

- **Single-Module:**

- #define PedestalNINTVals 1
#define PedestalNFLOATVals 2
#define PedestalNDOUBLEVals 0
- **Constructor:**
Pedestal(int channel,
float PedestalValue, float PedestalWidth);
- **Channel Information:**
int getChannel();

- **Multi-Module:**

- #define PedestalNINTVals 2
#define PedestalNFLOATVals 2
#define PedestalNDOUBLEVals 0
- **Constructor:**
Pedestal(int channelID, int moduleID,
float PedestalValue, float PedestalWidth);
or
Pedestal(std::pair<int,int> hardwareID,
float PedestalValue, float PedestalWidth);
- **Channel Information:**
std::pair<int,int> getHardwareID();

- Changed the following classes to fully support the new pedestal format:
 - tpcconddata: Pedestal
 - calibration: PedestalCalculator
 - reconstruction: PedestalHandler, PedestalSubtractor, PulseFinder

- In these classes only single module values are produced and module ID is simply set to zero:
 - digitization: TPCElectronics
 - simulation: TPCCloudSimulation/SignalDigitisation

- Reads in pedestal information from database and provides interface for other processors to get them
- First idea was to use a Map
 - CALICE group said that this slows down their software (they also have to deal with pedestals and calibration constants etc.)
- Second idea was to use a vector (for the different modules) of vectors (for the pads on a module) and get the pedestal by referring the ModuleID/PadID to positions in vectors
 - Vectors get to big if users use large integer values for pad/module identification
- Current: Use a map (for the modules) of maps (for the pads):

```
map< int, ( map< int, tpcconddata::Pedestal* >)* >
      module ID          pad ID
```

- “Algorithm” modifications complete, compiles without warnings
- Design modifications still in progress:
 - Using Marlin streamlog and replacing all `cout` by `m_out(“IDENTIFIER”)`
 - Bringing all modified processors to the style (see Oliver's contribution)
 - Backwards compatibility under consideration
- Therefore not merged with trunk yet,
but working code available under
`svn://pi.physik.uni-bonn.de/MarlinTPC/branches/diener`