Electromagnetic and hadronic physics in Geant4

Luciano Pandola
INFN-LNGS

Queen's University, Belfast (UK), January 24, 2013

Based on a presentation by G.A.P. Cirrone (INFN-LNS)

Outline

- The philosophy of the physics definition
- How to define and activate models
- Electromagnetic physics
- Hadronic physics

Philosophy

- Provide a **general model framework** that allows the implementation of **complementary/alternative models** to **describe the same process** (e.g. Compton scattering)
  - A certain model could work better in a certain energy range
- **Decouple** modeling of cross sections and of final state generation
- Provide **processes** containing
  - Many possible models and cross sections
  - Default cross sections for each model

Models under continuous development

G4VUserPhysicsList

- All physics lists **must** derive from this class
  - And then be registered to the G4RunManager
  - Mandatory class in Geant4

```
class MyPhysicsList: public G4VUserPhysicsList {
public:
MyPhysicsList();
~MyPhysicsList();
void ConstructParticle();
void ConstructProcess();
void SetCuts();
}
```

- User must implement the following (purely virtual) methods:
  - **ConstructParticle(), ConstructProcess(), SetCuts()**

ConstructParticle()

- Choose the particles you need in your simulation and define all of them here
  - G4Electron::ElectronDefinition()
  - G4Gamma::GammaDefinition()
  - …
- It is possible use Geant4 classes that create groups of particles
  - G4BosonConstructor()
  - G4LeptonConstructor()
  - …

SetCuts()

- Define all production cuts for gamma, electrons and positrons
  - Recently also for protons
- Notice: this is a **production cut**, not a tracking cut
  - All particles, once created, are tracked down to zero kinetic energy
  - The cut is used to limit the generation of secondaries (e.g. δ-rays from ionization, or gammas from bremsstrahlung)
  - The cut is expressed in equivalent range
    - This is converted in energy for each material

Physics definition

- Different ways to implement the physics models

1. Explicitly associating a given model to a given particle for a given energy range
   - Error prone
   - Done at code level (requires C++ coding)
2. Use of **BUILDER** and **REFERENCE PHYSICS LISTS**
   - The BUILDERS are process-related (standard, lowenergy, Bertini, etc.)
     - **Building blocks** to be used in a physics list
     - Allows **mix-and-match** done by the user
   - THE REF PHYSICS LISTS are complete physics lists
     - Can be instantiated by UI (macro files)

Builder with the G4VModularPhysicsList

- It is used to build a realistic physics list which would be too long and complicated with the previous approach
- It is derived from **`G4VUserPhysicsList`**
- **`AddTransportation()`** automatically called
- Allows the definition of **"physics modules"** for a given process
  - Electromagnetic
  - Hadronic
  - Decay
  - Optical physics
  - Ion physics

Reference physics lists

- Provide a complete and realistic physics with ALL models of interest
- Provided according to some use-cases
- Few choices are available for EM physics
- Several possibilities for hadronic
- They are intended as starting point and their builders can be reused
  - They are made up of builders, so easy to change/replace each given block

Reference physics lists

- These families share components to attach certain types of processes to groups of particles. These components are:
  - electromagnetic interactions for all particles. Different settings are offered:
    - Default transport parameters (best performance)
    - Some optimised choice (_**EMV** extension)
    - Some high precision choice (_**EMY** extension)
  - Inelastic interactions
  - Elastic scattering
  - Capture
  - Decay of unstable particles
  - Specialised treatment of low energy neutrons (< 20 MeV)

How to build a modular physics list

- Create a class derived by **`G4VModularPhysicsList`**
  - **`class myList : public G4VModularPhysicsList`**
- Implement the mandatory methods **`ConstructParticle()`** and **`ConstructProcess()`** and use the appropriate builders (or create your own)

```
void myList::ConstructProcess()
{
  AddTransportation();
  //Em physics
  G4VPhysicsConstructor* emList = new G4EmStandardPhysics();
  emList->ConstructProcess();
  //Inelastic physics for protons
  G4VPhysicsConstructor* pList = new G4QGSPProtonBuilder();
  pList->ConstructProcess();
```

How to use a Geant4 physics list

- In your main(), just register an instance of the physics list to the **G4RunManager**

```
#include "QGSP_BERT.hh"
int main()
{
  // Run manager
  G4RunManager * runManager = new G4RunManager();

  …
 G4VUserPhysicsList* physics = new QGSP_BERT();
  runManager-> SetUserInitialization(physics);
}
```

The complete lists of Reference Physics List

......./**source/physics_lists/lists**

```
-rw-r--r--  1 cirrone  staff   4102 16 Aug 09:14 QGSP_BERT_EMV.icc
-rw-r--r--  1 cirrone  staff   2564 11 May  2009 QGSP_BERT_EMX.hh
-rw-r--r--  1 cirrone  staff   4232 16 Aug 09:14 QGSP_BERT_EMX.icc
-rw-r--r--  1 cirrone  staff   2542 31 Oct  2006 QGSP_BERT_HP.hh
-rw-r--r--  1 cirrone  staff   4322 16 Aug 09:14 QGSP_BERT_HP.icc
-rw-r--r--  1 cirrone  staff   2586 17 Oct  2008 QGSP_BERT_NOLEP.hh
-rw-r--r--  1 cirrone  staff   4224 16 Aug 09:14 QGSP_BERT_NOLEP.icc
-rw-r--r--  1 cirrone  staff   2580 26 Apr  2007 QGSP_BERT_NQE.hh
-rw-r--r--  1 cirrone  staff   4240 16 Aug 09:14 QGSP_BERT_NQE.icc
-rw-r--r--  1 cirrone  staff   2557  7 May  2007 QGSP_BERT_TRV.hh
-rw-r--r--  1 cirrone  staff   4236 16 Aug 09:14 QGSP_BERT_TRV.icc
-rw-r--r--  1 cirrone  staff   2496 31 Oct  2006 QGSP_BIC.hh
-rw-r--r--  1 cirrone  staff   4578 16 Aug 09:14 QGSP_BIC.icc
-rw-r--r--  1 cirrone  staff   2552 11 May  2009 QGSP_BIC_EMY.hh
-rw-r--r--  1 cirrone  staff   4176 16 Aug 09:14 QGSP_BIC_EMY.icc
-rw-r--r--  1 cirrone  staff   2550 24 Nov  2006 QGSP_BIC_HP.hh
-rw-r--r--  1 cirrone  staff   4140 16 Aug 09:14 QGSP_BIC_HP.icc
-rw-r--r--  1 cirrone  staff   2563 13 Nov  2007 QGSP_DIF.hh
-rw-r--r--  1 cirrone  staff   4317 16 Aug 09:14 QGSP_DIF.icc
-rw-r--r--  1 cirrone  staff   2502 31 Oct  2006 QGSP_EMV.hh
-rw-r--r--  1 cirrone  staff   4822 16 Aug 09:14 QGSP_EMV.icc
-rw-r--r--  1 cirrone  staff   2541 26 Apr  2007 QGSP_EMV_NQE.hh
-rw-r--r--  1 cirrone  staff   4260 16 Aug 09:14 QGSP_EMV_NQE.icc
-rw-r--r--  1 cirrone  staff   2582 23 Apr  2009 QGSP_FTFP_BERT.hh
-rw-r--r--  1 cirrone  staff   4174 16 Aug 09:14 QGSP_FTFP_BERT.icc
-rw-r--r--  1 cirrone  staff   3499 19 Jul  2009 QGSP_INCL_ABLA.hh
-rw-r--r--  1 cirrone  staff   4262 16 Aug 09:14 QGSP_INCL_ABLA.icc
-rw-r--r--  1 cirrone  staff   2528 26 Apr  2007 QGSP_NQE.hh
-rw-r--r--  1 cirrone  staff   4234 16 Aug 09:14 QGSP_NQE.icc
-rw-r--r--  1 cirrone  staff   2523 28 Nov  2006 QGSP_QEL.hh
-rw-r--r--  1 cirrone  staff   4413 16 Aug 09:14 QGSP_QEL.icc
-rw-r--r--  1 cirrone  staff   2507 13 Nov  2007 QGS_BIC.hh
-rw-r--r--  1 cirrone  staff   4188 16 Aug 09:14 QGS_BIC.icc
-rw-r--r--  1 cirrone  staff   2521  8 Jun 18:05 Shielding.hh
-rw-r--r--  1 cirrone  staff   4113 16 Aug 09:14 Shielding.icc
-rw-r--r--  1 cirrone  staff   3710 31 Oct  2006 SpecialCuts.hh
lists Lavora! > _
```

Electromagnetic physics

EM concept - 1

- The same physics processes (e.g. Compton scattering) can be described by different models, that can be **alternative** or **complementary** in a given energy range
- For instance: Compton scattering can be described by
  - **`G4KleinNishinaCompton`**
  - **`G4LivermoreComptonModel`** (specialized low-energy, based on the Livermore database)
  - **`G4PenelopeComptonModel`** (specialized low-energy, based on the Penelope analytical model)
  - **`G4LivermorePolarizedComptonModel`** (specialized low-energy, Livermore database with polarization)
  - **`G4PolarizedComptonModel`** (Klein-Nishina with polarization)
- Different models can be combined, so that the appropriate one is used in each given energy range (→ performance optimization)

EM concept - 2

- A physical interaction or process is described by a process class
  - Naming scheme : « G4ProcessName »
  - Eg. : « G4Compton » for photon Compton scattering
- A physical process can be simulated according to several models, each model being described by a model class
  - The usual naming scheme is: « G4ModelNameProcessNameModel »
  - Eg. : « G4LivermoreComptonModel » for the Livermore Compton model
  - Models can be alternative and/or complementary on certain energy ranges
  - Refer to the Geant4 manual for the full list of available models

Packages overview

- Models and processes for the description of the EM interactions in Geant4 have been grouped in several packages

Package	Description
Standard	γ-rays, $e^{\pm}$ up to 100 TeV, Hadrons, ions up to 100 TeV
Muons	Muons up to 1 PeV
X-rays	X-rays and optical photon production
Optical	Optical photons interactions
High-Energy	Processes at high energy (> 10 GeV). Physics for exotic particles
Low-Energy	Specialized processes for low-energy (down to 250 eV), including atomic effects
Polarization	Simulation of polarized beams

EM processes for γ-rays, $e^{\pm}$

Particle	Process	G4Process
Photons	Gamma Conversion in $e^{\pm}$	`G4GammaConversion`
	Compton scattering	`G4ComptonScattering`
	Photoelectric effect	`G4PhotoElectricEffect`
	Rayleigh scattering	`G4RayleighScattering`
$e^{\pm}$	Ionisation	`G4eIonisation`
	Bremsstrahlung	`G4eBremsstrahlung`
	Multiple scattering	`G4eMultipleScattering`
e^{+}	Annihilation	`G4eplusAnnihilation`

EM processes muons

Particle	Process	G4Process

$\mu^{\pm}$	Ionisation
	
	Bremsstrahlung
	
	Multiple scattering
	
	$e^{\pm}$ pair production
		`G4MuIonisation`
		
		`G4MuBremsstrahlung`
		
		`G4MuMultipleScattering`
		
		`G4MuPairProduction`

Only one model available for these processes (but in principle users may write their own processes, if needed)

When/why to use Low Energy Models

- **Use** Low-Energy models (Livermore or Penlope), as an *alternative* to Standard models, when you:
  - need precise treatment of EM showers and interactions at low-energy (keV scale)
  - are interested in atomic effects, as fluorescence x-rays, Doppler broadening, etc.
  - can afford a more CPU-intensive simulation
  - want to cross-check an other simulation (e.g. with a different model)
- Do **not** use when you are interested in EM physics > MeV
  - same results as Standard EM models, performance penalty

Example: PhysicsList, γ-rays

Only PostStep

```
G4ProcessManager* pmanager =
   G4Gamma::GetProcessManager();
pmanager->AddDiscreteProcess(new G4PhotoElectricEffect);
pmanager->AddDiscreteProcess(new G4ComptonScattering);
pmanager->AddDiscreteProcess(new G4GammaConversion);
pmanager->AddDiscreteProcess(new G4RaleighScattering);
```

- Use **`AddDiscreteProcess`** because γ-rays processes have **only PostStep actions**
- For each process, the **default model** is used among all the available ones (e.g. **`G4KleinNishinaCompton`** for **`G4ComptonScattering`**)

EM Physics Constructors for Geant4 9.6 - ready-for-the-use

G4EmStandardPhysics – default
G4EmStandardPhysics_option1 – HEP fast but not precise
G4EmStandardPhysics_option2 – Experimental
G4EmStandardPhysics_option3 – medical, space
G4EmStandardPhysics_option4 – optimal mixture for precision
G4EmLivermorePhysics
G4EmLivermorePolarizedPhysics
G4EmPenelopePhysics
G4EmDNAPhysics

Combined Physics
Standard > 1 GeV
LowEnergy < 1 GeV

- **`$G4INSTALL/source/physics_list/builders`**
- Advantage of using of these classes – they are tested on regular basis and are used for regular validation

How to extract Physics ?

- Possible to retrieve physics quantities via **`G4EmCalculator`** or directly from the physics models
  - Physics List should be initialized
- Example for retrieving the **total cross section** (cm^{-1}) of a process with name *procName*: for particle *partName* and material *matName*

```
G4EmCalculator emCalculator;
G4Material* material =
  G4NistManager::Instance()->FindOrBuildMaterial(“matName);
G4double massSigma = emCalculator.ComputeCrossSectionPerVolume
   (energy,particle,procName,material);
G4cout << G4BestUnit(massSigma, "Surface/Volume") << G4endl;
```

A good example: $G4INSTALL/examples/extended/ electromagnetic/TestEm14

Hadronic physics

Hadronic Physics

- Data-driven models
- Parametrised models
- Theory-driven models

Hadronic physics challenge

- Three energy regimes
  - < 100 MeV
  - resonance and cascade region (100 MeV - 10 GeV)
  - > 20 GeV (QCD strings)
- Within each regime there are several models
- Many of these are phenomenological

Reference physics lists for Hadronic interactions

- Are part of the Geant4 code
- Four families of lists
  - **LHEP**, parameterised modelling of hadronic interactions
    - Based on the old GEISHA package
  - **QGS**, or list based on a model that use the Quark Gluon String model for high energy hadronic interactions of protons, neutrons, pions and kaons
  - **FTF**, based on the FTF (FRITIOF like string model) for protons, neutrons, pions and kaons
  - **Other** specialized physics lists

Hadronic processes

- At rest
  - Stopped muon, pion, kaon, anti-proton
  - Radioactive decay
  - Particle decay (decay-in-flight is PostStep)
- Elastic
  - **Same process** to handle all long-lived hadrons (multiple models available)
- Inelastic
  - **Different processes** for each hadron (possibly with multiple models vs. energy)
  - Photo-nuclear, electro-nuclear, mu-nuclear
- Capture
  - Pion- and kaon- in flight, neutron
- Fission

Cross sections

- Default cross section sets are provided for each type of hadronic process:
  - Fission, capture, elastic, inelastic
- Can be **overridden** or **completely replaced**
- Different types of cross section sets:
  - Some contain only a few numbers to parameterize cross section
  - Some represent large databases (data driven models)
- Cross section management
  - `GetCrossSection()` → sees last set loaded for energy range

Alternative cross sections

- To be used for specific applications, or for a given particle in a given energy range, for instance:
- Low energy neutrons
  - elastic, inelastic, fission and capture (< 20 MeV)
- Neutron and proton inelastic cross sections
  - 20 MeV < E < 20 GeV
- Ion-nucleus reaction cross sections (several models)
  - Good for E/A < 1 GeV
- Isotope production data
  - E < 100 MeV
- Photo-nuclear cross sections

Information on the available cross sections at **`http://geant4.cern.ch/support/proc_mod_catalog/cross_sections/`**

Thermal neutron scattering

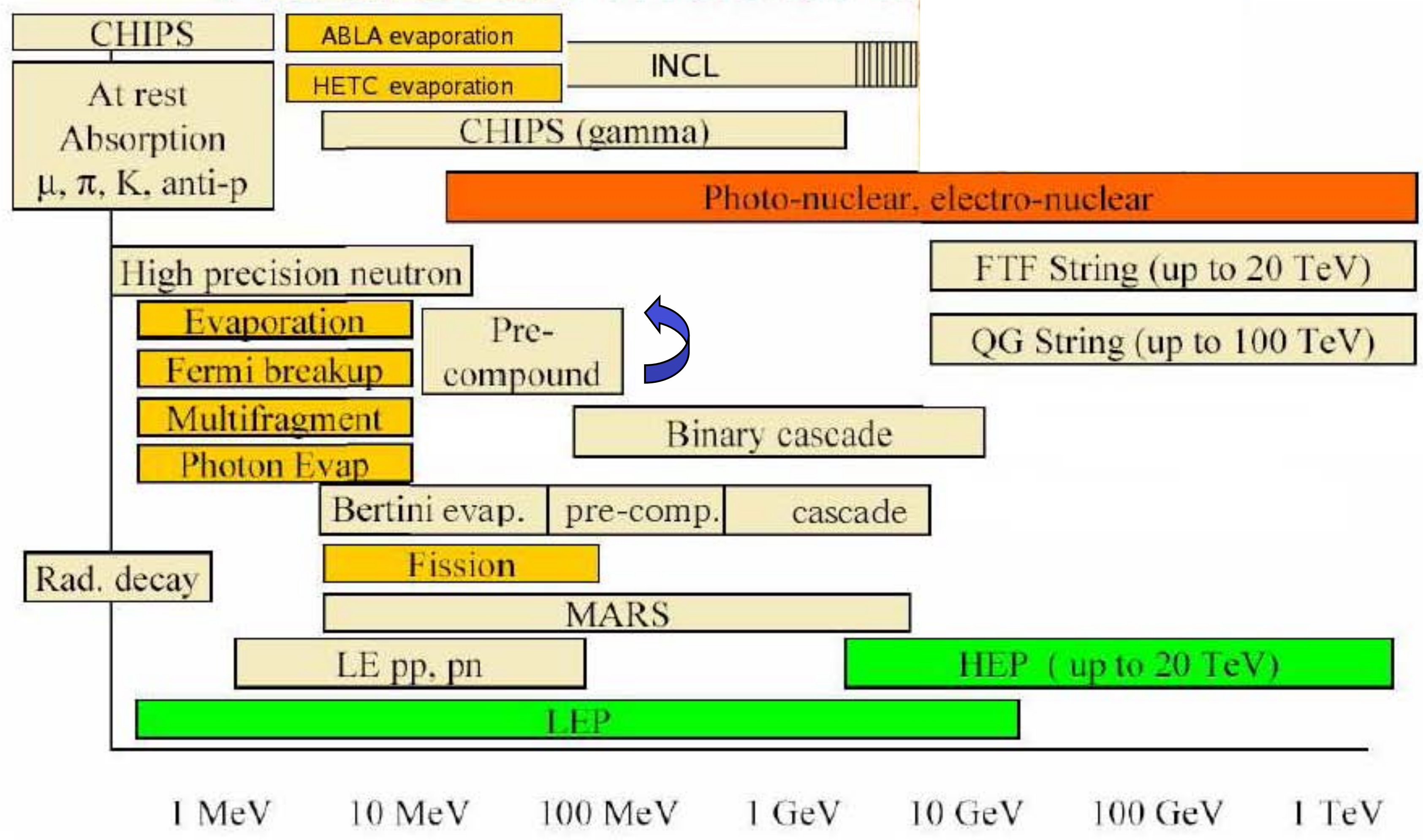
- At thermal neutron energies, atomic translational motion as well as vibration and rotation of the chemically bound atoms affect the **neutron scattering cross section** and the energy and angular distribution of secondary neutrons
- The energy loss or gain of incident neutrons can be different from interactions with nuclei in unbound atoms
- Only individual Maxwellian motion of the target nucleus (Free Gas Model) was taken into account in the default NeutronHP models

Neutron HP Models

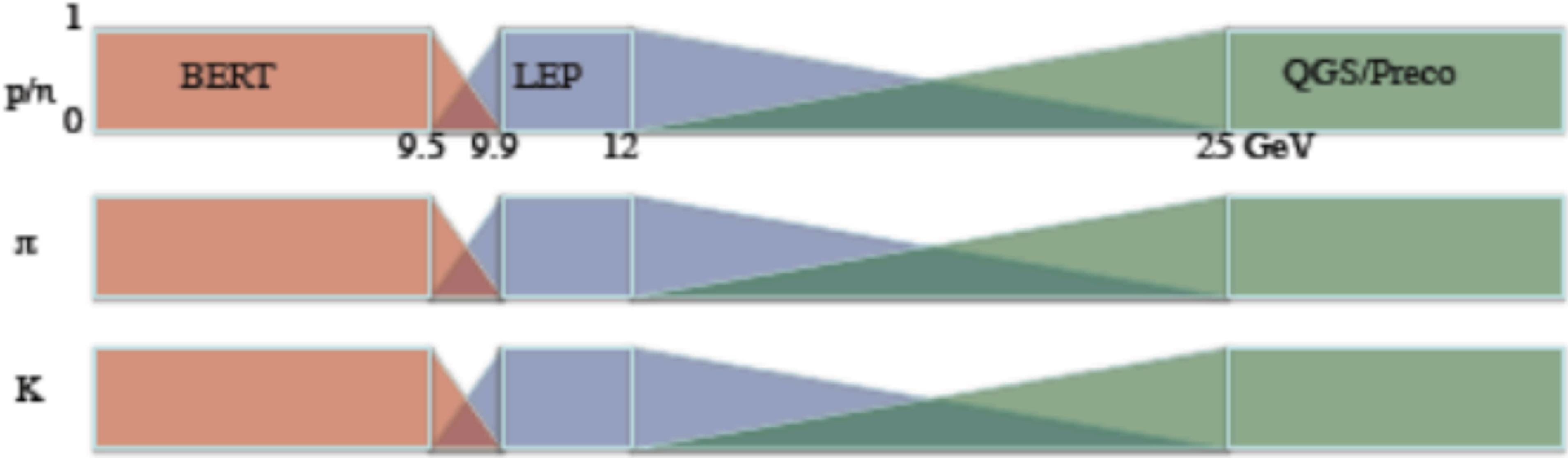
- Transport of **low-energy neutrons** in matter:
  - The energy coverage of these models is from thermal energies to 20 MeV
  - The modeling is based on the data formats of ENDF/B-VI, and all distributions of this standard data format are implemented
  - Includes cross sections and final state information for *elastic* and *inelastic scattering, capture*, *fission* and *isotope production*
  - The file system is used in order to allow granular access to, and flexibility in, the use of the cross-sections for different isotopes, and channels
  - Code in sub-directory: **`/source/processes/hadronic/models/neutron_hp`**

Hadronic model inventory

http://geant4.cern.ch/support/proc_mod_catalog/models



Hadronic models match – inelastic interactions



Recommended reference physics lists

- A dedicated web page

> Info to help users to choose the proper physics list:
>
> **http://geant4.cern.ch/support/proc_mod_catalog/physics_lists/physicsLists.shtml**

- Application fields are identified
  - High energy physics
  - LHC neutron fluxes
  - Shielding
  - Medical
  - ...

Where to find information?

User Support

1. Getting started
2. Training courses and materials
3. Source code
   a. Download page
   b. LXR code browser -or- draft doxygen documentation
4. Frequently Asked Questions (FAQ)
5. Bug reports and fixes
6. User requirements tracker
7. User Forum
8. Documentation
   a. Introduction to Geant4
   b. Installation Guide
   c. Application Developers Guide
   d. Toolkit Developers Guide
   e. Physics Reference Manual
   f. Software Reference Manual
9. Physics lists
   a. Electromagnetic
   b. Hadronic

Code Example (1/2)

```
G4ParticleDefinition* proton=
     G4Proton::ProtonDefinition();
G4ProcessManager* protonProcessManager =
     proton->GetProcessManager();
```

retrieve the process manager for proton

```
// Elastic scattering
G4HadronElasticProcess* protonElasticProcess =
 new G4HadronElasticProcess();
```

create the process for elastic scattering

```
G4LElastic* protonElasticModel =
 new G4LElastic();
```

*get the LE parametrized **model** for elastic scattering*

```
protonElasticProcess->
 RegisterMe(protonElasticModel);
```

***register** the model to the process*

```
protonProcessManager->
 AddDiscreteProcess(protonElasticProcess);
```

attach the process to proton

Code example (2/2)

```
...
// Inelastic scattering
G4ProtonInelasticProcess* protonInelasticProcess
  = new G4ProtonInelasticProcess();
```

*creates the **process** for inelastic scattering*

Model 1

```
G4LEProtonInelastic* protonLEInelasticModel
 = new G4LEProtonInelastic();
protonLEInelasticModel->
   SetMaxEnergy(20.0*GeV);
```

*gets the LEP **model** up to 20 GeV*

```
protonInelasticProcess->
   RegisterMe(protonLEInelasticModel);
```

registers LEP model to the process

Model 2

```
G4HEProtonInelastic* protonHEInelasticModel =
   new G4HEProtonInelastic();
protonHEInelasticModel->SetMinEnergy(20.0*GeV);
```

*gets the HEP **model** from 20 GeV*

```
protonInelasticProcess
   ->RegisterMe(protonHEInelasticModel);
```

registers HEP model to the process

Quick overview of validation

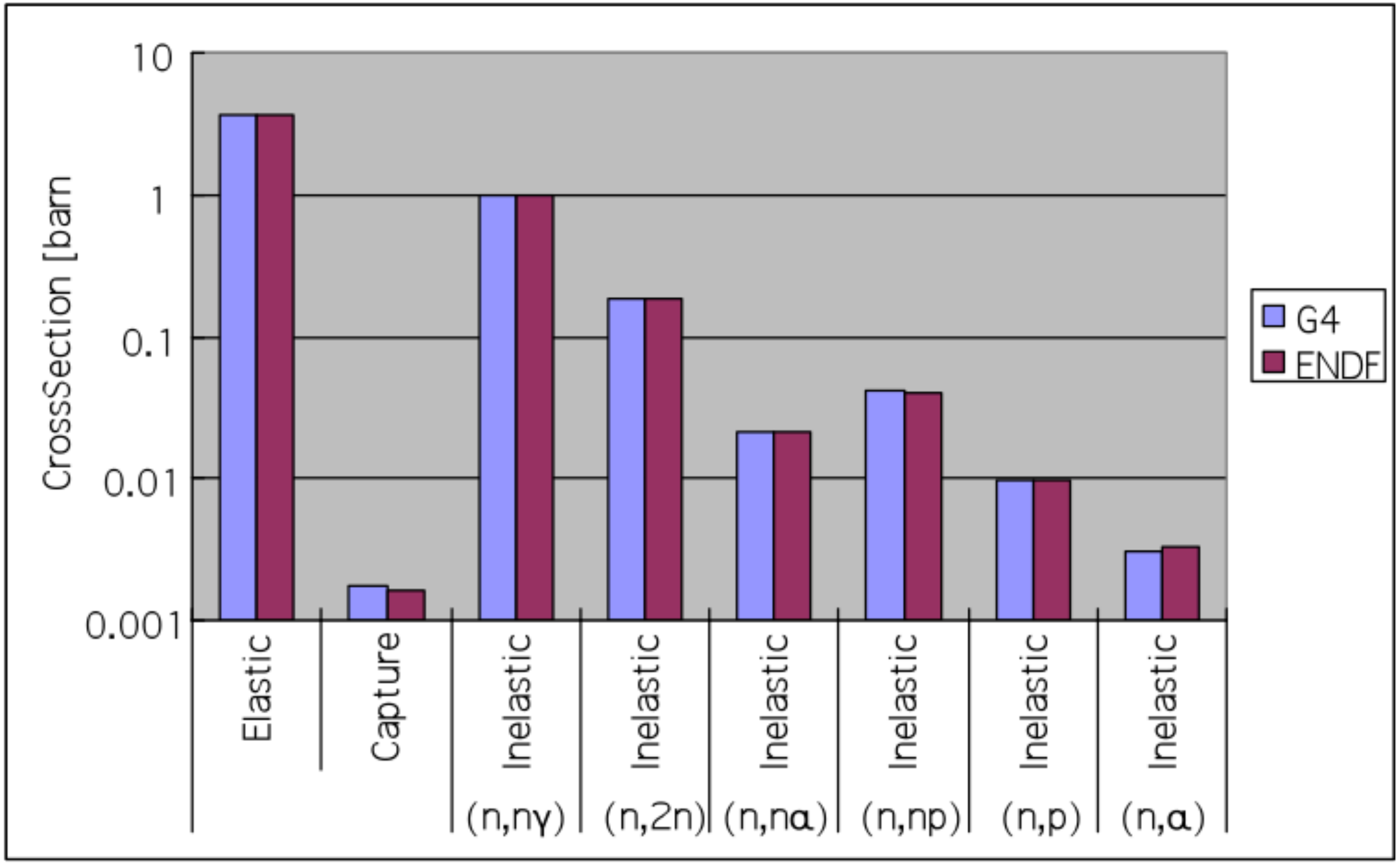
Hadronic validation

- A website is available to collect relevant information for validation of Geant4 hadronic models (plots, tables, references to data and to models, etc.)

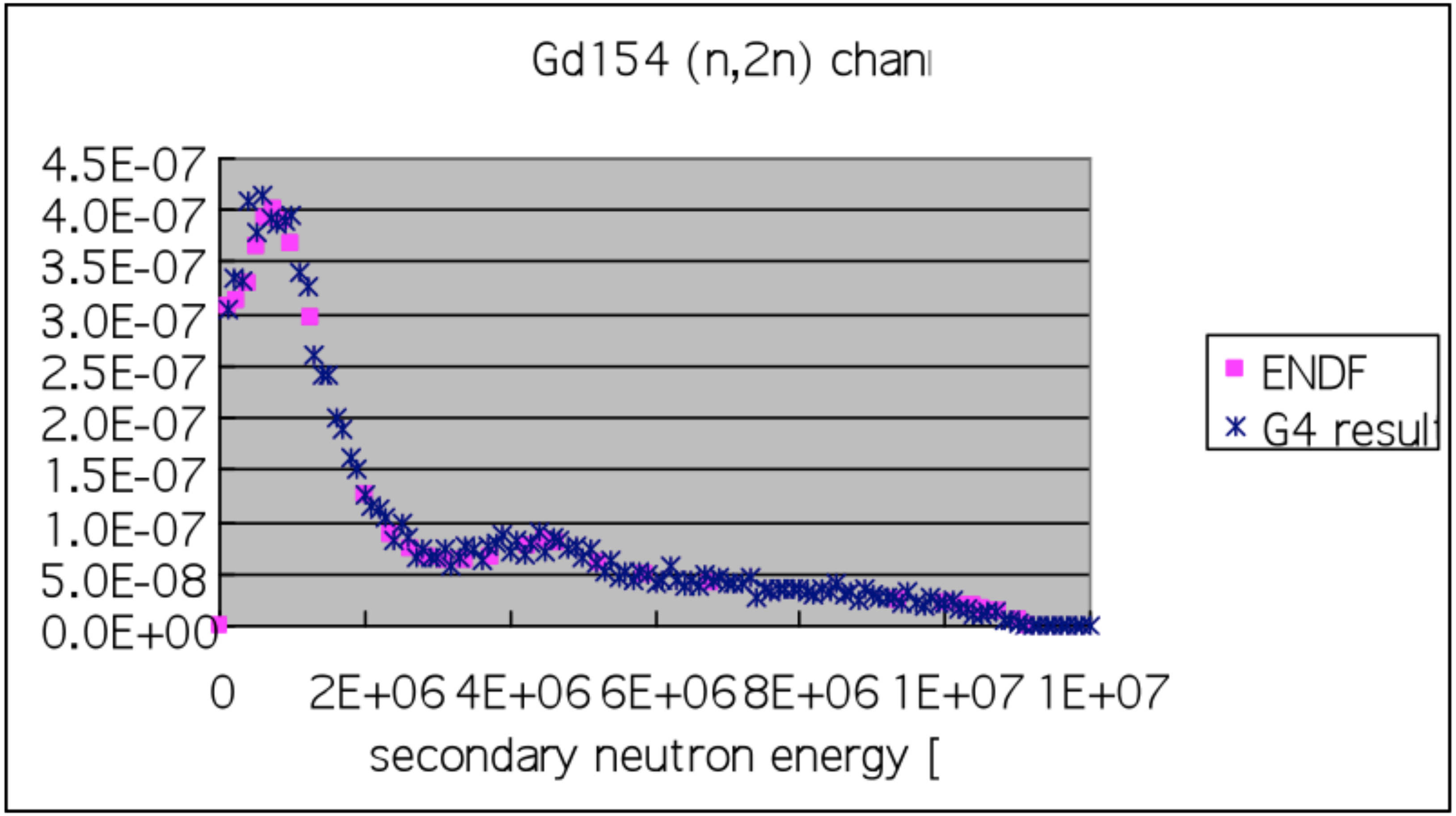
  `http://geant4.fnal.gov/hadronic_validation/validation_plots.htm`

- Several physics lists and several use-cases have been considered (e.g. thick target, stopped particles, low-energy)
- Includes final states and cross sections

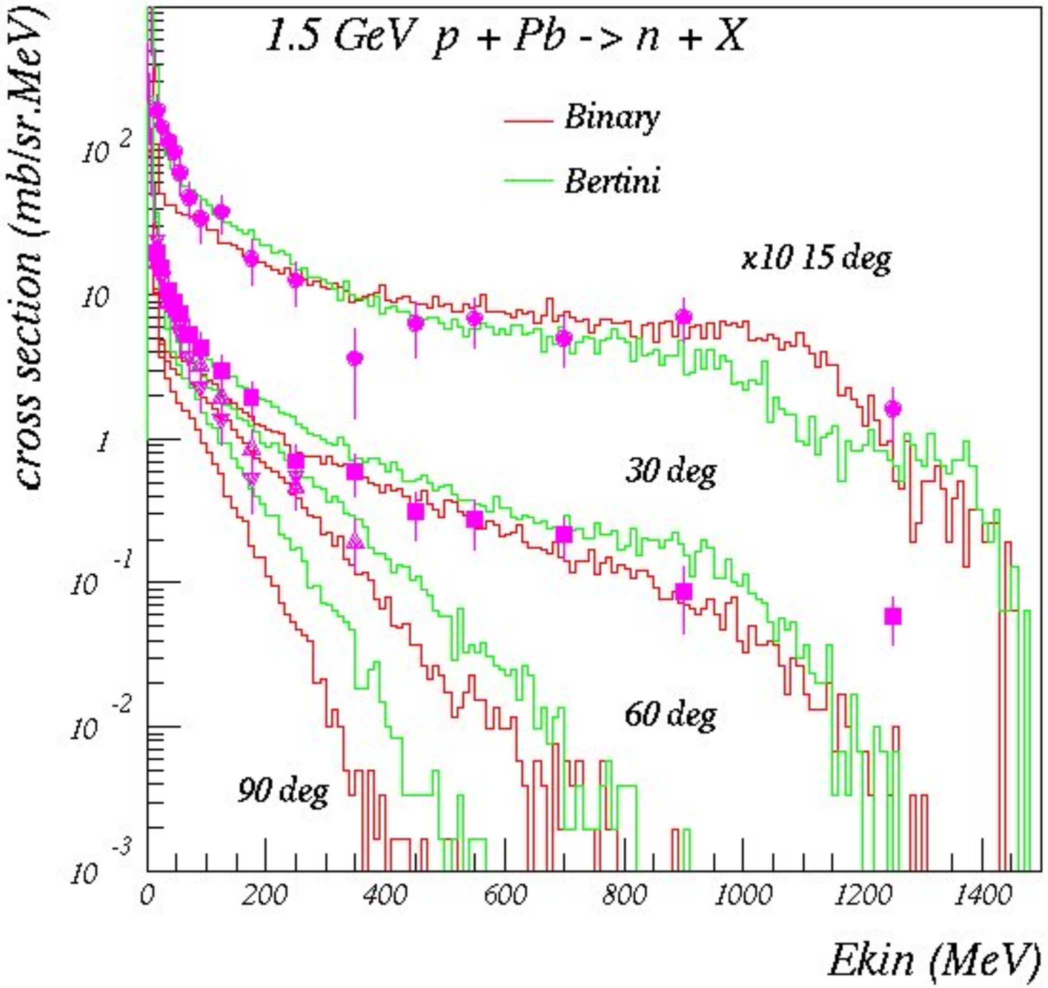
Some verification: channel cross section



Some verification: secondary energy spectrum

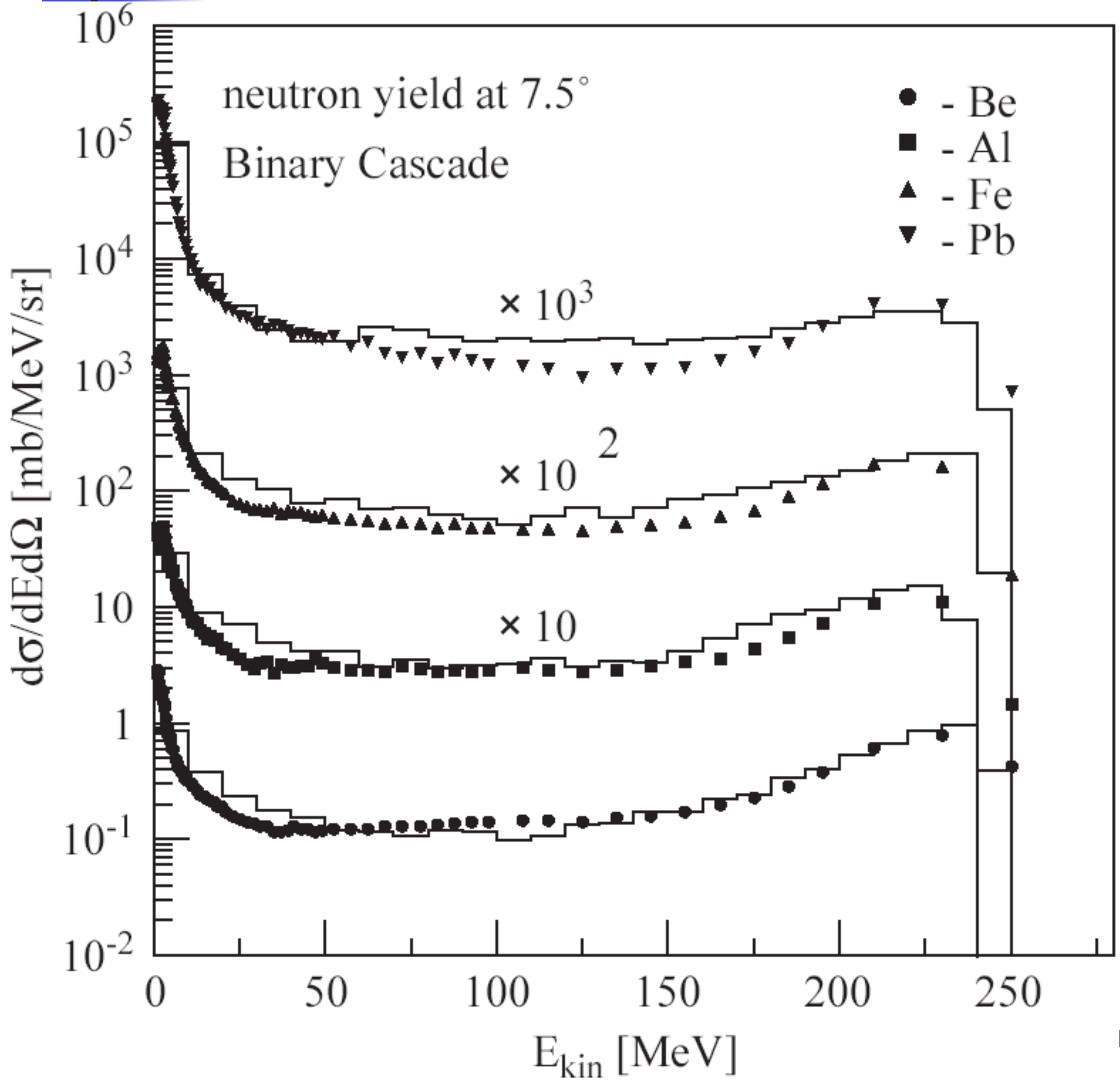


Nuclear fragmentation



Bertini and **Binary cascade** models: neutron production vs. angle from 1.5 GeV protons on Lead

Neutron production by protons



Binary cascade model: double differential cross-section for neutrons produced by 256 MeV protons impinging on different targets