MarlinTPC
Conditions Handling and Database

Ralf Diener

MarlinTPC Topics

- Which Condition Data should we store?
- Design of Conditions Objects
- How they are stored
- Database Layout
  Physical and Logical

MarlinTPC Which Condition Data

- **ADCChannelMapping:** Mapping of H/W channels to GEAR pad ind.
  - ChannelID
  - PadID
  - Type
- **ChannelCorrection:** Per channel calib.
  - Quality flags (broken, noisy)
  - Calibration factors
  - Time offset
- **Pedestal:** Per channel
  - Value
  - Width
- **TPCConditions:** Calibr. TPC Parameters
  - DriftVelocity
  - Diffusion (trans/long)
  - "Defocussing"
  - Amplification
- **GasConditions**
  - Pressure
  - Temperature
  - Flow
  - OxygenContent
  - WaterContent
- **GasMixture**
  - Contents
  - Fractions
- **FieldSettings**
  - Nominal drift field
  - Nominal B-Field
  - Especially for GEMs
  - GEM voltages
  - Transfer fields

See http://forum.linearcollider.org/index.php?t=msg&th=565&rid=0

MarlinTPC Which Condition Data → Objects

- **TimePixPixelMode**
  - Mode
  - Status (broken/noisy)
- **Read-Out Electronics**
  - Polarity
  - Readout frequency
  - Precision of electronics (maximum ADC count)
  - Specific settings for each type of read-out (ALTRO, T2K, TDC) information? → specific objects per type
- **WeatherConditions**
  - Temperature
  - Humidity
  - Pressure
- **Unsorted**
  - ADC ↔ Primary Electrons

- **My proposal:**
  - Take this list as a basis and check if we have everything that is needed
  - Every main item in this list will get an object assigned
  - Implement these Objects to store them in the database
- **Already in SVN** (status unknown to me)
  - GasConditions + GasMixture
  - Pedestal
  - Fieldsettings
  - Channelcorrection + Mapping
  - TPCConditions
  - WeatherConditions
  - TimePixPixelMode

See http://forum.linearcollider.org/index.php?t=msg&th=565&rid=0

MarlinTPC How are Conditions Data stored

- Use of LCCD package which is based on ConditionsDBMySQL
- Conditions can be stored in several ways, all writing and reading over lccd::DBInterface class.
  - Simple File:
    - LCIO file with one event that has a collection with the conditions data for the given time stamp and tag
  - MySQL database:
    - LCCollections of condition objects are stored in a folder structure in a MySQL database (one kind of object per folder)
    - Conditions data has a start and end time information:
      usually end time is set to far future and conditions handler always chooses the newest information (in the conditions time line)
    - Information can be tagged, that's the only way to get older conditions data if you write newer data for the same time slot
  - DBFile:
    - Creates an LCIO file with the all conditions data in the folder for the given tag
    - Useful when no database connection is available
- Reading conditions over ConditionsProcessor [combined with handler objects?]

MarlinTPC Database Layout

- **Need to decide on folder structure**
  - CALICE has sorted by detector and location
  - Proposal: /lctpc/large_prototype_1/<location>/<condition>[/<module>]
    Should be discussed!

- **Tagging?**
  - Proposal: start with 1.0 for all conditions and then increase version number by 0.1 for minor, by 1.0 for major corrections

- **Physical servers and database setup**
  - ConditionsDBMySQL is able to store data splitted over several database servers
    Do we need this?
  - CALICE uses 2 machines:
    - One is the main database, to which the Slow Control etc. writes the data
    - This data is duplicated to a second machine, which can be accessed by the users for the reconstruction