Status of CEPC Simulation Framework

Tao Lin

<lintao@ihep.ac.cn>

28 October 2022

The 2022 International Workshop on the High Energy Circular Electron Positron Collider

Outline

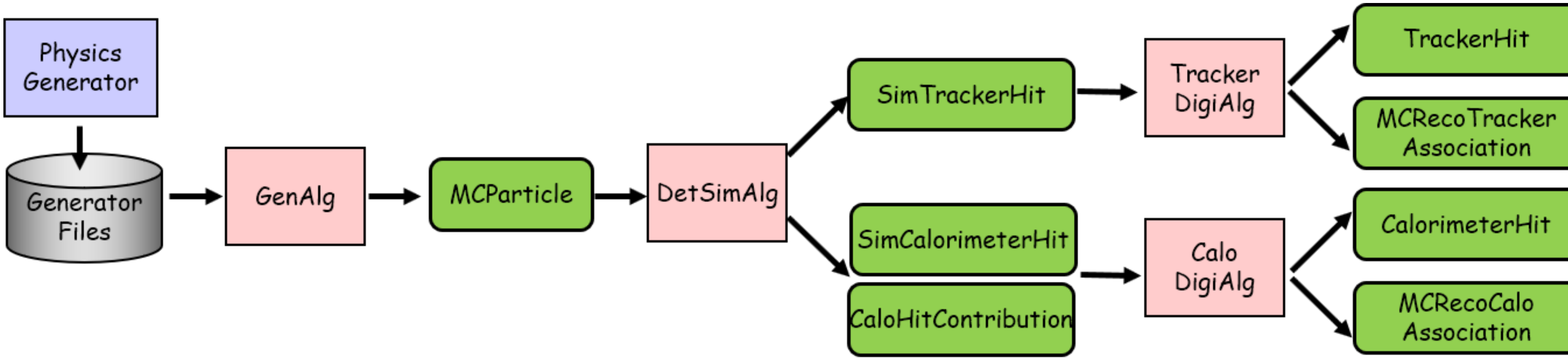
- Overview and recent developments of simulation framework
- Explore Gaussino: a simulation framework from LHCb
- Summary

Overview of Simulation Framework

A simulation framework is being developed for CEPC.

- A complete simulation chain for physics and detector performance studies.
  - Event generation with different physics generator
  - Detector simulation with Geant4 and fast simulation
- Well integration with Key4hep software stack.
  - EDM4hep as Event Data Model
  - DD4hep as Detector Description (including magnetic fields)
  - Gaudi as underlying framework
- Lightweight and modular design.
  - Gaudi Algorithms to take charge of the complete simulation workflow
  - A customized G4 Run Manager to integrate Gaudi and Geant4
  - Detector response and the MC truth information handling as plugins
  - Support of full and fast simulation transparently
  - Support of background mixing

Simulation data flow with EDM4hep

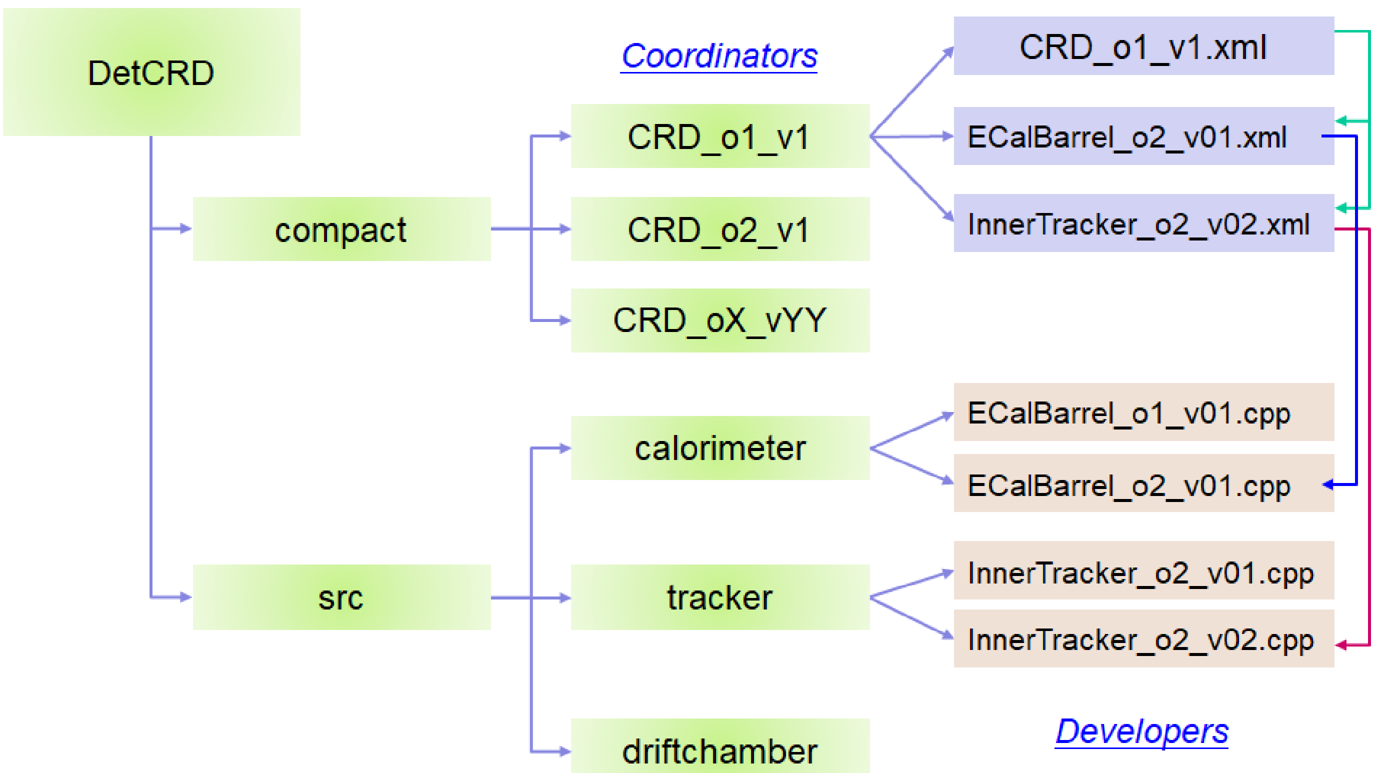


Simulation is flexible: run all steps in one job or run them separately

- Physics generator
  - MCParticle
- Detector Simulation
  - MCParticle (with secondaries), SimTrackerHit, SimCalorimeterHits
- Digitization
  - TrackerHit, CalorimeterHit

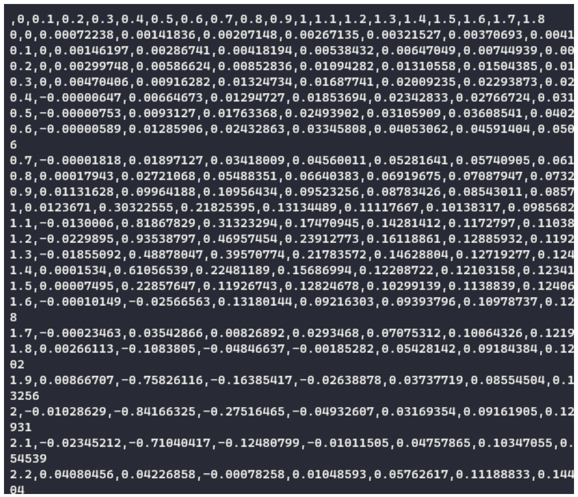
Detector Description with DD4hep

- Both geometry and magnetic field are described by DD4hep. Then, DDG4 is used to convert them to Geant4 automatically.
- All the detector options are managed in CEPCSW git repository.

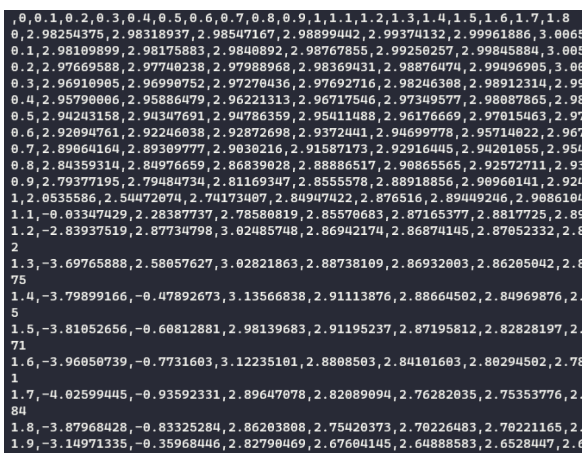


Non-uniform magnetic fields

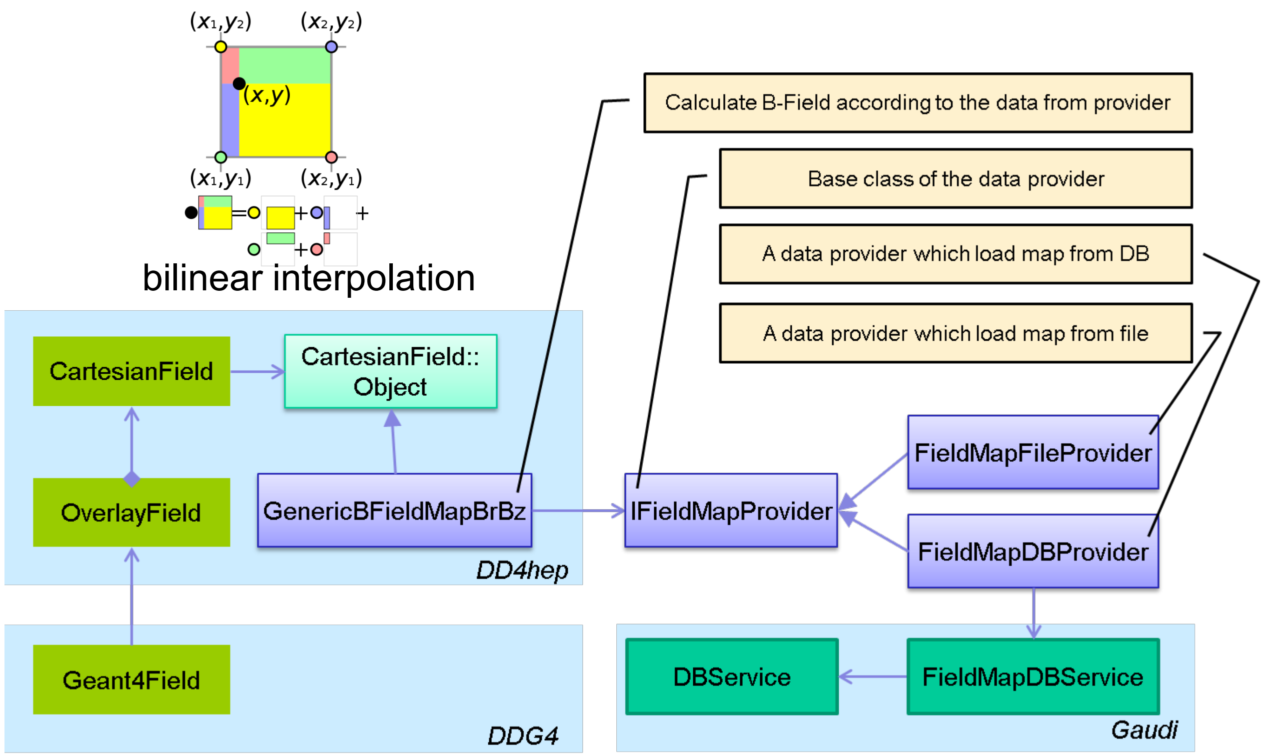
- A DD4hep extension is developed to support non-uniform magnetic fields in CEPCSW. The Br/Bz csv files are provided by magnetic group.



```
,0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8
0,0,0.00072238,0.00141836,0.00207148,0.00267135,0.00321527,0.00370693,0.0041
0.1,0,0.00146197,0.00286741,0.00418194,0.00538432,0.00647049,0.00744939,0.00
0.2,0,0.00299748,0.00586624,0.00852836,0.01094282,0.01310558,0.01504385,0.01
0.3,0,0.00470406,0.00916282,0.01324734,0.01687741,0.02009235,0.02293873,0.02
0.4,-0.00000647,0.00664673,0.01294727,0.01853694,0.02342833,0.02766724,0.031
0.5,-0.00000753,0.0093127,0.01763368,0.02493902,0.03105909,0.03608541,0.0402
0.6,-0.00000589,0.01285906,0.02432863,0.03345808,0.04053062,0.04591404,0.050
6
0.7,-0.00001818,0.01897127,0.03418009,0.04560011,0.05281641,0.05740905,0.061
0.8,0.00017943,0.02721068,0.05488351,0.06640383,0.06919675,0.07087947,0.0732
0.9,0.01131628,0.09964188,0.10956434,0.09523256,0.08783426,0.08543011,0.0857
1,0.0123671,0.30322555,0.21825395,0.13134489,0.11117667,0.10138317,0.0985682
1.1,-0.0130006,0.81867829,0.31323294,0.17470945,0.14281412,0.1172797,0.11038
1.2,-0.0229895,0.93538797,0.46957454,0.23912773,0.16118861,0.12885932,0.1192
1.3,-0.01855092,0.48878047,0.39570774,0.21783572,0.14628804,0.12719277,0.124
1.4,0.0001534,0.61056539,0.22481189,0.15686994,0.12208722,0.12103158,0.12341
1.5,0.00007495,0.22857647,0.11926743,0.12824678,0.10299139,0.1138839,0.12406
1.6,-0.00010149,-0.02566563,0.13180144,0.09216303,0.09393796,0.10978737,0.12
8
1.7,-0.00023463,0.03542866,0.00826892,0.0293468,0.07075312,0.10064326,0.1219
1.8,0.00266113,-0.1083805,-0.04846637,-0.00185282,0.05428142,0.09184384,0.12
02
1.9,0.00866707,-0.75826116,-0.16385417,-0.02638878,0.03737719,0.08554504,0.1
3256
2,-0.01028629,-0.84166325,-0.27516465,-0.04932607,0.03169354,0.09161905,0.12
931
2.1,-0.02345212,-0.71040417,-0.12480799,-0.01011505,0.04757865,0.10347055,0.
54539
2.2,0.04080456,0.04226858,-0.00078258,0.01048593,0.05762617,0.11188833,0.144
04
```



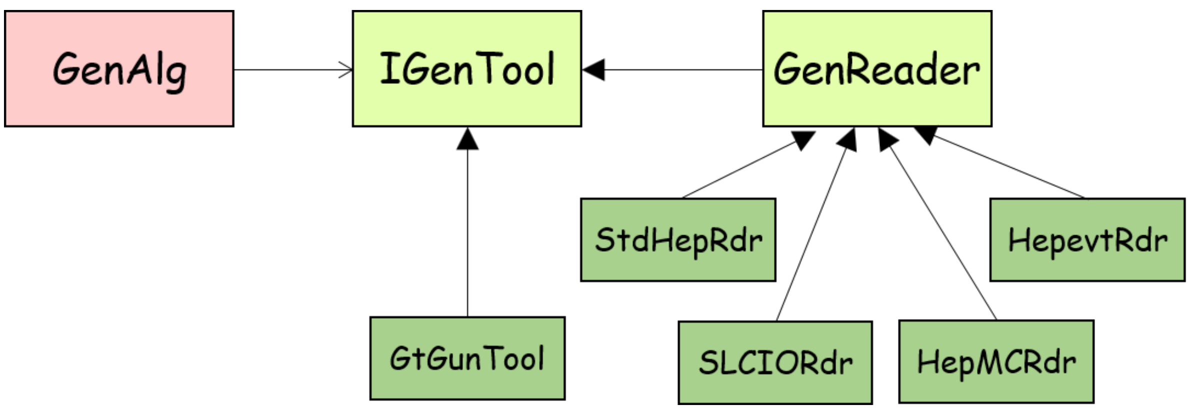
```
,0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1,1.1,1.2,1.3,1.4,1.5,1.6,1.7,1.8
0,2.98254375,2.98318937,2.98547167,2.98899442,2.99374132,2.99961886,3.0065
0.1,2.98109899,2.98175883,2.9840892,2.98767855,2.99250257,2.99845884,3.005
0.2,2.97669588,2.97740238,2.97988968,2.98369431,2.98876474,2.99496905,3.00
0.3,2.96910905,2.96990752,2.97270436,2.97692716,2.98246308,2.98912314,2.99
0.4,2.95790006,2.95886479,2.96221313,2.96717546,2.97349577,2.98087865,2.98
0.5,2.94243158,2.94347691,2.94786359,2.95411488,2.96176669,2.97015463,2.97
0.6,2.92094761,2.92246038,2.92872698,2.9372441,2.94699778,2.95714022,2.967
0.7,2.89064164,2.89309777,2.9030216,2.91587173,2.92916445,2.94201055,2.954
0.8,2.84359314,2.84976659,2.86839028,2.88886517,2.90865565,2.92572711,2.93
0.9,2.79377195,2.79484734,2.81169347,2.8555578,2.88918856,2.90960141,2.924
1,2.0535586,2.54472074,2.74173407,2.84947422,2.876516,2.89449246,2.908610
1.1,-0.03347429,2.28387737,2.78580819,2.85570683,2.87165377,2.8817725,2.89
1.2,-2.83937519,2.87734798,3.02485748,2.86942174,2.86874145,2.87052332,2.8
2
1.3,-3.69765888,2.58057627,3.02821863,2.88738109,2.86932003,2.86205042,2.8
75
1.4,-3.79899166,-0.47892673,3.13566838,2.91113876,2.88664502,2.84969876,2.
5
1.5,-3.81052656,-0.60812881,2.98139683,2.91195237,2.87195812,2.82828197,2.
71
1.6,-3.96050739,-0.7731603,3.12235101,2.8808503,2.84101603,2.80294502,2.78
1
1.7,-4.02599445,-0.93592331,2.89647078,2.82089094,2.76282035,2.75353776,2.
84
1.8,-3.87968428,-0.83325284,2.86203808,2.75420373,2.70226483,2.70221165,2.
1.9,-3.14971335,-0.35968446,2.82790469,2.67604145,2.64888583,2.6528447,2.6
```



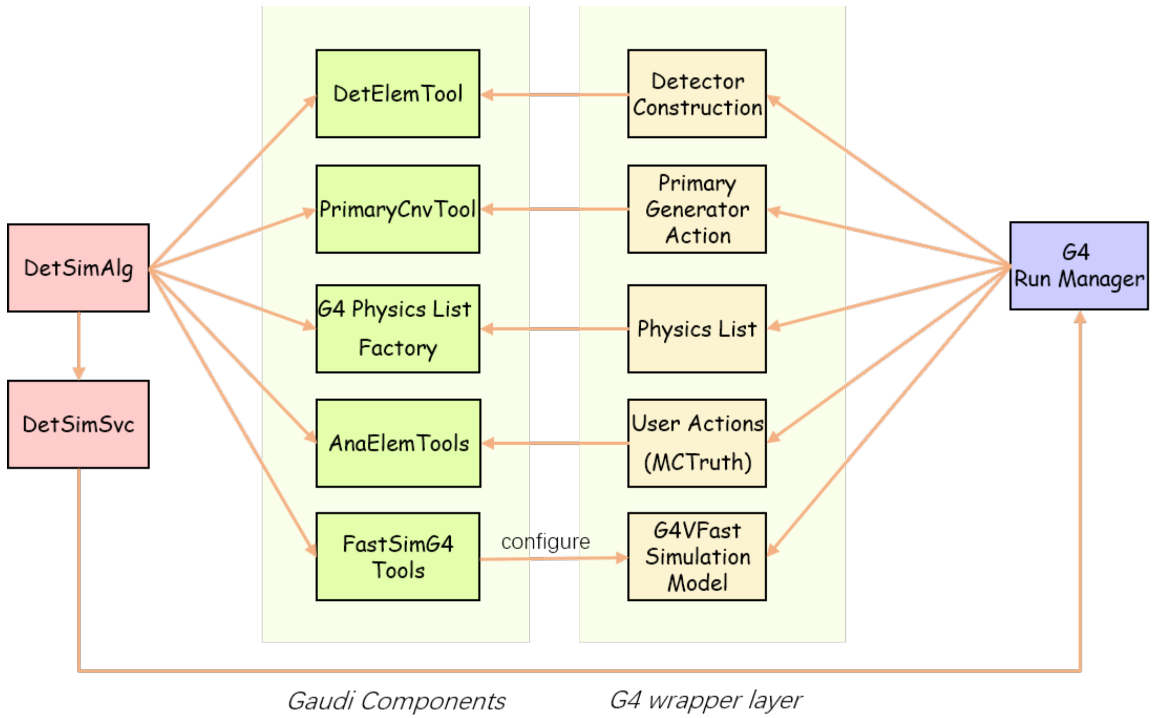
Physics generator interface

Different physics generators are integrated, including StdHep, HepEvt, LCIO, HepMC formats. Particle gun is also supported.

- A physics generator algorithm is in charge of a list of GenTools.
- Easy to extend by adding a new tool.



Integration with Geant4 and Gaudi



Detector simulation is fully integrated with the framework.

- A thin layer is developed to manage corresponding Geant4 objects.
  - All parameters could be configured in Python script.
- Event loop is controlled by an algorithm and a customized run manager.

Detector response and MC Truth

Detector response of all the detectors are available, including silicon detectors, time projection chamber, drift chamber and calorimeters.

- Generation of hits in different detectors are handled by corresponding Geant4 sensitive detectors.
- For each hit: ID is calculated by DD4hep; position/energy is from Geant4.
- At end of each event, store all the hit collections in EDM4hep format.

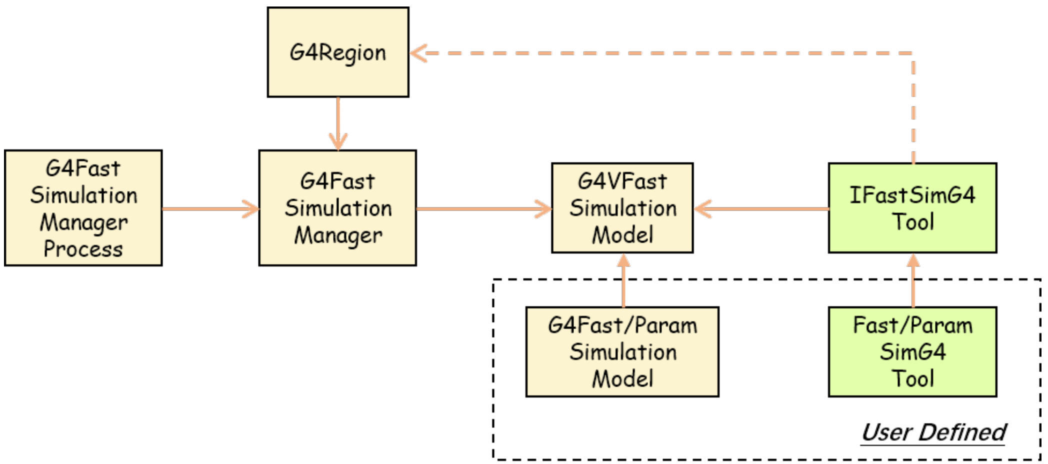
All necessary relationship are stored to rebuild relation between reconstructed particles and MC particles.

- Primary particles are cloned from generation stage
- Decayed secondaries are collected during Geant4 simulation
- Retrieve MC particle from a hit.

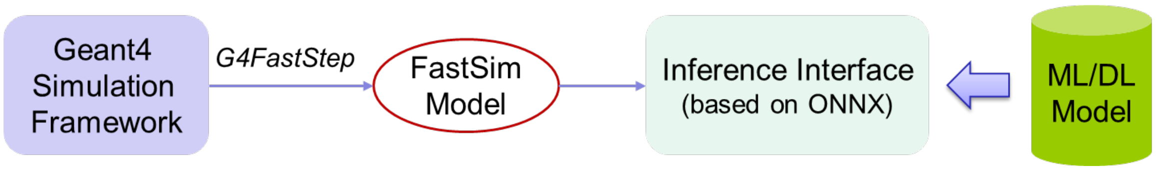
Integration with Fast Simulation

Fast simulation interface is developed to integrate different fast simulation models into Geant4.

- Region based: when a particle enter a region, fast simulation will be triggered by Geant4.



- Support ML methods via ONNX inference interface.

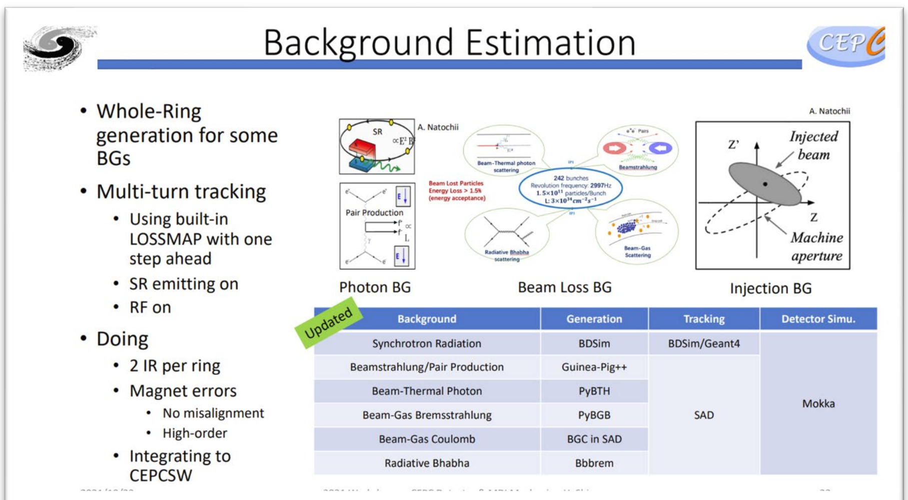


Background Simulation

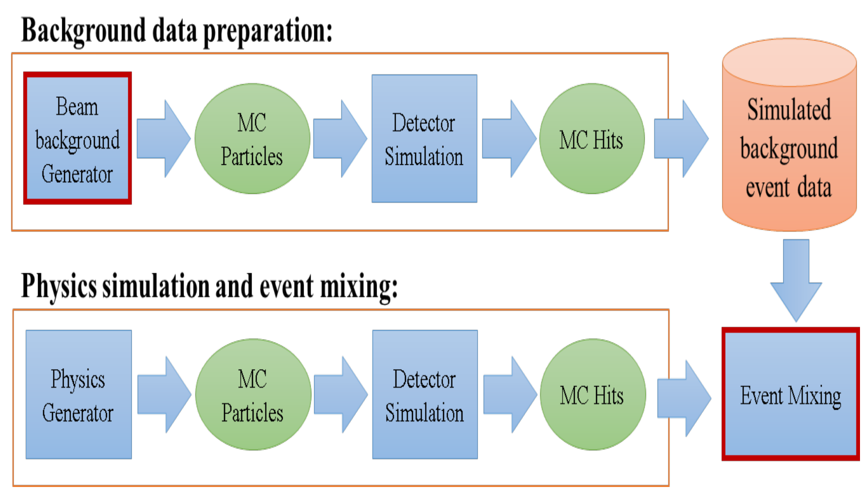
Tao, Haoyu Shi, Wei Xu

Simulation of background is supported in CEPCSW.

- A unified solution for different backgrounds by using the GenTools design.
- Integration of beam gas, pair production (Guinea-Pig) have been done.
- Hit level background mixing: store the hits and used as a library.



Summary of backgrounds. From Haoyu Shi



Hit-level background mixing

Towards multi-threaded simulation

CEPC simulation framework needs further developments.

- The current simulation framework in CEPCSW is single threaded.
- However, both Gaudi and Geant4 already support multi-threading.
- Need work to integrate them and simulate events in parallel.

Examples

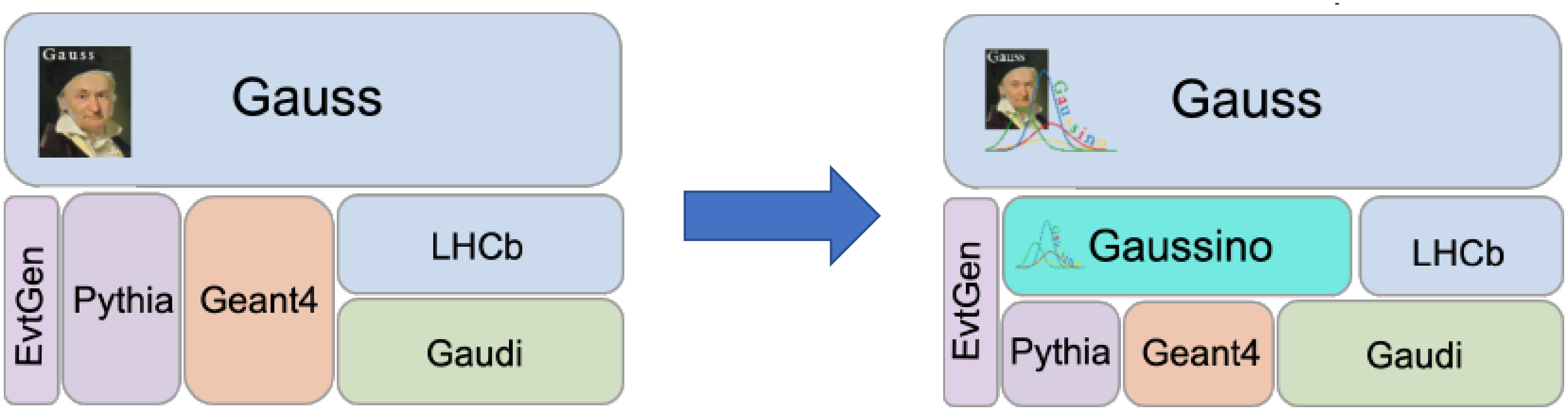
- ATLAS: from AthenaMP to AthenaMT
- LHCb: from Gauss to Gauss-on-Gaussino
- CMS: CMSSW
- ALICE: FairROOT and O2

=> Gaussino: experiment independent framework based on Gaudi

Ref: See Talk given by Michał Mazurek, Gauss and Gaussino, ICHEP 2022

Explore Gaussino

Evolution of simulation software in LHCb

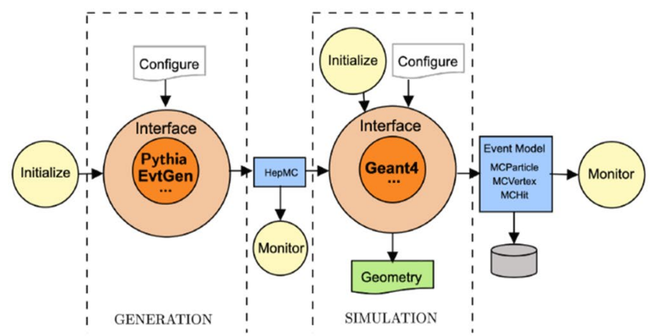


- Gaussino as the new core simulation framework, created by extracting experiment-independent from Gauss.
- Gauss-on-Gaussino as the new version of LHCb simulation framework, based on Gaussino's core functionalities.

Ref: [See Talk given by Michał Mazurek, Gauss and Gaussino, ICHEP 2022]

Features of Gaussino

- A complete simulation framework architecture with multi-threaded implementation



- similar modularity,
- integrated generation and simulation phase,
- similar event model,
- Gaudi tools, algorithms etc.
- python configuration,

Key concepts

- higher-level configuration in python,
- multi-threaded event loop,
- multi-threaded Geant4,
- interface to fast simulations,
- interfaces to new external libraries, e.g. DD4Hep,
- possibility to run in a standalone mode,

Execution structure

- use Gaudi functional,
- every algorithm as a ‘task’,

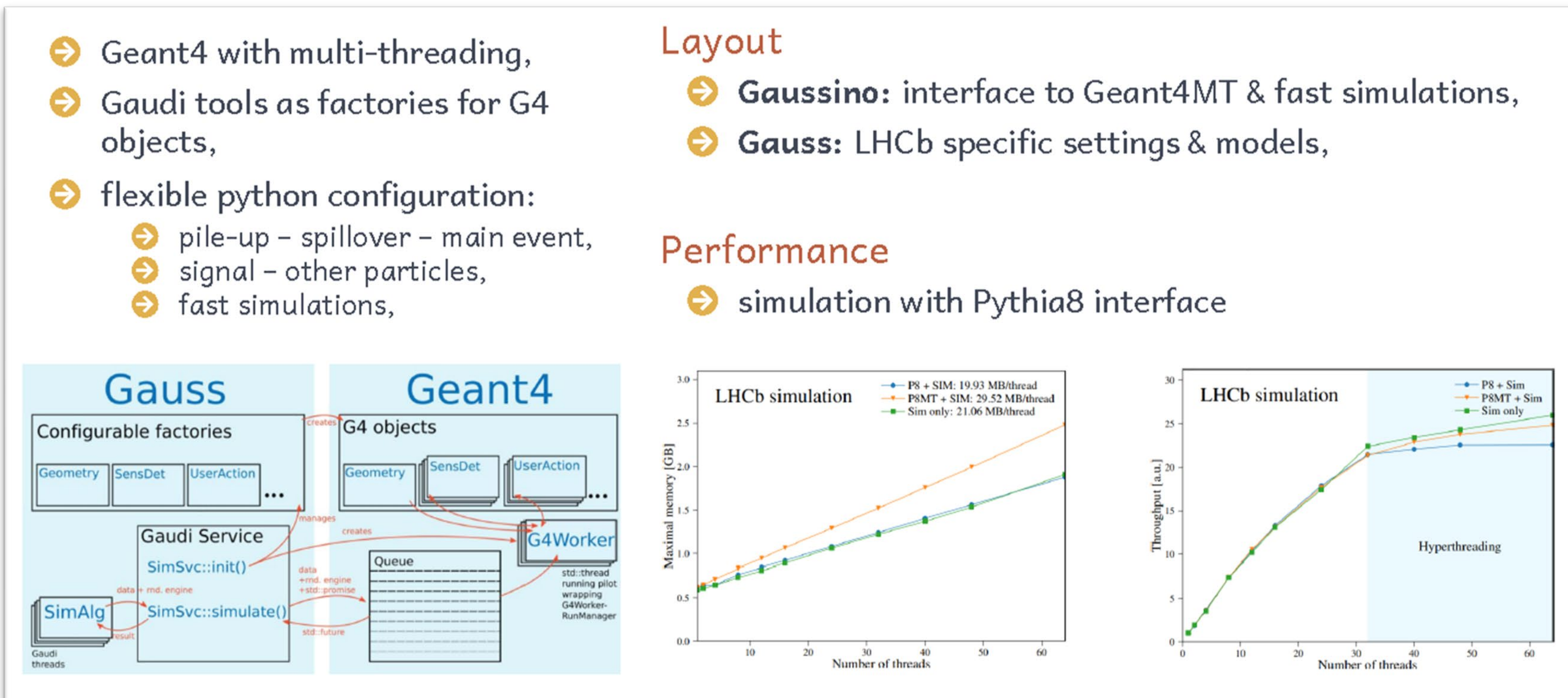
Random numbers

- ensure reproducibility,
- seed initialized with:
  - run #,
  - event #,
  - algorithm instance name,
- create random engines on the stack,

Ref: See Talk given by Michał Mazurek, Gauss and Gaussino, ICHEP 2022

Multi-threaded simulation in Gaussino

- Integrate multi-threaded Gaudi and multi-threaded Geant4.
- Achieve good scalability (see example below)



Ref: [See Talk given by Michał Mazurek, Gauss and Gaussino, ICHEP 2022]

Fast simulation in Gaussino

- Rich models for LHCb.
  - Maybe reuse them in CEPC.

Model	Generation	Decay	Propagation	Status in G-on-G
ReDecay	✓	✓	✓	done
ParticleGun	✓	✓	✓	done
SplitSim	✓	✗	✓	done
RICHless	✗	✗	✓	under tests
TrackerOnly	✗	✗	✓	under tests
Lamarr	✗	✗	✓	in progress
Point library	✗	✗	✓	in progress
GANs	✗	✗	✓	in progress

In-house parametrizations

Lamarr

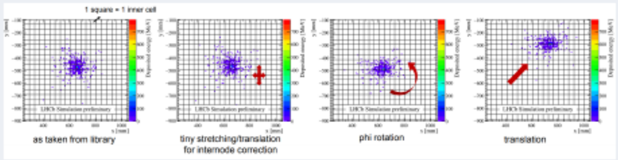
Idea: ultra-fast simulation option where not only the detector response, but also the reconstruction is parametrized

more on Lamarr in another ICHEP 2022 talk by L.Anderlini

Fast simulation models with Geant4

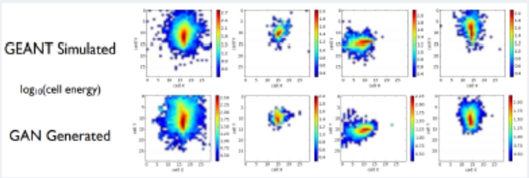
Point library

Idea: Extract points from a collection and transform them based on properties of the particle



Generative Adversarial Networks (GANs)

Idea: use GANs trained on the data produced by a detailed simulation to generate showers in ECAL



Ref: See Talk given by Michał Mazurek, Gauss and Gaussino, ICHEP 2022

Possibility of using Gaussino in CEPC

Gaussino and the current simulation framework in CEPCSW share a lot of similar underlying libraries, such as Gaudi, Geant4, DD4hep. Therefore, it is possible to use a unified simulation framework.

Several major technical issues need to be solved.

- How to build the Gaussino in CEPCSW environment?
- How to use EDM4hep in Gaussino?
- How to reuse the existing geometry service and detector description?
- How to reuse the detector response?

Current status and Plan

✓ Build and run Gaussino in LHCb build environment

☐ Build and run Gaussino in CEPCSW environment

☐ Integration with CEPC geometry

Testing Gaussino in LHCb env

- The dependencies of Gaussino is deployed via LbEnv.
  - The nightlies builds in CVMFS are used for the testing.
  - External Libraries: LCG 101x + x86_64_v2-centos7-gcc11-dbg
  - A trick is using a URL shortener to download the binaries from LHCb web server instead of xrootd server.

```
export CMAKE_PREFIX_PATH=/cvmfs/lhcbdev.cern.ch/nightlies/lhcb-gaussino/latest:$CMAKE_PREFIX_PATH
export CMAKE_PREFIX_PATH=/cvmfs/lhcbdev.cern.ch/nightlies/lhcb-gaussino/latest/GaussinoExtLibs/InstallArea/x86_64_v2-centos7-gcc11-dbg:$CMA
export CMAKE_PREFIX_PATH=/cvmfs/lhcbdev.cern.ch/nightlies/lhcb-gaussino/latest/LHCb/InstallArea/x86_64_v2-centos7-gcc11-dbg:$CMAKE_PREFIX_P
export CMAKE_PREFIX_PATH=/cvmfs/lhcbdev.cern.ch/nightlies/lhcb-gaussino/latest/Gaudi/InstallArea/x86_64_v2-centos7-gcc11-dbg:$CMAKE_PREFIX_
export CMAKE_PREFIX_PATH=/cvmfs/lhcbdev.cern.ch/nightlies/lhcb-gaussino/latest/Detector/InstallArea/x86_64_v2-centos7-gcc11-dbg:$CMAKE_PREF
export CMAKE_PREFIX_PATH=/cvmfs/lhcbdev.cern.ch/nightlies/lhcb-gaussino/latest/Geant4/InstallArea/x86_64_v2-centos7-gcc11-dbg:$CMAKE_PREFIX
```

- Detector setup: "External Tube" in example
  - enable decay physics constructor fix the crash problem
- All the testing are passed.

	Single threaded	Multi threaded
Gen only	OK	OK
Gen + Sim	OK	OK

Thanks the help from Graeme A Stewart

Summary and Plan

Summary

- CEPC simulation framework is developed to support the physics and detector performance studies.
- Explore Gaussino from LHCb as a possibility of unified simulation framework.

Plan

- Evaluate the integration of Gaussino into CEPCSW while reuse the existing code.

Thank you!