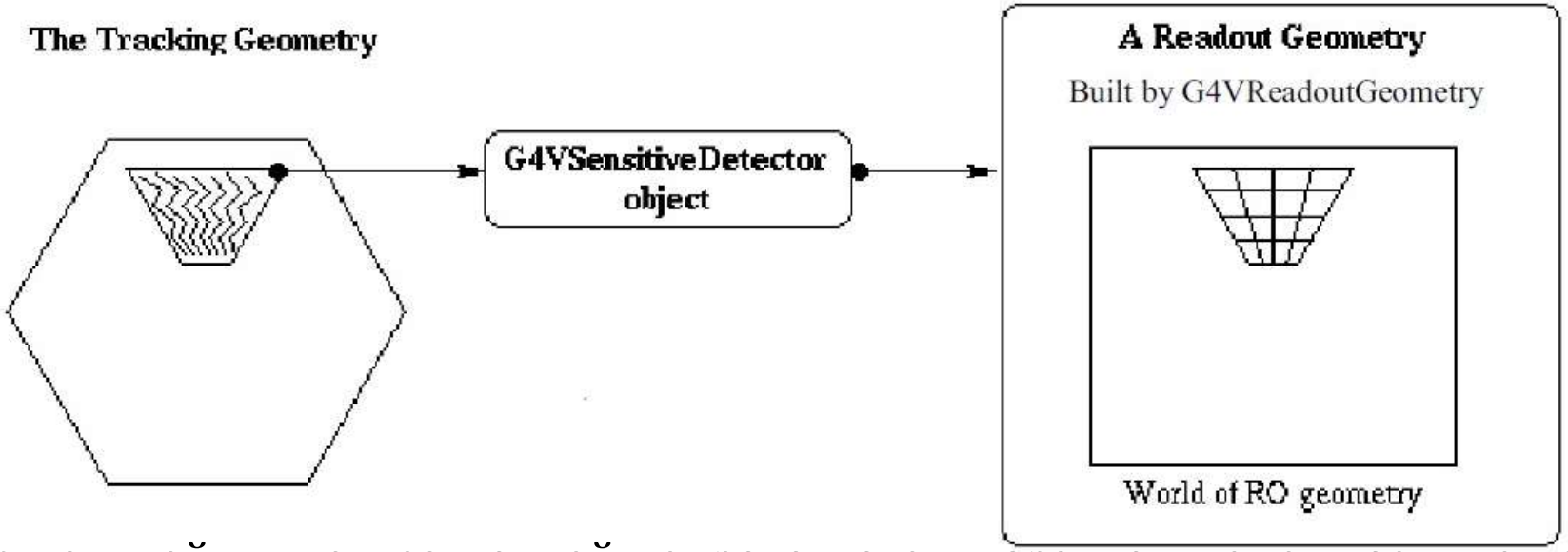
Параллельная геометрия для считывания (Readout geometry)

Больше не используется!!!

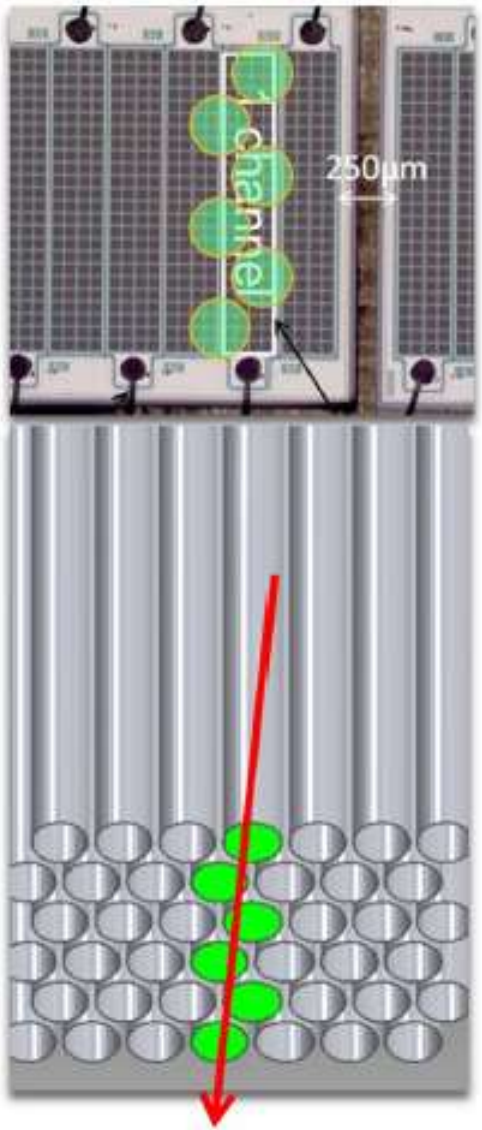
Геометрия детекторов может быть сложной и отличаться от геометрии каналов считывания сигнала.

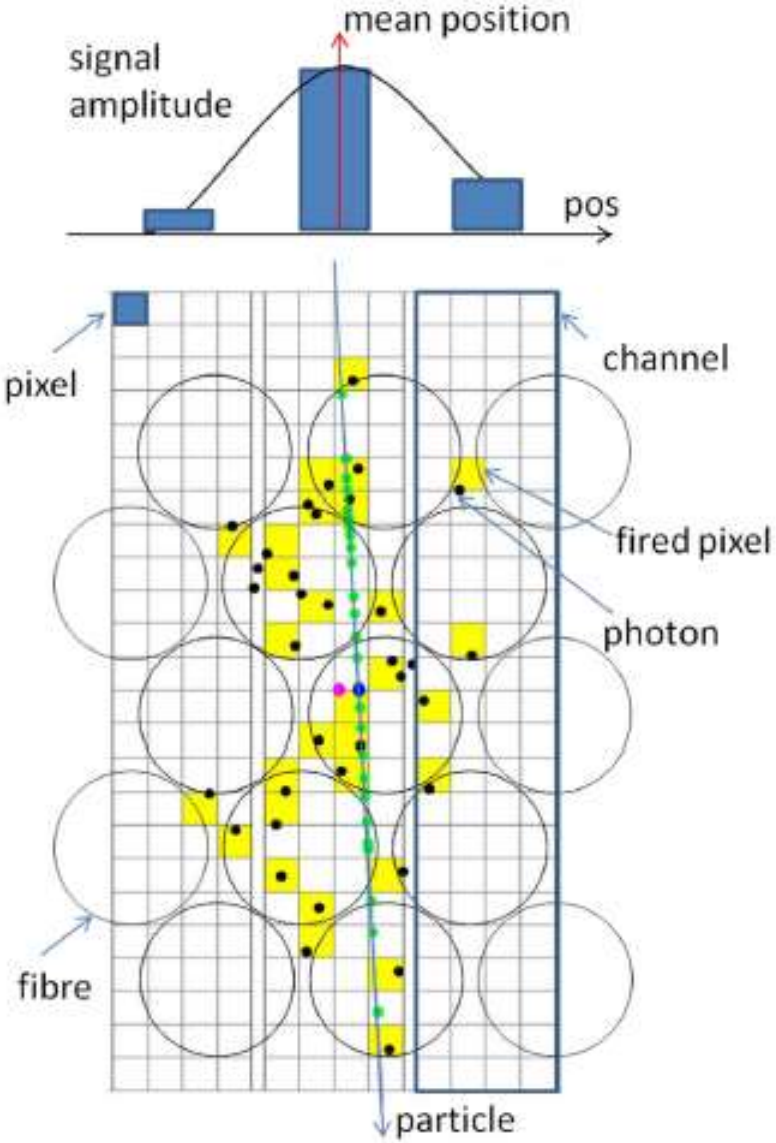


- Readout геометрия является виртуальной и искусственной, определяется параллельно существующей **реальной** геометрии.
- Трекинг частиц реализуется в **реальной** геометрии, но чувствительные детекторы имеют доступ к дополнительной геометрии для моделирования процедуры считывания (определение канала считывания для текущего отклика или хита).
- Readout геометрия должна быть связана с чувствительным детектором.
- Границы объёмов Readout геометрии не влияют на величину шага в **реальной** геометрии.

Параллельная геометрия для считывания (Readout geometry)

Трекер: сцинтилляционные файберы





Параллельная геометрия

Начиная с версии 8.2 в Geant4 появилась возможность определять несколько миров (параллельных геометрий) одновременно с использованием абстрактный класс G4VUserParrallelWorld.

G4VUserParallelWorld Class Reference

Public Member Functions

	G4VUserParallelWorld (G4String worldName)
virtual	~G4VUserParallelWorld ()
virtual void	Construct ()=0
G4String	GetName ()

Чисто виртуальный метод

Protected Member Functions

G4VPhysicalVolume *	GetWorld ()

G4VUserDetectorConstruction Class Reference

Public Member Functions

	G4VUserDetectorConstruction ()
virtual	~G4VUserDetectorConstruction ()
virtual G4VPhysicalVolume *	Construct ()=0
void	RegisterParallelWorld (G4VUserParallelWorld *)
G4int	ConstructParallelGeometries ()
G4int	GetNumberOfParallelWorld () const
G4VUserParallelWorld *	GetParallelWorld (G4int i) const

Управление трекингом в параллельной геометрии обеспечивается классом G4RunManager.

Параллельная геометрия рассматривается как клон **основной** геометрии.

Каждая параллельная геометрия должна быть зарегистрирована в объекте-наследнике абстрактного класса G4VUserDetectorConstruction

Параллельная геометрия

1. Процесс транспортировки одновременно имеет доступ ко всем параллельным мирам.

2. Размер шага ограничивается как границами объёмов в основной геометрии, так и границами объёмов в параллельных геометриях.

3. Объёмы и чувствительные детекторы в каждой параллельной геометрии определяются независимо.

4. Объёмы в разных геометриях могут пересекаться.

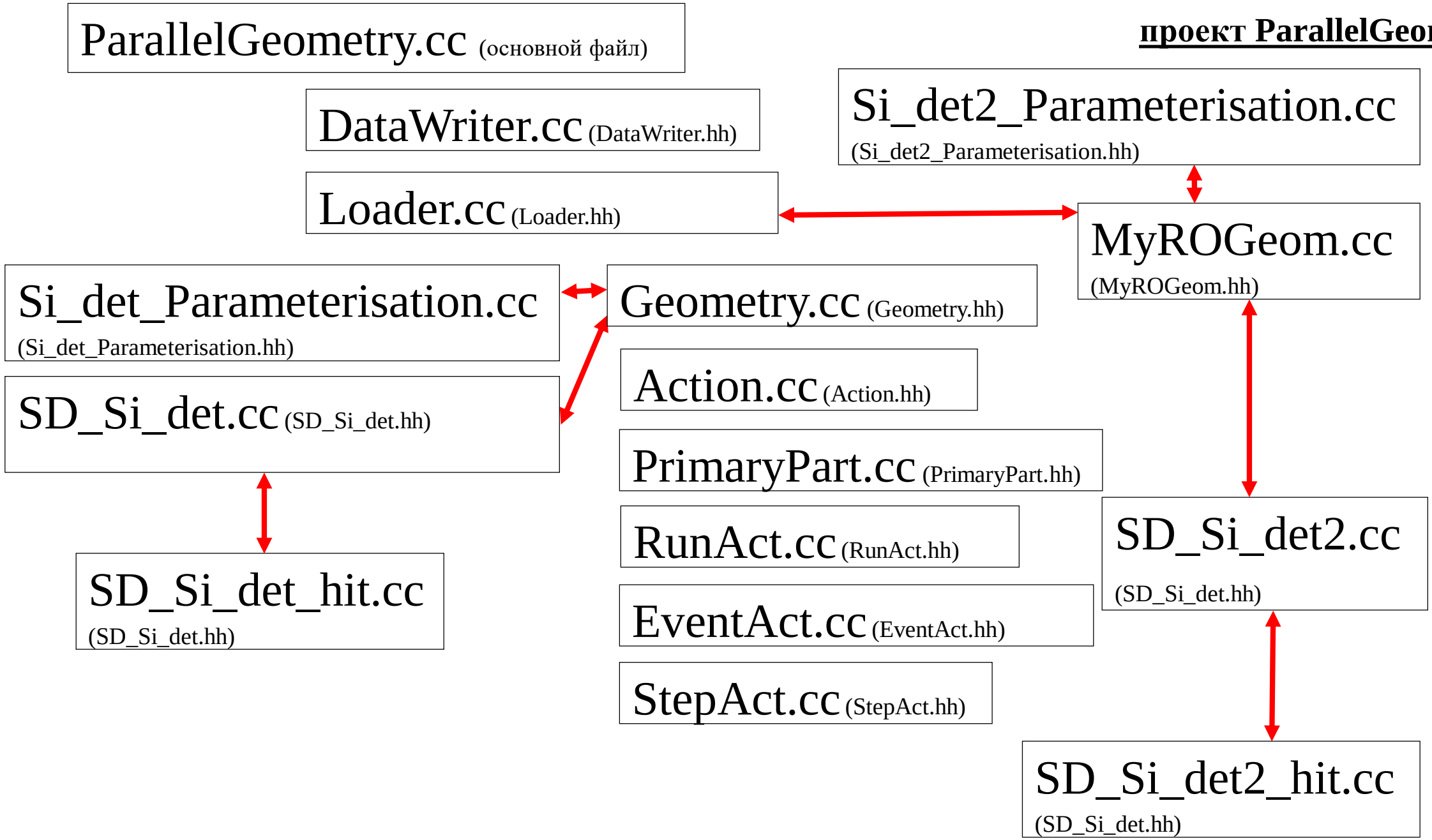
5. Материалы, пороги рождения, электромагнитные поля используются только в основной геометрии. Определения этих параметров в параллельной геометрии не учитываются при моделировании.

6. G4SteppingManager реализует моделирование только в основной геометрии. Для моделирования в параллельной геометрии необходимо к физлисту добавить объект G4ParallelWorldPhysics.

Параллельная геометрия создаётся на этапе инициализации объекта класса G4RunManager

Параллельная геометрия

проект ParallelGeometry



Параллельная геометрия

Loader.cc

```
Loader::Loader(int argc, char **argv, std::ofstream& ofsa)
{
 ……..
 G4VUserDetectorConstruction* realWorld = new Geometry(*this->ofs_sn);
 G4VUserParallelWorld* parallelWorld = new MyROGeom("parallelWorld");
 realWorld->RegisterParallelWorld(parallelWorld);
 runManager->SetUserInitialization(realWorld);

 G4VModularPhysicsList* physicsList = new QBBC;
 physicsList->RegisterPhysics(new G4ParallelWorldPhysics("parallelWorld",true));
 runManager->SetUserInitialization(physicsList);
 runManager->SetUserInitialization(new Action(*this->ofs_sn));
 runManager->Initialize();
 ……..
}
```

Определение основной геометрии realWorld в классе Geometry, наследнике класса G4VUserDetectorConstruction

Определение параллельной геометрии parallelWorld в классе MyROGeom, наследнике класса G4VUserParallelWorld

Добавление основной геометрии к менеджеру

Регистрация параллельной геометрии parallelWorld в классе основной геометрии

Многослойность

Добавление к физлисту объекта G4ParallelWorldPhysics для моделирования в параллельной геометрии

Добавление физлиста в менеджер

Добавление классов пользовательских действий в менеджер

Инициализация менеджера

Параллельная геометрия

MyROGeom.cc

*Параллельная геометрия рассматривается как клон **основной** геометрии*

Доступ к логическому объёму мира в параллельной геометрии

Своя параметризация объёма (класс Si_det2_Parameterisation) для параллельной геометрии

```
void MyROGeom::Construct()
{
 G4VPhysicalVolume* ghostWorld = GetWorld();
G4LogicalVolume* worldLogical= ghostWorld->GetLogicalVolume();
……..
G4VPVParameterisation* Si_det2 = new Si_det2_Parameterisation();
 Si_det2_pvpl = new G4PVParameterised("Si_det2", Si_det2_log, Si_log2, kXAxis, 2, Si_det2);
 G4SDManager* sdman = G4SDManager::GetSDMpointer();
 SD_Si_det2* sensitive_Si_det2 = new SD_Si_det2("/mySi_det2");
 sdman->AddNewDetector(sensitive_Si_det2);
 Si_det2_log->SetSensitiveDetector(sensitive_Si_det2);
……..
}
```

Параметризация для 2-ух объёмов

Свой чувствительный детектор (класс SD_Si_det2) для логического объёма в параллельной геометрии

Параллельная геометрия

MyROGeom.hh

```
class MyROGeom  : public G4VUserParallelWorld
{
 public:
  virtual void Construct();
```

*Нет параметров мира (клон мира **основной** геометрии)*

```
   G4Box* Si_box2;
   G4LogicalVolume* Si_log2;
   G4VPhysicalVolume*  Si_pvpl2;
   G4ThreeVector Si_vect;
   G4Box* Si_det2_box;
   G4LogicalVolume* Si_det2_log;
   G4VPhysicalVolume* Si_det2_pvpl;
   ……
}
```

Параллельная геометрия

SD_Si_det2.cc

```
G4bool SD_Si_det2 :: ProcessHits(G4Step* step, G4TouchableHistory*)
{
  G4TouchableHandle touchable = step->GetPreStepPoint()->GetTouchableHandle();
  G4int copyNo   = touchable->GetVolume(0)->GetCopyNo();
  G4double edep  = 0.;
  edep           = step->GetTotalEnergyDeposit();
  this->AddSumE(edep,copyNo);
  this->SetCopyNum(copyNo);
  SD_Si_det2_hit *aHit = new SD_Si_det2_hit();
  aHit->SetEdep(step->GetTotalEnergyDeposit());
  aHit->SetLayerNumber(step->GetPreStepPoint()->GetTouchableHandle()->GetVolume(0)->GetCopyNo());
  hitCollection->insert(aHit);
  return true;
}
```

Доступ к трековым переменным параллельного мира через класс моделирования шагов G4Step

Номер копии сработавшего детектора в параллельной геометрии

Выделившаяся в чувствительном детекторе параллельного мира энергия на текущем шаге

Сохранение информации для параллельной геометрии в переменных хита

Добавление хита в коллекцию

Параллельная геометрия

SD_Si_det2.cc

```
void SD_Si_det2 :: EndOfEvent(G4HCofThisEvent* HCE)
{
 G4int i; for (i=0; i<2; i++)
 {hit_SD_Si_det[i] << std::setw(10) << this->GetSumE(i) << std::setw(10) << this->GetCopyNum(i) << G4endl;}
 for (i=0; i<2; i++) {SumE[i]=0.;}
 G4SDManager* SDman= G4SDManager::GetSDMpointer();
 G4int hitsCollID= SDman->GetCollectionID("SD_Si_det2_hitCollection");
 SD_Si_det2_hitCollection* THC = NULL;
 THC = (SD_Si_det2_hitCollection*)(HCE->GetHC(hitsCollID));
 if (THC)
 {
  G4int n_hit=THC->entries();
  for (i=0; i<n_hit; i++)
  {
   SD_Si_det2_hit* hit = (*THC)[i];
   G4cout<<"collection ID="<<hitsCollID<<" "<<THC->GetName()<<" "<<THC->GetSDname()<<" "<<hit->GetEdep()
         <<" "<<hit->GetLayerNumber()<<G4endl;
  }
 }
}
```

Определение ID коллекции хитов для данного события по имени

Доступ к коллекции хитов по ID

Извлечение хита из коллекции

Из класса для чувствительного детектора параллельной геометрии вывод на экран информации о хитах

Параллельная геометрия

Event_Act.cc

```
void EventAct::EndOfEventAction(const G4Event *EVE)
{
 ……
 G4SDManager* SDman= G4SDManager::GetSDMpointer();
 G4int hitsCollID = SDman->GetCollectionID("SD_Si_det_hitCollection");
 SD_Si_det_hitCollection* THC = NULL;
 G4HCofThisEvent* HCE = EVE->GetHCofThisEvent();
 if (HCE)  {THC = (SD_Si_det_hitCollection*)(HCE->GetHC(0)); }
 if (THC)
 {
 G4int n_hit=THC->entries();
 G4cout<<"EventID="<<EVE->GetEventID()<<G4endl;
 for (G4int i=0; i<n_hit; i++)
 {
  SD_Si_det_hit* hit = (*THC)[i];
  G4cout<<"collection ID="<<hitsCollID<<" "<<THC->GetName()<<" "<<THC->GetSDname()<<
          " "<<hit->GetEdep()<<" "<<hit->GetLayerNumber()<<G4endl;
 }
 }
}
```

Определение ID коллекции хитов для данного события по имени

Доступ к коллекции хитов для данного события

Доступ к коллекции хитов с ID=0

Извлечение хита из коллекции

*Из класса событий вывод на экран информации о хитах в **основной** геометрии*