Software for PED studies

Key4hep and FCC Software

FCC Physics Workshop 2023
Krakow, Poland

January 24, 2023
G Ganis, CERN-EP

A reminder of what Key4hep is today

Key4hep, the common software vision

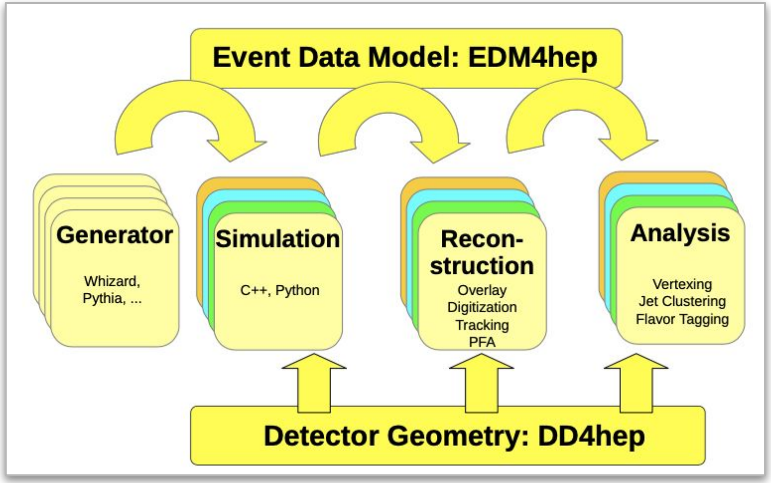
Create a software ecosystem integrating in optimal way various software components to provide a ready-to-use full-fledged data processing solution for HEP experiments

Complete set of tools

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

Common Core ingredients

- PoDIO for EDM4hep, based on LCIO and FCC-edm
- Gaudi framework, devel/used for (HL-)LHC
- DD4hep for geometry, adopted at LHC
- Spack package manager, lot of interest from LHC



An international community project

- Unifying communities, synergetic enterprise
- Contributions/interest from CEPC, CLIC, ILC, FCC, EIC and MuonCol
- In person meetings Bologna (6/2019, kick-off), Hong Kong (1/2020), DESY (5/2022)
- Weekly working meetings
- Funded by CERN EP R&D (WP7)
    - Two fellows in Phase I (2020-2024): V Volkl, P Declara, now replaced
    - Proposal being submitted for Phase II (2024-2028)
- Part of AIDAInnova software R&D (WP12)
    - Could potentially also profit of some person power for specific tasks
- Supported by ECFA
    - Connections w/ WG2, WG3

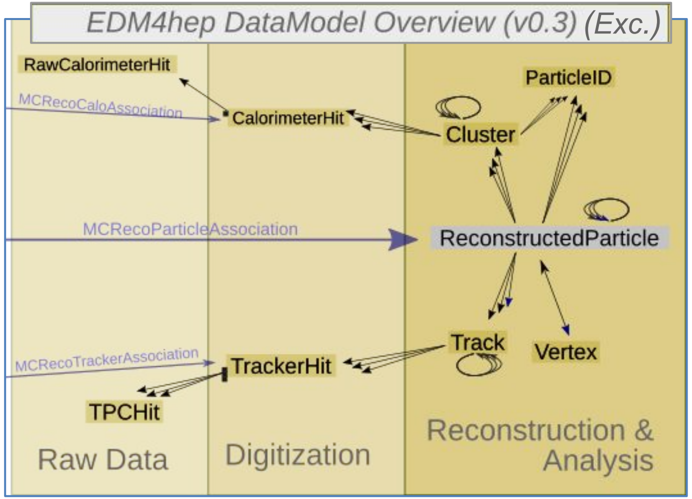
Adoption

- FCC
- CEPCSW using EDM4hep and switched from Marlin to Gaudi
- CLIC and ILD reconstruction can be run in Gaudi
  - Part of AIDAInnova WP12
- EIC adopted EDM4hep
  - With an extension
- Several ongoing activities to
  - Migrate into EDM4hep several tools
  - Integrate more packages of general interest
    - E.g. ACTS, Pandora
  - Migration opportunity to modernize

EDM4hep, a common event data model

Provides common language for exchange among framework components

- Based on LCIO and FCC-EDM
- Data structures supporting needs of different collision environments (e+e-, pp, ...)
  - Managed to adapt both FCC-ee and FCC-hh software specific components
- And *truth* information (MC particles)
  - ECFA advised format for MC output
- Underline tool: Podio (AIDA development)
- Ongoing developments / prototypes
  - Conversion to Phoenix JSON (for visualisation)
  - Schema evolution to adapt to sub-detector needs
    - Eg. cluster counting for the IDEA Drift Chamber
  - I/O optimisations (RNtuple)
    - Plain-Old-Data layer designed for efficient I/O



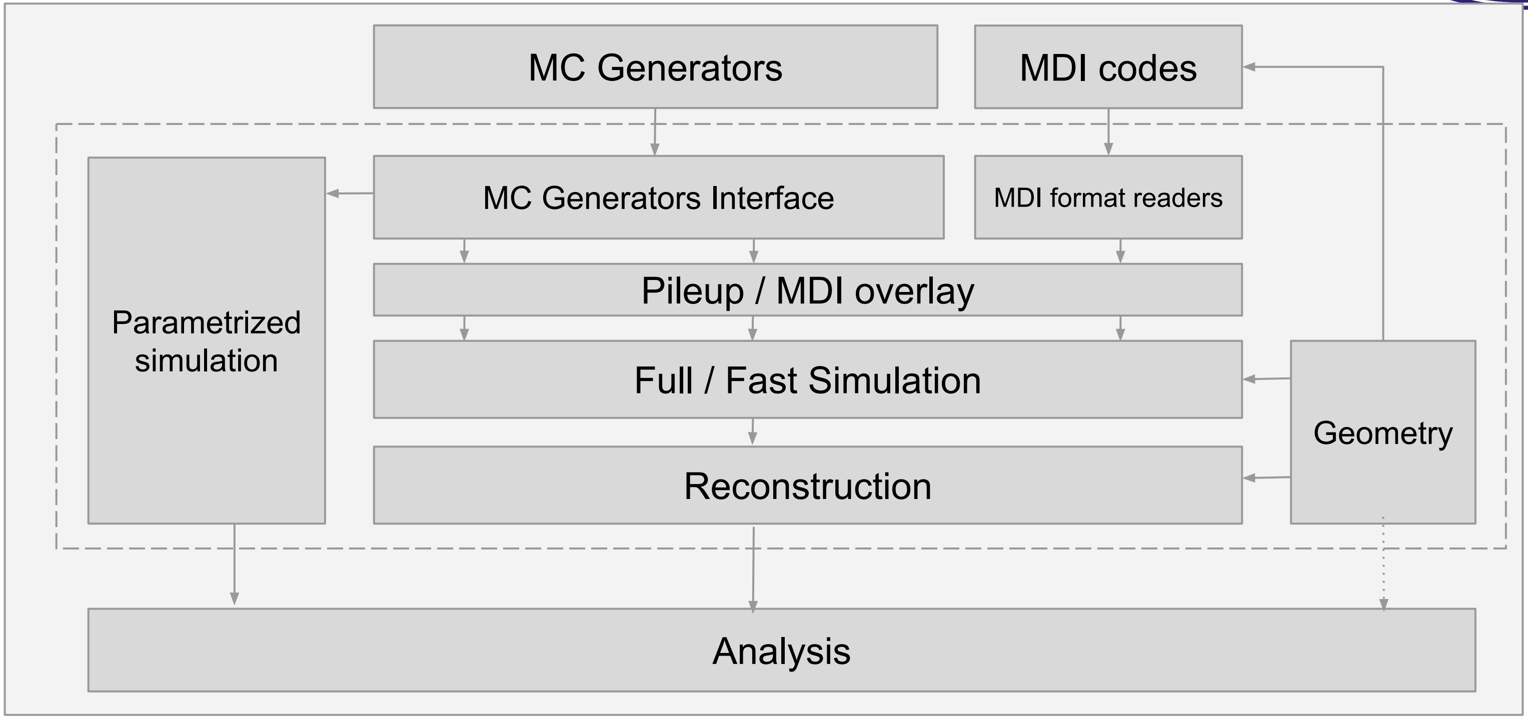
Key4hep practicalities (sources, binaries)

- Ecosystem of about 70 integrated packages specific to future HEP projects needs
  - A few main GitHub projects (key4hep, HEP-FCC, iLCSoft)
- Some of main core repositories
  - k4FWCore, Gaudi steering, produces k4run command and data input handlers
  - EDM4hep, the common event data model
  - key4hep-doc, the documentation, source for https://cern.ch/key4hep
  - k4SimDelphes, the Delphes integration
  - …
- Managed with Spack package manager
  - key4hep-spack, recipe repository of Key4hep software used by all communities
- Deployed on CVMFS, e.g. on lxplus
  - `/cvmfs/sw.hsf.org/key4hep/`

Check tutorials for examples of usage

Key4hep and FCC

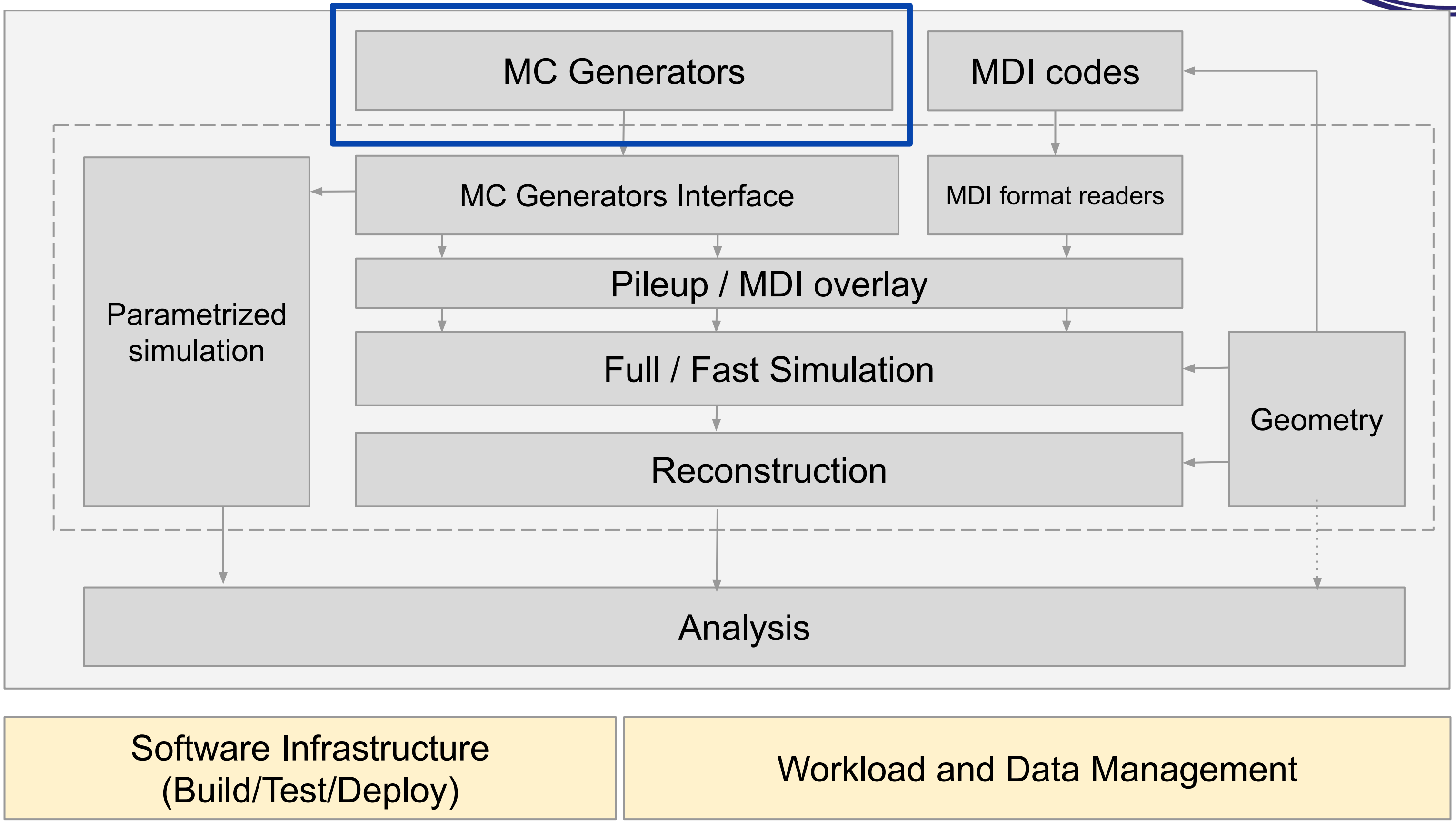
Workflows we need to support



Software Infrastructure (Build/Test/Deploy)

Workload and Data Management

Workflows we need to support

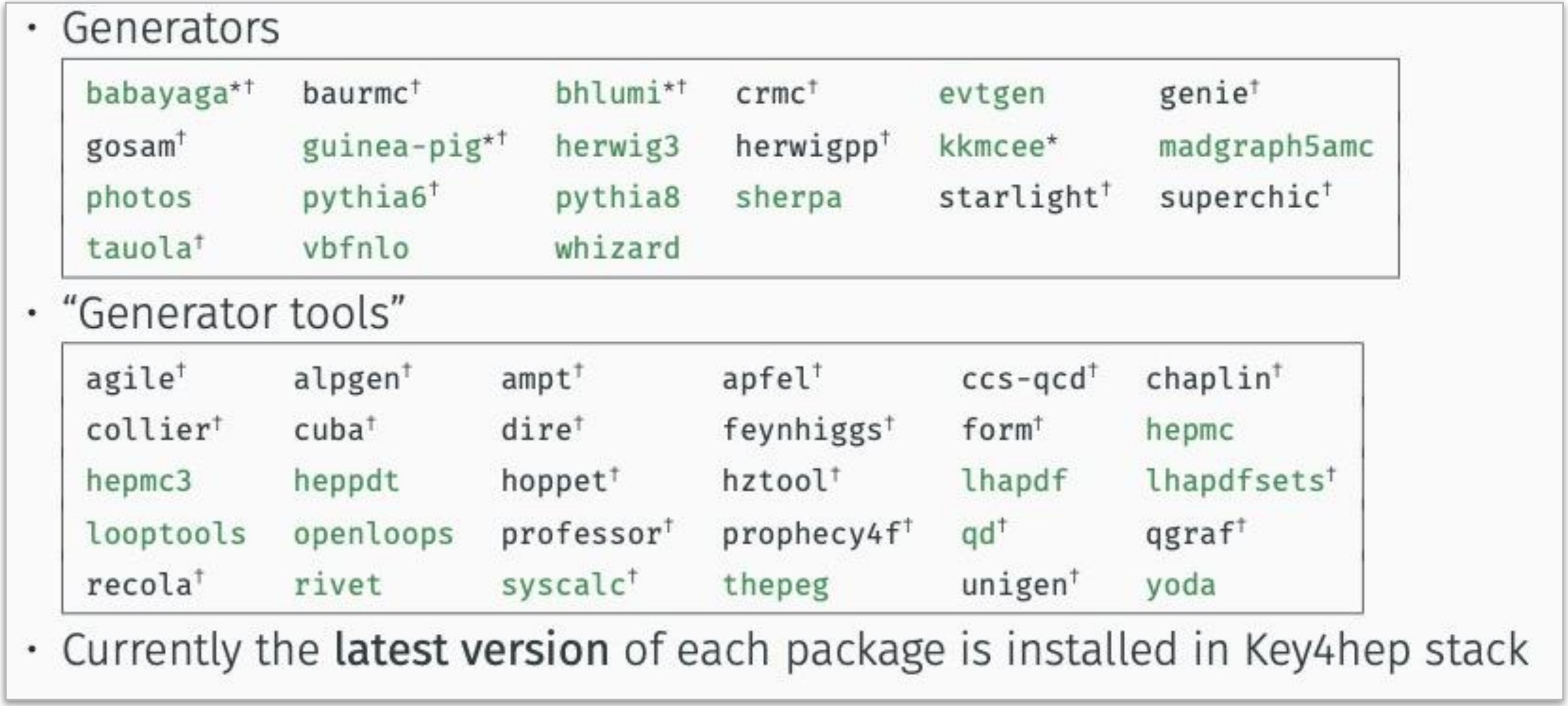


Monte Carlo Generators in key4hep

- A Monte Carlo generator is a package
- 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
- Key4hep includes already many generators as packages

Very useful 1st ECFA workshop on Simulation

List of generators currently available in key4hep



- Generators

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

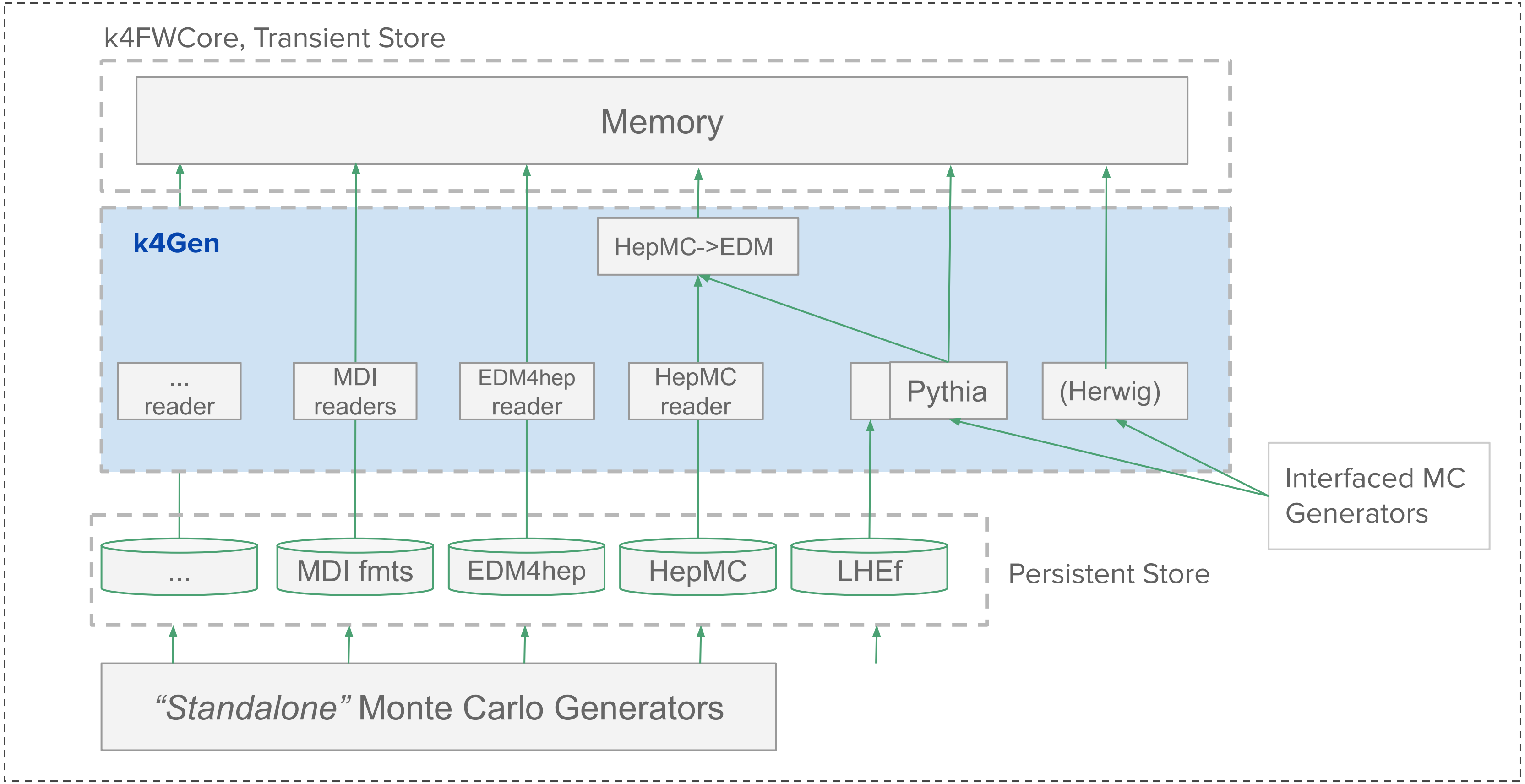
- “Generator tools”

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

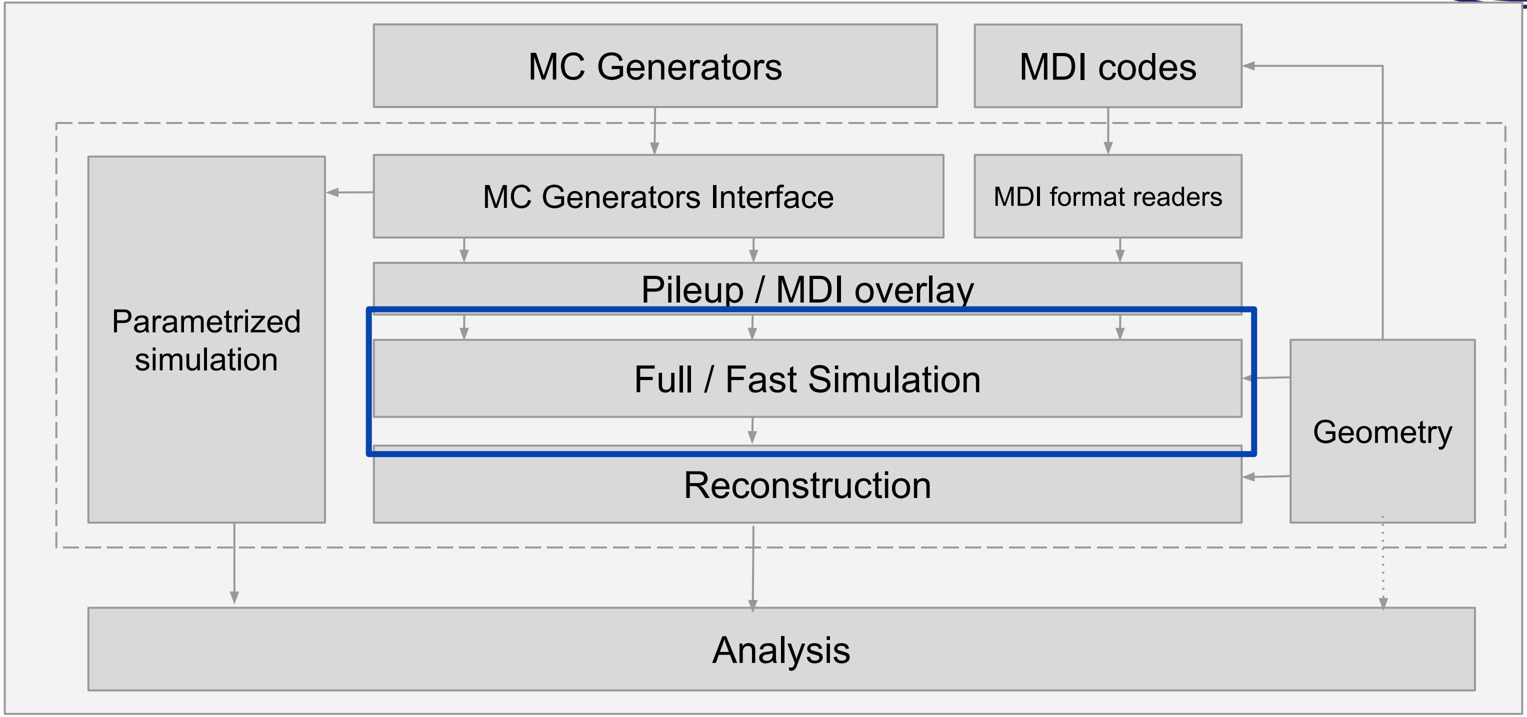
- Currently the **latest version** of each package is installed in Key4hep stack

T Mandlener, Oct 2022

Managing interoperability in Gaudi



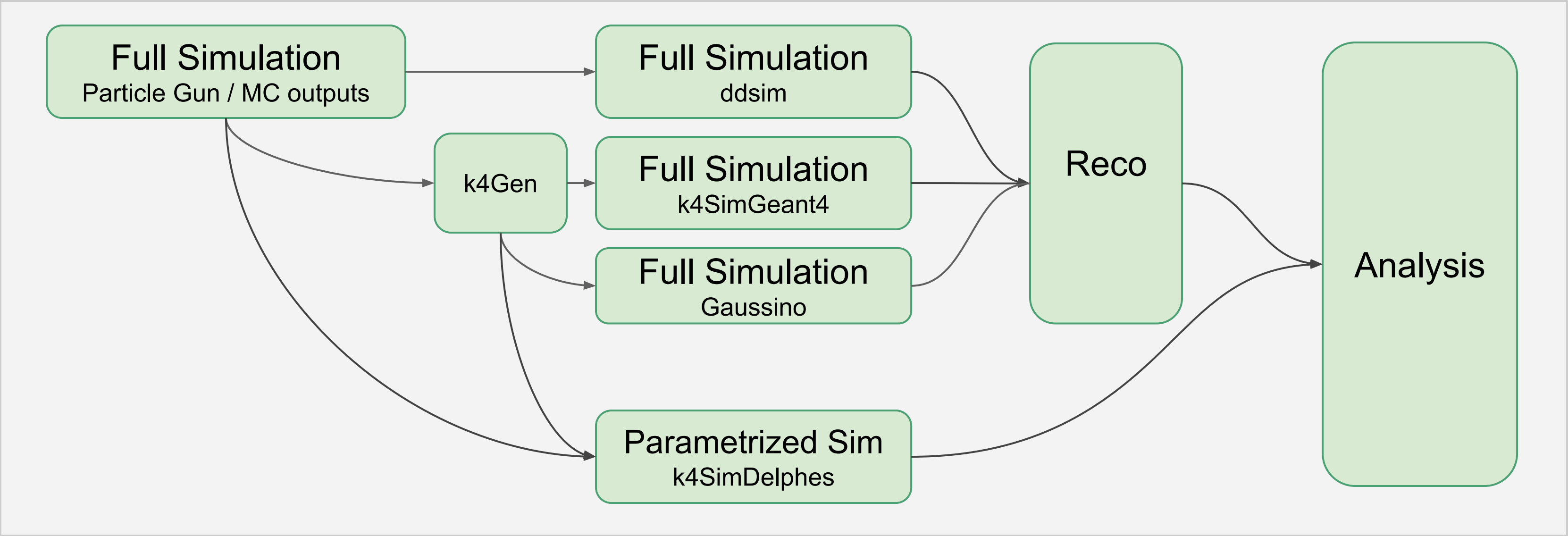
Workflows we need to support



Software Infrastructure (Build/Test/Deploy)

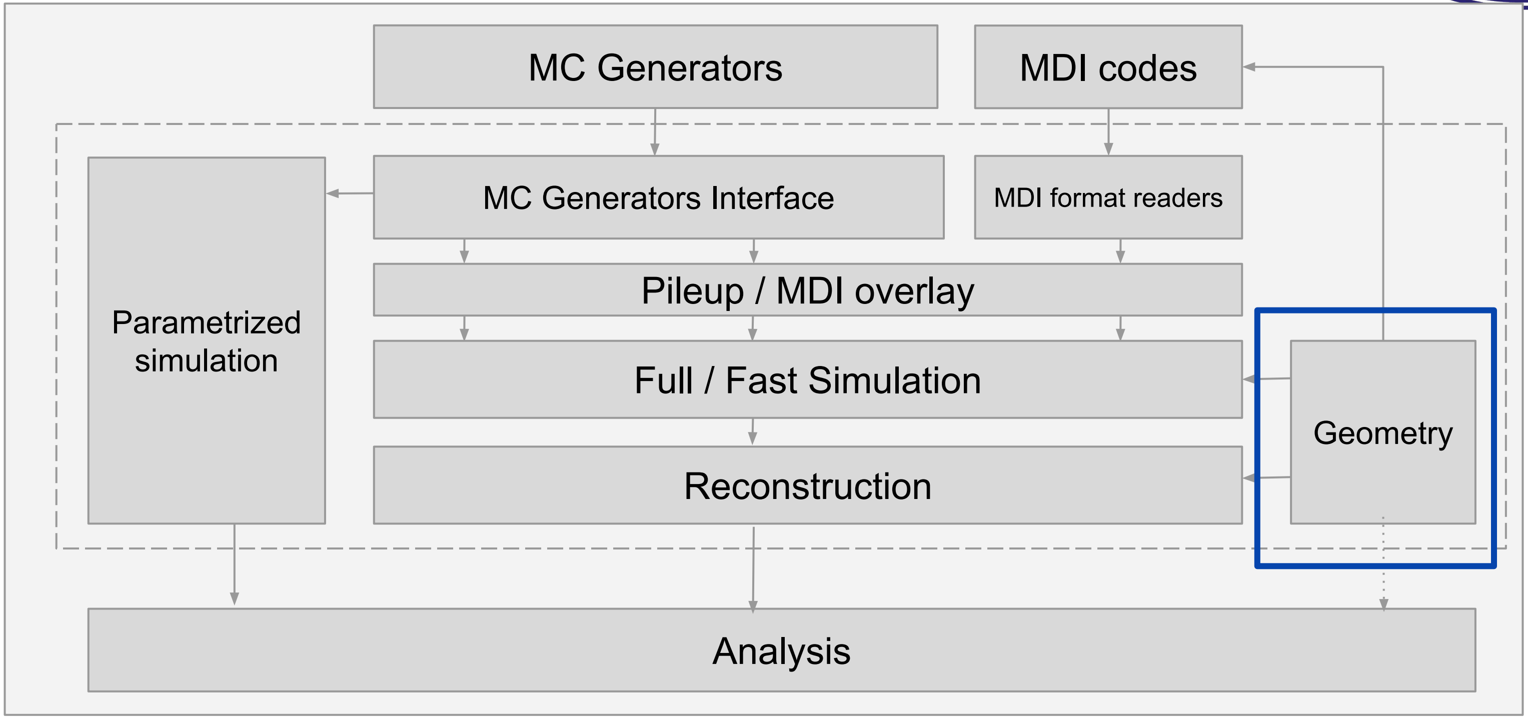
Workload and Data Management

Integrating simulation tools



- Can be run in standalone on the MC output or as a framework
  - Particle guns available at several levels
- In all cases the following steps (reconstruction, analysis) are the same

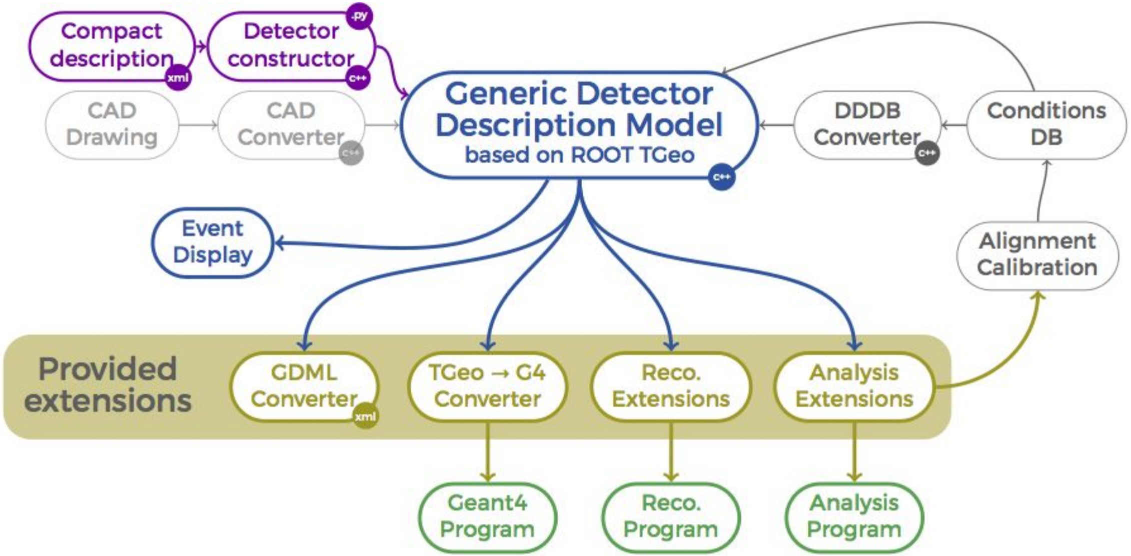
Workflows we need to support



Software Infrastructure (Build/Test/Deploy)

Workload and Data Management

DD4hep - Detector Description

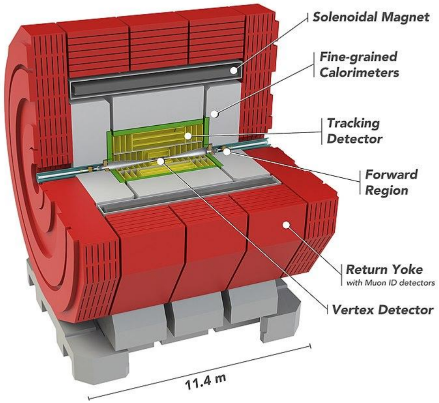


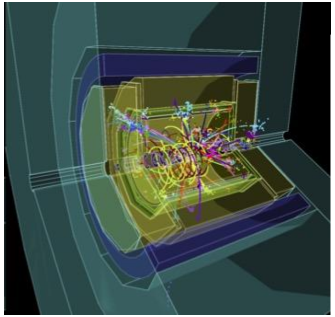
- Complete detector description: geometry, conditions, alignment, ...
- Used by CLIC, ILC, CEPC, FCC, EIC, CMS, LHCb, …

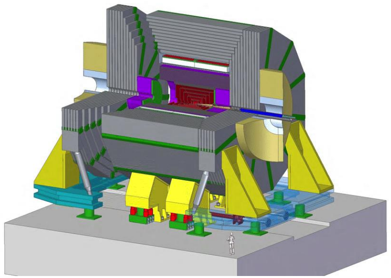
k4geo: common repository for detector models

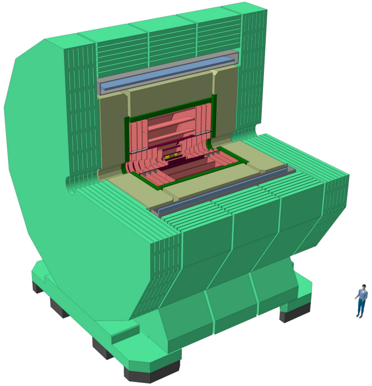
NEW

- DD4hep detector models spread in several project repositories
  - iLCSoft/lcgeo, FCCDetectors, CEPCSW/Detector
- k4gen is an attempt to reduce risk of duplications
  - E.g. CLD will only appear in one place
- Features versioning convention developed by LC community (CLD == FCCee_o1_v05)

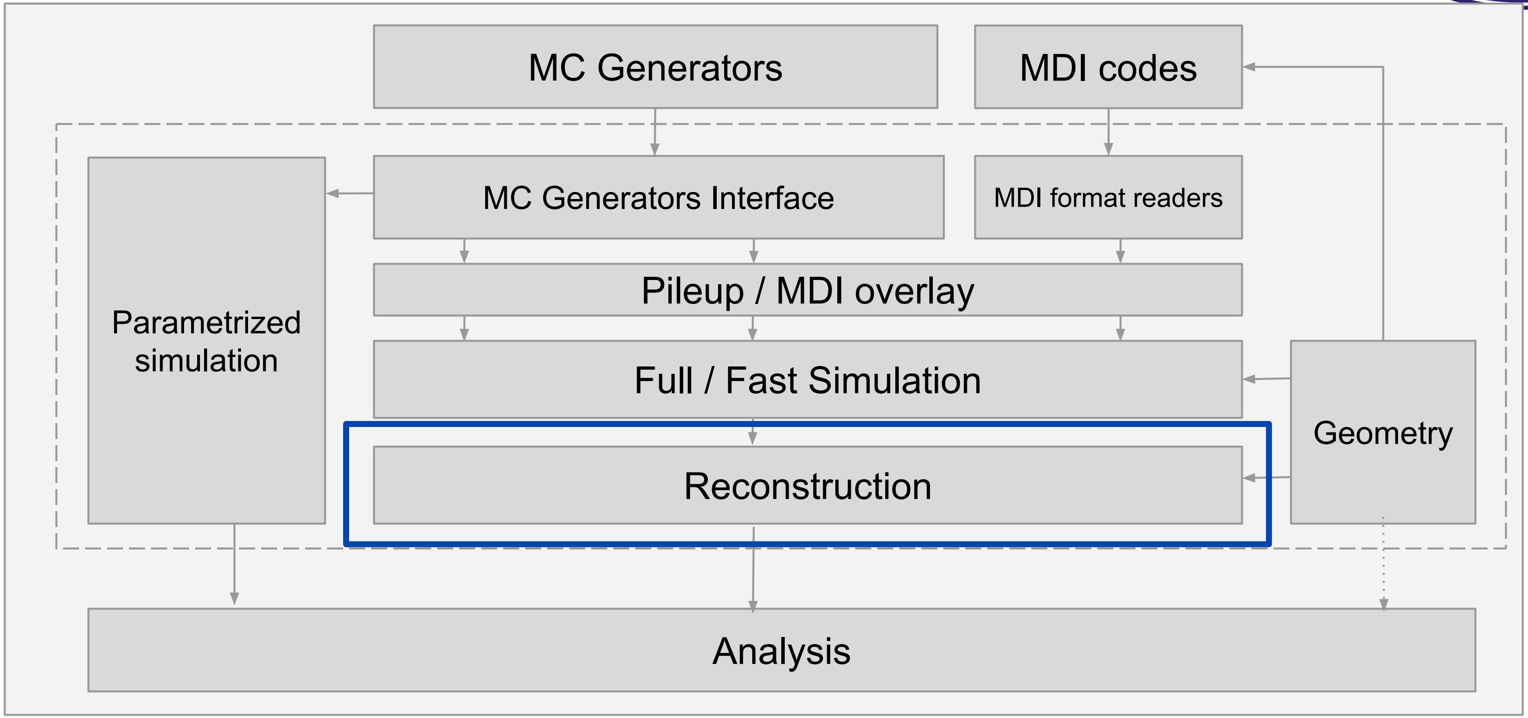








Workflows we need to support

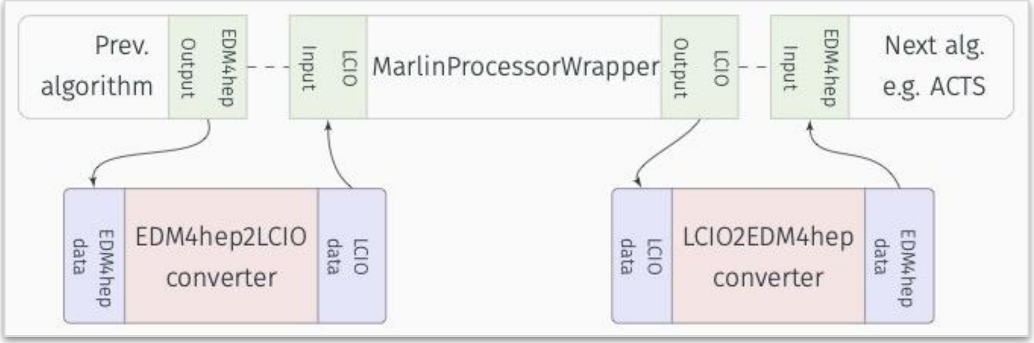


Software Infrastructure (Build/Test/Deploy)

Workload and Data Management

Reconstruction

- For CLD, CLICdp algorithms from iLCSoft available through k4MarlinWrapper



Possibly the area where there is more room for contribution

- Calo reconstruction algorithms in Gaudi
  - LAr, derived form FCC-hh LAr (see tutorial), IDEA Dual Readout
- Particle Flow: PandoraPFA available through wrapper, native implementation in the pipeline
- Other
  - CLUE, developed for CMS HGCAL, available as k4CLUE
  - ACTS, tracking in high density scenario, started by ATLAS; in the pipeline

Very useful 1st ECFA workshop on Reconstruction

Conclusions

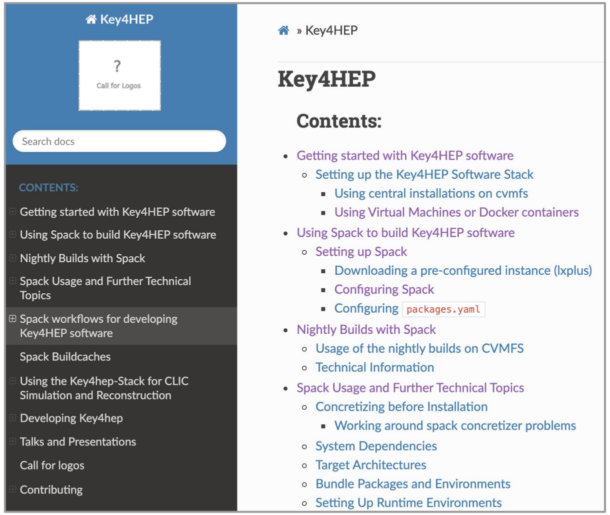
A few considerations

- Key4hep is undoubtedly successful
  - Used, to a different degree, by many projects, the concept behind had made its way in the spirit of many, as demonstrated by the support got from ECFA
- However
  - Community needs to stay focused on the main goal
  - The workforce is currently mostly staffed with short /medium term positions from R&D projects
    - R&D requires ‘innovation’ to get the money
    - A software project requires maintenance and support which do not justify R&D funds
    - We are are currently facing a change of crew and need to watch and operate so to minimise the impact
  - The question of long term support is coming up
    - A consortium model might be appropriate given the nature of the collaboration (Eg. DIRAC, XRootD)
    - Requires engagement of the laboratories

Take away messages

- After three years Key4hep is in rather good shape
  - Provides a consolidated set of workflows
    - And a framework for new contributions potentially of general interest
  - Considered de facto the standard in the mind of people
- FCC software is fully based on Key4hep and engaged in its support
  - The benefits of such approach are visible
- The community needs to start thinking the long term support

Documentation, tutorials, …



Key4HEP

?

Call for Logos

Search docs

CONTENTS:

- Getting started with Key4HEP software
- Using Spack to build Key4HEP software
- Nightly Builds with Spack
- Spack Usage and Further Technical Topics
- Spack workflows for developing Key4HEP software
- Spack Buildcaches
- Using the Key4hep-Stack for CLIC Simulation and Reconstruction
- Developing Key4hep
- Talks and Presentations
- Call for logos
- Contributing

» Key4HEP

Key4HEP

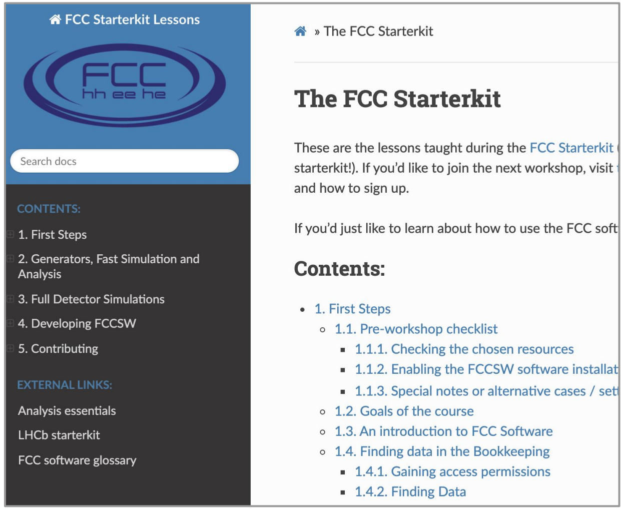
Contents:

- Getting started with Key4HEP software
  - Setting up the Key4HEP Software Stack
    - Using central installations on cvmfs
    - Using Virtual Machines or Docker containers
- Using Spack to build Key4HEP software
  - Setting up Spack
    - Downloading a pre-configured instance (lxplus)
    - Configuring Spack
    - Configuring `packages.yaml`
- Nightly Builds with Spack
  - Usage of the nightly builds on CVMFS
  - Technical Information
- Spack Usage and Further Technical Topics
  - Concretizing before Installation
    - Working around spack concretizer problems
  - System Dependencies
  - Target Architectures
  - Bundle Packages and Environments
  - Setting Up Runtime Environments

Key4hep GitHub Project

Main documentation page

Doxygen software documentation



FCC Starterkit Lessons

Search docs

CONTENTS:

- 1. First Steps
- 2. Generators, Fast Simulation and Analysis
- 3. Full Detector Simulations
- 4. Developing FCCSW
- 5. Contributing

EXTERNAL LINKS:

- Analysis essentials
- LHCb starterkit
- FCC software glossary

» The FCC Starterkit

The FCC Starterkit

These are the lessons taught during the FCC Starterkit (starterkit!). If you'd like to join the next workshop, visit t and how to sign up.

If you'd just like to learn about how to use the FCC soft

Contents:

- 1. First Steps
  - 1.1. Pre-workshop checklist
    - 1.1.1. Checking the chosen resources
    - 1.1.2. Enabling the FCCSW software installat
    - 1.1.3. Special notes or alternative cases / sett
  - 1.2. Goals of the course
  - 1.3. An introduction to FCC Software
  - 1.4. Finding data in the Bookkeeping
    - 1.4.1. Gaining access permissions
    - 1.4.2. Finding Data

FCCSW GitHub Project

Main documentation page