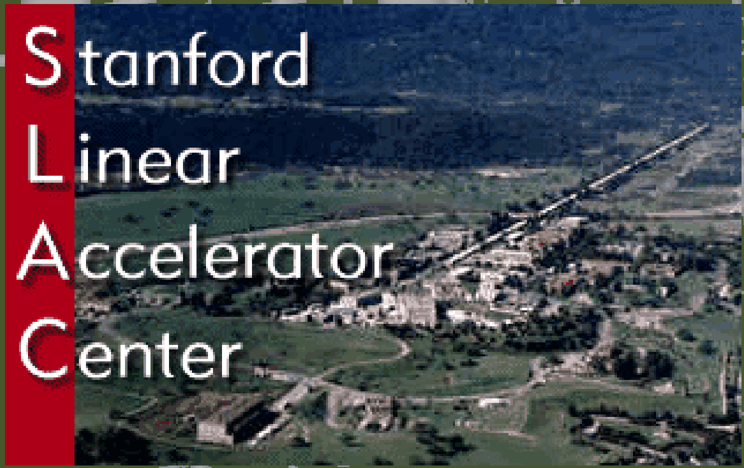


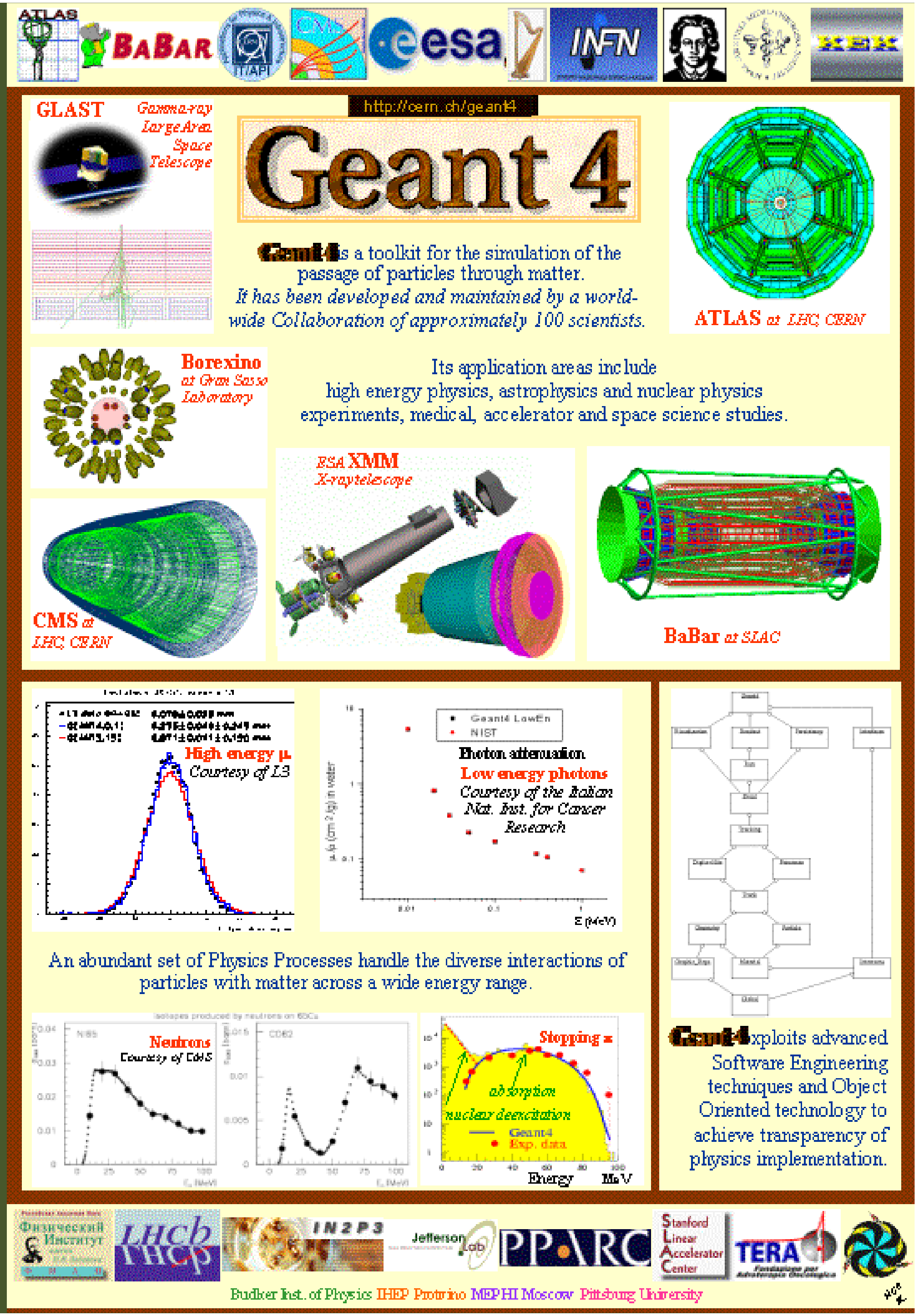
Kernel I

Makoto Asai (SLAC)

Geant4 Tutorial Course

Contents

- General introduction and brief history
- Geant4 license
- Geant4 kernel
  - Basic concepts and kernel structure
  - User classes



General introduction and brief history

What is Geant4?

- Geant4 is the successor of GEANT3, the world-standard toolkit for HEP detector simulation.
- Geant4 is one of the first successful attempt to re-design a major package of HEP software for the next generation of experiments using an Object-Oriented environment.
- A variety of requirements have also taken into account from heavy ion physics, CP violation physics, cosmic ray physics, astrophysics, space science and medical applications.
- In order to meet such requirements, a large degree of functionality and flexibility are provided.
- G4 is not only for HEP but goes well beyond that.

Flexibility of Geant4

- In order to meet wide variety of requirements from various application fields, a large degree of functionality and flexibility are provided.
- Geant4 has many types of geometrical descriptions to describe most complicated and realistic geometries
  - CSG, BREP and Boolean solids
  - Placement, replica, divided, parameterized, reflected and grouped
  - XML interface
- Everything is open to the user
  - Choice of physics processes/models
  - Choice of GUI/Visualization/persistency/histogramming technologies

Physics in Geant4

- It is rather unrealistic to develop a uniform physics model to cover wide variety of particles and/or wide energy range.
- Much wider coverage of physics comes from mixture of theory-driven, parameterized, and empirical formulae. Thanks to polymorphism mechanism, both cross-sections and models (final state generation) can be combined in arbitrary manners into one particular process.
- Geant4 offers
  - EM processes
  - Hadronic processes
  - Photon/lepton-hadron processes
  - Optical photon processes
  - Decay processes
  - Shower parameterization
  - Event biasing techniques
  - And you can plug-in more

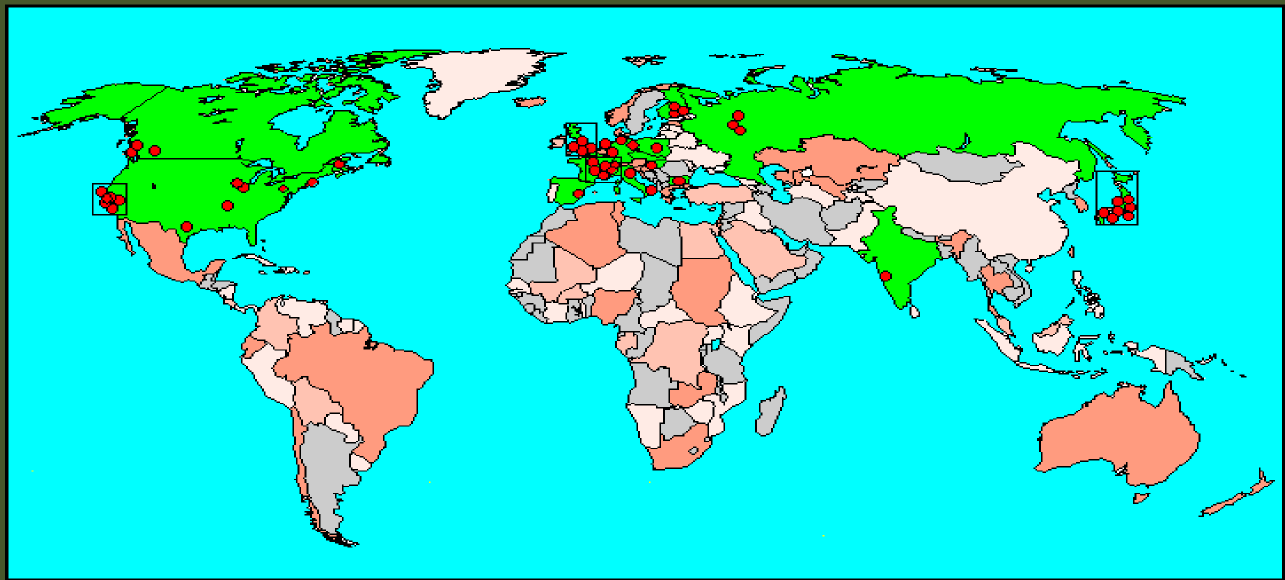
Physics in Geant4

- Each cross-section table or physics model (final state generation) has its own applicable energy range. Combining more than one tables / models, one physics process can have enough coverage of energy range for wide variety of simulation applications.
- Geant4 provides sets of alternative physics models so that the user can freely choose appropriate models according to the type of his/her application.
  - In other words, it is the user's responsibility to choose reasonable set of physics processes/models that fits to his/her needs.
  - For example, some models are more accurate than others at a sacrifice of speed.

Geant4 – Its history

- Dec ’94 - Project start
- Apr ’97 - First alpha release
- Jul ’98 - First beta release
- Dec ’98 - First Geant4 public release - version 1.0
- …
- June 30th, ’06 - Geant4 version 8.1 release
  - July 27th, ’06 - Geant4 8.1-patch01 release
- Dec 15th, ’06 - Geant4 version 8.2 release
  - Feb 23rd, ’07 - Geant4 8.2-patch01 release
- May 11th, ’07 - Geant4 version 8.3 release ← Current version
- June 29th, ’07 - Geant4 version 9.0 release (planned)
- We currently provide two to three public releases every year.
  - Bimonthly beta releases are available to the registered beta-testers.

Geant4 Collaboration



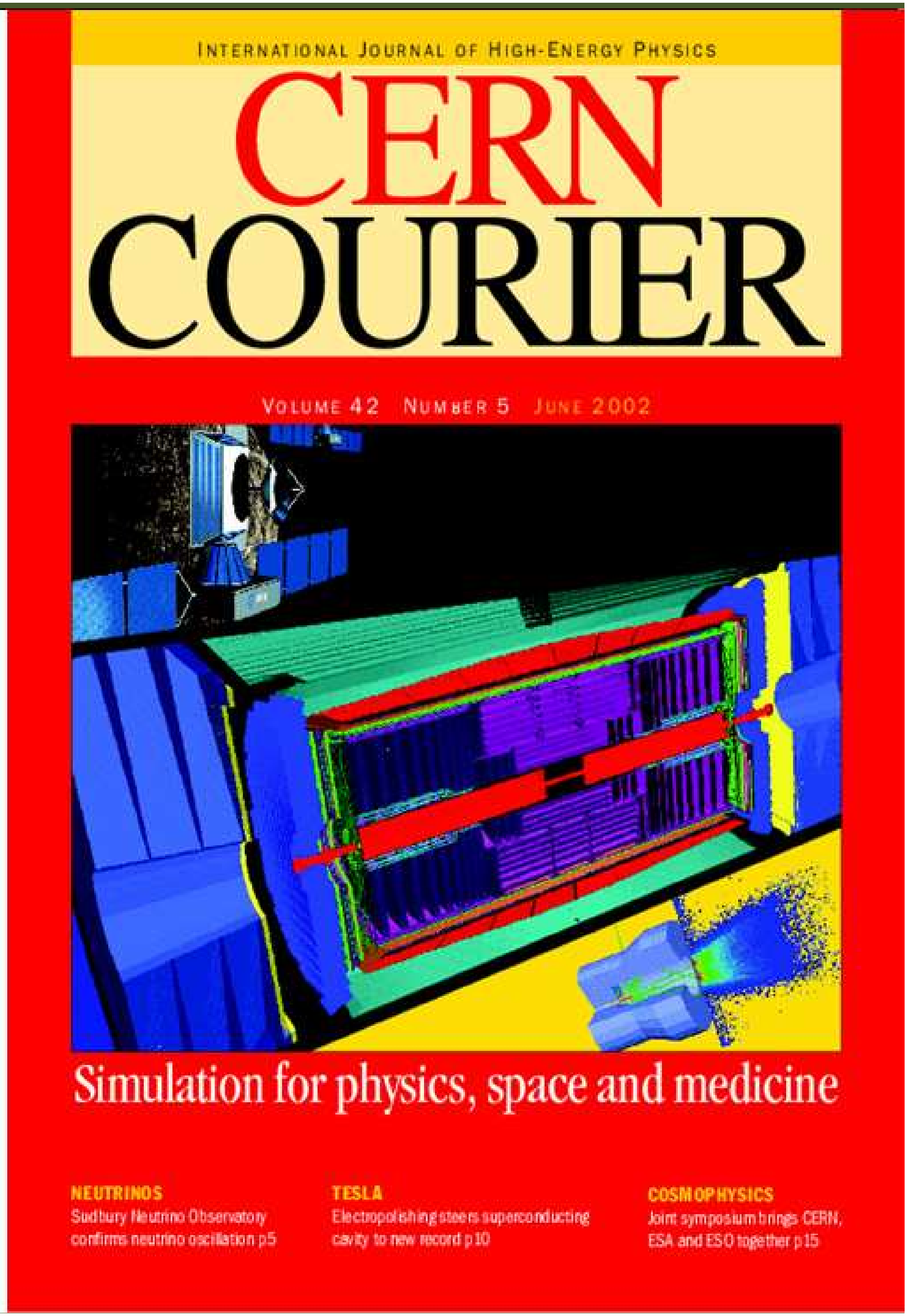
TRIUMF

Lebedev

J.W.Goethe Universität

Collaborators also from non-member institutions, including
Budker Inst. of Physics
IHEP Protvino
MEPHI Moscow
Pittsburg University

Technology transfer



Particle physics software aids space and medicine

Geant4 is a showcase example of technology transfer from particle physics to other fields such as space and medical science […].

CERN Courier, June 2002

http://top25.sciencedirect.com/index.php?subject_area_id=21

SCIENCE DIRECT

TOP 25 HOTTEST ARTICLES ON ScienceDirect

INTRODUCTION

The **ScienceDirect TOP25 Hottest Articles** is a **free** quarterly service from ScienceDirect. When you subscribe to the **ScienceDirect TOP25**, you'll receive an e-mail every three months listing the ScienceDirect users' 25 most frequently downloaded journal articles, from any selected journal among more than 2,000 titles in the ScienceDirect database, or from any of 24 subject areas.

Now you can keep track of the latest trends in your speciality and find out what your colleagues are reading. A simple click on any of the listed articles will take you to the journal abstract, and of course, you have the option to download any article straight to your desktop, depending on your access rights.

Read more about how the TOP25 is generated and what it reflects.

Check out the ScienceDirect TOP25 Hottest Articles within your area of interest.

Subject Area: Physics and Astronomy

Journal: [all journals]

With these drop-down menus, the ScienceDirect TOP25 Hottest Articles are selected. Please refine your selection if necessary. To see the overall TOP25 within a certain subject area or journal, please select 'all subjects' or 'all journals' from the drop-down menus.

TOP25 articles within the subject area: Physics and Astronomy

1. Nanoscience and engineering in mechanics and materials • **Article**
   Journal of Physics and Chemistry of Solids, Volume 65, Issue 8-9, 1 August 2004, Pages 1501-1506
   Chong, K.P.
2. Geant4-a simulation toolkit • **Article**
   Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, Volume 506, Issue 3, 1 July 2003, Pages 250-303
   Agostinelli, S.; Allison, J.; Amako, K.; Apostolakis, J.; Araujo, H.; Arce, P.; Asai, M.; Axen, D.; Banerjee, S.; Barrand, G.; Behner, F.; Bellagamba, L.; Boudreau, J.; Broglia, L.; Brunengo, A.; Burk
3. Radiation pneumonitis and pulmonary fibrosis in non-small-cell lung cancer: Pulmonary function,

http://www.in-cites.com/hotpapers/2004/november04-eng.html

- http://www.in-cites.com/hotpapers/2005/jan05-eng.html
  - http://www.in-cites.com/hotpapers/2005/mar05-eng.html
    - http://www.in-cites.com/hotpapers/2005/may05-eng.html
      - http://www.in-cites.com/hotpapers/2005/july05-eng.html

SCIENTISTS | PAPERS | INSTITUTIONS | JOURNALS | COUNTRIES | HOME

in-cites - July 2005
Citing URL: http://www.in-cites.com/hotpapers/2005/july05-eng.html

The Top 3 Hot Papers

Published In The Last 2-Years For:

Engineering

Previous | Field Menu 2006 | Field Menu 2005 | Field Menu 2004
"Super Hot" Papers in Science Published Since 2003

Engineering

(Sorted by citations. 3 of 128)

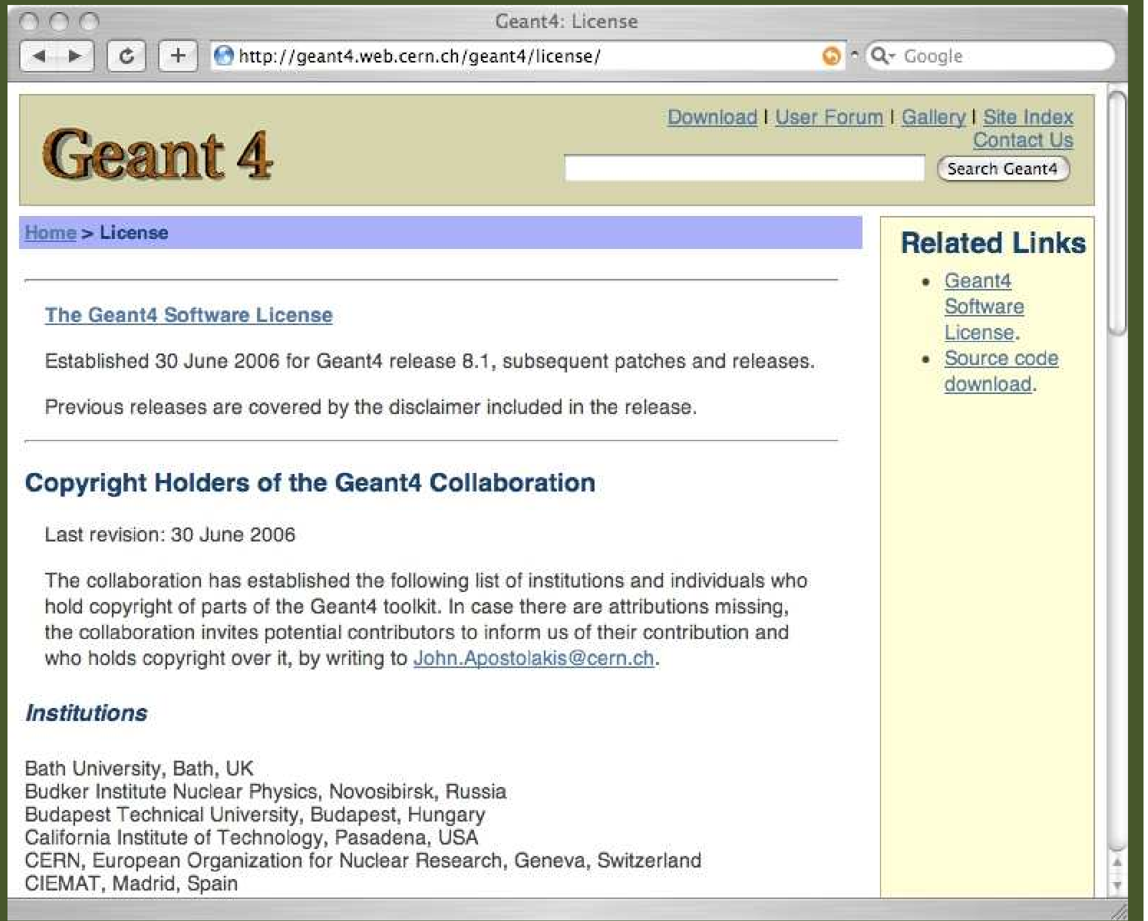
1 Citations: 133
Title: GEANT4-A SIMULATION TOOLKIT
Authors: AGOSTINELLI S; ALLISON J; AMAKO K; APOSTOLAKIS J; ARAUJO H; ARCE P; ASAI M; AXEN D; BANERJEE S; BARRAND G; BEHNER F; BELLAGAMBA L; BOUDREAU J; BROGLIA L; BRUNENGO A; BURKHARDT H; CHAUVIE S; CHUMA J; CHYTRACEK R; COOPERMAN G; COSMO G; DEGTYARENKO P; DELL'ACQUA A; DEPAOLA G; DIETRICH D; ENAMI R; FELICIELLO A; FERGUSON C; FESEFELDT H; FOLGER G; FOPPIANO F; FORTI A; GARELLI S; GIANI S; GIANNITRAPANI R; GIBIN D; CADENAS JJG; GONZALEZ I; ABRIL GG; GREENIAUS G; GREINER W; GRICHINE V; GROSSHEIM A; GUATELLI S; GUMPLINGER P; HAMATSU R; HASHIMOTO K; HASUI H; HEIKKINEN A; HOWARD A; IVANCHENKO V; JOHNSON A; JONES FW; KALLENBACH J; KANAYA N; KAWABATA M; KAWABATA Y; KAWAGUTI M; KELNER S; KENT P; KIMURA A; KODAMA T; KOKOULIN R; KOSSOV M; KURASHIGE H; LAMANNA E; LAMPEN T; LARA V; LEFEBURE V; LEI F; LIENDL M; LOCKMAN W; LONGO F; MAGNI S; MAIRE M; MEDERNACH E; MINAMIMOTO K; DE FREITAS PM; MORITA Y; MURAKAMI K; NAGAMATU M; NARTALLO R; NIEMINEN P; NISHIMURA T; OHTSUBO K; OKAMURA M; O'NEALE S; OOHATA Y; PAECH K; PERL J; PFEIFFER A; PIA MG; RANJARD F; RYBIN A; SADILOV S; DI SALVO E; SANTIN G; SASAKI T; SAVVAS N; SAWADA Y; SCHERER S; SEIL S; SIROTENKO V; SMITH D; STARKOV N; STOECKER H; SULKIMO J; TAKAHATA M; TANAKA S; TCHERNIAEV E; TEHRANI ES; TROPEANO M; TRUSCOTT P; UNO H; URBAN L; URBAN P; VERDERI M; WALKDEN A; WANDER W; WEBER H; WELLISCH JP;

Geant4 license

The New Geant4 License

In response to user requests for clarification of Geant4's distribution policy, the collaboration recently announced a new license.

- Makes clear the user's wide-ranging freedom to use, extend or redistribute Geant4, even as part of some for-profit venture.
- The license was released along with the latest Geant4 release 8.1.
- Simple enough that you can read and understand it.



- http://cern.ch/geant4/license/

The New Geant4 License

License has 8 points. The points are written clearly and simply.

1,2 and 3) Tell the world who the software came from, and don't claim you are us.

Installation, use, reproduction, display, modification and redistribution of this software, with or without modification, in source and binary forms, are permitted on a non- exclusive basis. Any exercise of rights by you under this license is subject to the following conditions:

1. Redistributions of this software, in whole or in part, with or without modification, must reproduce the above copyright notice and these license conditions in this software, the user documentation and any other materials provided with the redistributed software.

2. The user documentation, if any, included with a redistribution, must include the following notice:
"This product includes software developed by Members of the Geant4 Collaboration (http://cern.ch/geant4)."

If that is where third-party acknowledgments normally appear, this acknowledgment must be reproduced in the modified version of this software itself.

3. The names "Geant4" and "The Geant4 toolkit" may not be used to endorse or promote software, or products derived therefrom, except with prior written permission by license@geant4.org. If this software is redistributed in modified form, the name and reference of the modified version must be clearly distinguishable from that of this software.

The New Geant4 License

4) If you choose to give it away free to everyone, we can have it for free too.

5) You can't patent the parts we did.

4. You are under no obligation to provide anyone with any modifications of this software that you may develop, including but not limited to bug fixes, patches, upgrades or other enhancements or derivatives of the features, functionality or performance of this software. However, if you publish or distribute your modifications without contemporaneously requiring users to enter into a separate written license agreement, then you are deemed to have granted all Members and all Copyright Holders of the Geant4 Collaboration a license to your modifications, including modifications protected by any patent owned by you, under the conditions of this license.

5. You may not include this software in whole or in part in any patent or patent application in respect of any modification of this software developed by you.

The New Geant4 License

We don’t claim that it works, and we’re not responsible if it doesn’t.

6. DISCLAIMER
THIS SOFTWARE IS PROVIDED BY THE MEMBERS AND COPYRIGHT HOLDERS OF THE GEANT4 COLLABORATION AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, IMPLIED WARRANTIES OF MERCHANTABILITY, OF SATISFACTORY QUALITY, AND FITNESS FOR A PARTICULAR PURPOSE OR USE ARE DISCLAIMED. THE MEMBERS OF THE GEANT4 COLLABORATION AND CONTRIBUTORS MAKE NO REPRESENTATION THAT THE SOFTWARE AND MODIFICATIONS THEREOF, WILL NOT INFRINGE ANY PATENT, COPYRIGHT, TRADE SECRET OR OTHER PROPRIETARY RIGHT.

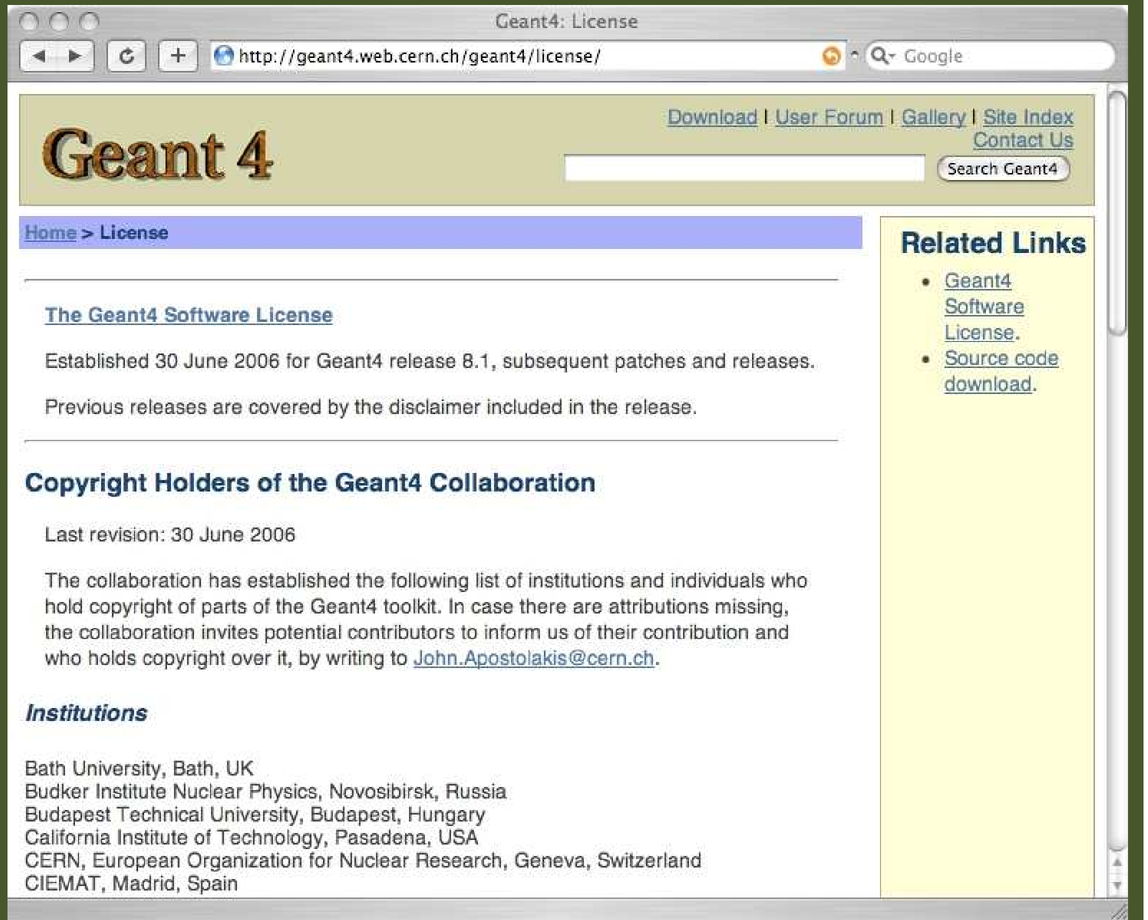
7. LIMITATION OF LIABILITY
THE MEMBERS AND COPYRIGHT HOLDERS OF THE GEANT4 COLLABORATION AND CONTRIBUTORS SHALL HAVE NO LIABILITY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, CONSEQUENTIAL, EXEMPLARY, OR PUNITIVE DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, LOSS OF USE, DATA OR PROFITS, OR BUSINESS INTERRUPTION, HOWEVER CAUSED AND ON ANY THEORY OF CONTRACT, WARRANTY, TORT (INCLUDING NEGLIGENCE), PRODUCT LIABILITY OR OTHERWISE, ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

8. This license shall terminate with immediate effect and without notice if you fail to comply with any of the terms of this license, or if you institute litigation against any Member or Copyright Holder of the Geant4 Collaboration with regard to this software.

The New Geant4 License

That's it.

- Through the generous support of many governments and institutions, you, the community, have already paid for Geant4.
- We, the Geant4 collaboration, want you to enjoy the software and use it where ever you can.



- http://cern.ch/geant4/license/

Basic concepts
and kernel structure

Terminology (jargons)

- Run, event, track, step, step point
- Track ←→ trajectory, step ←→ trajectory point
- Process
  - At rest, along step, post step
- Cut = production threshold
- Sensitive detector, hit, hits collection

Run in Geant4

- As an analogy of the real experiment, a run of Geant4 starts with "Beam On".
- Within a run, the user cannot change
  - detector setup
  - settings of physics processes
- Conceptually, a run is a collection of events which share the same detector and physics conditions.
  - A run consists of one event loop.
- At the beginning of a run, geometry is optimized for navigation and cross-section tables are calculated according to materials appear in the geometry and the cut-off values defined.
- G4RunManager class manages processing a run, a run is represented by G4Run class or a user-defined class derived from G4Run.
  - A run class may have a summary results of the run.
- G4UserRunAction is the optional user hook.

Event in Geant4

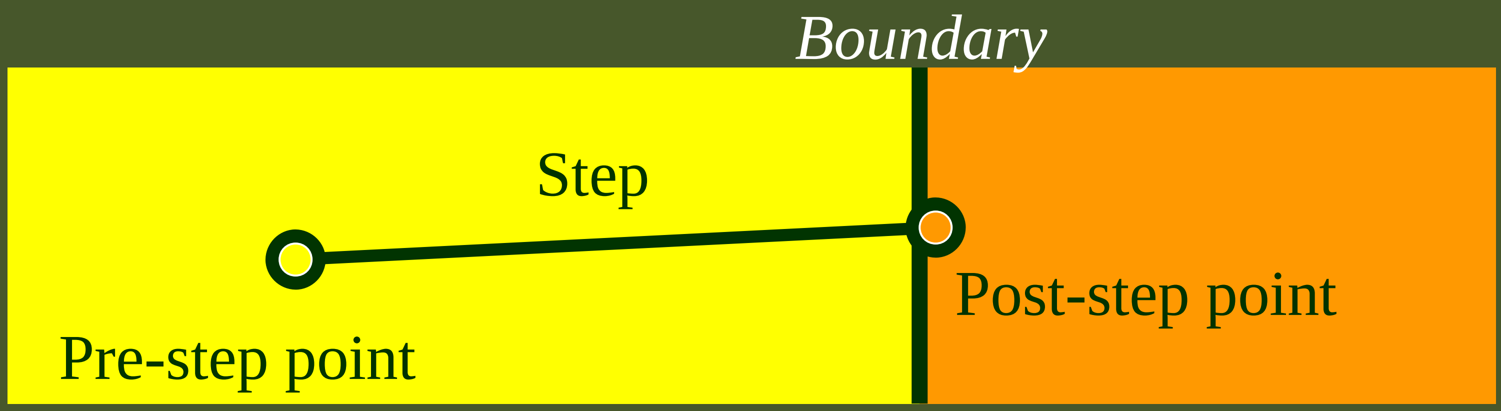
- An event is the basic unit of simulation in Geant4.
- At beginning of processing, primary tracks are generated. These primary tracks are pushed into a stack.
- A track is popped up from the stack one by one and "tracked". Resulting secondary tracks are pushed into the stack.
  - This "tracking" lasts as long as the stack has a track.
- When the stack becomes empty, processing of one event is over.
- G4Event class represents an event. It has following objects at the end of its (successful) processing.
  - List of primary vertices and particles (as input)
  - Hits and Trajectory collections (as output)
- G4EventManager class manages processing an event. G4UserEventAction is the optional user hook.

Track in Geant4

- Track is a snapshot of a particle.
  - It has physical quantities of current instance only. It does not record previous quantities.
  - Step is a "delta" information to a track. Track is not a collection of steps. Instead, a track is being updated by steps.
- Track object is deleted when
  - it goes out of the world volume,
  - it disappears (by e.g. decay, inelastic scattering),
  - it goes down to zero kinetic energy and no "AtRest" additional process is required, or
  - the user decides to kill it artificially.
- No track object persists at the end of event.
  - For the record of tracks, use trajectory class objects.
- G4TrackingManager manages processing a track, a track is represented by G4Track class.
- G4UserTrackingAction is the optional user hook.

Step in Geant4

- Step has two points and also "delta" information of a particle (energy loss on the step, time-of-flight spent by the step, etc.).
- Each point knows the volume (and material). In case a step is limited by a volume boundary, the end point physically stands on the boundary, and it logically belongs to the next volume.
  - Because one step knows materials of two volumes, boundary processes such as transition radiation or refraction could be simulated.
- G4SteppingManager class manages processing a step, a step is represented by G4Step class.
- G4UserSteppingAction is the optional user hook.



Trajectory and trajectory point

- Track does not keep its trace. No track object persists at the end of event.
- G4Trajectory is the class which copies some of G4Track information. G4TrajectoryPoint is the class which copies some of G4Step information.
  - G4Trajectory has a vector of G4TrajectoryPoint.
  - At the end of event processing, G4Event has a collection of G4Trajectory objects.
    - /tracking/storeTrajectory must be set to 1.
- Keep in mind the distinction.
  - G4Track ←→ G4Trajectory, G4Step ←→ G4TrajectoryPoint
- Given G4Trajectory and G4TrajectoryPoint objects persist till the end of an event, you should be careful not to store too many trajectories.
  - E.g. avoid for high energy EM shower tracks.
- G4Trajectory and G4TrajectoryPoint store only the minimum information.
  - You can create your own trajectory / trajectory point classes to store information you need. G4VTrajectory and G4VTrajectoryPoint are base classes.

Particle in Geant4

- A particle in Geant4 is represented by three layers of classes.
- G4Track
  - Position, geometrical information, etc.
  - This is a class representing a particle to be tracked.
- G4DynamicParticle
  - "Dynamic" physical properties of a particle, such as momentum, energy, spin, etc.
  - Each G4Track object has its own and unique G4DynamicParticle object.
  - This is a class representing an individual particle.
- G4ParticleDefinition
  - "Static" properties of a particle, such as charge, mass, life time, decay channels, etc.
  - G4ProcessManager which describes processes involving to the particle
  - All G4DynamicParticle objects of same kind of particle share the same G4ParticleDefinition.

Tracking and processes

- Geant4 tracking is general.
  - It is independent to
    - the particle type
    - the physics processes involving to a particle
  - It gives the chance to all processes
    - To contribute to determining the step length
    - To contribute any possible changes in physical quantities of the track
    - To generate secondary particles
    - To suggest changes in the state of the track
      - e.g. to suspend, postpone or kill it.

Processes in Geant4

- In Geant4, particle transportation is a process as well, by which a particle interacts with geometrical volume boundaries and field of any kind.
  - Because of this, shower parameterization process can take over from the ordinary transportation without modifying the transportation process.
- Each particle has its own list of applicable processes. At each step, all processes listed are invoked to get proposed physical interaction lengths.
- The process which requires the shortest interaction length (in space-time) limits the step.
- Each process has one or combination of the following natures.
  - AtRest
    - e.g. muon decay at rest
  - AlongStep (a.k.a. continuous process)
    - e.g. Celenkov process
  - PostStep (a.k.a. discrete process)
    - e.g. decay on the fly

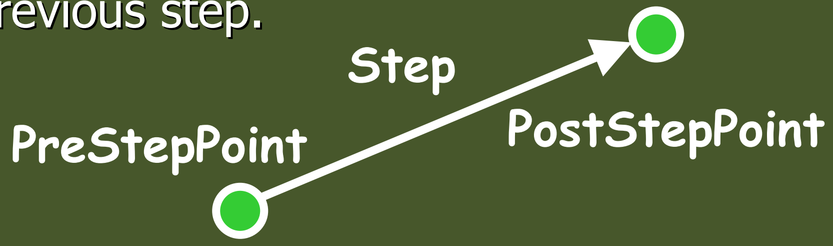
Track status

- At the end of each step, according to the processes involved, the state of a track may be changed.
  - The user can also change the status in UserSteppingAction.
  - Statuses shown in yellow are artificial, i.e. Geant4 kernel won't set them, but the user can set.

- fAlive
  - Continue the tracking.
- fStopButAlive
  - The track has come to zero kinetic energy, but still AtRest process to occur.
- fStopAndKill
  - The track has lost its identity because it has decayed, interacted or gone beyond the world boundary.
  - Secondaries will be pushed to the stack.
- fKillTrackAndSecondaries
  - Kill the current track and also associated secondaries.
- fSuspend
  - Suspend processing of the current track and push it and its secondaries to the stack.
- fPostponeToNextEvent
  - Postpone processing of the current track to the next event.
  - Secondaries are still being processed within the current event.

Step status

- Step status is attached to G4StepPoint to indicate why that particular step was determined.
  - Use "PostStepPoint" to get the status of this step.
  - "PreStepPoint" has the status of the previous step.



- fWorldBoundary
  - Step reached the world boundary
- fGeomBoundary
  - Step is limited by a volume boundary except the world
- fAtRestDoItProc, fAlongStepDoItProc, fPostStepDoItProc
  - Step is limited by a AtRest, AlongStep or PostStep process
- fUserDefinedLimit
  - Step is limited by the user Step limit
- fExclusivelyForcedProc
  - Step is limited by an exclusively forced (e.g. shower parameterization) process
- fUndefined
  - Step not defined yet

- If you want to identify the first step in a volume, pick fGeomBoudary status in PreStepPoint.
- If you want to identify a step getting out of a volume, pick fGeomBoundary status in PostStepPoint

Cuts in Geant4

- A Cut in Geant4 is a production threshold.
  - Not tracking cut, which does not exist in Geant4 as default.
    - All tracks are traced down to zero kinetic energy.
  - It is applied only for physics processes that have infrared divergence
- Much detail will be given at later talks on physics.

Extract useful information

- Given geometry, physics and primary track generation, Geant4 does proper physics simulation "silently".
  - You have to add a bit of code to extract information useful to you.
- There are two ways:
  - Use user hooks (G4UserTrackingAction, G4UserSteppingAction, etc.)
    - You have an access to almost all information
    - Straight-forward, but do-it-yourself
  - Use Geant4 scoring functionality
    - Assign G4VSensitiveDetector to a volume
    - Hits collection is automatically stored in G4Event object, and automatically accumulated if user-defined Run object is used.
    - Use user hooks (G4UserEventAction, G4UserRunAction) to get event / run summary

Unit system

- Internal unit system used in Geant4 is completely hidden not only from user's code but also from Geant4 source code implementation.
- Each hard-coded number must be multiplied by its proper unit.

```
radius = 10.0 * cm;
kineticE = 1.0 * GeV;
```

- To get a number, it must be divided by a proper unit.

```
G4cout << eDep / MeV << " [MeV]" << G4endl;
```

- Most of commonly used units are provided and user can add his/her own units.
- By this unit system, source code becomes more readable and importing / exporting physical quantities becomes straightforward.
  - For particular application, user can change the internal unit to suitable alternative unit without affecting to the result.

G4cout, G4cerr

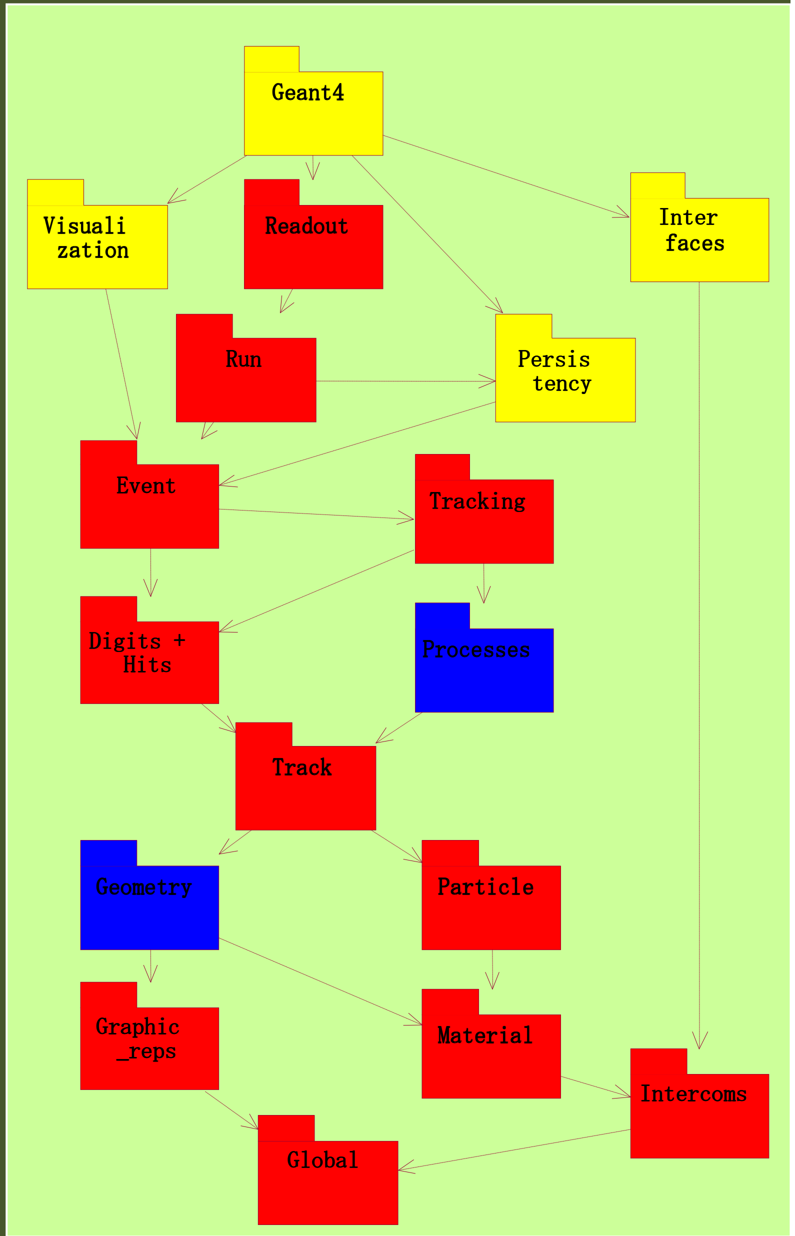
- G4cout and G4cerr are *ostream* objects defined by Geant4.
  - G4endl is also provided.

```
G4cout << "Hello Geant4!" << G4endl;
```

- Some GUIs are buffering output streams so that they display print-outs on another window or provide storing / editing functionality.
  - The user should not use std::cout, etc.
- The user should not use std::cin for input. Use user-defined commands provided by intercoms category in Geant4.
  - Ordinary file I/O is OK.

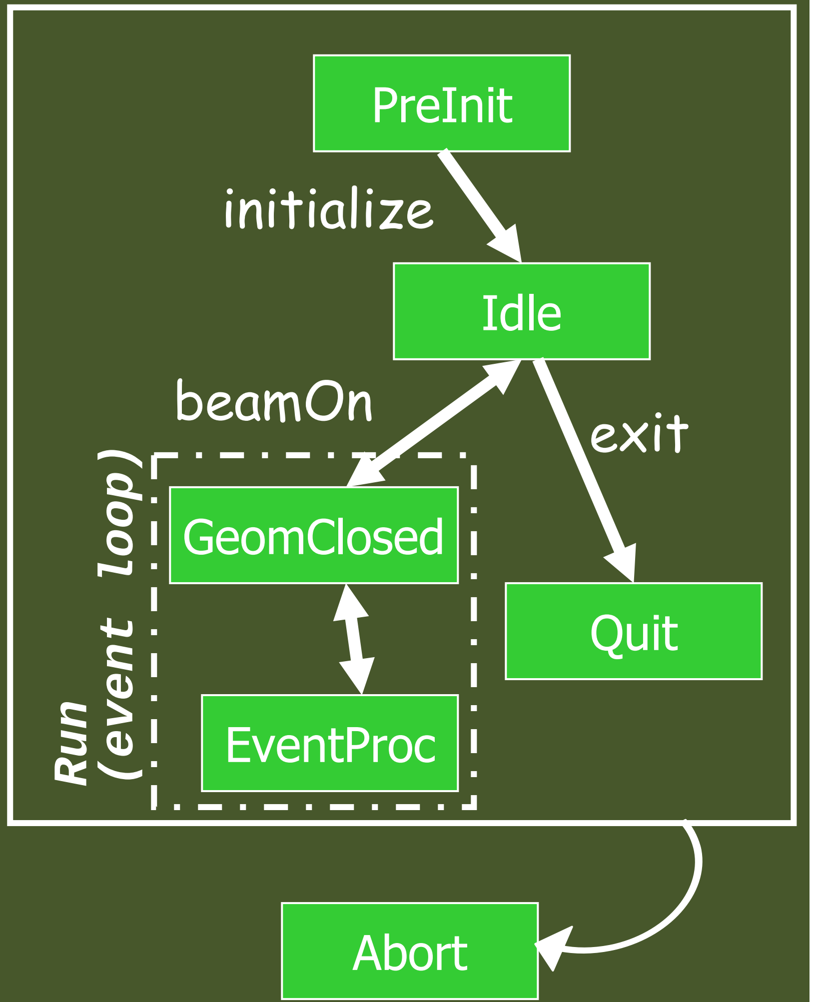
Geant4 kernel

- Geant4 consists of 17 categories.
  - Independently developed and maintained by WG(s) responsible to each category.
  - Interfaces between categories (e.g. top level design) are maintained by the global architecture WG.
- Geant4 Kernel
  - Handles run, event, track, step, hit, trajectory.
  - Provides frameworks of geometrical representation and physics processes.



Geant4 as a state machine

- Geant4 has six application states.
  - G4State_PreInit
    - Material, Geometry, Particle and/or Physics Process need to be initialized/defined
  - G4State_Idle
    - Ready to start a run
  - G4State_GeomClosed
    - Geometry is optimized and ready to process an event
  - G4State_EventProc
    - An event is processing
  - G4State_Quit
    - (Normal) termination
  - G4State_Abort
    - A fatal exception occurred and program is aborting



User classes

To use Geant4, you have to…

- Geant4 is a toolkit. You have to build an application.
- To make an application, you have to
  - Define your geometrical setup
    - Material, volume
  - Define physics to get involved
    - Particles, physics processes/models
    - Production thresholds
  - Define how an event starts
    - Primary track generation
  - Extract information useful to you
- You may also want to
  - Visualize geometry, trajectories and physics output
  - Utilize (Graphical) User Interface
  - Define your own UI commands
  - etc.

User classes

- main()
  - Geant4 does not provide *main().*
- Initialization classes
  - Use G4RunManager::SetUserInitialization() to define.
  - Invoked at the initialization
    - G4VUserDetectorConstruction
    - G4VUserPhysicsList
- Action classes
  - Use G4RunManager::SetUserAction() to define.
  - Invoked during an event loop
    - G4VUserPrimaryGeneratorAction
    - G4UserRunAction
    - G4UserEventAction
    - G4UserStackingAction
    - G4UserTrackingAction
    - G4UserSteppingAction

Note : classes written in yellow are mandatory.

The main program

- Geant4 does not provide the *main*().
- In your *main(),* you have to
  - Construct G4RunManager (or your derived class)
  - Set user mandatory classes to RunManager
    - G4VUserDetectorConstruction
    - G4VUserPhysicsList
    - G4VUserPrimaryGeneratorAction
- You can define VisManager, (G)UI session, optional user action classes, and/or your persistency manager in your *main()*.

Describe your detector

- Derive your own concrete class from G4VUserDetectorConstruction abstract base class.
- In the virtual method *Construct()*,
  - Instantiate all necessary materials
  - Instantiate volumes of your detector geometry
  - Instantiate your sensitive detector classes and set them to the corresponding logical volumes
- Optionally you can define
  - Regions for any part of your detector
  - Visualization attributes (color, visibility, etc.) of your detector elements

Select physics processes

- Geant4 does not have any default particles or processes.
  - Even for the particle transportation, you have to define it explicitly.
- Derive your own concrete class from G4VUserPhysicsList abstract base class.
  - Define all necessary particles
  - Define all necessary processes and assign them to proper particles
  - Define cut-off ranges applied to the world (and each region)
- Geant4 provides lots of utility classes/methods and examples.
  - "Educated guess" physics lists for defining hadronic processes for various use-cases.

Generate primary event

- Derive your concrete class from G4VUserPrimaryGeneratorAction abstract base class.
- Pass a G4Event object to one or more primary generator concrete class objects which generate primary vertices and primary particles.
- Geant4 provides several generators in addition to the G4VPrimaryParticlegenerator base class.
  - G4ParticleGun
  - G4HEPEvtInterface, G4HepMCInterface
    - Interface to /hepevt/ common block or HepMC class
  - G4GeneralParticleSource
    - Define radioactivity

Optional user action classes

- All user action classes, methods of which are invoked during “Beam On”, must be constructed in the user’s *main*() and must be set to the RunManager.
- G4UserRunAction
  - G4Run* GenerateRun()
    - Instantiate user-customized run object
  - void BeginOfRunAction(const G4Run*)
    - Define histograms
  - void EndOfRunAction(const G4Run*)
    - Analyze the run
    - Store histograms
- G4UserEventAction
  - void BeginOfEventAction(const G4Event*)
    - Event selection
  - void EndOfEventAction(const G4Event*)
    - Output event information

Optional user action classes

- G4UserStackingAction
  - void PrepareNewEvent()
    - Reset priority control
  - G4ClassificationOfNewTrack ClassifyNewTrack(const G4Track*)
    - Invoked every time a new track is pushed
    - Classify a new track -- priority control
      - Urgent, Waiting, PostponeToNextEvent, Kill
  - void NewStage()
    - Invoked when the Urgent stack becomes empty
    - Change the classification criteria
    - Event filtering (Event abortion)

Optional user action classes

- G4UserTrackingAction
  - void PreUserTrackingAction(const G4Track*)
    - Decide trajectory should be stored or not
    - Create user-defined trajectory
  - void PostUserTrackingAction(const G4Track*)
    - Delete unnecessary trajectory
- G4UserSteppingAction
  - void UserSteppingAction(const G4Step*)
    - Kill / suspend / postpone the track
    - Draw the step (for a track not to be stored as a trajectory)

Let me remind you…

- Define material and geometry
  - ➔ G4VUserDetectorConstruction
    - Material and Geometry lectures
- Select appropriate particles and processes and define production threshold(s)
  - ➔ G4VUserPhysicsList
    - Physics lectures
- Define the way of primary particle generation
  - ➔ G4VUserPrimaryGeneratorAction
    - Primary particle lecture
- Define the way to extract useful information from Geant4
  - ➔ G4UserSteppingAction, G4UserTrackingAction, etc.
  - ➔ G4VUserDetectorConstruction, G4UserEventAction, G4Run, G4UserRunAction
  - ➔ G4SensitiveDetector, G4VHit, G4VHitsCollection
    - Scoring lectures