New library structure of AdePT

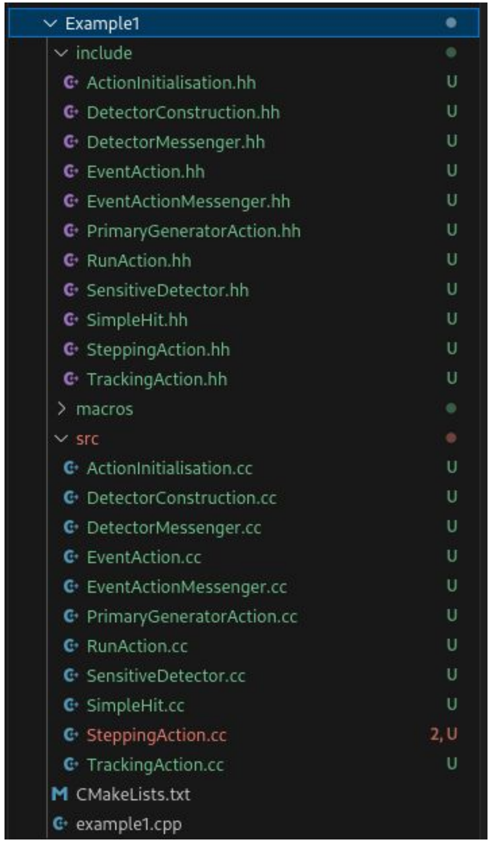
Previous AdePT examples

- Little shared code
- Creating a new example involved copying most files from the one used as a starting point
- Functionality was spread among the user actions
- The AdePT core classes depended on user-defined code

New library structure

- All “Pure AdePT” code has been made common and moved out of the examples
- All AdePT configuration is now centralized in a single class
- Single entry point for the user code: A G4 application only needs to register the AdePT physics list in order to use the GPU
- Code will be compiled into three libraries that examples will need to link to:
  - AdePT-G4_interface
  - AdePT
  - AdePT_cuda

Overview of an AdePT example using the new library



```
Example1
  include
    ActionInitialisation.hh          U
    DetectorConstruction.hh          U
    DetectorMessenger.hh             U
    EventAction.hh                   U
    EventActionMessenger.hh          U
    PrimaryGeneratorAction.hh        U
    RunAction.hh                     U
    SensitiveDetector.hh             U
    SimpleHit.hh                     U
    SteppingAction.hh                U
    TrackingAction.hh                U
  macros
  src
    ActionInitialisation.cc          U
    DetectorConstruction.cc          U
    DetectorMessenger.cc             U
    EventAction.cc                   U
    EventActionMessenger.cc          U
    PrimaryGeneratorAction.cc        U
    RunAction.cc                     U
    SensitiveDetector.cc             U
    SimpleHit.cc                     U
    SteppingAction.cc              2, U
    TrackingAction.cc                U
  CMakeLists.txt
  example1.cpp
```

- AdePT examples are now pure Geant4 applications with the exception of registering the AdePT Physics list

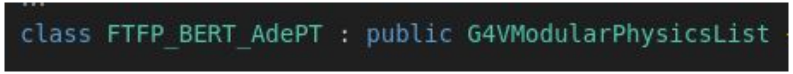
- Scoring done on the Host by default, transparent for the user, the only requirement is that some SensitiveDetector code is registered

Overview of an AdePT example using the new library

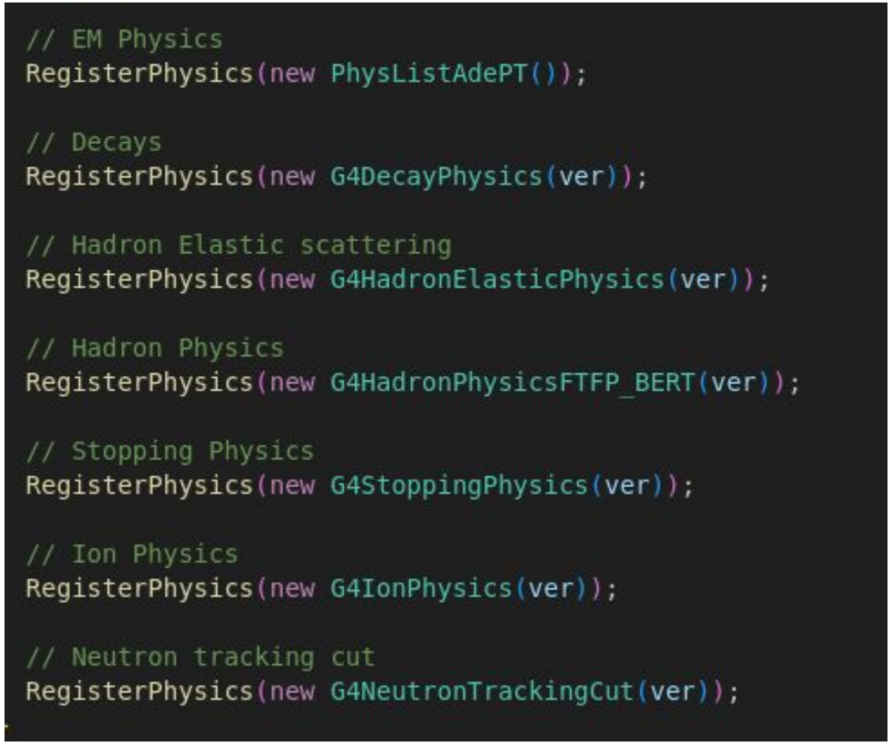
- The user needs to register PhysListAdePT for EM Physics

- This is the only mandatory AdePT dependency a G4 application needs to have

- As an example we provide the FTFP_BERT_AdePT modular physics list



```
class FTFP_BERT_AdePT : public G4VModularPhysicsList
```



```
// EM Physics
RegisterPhysics(new PhysListAdePT());

// Decays
RegisterPhysics(new G4DecayPhysics(ver));

// Hadron Elastic scattering
RegisterPhysics(new G4HadronElasticPhysics(ver));

// Hadron Physics
RegisterPhysics(new G4HadronPhysicsFTFP_BERT(ver));

// Stopping Physics
RegisterPhysics(new G4StoppingPhysics(ver));

// Ion Physics
RegisterPhysics(new G4IonPhysics(ver));

// Neutron tracking cut
RegisterPhysics(new G4NeutronTrackingCut(ver));
```