Status of the Digital Simulation of the PXD

Peter Kvasnička
Institute of Particle and Nuclear Physics,
Charles University, Prague

6th International Meeting on DEPFET detectors and applications, Bonn, 7-9 February 2011

CU Prague

Implementation of the PXD and SVD Digitizers in basf2

- **Digitizers simulate detector response based on simulated particle tracks.**
  - Currently, there are only simple gaussian-smear digitizers in basf2 using resolution data from the full ILC/-based simulation. Together with the Sensitive detector types and data objects (“hit types”) they form a simplistic simulation chain.
  - For real simulation studies, in particular of PXD background and SVD pattern recognition, more sophisticated simulation of detector response is **vital**.
- **Task: Implement the PXD/SVD digitizers used in the ILCsoft framework in basf2.**
  - The digitizers (written by Zbynek Drasal) are quite detailed and proven.
  - They are undergoing development towards even more detailed description of detector response (Zbynek – improved simulation of energy straggling; Benjamin Schwenker – lateral electric fields)
  - Also, the Sensitive Detector classes have to be updated, to properly handle secondaries.
- **The plan was to include the PXD digitizer in the first release of basf2.**

CU Prague

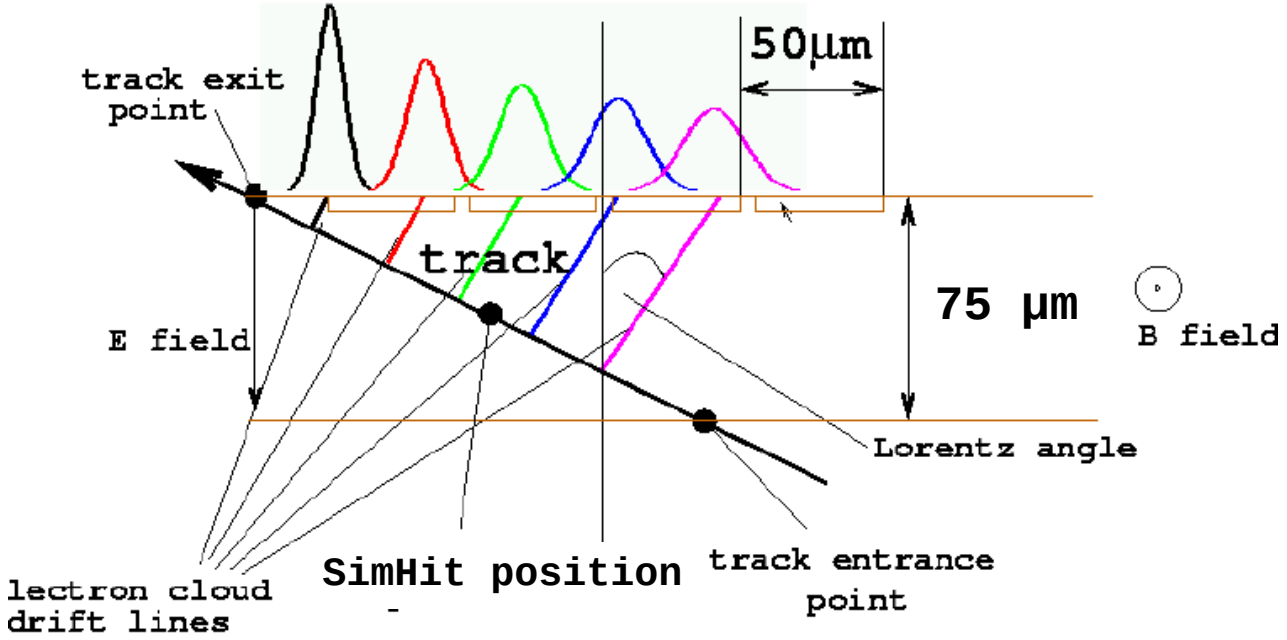
Only Simple Digitizer in basf2 1.0

- **The “full” digitizer implementation is not ready for the first release of basf2:**
  - **The code needs a couple of days to go to svn,** and some more to be tested against the ILC digitizer.
  - T**he SeinsitiveDetector class** (common to PXD and SVD) is in a similar state, albeit it is a simpler piece of software and also testing is much simpler.

CU Prague

The PXD digitizer: What it does

- **For each simulated hit (PXDSimHit) obtained from Geant4**
  - **Track generation.** The SimHits are created where Geant4 detects a passage of a particle through an active detector. δ electrons are treated separately
  - **Ionization.** The track segment is divided into sub-segments ("ionization points"), whose charge is smeared according to the Landau distribution using the code borrowed from Geant4 (G4UniversalFluctuation). This provides substantially higher precision than in the generating "all-detector" Geant4 simulation.



CU Prague

The PXD digitizer: What it does (cont'd)

- **Drift and diffusion.** The ionisation points are then drifted to the readout plane, Lorentz-shifted in the presence of a magnetic field and smeared by a Gaussian distribution to account for diffusion during the drift time.
- **Lateral diffusion**. Finally, the total charge of an ionization point is split into carrier groups of ~100 electrons. For each carrier group, a random walk iin the readout plane is sampled until the internal gate of a pixel cell is hit.
- **Digitization.** For each pixel electrode one gets a signal dependent on how many ionization points contributed).
- **Background and noise.** The pixels are then populated by random electronic noise and background signal as appropriate.
- **Analog-To-Digital converters** will be used in the data processing pipeline, Therefore, the analog signals are converted first into digital values.

CU Prague

The PXD digitizer: What it does (cont'd)

- **Hit reconstruction.** For practical purposes, digitization is coupled with hit reconstruction to avoid the storage of digits when they are not needed (the digits can however be saved if desired).
- **Clustering i**s based on seed and neighbor threshold values.
- **Hit position** is estimated separately in each coordinate, by center-of –gravity for cluster projections of 1 or 2 pixels, and analog head-tail (approximate the cluster by a uniform charge distribution consistent with pixel signals) for larger clusters.
- **Output** The output of the digitizer is a collection of PXDHits, with each reconstructed hit containing (apart from sensor identification)
- position estimate (corrected for mean Lorentz shift) and its error covariance
- total collected charge (analog in electrons or digital in ADU
- references (relations) to SimHits that contributed.

CU Prague

So where's the problem?

- **There is no major problem**
  - The main body of work is to translate the code from ILCsoft to basf2. There are no significant issues, rather a lot of small ones (units in empirical formulas of the Geant4 code, for example).
  - Basf2 is evolving – new features (implementation of MCParticles and relations) appearing – this makes the transition simpler than it was initially.
  - Geometry is no problem, in the basf2 implementation the hits are in local sensor coordinates, so the original geometry code can be disposed off completely.
  - **Things the actually need to be changed will be changed later:**
    - Automate calculation of error covariance of reconstructed hits.
    - Disentangle clustering/hit reconstruction as much as possible from the digitization code, but keep the efficiency of not storing digits if they are not needed. Maybe a plug-in that can be used in a standalone module or as a part of the digitizer module, and can be replaced by a different code to test.
    - Major task: simulate DHP and following processing.
    - "Last minute" issues (Andreas): check / fix the deposited energy smearing and handling of photons by the digitizer.

CU Prague

Schedule update

- There is now little to finish for the first submission.
- It is to appear in the svn within one week (by 15. February).
- One more week is needed for the SVD digitizer (different code, but similar modifications to be made as for the PXD) .
- A test suite for the digitizers will be a necessary tool in further development. This will be proposed/discussed at the Munich framework/DAQ meeting in Munich.

CU Prague

Thanks for your attention!