B-factory Programme Advisory Committee
Focused Review on Software and Computing

27 – 28 June 2016 at KEK

G. Carlino (Naples), G. Cerminara (CERN), M. Elsing (CERN),
A. Formica (CEA/Saclay), F. Gaede (DESY), W. Hulsbergen (Nikhef),
M. Ishino (Tokyo), A. Krasznahorkay (CERN), P. Mato (CERN),
P. McBride (FNAL), H. Tajima (Nagoya)
and chaired by T. Nakada (EPFL)

29 October 2016

Contents

1 Executive Summary

A focused review on the offline software and computing for the Belle II experiment took place at KEK on 27th and 28th of June 2016 by a review committee that consists of three members of the Belle Programme Advisory Committee and nine experts. Reconstruction, simulation and data analysis software and database were covered on the first day and data processing and resources on the second day. This executive summary is followed by sections of full comprehensive reports.

Development and test of various individual components for simulation, event reconstruction, databases and analysis are well advancing. On the other hand, the committee feels that less attention is being paid to global software coordination and integration aspects. Overall stress tests of the whole reconstruction chain, including calibration and alignment, are missing. The reconstruction time is the major driver for the CPU needs of the experiment and further optimisation work for the software and event data model is needed.

Preparation of the framework for the Belle II distributed computing is progressing well. However, essential missing functionality needs to be implemented and the scalability of the production and data management tools up to full Belle II production levels must be demonstrated. The level of automatisation achieved in the production and data management system is also a little concern. The analysis model must cope with a huge number of events efficiently and within available resources. Although those issues are addressed individually by the various software and computing subgroups, problems are tightly connected and should be examined globally. In addition, the overall architecture of the software and computing should also be revisited in order to avoid any builtin limitation.

The committee heard little about the connection to the online subsystem, e.g. database relation and High Level Trigger (HLT). If the HLT makes event selections, the software implementing the algorithms must be of extremely high quality and practically free from errors and memory leaks. The selections must be offline reproducible in order to measure the efficiencies. Any event reduction there changes the resource requirements for offline. The HLT is not crucial for the cosmic ray run nor for the BEAST Phase 2 run. However, a clear communication line with the online project must be established now.

In general, the committee misses the overall planning of the software and computing project. A plan must be established now that clearly defines a performance goal with tasks, milestones, and available and required resources for the cosmic ray run, BEAST Phase 2 run, Phase 3 physics run. Expertise and experience of people contributing to the project should be well balanced and matched to the needs. The plan should cover not only the software related to reconstruction and analysis, but also all the infrastructure needed for data processing as well as the HLT. For this purpose, the committee recommends the Belle II management bring the core software group and subsystem software groups together through regular meetings and to consider introduction of a person having a clear mandate and authority to coordinate the effort and to monitor the progress. This would permit the optimisation of resources and enhance the ability to set priorities from a more global point of view, and enable the software and computing project to request the necessary resources from the Belle II management in a more coherent manner.

The committee was presented with a draft Memorandum of Understanding (MOU) for a share of CPU power and storage to be provided by the participating funding agencies. Furthermore, opinion was asked on the projected resource requirements until 2024. The committee fully supports that such an MOU is essential to ensure the required computing resources. The initial resource requirement for the next two to three years reflects the needs estimated based on the current status of the offline software and the current plans for the computing and analysis model. The ramping up of the machine luminosities at a startup period is always difficult to predict. The current resource requirement will need to be updated following anticipated improvements in the reconstruction software, as well as the eventual changes and refinements in the analysis model. However, the committee notes that resource requirement for the initial phase will be in any case limited. The long term evolution of the Belle II computing resource needs will follow the evolution of the integrated luminosity delivered by SuperKEKB, which has large uncertainties. One should also expect continuous improvements in the offline software and evolution of the model for distributed computing that takes into account the whole lifetime of the experiment, which are difficult to quantify now. The committee suggests to follow the LHC computing resource model and to restrict the discussion on the concrete resource requirements to a period of the coming two to three years, keeping in mind the overall view on the longer term evolution.

It is worth noting that the MOU could be extended to cover the human resources needed for the development and maintenance of software, particularly in the area of core software.

2 Software

2.1 Organisation for development and maintenance

2.1.1 Status

The committee was presented with an update of the status of the offline software project organisation and development process, to cope with the main challenges such as regional distribution, different developer backgrounds and different skills. The software process makes use of state-of-the-art tools, commonly accepted coding rules, well defined procedures and efficient communication channels.

Each major component of the software (i.e. generators, simulation, tracking, etc.) has a coordinator. Each detector has nominated a contact-person for the detector specific software. A series of general and sub-system meetings completes the organisation.

Concerning the structure of the software, a number of external software packages are provided with scripts to build the Belle II software and to setup a consistent running environment. Most of the Belle II specific code is written in C++ and Python organised in packages with one or two responsible librarians, these librarians are responsible for the detector-specific code and to ensure that is delivered in time and good quality.

The SCons tool is used to build the software. The full software process has all the elements of a modern software development activity (version control repository, automated quality checks, unit and regression tests, continuous integration, validation framework, performance monitoring, issue tracking, etc.). Shifts of one week duty are organised to monitor quality indicators and reports are given to the weekly meetings.

Main software releases are feature driven and scheduled at convenient times coherently with the overall Belle II planning. In addition, monthly builds are made available to developers to ensure that they develop against a stable and recent version of the software.

One of the main foreseen changes is the migration of the collaborative services to DESY. The main motivation is to reduce the load on Belle II collaborators in the maintenance of these services and provide continuity when the KEK computing centre is renewed.

2.1.2 Concerns

- It is foreseen to profit from the migration of the collaborative services to DESY to also migrate the code repository currently based on the SVN tool to the new GIT tool. The committee see clearly the strong motivation for this migration since GIT has many technical advantages and is clearly very well accepted among young developers. The migration implies deep changes in the development model and procedures and this may have a negative impact and some slowdown in the development of critical components.

- The roadmap for the software project were presented in a presentation during the review as well as in an accompanying roadmap document. The document list the “features” required for each major release of Belle II software with dates matching major experiment milestones. Obviously the level of detail in terms of tasks for the next releases is much higher than for the later ones. The committee thinks that this is in the good direction, but more details should be given in the roadmap, in particular breaking down some of the generic tasks affecting all detector groups (i.e. calibration procedures for all detectors). It would be nice to list all of them and see their status of completion. In addition the roadmap should state clearly if the resources to achieve each of the tasks have been identified and identify uncovered areas.

- Some parts of the offline software will need to run in the online high-level trigger. This imposes strong requirements on the quality and performance of this software (i.e. free of memory leaks, failure rates, optimisations needed to fit in the time budget, etc.) The committee is concerned with the apparent lack of interaction between the online and offline teams. Many aspects of the software and its integration in the online system will need to be agreed before providing a working high level trigger for the experiment.

2.1.3 Recommendations

- Continuous integration is essential for any large software project. The sooner one sees the effects of the new software and how it interplay with other pieces of software the better for applying corrective actions. If the results of the daily integration builds were made available to developers, they could build against them and perform basic checks before committing to the common repository reducing overall integration time.

- The committee recommends produce a more detailed project plan (roadmap) with a bit more detail and resource loaded.

- The committee recommends organise a series of dress rehearsals including alignment, calibration, etc. with clear goals at convenient times following the major experiment milestones.

2.2 Software framework and data model

2.2.1 Status

During the review, the status of the software framework (`basf2`) was presented. It is a C++ framework with a Python configuration layer. The raw and reconstructed data of the experiment is saved using ROOT I/O technology at all stages of the software processing.

The framework splits the event processing into "modules", which all perform specific tasks over data that they collect from a data store, to which they record any new data objects that they create.

The framework was migrated to use ROOT 6 and Python 3 already, which should give the experiment a relatively stable environment for the next few years.

2.2.2 Concerns

- The "data store" of the framework seems to be heavily based on ROOT technologies that are outdated by now. As an example, using `TClonesArray` is not recommended for new code since a while.

- The I/O system implemented between the "data store" and the input/output ROOT files does not seem to implement enough functionality for the data preservation needed by the experiment. During the review it was noted that relatively simple modifications in the data model were treated as major changes in the offline software, with possibly no backwards compatibility given.

- The software framework is solely based on multi-processing, and the Copy-On-Write (COW) functionality that most UNIX OSes implement. With the memory required to reconstruct events in a single process being around 2 GBs in the current software, multi-processing may not scale to the hardware environments expected around the end of the lifetime of the experiment.

2.2.3 Recommendations

- The Event Data Model (EDM) of the experiment should be thoroughly reviewed. The main emphasis should be put on minimising the size of events on disk, and creating a framework for implementing minor and major schema evolutions in the EDM during the lifetime of the experiment. The EDMs used by the LHC experiments could be a good reference point during the review. possible.

- The storage options used for the files that are produced at different stages of the data processing, should continue to be reviewed in order to ensure that the data access patterns used at the different stages of the data analysis model can be executed efficiently.

- With a lower priority, it should be investigated how the software framework will be able to support architectures that may become significant resources during the lifetime of the experiment, such as servers with hundreds of low powered CPU cores, GPGPUs, etc. Most importantly, it should be studied how multi-threading could be brought into the framework on the longer timescale to make use of these new architectures.

- Keep the option open adopting a software framework used by a larger community if it is discovered that the current framework introduces a serious drawback. C

2.3 Simulation

2.3.1 Status

The event simulation of Belle II is in good shape. Generators for all the required signal and background processes exist. Difficulties in the transition from Pythia6 to Pythia8 have been resolved. Beam background events are provided by the accelerator group and can be superimposed on generated signal events. The generation-digitisation-reconstruction-analysis chain has been exercised during dedicated campaigns in which the equivalent of atto-barns of data was processed.

2.3.2 Concerns

- Maintenance of EvtGen is currently performed by a group in LHCb. However, it seems that features of specific interest to Belle II (such as keeping the DECAY.DEC file containing all B decay up to date with the PDG) are not well-maintained.

- Belle II analyses will largely rely on efficiencies and vertex and momentum resolutions extracted from simulated events. For the foreseen production of MC events, event generation, digitisation and reconstruction are performed in one step. For HLT development, and to ensure that simulated events go through the exact same chain as real events, it would be beneficial if the event simulation could write events in the same format as the raw data. As far as the committee understands, this is not yet the case, as the encoding of the digitised event to the raw data format is not yet complete for a subset of the sub-detectors.

- Belle II projections for computing needs in the long term are driven by the MC statistics for generic hadronic background events ($\Upsilon(4S) \to b\bar{b}$, continuum $udcs$). These events are used to determine the shape of the background distributions in physics analysis. A total of four times the statistics of the data are foreseen in order to limit the contribution to the statistical error to a few percent. The committee wonders to what extend generic simulated events are actually necessary for this purpose. Peaking backgrounds from B decays could be identified with smaller samples of generic events (up to a fraction of the statistics of the data) and then simulated mode-by-mode with much larger statistics. Data driven approaches to parameterise the shape of continuum events (e.g. "sidebands") would require far less generated events and also be less sensitive to uncertainties in the generators or detector simulation.

2.3.3 Recommendations

- Agree with the EvtGen group in Warrick on a procedure for common development and validation, in particular for the support of Belle II specific features.
- Complete the digitisation/raw data encoding for all sub-detectors.
- Reconsider the use of high-statistics generic event samples and investigate alternative approaches that may allow to reduce the computing resources at no or negligible loss of physics performance.
- As Monte Carlo productions take large resources, the committee recommends that central procedures are put in place to validate new simulation releases and to monitor the data quality in the subsequent productions.

2.4 Reconstruction

2.4.1 Status

The reconstruction presentations at the review focused mainly on track reconstruction and also briefly covered TOP and ARICH reconstruction for the purpose of particle identification (PID). The committee has not heard anything about calorimeter reconstruction in the ECL and KLM. The pattern recognition is performed independently in the vertex detectors (PXD, SVD) and in the central drift chamber (CDC), followed by a dedicated track merging algorithm and the final track fitting. After filtering out hits from machine induced background based on topological isolation criteria, a global method using Legendre transformation and a local method using Cellular Automatons (CA) are applied in parallel in the CDC. Segment and track merging algorithms provide the final CDC track candidates. The tracking efficiency is found to be stable with respect to increased machine background of up to a factor of 2-3 of the expected value. Following recommendations of previous reviews, a new `RecoTrack` class has been introduced into the reconstruction EDM. The CDC tracking software has been used successfully with commissioning data with cosmic rays.

The CPU performance of the CA based stand-alone track finding in the VXD is dominated by combinatorics from background hits. An early implementation showed worse resolution compared to BaBar, whereas first results from a recent rewrite of the VXD-TF show a very high tracking efficiency, albeit still with a large fake/clone rate. The finalisation of the new pattern recognition is ongoing and foreseen to be finished before the Physics Run.

The combined track reconstruction is the dominating contribution to the total CPU usage for reconstruction jobs, apart from ROOT I/O. Two thirds of the tracking time is spent in the actual fitting, even though currently only one mass hypothesis is used. This is assumed to be in parts due to the fact that for track fitting the detailed Geant4 geometry model is used in order to take material effects into account. It was as well stated that more involved fitting techniques were used like Deterministic Annealing Filters (DAF) alongside the faster simple versions of Kalman Filtering.

2.4.2 Concerns

- The performance goals for the tracking efficiency are not clear. The CDC tracking efficiency for the CDC of 95% for tracks with a p_T of $0.3 - 0.6$ GeV/c seems rather low.

- The VXD track finding has not yet reached the performance of the BaBar tracking and even though the new development of the VXD-TF2 shows some promising performance, it is not guaranteed that it will be fully available for the Physics run start.

- The rather high CPU requirements for the track fitting might eventually cause severe problems once the luminosity starts to approach design values.

- The expected loss of experienced developers over the year might turn out to be difficult to replace and result in further delays in finalising the track reconstruction in time, both for the GenFit fitter and the pattern recognition.

  The fact that tracking is not an institutional responsibility might cause maintenance problems during the data taking period, if the tracking code needs to be adjusted to meet new, more challenging requirements in terms of efficiency and CPU performance or if it will have to be adjusted to detector deficiencies.

- The relation of the online (HLT) and offline reconstruction was not clear at the review. It is not clear that the current CPU performance of the reconstruction code meets the requirements of the HLT and if there are different requirements (working points) with respect to a trade-off between efficiency and purity.

2.4.3 Recommendations

- Complete the studies of the track finding efficiency in the CDC in order to understand, if the tracks that are not found are lost due to deficiencies of the algorithms or due to geometrical and detector resolution effects.

- In order to address the expected drain in experienced man power working on track reconstruction one needs to actively seek for new institutes or groups to join Belle II tracking and try to make the tracking reconstruction a formal responsibility of one or more institutes. In particular, it is vital to ensure the development and support for the GenFit package.

- The rather high CPU demands for the track reconstruction seem to a large extend be due to the detailed simulation model being used and the choices of track fitting techniques. The committee therefore recommends start investigating the use of faster and more modern Runge-Kutta integration codes and of a simplified reconstruction geometry as soon as time allows. It should be investigated if the use of DAF is strictly necessary and if fast track fitting techniques could be sufficient in most cases.

- In order to better understand the performance requirements and potentially needed improvements of the tracking code for the HLT, the committee proposes to organise HLT dress rehearsals with realistic assumptions on the timing and data rates for the first data taking period as soon as the status of the track reconstruction software allows.

2.5 Alignment and calibration

2.5.1 Status

Substantial effort has been deployed in the design and implementation of the Calibration and Alignment Framework (CAF), which now appears to be in a quite advanced stage of development. The framework is designed to centralise all alignment and calibration processing and to help automising it. This includes also a system for the fast calibration of conditions needed for online and fast/prompt processing.

Concerning the detector related aspects, during the review, only the alignment use case was presented in detail: the presentation addressed the status of the internal track based alignment of the vertex detector and of the first studies for the alignment of the CDC track based alignment. The alignment procedure of the silicon vertex detector is already in advanced status for what concern the setup of the millepede algorithm and framework.

2.5.2 Concerns

- The talks during the review did not present a master-plan addressing which calibrations will be needed to reconstruct the data of all the sub-detectors on the basis of the target physics performance of the reconstructed objects (both for HLT and offline reconstruction). Such a plan is essential to steer any further development of the CAF infrastructure and to establish the requirements concerning the computation infrastructure and input data of each of the calibration workflows.

- While there is substantial progress on the internal track-based alignment of the vertex detector and of the CDC, a global alignment strategy, including the plans to integrate the survey measurements and to determine absolute and relative positions of each sub-detector with respect to the others as well as to the magnetic field, has not yet been addressed.

- In the context of the alignment of the silicon vertex detector, the presentation did not address the requirement in terms of input data necessary to achieve the needed accuracy, both in terms of statistics and in terms of different event topologies. The potential deformations of the system that could be reabsorbed by the track parameters are difficult to be constrained in the millepede procedure. (i.e. the so called weak modes of the alignment procedure) have not been covered in details.

- In the current implementation the alignment of the silicon vertex detector does not parametrise the geometry according to the mechanical structure of the detector:

each module is aligned independently. A hierarchical parametrisation matched to mechanical construction structure presents several advantages, among them:

  - allows to constraint the most poorly measured coordinates according for example to survey measurements;
  - allows to determine the alignment parameters with a lower granularity w.r.t to the module level when the statistics of the input dataset is not sufficient or a faster turnaround is necessary;
  - allows for the correction of time dependent coherent movements of the support structures.

  The last point might be particularly important to exploit the 0T data together with tracks collected at full field, taking into account possible coherent movements of the detector structures at each magnet cycle.

- In the current implementation, the track-based alignment of the vertex detector uses the "General Broken Lines" (GBL) method for the track model while other track models are used during the track reconstruction. This usage of two different models may lead to a set of alignment constants that are not optimal

- The strategy for the monitoring and validation of the performance of the calibration quantities computed with the common framework have not been presented. This is of crucial importance for all the workflows which are expected to produce calibrations for HLT and prompt reconstruction, especially when fast turnaround time is required.

2.5.3 Recommendations

- The calibration plans of each sub-detector group and reconstruction algorithm should be collected, and the requirements in terms of target accuracy should be evaluated on the basis of the physics goals of the experiment. On this basis, the needs in terms of input data selection, frequency of the computation of the calibrations, computing resources and database storage should be assessed to steer and prioritise any further development of the CAF infrastructure.

- A global alignment strategy needs to be determined beyond the internal track based alignment of each sub-detector. This should include an estimate of the needed accuracy in the knowledge of the position of each sub-detector (including the calorimeters) based on the target performance of the physics object and possibly evaluating the need to assign position errors related to the alignment. Moreover, this plan should include a strategy for the inclusion of the constraints derived by the survey measurement during the installation, and for a track based measurement of the relative position of the various components.

- The performance studies of the millepede-based alignment of the vertex detector should be be extended to cover both the random misplacement of the modules and the potential coherent deformation which could go undetected using only certain topologies of tracks. These studies require a careful evaluation of what are the possible movements according to the mechanics of the support structures of the modules. This would allow to establish the composition of the input dataset and the possible constraints that can be exploited to achieve the needed performance.

- A hierarchical parametrisation of the alignment geometry of the tracker should be implemented on the basis of the mechanical construction structure. The alignment algorithm should be exercised with different levels of granularity, from module level to larger structures.
- The choice of the GBL track fitter for the Millipede-based alignment of the vertex detector should be supported by dedicated studies to demonstrate that the differences between the GBL implementation and GenFit does not introduce any bias when a different fitting technique is used for the track reconstruction. Alternatively, it should be considered to use the Kalman filter inside the alignment algorithm
- The design of the common calibration framework and of the calibration workflows, in particular for those which are considered critical for the performance of the HLT and prompt reconstruction, should include a system for the verification and validation of the newly computed calibration constants possibly based on the DQM framework of the experiment. This evaluation should be performed on the relevant data: depending on the needed latency and turnaround time for the feedback, dedicated data streams might be necessary.

2.6 Database

2.6.1 Status

The architecture of the conditions database in Belle II is structured as a standard multi-tier model, with an intermediate application server sitting in between the client and the database components: the middle tier implements all the relevant functionalities for database management, and expose them to clients via a Representational State Transfer (REST) Application Programming Interface (API).

The implementation uses standard java technologies both for the web controllers layer (JAX-RS) and for persistence (JPA), and is deployed in a Payara cluster of servers providing High Availability (HA) and load balancing features. On top of this an Apache web server redirects the requests to Payara cluster (caching can be added if needed). The REST API uses standard HTTP methods to implement management operations (GET,POST and DELETE), following the principles of a pure REST architecture.

The backend is a PostgreSQL server (but this could be moved to MySQL in the future, if the adoption of the Openstack/Trove solution to provide a DBaaS platform presents limitation with Postgres in term of database replication, TBC). Also for the database server the idea is to deploy HA features.

The client is a C++ application, but can easily be ported to other languages if needed, since the REST API provides data in JSON or XML and the network layer is HTTP.

In terms of data model, the database layer contains the relevant tables to handle all the metadata related to conditions, in particular Global Tags and Interval of Validities (IOVs), while the conditions data themselves (called payloads) are stored under a normal file system, and can be accessed via Apache (using the REST API).

The global application architecture and implementation have proved a flexible design since they could easily migrate to Payara application server (the first version was deployed under Glassfish), and migration from Postgres to MySQL should be transparent as well. Even though the deployment of an HA database cluster is not yet realised, the system as it is today can probably be used for the first steps of data taking that Belle II will perform starting from the end of 2016, after some stress tests validation. Its usage at production level will help in determining which operational issues are still to be addressed for the final implementation.

2.6.2 Concerns

The committee expresses in the following some general concerns that are mainly derived from the experience in operations at the LHC experiments. For the moment certain elements like the number of payload types in Belle II, their volume and also the update frequencies are not well known. These parameters are important for the final design and should be investigated by the experts.

- IOV/Payload update policy: IOV/Payload sequences are frozen in the Global Tag once it has been labeled as published, but it is not clear how to support use cases where the IOV list needs to evolve (e.g express and prompt reconstruction).

- Creation of conditions database snapshots for online usage by the HLT seems to introduce overhead in the operation of the Global Tag. More in general, the production of a new Global Tag every time that new conditions data need to be taken into account by data processing may result in a large number of Global Tags that could be difficult to manage.

- Global Tag containing directly the full set of IOVs for every module (Payload type): in the present data model implementation the Global Tag coordinator is basically directly responsible for all updates coming from every system. The introduction of an intermediate hierarchical layer between the Global Tag and the IOVs could help in the administration, depending on how many different systems are going to store conditions for the new Global Tag.

- Authentication and Authorisation: security aspects in server access are for the moment not presented, which is normal in this development phase. Nevertheless, in an environment in which many users may update the conditions database it could be useful to provide some basic authorisation capabilities.

- Monitoring tools at the level of database servers have been explored by the developers. What can also be useful is to study a monitoring plugin for the middle-tier itself and for the bookkeeping of the conditions usage both during the creation step but mainly at the level of the job consuming conditions. Keeping track in the data of the payloads actually consumed by the client application might be an asset when debugging/reproducing the output of production jobs.

- It is better to think about the preservation of conditions data in the early phase of the development. The choice of open-source databases is already a good one in this direction.

- Payload storage and distribution: the usage of a file system for storing the payload can be seen as a simplification but presents some drawbacks; it is not clear how the database and the file system can be kept synchronised in time and in all relevant data flows (in particular for online usage).

2.6.3 Recommendations

- Review the use cases for Global Tag usage in different production data flows (online/offline) and define accordingly a policy for the insertion/update of IOVs and payloads which guarantees reproducibility of the results and minimise the operations on Global Tags and IOVs. Consequently, review the need for different lifetimes of the caches depending on the expected latency (e.g. online, express caches need to refresh the IOV sequences more frequently than offline ones). For workflows which need to consume the most recent calibrations, it might be useful to implement the capability to force the flushing of the caches.

- evaluate the possibility and possible benefits of introducing a further hierarchical layer in the database schema to map IOV sequences belonging to the same calibration/object within a given Global Tag. Among the possible advantages:
  - ease the maintenance and operation of the Global Tags by the Global Tag coordinator, delegating the assembly of detector/calibration specific IOV sequence to the corresponding expert;
  - allow to test only a particular calibration on the top of the existing Global Tag;
  - simplify the evolution of the Global Tags allowing for the modification of only a specific calibration

- Study simple solutions for authentication and authorisation to protect against misuse of the REST API itself when inserting data inside the conditions database.

- Payload storage and distribution: explore and characterise different storage solutions (e.g CVMFS) evaluating all the possible use cases in terms of latency of the updates and needs for distributed access.

- Payload schema evolution: verify that all elements are present in the data model to guarantee the possibility to perform transparent schema evolution of the payload data types, or to enforce the schema safety in case its evolution is not provided.

2.7 Analysis tools

2.7.1 Status

During the review a nice overview was shown for the analysis tools currently available in the basf2 framework. A lot of effort was made to provide ready-to-use analysis modules for the framework that just need an appropriate configuration to run most of the analyses currently foreseen for the experiment.

A fair number of external packages were also incorporated into the Belle II framework by now, implementing different ways of vertex and decay fits, flavour tagging, and B reconstruction.

A good emphasis seems to be put on documenting these tools, and organising tutorials to teach the collaboration about their usage. An effort that will have to be kept up with a high priority, especially in the first years of data taking.

2.7.2 Concerns

- Many of the centrally provided analysis modules, by construction, seem fairly "opaque". A large emphasis was put on providing as minimal user interface to these tools as possible. This may make it hard, especially for newcomers to the experiment, to understand how the analysis code works.

- There was some discussion during the review on how physicists can write their own analysis module for the offline framework, or modify an existing module, and submit jobs to the grid with their module included. The way to do this seemed to require some amount of expert steps at the moment.

- The development plan of the analysis tools is tightly coupled only to that of offline framework at the moment, while during the running of an experiment the development of reconstruction and analysis code usually happens on a different schedule.

- The analysis model provides tools for the physicists up to the point of making small, final n-tuples with the data that they need. There was no mention however about providing help for the physicists with the final steps of their analysis.

2.7.3 Recommendations

- The Particle and ParticleList classes, should be reviewed as a part of the event data model. They should be optimised for ROOT to be able to compress them as much as possible, and that the μDST files written with these classes included, have optimal settings for analysis usage.

- Tutorials on the analysis software should include parts explaining how new analysis modules can be implemented from scratch, and how existing modules can be tweaked during an analysis. As these tutorials should serve as a main entry point for most of the collaboration into the software area, they need to be designed to pick up new developers from the collaboration as much as possible.

- The tutorials should also consider to include a description of the final steps of the analysis, giving users recommendations for efficiently reading the n-tuples that they made using the analysis tools, and providing the information from them to different statistical tools. As inefficient usage of these files can add up to a significant CPU and I/O time, because of the high number of times that they need to run.

- It should be investigated, as part of the analysis model, what release strategy should be followed with the analysis code during data taking, allowing to implement more frequent fixes to analysis code than what the reconstruction code's development plan allows for.

- The development of new analysis code for the offline software, and its testing using grid jobs should be streamlined, making it easy for users to make modifications on top of an existing release, and try their modification on different scales before requesting its inclusion into the software release.

3 Computing

3.1 Organisation

3.1.1 Status

The computing project for Belle II is organised separately from the software project. Both projects are progressing in parallel with a strong coordination between them. The objective of the computing project is to put in place the computing system and associated services to be able to process and analysis the data collected by the Belle II experiment, while the objectives of the software project is to develop the applications that will run on the computing system. The chosen computing model is based on a distributed approach using grid and cloud technologies to access a heterogeneous system of computing resources hosted in centres of different sizes and types. The overall distributed computing system is getting more concrete. It is described in terms of a data flow diagram indicating the storage points distributed between Japan, Europe and US and the required bandwidth between them, as well as the data processing steps foreseen to be done in each location. The scale of the computing system and the worldwide distributed nature implies a rather sophisticated organisational structure involving people from the different centres in the organisation. The identified subgroups are: the overall distributed computing architecture, the network architecture including data transfer tests, the data processing services, and the training activity. The last one is very important for physicists using the computing system to perform their distributed analysis.

People have been appointed to coordinate the different activities, but some holes still need to be covered. Meanwhile some of these roles are covered in an interim fashion.

The Computing Steering Group has been put in place to monitor and do the accounting for the computing resources, as well as to collect and justify the future projections of the computing needs for Belle II. The steering group is chaired by a different person than the computing coordination, which is focusing on developing the infrastructure and services. This has been a positive change with respect to the situation in the past.

3.1.2 Concerns

- The main concern is the missing manpower to develop all the needed services. Although this need has been repeatedly indicated to the committee, not much has been made, besides changing priorities such that essential things are covered and less essential are delayed. Building the computing system for Belle II is a complex endeavour and requires a range of competences. The strategy that has been put in place is to re-use the existing components and services developed mainly by the LHC collaborations. This is a very good strategy, but still dedicated Belle II resources will still be required to adapt the general solutions to specific case.

- In order to guarantee a sustainable collaboration with some of the providers of services and middle-ware software a collaboration agreement with WLCG has been pursued. As far as the committee has been informed, the actual agreement has not yet been concluded. This is important for formalising the support terms on some of the components that will become critical to Belle II.

3.1.3 Recommendations

- To cope with the serious lack of human resources, it is recommended to re-evaluate what services are really essential to the operation of the Belle II computing. In general, not all the distributed computing services in place for the large LHC experiments are absolutely needed for an experiment of the size and complexity of Belle II. Therefore, the development of a process to evaluate all of them, and eventually re-assign resources to the absolutely needed ones, is highly recommended.

- Campaigns to produce simulated data are an excellent way to put the teams working together in delivering the data required for various studies and the development of the software. The committee feels that these campaigns could also be used for the integration and overall stress tests of the whole reconstruction chain, including calibration and alignment, as well as stressing and testing the usability aspects of the computing services already in place.

3.2 Database and data management

3.2.1 Status

The distributed database system is based on the conditions database infrastructure. The central infrastructure is deployed at PNNL, providing HA and load balancing capabilities both for client HTTP access as well as for database access from the intermediate conditions data server. Several layers of caching are foreseen in the architecture exposed for the conditions database, and the conditions access from a distributed computing environment (basically access from grid jobs) is being investigated. In order to further increase the capacity to handle several data flows, it is foreseen to have alternative clones of the conditions database infrastructure deployed elsewhere as sort of slaves for serving conditions more efficiently. The way this can be implemented is still to be defined. The design of the conditions database infrastructure seems well suited in general for a distributed computing environment and access via grid jobs.

3.2.2 Concerns

- The present schema for distributed computing in the database area is lacking a clear view about how the conditions data are used at the level of the prompt and express reconstruction data flows, mainly in the case where conditions need to be updated from time to time.

- The conditions data access (both in read and write mode) is in general not clear enough, probably because of lack of interactions among database experts and the detector people involved in alignment and calibration data flows, as well as the HLT community. It is important for the design and tests of the distributed databases that detector community helps in the definition of the data volumes, update frequencies and that the necessary information about use cases is gathered in advance, to help the developers in designing of the system.

- The idea of replicating the conditions database infrastructure to ease the distribution of conditions needs to be carefully studied, once the informations mentioned in the previous item have been gathered. The choice of the technologies here is very important, and the collaboration has to evaluate carefully the manpower available on the sites where conditions database clones could be installed: e.g., no database administrators are available at KEK computing centre, so all the burden of database administration would be in the hand of Belle II database experts. It seems then natural to use technologies where data replication and synchronisation from master to slaves is well supported.

3.2.3 Recommendations

- Gather relevant use cases in alignment and calibration data flows: the data volumes and update frequencies are important to dimension the system.

- Continue to test the infrastructure in the distributed computing environment.

- Profit from standard tools (from open source community) to simulate heavy loads and different access configurations to the conditions database system, in order to test both the http loads and the database access without hitting the caches (to spot the real limits of the system). Tools available for this purposes are Apache Jmeter and Gatling.

- Evaluate the need for authentication and authorisation capabilities in the infrastructure proposed (mainly for the write mode).

- Monitor the system not only at database level, but also at the level of the access to Payara servers and foresee monitoring at the level of reconstruction job themselves to have a clear view of which conditions have been loaded and used.

3.3 Computing model for production and analysis

3.3.1 Status

Belle II plans to use an analysis model inspired by former and current HEP experiments. In the current model the experiment will centrally produce the ROOT-formatted RAW files and the reconstructed DST (with sampling) and mini-DST (mDST).

As the full mDST datasets will become too large for physicists to process individually, Skims (datasets after event selections) will also be produced centrally. The users will be expected to run their individual analyses on these skimmed datasets to produce n-tuple datasets that would be small enough to be downloaded to local resources for the final stages of the analysis.

Skimming with multiple scenarios is investigated at the moment.

1. Producing skimmed mDST datasets with exactly the same payload of low level reconstructed objects (tracks, clusters, etc.) that are in the full mDSTs. In order to analyse these, the users need to re-run the high-level object reconstruction using the analysis tools provided by the experiment and make their analysis on these dynamically created objects.

2. Producing skimmed mDST datasets, with high level reconstructed objects added to the files on top of the low level reconstructed objects coming from the input files. Those datasets are called μDST. Analysing these files can be done by simply making use of the high-level objects found in the files.

3. Producing “index files” that have references to individual mDST events, and no other payload. To analyse these index files, one has to run a job that can access the original mDST events using the information stored in the index, reconstruct the high-level objects out of the mDST payload, and provide these to the user for analysis.

The setup of the organisational structure for producing these central Skims is progressing well. A dedicated group, in cooperation with the Physics Coordinator, is taking care of defining event selection criteria for the various datasets. Skim production will require an automated fabrication system, which would probably evolve from the DIRAC automatic system

3.3.2 Concerns

- The idea of "index files" unfortunately failed in the LHC experiments so far. In order for these to make sense, one has to be able to analyse the full dataset using the index files efficiently, while only reading/streaming a small part of the original full dataset. For this to happen, event selection criteria for generating the index files must reject most of the events so that not all the input files must be read in the production, if the production happens by first copying the input files to the worker nodes. With an average event size of $O(\mathrm{kB})$ and an input file size of $O(\mathrm{GB})$ (for efficient grid storage), an average number of events stored in one file is $O(10^6)$. Therefore. an event rejection of better than 10^6 is needed. If the production system accesses files directly (through xrootd for instance), the dataset for the selected events should not exceed the flush size of the mDST ROOT files. With an average event size of $O(\mathrm{kB})$, and an ideal flush size of $O(\mathrm{MB})$, an event rejection of better than 10^3 is needed. The performance numbers of the event rejection shown in the review (and extended information later available on Indico) confirms the worry that the event rejection rates are too low.

- The dataset handling in the Belle II production system does not allow:

  - defining container datasets over individual datasets;
  - a single file in DDM to belong to multiple datasets.

  This means that in order to run on real data, physicists have to declare selection rules for the metadata of the datasets that they want to run on. This has some worrisome implications in terms of reproducibility, such as a "too loosely" defined selection picking up a different set of datasets when run at two different times and making it hard to document what was done exactly for the individual analyses. This concerns only for the real data and for simulation datasets such a complexity is usually not required.

- The committee did not hear much about the technicalities of working with "index files" during code development. A too high barrier on setting this up (accessing the file catalogues, having the right certificates for the file access, etc.) can have a large effect on the number of people who could contribute to the software development.

- Concrete goals for the properties of the Skims in terms of selection rates, (relative) file size and CPU usage, have not been shown. In order to produce reasonable models for the resource estimates, these resource goals are absolutely necessary. This is also necessary to estimate the "turn around time" of the Skims when some new calibration is introduced in the conditions database, for example.

- The committee did not hear about any special consideration given to Skims produced from datasets for simulated events, which usually adds complexity in organising their productions.

- The central production of mDST files was not extensively discussed during the review. It is not clear how much planning is made at the moment to set up the operational structure for ensuring that data collected by the detector is calibrated, and then reconstructed according to a fixed schedule.

3.3.3 Recommendations

- Beside the currently investigated skim file formats, a format that physically holds the high-level reconstructed objects created during the skimming and links back to the original mDST events should be investigated. Through those links, the low-level reconstructed objects could be accessed when necessary.

- All of these models need to be carefully evaluated in terms of viability according to the criteria described in the first concern. It has to be demonstrated that a system making use of indices can be used efficiently in the Belle II computing model.

- A target model needs to be established for the central production of the Skims by considering

  - Target CPU time per event in which Skims could be produced from the full mDST datasets for real and simulated data.
  - Target size for the individual skimmed datasets and their total size with respect to the full mDST dataset.
  - A “lifetime model” for the Skims by establishing how many different versions of skimmed datasets could exist at once and for how long.

- A target turnaround time needs to be established for users running their analysis on skimmed datasets. The performance of the n-tuple making analysis code has to be checked for this goal.

- The missing components of the computing model need to be worked out, including the model for the production of analysis Skims and for the lifetime and replication policies of the different data formats.

- An operational body should be created (if it does not exist yet) that will manage the prompt reconstruction of the data taken by Belle II, and will oversee the reprocessings of this data. This body would need to decide together with physics and computing coordinators about the short- and long term plans of data processing in the experiment.

3.4 Production system

3.4.1 Status

The Belle II production system is built on top of the DIRAC as suggested by the committee previously and consists of Production Management, Fabrication, Distributed Data Management (DDM) and Monitoring. Production management defines and manages the properties such as software release, event type and the volume of the production and lets Fabrication to deploy and manage (replace failed jobs and verify outputs) individual production jobs. The DDM gathers the output files and distribute final products over the grid and Monitoring allows shifters to monitor progress of individual jobs.

Fabrication is implemented as a plugin extension of DIRAC Transformation System and takes advantage of DIRAC Workload Management System. So far, a prototype of the Fabrication system has been implemented and used to produce data where a total of 10 million jobs were required and 20,000 jobs were constantly processed. Job submission rate was about 1 Hz, limited by DIRAC performance. Bulk submission capability is expected in DIRAC, which will improve this performance. Database access is not yet fast enough, but will be improved through further optimisation of the DB schema. Communication with DDM has been implemented and is being tested in the current production exercise.

DDM takes care of storage element accounting, data transfer, bulk data deletion and data integrity assurance. The data transfer and bulk data deletion components replace DIRAC functionality for optimised performance. Currently, core services of DDM are available and being tested in the current production exercise. So far, it handled more than 250k file transfers. Tests of DDM for raw data handling are also in progress with cosmic ray tests used to verify raw data transfers from KEK to PNNL. These tests provide an opportunity for DAQ and Distributed Computing groups to work together. The same tests using LHC Open Network Environment (LHCONE), the dedicated LHC network, will be performed in autumn.

The monitoring system monitors progress of various components for effective resource usage and restarting failed jobs, and gives visual displays of those progresses and notices to shifters and/or experts. It also finds common causes when large number of jobs fails.

An initial test of full Production Management System with Monitoring system will be performed in the next production exercise planned in November 2016. In addition to the above tests with productions of simulated events, scalability tests and tests of raw data handling and reconstruction with cosmic ray data are planned in 2017.

3.4.2 Concerns

- The committee notes that the design, the implementation and the use in simulation campaigns of the Production System is greatly improved with respect to last meetings. However, the performance requirements, in particular scalability, are not yet well defined.

- There are still many missing components that are not implemented in the Production Management and Monitoring systems. The Fabrication system is only partially implemented at this time.

3.4.3 Recommendations

- Performance requirements for the Production System need to be defined and the overall performance must be demonstrated before the physics run. In particular, the scalability of the system should be tested with great accuracy in the foreseen stress tests.
- The missing components need to be implemented, while leveraging the DIRAC basic functionality as much as possible, and their functionality and performance should be verified in realistic environments. In particular, priority should be given for the development of missing production system components to fully automatise the system, including an user-interface for production and analysis Skims task definition.
- A fully functional GRID data management system needs to be developed, probably, building on the DIRAC file catalog and transfer functionality already in use as two of its central components. It should be investigated if LHC solutions can be reused.

3.5 Infrastructure and resource requirement

3.5.1 Status

The Belle II computing model is based on a hierarchical infrastructure composed of four layers of data centres with different functionalities: the *Raw Data Centres* with tape capabilities and high network bandwidth where raw data are stored and processed, the *Regional Data Centres* where copies of the reconstructed data (mDST) are stored and Monte Carlo production is performed, the *MC Production Sites* dedicated to the Monte Carlo production and physics analysis and the *local sites* where the last step of the user analysis is performed.

The use of the computing and storage resources of heterogeneous nature and distributed over the different layers of data centres in the four continents, is orchestrated by the DIRAC framework that has the role of both workload and data management system. The collaboration has presented to the committee the status of the development of all the necessary components of the computing infrastructure and the committee is pleased to note that the level of maturity is increasing very fast.

The role of the network is extremely important in such a distributed environment. Belle II US and EU sites have joined LHCONE, KEK will join after summer 2016. Moreover, the Japanese national research and education network (SINET) is planning to upgrade the US West Coast - Japan line to 100 Gbps and considering to have a direct line to Europe exploring the possibility of reaching the same 100 Gpbs bandwidth in future. During the February BPAC meeting, the results of the data transfer rehearsal carried out measuring the throughput among sites and comparing it with the 2019 and 2024 network bandwidth requirements have been presented. The requirements have been mostly achieved with the current available bandwidth, so the committee can conclude that network does not seem to be an issue for the experiment.

The collaboration has presented to the committee the estimation of the computing and storage needs from 2017 to 2024 already shown at February BPAC meeting with addiction of a preliminary estimation of the needs for the skimming procedure. This estimation is based on four ingredients: the number of events to be simulated or collected and the subsequent processings, the input parameters such as CPU processing times and event sizes, the data replication model and the analysis model.

Considering only the years 2016-2018 for which the committee has a detailed breakdown of the resource needs, the activities foreseen are: signal and generic events simulation, Phase 1 and 2 real and simulated data processing and cosmic and, later in 2018, physics data processing and analysis. Among these activities, the simulation and reconstruction of 5ab^{-1} of generic type of events is the one requiring the largest amount of resources. The effect of the stream multiplicity on the estimated statistical uncertainties as a function of the integrated luminosity is shown for a set of golden mode analyses. Having only two streams rather than five causes a degradation of the error not larger than about 10%.

The resources needed for these activities are determined by the value of the CPU processing times for simulation, reconstruction and analysis and the event sizes of the various data format. These parameters come from measurement with the current codes adding safety factors to take into account uncertainties and further improvements of the code.

The data replication model requires that RAW data will be permanently stored on tape in two copies. In the first years of data taking, the copies will only be at KEK and PNNL. Later other Raw Data Centres will be integrated into the system with shares custodial responsibility depending on the PhD count. mDST data will be replicated in three copies, one copy per continent, in order to avoid bottleneck in the analysis.

The analysis model is based on a process of data skimming for the selection of subset of events. It is planned to be a coordinated activity with group official Skims organised on the train model, where necessary technology is still under development. Two options have been considered for the production of skimmed files with a copy of the events plus additional quantities, and index files, i.e. a collection of pointers to the selected events. The choice of one of the two technologies will have a strong impact on the storage needs. A preliminary estimation of the computing and storage resources necessary for the skimming has been shown and added to the overall amount of requirements.

3.5.2 Concerns

- The committee notes that there are some uncertainties in the resource requirement even for the short term caused by a very conservative estimation applied for the current plan for the computing and analysis model.

- The safety factors of the computing model parameters used as input to the resource estimations seem to be too large (∼50% or ∼100%) and not completely justified because they do not take into account the expected improvements of the offline code and of the event data model. This makes sense only for the short term estimates.

- Not all the activities reported in the tables in the Appendices A, B and C of the "Belle II Computing Resources Estimates for 2016-2024" document have been clearly described and it is quite difficult to assess the real resource needs.
- Estimation of the resources needed for the skimming activity is very preliminary regardless of the scenario that will be chosen and in particular the CPU requirement seems to be large when compared to activities with larger CPU demands such as production of generic simulation events.

3.5.3 Recommendations

- The committee finds that it is not possible to give a quantitative evaluation of the long term computing and storage needs because of the large uncertainties still present and recommends the collaboration provide a revised version of the resource requirements restricting the discussion to a period of the coming two to three years in line with the LHC model for computing resource requirements.
- The committee recommends the use of more realistic CPU processing times in the computing estimates in order to avoid large safety factors. Clear details of the computing resources are needed and should include all planned activities and simulation needs.
- The committee recommends the development of a report which compares the available and used resources at all computing centres. This report should be made available to be evaluated alongside new computing resource requests.