Computing at Belle II

Thomas Kuhr
Karlsruhe Institute of Technology, Institut für Experimentelle Kernphysik,
Wolfgang-Gaede-Str. 1, 76131 Karlsruhe, Germany

E-mail: Thomas.Kuhr@kit.edu

Abstract. Belle II, a next-generation B-factory experiment, will search for new physics effects in a data sample about 50 times larger than the one collected by its predecessor, the Belle experiment. To match the advances in accelerator and detector technology, the computing system and the software have to be upgraded as well. The Belle II computing model is presented and an overview of the distributed computing system and the offline software framework is given.

1. Introduction

The B-factory experiments, Belle at the KEKB accelerator [1] and BaBar at the PEP-II accelerator [2], have carried out a remarkable physics program and successfully confirmed the theory of Kobayashi and Maskawa [3], which explains the origin of CP-violation in the standard model of particle physics. However, this kind of matter-antimatter symmetry breaking is not sufficient to explain the dominance of matter in our universe. The task of the next generation B-factory experiment Belle II [4] is to search for new sources of CP-violation that could generate the observed asymmetry between matter and antimatter. Since the expected effects from a new kind of physics are tiny, very high precision measurements have to be performed. This requires a significant increase in the number of analyzed events. The upgraded accelerator, SuperKEKB, is designed to deliver a 40 times larger data rate than KEKB. The strategies to deal with this high rate are discussed in this article.

2. Computing Model

One of the main aspects that have to be considered for the design of the Belle II computing model is the expected data size. The planned commissioning of the SuperKEKB accelerator is in the year 2015. In about 2019 it is expected to reach its design luminosity of 8×10^{35} cm^{-2}s^{-1} so that a data sample of 50 ab^{-1} can be collected until 2021. With an estimated raw data event size of approximately 300 kB, this gives a rate of recorded events of up to 1.8 GB/s, higher than the current rate at the LHC experiments.

Another important aspect is the geographical distribution of the international collaboration. About 400 scientists from 19 different countries in Asia, Europe, America, and Australia participate in the Belle II experiment. To enable all collaborators to contribute to the computing resources needed to process and store the huge data samples, we decided to adopt a distributed computing model. More effort for setup and maintenance compared to the existing centralized Belle computing is required, but in addition to the sharing of computing resources it provides further advantages. In particular it provides redundancy. The experience of the Belle experiment after the earthquake in Japan has demonstrated how important this aspect can be.

Another advantage for the Belle II experiment is the already existing distributed computing infrastructure. In most of the Belle II member countries, Tier2 or even Tier1 grid sites are in production for the LHC experiments. Several of them already support the belle VO, which is used for the development of the Belle II distributed computing system. A few countries are also exploring cloud computing technologies.

The tasks to be addressed by the computing model are the raw data processing, the Monte-Carlo (MC) production, and the physics analysis. For the raw data processing we decided to adopt a simpler model than the LHC experiments and do all raw data processing and storage at KEK. The output in mDST format will be distributed to several grid sites. For redundancy we will copy the raw data to just one site. The raw data copy provides not only a backup copy, but also adds flexibility for reprocessings of the data.

To have sufficient statistics of simulated events, we will produce a MC sample that corresponds to 6 times the integrated luminosity of recorded data and thus has approximately a data size of 6 times that of real data mDST. As almost no input data is needed, this task is well suited for a distributed system and will be performed on grid sites and possibly cloud resources. Because of the potential vendor lock-in problem we see commercial cloud computing only as a solution for peak demands and do not intend to use it for permanent data storage. The generated MC samples will be stored on disk at grid sites.

Finally, the planning of the computing system and resources for the physics analysis is the most difficult part, because it involves an uncoordinated and random access to the data. We anticipate that users will process data and MC samples at grid sites and produce analysis specific data files (ntuples), which are then copied to local resources for an analysis with fast turn-around time. The Belle II computing model is illustrated in Fig. 1. Projections of the required CPU and disk resources are presented in Fig. 2.

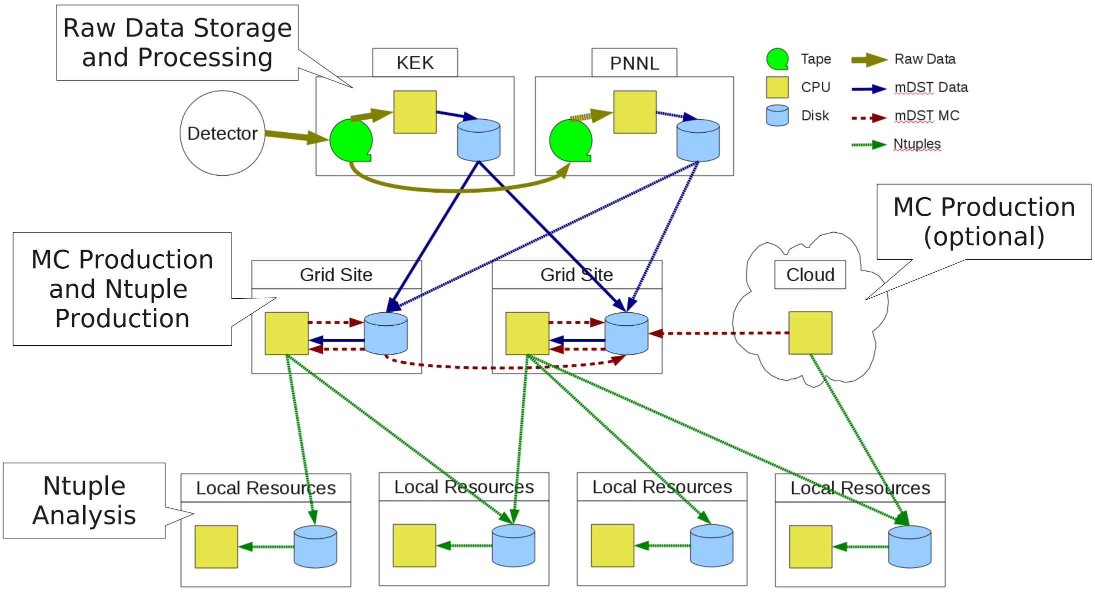

Figure 1. The Belle II computing model.

As the experience and skills of the users varies and their activities are usually not coordinated, the computing resources may not be used optimally by all users. While we want to keep the free access to all mDST data, we plan to limit the resources available to a single user and to offer a service to run his analysis code in a centrally coordinated and high-performance way.

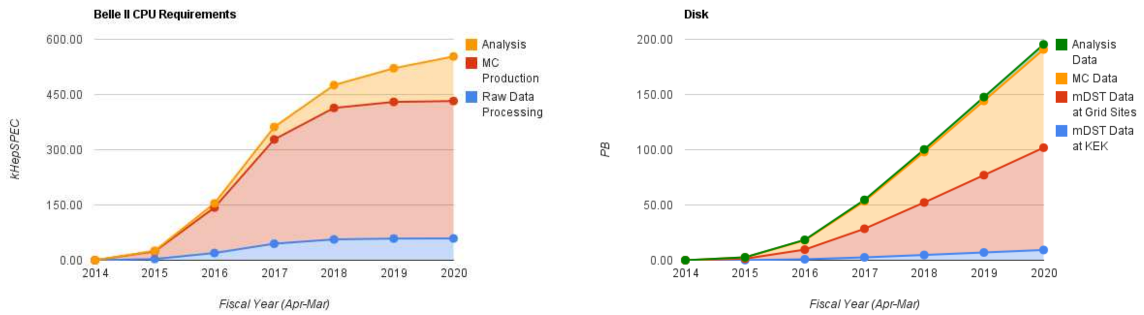

Figure 2. Disk and CPU resource requirement estimates.

The scheme foresees that users provide analysis code to group conveners who check it and pass it on to a production team that then executes the code of several people. This improves the code quality and reduces the number of data accesses. Such an organized analysis scheme was successfully applied by the Belle experiment in times when the computing resources were limited.

3. Distributed Computing System

For the implementation of our distributed computing software we rely on several existing products that provide the features we need. The basis of our system is DIRAC [5], which was developed by LHCb, but is now an independent project. One of the key features of DIRAC are the pilot jobs, a technique that is proven to increase the reliability of grid jobs. It also gives more control over the resource usage to the experiment and its modular design allows us to extend its functionality. Because the DIRAC system hides the underlying grid technology, the Belle experiment was able to use it for a MC production on grid, local cluster, and cloud resources simultaneously.

For the handling of the metadata of files and datasets we use AMGA [6]. The main AMGA server will be hosted at KEK and replications to a few grid sites are considered. For the monitoring of grid sites we started to develop a system based on HappyFace [7], a tool that is used by CMS. For the distribution of the offline software we plan to use CVMFS [8]. Solutions like FroNTier [9] are evaluated for the distributed access to detector conditions data stored in a relational or NoSQL database.

Based on these tools we develop a system that allows the user to operate on a higher level of abstraction than single files and jobs. A typical use case is that a large number of files is processed with the same code. The input files are defined by an input dataset query on the metadata of files. This allows the user to select the files that are relevant for a given physics analysis. The collection of jobs is referred to as a project. Projects can be monitored, aborted, or restarted via command-line tools or a web portal. The collection of output files defines an output dataset. Tools for dataset operations, like replication, deletion, or download, are provided. The tools are based on existing grid software, like SRM and FTS, encapsulated by DIRAC, but shield the user from details of the underlying technology.

To make it easy for users to use the distributed computing system, the changes that are required to run a job on the grid instead of a local machine are minimized. The same steering file with only a few additions, like the project name and the input dataset query, is used. The command to submit the grid job project is simply `gbasf2` instead of `basf2`.

If the user has, in addition to his custom steering file, custom code that should be executed, we foresee three options. If the code was compiled on a system that is compatible with the one at the grid sites, the compiled code can be directly included in the input sandbox. Otherwise the source code is taken and compiled automatically in a preprocessing job. The third option gives the user the possibility to automatically commit the code to a code repository from which it is taken and compiled in the preprocessing job. Because we store the information about the code version in the output dataset metadata it is possible to track exactly which code was used for a given output file.

4. Offline Software and Code Management

The offline software framework is called basf2, Belle II AnalysiS Framework [10]. It is developed from scratch, but adopts many ideas from existing frameworks, in particular from the basf framework of the Belle experiment. One of the inherited features is the parallel execution on multiple cores, a feature that will become more and more important with the increasing number of cores per CPU. Since the framework is used for simulation, reconstruction, physics analysis, and also data acquisition, code can be easily shared and transferred between these stages. The software is written in C++ to reach a high performance and to interface well with other tools used in particle physics. The scripting language Python is used for the configuration of modules executed by the framework.

Like in the case of the distributed computing system, several external software packages are employed. These include boost, ROOT [11], EvtGen [12] for the generation of particles, Geant4 [13] for the detector simulation, and GenFit [14] for the fitting of tracks.

The development of software with a team of developers distributed around the world and with different levels of expertise and background is challenging. To help the developer to deliver code that is reliable and well maintainable, we use a set of tools. One of the important tools is the continuous integration system based on the buildbot software. After a commit of code to the central repository, the software is automatically compiled on different operating systems, including Scientific Linux, Fedora and Ubuntu distributions. In case of new compiler errors or warnings the author of the code is informed via email. In a regular nightly build it is checked in addition that the code can be compiled with another compiler, that the code is documented, that the linking of libraries is correct, that the detector geometry has no overlaps, and that (unit) tests are passed successfully.

5. Summary

The next generation B-factory experiment Belle II will collect a data sample that is about 50 times larger than the one of Belle, its predecessor. To deal with this huge data volume, a distributed computing system is set up. In the development of this system we take advantage of the already existing grid technologies and infrastructures. Our aim in the development of the software for the distributed computing system and for the offline framework is to enable physicists to get results quickly without the need to dig into technical details. With these efforts we want to complement the upgrades of the accelerator and detector with improvements in the computing system and software.

References

[1] A. Abashian *et al.*, Nucl. Instrum. Meth. A **479** (2002) 117.
[2] B. Aubert *et al.*, Nucl. Instrum. Meth. A **479** (2002) 1.
[3] M. Kobayashi and T. Maskawa, Prog. Theor. Phys. **49** (1973) 652.
[4] T. Abe *et al.*, arXiv:1011.0352 [physics.ins-det].
[5] A. Casajus *et al.* [LHCb DIRAC Collaboration], J. Phys. Conf. Ser. **219** (2010) 062049; A. Tsaregorodtsev *et al.*, J. Phys. Conf. Ser. **219** (2010) 062029; R. Graciani Diaz *et al.*, J. Grid Comp. **9-1** (2011) 65.
[6] N. Santos and B. Koblitz, Nucl. Instrum. Meth. A **559** (2006) 53.
[7] V. Buge, V. Mauch, G. Quast, A. Scheurer and A. Trunov, J. Phys. Conf. Ser. **219** (2010) 062057.
[8] P. Buncic, C. Aguado Sanchez, J. Blomer, L. Franco, S. Klemer and P. Mato, PoS ACAT **08** (2008) 012.

[9] B. J. Blumenfeld *et al.* [USCMS Collaboration], J. Phys. Conf. Ser. **119** (2008) 072007.
[10] A. Moll, J. Phys. Conf. Ser. **331** (2011) 032024.
[11] R. Brun and F. Rademakers, Nucl. Instrum. Meth. A **389** (1997) 81.
[12] D. J. Lange, Nucl. Instrum. Meth. A **462** (2001) 152.
[13] S. Agostinelli *et al.* [GEANT4 Collaboration], Nucl. Instrum. Meth. A **506** (2003) 250.
[14] C. Hoppner, S. Neubert, B. Ketzer and S. Paul, Nucl. Instrum. Meth. A **620** (2010) 518.