

Jan 09, 15 15:41

RootEvent.hh

Page 1/1

```

#ifndef RootEvent_h
#define RootEvent_h 1
//
//
#include <vector>
#include "TObject.h"
#include "RootTrack.hh"
#include "RootVertex.hh"
#include "RootCaloHit.hh"
#include "RootTrackerHit.hh"

//
//
class RootEvent : public TObject
{
public:
    RootEvent ();
    RootEvent (int aEventID,
               std::vector<RootTrack> aTracks,
               std::vector<RootVertex> aVertex,
               std::vector<RootCaloHit> aCaloHit,
               std::vector<RootCaloHit> aVetoHit,
               std::vector<RootTrackerHit> aTkHit);

    ~RootEvent ();
    RootEvent (const RootEvent&);

    inline int EventID() {return eventID;}
    inline std::vector<RootTrack> GetTracks(){return theTrackCollection;}
    inline std::vector<RootVertex> GetVertex(){return theVertexCollection;}
    inline std::vector<RootCaloHit> GetCaloHit(){return theCaloHitCollection;}
    inline std::vector<RootCaloHit> GetVetoHit(){return theVetoHitCollection;}
    inline std::vector<RootTrackerHit> GetTrackerHit(){return theTrackerHitCollection;}

    inline void SetEventID(int aEventID) {eventID=aEventID;}
    inline void SetTracks(std::vector<RootTrack> aTracks) {theTrackCollection=aTracks;}
    inline void SetVertex(std::vector<RootVertex> aVertex) {theVertexCollection=aVertex;}
    inline void SetCaloHit(std::vector<RootCaloHit> aCaloHit) {theCaloHitCollection=aCaloHit;}
    inline void SetVetoHit(std::vector<RootCaloHit> aVetoHit) {theVetoHitCollection=aVetoHit;}
    inline void SetTrackerHit(std::vector<RootTrackerHit> aTrackerHit) {theTrackerHitCollection=aTrackerHit;}

    ClassDef(RootEvent,2);

private:
    int eventID;
    std::vector<RootTrack> theTrackCollection;
    std::vector<RootVertex> theVertexCollection;
    std::vector<RootCaloHit> theCaloHitCollection;
    std::vector<RootCaloHit> theVetoHitCollection;
    std::vector<RootTrackerHit> theTrackerHitCollection;
};

#endif

```