
Subject: CbmGenerators and MCEvent

Posted by [Volker Frieze](#) on Tue, 26 May 2009 08:55:20 GMT

[View Forum Message](#) <> [Reply to Message](#)

Latest changes in the framework offer the possibility to define experiment-specific classes for event information on the MC level.

For CBM, there is now the class CbmMCEvent (library cbmdata). It is very similar to the (still existing) class FairMCEventHeader, but adds the variable event plane angle. Further additions can be made on request.

The new class will be in the branch MCEvent. The old branch MCEventHeader (class FairMCEventHeader) is still present such as not to interfere with other fairroot users.

In the simulation macros, the instantiation of FairPrimaryGenerator must be replaced by the one of CbmPrimaryGenerator:

```
CbmPrimaryGenerator* primGen = new CbmPrimaryGenerator();
fRun->SetGenerator(primGen);
```

The proper filling of the CbmMCEvent class is in the responsibility of the concrete generator class. The PrimaryGenerator takes care for the event vertex, event time and number of primary tracks. All other variables have to be filled by the concrete classes (derived from FairGenerator). Example:

```
// ----- Public method ReadEvent -----
Bool_t CbmHsdLvmGenerator::ReadEvent(FairPrimaryGenerator* primGen) {

    ....
    ....

    // --> Convert to CbmPrimaryGenerator
    CbmPrimaryGenerator* cbmGen = (CbmPrimaryGenerator*) primGen;

    // Set event information
    CbmMCEvent* event = cbmGen->GetEvent();
    if ( ! event->IsSet() ) {
        event->SetEventID(++fCurrentEvent);
        event->SetB(b);
        event->SetPhi(0.);
    }

}
```
