In the name of God, the compassionate, the merciful

Simulations:
from proton-proton collisions
to particle interactions with detectors

Mohammad Sedghi

msedghi@cernn.ch

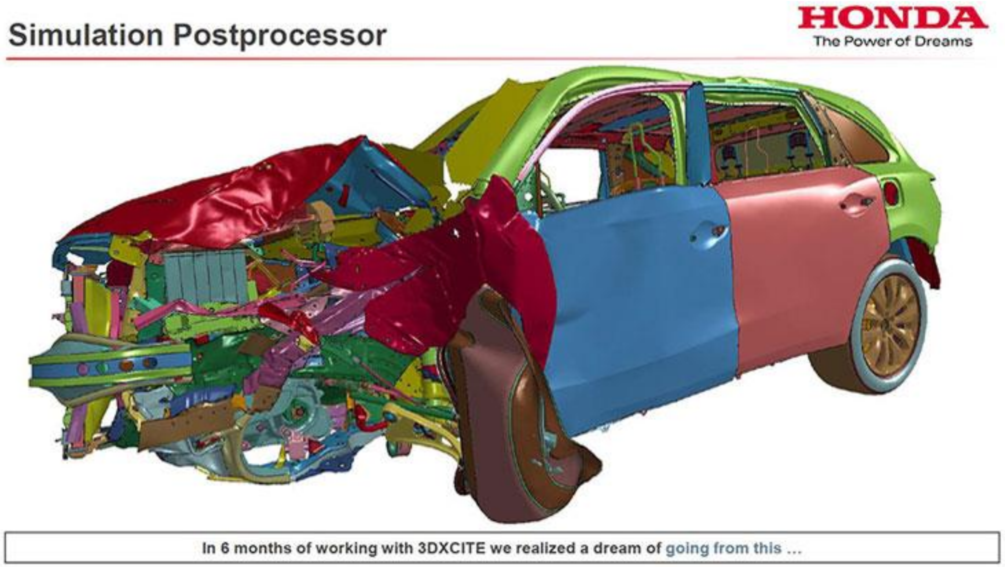
Department of Electrical and Computer Engineering
Isfahan University of Technology

Why do we do simulations?

- Simulation: the re-creation of real world process in a controlled environment.
- Mode: a representation of an object or process that describes and explains phenomenon when it cannot be experimented directly.

Simulation is a very useful, essential tool in modern particle physics

- **designing** an experiment (e.g. now ILC/CLIC, FCC)
- **analyzing** the data (e.g. now LHC experiments)



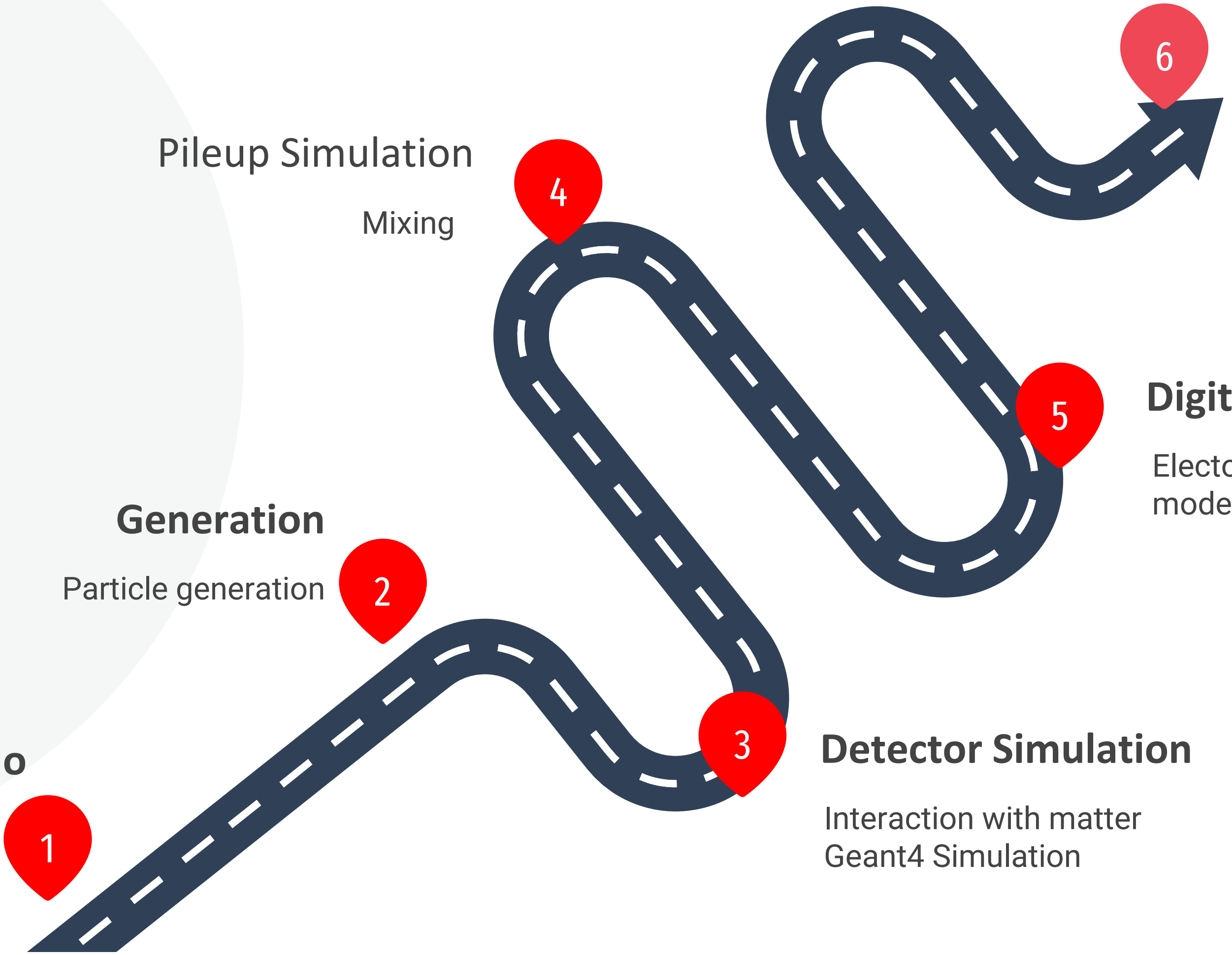
Simulation Postprocessor

In 6 months of working with 3DXCITE we realized a dream of going from this ...

DELTAGEN

To this...

Simulation steps



monte carlo

An introduction

Generation

Particle generation

Detector Simulation

Interaction with matter
Geant4 Simulation

Pileup Simulation

Mixing

Digitization

Electonics response modeling

Validation & Analsis

Validation with real data

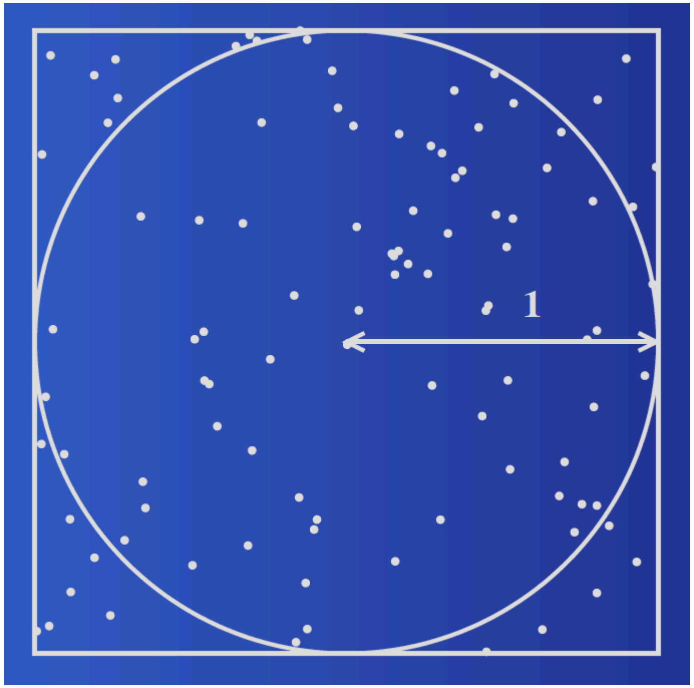
For preparation of this presentation, I owe to many contributors who I borrowed some slides from them. The list and acknowledgments are at the last slide.

Laplace's method of calculating π (1886)

- General idea is "Instead of performing long complex calculations, perform large number of experiments using random number generation and see what happens"

- Area of the square = 4
  - Area of the circle = π
  - Probability of random points inside the circle = $\pi / 4$
- Random points : *N*
  - Random points inside circle : *Nc*

  $\pi \approx 4\, Nc / N$



Area = (# Hits)/(# Total) x total area

Monte Carlo method : definition

- The Monte Carlo method is a stochastic method for numerical integration.
- Generate N random points x_i in the problem space
- Calculate the "score" $f_i = f(x_i)$ for N points.
- Calculate

$$\langle f \rangle = \frac{1}{N}\sum_{i=1}^{N} f_i \qquad \langle f^2 \rangle = \frac{1}{N}\sum_{i=1}^{N} f_i^2$$

- According to central limit theorem, for large N, then $\langle f \rangle$ will approach the true value f.

$$p(\langle f \rangle) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(\frac{-(\langle f \rangle - \bar{f})^2}{2\sigma^2} \right), \qquad \sigma^2 = \frac{\langle f^2 \rangle - \langle f \rangle^2}{N-1}$$

Monte Carlo event generators

- Monte Carlo event generators are essential for experimental particle physics.

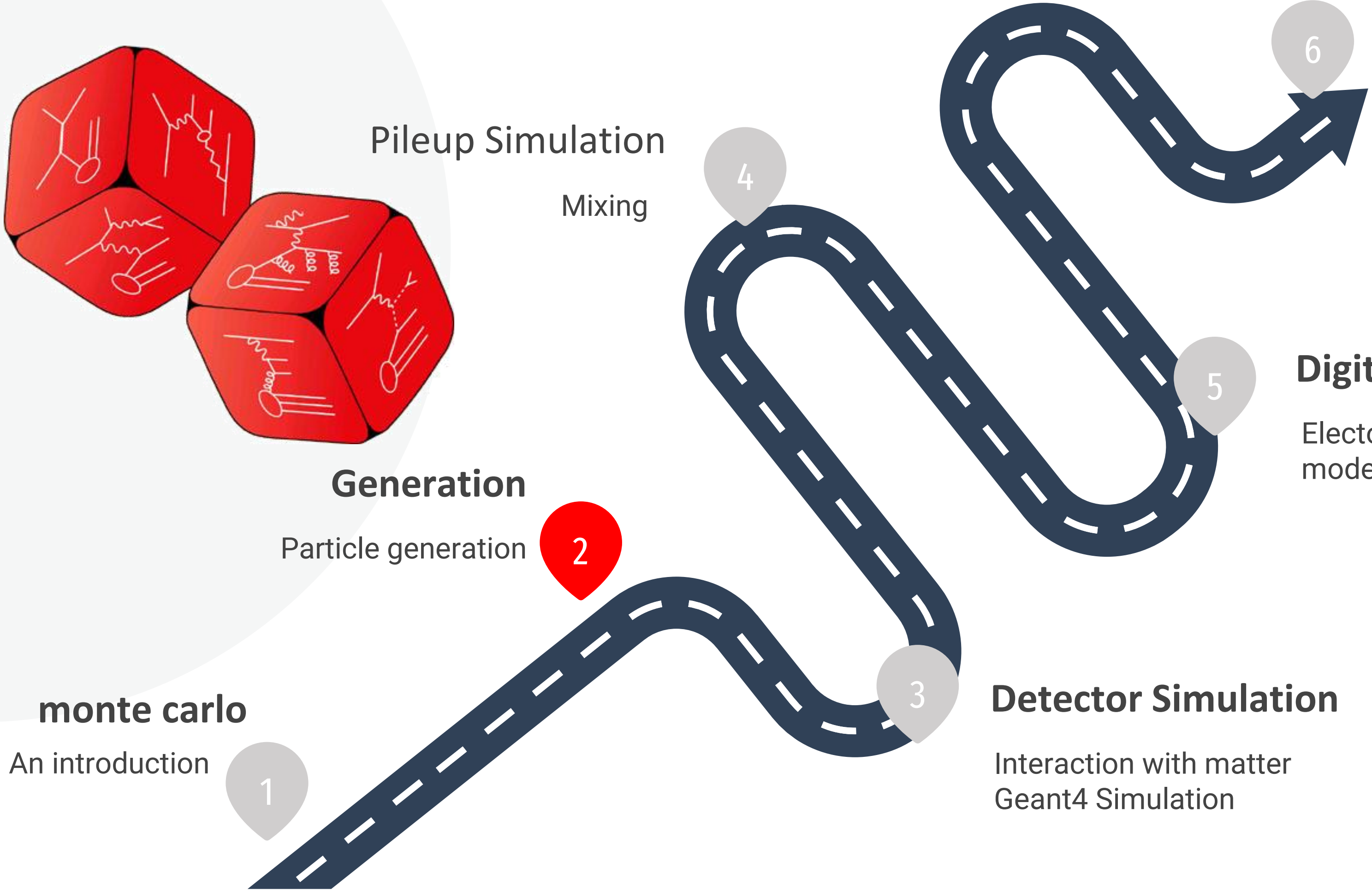
- They are used for:
  - Comparison of experimental results with theoretical predictions;
  - Studies for future experiments.

- Often these programs are ignored by theorists and treated as black boxes by experimentalists.

- It is important to understand the assumptions and approximations involved in these simulations.

- Monte Carlo simulations can be used to simulate a wide range of processes.

Simulation steps



monte carlo

An introduction

Generation

Particle generation

Detector Simulation

Interaction with matter
Geant4 Simulation

Pileup Simulation

Mixing

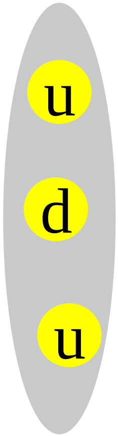
Digitization

Electonics response modeling

Validation & Analsis

Validation with real data

Parton Distribution Functions



- An illustrative simple model f
  - 3 quarks u u d
  - 2/3 chance of getting up quark
  - 1/3 chance of getting down quark
- Guess each carries 1/3 of momentum

- Defiantly a more accurate estimation is needed for the chance

- Need to multiple matrix element by probability f(x) of finding parton i with fraction of momentum x

- **Parton Distribution Functions**
  - Parton distribution functions give the probability to find partons in a hadron as a function of the fraction x of the proton's momentum carried by the parton.

Les Houches Accord PDF

- Various models for Parton distribution functions
  - PDF: Parton distribution functions

- 2001 Les Houches meeting – LHAPDF interface was conceived to
  enable the usage of PDF sets with uncertainties in a uniform manner.

- Using LHAPDF routines to evaluate PDFs

- Many PDF sets are now available in

https://lhapdf.hepforge.org/

MC event generation

- HEP MC event generation can typically be split into the following steps:

1) Process level calculations, needing
   - Matrix Element calculations
     - Parton distribution function evaluation

There are special purpose software for this step, e.g. Madgraph

2) Parton Shower
3) Hadronization

evolve at the parton-level to its final state

- All these latter steps rely heavily on models and are generally independent from the Matrix Element calculation.

- Therefore only few, typically multi-purpose event generators, implement those additional steps.
  - Examples are
    - Pythia (6 and 8)
    - Herwig (Fortran and C++ versions)
    - Sherpa

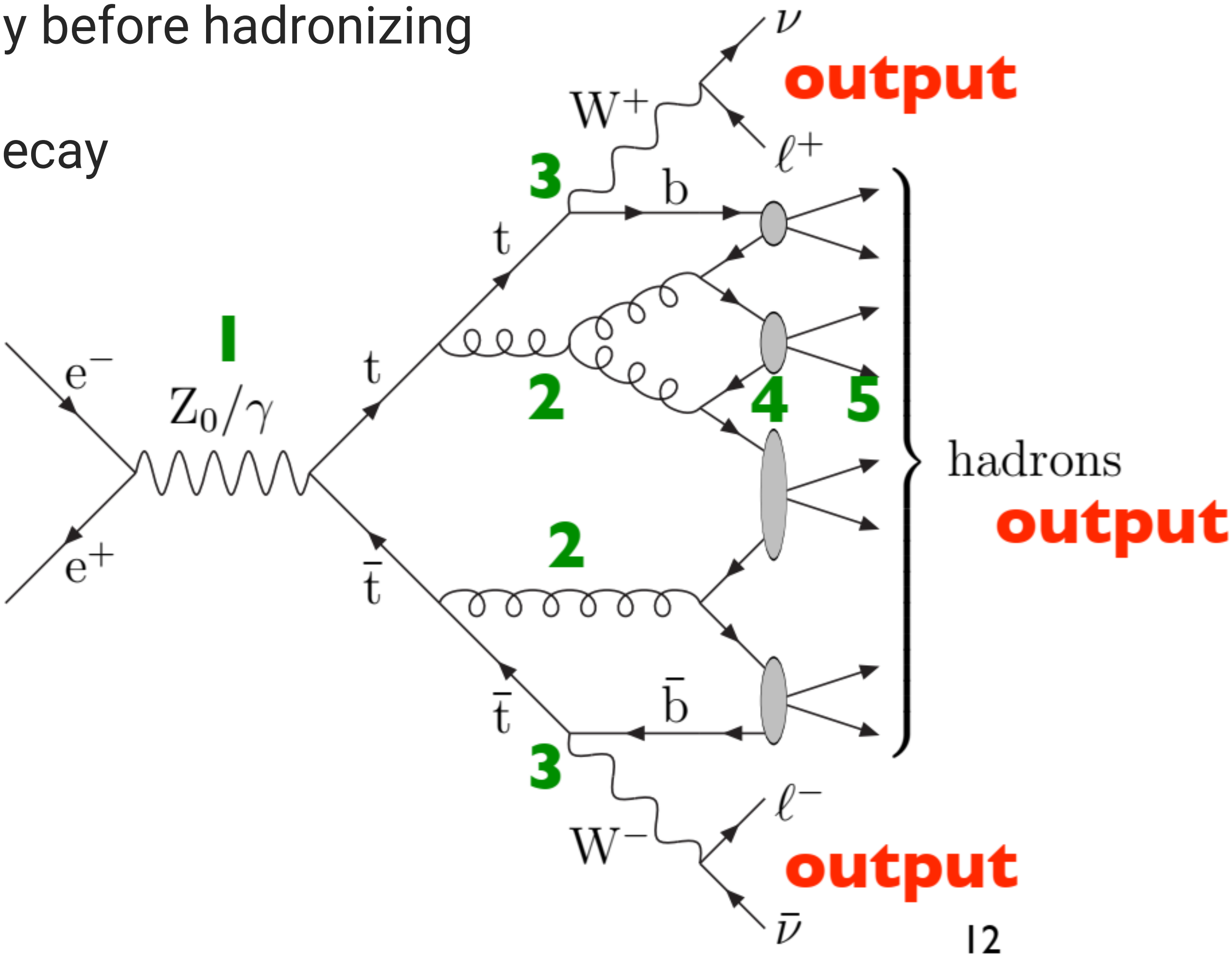
Pythia

- Pythia is a general-purpose Monte Carlo event generator that has usually been used
  in the analyses of particle collisions in high-energy physics.

- It contains theory and models for a number of physics aspects, including
  - hard and soft interactions,
  - parton distributions,
  - initial- and final-state parton showers,
  - multiple interactions,
  - fragmentation and decay.

Pythia Monte Carlo event generator process

1) Hard process (ME ⊕ PDF)
2) Parton-shower phase
3) Hard particles decay before hadronizing
4) Hadronization
5) Unstable hadrons decay



Interface standard between generator

- You may need more than a MC generator to generate particles for the new physics.
- For example you may need mass spectrum from one generator and hadronization process from another.
  - How various generators should talk together?

  - The `Les Houches Events` (LHE) file format is an agreement between Monte Carlo event generators and theorists to define **Matrix Element** level event listings in a common language.

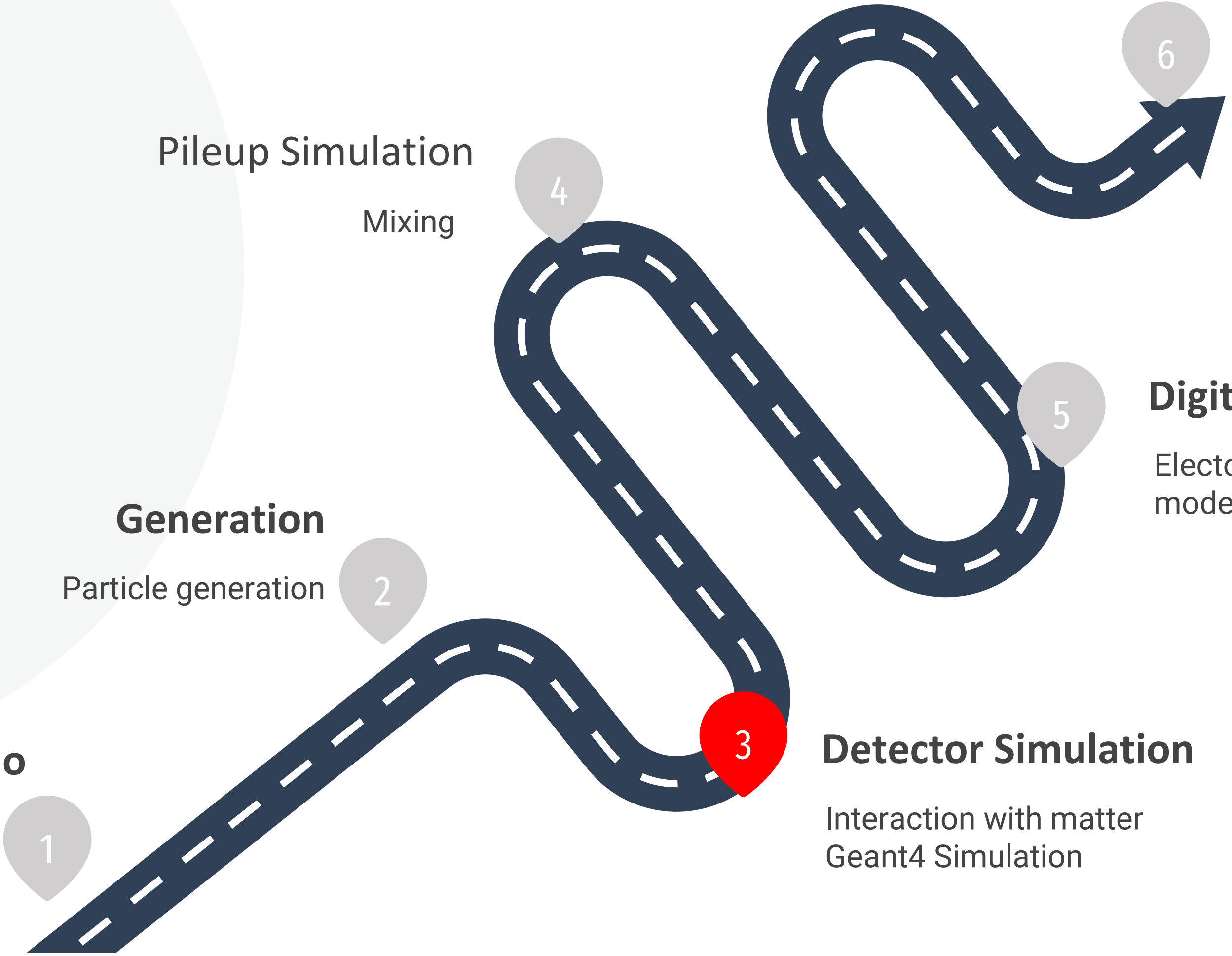
Madgraph and Dire

- Dire
  - Dire is available as a plugin to Pythia and Sherpa for parton showering.

- Madgraph
  - Madgraph is a matrix-element generator that can be interfaced with parton shower from an MC event generator.
  - It creates matrix elements for events with more than 2 outgoing particles, such as 2 → 3 events.
  - This helps Pythia and other MC event generators that generate only 2 → 2 events.
  - Returns Feynman diagrams
  - Self-Contained Fortran Code for |M|^2
  - compute a cross-section

- In case we just want to compute a cross-section at the parton level, it suffices to use the basic MadGraph software.

Simulation steps



monte carlo

An introduction

Generation

Particle generation

Detector Simulation

Interaction with matter
Geant4 Simulation

Pileup Simulation

Mixing

Digitization

Electonics response modeling

Validation & Analsis

Validation with real data

Monte Carlo radiation transportation codes

- The detector simulation is different for each experiment. However, general codes exist that can be used for simulating any detector
- These general codes, e.g. Geant4, are called "**Monte Carlo radiation transportation codes**"

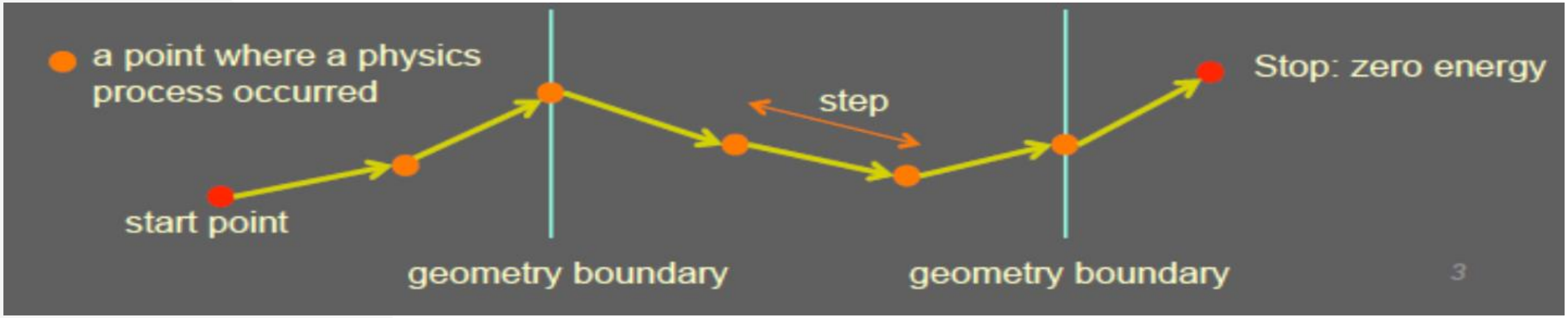
- Non-deterministic (e.g. do not solve equations); use random numbers to reproduce distributions
- Transport particles through matter

Simulation engines
– Geant4 and Fluka are well established packages

Geant4

Geant4 is a toolkit to simulate the interaction of particles in matter, created by the Geant4 Collaboration.

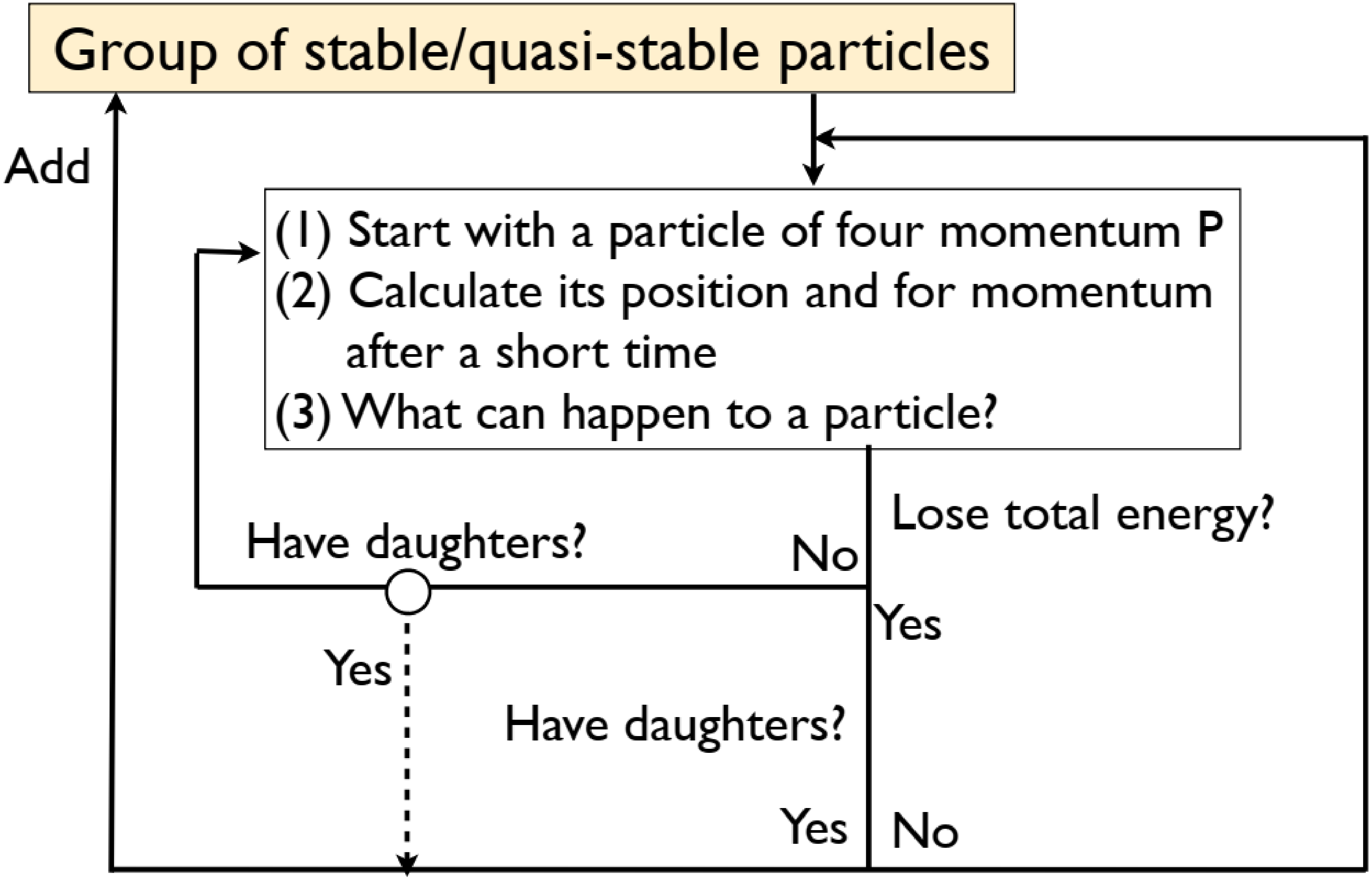
How does it work?



- Treat one particle at the time
- Treat a particle in steps
- For each step
- the step length is determined by the cross sections of the physics processes and the geometrical boundaries; if new particles are created, add them to the list of particles to be transported;
  - local energy deposit; effect of magnetic and electric fields;
  - if the particle is destroyed by the interaction, or it reaches the end of the apparatus, or its energy is below a (tracking) threshold, then the simulation of this particle is over; else continue with another step.
- Output - new particles created (indirect)

\- local energy deposits throughout the detector (direct)

How does MC work in detector simulation



Group of stable/quasi-stable particles

Add

(1) Start with a particle of four momentum P

(2) Calculate its position and for momentum after a short time

(3) What can happen to a particle?

Have daughters?

No

Lose total energy?

Yes

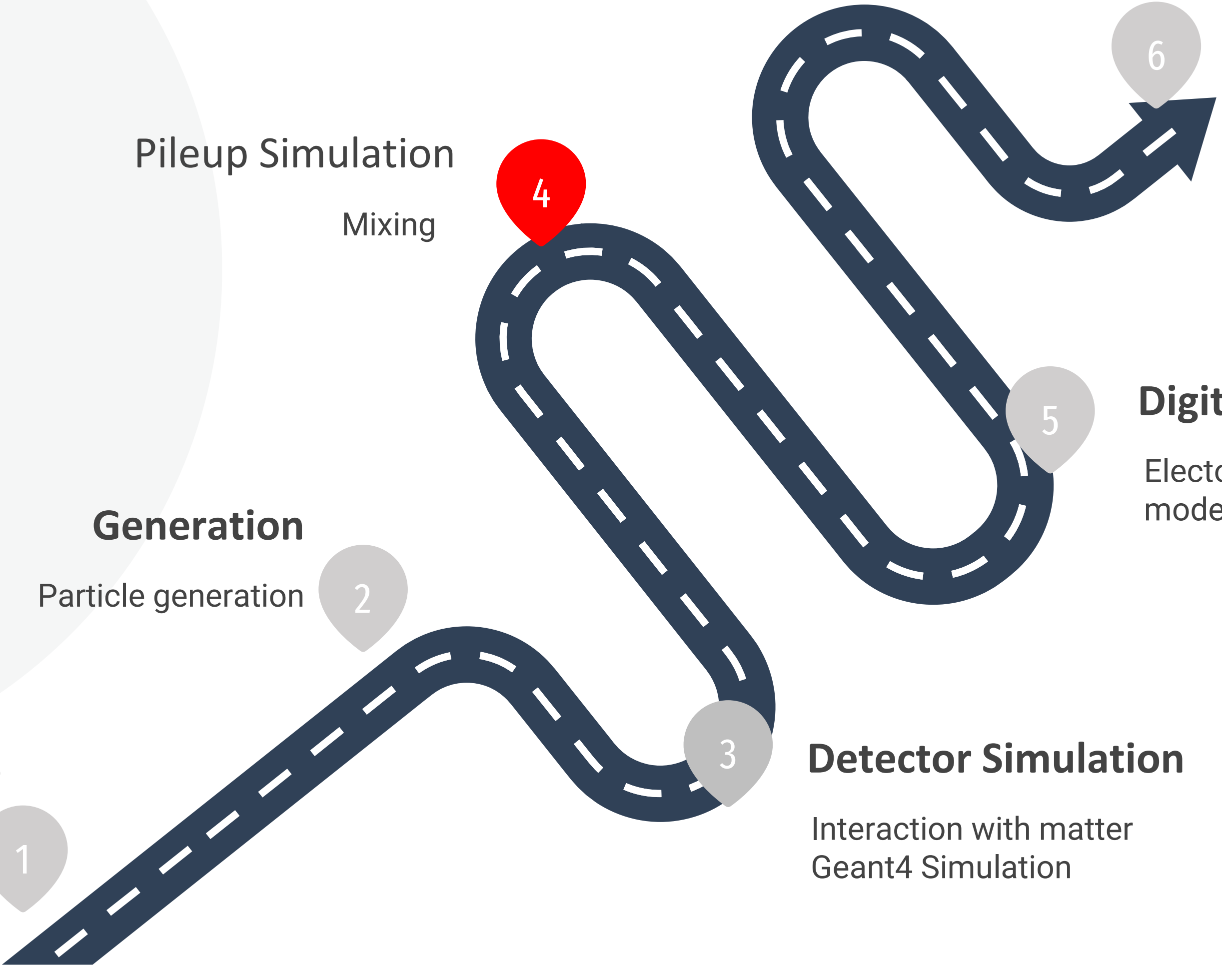
Yes

Have daughters?

Yes

No

Simulation steps



monte carlo

An introduction

Generation

Particle generation

Detector Simulation

Interaction with matter
Geant4 Simulation

Pileup Simulation

Mixing

Digitization

Electonics response modeling

Validation & Analsis

Validation with real data

Pileup simulation

- Particle accelerators are designed to deliver two parameters to the HEP user
  - Energy
  - Luminosity (L)
    - Measure of collision rate per unit area
    - Event rate for a given event probability (“cross-section”) given by: $\boxed{R = \mathscr{L}\sigma}$

  - Suppose L_b corresponds to the luminosity of one pp head-on collision
    - Or the so-called one bunch-crossing

    Then L_b dc is proportional to the mean number of interaction per pp collision.

$$\mu = \langle N_{\text{int.}} \rangle = \frac{\mathcal{L}_b \sigma}{f_{rev}}$$

where f_r = 11245.6 Hz is the LHC revolution frequency during collisions,

σ is the total interaction cross section.

At the LHC, L_b is typically expressed in units of Hz/μb ≡ 10^{30} cm^{-2} s^{-1}.

- the mean number of interaction per pp collision (μ) is called pileup.
- At HL-LHC $\mu \approx 200$ → ~ L_b =28 Hz/μb

Event mixing

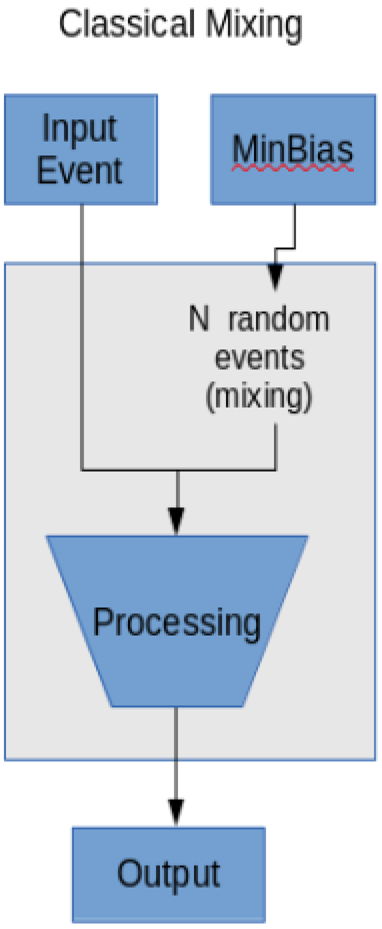
- LHC will produce minimum bias interactions per bunch crossing on average
  - Phase I, run-2 : ~3 or ~30 (nominal)
  - Phase II, ~140 (physics) or ~200 (high lumi.)

- pileup events simulated independently from physics events by **Mixing**

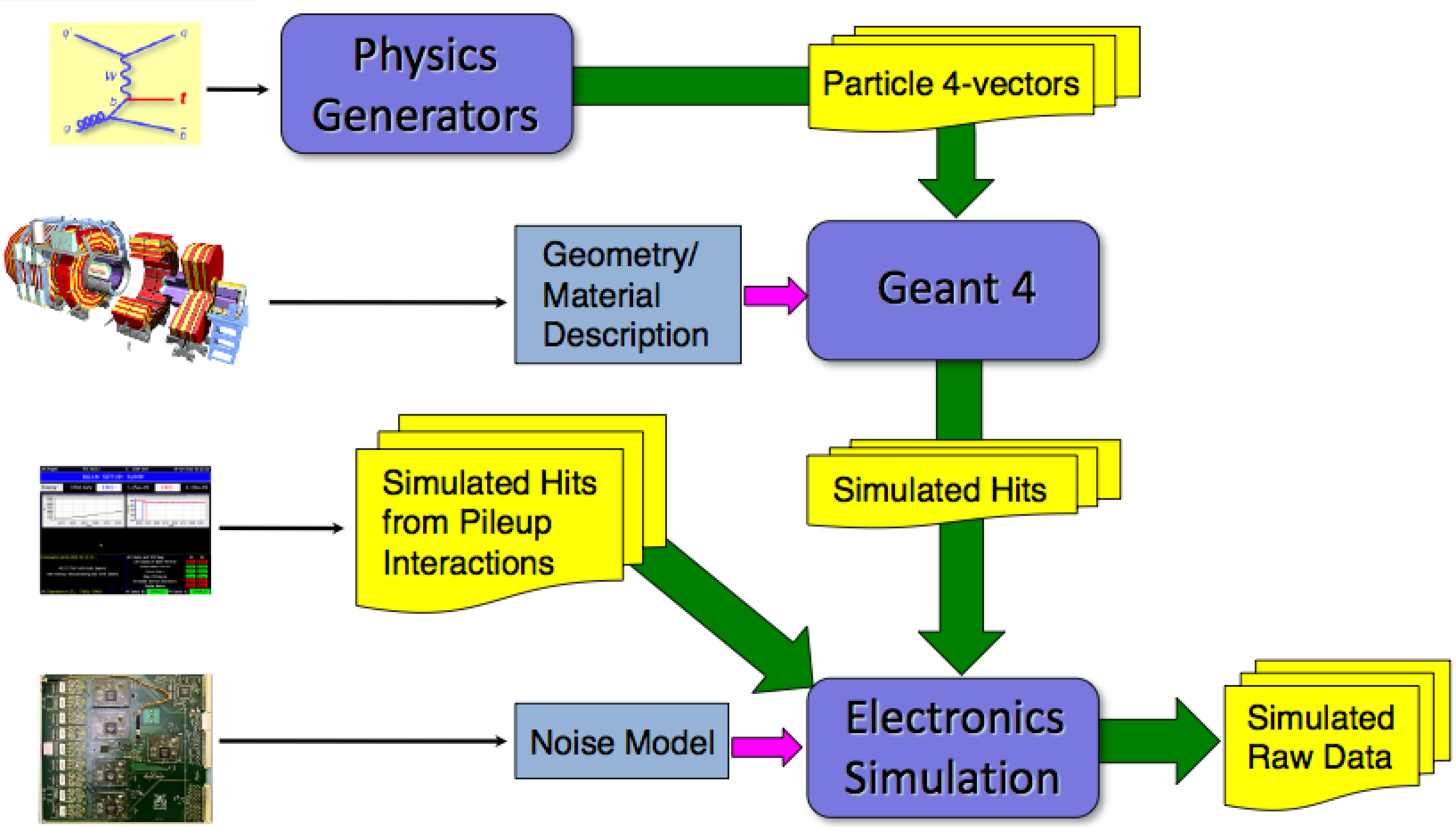
- mixing followed by the simulation of the electronic readout
  - dedicated digitization for each subdetector

- Pileup events are mostly soft QCD processes.
  - They can be simulated as minimum-bias events using the Pythia event generator and mixed with hard interaction of physics interest.

- In the classical pileup mixing, the Geant4 SimHits from N-random minimum bias events are mixed into the full simulation chain.



Event mixing



Physics Generators

Particle 4-vectors

Geometry/ Material Description

Geant 4

Simulated Hits from Pileup Interactions

Simulated Hits

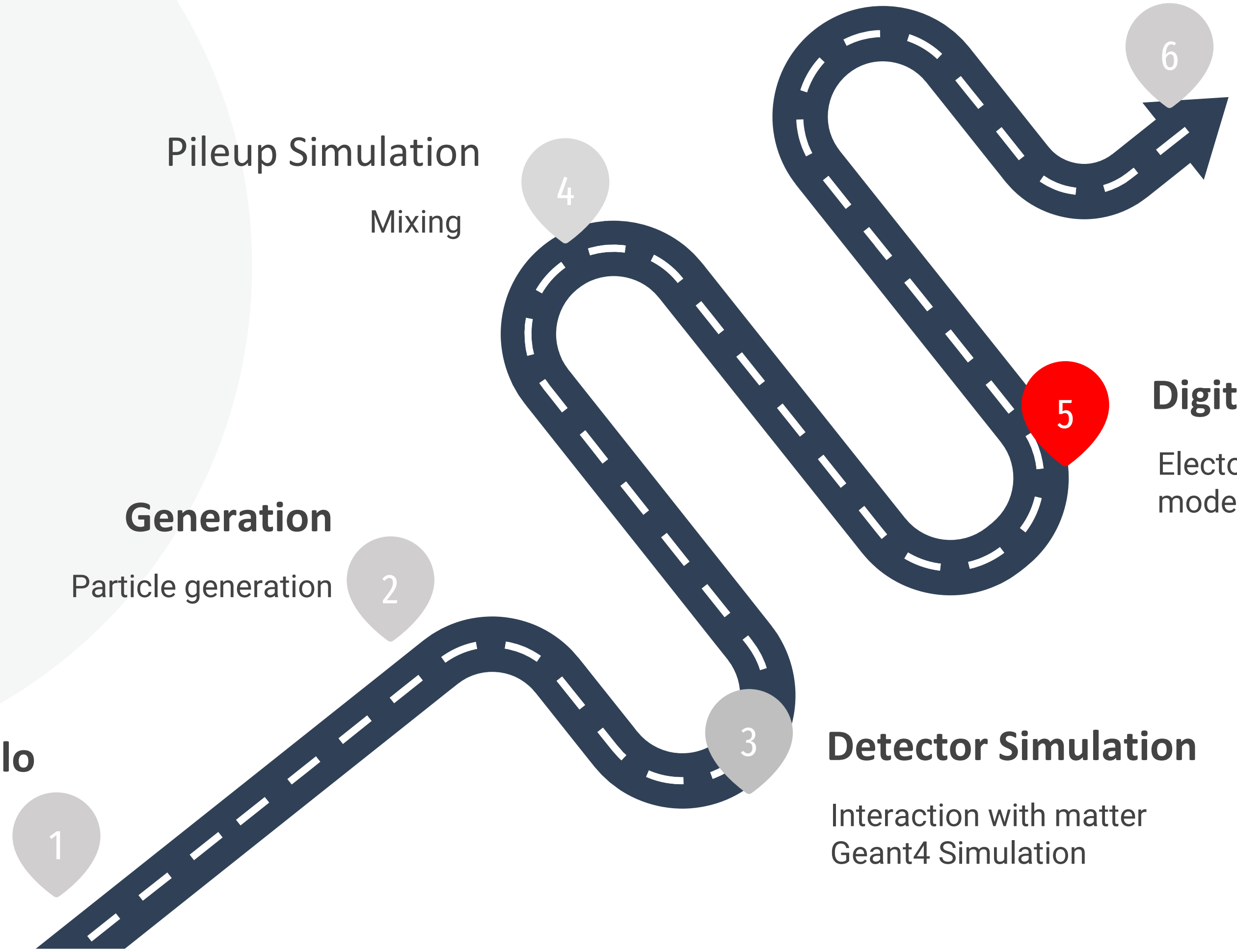
Noise Model

Electronics Simulation

Simulated Raw Data

= “Digitization”

Simulation steps



monte carlo

An introduction

Generation

Particle generation

Detector Simulation

Interaction with matter
Geant4 Simulation

Pileup Simulation

Mixing

Digitization

Electonics response modeling

Validation & Analsis

Validation with real data

Digitization

- Besides the geometry, another experiment-specific aspect of the detector simulation is the “digitization”

- It is not part of the general radiation transportation codes

- It consists of producing the detector response in terms of electric current & voltage signals, as in the real experiment

- The same reconstruction chain can be applied for both real and simulated data

- The general radiation transportation code provides energy deposits in the whole detector; from these, the “digitization” simulates the electrical signals induced in the sensitive parts of the detector

Digitization step


Generation

Detector Simulation

Digitization

MC truth collection include info from particle gun or physics generator about vertices and particles.

Hit collection.

Hit object with timing, position, energy loss info.

Based on Geant4

Digi Collection

Digi objects which include realistic modeling of electronic signal.

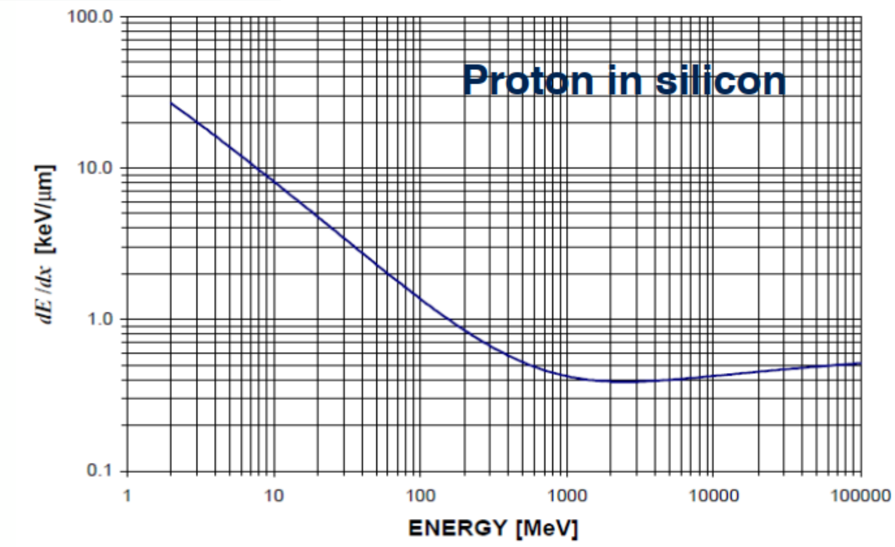
Example: silicon detector

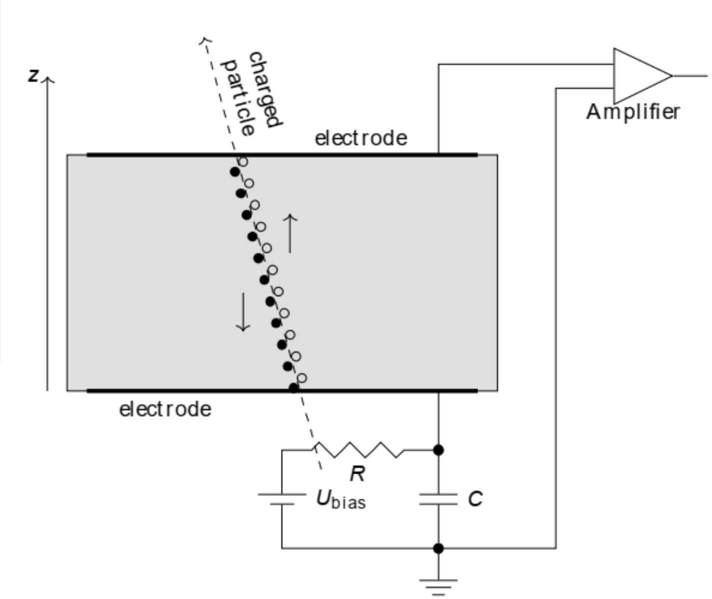
Mean ionization energy **I_0 = 3.62 eV**

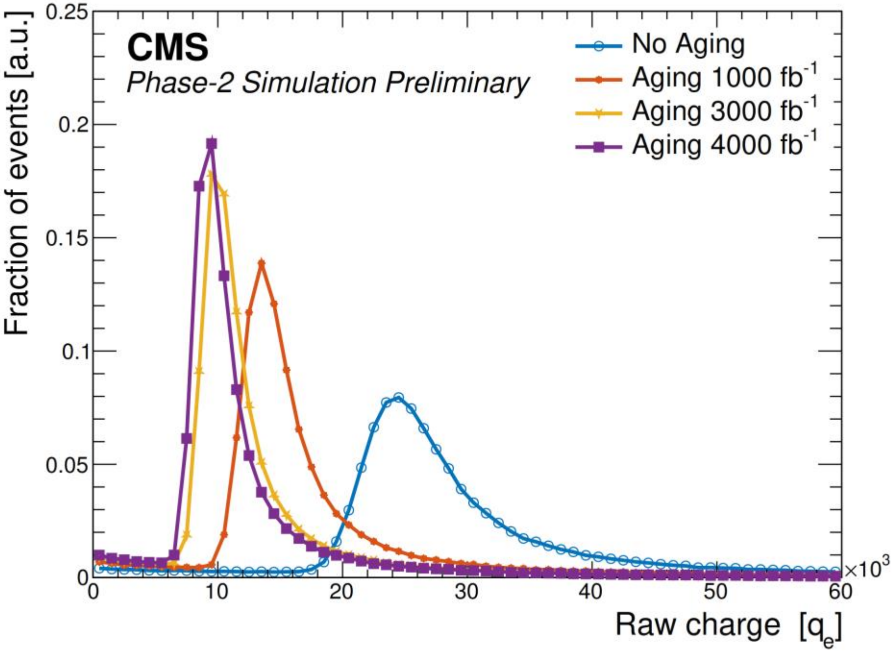
- mean energy loss per flight path of a mip **dE/dx = 3.87 MeV/cm**

Signal of a mip in such a detector:

$$\frac{dE/dx \cdot d}{I_0}$$

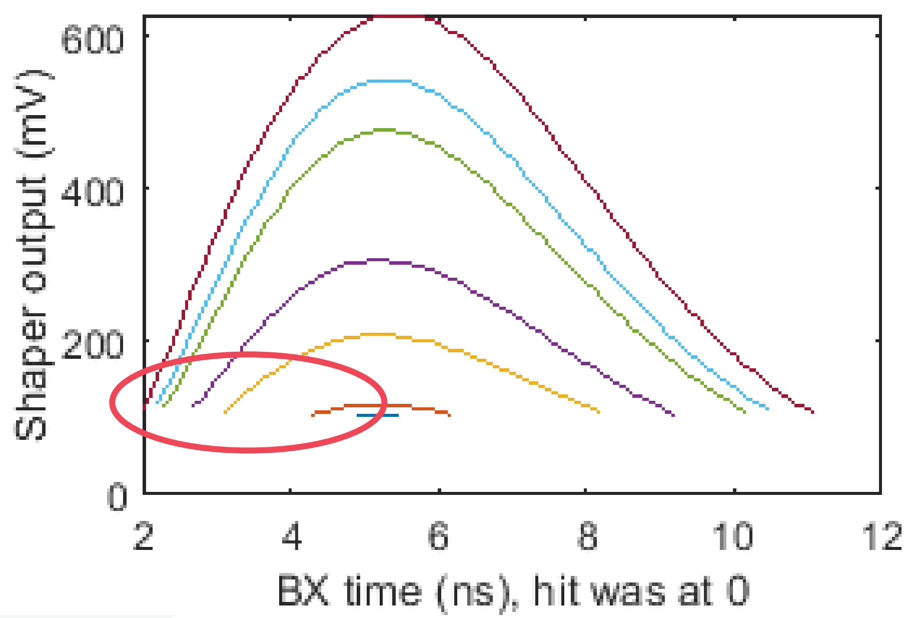


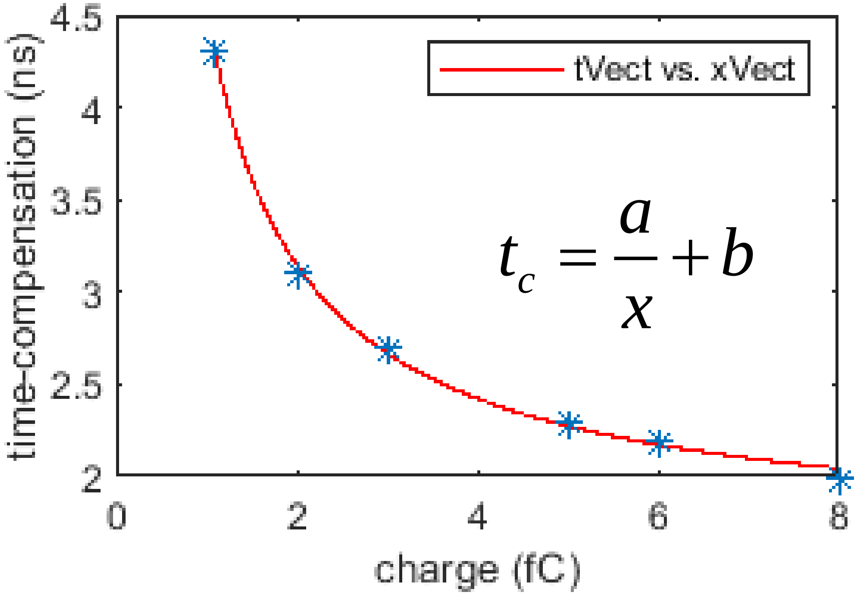




Meddling based on fitting/ behavioral function

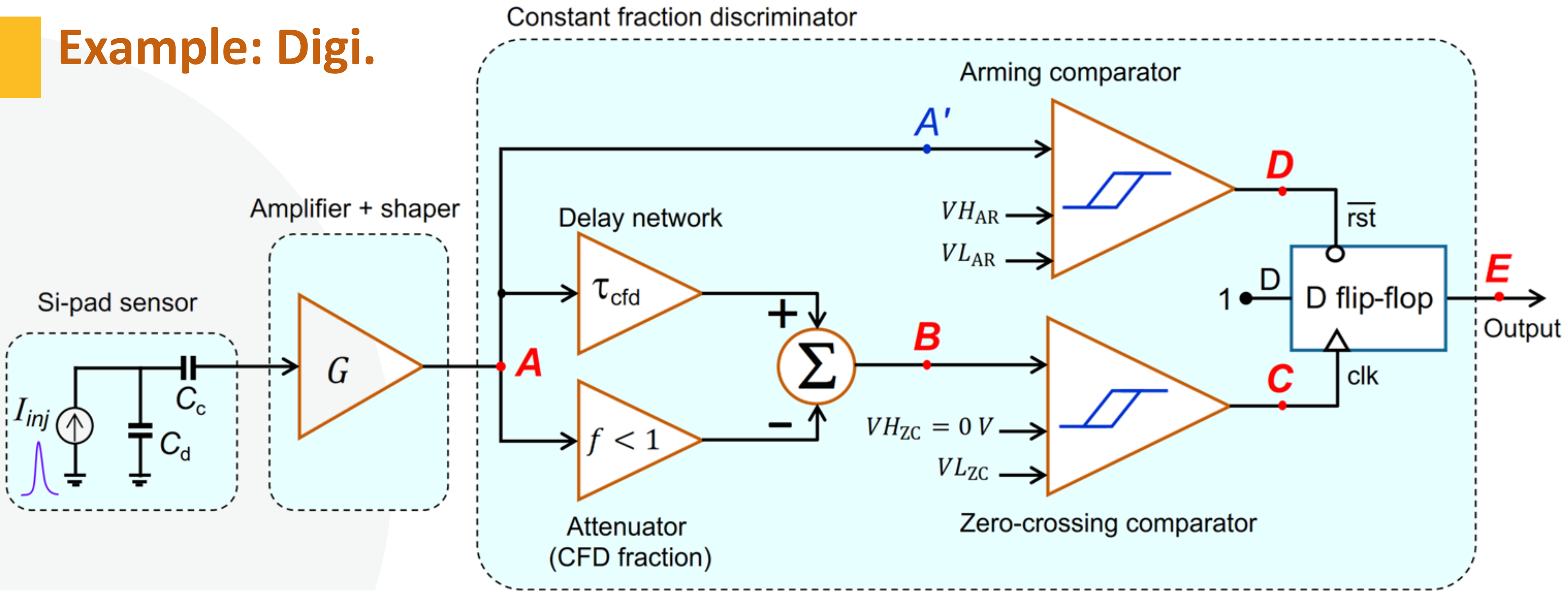
- Example: time-walk compensation

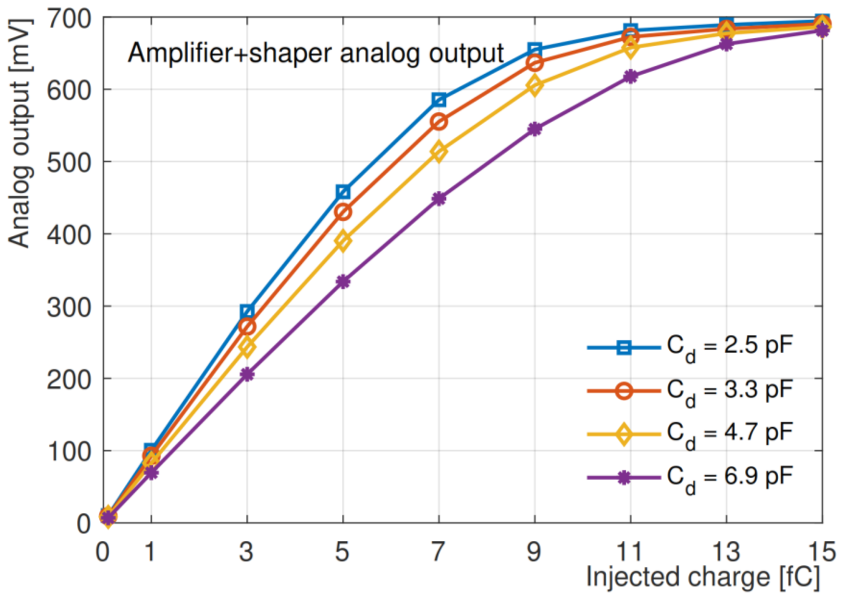


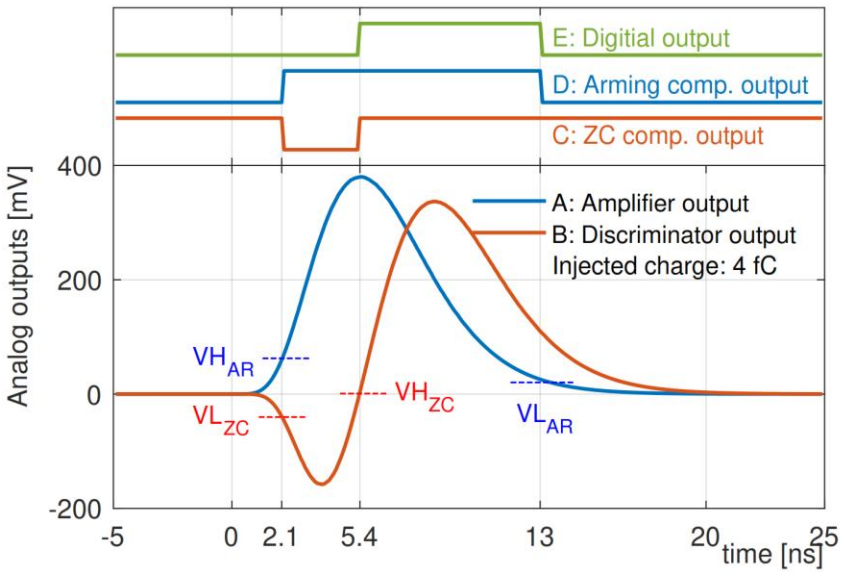


$$t_c = \frac{a}{x} + b$$

Example: Digi.







Geometry

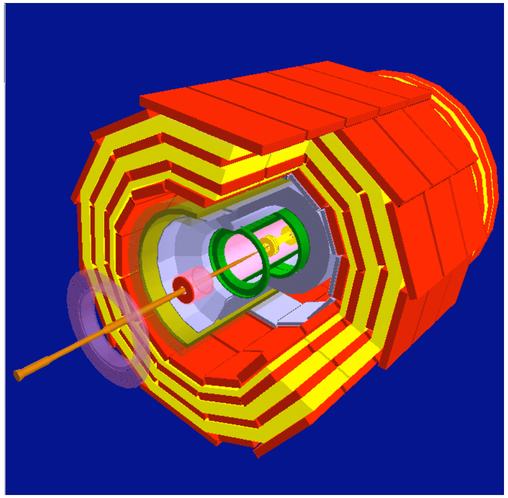
detector description (DDD)

- The CMS detector description system (DDD) provides an application independent way to describe the geometry
  – Simulation, Reconstruction, Event Display etc. use the same basic geometry but with different views.
  Geometry data are stored in a database with a Hierarchical Versioning
  System Alignment corrections are applied with reference to a given baseline geometry

- Provides Stores for Materials, Solids, Logical Parts, Specifics, Rotation matrix
  Modular sub--detector description in XML and C++

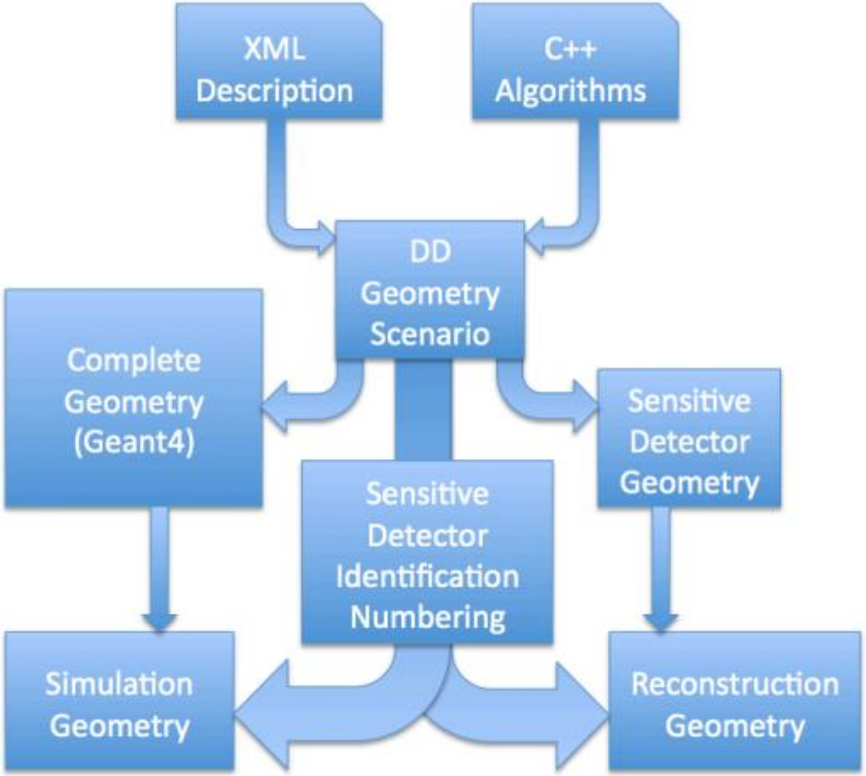
Converts DD solids and materials to Geant4 counterparts

XML based, but independent on the language chosen for detector implementation



Multiple geometries describing one detector

- DD to G4 geometry for simulation
  - Simulation geometry is constructed from DD, sensitive volumes assigned unique IDs
- DD to for Reco geometry
  - Only sensitive volumes with their unique IDs
- DD to ROOT (TGeo) for visualization
  - Two geometries constructed
  - simulation reconstruction



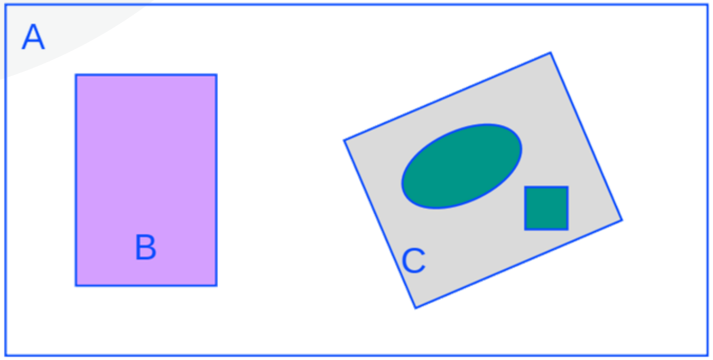
Geometry

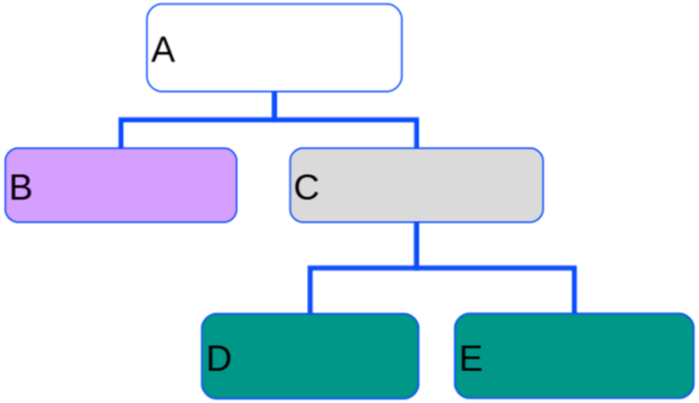
- The way to describe the geometry varies widely between the different simulation engines
  - In Geant4, you need to write some C++ code
    - Geometry objects are instances of classes
    - Geometry parameters (e.g. dimensions) are arguments of the constructors

- The geometry can be “flat” or “hierarchical”

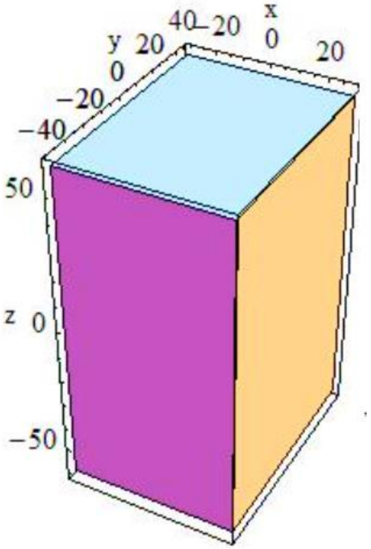
In Geant4, it is hierarchical: a volume is placed in its mother volume; there are mother-daughter relationships

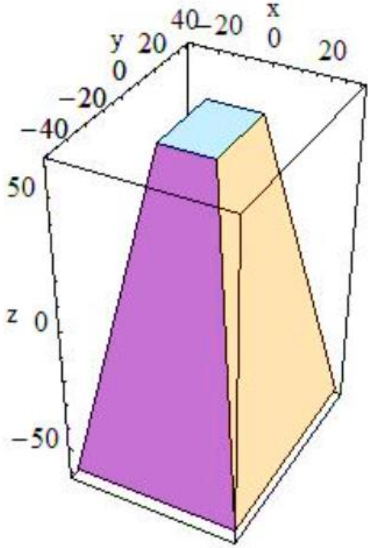
- A **material** should be assigned to each volume

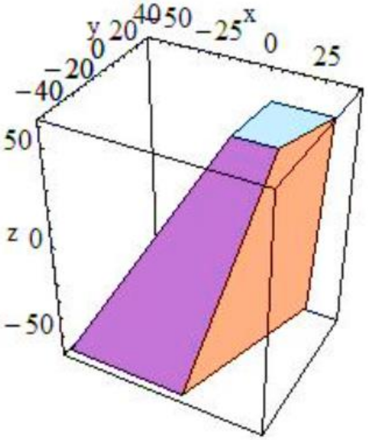


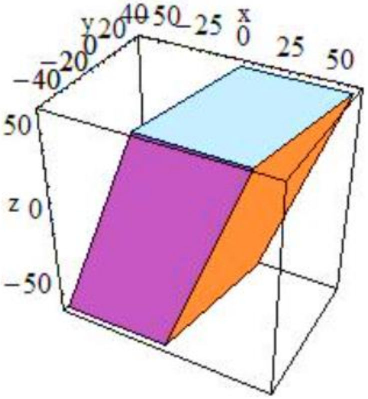


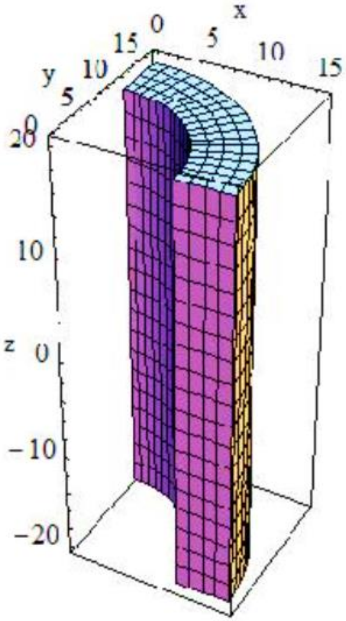
Constructed Geometry Solids

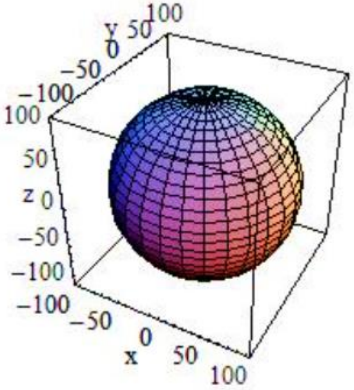


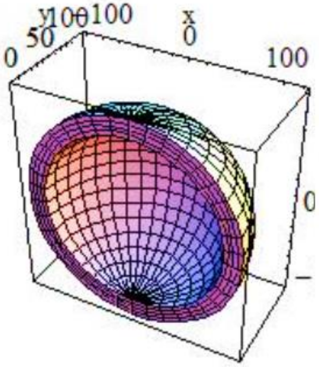


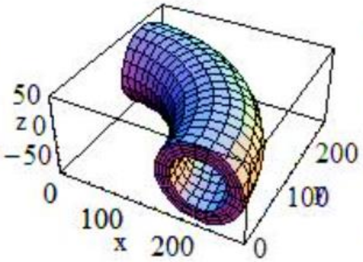


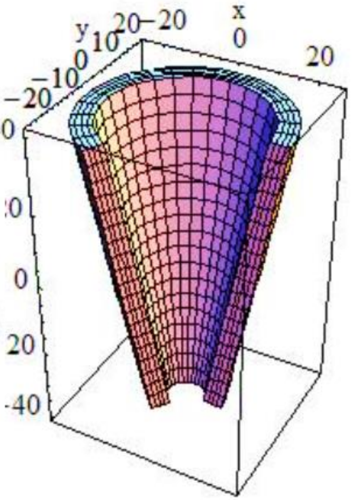




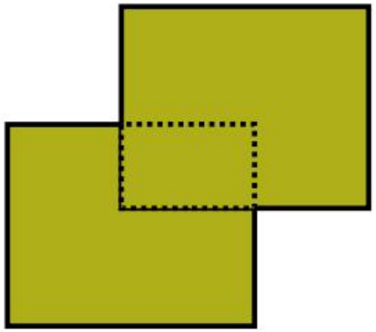




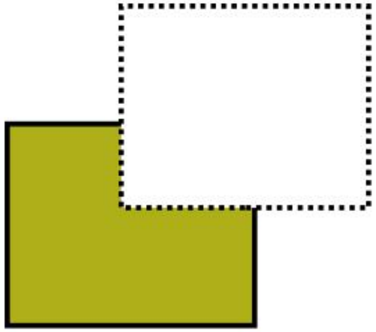




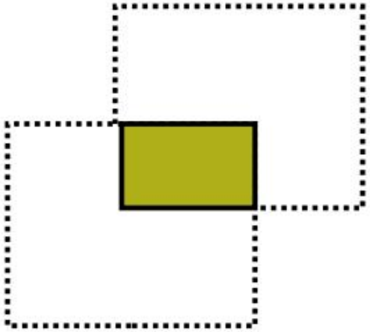
Boolean solids



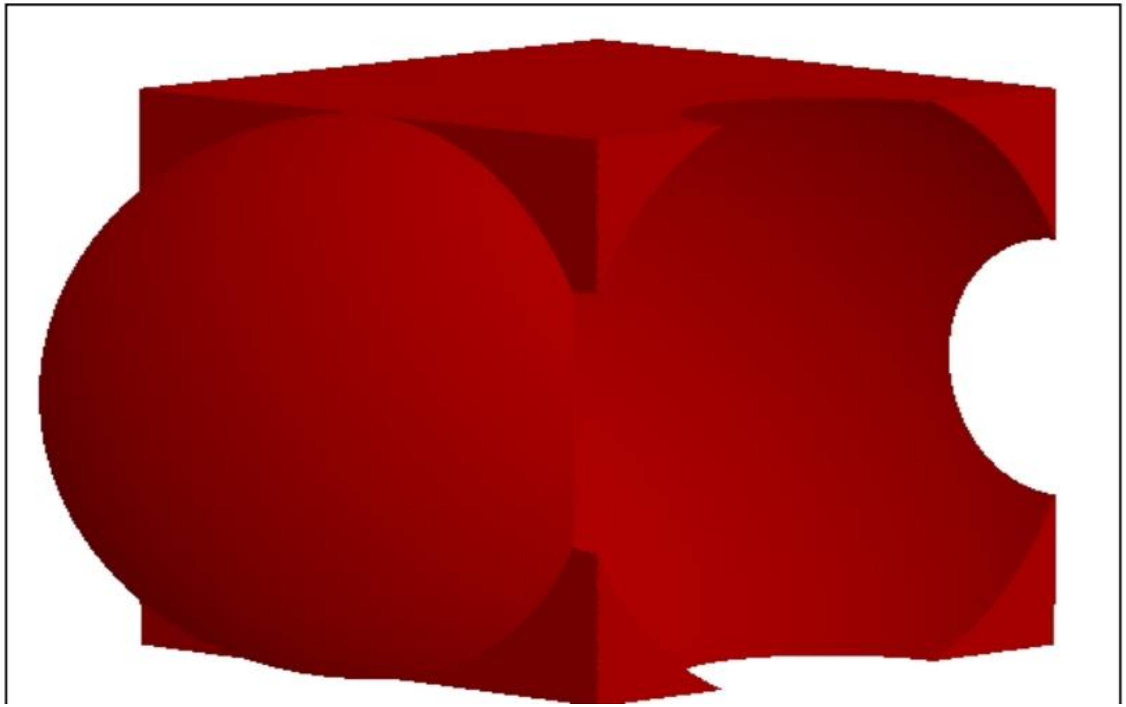
Union



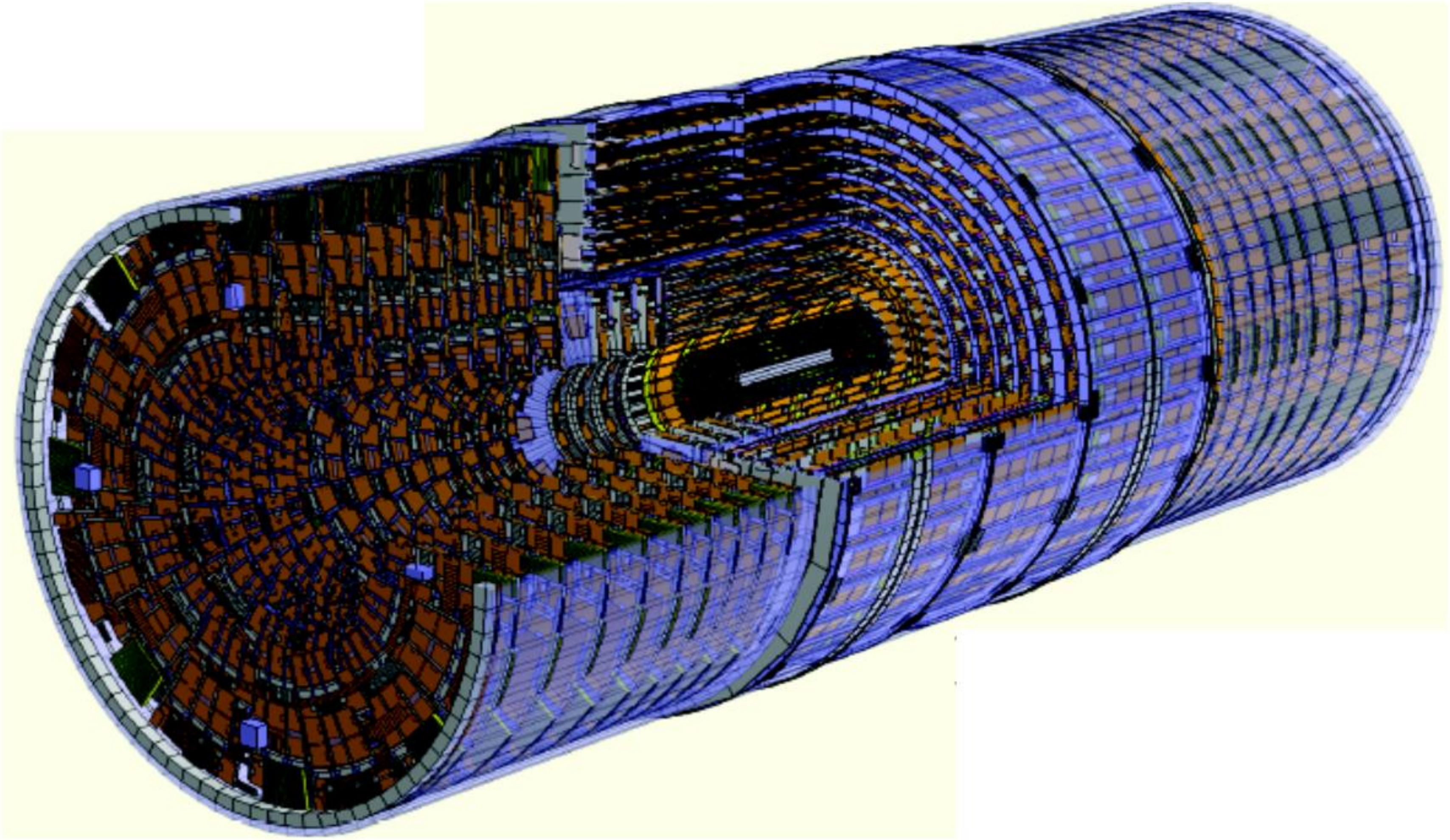
Subtraction



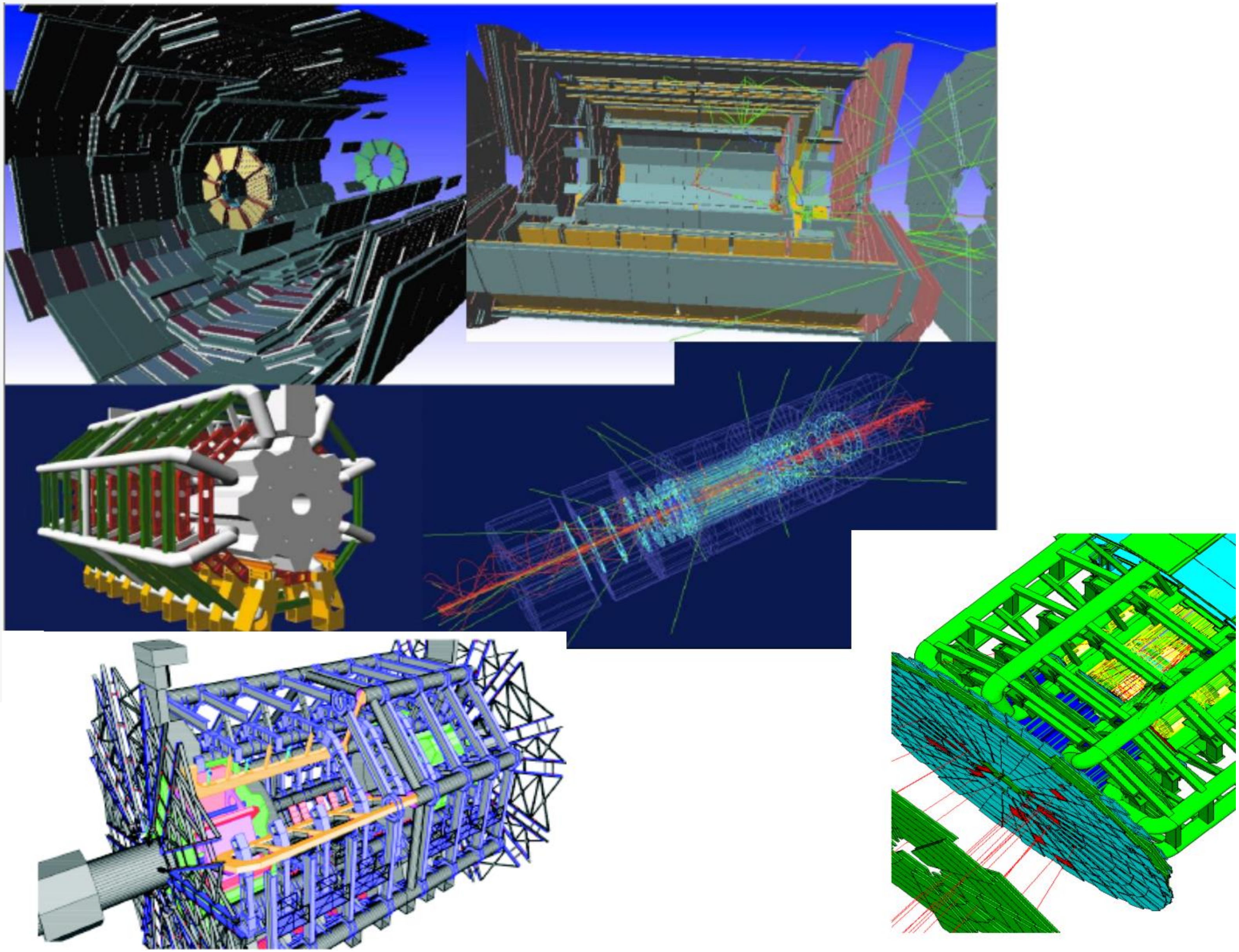
Intersection



CMS tracker Geometry

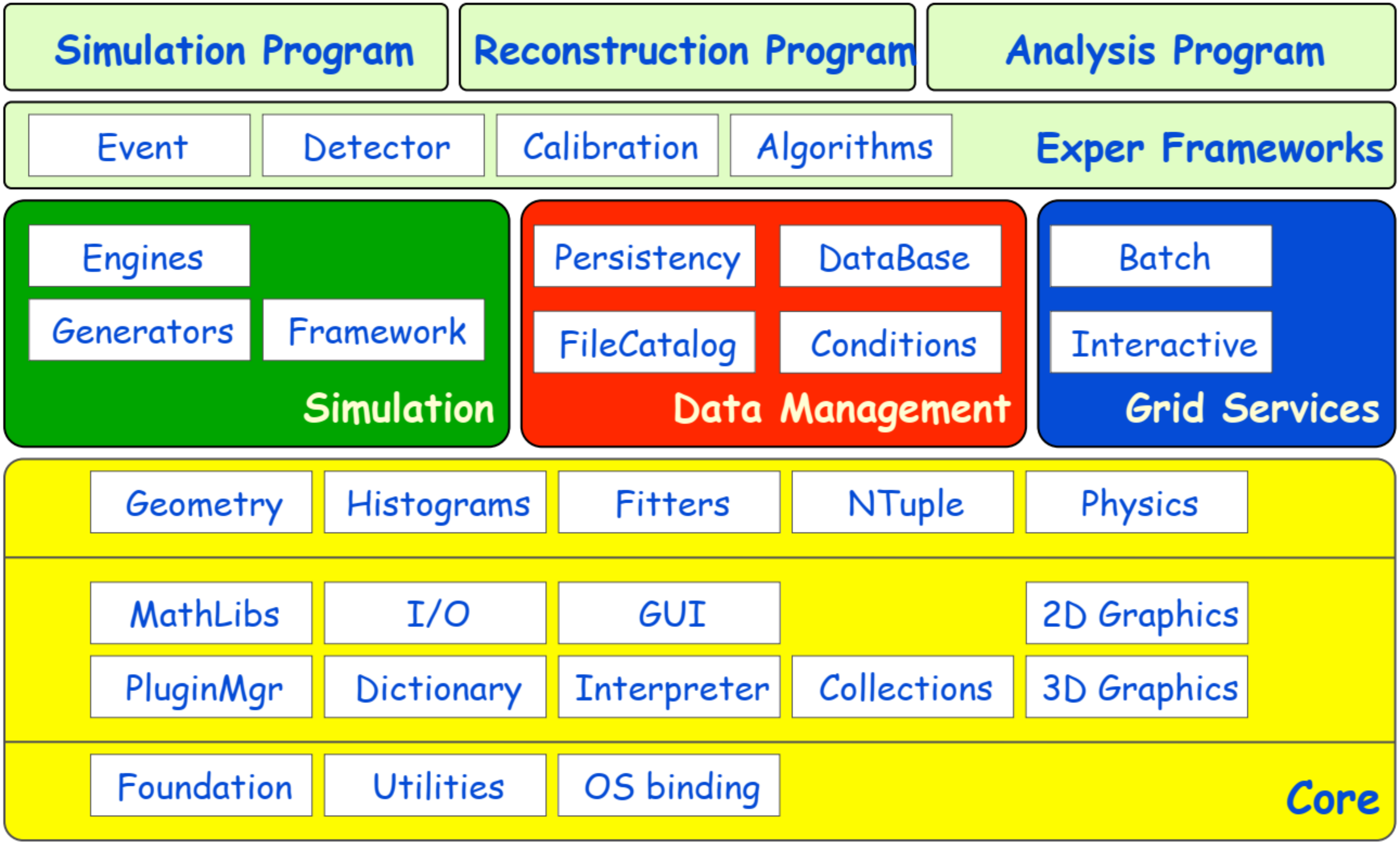


ATLAS Geometry

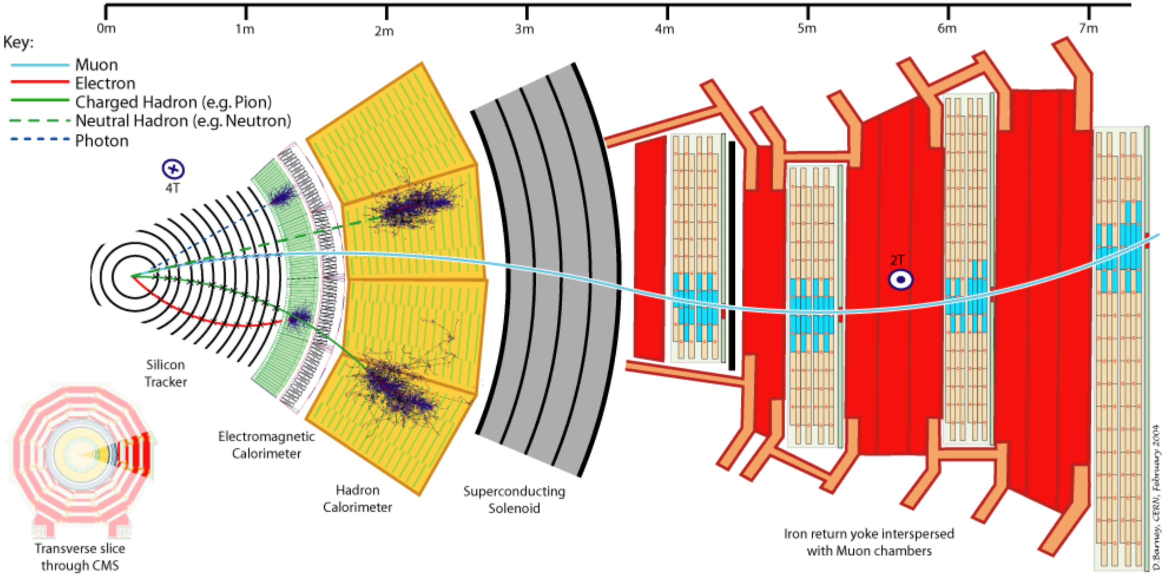


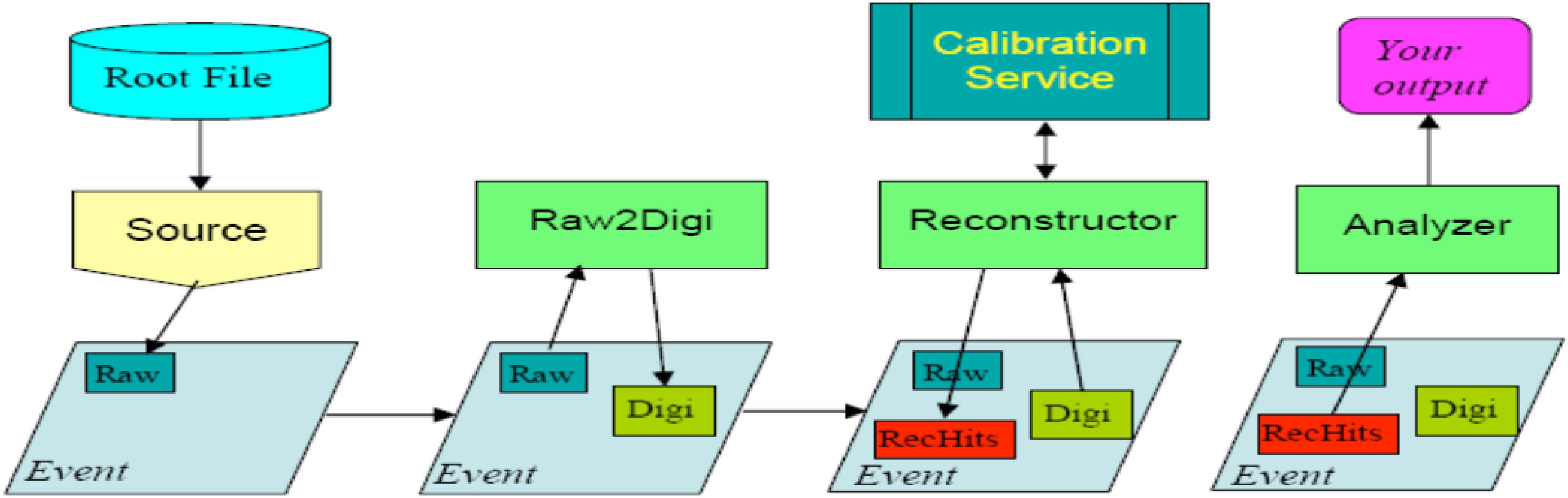
CMSSW
CMS software framework

Software and Analysis in CMS



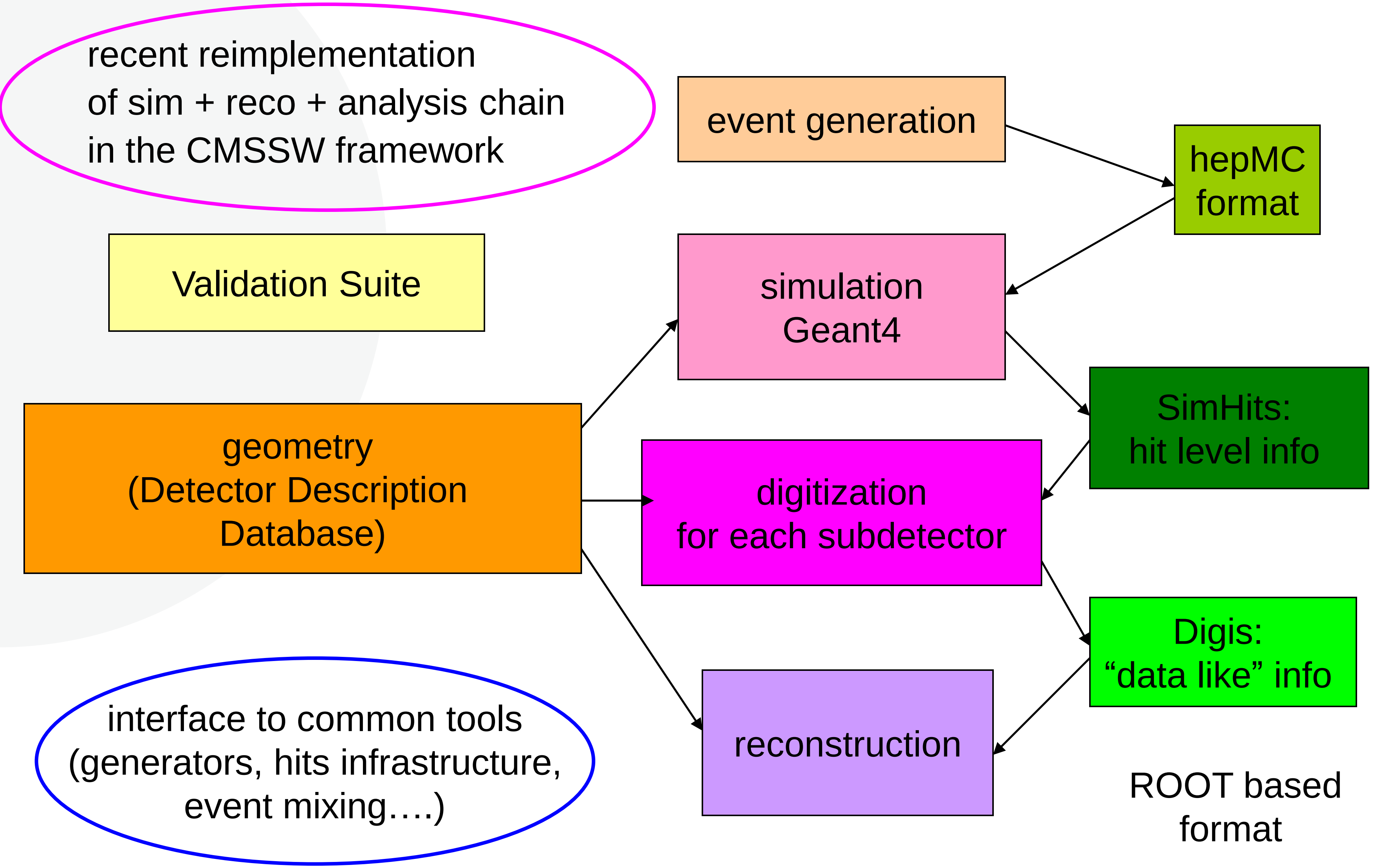
CMSSW





- Modules are configurable and communicate via the Event https://twiki.cern.ch/twiki/bin/view/CMS/WorkBookCMSSWFramework

CMS scheme for simulation



recent reimplementation
of sim + reco + analysis chain
in the CMSSW framework

Validation Suite

geometry
(Detector Description
Database)

interface to common tools
(generators, hits infrastructure,
event mixing….)

event generation

hepMC
format

simulation
Geant4

SimHits:
hit level info

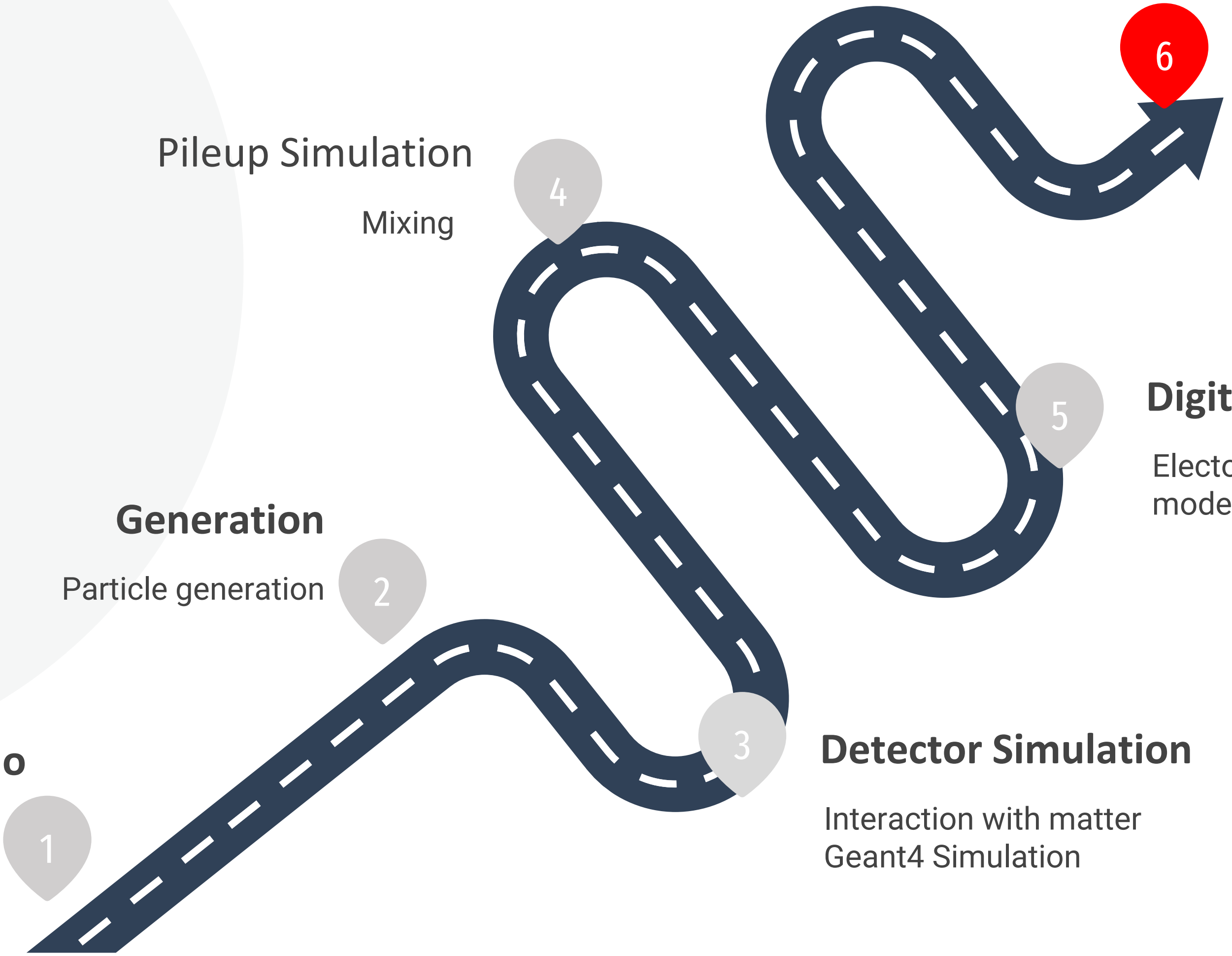
digitization
for each subdetector

Digis:
“data like” info

reconstruction

ROOT based
format

Simulation steps



monte carlo

An introduction

Generation

Particle generation

Detector Simulation

Interaction with matter
Geant4 Simulation

Pileup Simulation

Mixing

Digitization

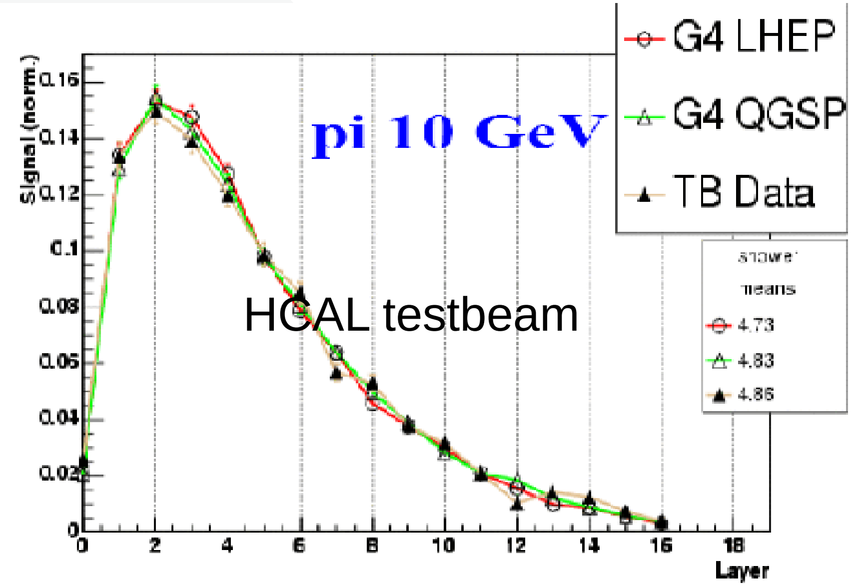
Electonics response modeling

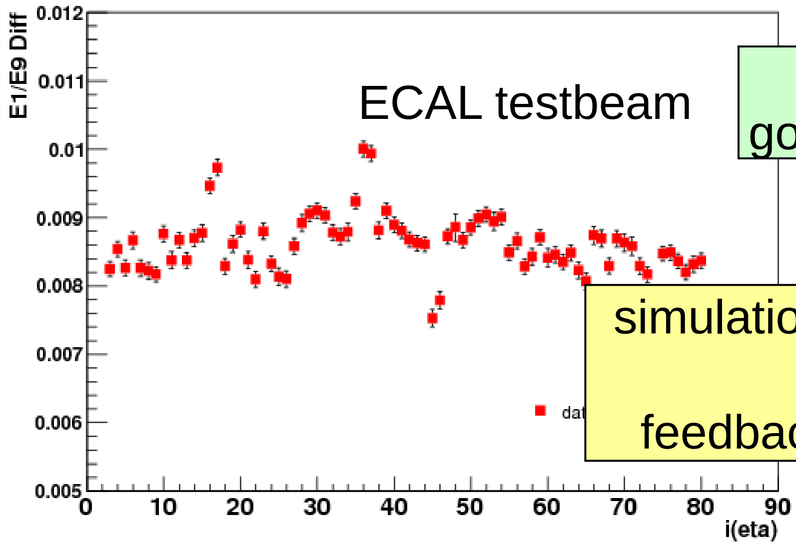
Validation & Analsis

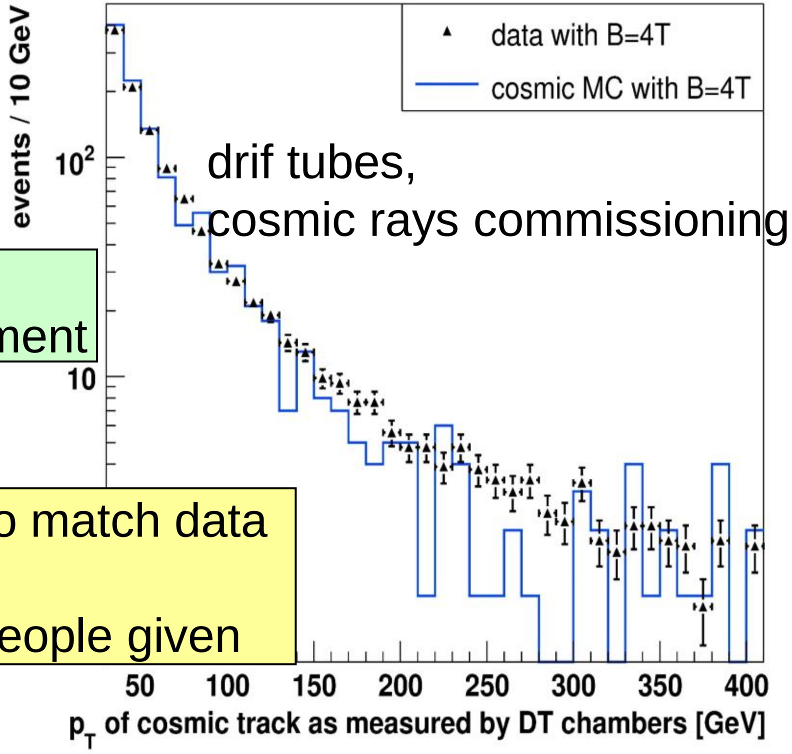
Validation with real data

Validation vs real data

validation versus testbeam, integration facilities, comics commissioning ongoing for all the subdetectors







general good agreement

simulation tuning to match data
AND
feedback to G4 people given

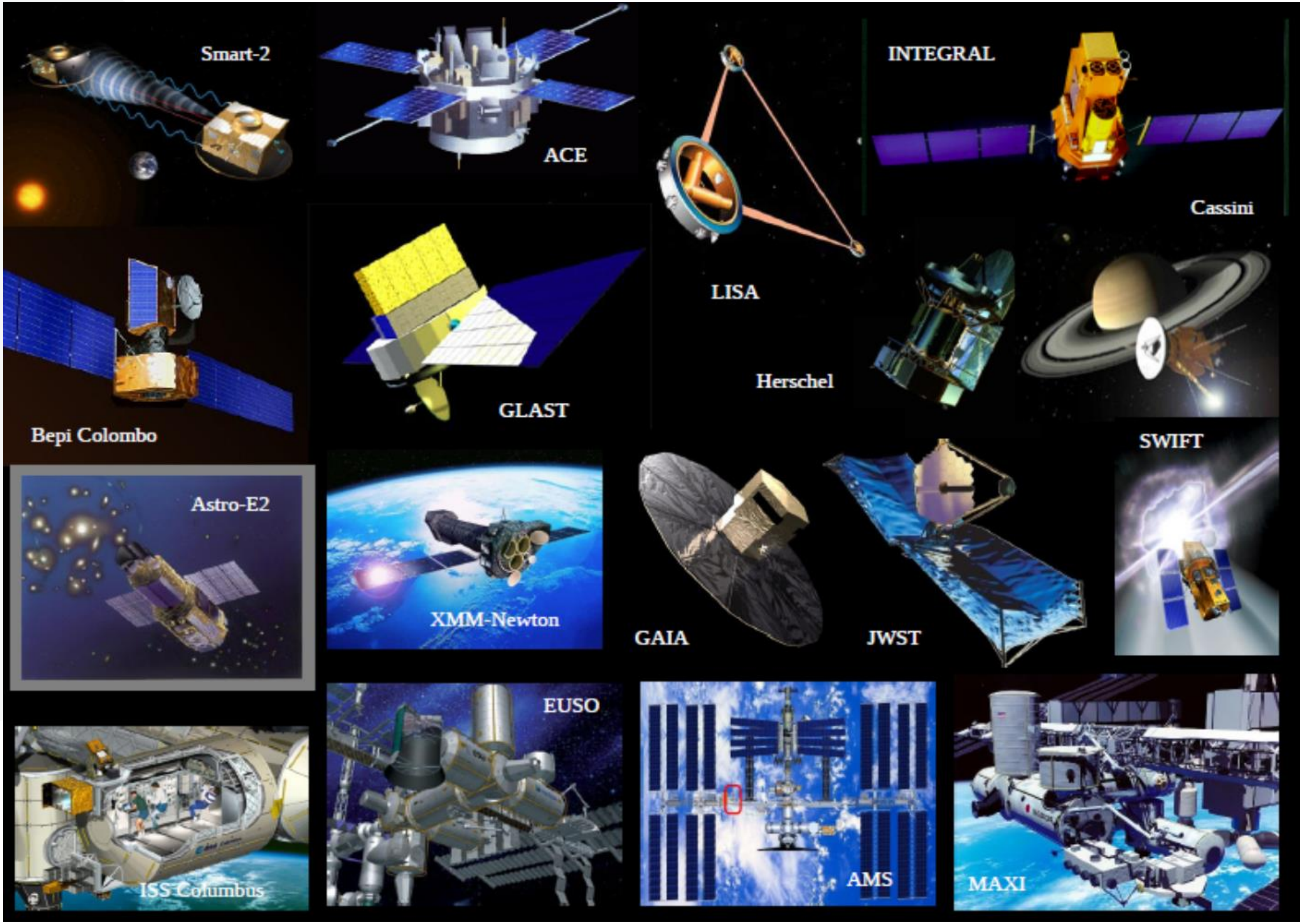
Application domains

- We are considering here mainly high-energy physics, but...

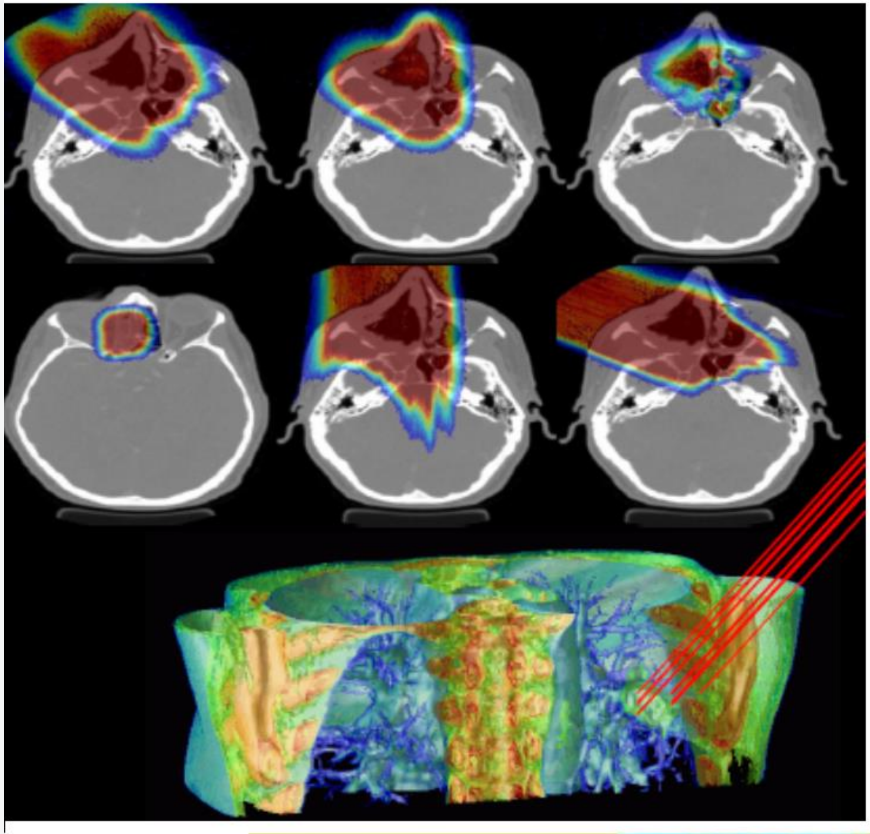
- There are other domains where the same radiation transportation codes are successfully used:
  - Nuclear physics
  - Accelerator science
  - Astrophysics
  - Space engineering
  - Radiation damage
  - Medical physics
  - Industrial applications

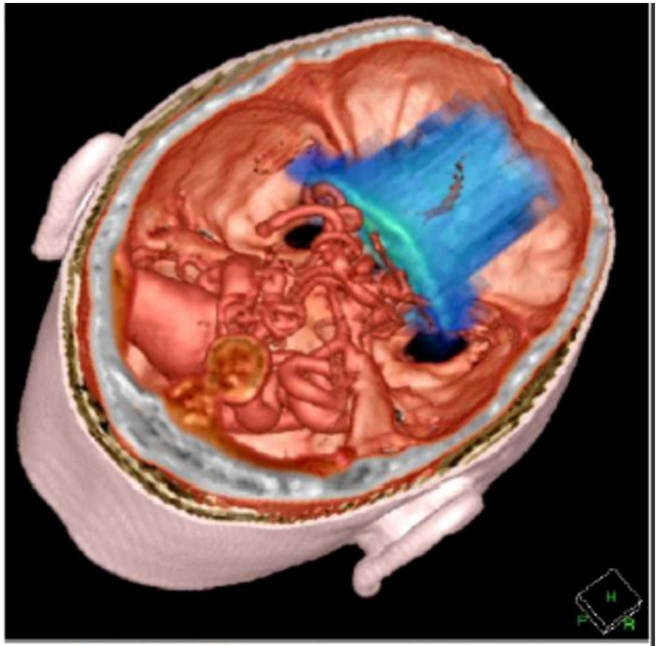
- So, detector simulation is a multi-disciplinary field!

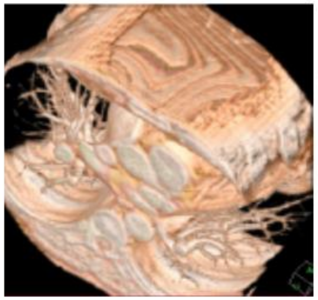
Space applications

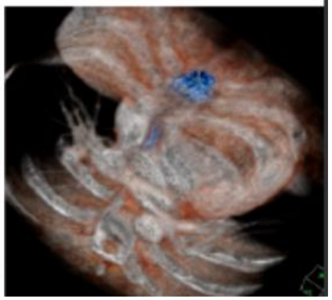


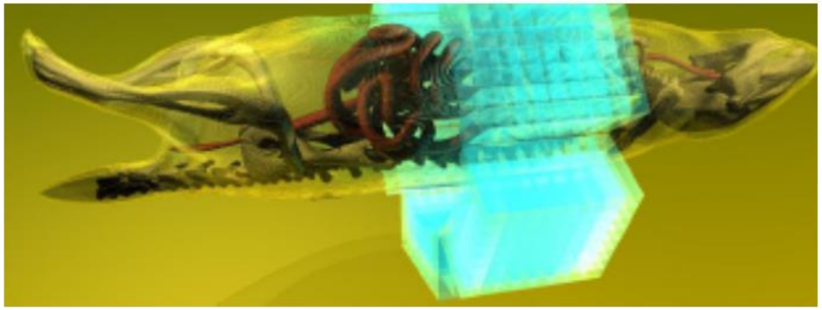
Medical applications











Thank you

Lets give the credit where the credit is

- For preparation of this presentation, I owe to many contributors who I borrowed some slides or sentences from them. I deeply appreciate them (not in order!)
  - C. Rovelli, “The detailed simulation of the CMS detector”
  - https://cp3.irmp.ucl.ac.be/projects/madgraph
  - M. Asai , “Basics of Monte Carlo Simulation”
  - M. Whalley, “LesHouchesAccordPDF Status Report and Future Plans”
  - L. Silvestris, “Software & Analysis in CMS”
  - M. Novak, “Detector Simulation”
  - N. Srimanobhas, “Introduction to Monte Carlo for Particle Physics Study”
  - G. Boudoul, I. Osborne , “CMS Detector Descrip.on for Run II and Beyond”
  - T. Sjostrand et al. “An Introduction to PYTHIA 8.2”, CERN-PH-TH-2014-190