

New Novice & Extended Examples

I. Hrivnacova, IPN Orsay

16th Geant4 Collaboration Meeting, 19 - 23 September, SLAC

Outline

- Motivation
- Novice examples
 - Classification of features present in novice examples
 - New novice examples
 - Documentation
- Extended examples
 - Repository with common classes
 - First new and re-written extended examples
- Outlook

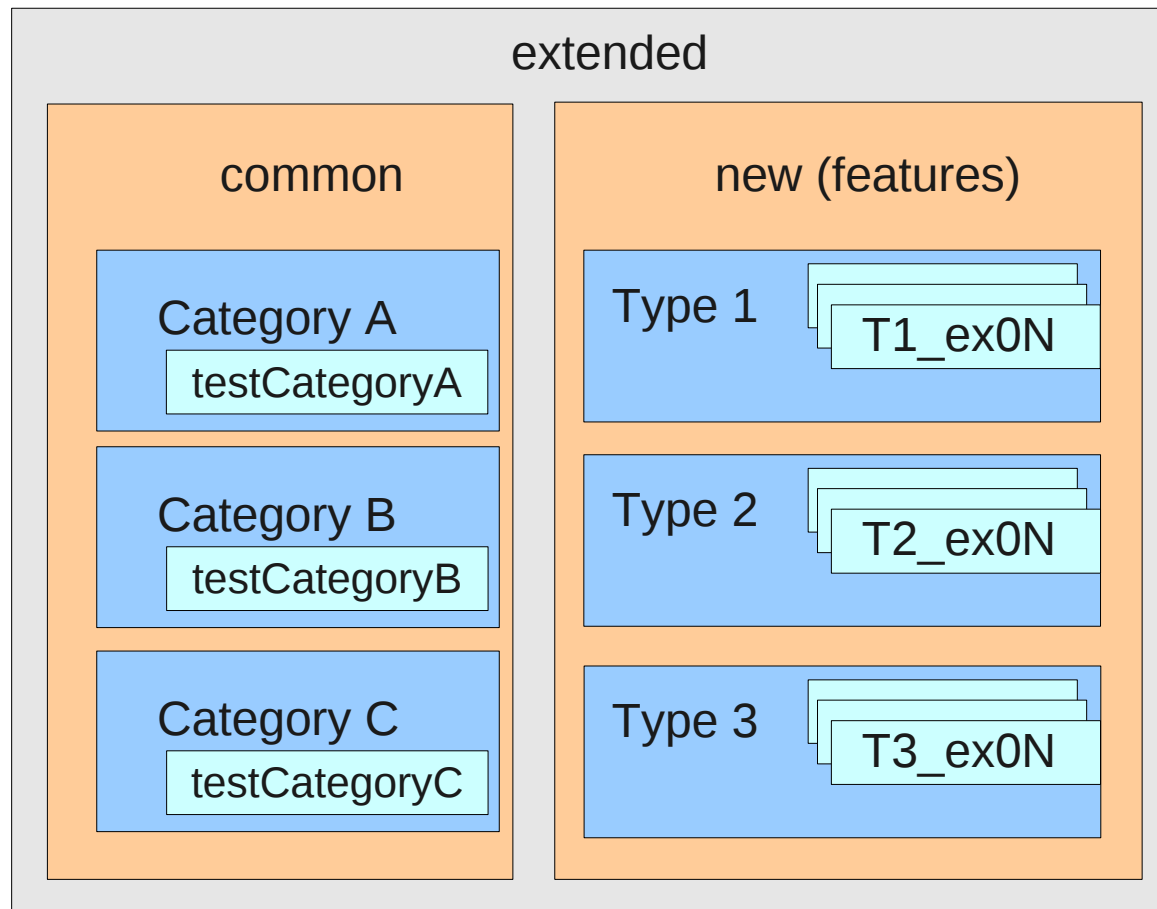
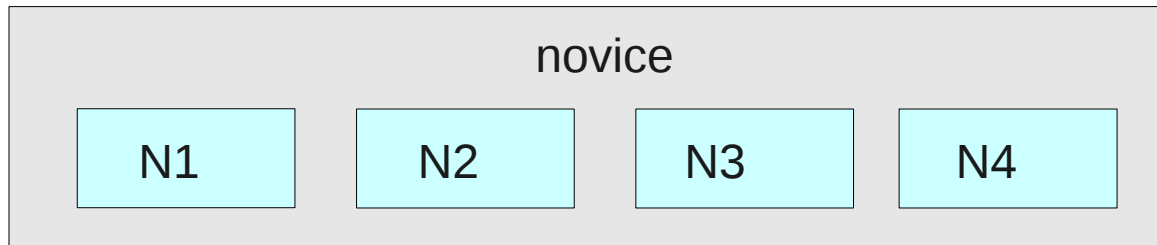
Motivation

- Novice examples:
 - Too heavy - some of them maybe too difficult for novice users
 - Can be reduced: in number of examples and also in complexity
- Extended examples
 - No code sharing – many code duplications: the same classes are cut/pasted from one example to another
 - The requirement for stand-alone examples makes more difficult for the user to find the important parts of the code for a demonstrated feature
 - Eg. extended/field/field03 example demonstrates definition of magnetic field on **18 classes** while only **3 are relevant** to magnetic field
 - With code sharing: we can reduce the code size to be maintained and underline the features which are demonstrated

Novice & Extended Examples WG

- New WG was proposed at the SB meeting on 3rd February
- I.H. (coordinator), P. Gumplinger (deputy coordinator)
- Members:
 - All coordinators of the other G4 working groups (12 people)
 - Example code developers: M. Maire, P. Guye
 - In total the group has 16 members
- Most communications/decisions via mail
- Meetings:
 - Phone meeting on 1st June
 - WG session on Monday this week at SLAC

New Concept



Novice & Extended Examples Coding Guidelines

1. Name conventions

1.1. Classes in common use names starting with the prefix 'EG4'; if there are more than one classes of the same name, then all classes names are followed by a number: (00), 01, 02, 03 ..., The number 00 should be used for helper classes which are not supposed to be directly usable in user application, eg. physics list with geantino only.

1.2. Classes in extended examples (features) use names with a prefix, specific to the demonstrated feature and different from the one reserved for classes in common, or names without a prefix.

1.3. Class member functions start with an upper case letter.

1.4. Class data members start with a prefix "f" followed with an upper case letter. This convention makes easier to understand the code.

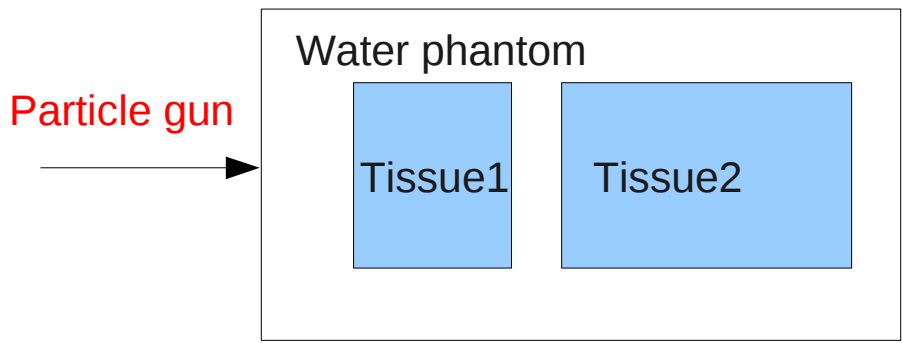
1.5. Local variables and functions argument names start with a lower case letter except for the names starting with ...

Novice Examples

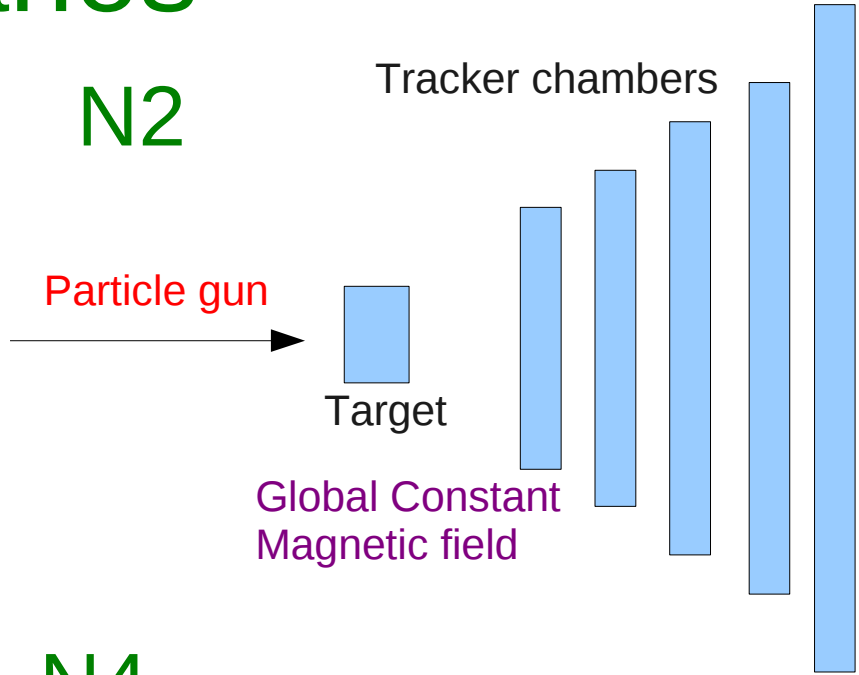
- Revision of the current novice examples:
 - A document with a summary of features demonstrated:
 - http://geant4.web.cern.ch/geant4/collaboration/working_groups/novice_extended_examples/novice_examples_features.pdf
 - Classification of the features: whether they should stay in novice or should be moved in extended examples
 - Two versions of the document, the later one:
 - http://geant4.web.cern.ch/geant4/collaboration/working_groups/novice_extended_examples/novice_examples_features_classification_v2.pdf
 - “New novice examples group”:
 - I.H., Peter Gumplinger, Michel Maire and Paul Guye

Novice Examples Scenarios

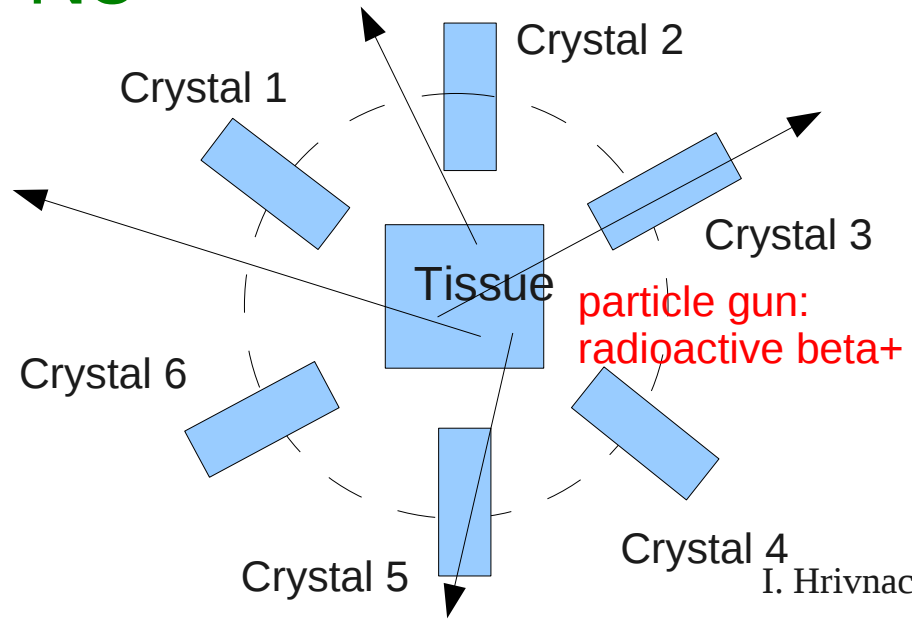
N1



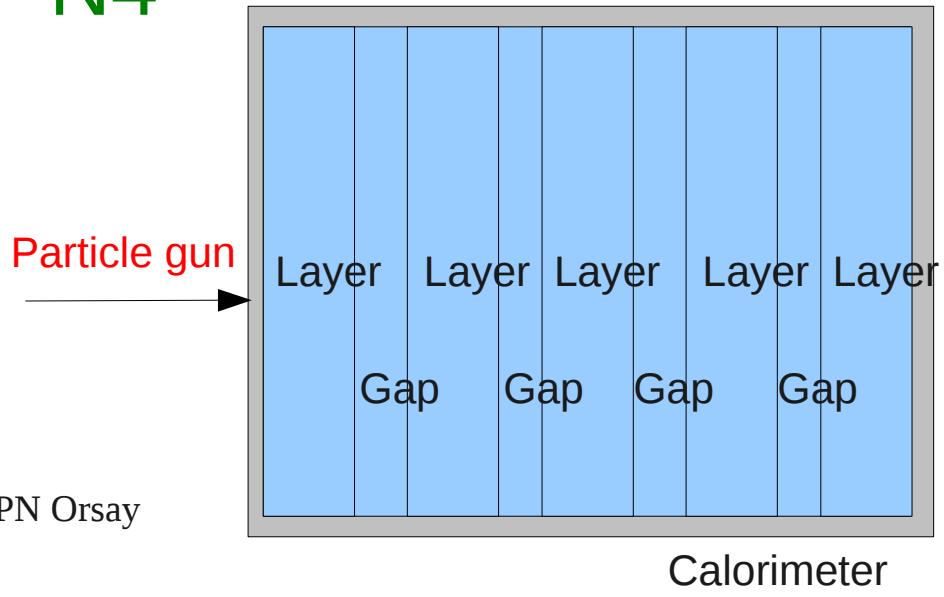
N2



N3



N4



Novice Examples

Features

N1:

- Box (Phantom), Sphere (Tissue1), Cons (Tissue2)
- PVPlacement, only translation
- Scoring within Tissue1,2 via Stepping action
- PL: QGSP_BERT

N2 (a,b):

- Tubs (Target), Tubs (Tracker),
- PVPlacement, only translation (a)
- PVParameterised (b)
- Global constant magnetic field
- Scoring within Tracker via Hits and SDs
- PL: QGSP_BERT + G4StepLimiter

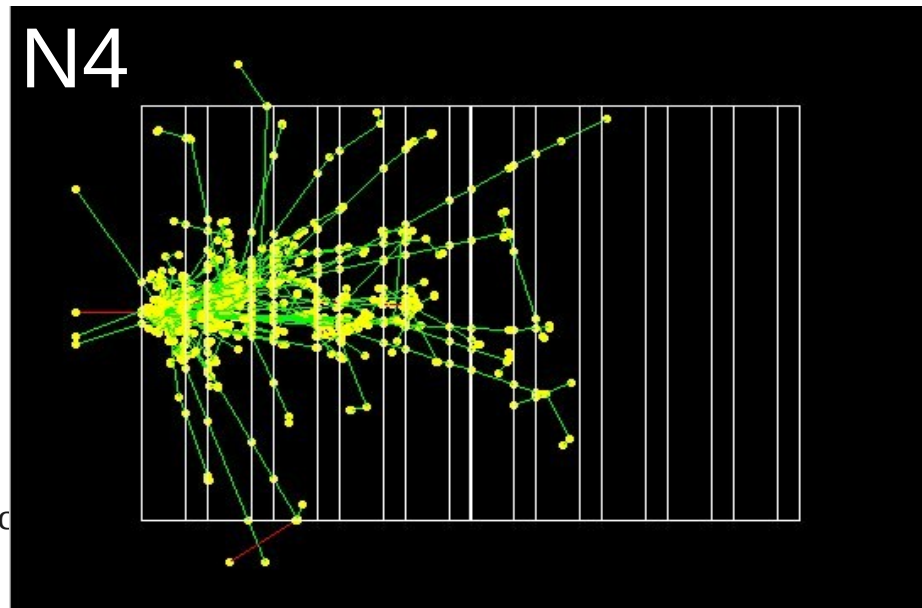
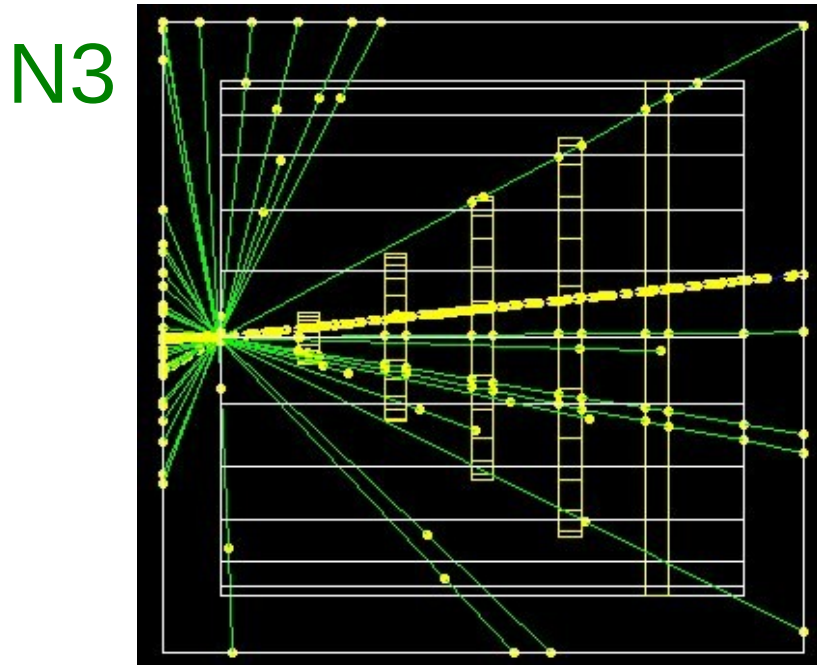
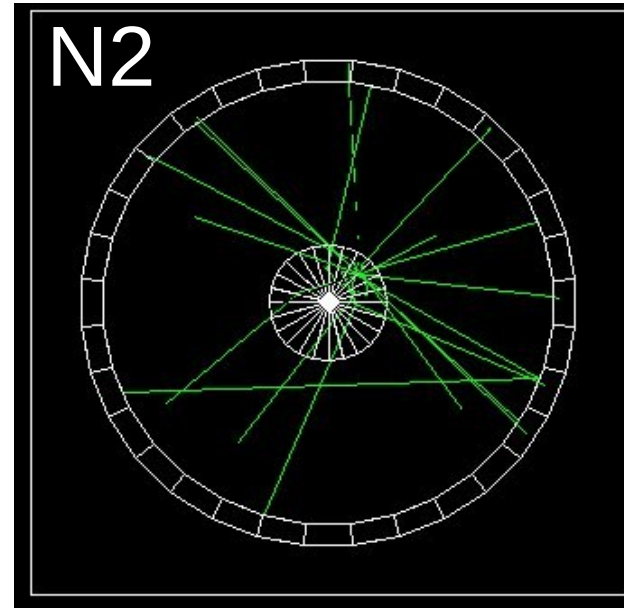
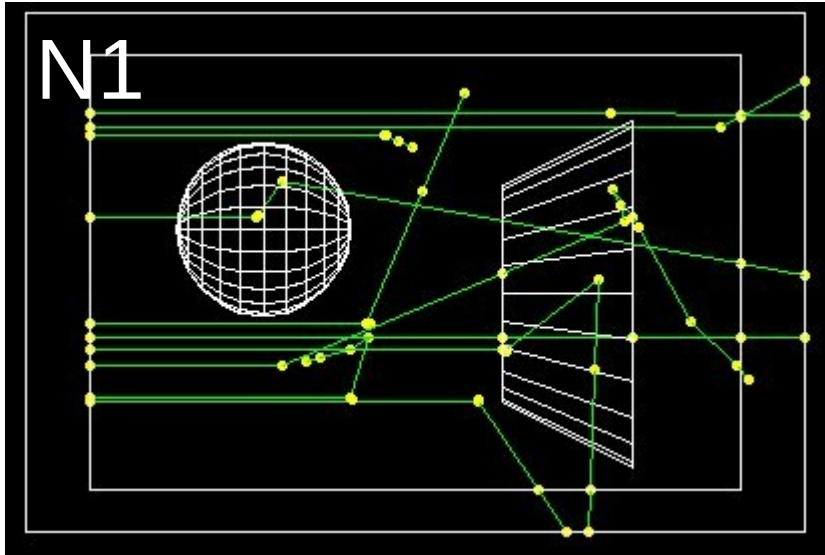
N3:

- Box (Tissue), Trd (Crystals),
- PVPlacement with rotation (within a circle) and PVReplica as alternative
- Scoring within Crystals via Scorers
- PL: Modular physics list with builders from Geant4

N4 (a,b,c,d):

- Box (Calorimeter, Layer, Gap)
- PVReplica
- Scoring within Layer, Gap in 4 ways: User actions(a), User data object (b), Hits and Sds (c), Scorers (d)
- Histograms, ntuples with new analysis tool
- PL: QGSP_BERT

Novice Examples Implementation



irivnac

Novice Examples To Be Done

- N1: Change volumes names (now specific to medical domain) in neutral ones
- N3 geometry: when filling a full circle with trapezoids, then placements with a rotation is not the best choice
 - Proposal to change G4Trd solid to G4Box
- Add stacking action (to kill neutrinos) to N2 or N3
- The following features were considered and decided not to be included in novice:
 - Regions with production cuts
 - Local magnetic field
 - Explicitly defined physics list
 - General particle source

Novice Examples

What Has Changed

- 7 examples -> 4
- Naming: N0X -> NX
 - In order to avoid confusion with the old novice examples
- Some examples have several variants:
 - N2: two detector constructions; N4: four ways of scoring
 - Each variant is a stand-alone example, the class name includes the option if the class has modifications related to the demonstrated option:
 - In N4a: [N4aEventAction](#), [N4aSteppingAction](#), [N4DetectorConstruction](#), [N4PrimaryGeneratorAction](#), [N4RunAction](#), ...
 - In N4b: [N4bEventAction](#), [N4bSteppingAction](#), [N4bRunData](#), [N4DetectorConstruction](#), [N4PrimaryGeneratorAction](#), [N4RunAction](#), ...
- Analysis code in N4 example – with use of new analysis tools

Extended Examples

- New directories (added in extended_branch):
 - **common:**
 - Common classes, scripts
 - **new**
 - New “feature” examples: two brand new one to demonstrate use of common classes, one re-written example (pythia6Decayer)

Extended Examples

Common Classes (1)

- analysis:
 - EG4HbookManager
- detectorConstruction:
 - Two detector construction classes with a messenger for each:
 - 01: with only a world volume (box shape)
 - 02: with a box placed in a world (of box shape)
 - The dimensions/materials can be set in the class constructors or via interactive commands
- primaryGenerator:
 - Two primary generator classes:
 - 01: with G4ParticleGun
 - 02. with G4GeneralParticleSource
 - The default particle properties (type, energy, position, momentum) can be set in the class constructors or via interactive commands

Extended Examples

Common Classes (2)

- userActions
 - One event action and one run action class, both with a messenger
 - Implement setting verbose level, event modulo printing, option to save and read random number status, option for automatic (time-based) random seeds for each run

New Extended Examples (1)

- Materials/mat01
 - [Mat01DetectorConstruction](#) class only, all other classes from common
 - To get classes from the common repository:
 - make setup
 - This will call the provided script `copy_files.sh` which calls a general script from common
 - The selection of classes via generic names, the version and specific keywords:
 - Eg. “DetectorConstruction 02 messenger” will trigger copying `EG4DetectorConstruction02.hh,cc` and `EG4DetectorConstruction02Messenger.hh,cc` files
 - One can easily interchange the available class versions

New Extended Examples (2)

- Geometry/transform
 - Demonstrates various ways how to define 3D transformations
 - [TrDetectorConstruction](#), [TrDetectorConstructionMessenger](#) classes
- Decay/pythia6Decayer:
 - The existing example [eventgenerator/pythia/decayer6](#) re-written with use of classes from common
 - 14 *.hh files, 3754 LOC in old => 7 *.hh files, 1934 LOC in new

Documentation

- Automatically generated with Doxygen:
 - <http://www.stack.nl/~dimitri/doxygen/index.html>
- From:
 - Source code
 - Adapted README files (.README)
 - Configuration files in `novice_branch/.doxygen`
- The documentation pages integrated in the generated documentation
- Hyper-links also with external packages
 - Allows to link with <http://www-geant4.kek.jp/Reference/>
- Looks nice !

Geant4 examples

Main Page	Related Pages	Modules	Namespaces
Classes	Files	<input type="text" value="Search"/>	

Geant4 Examples


This module collects three sets of user examples aimed to demonstrate to the user how to make correct use of the GEANT4 toolkit by implementing in a correct way those user-classes which the user is supposed to customize in order to define his/her own simulation setup.

One set of examples is oriented to "novice" users and covering many basic general use-cases typical of an "application"-oriented kind of development. An "extended" set of examples is covering various use-cases and may require some additional libraries besides of Geant4. This set covers some specific use cases for actual detector simulation. An "advanced" set of examples covers the use-cases typical of a "toolkit"-oriented kind of development, where real complete applications for different simulation studies are provided; may require additional third party products to be built.

Most of the examples can be run both in interactive and batch mode, and input macro files (*.in) and reference output files (*.out) are provided. Novice and most of the extended examples are considered part of the system testing suite for validation of the official releases of the GEANT4 toolkit. Novice examples and some of the extended and advanced examples are also used as "acceptance"-tests for the release process.

See more on each examples category pages:

- [Geant4 Novice Examples](#)
- [Geant4 Extended Examples](#)

Generated on Fri Sep 9 2011 for Geant4 examples by  1.7.4

Plan for Release 9.5

- New novice examples
 - Close to a final state
 - Only a few more features need to be added; run and visualization macros need to be finalized
 - Help from visualization WG is welcome
 - Will be ready for the December release
- New extended examples
 - The new examples can coexist with the old ones, those which are ready can go in the December release
 - The “old” novice examples not included in the new set should be moved to extended

Plan for 2012

- New extended examples
 - Revision of existing examples
 - Migration to the use of common classes
 - Applying coding guidelines
 - Adding new features not yet covered
 - An effort of all WG members will be needed to progress