

## The simulation library of the Belle II software system

**D Y Kim<sup>1,17</sup>, M Ritter<sup>2</sup>, T Bilka<sup>3</sup>, A Bobrov<sup>4,5</sup>, G Casarosa<sup>6</sup>, K Chilikin<sup>7</sup>, T Ferber<sup>8</sup>, R Godang<sup>9</sup>, I Jaegle<sup>10</sup>, J Kandra<sup>3</sup>, P Kodys<sup>3</sup>, T Kuhr<sup>2</sup>, P Kvasnicka<sup>3</sup>, H Nakayama<sup>11</sup>, L Piilonen<sup>12</sup>, C Pulvermacher<sup>11</sup>, L Santelj<sup>11</sup>, B Schwenker<sup>13</sup>, A Sibidanov<sup>14</sup>, Y Soloviev<sup>15</sup>, M Starič<sup>16</sup> and T Uglov<sup>7</sup>**

<sup>1</sup>Soongsil University, Seoul 06978, Republic of Korea

<sup>2</sup>Ludwig-Maximilians-Universität München, D-85748 Garching, Germany

<sup>3</sup>Charles University, 121 16 Prague, Czech Republic

<sup>4</sup>Budker Institute of Nuclear Physics SB RAS, Novosibirsk 630090, Russian Federation

<sup>5</sup>Novosibirsk State University, Novosibirsk 630090, Russian Federation

<sup>6</sup>Johannes Gutenberg University of Mainz, D-55128 Mainz, Germany

<sup>7</sup>P. N. Lebedev Physical Institute of the Russian Academy of Sciences, Moscow 119991, Russia

<sup>8</sup>University of British Columbia, Vancouver BC V6T 1Z1, Canada

<sup>9</sup>University of South Alabama, AL 36688, USA

<sup>10</sup>University of Florida, Gainesville, FL 32611-8440, USA

<sup>11</sup>High Energy Accelerator Research Organization (KEK), Tsukuba 305-0801, Japan

<sup>12</sup>Virginia Polytechnic Institute and State University, Blacksburg, VA 24061, USA

<sup>13</sup>University of Göttingen, Faculty of Physics, D-37077 Göttingen, Germany

<sup>14</sup>University of Victoria, Victoria, BC V8W 3P6, Canada

<sup>15</sup>Deutsches Elektronen-Synchrotron, 22607 Hamburg, Germany

<sup>16</sup>J. Stefan Institute, 1000 Ljubljana, Slovenia

<sup>17</sup>Corresponding author

E-mail: dorisykim@ssu.ac.kr

**Abstract.** SuperKEKB, the next generation B factory, has been constructed in Japan as an upgrade of KEKB. This brand new e<sup>+</sup> e<sup>-</sup> collider is expected to deliver a very large data set for the Belle II experiment, which will be 50 times larger than the previous Belle sample. Both the triggered physics event rate and the background event rate will be increased by at least 10 times than the previous ones, and will create a challenging data taking environment for the Belle II detector. The software system of the Belle II experiment is designed to execute this ambitious plan. A full detector simulation library, which is a part of the Belle II software system, is created based on Geant4 and has been tested thoroughly. Recently the library has been upgraded with Geant4 version 10.1. The library is behaving as expected and it is utilized actively in producing Monte Carlo data sets for various studies. In this paper, we will explain the structure of the simulation library and the various interfaces to other packages including geometry and beam background simulation.

## 1. The Belle II detector

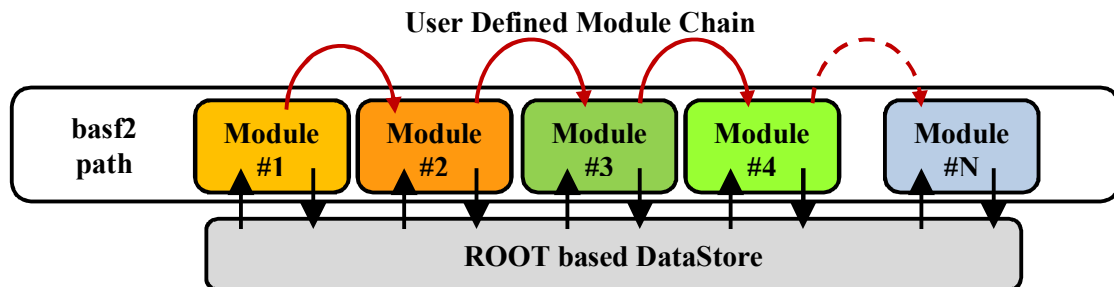
The SuperKEKB collider, which has been commissioned in 2016, provides a 7 GeV  $e^-$  beam and a 4 GeV  $e^+$  beam for the Belle II detector, which is being upgraded from the Belle detector. The overall size of the Belle II detector is about 7.5 m long and 7 m high [1].

At the interaction point, the diameter of the Beryllium beam pipe (IR) was reduced to 2 cm. A new vertex detector consisting of a 2-layer pixel detector (PXD) and a 4-layer of double-sided silicon detector (SVD) is being constructed. The central drift chamber (CDC) has been upgraded with small cells for the inner part, longer lever arm and fast electronics. These detectors are used to discover tracks of charged particles [1].

Just outside of the drift chamber, a time-of-propagation counter (TOP) is constructed for the barrel region and an aerogel ring Cerenkov detector (ARICH) is built for the forward endcap region. The original electromagnetic calorimeter (ECL) of the Belle detector has been refitted with a waveform sampling readout system and placed outside of these counters. The CsI(Tl) crystals from the Belle calorimeter are reused. There is a plan to upgrade the endcap regions of the calorimeter with pure CsI crystals. The outermost component (KLM) of the Belle II detector is used for the identification of  $K_L$  and muon particles. Originally constructed with resistive plate counters, the inner two barrel layers and the components for the endcap regions are replaced with scintillators [1].

## 2. The basic structure of the Belle II software system

The Belle II software system is a framework named *basf2* [2], capable of parallel processing [3]. The core part of the system is written in C++ and the job scripts are described by Python [4]. Each event is executed on a path, i.e., a chain of modules for specific tasks selected by users. The task itself could be as simple as printing out statistics or could be a complicated one such as reconstruction of an event. I/O for data objects are handled by ROOT [5] and the persistent data storage (*DataStore*) used by *basf2* is in the ROOT format. Many ideas from the original Belle software system [6], the ILC framework, and other experiments are incorporated to construct *basf2*. The simulation package is based on the Geant4 software [7]. Figure 1 shows a typical *basf2* execution in the framework aspect. Figure 2 is an example of a *basf2* job script.



**Figure 1.** An example of a *basf2* job consists of user selected modules. Event data objects are kept in memory during the processing of the event (*DataStore*), and all or parts of these objects can be written to a persistent storage in the ROOT format. This is a variation of a version shown in [2].

## 3. 3-D magnetic field in *basf2*

The main component of the magnetic field inside the Belle II detector is a 1.5 T solenoid. Outside of it is an iron yoke, intertwined with the outermost sub-detector, KLM. In addition, the beam-focusing system of the SuperKEKB collider, mostly quadrupole magnets, is modifying the field value. The yokes and shields around the quadrupole magnets create non-linearity in the magnetic field [1].

To exploit the Belle II detector to its full potential, an understanding of the magnetic field with a precision of 0.1% is necessary. Recently, a dedicated study of the magnetic field has been conducted with a simulation model based on the OPERA3D/Tosca software [8]. In-situ measurements of the magnetic field were executed in September 2015. The results from these studies are utilized to create a 3-D magnetic field map, which is implemented in basf2 [9]. More simulation studies and in-situ measurements are planned to improve the precision of the magnetic field map.

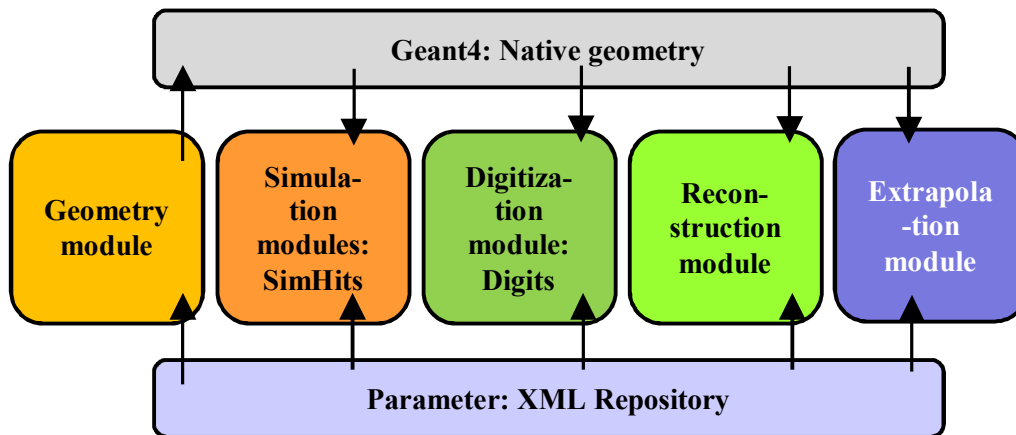
#### 4. Handling of geometry

For the offline software library, the geometry of the Belle II detector is implemented by C++ codes and created during the Geant4 simulation in a basf2 path [1]. The input parameters for the geometry are stored in an XML repository file. We are developing a database scheme to replace the repository file. Both the created geometry and the XML repository are accessible to other modules further downstream in the basf2 path, such as simulation, reconstruction, track extrapolation, etc. Figure 3 shows this process in detail.

In addition of the nominal geometry, we are developing strategies to fill the gaps between the simulated geometry and the real detector geometry. The displacement library represents the design values and the constructed detector, in the run-independent aspects. The run-dependent aspects are covered by the alignment schemes. Finally, if necessary, run-dependent differences will be implemented in the simulation library as mis-alignment.

<pre> <b># Main steering script of basf2</b>  <b>from basf2 import *</b>  <b>main = create_path( )</b>  <b># No of events to be generated</b> <b>main.add_module(EventInfoSetter)</b>  <b># Generator</b> <b># (EvtGen, particle gun, etc.)</b> <b>main.add_module(EvtGen)</b>  <b># Simulation</b> <b>add_simulation (main)</b>  <b># Reconstruction</b> <b>add_reconstruction (main)</b>  <b># Output</b> <b>main.add_module (output)</b>  <b>process (main)</b> </pre>	<pre> <b># Script for simulation</b>  <b>from basf2 import *</b>  <b>def add_simulation(path):</b>  <b># background mixing</b> <b>path.add_module (BGMixer)</b>  <b># geometry parameter database</b> <b>path.add_module (Gearbox)</b>  <b># detector geometry</b> <b>path.add_module (Geometry)</b>  <b># detector simulation/Interface to Geant4</b> <b>path.add_module (FullSim)</b>  <b># PXD simulation (digitization, clustering)</b> <b>path.add_module (PXD)</b>  <b># SVD simulation (digitization, clustering)</b> <b>path.add_module (SVD)</b>  <b># Other sub-detector codes here.</b> </pre>
---	---

**Figure 2.** To the left is an example of a basf2 job steering script in Python. To the right is the script for the simulation packages of basf2. This is an update of the version shown in [2].



**Figure 3.** An example of a basf2 job. Time flows from left to right. The geometry of the Belle II detector is created using a set of parameters stored in a XML repository file. The actual creation is done in the Geant4 simulation software via C++ coding. The resulting geometry is used by other modules in the downstream of the basf2 path. This is a variation of a version shown in [2].

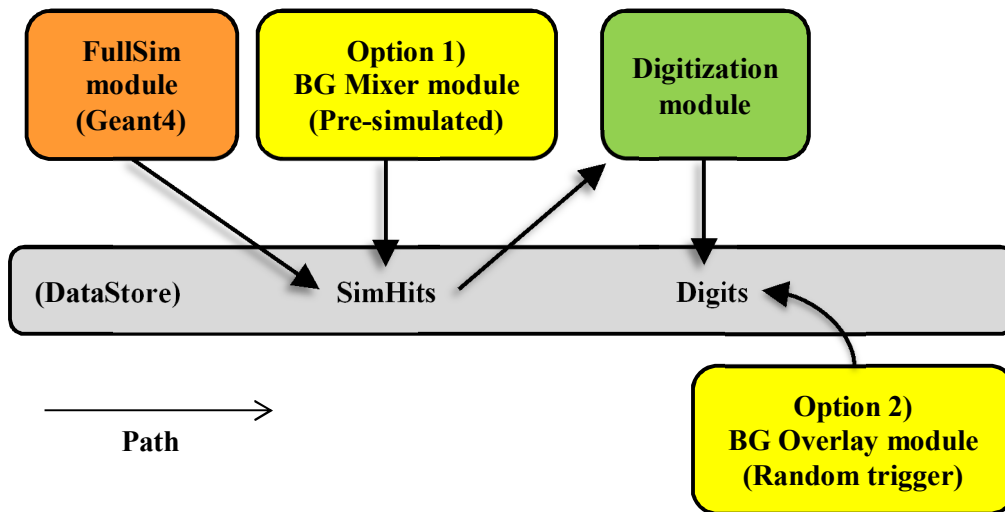
### 5. Mixing of beam background hits

With much more intense luminosity created by SuperKEKB, we expect the level of beam background will be much higher in the Belle II detector. It's important to emulate the effects of this background in simulation precisely, which will definitely affect the performance of the Belle II detector.

We approach the subject with two different strategies: the background mixing method and the background overlay method [9]. The first one uses a separate background simulation while the second one uses real data events. Figure 4 shows the position of these two methods in a basf2 path.

The background mixing method is built on the assumption that the majority of the beam background effects could be emulated by simulated beam background events. We also use Geant4 to simulate of background events, and steps of each particle trajectory created inside the Belle II detector are stored in a file. Then this file is used to mix the steps from the background events with the steps from a physics event. Next, the mixed sample is sent to the digitization module in basf2 for further processing.

The background overlay method uses random trigger events to create a background event sample. Then the sample is mixed with a physics event during the digitization process. Even the noise from the detector electronics can be included in this way. However, some background properties may be lost due the threshold effects.



**Figure 4.** Two strategies of mixing beam background hits with the physics simulation. Time flows from left to right. With option 1, a set of pre-simulated background *SimHits* is mixed with the *SimHits* from a physics simulation event. *SimHits* are steps of a trajectory generated during the execution of Geant4. At the following stage, the mixed set of *SimHits* is sent for digitization. The digitized output is called *Digits*. With option 2, data obtained from random trigger events are mixed at the digitization level.

## 6. Validation tools

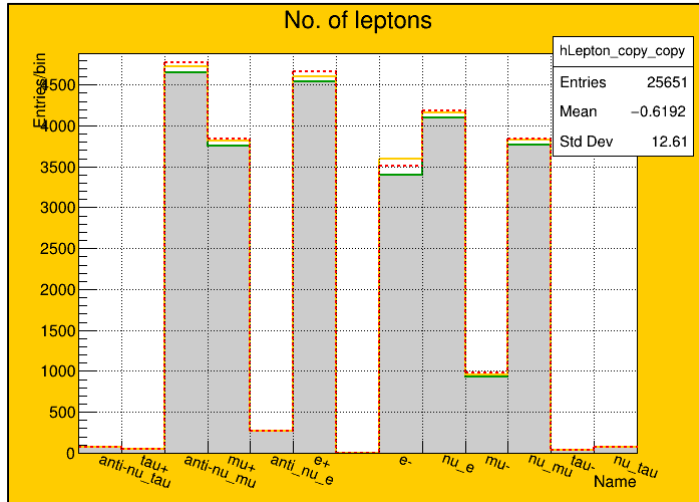
The development version of basf2 is updated continuously. To ensure accuracy and stability of library, a set of validation tools is executed after the daily build-up of the development version. The most recent versions of daily build-up are compared with the references including release and monthly versions of basf2 library [10].

For the core of the simulation package, a set of basic parameters are checked in histograms:

- Number of particles such as leptons, stable mesons, and stable baryons.
- Energy and vertex distributions of electrons and photons.
- Energy and vertex distributions of secondary atoms or their fragments. (The number of these secondary particles is very small, though.)
- Energy spectrums of muons, stable mesons, and stable baryons.

Figure 5 shows an example histogram from this package. We found that this set of simulation histograms is very sensitive to changes in the detector geometry, which makes it a very good indicator for changes in geometry. In addition to this, the generator group and each sub-detector group also maintains a set of histograms to check the modules used for simulation.

Belle II Validation	
☰	Revisions
<input checked="" type="checkbox"/>	reference
<input checked="" type="checkbox"/>	28533
<input checked="" type="checkbox"/>	28564
<input checked="" type="checkbox"/>	28579
<input checked="" type="checkbox"/>	28589
<input checked="" type="checkbox"/>	28610
<input checked="" type="checkbox"/>	build-2016-05-01
<input checked="" type="checkbox"/>	pre-release-2016-04-20
<input checked="" type="checkbox"/>	release-00-07-00



**Figure 5.** An example of validation histogram. The left table shows the version of basf2 libraries used for comparison. The right histogram is counting of electrons, muons, taus, neutrinos, and their anti-particles during the simulation. All the jobs were created with a same random number seed.

## 7. Summary and future plan

### 7.1. Summary

The nominal geometry of the Belle II detector has been implemented in basf2. Currently we are implementing the displacement geometry, which describes the differences between the design values and the actual construction. For the sub-detectors, we are also developing run-dependent alignment tools, which are separate from the displacement geometry.

The core part of the simulation library, the *FullSim* module, has been upgraded Geant4 from version 9.6.2 to version 10.1.2. The upgrade activities were completed without big incidents. Comparisons between two versions show that they behave quite similar except minor differences. One notable difference was that the amount of the neutrons in the beam background simulation samples based on Touscheck, radiative Bhabha and Coulomb interactions was decreased by 25%. The physics list input used for the FullSim module is FTFP\_BERT, which is provided by the Geant4 group for the collider experiments.

### 7.2. Future plan

Currently the geometry parameters are stored in an XML repository file. This repository will be replaced by a Database. The simulation of the magnetic field will be further refined to reach a precision of 0.1%. For this, additional in-situ measurements and simulation studies are planned. Several strategies are in development to use the cosmic and Phase 2 data to validate and monitor the simulation.

There is also a plan to pursue the option of a fast simulation library in place of Geant4. However, the scale of the calorimeters of the Belle II detector is smaller compared to those of the LHC experiments, which translates into relatively lighter usage of computing resources by the Belle II calorimeters.

## Acknowledgements

The author Doris Y Kim is supported partially by the National Research Foundation of Korea (NRF) Grant No. 2016R1D1A1B02012900 and No. 2016K1A3A7A0900560. This author also thanks Korea

Institute of Science and Technology Information (KISTI) for the usage of the Global Science Experimental Data Hub Center (GSDC) and Korea Research Environment Open Network (KREONET/GLORIAD) resources, and for the technological supports.

### References

- [1] Abe T *et al.* arXiv:1011.0352 [physics.ins-det]
- [2] Moll A 2011 *J. Phys. Conf. Ser.* **331** 032024
- [3] Itoh R *et al.* 2012 *J. Phys. Conf. Ser.* **396** 022026
- [4] <http://www.python.org/>
- [5] <http://root.cern.ch/>
- [6] Itoh R *et al.* <https://inspirehep.net/record/445749>
- [7] Agostinelli A *et al.* 2003 *Nucl. Instr. Meth. Phys. Res. A* **506** 250-303  
Allison J *et al.* 2006 *IEEE Trans. Nucl. Sci.* **53** 270-8
- [8] <http://operafea.com/>
- [9] The Belle II collaboration and B2TiP theory community The physics prospects for Belle II, to be published.
- [10] Kuhr T and Hauth T 2015 *J. Phys. Conf. Ser.* **664** 062024