Choosing a Physics List

L.G. Sarmiento Pico

Lund University, Sweden
(Some slides/inspiration from Dennis Wright/Alberto Ribbon/Mihaly Novak)

12th International Geant4 Tutorial in Korea, Pohang 2025.

What is a Physics List?

The object responsible for **specifying all particles to be used** in a simulation application and **specifying physics processes** and **assigning them to each particle type**

- ▶ Remember that is one of three mandatory objects for any `Geant4` application.
- ▶ Provides a very flexible way to set up the physics environment
  - ▶ user can choose and specify particles he wants
  - ▶ user can choose the physics (processes) to assign to each particle
- ▶ BUT, user must have a good understanding of the physics required to describe the problem. **There is no default**, `FTFP_BERT` kind of is, but not really.
  - ▶ omission of relevant particles and/or physics interactions could lead to poor modeling results
- ▶ As we have seen during the week, Geant4 takes an atomistic, rather than an integral approach to physics. **The user is fully responsible for getting the physics right.**

Why do we need to set up our own Physics List?

Nature has just one *physics*: so why `Geant4` does not provide a complete and unique set of particles and physics processes that everyone can use? shouldn't it?

NO!

▶ t**here are many different approximations and models to describe the same interaction**
  ▶ very much the case for hadronic but also for electromagnetic physics
▶ **computation time might the designing principle**
  ▶ some users may want a less accurate but significantly faster model for a given interaction while others need the most accurate description regardless of CPU time
▶ there is likely no simulation application that would require all the particles and all the possible interactions that `Geant4` can provide
  ▶ e.g. most medical/nuclear applications are not interested in multi-GeV physics

In the Physics Overview talk we learned how to create our own Physics List.
BUT **does every user have to reinvent the wheel?**

Reference Physics Lists

There exists a number of **Production physics lists** that are packaged by large user groups like ATLAS, CMS, ... Medical Physics Community, Space Domain, ...

▶ because of their importance, **they are well-maintained**

▶ they are changed/updated less frequently → expected to be **stable**

▶ they are **regularly validated** by the developers and the user communities

There are at least 36 different reference lists to choose from, however, remember:

▶ these lists are provided as a *best guess* of the physics needed in some given use cases

▶ when a user decide to use them, **the user is responsible** for *validating* the physics for that given application

▶ they can be used directly or as a starting point or template for the user's own physics list

▶ this means the user is allowed/encourage to customize, add, or remove, physics components, using the settings they consider suitable for their application.

Reference Physics Lists: naming convention[1]

Some Hadronic options, parenthesis indicates commonly optional

QGS	Quark Gluon String mode ($> 15\,\mathrm{GeV}$)
FTF	FRITIOF String mode ($> 5\,\mathrm{GeV}$)
(P)	G4Precompound model used for de-excitation
BIC	Binary Cascade model ($< 10\,\mathrm{GeV}$)
BERT	Bertini Cascade model ($< 10\,\mathrm{GeV}$)
INCLXX	INCL++ Liège IntraNuclear Cascade ($< 10\,\mathrm{GeV}$)
(All)HP	High Precision mode ($< 20\,\mathrm{MeV}$). All uses the TENDL database
LEND	Low Energy Neutron Dataset ($< 20\,\mathrm{MeV}$)
(T)	... adding neutron thermal scattering ($< 20\,\mathrm{MeV}$)

Some EM options. See later, and EM Physics talk for more details

–	No suffix: standard EM i.e. the default G4EmStandardPhysics constructor (option 0)
EMV	G4EmStandardPhysics_option1, fast but less precise
EMY	G4EmStandardPhysics_option3, rather precise
EMZ	G4EmStandardPhysics_option4, most precise

[1] complete documented list of Physics Lists in the Guide for Physics Lists (link)

Reference Physics Lists: naming convention[1]

Some Hadronic options, parenthesis indicates commonly optional

QGS	Quark Gluon String mode ($> 15\,\text{GeV}$)
FTF	FRITIOF String mode ($> 5\,\text{GeV}$)
(P)	G4Precompound model used for de-excitation
BIC	Binary Cascade model
BERT	Bertini Cascade
INCLXX	INCL++ Liège
(All)HP	High Precision
LEND	Low Energy
(T)	... adding neutron thermal scattering ($< 20\,\text{MeV}$)

Decoding scheme:

String(Pre?)_Cascade_Neutron_EM

Some EM options. See later, and EM Physics talk for more details

–	No suffix: standard EM i.e. the default G4EmStandardPhysics constructor (option 0)
EMV	G4EmStandardPhysics_option1, fast but less precise
EMY	G4EmStandardPhysics_option3, rather precise
EMZ	G4EmStandardPhysics_option4, most precise

[1] complete documented list of Physics Lists in the Guide for Physics Lists (link)

Reference Physics Lists: naming convention[1]

Physics List Factory

- Users can use *G4PhysicsListFactory* which knows about all the available reference physics lists and extensions →

```
// Get QGSP_BIC_HP_EMZ physicslist using G4PhysicsListFactory
G4PhysicsListFactory physicsFactory;
const G4String plName = "QGSP_BIC_HP_EMZ";
auto physicsList =
```

- Alternatively, buil

  strings of referen

  and a step limit t

```
// Add Optical and StepLimit to the default physics list
G4PhysicsListFactoryAlt physicsFactory;
const G4String plName = "FTFP_BERT+OPTICAL+STEPLIMIT";
auto physicsList = physicsFactory.GetReferencePhysList(plName);
```

- Users may build a reference physics list with custom physics constructors! ← add/remove or replace physics constructors

  – G4VModularPhysicsList::RegisterPhysics(G4VPhysicsConstructor*)

  – G4VModularPhysicsList::ReplacePhysics(G4VPhysicsConstructor*)

Decoding scheme:

String(Pre?)_Cascade_Neutron_EM

- FTFP_BERT_HP
- FTFP_BERT_TRV
- FTFP_BERT_ATL
- FTFQGSP_BERT
- FTFP_INCLXX
- FTFP_INCLXX_HP
- FTF_BIC
- LBE
- QBBC
- QGSP_BERT
- QGSP_BERT_HP
- QGSP_BIC
- QGSP_BIC_AllHP
- QGSP_FTFP_BERT
- QGSP_INCLXX
- QGSP_INCLXX_HP
- QGS_BIC
- Shielding
- ShieldingLEND
- ShieldingM
- NuBeam
- FTFP_BERT_HPT
- FTFP_INCLXX_HPT
- QGSP_BERT_HPT
- QGSP_BIC_HPT
- QGSP_INCLXX_HPT
- Shielding_HPT
- ShieldingM_HPT

⊕

- EM0
- EMV
- EMX
- EMY
- EMZ
- LIV
- PEN
- GS
- SS
- WVI
- LE
- HPT

Choosing your physics list:

Recommendation:

- ▶ Ideal situation: the user(s) have a good understanding of the physics relevant for a given application
  - ▶ the user can either build its own physics list or decide to use a pre-defined one
  - ▶ the chosen physics list needs to be validated for the given application
  - ▶ during the validation procedure, some parts of the physics list might be changed add physics, remove physics, change settings, etc.
- ▶ The given application belongs to a well defined application area (e.g. medical applications)
  - ▶ the user can choose the reference physics list recommended for the given application area as a staring point
  - ▶ the chosen physics list needs to be validated for the given application (same as above)
- ▶ Something that always works (but time consuming)
  - ▶ the user can take the most accurate physics settings (e.g. opt4 for EM)
  - ▶ run some simulation with lower statistics to obtain the most accurate result
  - ▶ then the user can take a less accurate but fast physics setting (e.g. opt0 for EM) as a starting point and obtain some simulation results
  - ▶ then granularly extend the initial physics list by using the accurate results as reference

`FTFP_BERT` (link)

Recommended for collider physics applications. It usually produces the best agreement with test beam calorimeter data, including shower shape, energy response and resolution.

It is also recommended for cosmic ray applications where good treatment of very high energy particles is required. Note, however, that is not suited to very high energy collisions of order $10\,\mathrm{TeV}$ or more.

See also: `FTFP_BERT_HP`, `FTFP_BERT_HPT`, `FTFP_BERT_ATL`, `FTFP_BERT_TRV`, `FTFP_INCLXX`, `FTFP_INCLXX`, `FTFP_INCLXX_HP`, `FTFP_INCLXX_HPT`, `FTFQGS_BERT`, `FTF_BIC`, *Electromagnetic options*.

---

[2] Once the Geant4/PYTHIA interface is released, Geant4 will be valid beyond its current $10\,\mathrm{TeV}$ limit

QBBC (QGSC[2] + BinaryCascade + Bertini) (link)

Recommended for applications where accurate simulation for low-energy transport of protons and neutrons is needed. It usually produces the best agreement in the energy range below $1\,\mathrm{GeV}$ for thin target experiments. For higher energies it is the same as the default FTFP_BERT physics. **It is recommended for medical and space applications.**

See also: QBBC_ABLA, *Electromagnetic options*.

---

[2]**CH**iral **I**nvariant **P**hase **S**pace is a 3D low energy parton model instead of the "usual" precompound (P)
[3]Once the Geant4/PYTHIA interface is released, Geant4 will be valid beyond its current $10\,\mathrm{TeV}$ limit

`QGSP_BERT` (link)

It can be used for collider physics applications, as an **alternative to the recommended physics list** `FTFP_BERT`.
It is can also be used for cosmic ray applications where good treatment of very high energy particles is required. Note, however, that is not suited to very high energy collisions of order $10\,\mathrm{TeV}$ or more.

See also: `QGSP_BERT_HP`, `QGSP_BERT_HPT`, `QGSP_FTFP_BERT`, `QGSP_INCLXX_HP`, `QGSP_INCLXX_HPT`, `QGSP_INCLXX`, *Electromagnetic options*.

---

[2]Once the Geant4/PYTHIA interface is released, Geant4 will be valid beyond its current $10\,\mathrm{TeV}$ limit

QGSP_BIC (link)

It can be used for collider physics applications, as an **alternative to the recommended physics list** FTFP_BERT.

It is can also be used for cosmic ray applications where good treatment of very high energy particles is required. Note, however, that is not suited to very high energy collisions of order 10 TeV or more.

See also: QGSP_BIC_HP, QGSP_BIC_HPT, QGSP_BIC_AllHP, QGSP_BIC_AllHPT, *Electromagnetic options*.

---

[2]Once the Geant4/PYTHIA interface is released, Geant4 will be valid beyond its current 10 TeV limit

Shielding (link)

It is recommended for simulation of deep shielding. Neutrons of $20\,\mathrm{MeV}$ and lower use the High Precision neutron models and cross sections to describe elastic and inelastic scattering, capture and fission. The `G4NDL` database is required for this physics list.

See also: `Shielding_HP` (an alias to itself), `ShieldingM`, `ShieldingM_HP`, `ShieldingM_HPT`, `ShieldingLEND`, *Electromagnetic options*.

---

[2]Once the Geant4/PYTHIA interface is released, Geant4 will be valid beyond its current $10\,\mathrm{TeV}$ limit

Electromagnetic options – again

Some EM options. See EM Physics talk for more details

- No suffix: standard EM i.e. the default `G4EmStandardPhysics` constructor (option 0). Recommended EM option for HEP applications (sampling calorimeters, . . .)

EMV `G4EmStandardPhysics_option1` → HEP, fast but less precise. Recommended EM option for HEP applications (homogeneous calorimeters, . . .)

EMY `G4EmStandardPhysics_option3` → medical, space sci., precise. Recommended EM option for medical applications (proton/ion therapy, . . .)

EMZ `G4EmStandardPhysics_option4` → most precise EM physics. The most accurate EM physics description. Can be used as a reference for all the cases above.

Last but not least, Do not forget about cuts!

- ▶ Geant4 uses secondary production thresholds in length
  - ▶ The default value is $0.7\,\mathrm{mm}$ for reference physics lists
  - ▶ UI commands to select cuts:
    - ▶ `/run/setCut 0.1 mm`
    - ▶ `/run/setCutForAGivenParticle e- 0.1 mm`
  - ▶ Internally, cuts are converted to energy values depending on the material and particle type. The corresponding energy has a minimum value (default is 1 keV) and the user can set it with UI command:
    - ▶ `/cuts/setLowEdge 100 eV`
  - ▶ γ production thresholds are used in bremsstrahlung and e^- are used in ionization
  - ▶ e^- production thresholds are almost always ignored as they annihilate and produce two photons that can fly away. They are used only for very high cuts on γ (at least higher than the electron mass, which is the minimal energy of each of the two photon generated)
  - ▶ Proton production thresholds mean that if a hadron scatters elastically on a nucleus (of the material), the (recoiling) nucleus becomes a new `G4Track` only if its kinetic energy is above the value $100\,\mathrm{KeV} \times$ *production-cut-in-mm*