Overview of Key4HEP interfaces to generators

ECFA Higgs Factories: 2nd Topical Meeting on Generators

June 21, 2023

G Ganis, CERN-EP

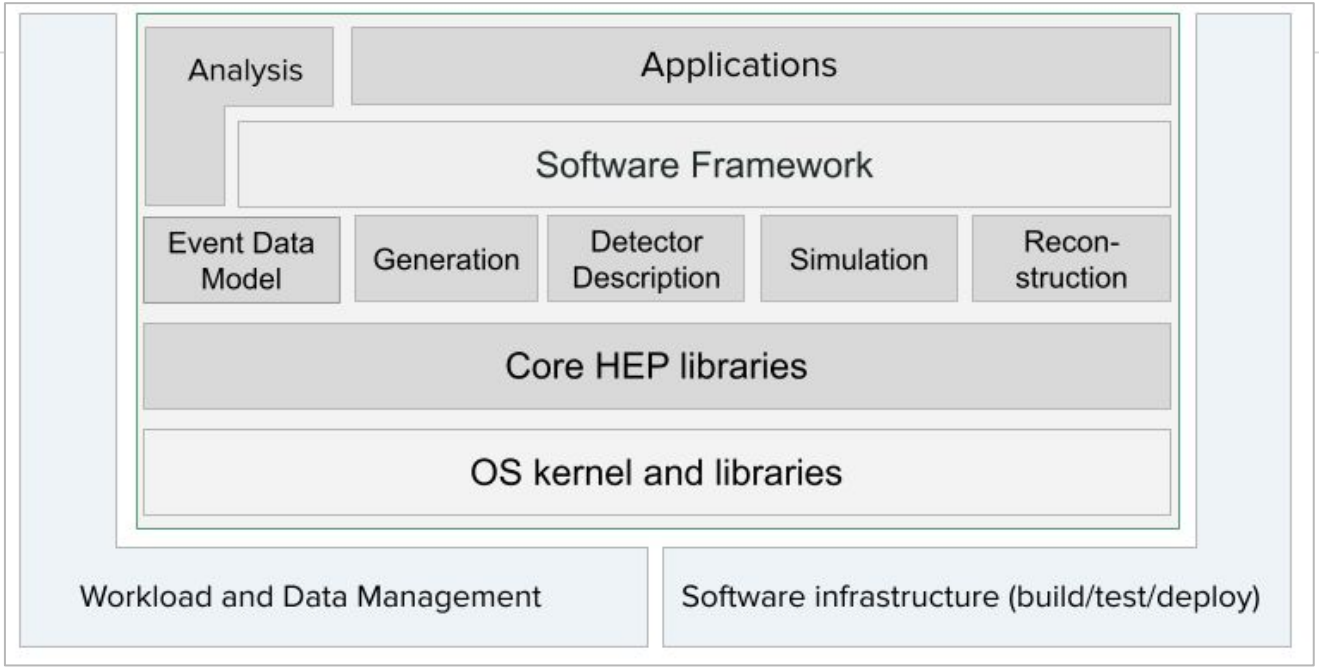
Outline

- Key4HEP reminder
- Monte Carlo generators @ key4HEP
  - Interoperability levels and managements
  - Data formats
  - Role and current status of k4Gen
- Summary

HEP Software Ecosystem

See G Ganis @ ECFA Kick-off, June 2021

Seamless integration and optimization between various networks of devices, software and services aimed to facilitate data processing for High Energy Physics experiments



Key4HEP: Turnkey Software Stack

Structured software stack integrating in optimal way various software components to provide a ready-to-use full-fledged solution for data processing of HEP experiments

Complete set of tools for

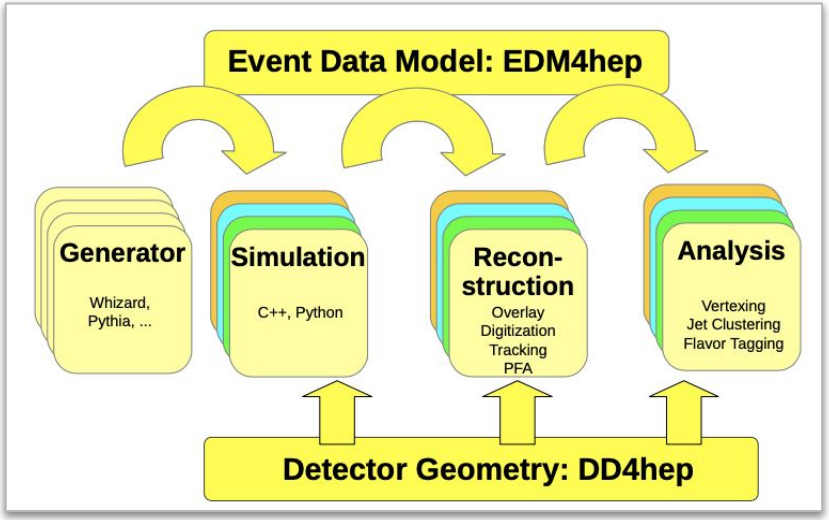
- Generation, simulation, reconstruction, analysis
- Build, package, test, deploy, run

Main Ingredients

- Event Data Model (PoDIO/EDM4hep), used also for persistency, processing framework (Gaudi), geometry information (DD4hep), Package manager (Spack)

Community project, unifying efforts

- Contributions from CLIC, ILC, FCC, CEPC, EIC, MuonCol
- Support from major R&D programs



Kick-off meetings in Bologna, Hong Kong

Key4HEP status and adoption

- Fully adopted by FCC
  - Building block of the feasibility study
  - Used also for FCC-hh studies
- Increasing adoption by CLIC, ILC, CEPC
- Interest by other projects (C3) and beyond the initial e+e- collider communities

Recent CHEP 2023 presentation

Where can key4hep be found?

- Key4hep is available on the CernVM-FS shared distributed file system

```
$ source /cvmfs/sw.hsf.org/key4hep/setup.sh
Setting up the latest Key4HEP software stack from CVMFS …
$ which KKMCee
/cvmfs/sw.hsf.org/spackages7/kkmcee/5.00.02/x86_64-centos7-gcc11.2.0-opt/vfxxk/bin/KKMCee
$ which mg5
/cvmfs/sw.hsf.org/spackages7/madgraph5amc/2.8.1/x86_64-centos7-gcc11.2.0-opt/nlauf/bin/mg5
$ pythia8-config --libdir
/cvmfs/sw.hsf.org/spackages7/pythia8/8.306/x86_64-centos7-gcc11.2.0-opt/dhhih/lib
```

- CernVM-FS optimizes remote access
  - LHC driven solution adopted widely
  - Available for any unix-like system
  - Requires internet connection (but solutions to work off-line available)

Monte Carlo generators @ key4hep

Monte Carlo Generators in key4hep

- A Monte Carlo generator is a package
- Key4hep includes already many generators as packages
  - Initial list derived from LCG stacks, so sort of LHC oriented
  - But several e+e- additions available: Whizard, KKMCee, BabaYaga, BHLUMI, …
    - Including wrappers for better user experience
- What does it mean “adding a generator to key4hep”?
  - Required information for inclusion in the package manager
    - Source location, minimal documentation on how to build and required dependencies, default configuration files, tests, ...
  - Key4hep infrastructure will
    - Build in shared installation mode
    - Run built-in tests, if any
    - Install in distributed shared file system

List of generators currently available in key4hep

T Mandlener, Oct 2022

- Generators

babayaga*†	baurmc†	bhlumi*†	crmc†	evtgen	genie†
gosam†	guinea-pig*†	herwig3	herwigpp†	kkmcee*	madgraph5amc
photos	pythia6†	pythia8	sherpa	starlight†	superchic†
tauola†	vbfnlo	whizard			

- "Generator tools"

agile†	alpgen†	ampt†	apfel†	ccs-qcd†	chaplin†
collier†	cuba†	dire†	feynhiggs†	form†	hepmc
hepmc3	heppdt	hoppet†	hztool†	lhapdf	lhapdfsets†
looptools	openloops	professor†	prophecy4f†	qd†	qgraf†
recola†	rivet	syscalc†	thepeg	unigen†	yoda

The availability of the latest version depends on requests from community

Levels of interoperability

- Level 0 - *Common Data Formats*
  - Maximal interoperability, even on different hardware
- Level 1 - *Callable Interfaces*
  - Defined for one or more programming languages
  - Implementation quality of interfaced components important
  - Required to define plugins
- Level 2 - *Introspection Capabilities*
  - Software elements to facilitate the interaction of objects in a generic manner such as Dictionaries and Scripting interfaces
  - Language bindings, e.g. PyROOT
- Level 3 - *Component Level*
  - Software components are part of a *common framework*, optimal interplay
  - Common configuration, log and error reporting, plug-in management, ...

Managing interoperability through Gaudi

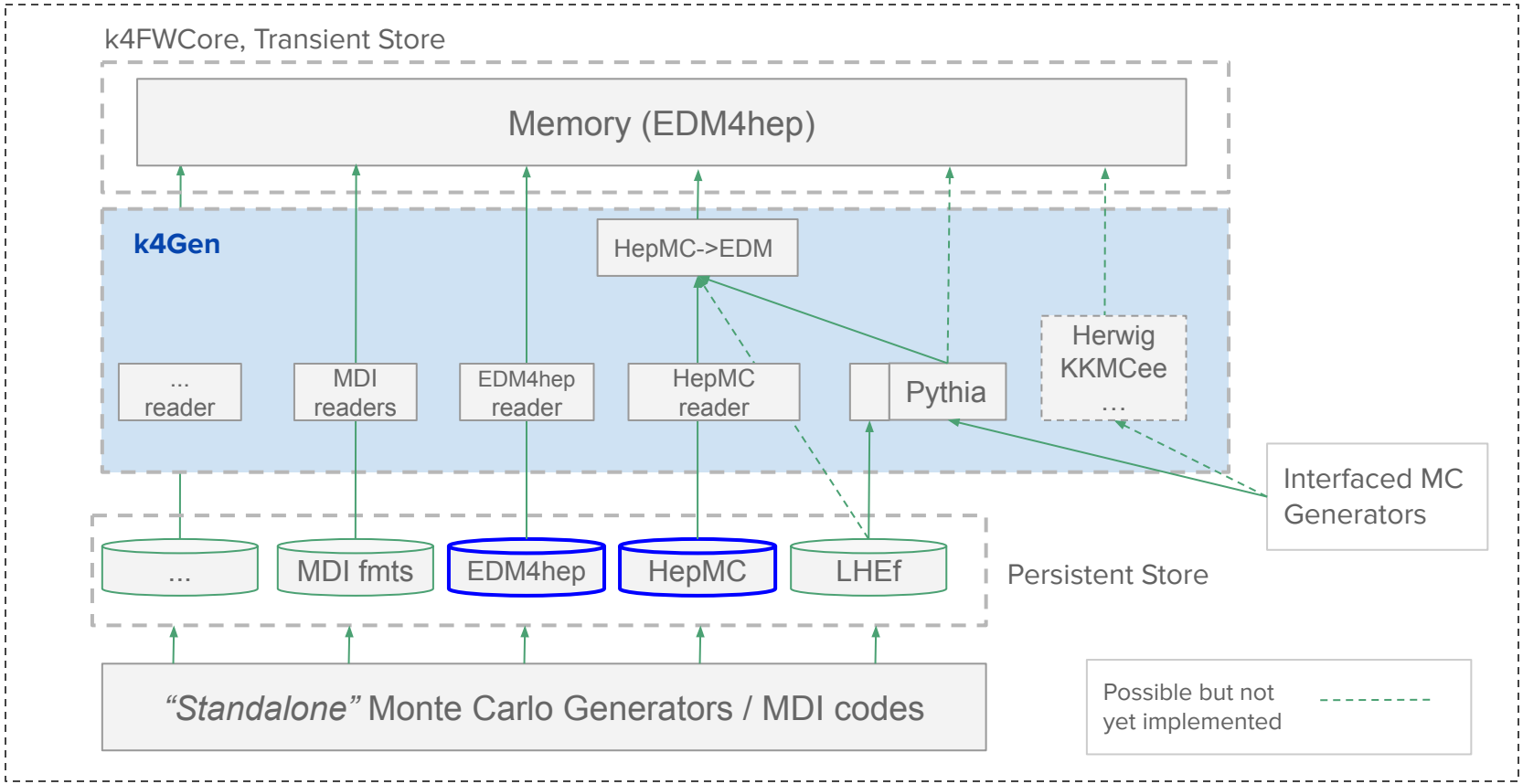
- Components tailored for specific tasks
  - Tools for inputting and managing MC outputs
  - Derived from LHCb approach
- What is required
  - Readers for data formats
    - Currently available: HepEVT, HepMC, LHEf, ...
  - Level-1 interfaces for MC
    - Only currently available Pythia8; KKMCee5 under prototyping
    - Place where to put an interface to Herwig

About data formats

- EDM4hep
  - Baseline event data model
  - Default format managed by the DataSvc in k4FWCore
- HEPMC
  - Generic framework for MC generator event record encoding and manipulation
    - Very popular among MC authors
  - Version 3 provides also several interfaces to other formats
- HepEVT, StdHEP
  - Formats used by old FORTRAN generators
- LHEf
  - Designed for MC internals (matrix elements, …), not for final states
  - Should be left for internal use of authors

Discussion at 1st ECFA generator workshop

Managing interoperability in Gaudi



Technical follow up of 1st ECFA Gen Workshop

- Have a small number of well defined I/O output(s)
  - Efficient(s) and compressed
  - Baseline: EDM4hep@{ROOT, SLCIO}, HEPMCv3@ROOT (common among MCs)
  - Possibly provide / contribute additional converters to HepMC, e.g. EDM4hep
- Boundary generators / key4hep
  - Baseline: generators provide fully hadronized events
    - In particular, dedicated decayers, hadronizers, should be run before, i.e, by the generator
  - Key4hep starts from the generator output
    - list of final state particles (typically including incoming and intermediate)
    - HEPMC3 / EDM4hep formats preferably
  - Modules providing a direct MC to EDM4hep interface can still be accommodated
    - E.g. PythiaInterface, HerwigInterface, …
    - Possibly under the responsibility of the authors (to be discussed)

k4Gen functionality

The repository is k4Gen, and includes also modules to perform actions on the MC events

- *Event readers, converters:* HepMC, HepMCv2, HepEVT, MDI, HepMC->EDM4hep->HepMC
- *Generator interfaces*: Pythia, KKMCee (coming)
- *Filters:* Generated particles status, Decays hooks
- *Pileup tools*: Const, Range, Poisson
- *Particle guns*: Momentum range, const Pt
- *Vertex smearing*: Flat, Gauss
- …

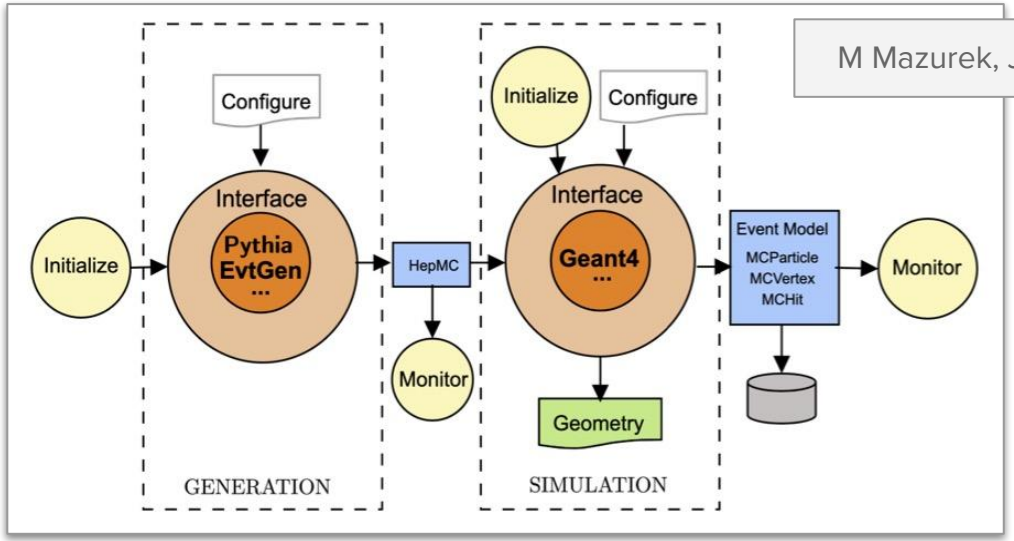
Some considerations

- Complexity of interaction regions and beam conditions might require “more” from the generation phase
- For example, in FCC
  - Crossing-angle
  - Beam Energy Spread
  - Vertex spread
  - Beam-beam effects
  - …
- Some of these can be approximately factorised out and accounted for in a common way for all generators
- But the ultimate treatment should come from a proper treatment inside generators, possibly w/ common technology

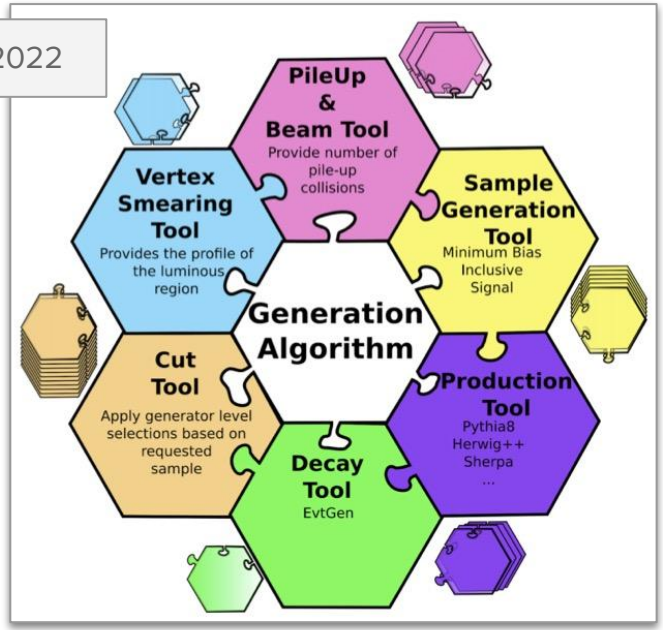
Gaussino: improving gen / sim Gaudi interfaces

Gaussino
Doc, GitLab

- LHCb latest development in the field, in production in Run3
- Streamlines integration of generations and fast/full simulation in Gaudi
  - Evolution of current Key4HEP approach improving on multi-threading and reproducibility



M Mazurek, Jul 2022



- Being imported in Key4hep

Summary

- Key4hep as acquired a prominent role in addressing the software needs of its immediate stakeholders, i.e. EW/Higgs factories
- Key4hep provides
  - A coherent software stack
    - Increasingly complete and centrally distributed
  - A sanitized environment, w/ internal consistency tests
    - Version tracking
- For MC Generators, a 2-level interoperability interface is provided
  - Matching current needs of projects
  - But ready to be adapt to evolving needs, exploiting the flexibility of the underlying framework

Useful links

- Key4hep GitHub Project
  https://github.com/key4hep
- Main documentation page
  https://key4hep.github.io/key4hep-doc/
- Doxygen available., e.g. for EDM4hep
  https://edm4hep.web.cern.ch/