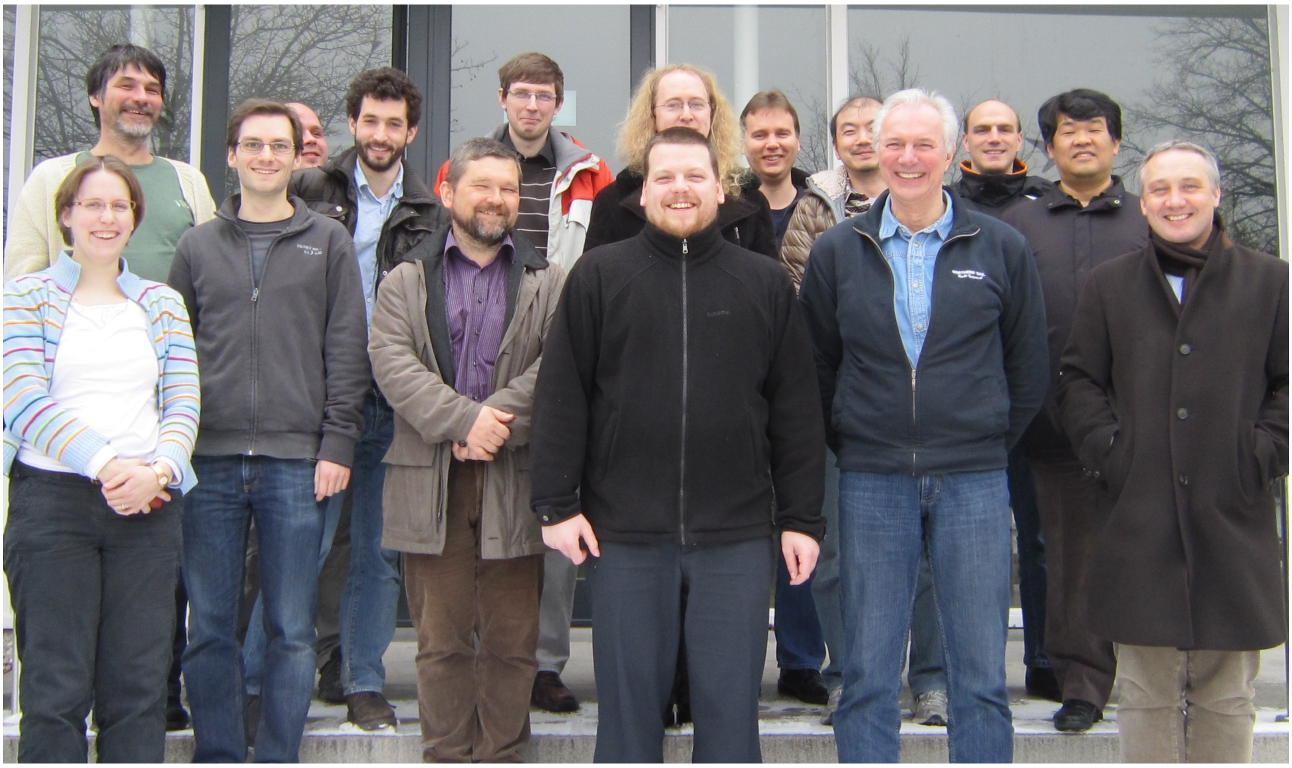
Minutes of the Belle II framework and PXD DAQ meeting

in Munich, 21 to 23 February



Geometry

It became apparent that we need to store additional information, such as volume ID's, sensitive detector flags or optical surface properties, for each TGeoVolume of the Geant4 simulation. Since the current implementation of the TGeoVolume class doesn't provide a way to embed additional information, we agreed to have a look at the solutions implemented by other experiments. We considered adding the missing feature to the TGeoVolume class by temporarily patching ROOT, while also sending a feature request to the ROOT developers.

Measurements have shown that the PXD ladder will bend due to gravitational and mechanical forces. We discussed the implementation of PXD misalignment in the framework and reached the conclusion that we will treat the PXD ladder as a rigid object in the Geant4 simulation. As per Peter Kvasnicka's idea, we will introduce the ladder sag into the digitization step through the correction of the space points. Misalignment due to rotation or translation is not taken in to account by this correction, but will already have been accounted for in the Geant4 simulation. Martin Ritter proposed that we subdivide the PXD ladder into smaller planar parts for the alignment process. The size of these subparts will depend upon the amount of bending. As Marko pointed out, bowing will also affect the TOP crystals.

Currently, all subdetector libraries have to be linked into the main framework at compile-time whether or not they are used. In order to overcome this problem, the subdetector libraries should be loaded dynamically only if they are required by the framework. Martin Ritter proposed that the subdetector library name be added as an attribute to the Creator tag in the subdetector's XML file. The framework would then be able to load the required libraries dynamically as it looped over all subdetectors specified in the Belle II detector XML file. The Creators defined in the library would then be registered to the framework automatically by loading the subdetector's library. We discussed the possibility of providing the user with a comfortable way of turning specific subdetectors on and off in the steering file. We reached the conclusion that the geometry building module should implement a list of all subdetectors that need to be built.

An overlap check of the full Belle II detector geometry revealed that we have over 150 overlaps at this time. As first simulation tests have shown, these overlaps can lead to endless loops in the Geant4 simulation. Hara-san will investigate the code that generates the overlaps. We are to add an automatic overlap check to the nightly build system. We will pursue the idea of automatic overlap checks by implementing an additional overlap check at the Geant 4 level. Frank Simon will have a look into this issue.

Hara-san presented Nakao-san's current implementation of the magnets in the interaction region. We agreed that Nakao-san's work should be committed to SVN as soon as possible so that background studies can be performed with the new geometry. Andreas Moll showed an overlay of the current data, simulated with SAD, on the current interaction region geometry. He found several mismatches between the geometry used in SAD and that implemented in the framework. We came to the agreement that we have to make sure that both geometries are identical.

The current material system used in the framework doesn't provide a predefined atom table. It was found that most developers implement their own atom definitions. Considering the various possible common atom tables, we reached the conclusion that the NIST table found in Geant4 is the most reliable solution. The NIST table guarantees that the correct value for atoms, isotope compositions and radiation lengths will be used throughout the whole framework. However, by applying this solution, we will be faced with the problem of how to feed back the Geant4 information into TGeo. Thomas Kuhr proposed to investigate the solutions found by other experiments.

Generating the geometry for each simulation job is a time consuming process. Martin Heck pointed out that the framework should offer the option of loading an already existing geometry from file. We discussed the mechanism of the restoration of the full geometry information with particular reference to the restoration of the connection between volumes and sensitive detector classes. We decided to separate the geometry building code from the sensitive detector assignment code. Andreas Moll proposed to use the additional information feature (described above) to store the sensitive detector classes for each sensitive volume.

Kai Bürger, from the visualization group at the Technical University of Munich, took part in the meeting as an observer. Together with MPI, they pursued the idea of offering a Diploma/Masters thesis for the development of the core features for an event display. We briefly discussed the basic hardware requirements for such an event display, and agreed upon having OpenGL 2.1 as a minimum requirement.

ARICH, TOP

Marko Petric presented the current status of the TOP and ARICH subdetectors. He found the TGeoXtru volume class to generate photons correctly, but lack the effect of total internal reflection. In order to generate physically correct simulation output, he dropped the use of the TGeoXtru volume type, and used TGeoBBox and TGeoArb8 instead. Marko will collect his experience with the shortcomings of the TGeo volumes in Geant4 into a list for the TWiki.

The first results obtained from the TOP simulation already look very promising.

Marko also found that certain types of comments in C++ are significant for the ROOT dictionary building process and should be avoided. Andreas Moll presented this issue at the last B2GM and will update the current TWiki page as soon as possible.

The propagation of optical photons takes up most of the simulation time in the TOP and ARICH detector. Marko and Luka Santelj investigated several methods for reducing the time spent in the Geant4 simulation. Since the photo-multipliers that are used for the detectors are sensitive only in a limited wavelength range, there is no need to track all produced photons through the TOP and ARICH volumes. Marko presented Luka's results on changing the way in which optical photons are rejected from the simulation. On Luka's computer the simulation of all photons produced in the ARICH took 58s. He implemented a StackingAction into Geant4 that eliminated 60% of all photons by removing Cerenkov photons, but left the scintillating photons untouched. This change in the Geant4 simulation code cut the simulation time down to 24s. We came to the conclusion that Luka will prepare the StackingAction code and Andreas will then implement it into the Geant4 kernel.

Optical photon processes play a special role in Geant4. They require a special physics list to be prepared. Marko expressed the concern that this optical photon physics list might lead to time consuming simulation steps even for subdetectors that don't require optical photon support. He proposed a feature that will allow the physics list for each detector component to be set separately. With the help of the Geant4 experts at KEK, Hara-san will ponder the possibility of implementing such a feature in Geant4.

For the TOP reconstruction, Marko is porting the existing Fortran code to C++. In order to check the new implementation, he proposed to link the Fortran code with the framework. As soon as the new code is validated, he will remove the Fortran code. We decided that this approach is a good idea.

Digitizer

Peter Kvasnicka reported the current status of the Digitizer. Due to the limitations of the ILC software framework, the digitizer and the clusterizer were combined into one module. For the implementation in basf2, Peter will separate both.

We debated the storage of pixels in the DataStore and how to support the reading and writing of raw data. We came to the agreement that the raw data will be stored as a list of integer arrays, where each array consists of the data collected by a single readout unit. In order to convert the raw data into convenient DataStore objects, Martin Heck will implement a base class for all DataStore objects that will represent real data. This base class will provide methods for reading and writing the DataStore object's data to and from the appropriate raw data format. Each subdetector group has to define the raw data format and implement these methods for their DataStore objects.

The current implementation of the digitizer uses two different Landau fluctuation calculations for the charge smearing along the track. We decided to follow Peter's suggestion and only use the Geant4 Landau fluctuation calculation. This requires a reduction of the step size for the PXD to 5 micron or less. To restrict the small step size value to the PXD, we will implement a mechanism which allows the developer to set the step size for each volume separately.

As it was discussed at the DEPFET workshop in Bonn, the digitizer is not yet able to handle photons. Peter will implement the correct treatment of photons into the digitizer.

Framework

We debated the current implementation of the master module handling in basf2. The framework allows for exactly one master module, to be determined automatically at runtime. Only the master module is allowed to set the experiment, run and event numbers for the event processing. Susanne Koblitz pointed out that in the case of the HepEvt input module, the module is able to set the event number but not the experiment and run number. We reached the conclusion that the HepEvt input module would be able to behave like a master module through the addition of the experiment and run numbers as parameters.

The DataStore provides the possibility to connect two DataStore objects using a Relation object. The Relation object uses internal *to* and *from* indexes to point to the objects in their corresponding collection. In addition, each Relation object also stores the names of the DataStore collections it connects. As it turns out, almost all Relations which are grouped inside one TClonesArray connect objects from the same collections. This leads to a redundant storage of the collection names inside each Relation. We decided to follow Martin Heck's proposal of implementing an inherited class of the TClonesArray for collections in general and for a Relation collection in particular. The Relation collection would then store the names of the collection it connects only once and the Relations as a list of index pairs.

Thomas Kuhr provided an insight into the problem of deleting and inserting elements into collections which are connected by Relations. We considered creating a base class for collections which would hide methods for deletion and insertion from the user.

To facilitate the work with Relations, we agreed on developing a Relation manager class. This class will provide a convenient interface for generating relations from either indexes or DataStore object pointers.

Peter Kvasnicka volunteered to implement and test the relations between PXD clusters and pixels.

The generators, Geant4 simulation and digitizers require a random number generator. In order to create unique and reproducible random numbers, the seed value for the random number generator has to be managed centrally. Special care has to be taken if the framework is executed in the parallel processing mode. We arrived at the agreement that we need a central random number generator service, to be based on TRandom3. Additionally, we decided that the random number seed should be saved in the output and the log file to allow the recreation of the exact same data easily.

Thomas Kuhr presented the new tools command *setoption*, which switches between a debug and an optimized build of the framework. The build process for the externals also takes the selected build mode into account.

The different readout times for the subdetectors make it necessary to overlay a different amount of background events over each signal event, depending on the subdetector. Andreas Moll, Susanne Koblitz and Martin Ritter proposed a pool of simulated DataStore objects and an overlay module which will add the right amount of background to an already simulated signal for each subdetector be created. We decided that Andreas will generate this background pool.

Susanne Koblitz implemented a CMS to Lab frame boost function, to be used inside the generators package. In order to make this function available to all users, Susanne proposed to create a new package, *utilities*, and add the method to it. The decision about the responsible librarian for the new package has not yet been made. If you are interested, please feel free to contact Thomas Kuhr.

The current version of basf2 contains a library dependency cycle between the packages *geometry* and *simulation*. Unfortunately, this cycle can't be resolved easily. We are considering merging both packages, as they both depend on functions provided by Geant4.

We briefly discussed the naming conventions for DataStore objects. We agreed that the naming conventions of collections of objects will be *[object class name]* + “s” and relation collections will be *[from collection]*To*[to collection]*. Thomas Kuhr will add details about the naming conventions to the TWiki.

Andreas Moll has to concentrate on his thesis, and so is no longer available as a package librarian. We decided to change the package librarians as follows:

framework	Martin Heck
geometry	Takanori Hara
ir	Takanori Hara
pxd	Jozef Koval
svd	Jozef Koval

Simulation

Hara-san provided insight into the magnetic field map implementation of BELLE. A constant and hardcoded B-field of 1.5 T is used for the tracking tools, while the simulation and the PID reconstruction tools use magnetic field maps that are loaded from files. We discussed the implementation for Belle II and reached the conclusion that we need a central library for retrieving the B-field. Andreas Moll will implement a first version of the interface for the B-field map library, to return a constant magnetic field of 1.5 T for now.

We discussed the drawbacks of TGeoVolumeAssembly objects in terms of Geant4 simulation speed. As it turns out, TGeoVolumeAssembly objects slow down the navigation for the Geant4 simulation significantly. Christian Oswald proposed to use TGeoTubes instead. Jozef Koval will implement them for the PXD and SVD and will report the impact on the Geant4 simulation.

Generators

Susanne Koblitz presented her ideas for the integration of EvtGen into basf2. We decided to use the centrally hosted and maintained release of EvtGen from Warwick university for the framework. In order to keep the improvements BELLE made to its own EvtGen version, we decided that those improvements should go into the central version.

PXD DAQ

The Gießen crew (Sören, Björn and David) showed the recent progress on the PXD DAQ hardware and the implementation of the various algorithms intended to be run on the ATCA system.

1. Compute nodes (CN): The IHEP group has prepared 2 CNs with "small memory" of 2 GB, but with a fast optical link reaching 6.5 GB/s. With these boards the benchmark tests for the final decision for (or against) the ATCA system will be executed. The IHEP group is also working towards the final design of a "large memory" board (4 GB per node) which will be sufficient to store about 5 sec worth of PXD data in the ATCA system. This board will become available by the summer of 2011.

2. Benchmark Test: Björn showed recent results on a bandwidth benchmark test (transfer an amount of data corresponding to 1% occupancy PXD ladder at 30kHz in and out of the memory of one FPGA using an IP Core). The progress on the system aspects is very good. However, the presently achieved memory access bandwidth corresponds to 1% pixel occupancy and is therefore not yet sufficient. There are, however, clear ideas how to improve the situation and finally reach the required speed.

3. VHDL Code development: David showed the progress in VHDL programming of the Hough algorithm, making use of the SVD data for PXD data reduction. The code is very demanding and it is clear that the "naive" code (no sectorization a la Claudio) will probably not be fast enough. In addition to the sectorization one can gain a factor of over 100 by using the adaptive, fast Hough transform with zooming 2-dim window size, a method shown in Claudio’s diploma thesis. This would certainly lead to the required execution speed (30kHz). On the other hand, it is not clear at the moment whether the existing FPGA's would be sufficient to hold the code.

4. Sources of ROIs: It was decided that for the final system two sources for ROIs will be

used: the ATCA system itself with the SVD input, and the results from the full track fitting on the HLT. It was pointed out that the present track efficiency from the full reconstruction is not sufficient to rescue all the important pixels.

5. Self-tracking: In order to rescue also the pixels from the slow pions with (transverse) momenta below 80 MeV, it was decided to introduce a stand-alone analysis on the ATCA system executing a clustering algorithm for the identification of higher-than-mip energy depositions. The cluster algorithm should assume ordered data from the individual DHH's. Due to its FIFO design, the DHP cannot guarantee a geometry-ordered stream of pixels from the individual ladders, and the ordering must be done at a later stage in the readout sequence. The most appropriate location to perform the ordering appears to be the DHH.

HLT-PXD DAQ Interface

1. DAQ-Flow and HLT development: Itoh-san showed the proposed architecture of the DAQ system for the Belle II detector. All sub-detectors except the PXD go through the COPPER/Finesse system. For the PXD the ATCA system is the default, the PC solution is the backup. Details of the data flow through the event-builder stages and the functionality of the High Level Trigger (HLT) were shown. The HLT will do a full event reconstruction using calibration and alignment constants from an online data base. This stage of reconstruction is optimized to achieve high efficiency for the final (software) trigger decision before the events go to final data storage. The raw data will find their final destination on the Tier 0 computing facility of KEK. This transfer will be the final action of the Belle II DAQ system.

2. Data Quality monitoring: Itoh-san also presented ideas for the data quality control of the PXD data by introducing an online (“prompt”) full reconstruction scheme. This scheme will also be used for the other sub-detectors. The idea of the “prompt reco” is to a physics-oriented full reconstruction with “offline” calibration and alignment constants. With this reconstruction level, physics analyses should be possible. Another function of the prompt reco is to integrate data over an adequate time period in order to establish time-dependent calibration and alignment constants for an update of the offline-and online data bases.

3. HLT and PXD data reduction: The PXD data reduction system should be prepared to use the results (ROIs) from the online reconstruction of the HLT, where all possible tracking algorithms could be executed in principle. For the ATCA system this means to provide enough storage to hold about 5 sec of data at 30 kHz (see also the section on PXD-DAQ). To wait for the HLT ROIs is now considered the default.