Geant 4

Introduction and Getting Started

1/20/2003 @ ESA/ESTEC

Makoto Asai (SLAC)
asai@slac.stanford.edu

Contents

- Scope and history of Geant4
- Key concepts in Geant4
- Global structure of Geant4
- Brief introduction to geometry components
- References to further information

Scope and history of Geant4

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 also came 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, Boolean
  - STEP compliant
  - XML under study
- 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 model to cover wide variety of particles and/or wide energy range.
- Much wider coverage of physics comes from mixture of theory-driven, cross-section tables, and empirical formulae. Thanks to polymorphism mechanism, both cross-sections and models can be combined in arbitrary manners into one particular process.
- Each cross-section table or physics model has its own applicable energy range. Combining more then one tables / models, one physics process can have enough coverage of energy range for wide variety of simulation applications.

Physics in Geant4

- Geant4 provides sets of alternative physics models so that the user can freely choose appropriate models according to the type of his/her application.
- Several individual universities / physicists groups are contributing their physics models to Geant4. Given the modular structure of Geant4, developers of each physics model are well recognized and credited.

Geant4 – Its history and future

- Dec '94 - Project start
- Apr '97 - First alpha release
- Jul '98 - First beta release
- Dec '98 - Geant4 0.0 release
- Jul '99 - Geant4 0.1 release
- …
- Jun '02 - Geant4 4.1 release
- Dec '02 – Geant4 5.0 release
- We currently provide two public releases and four beta releases in between public releases every year.
- We expect to continue to maintain, upgrade and support users of Geant4 for at least 10 years.

Geant4 Collaboration

Stanford Linear Accelerator Center

HARP

Applications for Physics & Infrastructure CERN IT/API

KEK

esa

BaBar

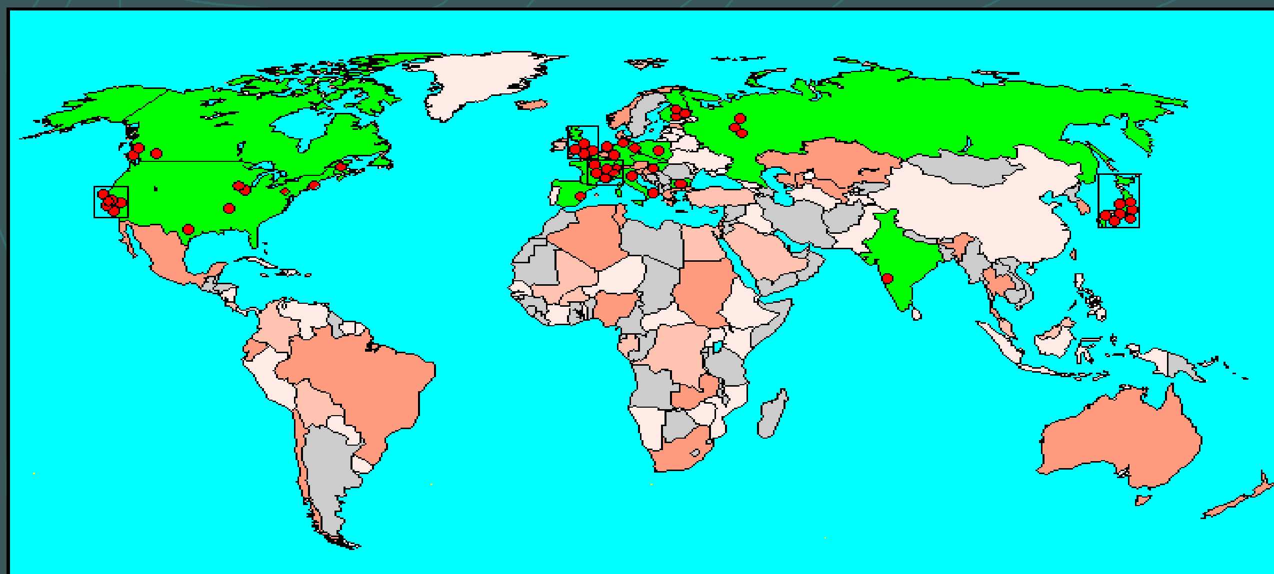
Kongl. Carolinska Medico Chirurgiska Institutet

TERA
Fondazione per Adroterapia Oncologica

ATLAS

INFN
Istituto Nazionale di Fisica Nucleare

CMS



Российская Академия Наук
Физический Институт имени П.Н. Лебедева
ФИАН

Lebedev

LHCb

IN2P3

Univ. Barcelona

PPARC

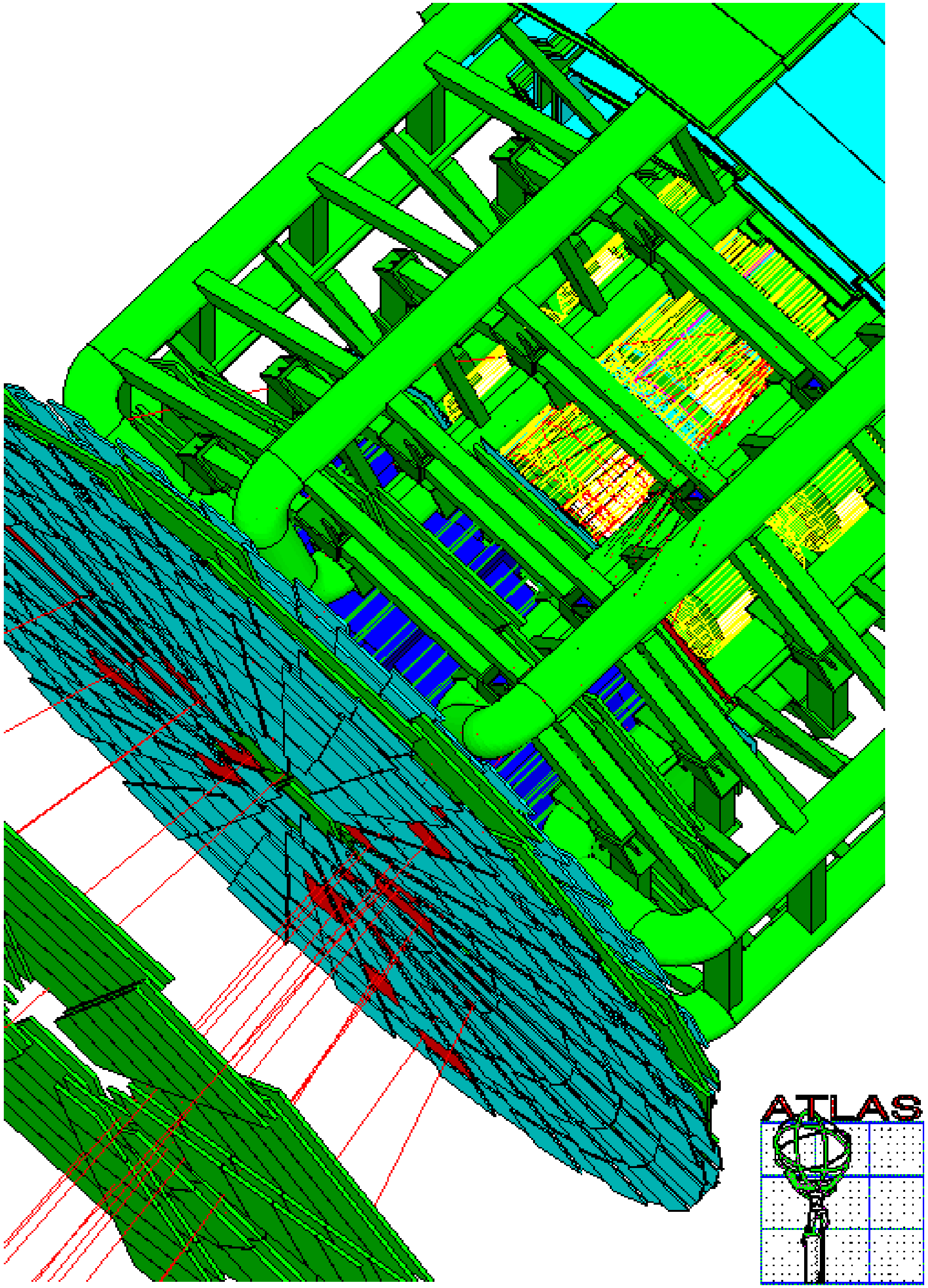
Helsinki Inst. Ph.

Jefferson Lab
Thomas Jefferson National Accelerator Facility

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

Geant4 in HEP

- ATLAS (CERN-LHC)
- 22 x 22 x 44 m^3
- 15,000 ton • • • •
- 4 million channels
- 40 MHz readout



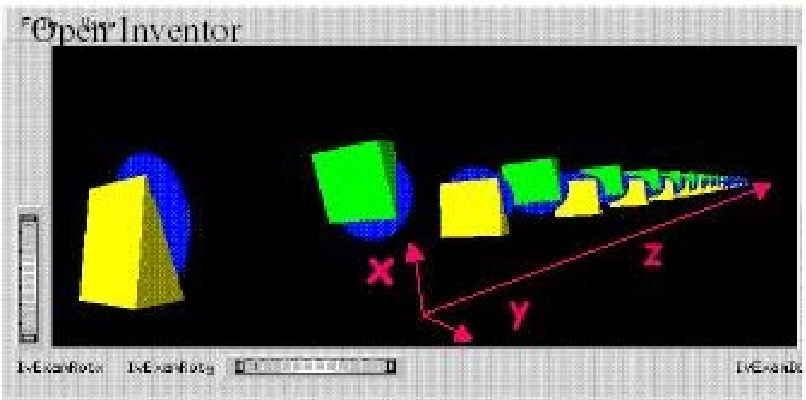
Geant4 for beam transportation

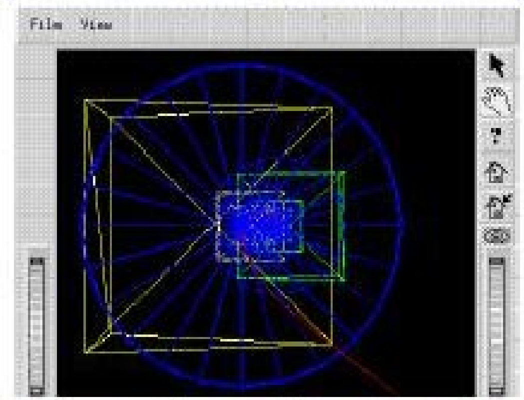
Example: Helical Channel

Published in proc. of PAC 2001 (Fermilab-Conf-01-182-T)

72 m long solenoidal + dipole field with wedge absorbers and thin cavities

$$B_{xy} = B_T \cos,\sin\left(\frac{2p}{L} z\right) \qquad B_z = B_0$$





Other simulations:

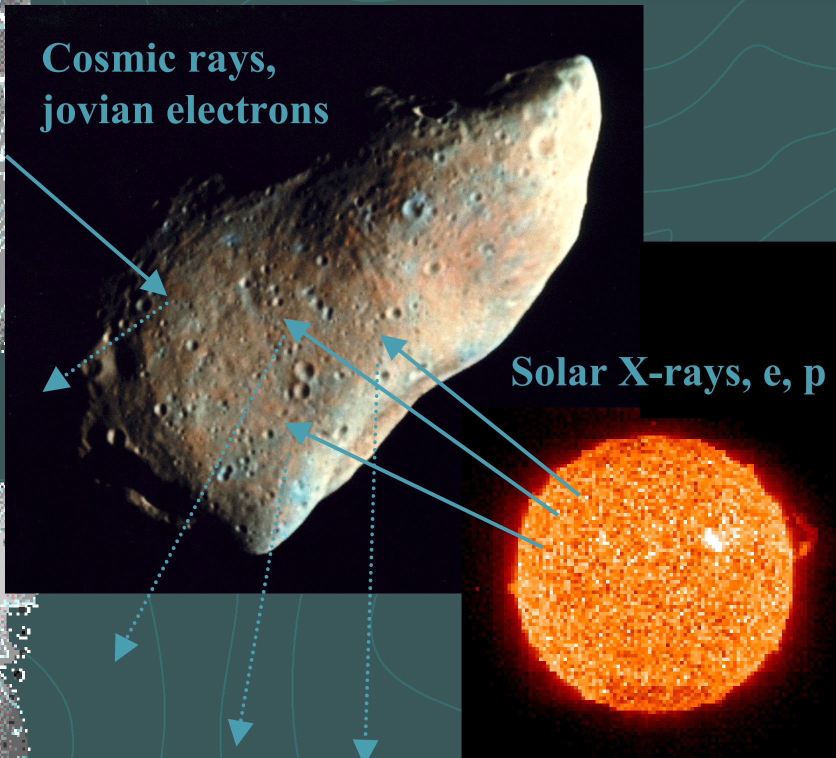
- Alternate Solenoid Channel (sFoFo), published in proceedings of PAC2001 and Feasibility Study II for a Neutrino Factory at BNL (2001)
- Bent Solenoid Channel, presented at Emittance Exchange Workshop, BNL 2000
- Low Frequency r.f. Cooling Channel, presented at International Cooling Experiment Workshop, CERN 2001
- Cooling Experiment (MICE) Simulation (in progress)

G4 Users Meeting, February 21st, 2002 *V. Daniel Elvira, Fermilab*

Geant4 in space science

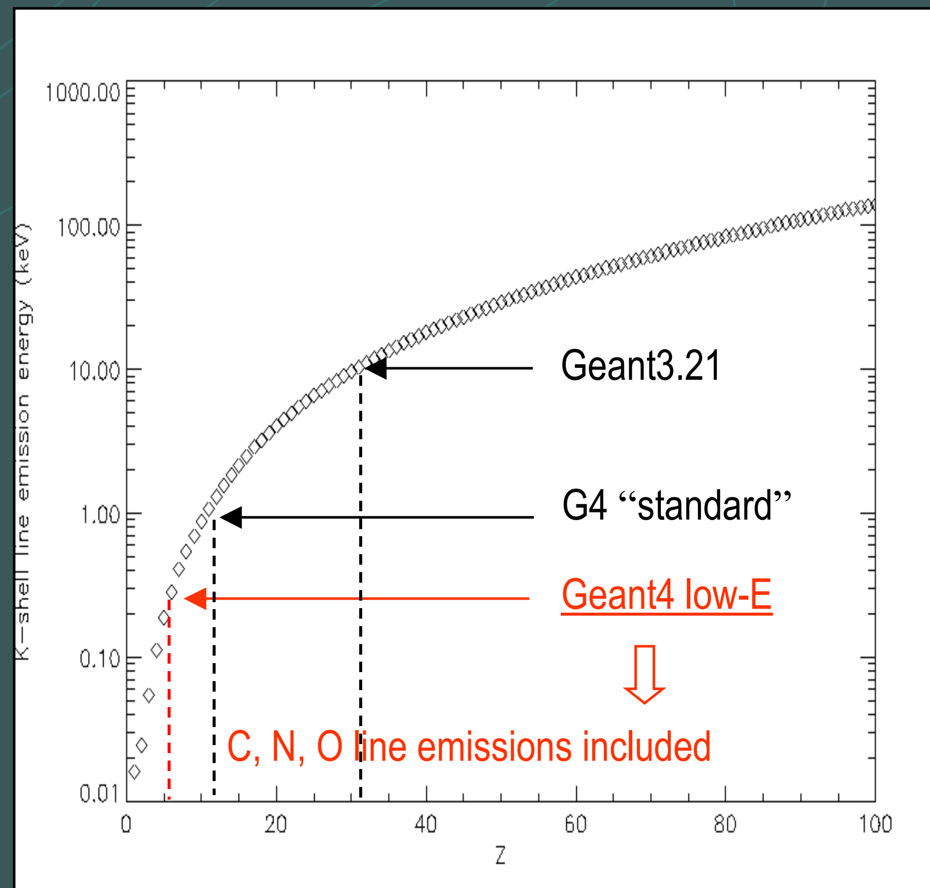
ESA Space Environment & Effects Analysis Section

X-Ray Surveys of Asteroids and Moons



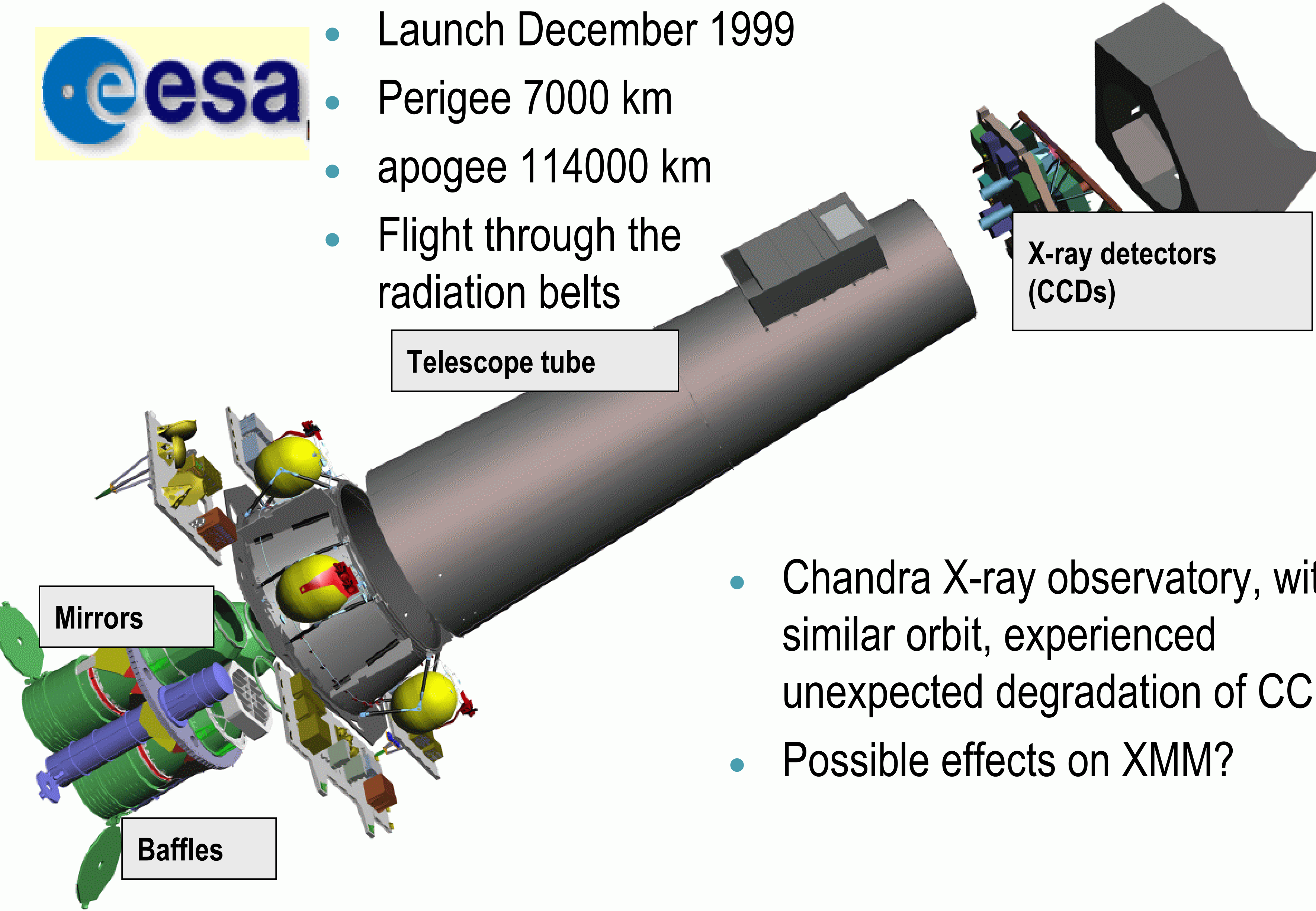
Courtesy SOHO EIT

Induced X-ray line emission: indicator of target composition (~100 μm surface layer)



X-ray Multi-Mirror mission (XMM)

- Launch December 1999
- Perigee 7000 km
- apogee 114000 km
- Flight through the radiation belts



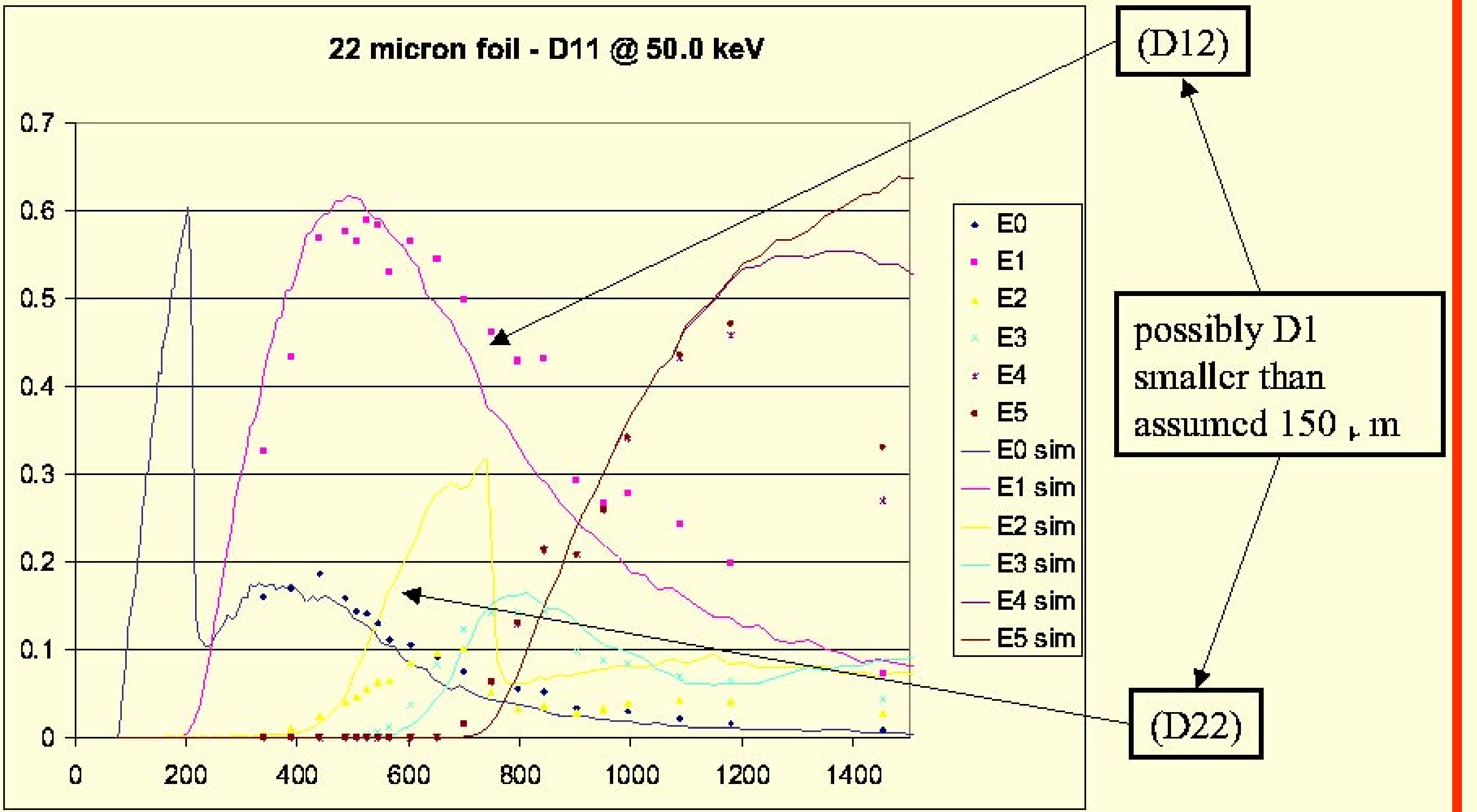
- Chandra X-ray observatory, with similar orbit, experienced unexpected degradation of CCDs
- Possible effects on XMM?

DESIRE (Dose Estimation by Simulation of the ISS Radiation Environment)

- KTH Stockholm, ESTEC, EAC, NASA Johnson
- Prediction of the ambient energetic particle environment (**SPENVIS** & additional models)
- Construction of COLUMBUS geometry in **Geant4**
- Radiation transport, including secondary particle production, through the geometry
- Calculation of astronaut radiation doses

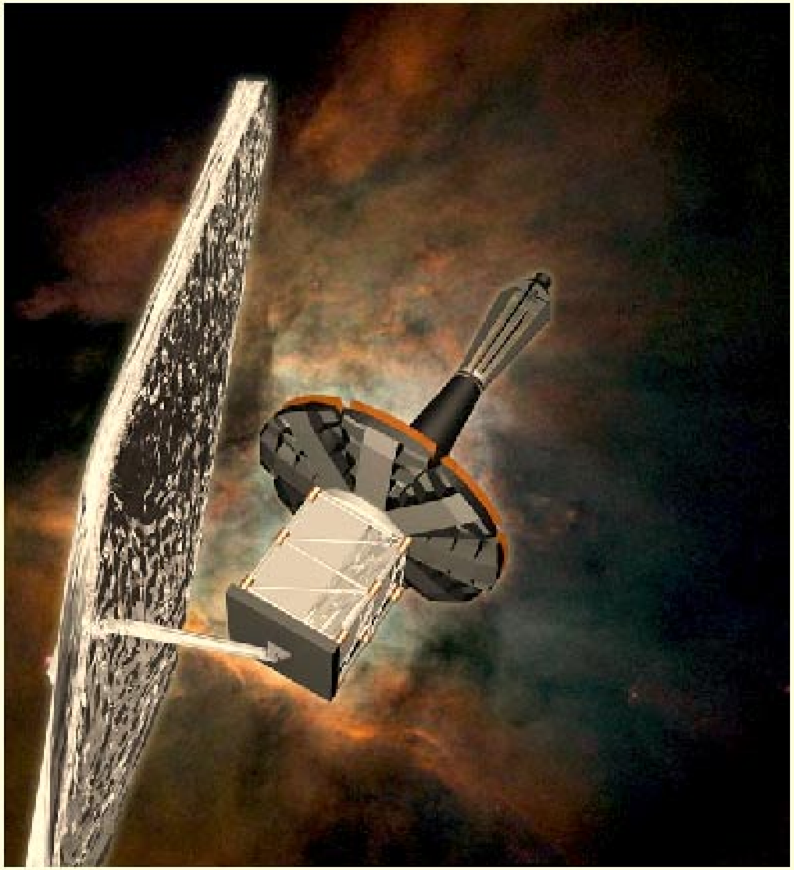
The CASSINI Spacecraft

Geant4 Simulation



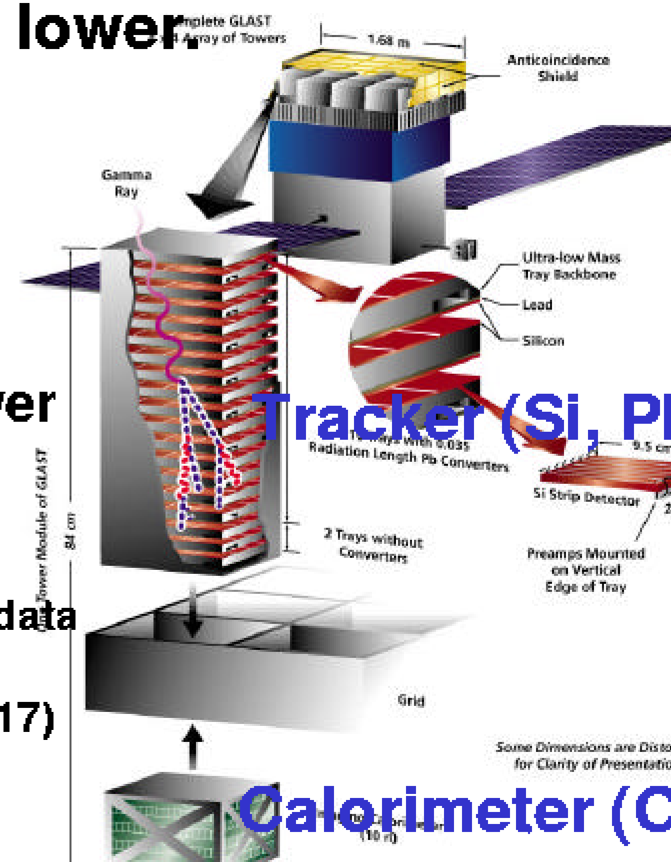
Spacecraft Modeling

- Secondary Environment depends on spacecraft structure and materials
- Spacecraft Modeling
  - Use CAD files where available to model spacecraft and instruments



Purpose and Plan of this Talk

We have validated EM processes in Geant4 important f gamma-ray satellite GLAST. Energy range of GLAST is 20 MeV - 300 GeV. So we need to validate physics processed down to ~10 MeV and lower.

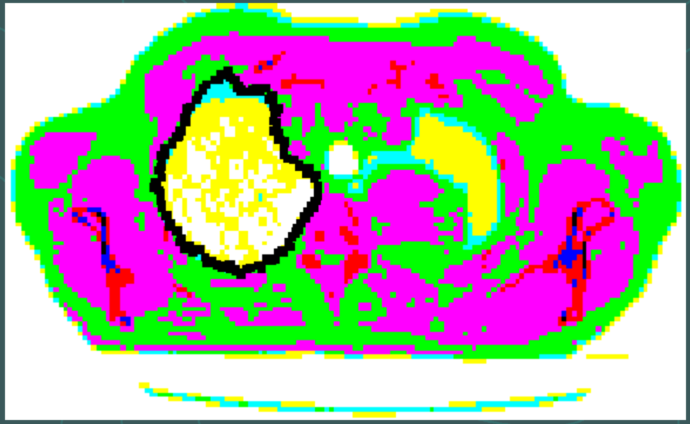


- Particle Ionization Energy Loss in Matters
  - Bethe-Bloch Formula (pp. 3-5)
  - Landau Distribution (pp. 3-5)
  - Range (p. 6)
- Processed related to electromagnetic shower
  - Pair creation (pp. 8-9)
  - Bremsstrahlung (p. 10)
  - Moller scattering/BhaBha scattering (pp. 11-12)
  - Comparison of shower profile with experimental data (pp. 13-14)
  - Comparison of shower profile with EGS4 (pp. 15-17)

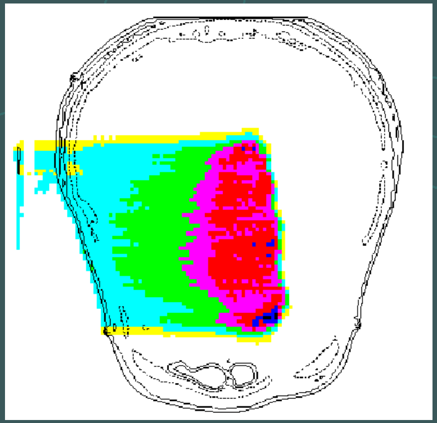
- We use Geant4 2.0

Treatment planning with Geant4

Geant interfaced to a file obtained with a Computer Tomography (CT) scan



Reconstruction of the lung contour (black pixels)

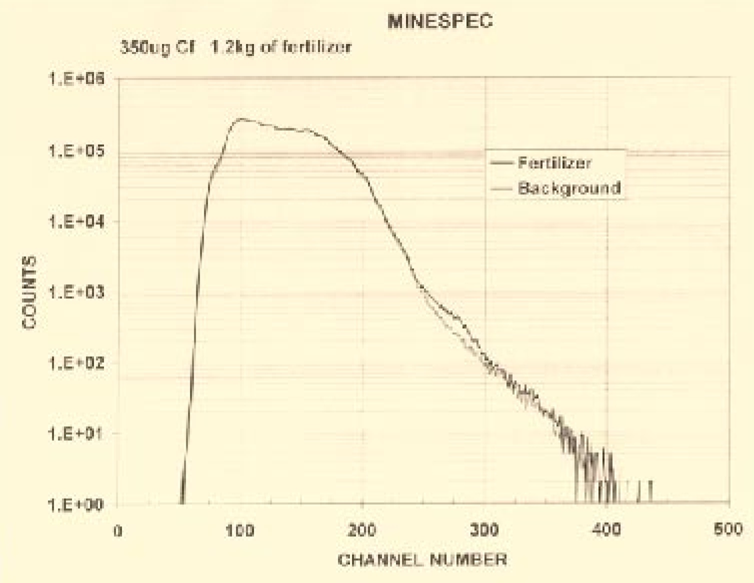


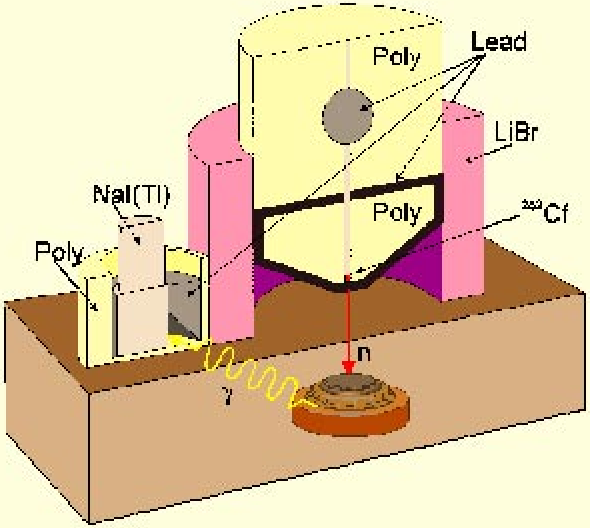
CT slice of a head with the dose deposition of a proton beam obtained with the GEANT code

Reconstruction of the optical nerves (black pixels)

Thermal Neutron Activation

- TNA detects explosive by properties of constituents
  - High concentration of N
  - Does not ID explosive
- Can confirm presence of all surface laid or shallow AT mines in few seconds to 1 minute
- AT up to 20 cm deep and large AP mines in < 5 minutes





Summary for introduction

- Geant4 is a worldwide collaboration providing a tool for simulation of particles interacting with matter.
- Geant4's object-oriented modular structure allows a large degree of functionality and flexibility.
- Geant4 provides sets of alternative physics models so that users can choose appropriate models.
- Geant4 is used by not only high energy and nuclear physics but also accelerator physics, astrophysics, space science and medical and other applications.

Basic concepts in Kernel

Run

- As an analogy of the real experiment in HEP, a run of Geant4 starts with “Beam On”.
- Within a run, the user cannot change
  - detector geometry
  - settings of physics processes

    ---> detector is inaccessible during a run
- Conceptually, a run is a collection of *events* which share the same detector and physics conditions.

Event

- At beginning of processing, an event contains primary particles. These primaries are pushed into a stack.
- When the stack becomes empty, processing of an event is over.
- G4Event class represents an event. It has following objects at the end of its processing.
  - List of primary vertexes and particles
  - Trajectory collection (optional)
  - Hits collections
  - Digits collections (optional)

Track

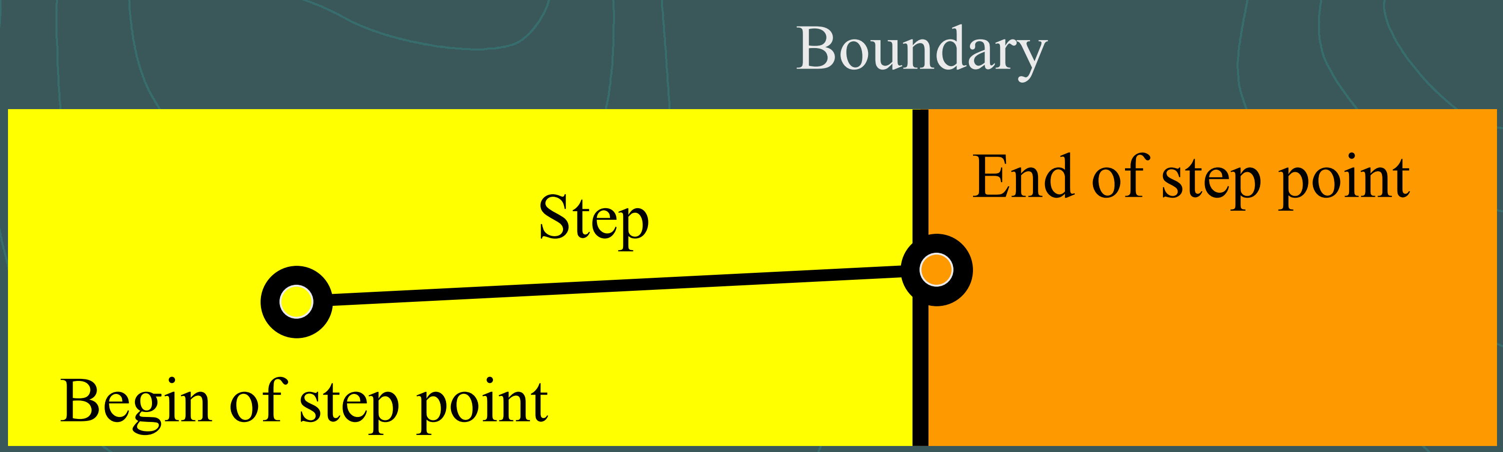
- Track is a snapshot of a particle.
- Step is a “delta” information to a track.
  - Track is not a collection of steps.
- Track is deleted when
  - it goes out of the world volume
  - it disappears (e.g. by decay or hard interaction)
  - it goes down to zero kinetic energy and no “at rest” additional process is required
  - the user decides to kill it

Track

- A track is made of three layers of class objects.
  - G4Track
    - Position, volume, track length, global ToF
    - ID of itself and mother track
  - G4DynamicParticle
    - Momentum, energy, local time, polarization
    - Pre-fixed decay channel
  - G4ParticleDefinition
    - Shared by all G4DynamicParticle of same type
    - Mass, lifetime, charge, other physical quantities
    - Decay table

Step

- 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. 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.



Trajectory

- Trajectory is a record of a track history. It stores some information of all steps done by the track as objects of G4TrajectoryPoint class.
- It is advised not to store trajectories for secondary particles generated in a shower because of the memory consumption.
- The user can create his own trajectory class deriving from G4VTrajectory and G4VTrajectoryPoint base classes for storing any additional information useful to the simulation.

Unit system in Geant4

- Geant4 has no default unit. To give a number, unit must be "multiplied" to the number.
  - for example :

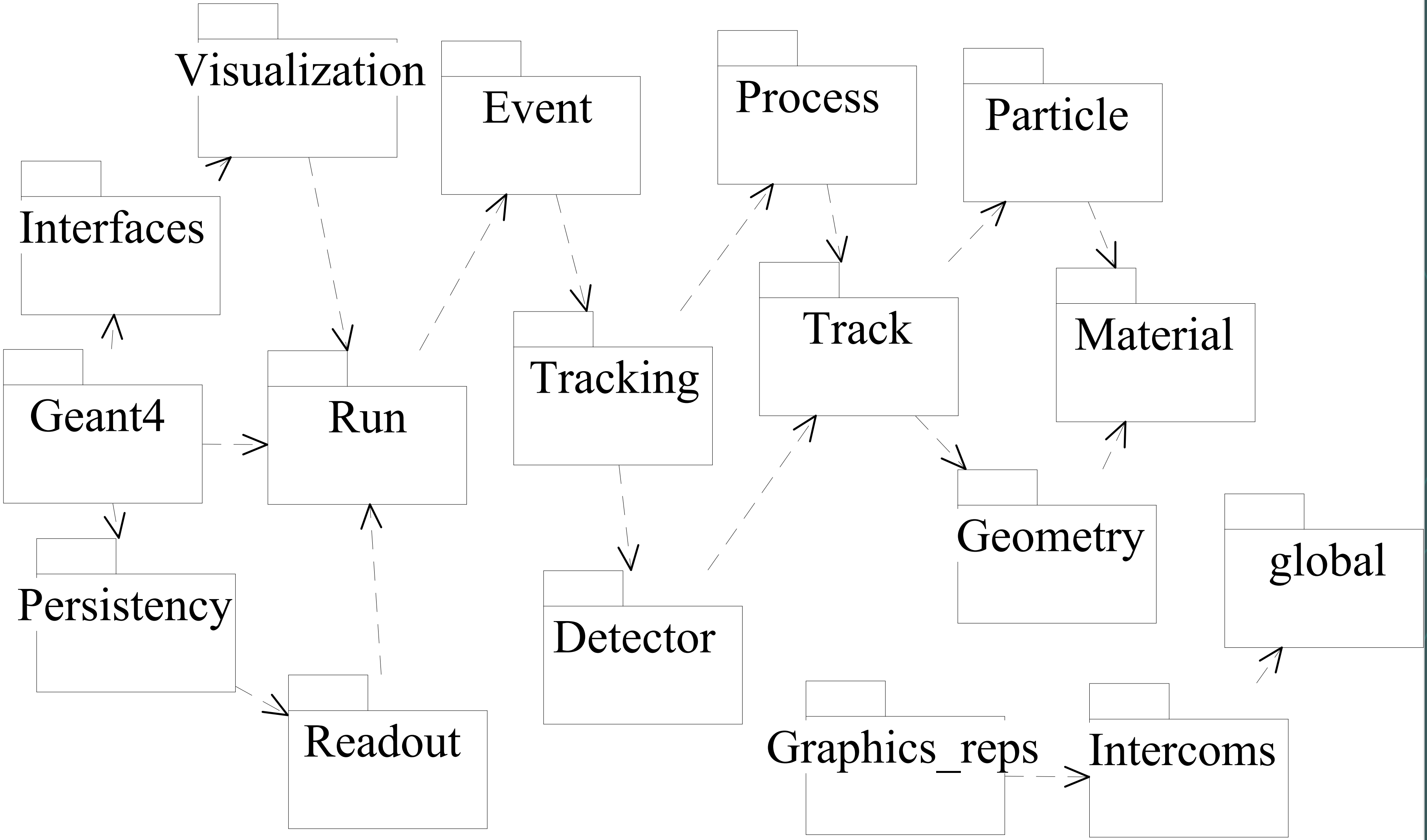
    ```
    double width = 12.5*m;
    double density = 2.7*g/cm3;
    ```

  - Almost all commonly used units are available.
  - The user can define new units.
  - Refer to SystemOfUnits.h
- Divide a variable by a unit you want to get.

  ```
  G4cout << dE / MeV << " (MeV)" << G4endl;
  ```

Structure of Geant4 toolkit

Global structure of Geant4

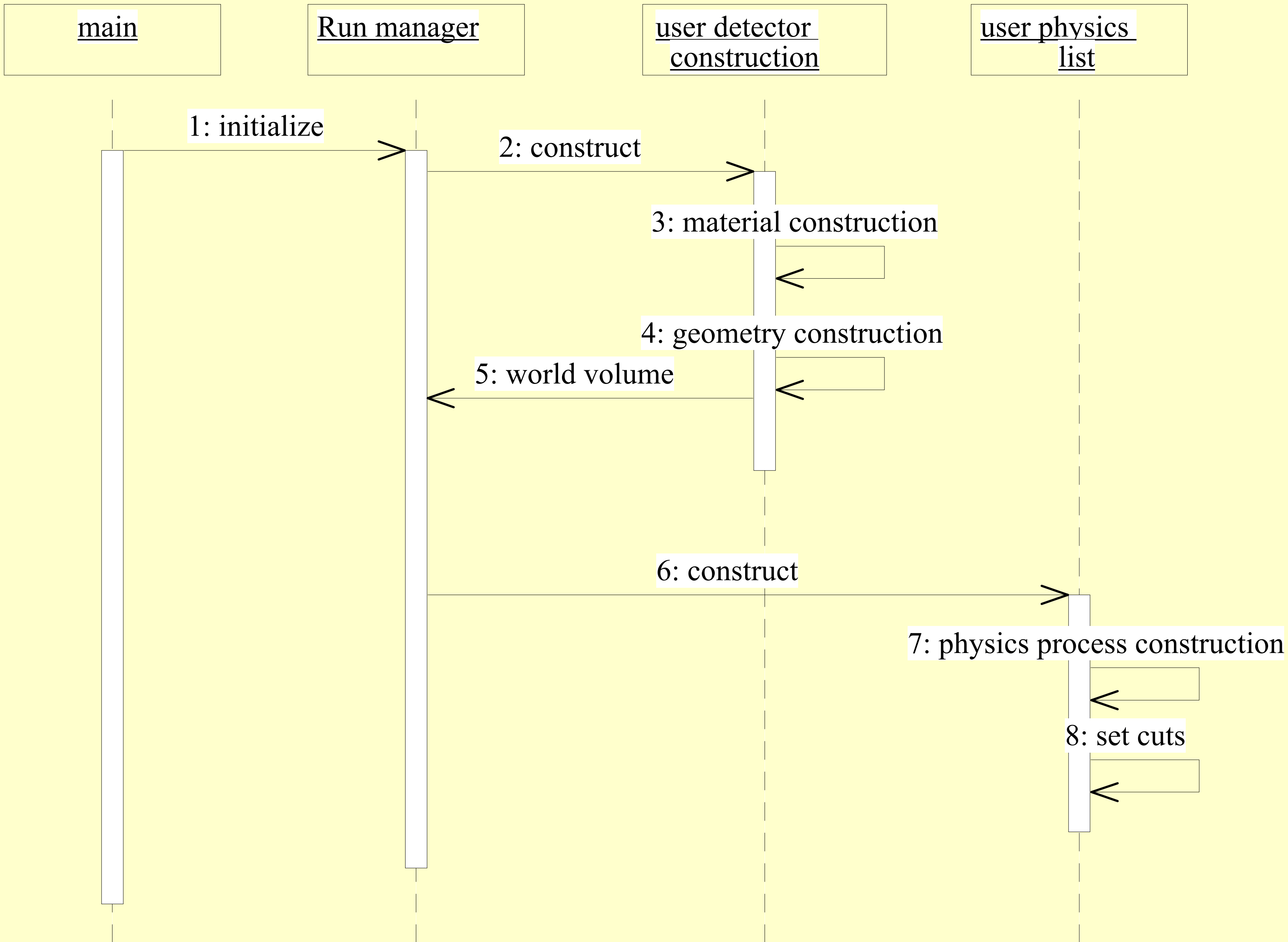


How Geant4 runs

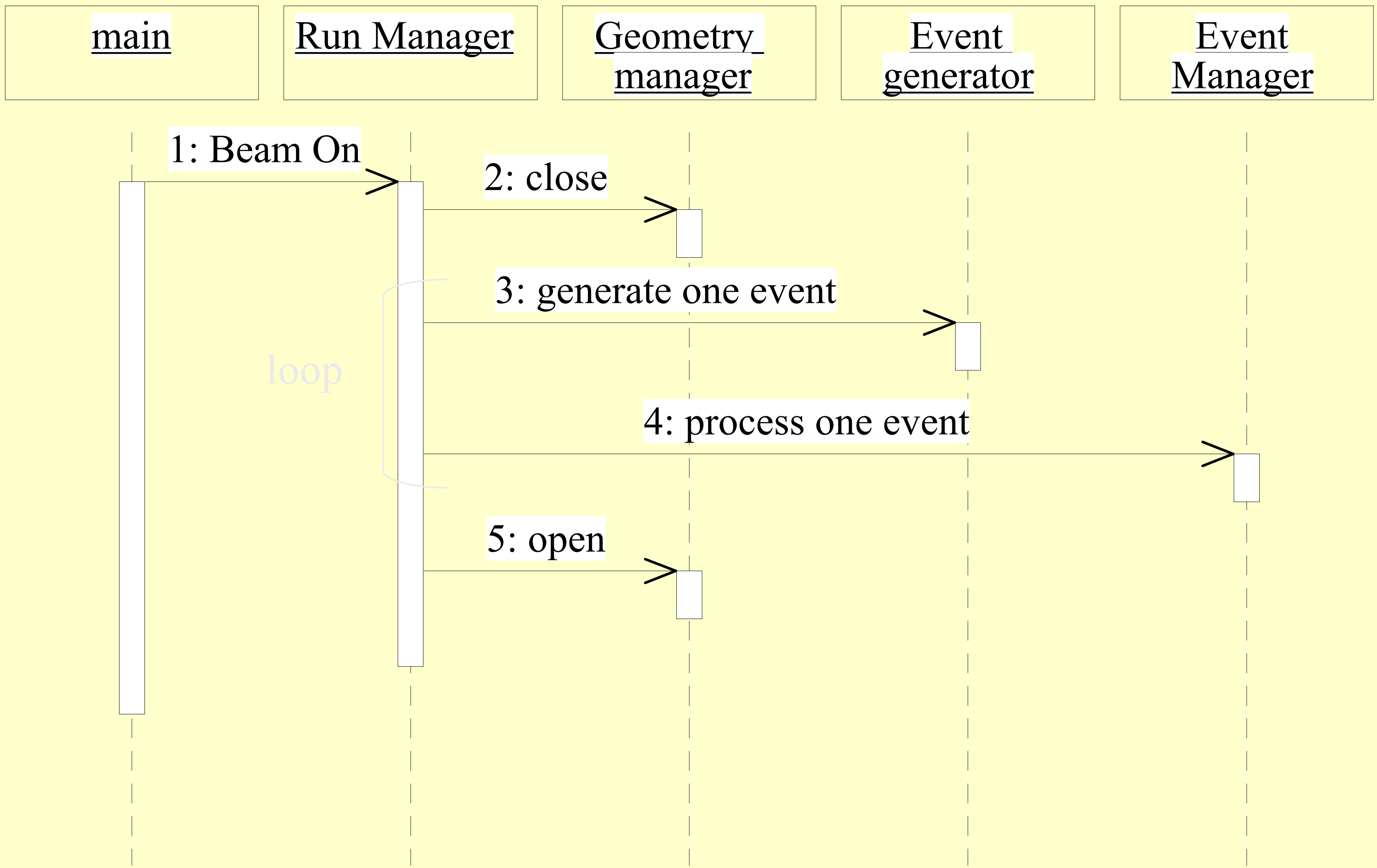
- Initialization
  - Construction of material and geometry
  - Construction of particles, physics processes and calculation of cross-section tables
- "Beam-On" = "Run"
  - Close geometry --> Optimize geometry
  - Event Loop

    ---> More than one runs with different geometrical configurations

Initialization

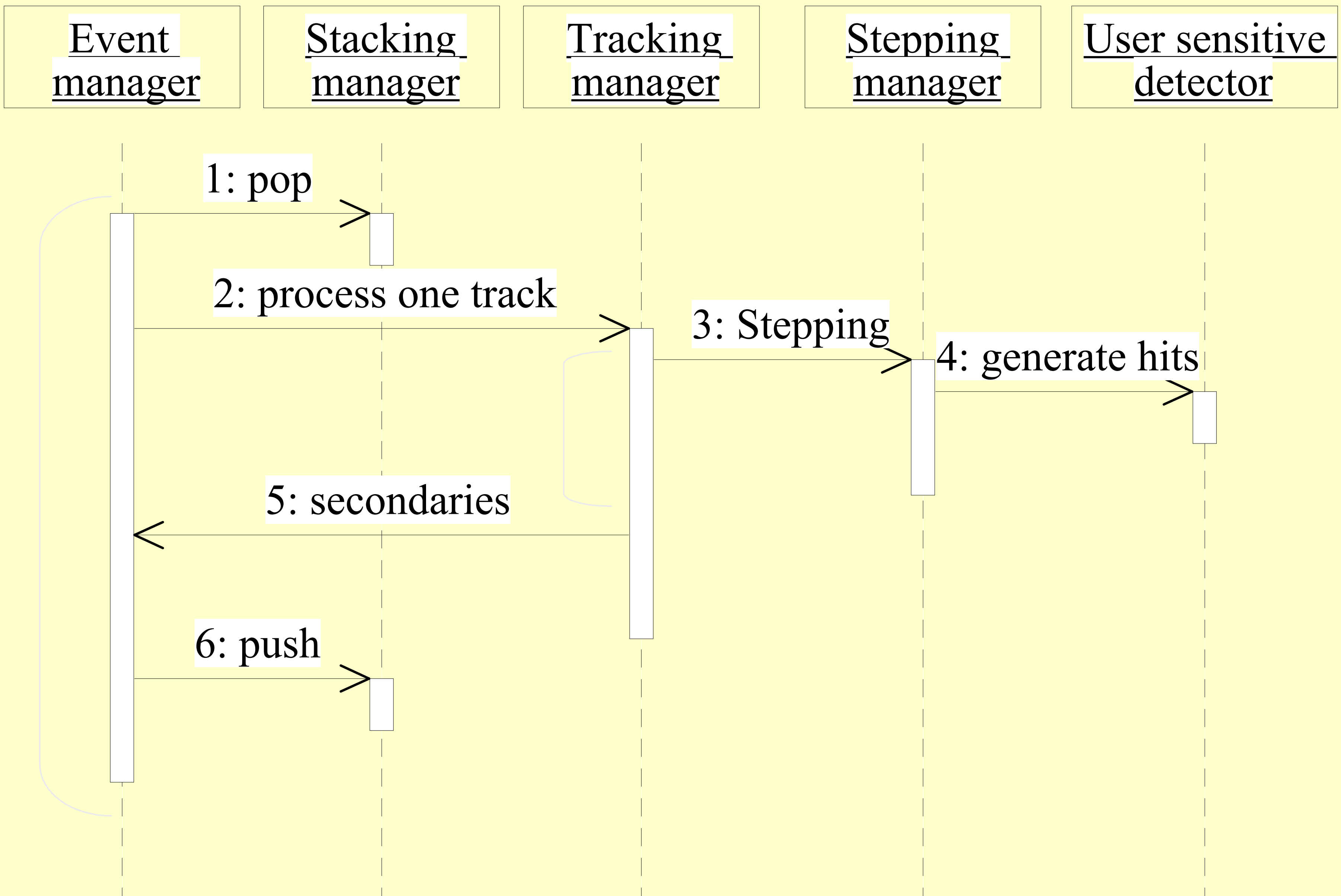


Beam on



main
Run Manager
Geometry_ manager
Event generator
Event Manager
1: Beam On
2: close
3: generate one event
loop
4: process one event
5: open

Event processing



User classes

- Initialization classes
  - Invoked at the initialization
    - G4VUserDetectorConstruction
    - G4VUserPhysicsList
- Action classes
  - Invoked during the event loop
    - G4VUserPrimaryGeneratorAction
    - G4UserRunAction
    - G4UserEventAction
    - G4UserStackingAction
    - G4UserTrackingAction
    - G4UserSteppingAction

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*(),
  - Construct all necessary materials
  - Construct volumes of your detector geometry
  - Construct your sensitive detector classes and set them to the detector volumes
- Optionally you can define visualization attributes 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
- Geant4 provides lots of utility classes/methods.

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 some generators.
  - G4ParticleGun
  - G4HEPEvtInterface, G4HepMCInterface
  - G4GeneralParticleSource

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
  - BeginOfRunAction(const G4Run*)
    - Define histograms
  - EndOfRunAction(const G4Run*)
    - Store histograms
- G4UserEventAction
  - BeginOfEventAction(const G4Event*)
    - Event selection
    - Define histograms
  - EndOfEventAction(const G4Event*)
    - Analyze the event

Optional user action classes

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

Optional user action classes

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

Environment variables

- You need to set following environment variables to compile, link and run Geant4-based simulation.
  - Mandatory variables
    - G4SYSTEM – OS (e.g. Linux-g++)
    - G4INSTALL – base directory of Geant4
    - G4WORKDIR – your temporary work space
    - CLHEP_BASE_DIR – base directory of CLHEP
  - Variable for physics processes
    - G4LEVELGAMMADATA – directory of PhotonEvaporation data
  - Additional variables for GUI/Vis/Analysis

Select (G)UI

- In your *main*(), according to your computer environments, construct a G4UIsession concrete class provided by Geant4 and invoke its *sessionStart*() method.
- Geant4 provides
  - G4UIterminal – (t)C-shell like character terminal
  - G4GAG -- Tcl/Tk or Java PVM based GUI
  - G4Wo -- Opacs
  - G4UIBatch -- Batch job with macro file

Visualization

- Derive your own concrete class from G4VVisManager according to your computer environments.
- Geant4 provides interfaces to graphics drivers
  - DAWN -- Fukui renderer
  - WIRED
  - RayTracer -- Ray tracing by Geant4 tracking
  - OPACS
  - OpenGL
  - OpenInventor
  - VRML

G4cout, G4cerr, G4endl

- G4cout, G4cerr and G4endl are iostream objects defined by Geant4. The user is recommended to use them instead of ordinary cout/cerr/endl. Don't forget to include "G4ios.hh".
- GUI manipulates output stream to store logs through G4cout etc., and ordinary cout/cerr/endl do not work properly for some GUI.
- G4cout/G4cerr should not be used in the constructor of a class if the instance of this class is intended to be used as "static". This restriction comes from the language specification of C++.
- "cin" should not be used. "cin" does not work properly for some GUI. Use intercoms.

Very basics of geometry

Definition of material

- In nature, materials are made of elements, and elements are made of isotopes. In Geant4, G4Material, G4Element and G4Isotope represent them.
- G4Element and G4Isotope describe properties of atom.
  - Atomic number, number of nucleons, mass of a mole
  - Shell energies, quantities e.g. cross-sections per atom
- G4Material describes macroscopic properties of matter
  - Density, state, temperature, pressure
  - Radiation length, mean free path
- Note that there should not be an absolute vacuum. Use very thin density material.

Definition a simple material

- Lead can be defined as follows.

```
G4doule density = 11.35*g/cm3;
G4double atomicNumber = 82.;
G4double massOfMole = 207.19*g/mole;
G4Material* Pb
   = new G4Material("Lead", atomicNumber,
                    massOfMole, density);
```

Definition of a molecule

- Water, which is made of hydrogen and oxygen, can be defined as follows.

```
G4double aH = 1.01*g/mole;
G4Element* H  = new G4Element("Hydrogen","H" , 1., aH);
G4double aO = 16.00*g/mole;
G4Element* O  = new G4Element("Oxygen","O" , 8., aO);
G4double density = 1.000*g/cm3;
G4Material* H2O = new G4Material("Water", density, 2);
H2O->AddElement(H, 2);
H2O->AddElement(O, 1);
```

Number of elements

Number of atoms (integer)

Definition of a mixture by fractional mass

- Air, which is composed of nitrogen and oxygen, can be defined as follows.

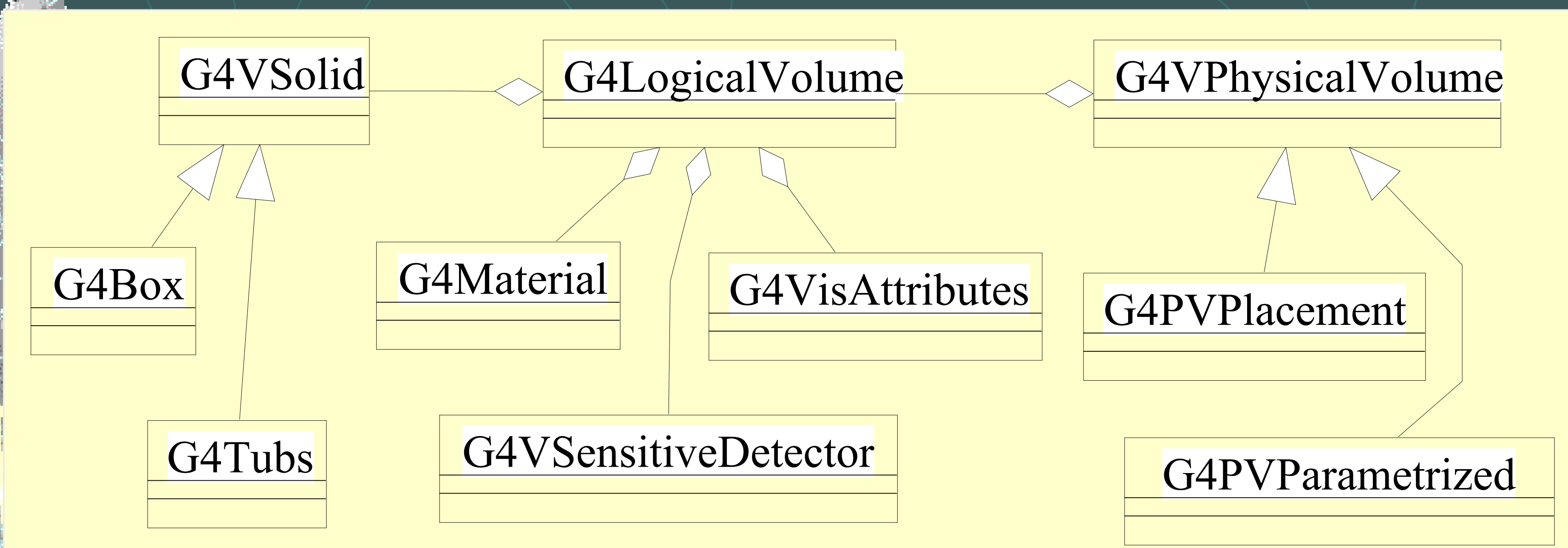
```
G4double aN = 14.01*g/mole;
G4Element* N  = new G4Element("Nytrogen", "N" ,7., aN);
G4double aO = 16.00*g/mole;
G4Element* O  = new G4Element("Oxygen","O" , 8., aO);
G4double density = 1.29*mg/cm3;
G4Material* Air = new G4Material("Air", density, 2);
Air->AddElement(N, 0.7);
Air->AddElement(O, 0.3);
```

Number of elements

Fraction of mass (double)

Description of geometry

- Three conceptual layers
  - G4VSolid -- shape, size
  - G4LogicalVolume -- daughter phys. volumes, material, sensitivity, user limits, etc.
  - G4VPhysicalVolume -- position, rotation



Description of geometry – basic strategy

- Basic strategy

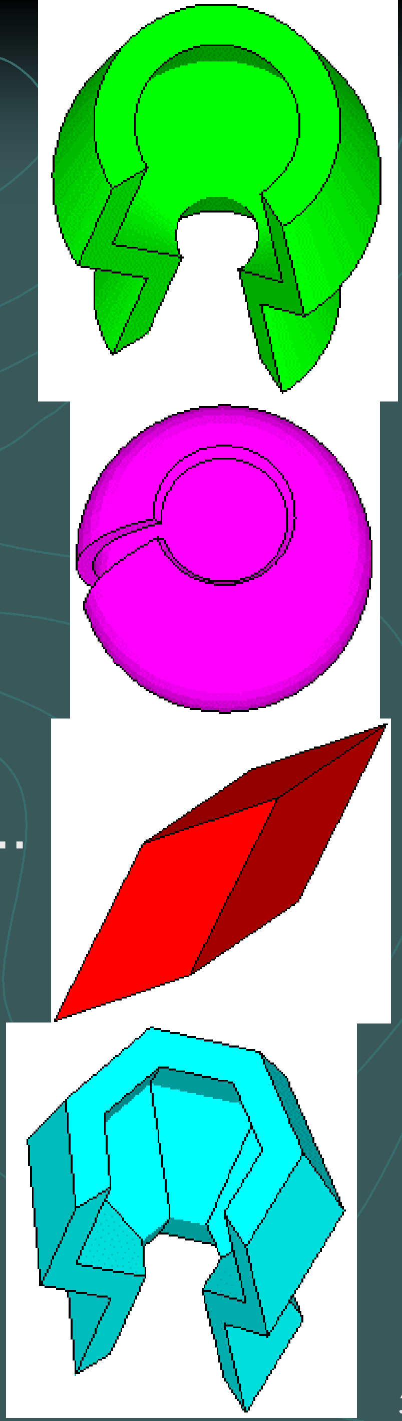
```
G4VSolid* pBoxSolid =
  new G4Box("aBoxSolid", 1.*m, 2.*m, 3.*m);
G4LogicalVolume* pBoxLog =
  new G4LogicalVolume( pBoxSolid, pBoxMaterial,
                       "aBoxLog", 0, 0, 0);
G4VPhysicalVolume* aBoxPhys = new G4PVPlacement(
      pRotation, G4ThreeVector(posX, posY, posZ),
      pBoxLog, "aBoxPhys", pMotherLog, 0, copyNo);
```

- A unique physical volume which represents the experimental area must exist and it fully contains all of other components.

  ---> The world volume

G4VSolid

- Types of solid
  - CSG (Constructed Solid Geometry) solids
    - G4Box, G4Tubs, G4Cons, G4Trd, …
  - Specific solids (CSG like)
    - G4Polycone, G4Polyhedra, G4Hype, …
    - Usage is CSG, while implementation is BREP
  - BREP (Boundary REPresented) solids
    - G4BREPSolidPolycone, G4BSplineSurface, …
    - Any order surface
  - Boolean solids
    - G4UnionSolid, G4SubtractionSolid, …
  - STEP interface
    - BREP solid models from CAD file
    - STEP compliant solid modeler

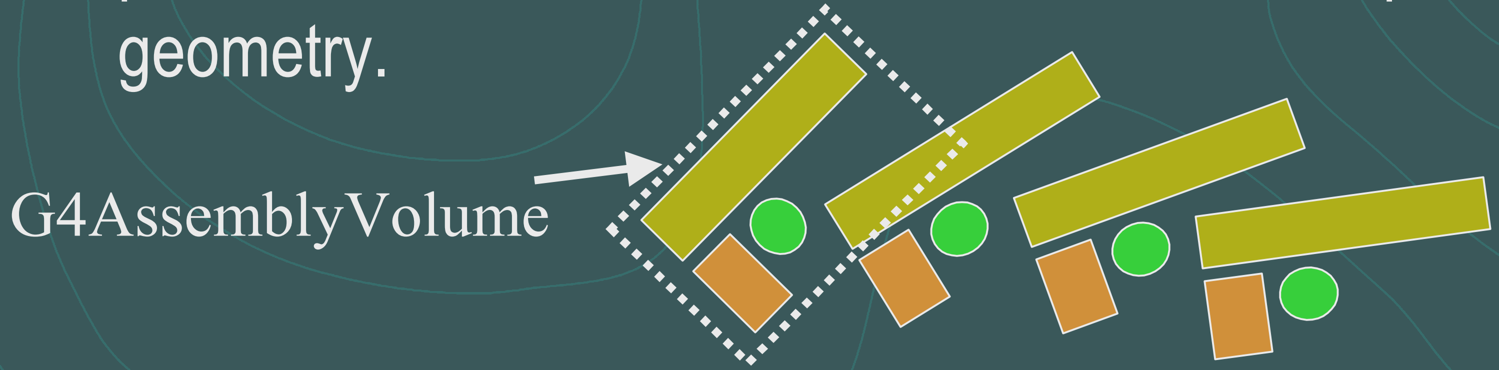


G4VPhysicalVolume

- G4PVPlacement 1 placement = 1 volume
  - A volume instance positioned once
- G4PVParameterised 1 placement = many volumes
  - Parameterized by copy number
  - Shape, size, material, position, rotation, color and sensitivity can be parameterized by implementing a concrete class of G4VPVParameterisation.
  - Parameterization can be used only for volumes that either have no further daughters or are identical in shape and size
- G4PVReplica 1 placement = many volumes
  - Sliced pieces of volumes without any gap space in between

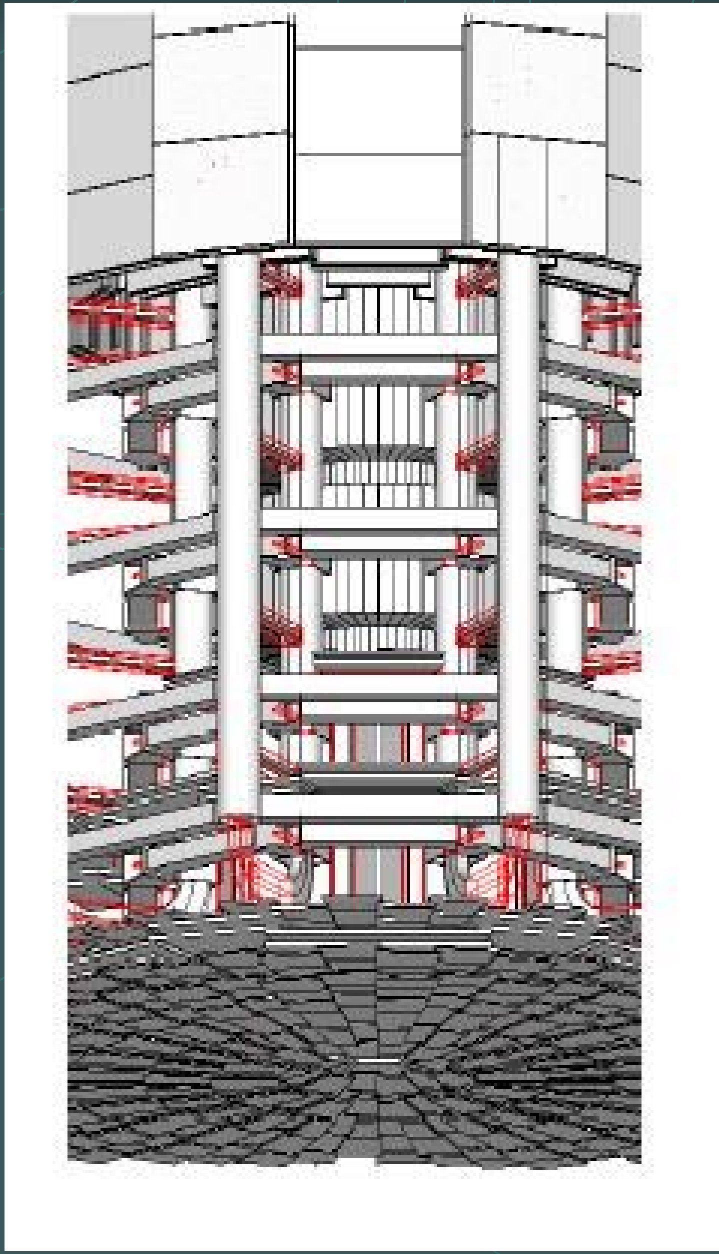
Overlapping volumes

- Geant4 does not allow volumes to overlap to each other.
  - A daughter must be fully contained by its mother.
  - Volumes in same level cannot share any space point.
- To describe complicated geometry, G4AssemblyVolume can be utilized.
  - Given G4AssemblyVolume is implemented as combination of placements, for memory consumption, parameterized volume is advised for most complicated geometry.



Geometry debugging tools

- Geant4 does not allow volumes to overlap to each other.
- Geant4 provides geometry debugging tools of alternative approaches for the user's convenience.
  - If you have odd results, it is advised to use these tools for checking your geometry description.



References to further information

References to further information

- Geant4 user's guides
  - http://geant4.web.cern.ch/geant4/G4UsersDocuments/Overview/html/index.html
- Further tutorial materials with hands-out exercises
  - http://geant4.web.cern.ch/geant4/meetings/users2002/schedule.html
- Full set of tutorial materials
  - http://geant4.slac.stanford.edu/UsersWorkshop/schedule.html
- Geant4 HyperNews
  - http://geant4-hn.slac.stanford.edu:5090/Geant4-HyperNews/index
- Geant4 training materials
  - http://geant4.web.cern.ch/geant4/milestones/training/training-milestone.html