Emc Edge Correction Reco Code

John Back
University of Warwick

11th Aug 2004
Edge Correction Analysis

- Several people have done the nice physics analysis for this:
  - Initial studies: E. Maly (BaBar Note 557)
  - Further studies by S. Schenk (Heidelberg)

- Code at the Beta level (see neutrals AWG web pages)

- Problems observed with:
  - propagation through decay chain
  - vertexing/new BtaCandidates

- Beta-type code from Roodman et al needed to be implemented at reco level
  - Specifically, in the (default) EmcMkIIComboCalibrator
Reco Implementation

- Apply edge correction in `EmcMkIIComboCalibrator::energyOf()` function.
  - Get EmcEdgeCorr object from the environment
  - If this is absent, no correction is applied (factor = 1.0).
- Picked up automatically in reco/BetaMiniUser jobs (even in “cache” mode).
- Emc Edge correction constants retrieved from conditions database
  - Constants stored as 10-column ntuples, via CdbTable/CdbNTuple interface.
- `EmcEnv/EmcEdgeCorrTable` retrieves the constants from the `CdbNTuple`
  - `EmcEnv/EmcEdgeCorrTable` inherits from general class `EmcEnv/EmcTable`. 
Software Structure

- Start of the job: `EmcSequence/EmcInitSequence` does general initialisation
- `EmcEnv` is built via `EmcEnv/EmcBuildEnv` within this sequence
  - Now gets edge correction tables from DB via `EmcEdgeCorrProxy`
  - These `EmcTables` are then stored in the `Emc` environment
- `EmcInitSequence` also includes `EmcCalibToo/EmcEdgeCorrLoader`
  - Creates the `EmcEdgeCorr` object from the `EmcTables` mentioned above
  - `EmcEdgeCorr` is initialised at the start of the job, not every event.
  - `EmcEdgeCorr` is stored in the environment under the key “Default”.
- `EmcCalibToo/EmcMkIIComboCalibrator.cc` then picks this up (at construction) to apply correction whenever `energyOf()` is called.
The above implementation is available in:

- EmcEnv V00-09-01
- EmcSequence V00-05-01
- EmcCalibToo V00-03-01

Code tested using 15.3.0 on Linux down to BetaMiniUser level.

Note: Crystal co-ordinates used for correction specified within EmcCalibToo/EmcLocal. A. Roodman said that current geometry in DB does not handle cracks/gaps.
Loading constants into DB

- Need to load constants into the database
  - `gmake EmcEnv.extrabin` makes `EmcTableBdbLoad` executable, which loads the constants using the datafiles in EmcEnv.
  - `gmake EmcEnv.binscripts` creates the `loadEmcTable` perl script, which uses `EmcTableBdbLoad` to do the actual loading

- Need to decide which databases need updating. Validity dates?

- Can easily specify the validity date interval in `loadEmcTable` script

- Need to co-ordinate with DB experts...

- If no correction in DB, code handles the case gracefully (correction = 1.0).
Things to do

- Put constants into appropriate DBs (validity dates?)
- Put code (package tags) in next release
- Would like to test on full $\mu\mu\gamma$ control sample
- Take into account $\phi$ alignment/rotation of Emc (EmcLocal class).