisition to user analysis. Although basf2 itself equips no interface to distributed resources, these functionality are provided by gbasf2 (grid basf2). Gbasf2 is separated into a UI which works on user desktop and a job wrapper which works on Worker Nodes (WN). The job wrapper runs on DIRAC clients as shown in figure 2.

In Belle II, the file catalog is provided by LCG File Catalog (LFC) [3]. Gbasf2 accesses the LFC through the DIRAC API which calls a wrapper library written in Python. We use a metadata catalog provided by ARDA Metadata Catalog Project (AMGA) [4]. Although the DIRAC API does not provide an interface to AMGA, AMGA developers and Belle II cooperatively developed an AMGA API [5] written in Python. Input and output data transfer between storage elements (SE) are also handled by gbasf2 through the DIRAC API. To determine the source SE, gbasf2 first looks in the local SE for the data, then accesses the next closest SEs and finally interrogates other SEs.

Gbasf2 is designed to provide transparent job execution to both local and distributed resources. Indeed gbasf2 works with same steering file employed by a local basf2 job. Remote job submission, execution and data management are enabled via a series of command-line python scripts called gb2 tools. Currently about 40 commands are implemented as gb2 tools, e.g. job monitoring, canceling, and re-submission. DIRAC enables extensive customization via extension modules which are completely integrated with the base system. In case of Belle II, the extension module is called BelleDIRAC and gbasf2 is implemented as a part of BelleDIRAC. As of May 2015, BelleDIRAC consists of more than 30000 lines of Python code and shell scripts. BelleDIRAC provides extensions of the existing API and also supports additional features including gbasf2 and monitoring. Detailed explanations of the monitoring implementation is available in Ref. [6, 7].

2.1. AMGA API

AMGA Python API is developed by both AMGA developers and Belle II as stated before. Direct communication to AMGA server is given by so called low level API that is provided by AMGA developers, while Belle II specific features like schema definition are provided by high level API, developed by Belle II. Both APIs have been cooperatively being developed with exploiting feedback from Belle II use case especially for its scaling evaluation, as shown in Ref. [8].

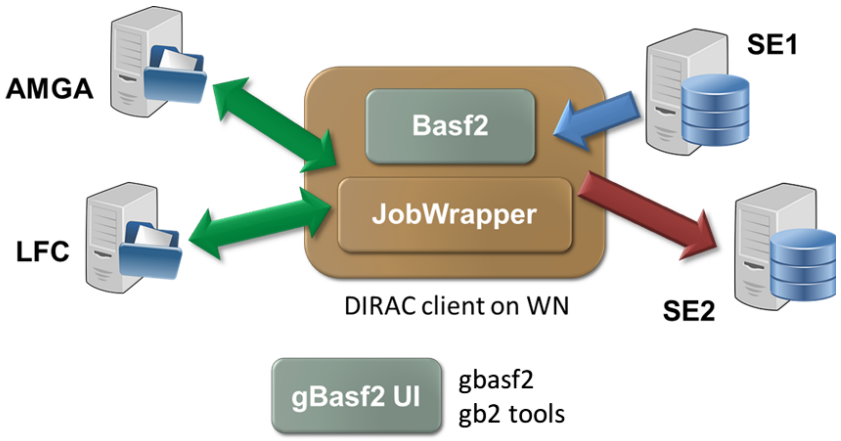

Figure 2. Schematic diagram of gbasf2 workflow

3. Central computing system

Currently KEK DIRAC cluster is composed of six DIRAC servers. There are special DIRAC slaves at University of Victoria (Canada), Pacific Northwest National Laboratory (USA), and Nagoya University (Japan). These remote slaves take up job submission to each local batch system. Nagoya slave also submits a job to other remote batch system through SSH access and also provides monitor of job submission and execution. All other functionalities are supported by KEK DIRAC cluster.

4. Belle II production system

As stated above gbasf2 and gb2 tools provide job submission and the management of submitted jobs. It works well for interactive job handling, however, is not suitable to take care of large amount of jobs which are continuously submitted and executed. If one (say production manager) wants to deal with such large number of jobs, assistant system is essential to support production manager by automated task handling, especially for monitoring and error recovery. Belle II production system (B2PS) is such automated task handling system dedicated for Belle II experiment.

Currently B2PS is based on DIRAC Transformation System, which is data driven task handler. As shown in the figure 3, B2PS exploits modular structure. Each sub component works in parallel under the tracking by Production Management System. This structure also allows us to develop each functionality by individuals in parallel.

Here is short description of each sub component.

Production Management accepts production request and tracks each step of the workflow.

Data Fabrication follows given steering file and fabricates output data. If input data exists, they are also handled by fabrication system and job execution site is determined to provide input data efficiently.

Data Validation tests fabricated events whether they always qualify given criteria.

Data Distribution transfers output data to remote distributed resources which are determined by production request or automatic determination using distributed resource status (e.g. network status, available disk space, and site down time).

Monitoring collects real time resource information and provide its summary to each sub component and also controls production workflow through the production management system.

5. Prototype production system

B2PS development has been started since 2014 and still the development is ongoing. To assure the system concept and also accumulate practical knowledge, we have developed prototype production system (PTPS). PTPS focuses on quite basic functionality of the B2PS, e.g. production management and data fabrication. Each sub component also equips the least functionality. For example there is neither GUI nor bookkeeping feature in the production management system. However it is very useful to prove our concept and find initial bottlenecks of our computing system.

Data fabrication system tracks submitted gbasf2 tasks and resubmits them if they fail. If the number of retry exceeds the limitation value, the system stops the specific task execution and asks manual treatment for production manager. To assist production manager, simple error diagnosis is performed based on error signal and summarized in data base.

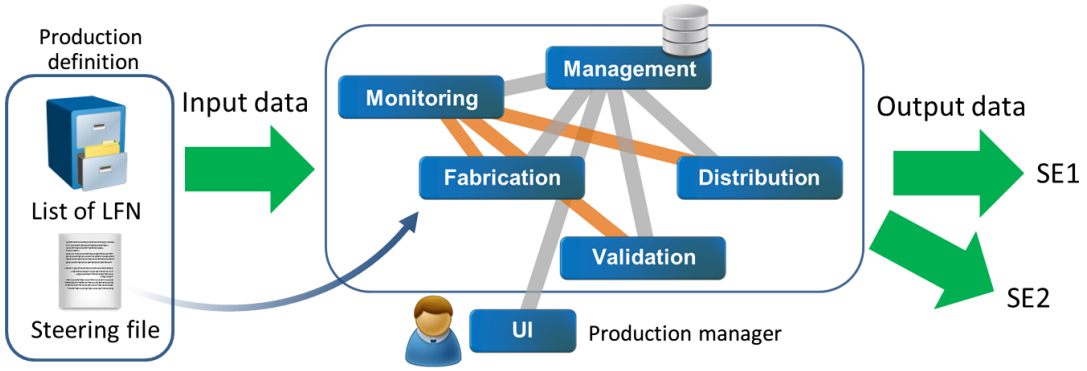

Figure 3. The conceptual design of the Belle II production system

6. Performance and next goal

Figure 4 shows concurrent running jobs of the Belle II distributed computing system. Typical running jobs recently processed are 15-20 K jobs and the system load is sufficiently lower than the saturation. Indeed the number of executable jobs are not limited by system but open slots at each distributed site. Maximum number was recorded during previous MC mass production campaign held at Fall in 2014 and exceeded 25K jobs. Thus the basic concept of Belle II production system is proven.

However MC production is rather simple task compared to practical operation which run production, data transfer, and distributed analysis. A production system should handle such complicated production requests.

Data fabrication system should be improved to handle such tasks. PTPS is useful to complete production task but when failure jobs are resubmitted, output sandbox and most logs are removed from DIRAC. From the view point of error diagnoses and monitoring the behavior is not adequate. Furthermore if output file name of new job is same as old job, sometimes the former

output prevents from job completion when the former output is stored on bad performance or down SE. We plan to replace PTPS with refactored fabrication system which does not resubmit failure jobs but submit new jobs which produces different file name.

In addition to the fabrication system, the data distribution system and the monitoring system are being developed in parallel. Web based GUI interface and bookkeeping system is essential for practical operation so that non expert can take computing shift and make production request. Next goal is thus integration of these sub components with the production system. After that B2PS is utilized to process various type of productions so that we can prove the system can be utilized coming data taking.

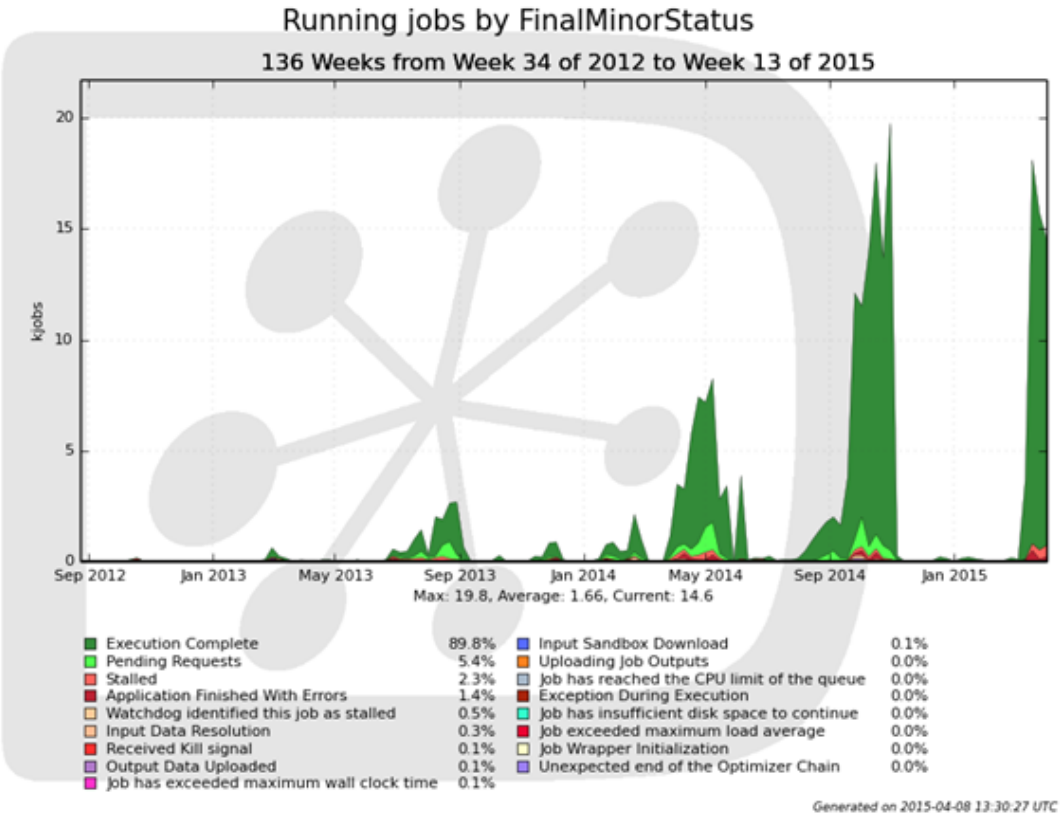

Figure 4. Concurrent running jobs of the Belle II distributed computing system.

7. Summary

Belle II collaboration develops own distributed computing system based on DIRAC interware. Since the production scale of the computing system is growing up, we need Belle II production system, which automatically performs data processing. Although the development of the Belle II production system is undergoing, its prototype was evaluated at the last MC mass production campaign and proved the basic concept. However, there are many necessary improvements and developments, especially for smart data distribution and automatic system control based on monitoring information. Our next goal is to integrate them all and perform further large scale production under realistic environment.

8. 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 Torsion, 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] "Computing at the Belle II experiment", Hara T, CHEP2015 proceedings.
[2] Casajus A *et al.* [LHCb DIRAC Collaboration] 2010 J. Phys. Conf. Ser. 219 062049, **2010**; Tsaregorodtsev A *et al.* J. Phys. Conf. Ser. 219 062029, **2010**.
[3] https://twiki.cern.ch/twiki/bin/view/LCG/LfcGeneralDescription
[4] Ahn S *et. al.* Journal of the Korean Physical Society 57 issue 4 715, **2010**.
[5] "Improvement of AMGA Python Client Library for the Belle II Experiment" , Kwak J. H, CHEP2015 proceedings.
[6] "Monitoring system for the Belle II distributed computing", Hayasaka K, CHEP2015 proceedings.
[7] "Job monitoring on DIRAC for Belle II distributed computing", Kato Y, CHEP2015 proceedings.
[8] "Directory Search Performance Optimization of AMGA for the Belle II Experiment", Park G, CHEP2015 proceedings.