The Software Library of the Coming Belle II Experiment and its Simulation Package

Doris Yangsoo Kim
Soongsil University

On behalf of the Belle II Software Group

October 30, 2013
IEEE 2013 NSS
SuperKEKB Luminosity Projection

Goal of Belle II/SuperKEKB

- Commissioning starts in early 2015.
- Shutdown for upgrade in early 2015.
Physics at a Super B Factory

• A B meson is a long lived particle with many interesting decay modes. Easy to identify and study the decay process.

• Precision test of CKM unitarity matrix

• There is a good chance to see new phenomena:
  – CP Violation from the new physics.
  – Lepton flavor violations in $\tau$ decays.
  – Search for the charged Higgs boson in $B \rightarrow \tau \nu$, $D(*)\tau \nu$ decays.
  – New particles affecting the flavor changing neutral current.
  – More topics: CP Violation in charm mesons, new hadrons, …
Upgrade Process to SuperKEKB

To get x40 higher luminosity
The Belle II Detector

- **Electrons (7 GeV)**
  - Beryllium beam pipe: 2cm diameter
  - Central Drift Chamber: He(50%):C$_2$H$_6$(50%), small cells, long lever arm, fast electronics
  - Vertex Detector: 2 layers DEPFET + 4 layers DSSD

- **Positrons (4 GeV)**
  - KL and muon detector:
    - Resistive Plate Counter (barrel outer layers)
    - Scintillator + WLSF + MPPC (end-caps, inner 2 barrel layers)

- **EM Calorimeter**
  - CsI(Tl), waveform sampling (barrel)
  - Pure CsI + waveform sampling (end-caps)

- **Particle Identification**
  - Time-of-Propagation counter (barrel)
  - Prox. focusing Aerogel RICH (fwd)

---

October 30, 2013
## DAQ - Event Rate

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Event Size [kB]</th>
<th>Rate [Hz]</th>
<th>Rate [MB/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>High rate scenario for Belle II DAQ</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Belle II</td>
<td>300</td>
<td>6,000</td>
<td>1,800</td>
</tr>
<tr>
<td>LCG TDR (2005)*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ALICE (HI)</td>
<td>12,500</td>
<td>100</td>
<td>1,250</td>
</tr>
<tr>
<td>ALICE (pp)</td>
<td>1,000</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>ATLAS</td>
<td>1,600</td>
<td>200</td>
<td>320</td>
</tr>
<tr>
<td>CMS</td>
<td>1,500</td>
<td>150</td>
<td>225</td>
</tr>
<tr>
<td>LHCb</td>
<td>25</td>
<td>2,000</td>
<td>50</td>
</tr>
</tbody>
</table>

* The LHC experiments are running at a factor of two or higher event rates
Distributed Computing System

- Distributed Computing based on DIRAC+AMGA
- Use GRID, cloud computing and local resources

Raw Data Duplex Reprocessing

Optional MC Production

Raw Data Storage and Processing

MC Production and Ntuple Production

Ntuple Analysis

Local Resources

Grid Site

KEK

PNNL
The Belle II Software System

- A “framework” system with dynamic module loading, parallel processing, Python steering, and ROOT I/O
- Full detector simulation with Geant4
- Code management systems at KEK: The Subversion software
- All common linux operating systems supported: SL, Fedora, Ubuntu, etc
- C++ 11 and gcc 4.7
- Formatting tool: astyle
- Building: scons and buildbot system
- Documentation: Doxygen, Twiki
- Issue tracking: Redmine
The Upgrade of the Software from Belle

• BASF2 (Belle Analysis Framework 2) 
  Basic ideas from the Belle software system + Constructed from scratch.

• Old system: C++. Panther tables to store data.

• New system: C++ based. Object oriented.

• Imported useful concepts from other experiments: 
  ILC, LHCb, CDF, and Alice

• Third-party software libraries: 
  ROOT, boost, CLHEP, and many others.

• Software developers are from all around the world.
The Basic Structure of BASF2

- **Module**: The basic processing unit
  - Examples: As simple as reading data from a file to complex ones like simulation or tracking
  - All the works are done in modules.
  - Selection and arrangement of the modules are done by a user.
- **A typical event processing** = a linear chain of modules on a **path**
- **Datastore**: A common storage for data
Multiple Processing Paths Available

- Multiple paths are allowed.
- The paths can be created and merged by conditions.
- Useful for skimming data.
Switching between normal and parallel processing is to be transparent to users.
Event-by-event parallel processing.
Linux processes forked after initialization.
Partial parallel processing:
  - Input and output processes are run in a single path.
  - Only the modules with parallel processing property are run in a parallel path. Example: MC simulation
• **Libraries**: Separated from modules to increase reusability.
  
  – Methods and algorithms are encapsulated in libraries.
  
  – A library (i.e., algorithm) can be used/shared by several modules.
All the geometry parameter values are stored in the central repository.

The actual geometry for simulation is created from the repository parameters via C++ algorithms.

For reconstruction, it is converted to ROOT TGeo via VGM library.

Parameter values are directly available to users. Easy maintenance and quick updates possible. Increased efficiency due to C++.
Scheme of Detector Simulation

Main Steering File

from basf2 import *
main = create_path()
# event meta data
main.add_module(event_meta_demo)
# generator (evtgen, particle gun, etc.)
...
# simulation
...
# reconstruction
...
# output
...
process(main)

Simulation Steering File

from basf2 import *
# geometry parameter database
...
# detector geometry
...
# detector simulation
...
# background mixing
...
# PXD simulation (digitization, clustering)
...
# SVD simulation (digitization, clustering)
...
# Other sub-detectors here.
...
Simulation Example:

“The Secondary Particles Created by Geant4”

- Most of the particles simulated by Geant4 are secondary particles.
  - For example, an $\psi(4s)$ into generic B Bbar decay in the inner detectors of Belle 2 (PXD, SVD, CDC, TOP, ARICH) creates 34,000 particles during Geant4 simulation.

Note: Optical photons and particles < 1 MeV kinetic energy are not included in the plot. They are simply too many.
• Decision 1: In the output file, optical photons and secondary particles with kinematic energy less than 1 MeV are not stored.

• At this stage, an $\psi(4s)$ decay in the full Belle II geometry creates 2,000 particles during Geant4 simulation. The output MDST file size is still too large.

• Decision 2: Only the necessary secondary particles are kept in the output file such as
  - Decays-in-flight
  - Secondary particles leaving hits in the sensitive detector area.
  - This achieves 90% reduction for the MDST file size, which is acceptable.
Digitization in the silicon vertex detectors.

Digitization Software

- Detailed information on the detector material and electronics is needed: Sub-detector experts write the digitization code.
- Coded as modules rather than complete integration into Geant4: Different versions of sub-detector digitization would be available.
• Pre-simulated Geant4 background hits are added as SimHits (Geant4 steps) to the already existing SimHits from the signal event.
• The background events could be even beam background (Touscheck).
• Then both contributions are digitized at the same time.
• Pros: Closer to reality.
• Cons: Can’t be used for measured background data.
Event Display

Basf2 + ROOT with OpenGL support
• Over 500 scientists. More from Canada and Italy recently.
• Active development in software: Half a billion events were recently simulated and reconstructed at an MC campaign without any basf2 crash.
• Many thanks for the colleagues who provided valuable ideas, data and plots for this poster.