FCC Software Status

3rd FCC Physics Week 2020

Jan 14, 2020

G Ganis, C Helsens
CERN-EP

Software Requirements to support FCC studies

- Detector design studies
  - Flexibility
  - Ideal detector descriptions
  - Open to evolution
- Broad range of event complexity
  - e^+e^- *vs* pp *vs* ep
- Need to support physics and detector studies
  - Parameterised, fast and full simulation
- Aim to de-duplicate efforts
  - One software stack to support all the cases,
    all detector concepts and future (proto-)collaborations

FCCSW today

Current software met basic requirements at a sufficient level to support CDRs Physics studies

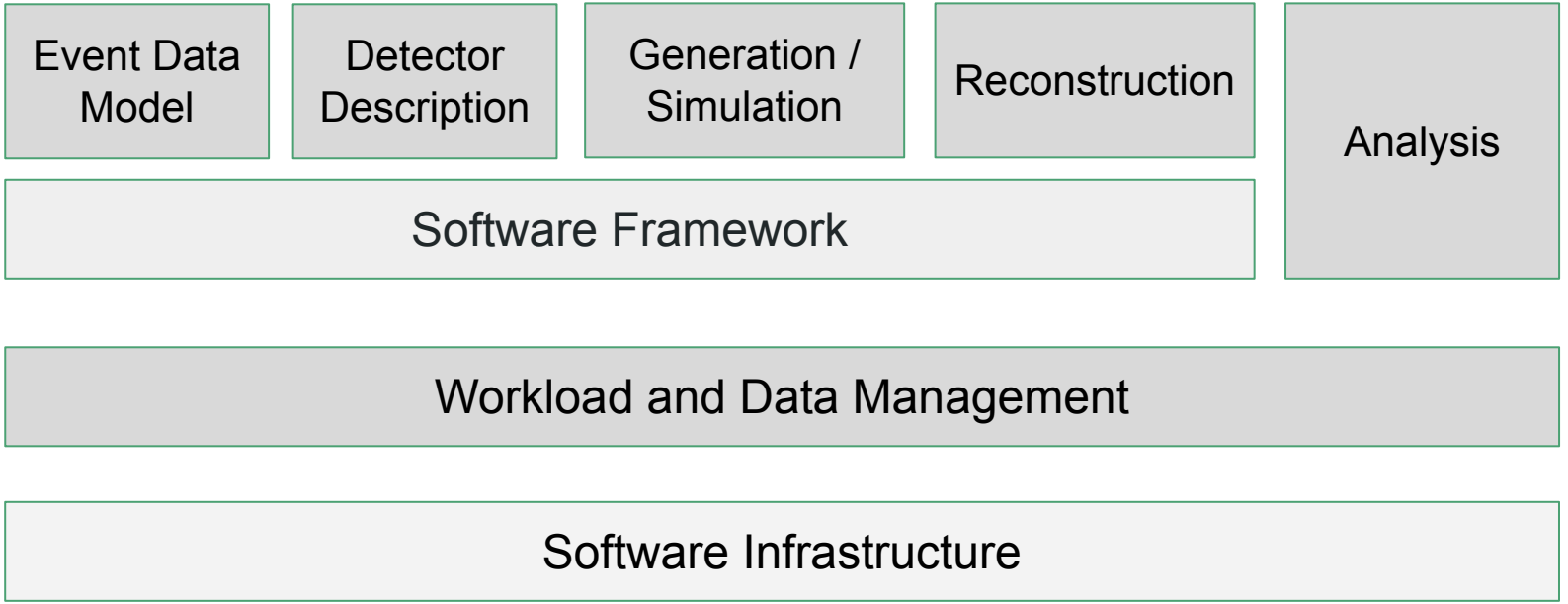
Next level of detail required for the Technical Design Report, with a special focus on FCC-ee

Adopted approach

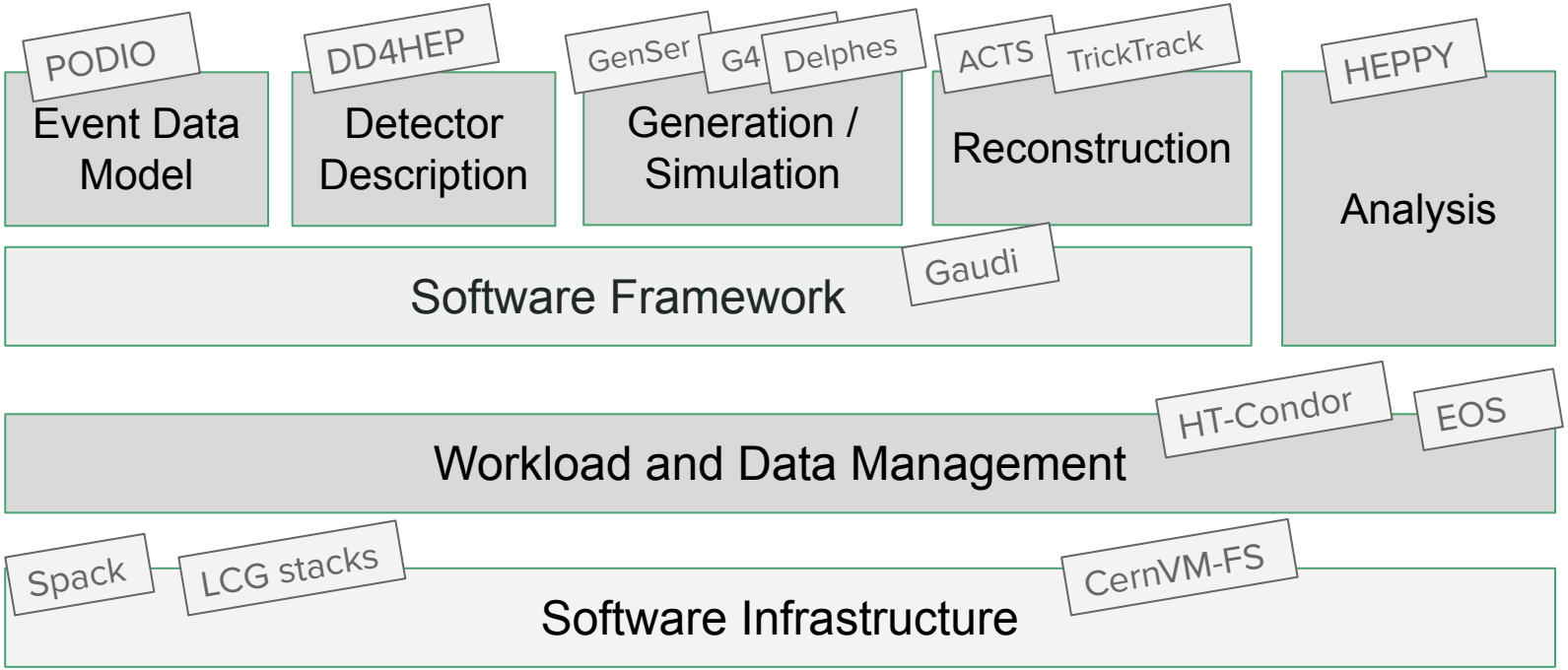
- Strategy
  - Adapt existing solutions from LHC
  - Look at ongoing common projects (AIDA)
  - Invest in streamlining of the event data model
- Future: towards a common software for future experiments
  - Bologna workshop, June 2019
    - Present: LHC, ILC, CLIC, FCC, CEPC, SCTF, HSF
  - Agreed to:
    - Investigate the possibility to have a common event data model (EDM4hep)
    - Contribute to the development of a Common Turnkey Software Stack (Key4hep)
      - One framework (Gaudi best candidate), DD4hep, EDM4hep, Geant4, ROOT, ...
  - Follow-up in Hong Kong, 17 January 2020

FCC Software Components

Components overview



Components overview



Event Data Model

Event Data Model

- Current FCC-EDM
  - Event/Run: EventInfo
  - MC truth: MCParticle, GenVertex, GenJet
  - Tracker: Track (PositionedTrackHit, TrackCluster, TrackState)
  - Calorimeter: CaloCluster (PositionedCaloHit)
  - Associations: ParticleMCParticleAssociation, DigiTrackHitAssociation, CaloHitAssociation, CaloHitMCParticleAssociation
  - High-Level objects: TaggedParticle, Vertex (WeightedTrack), TaggedJet, ResolvedJet, MET
- Tuned on the needs of FCC-hh
  - High-level objects of LHC inspiration
- TaggedParticle contains cross association between tracks and calo objects

Event Data Model Evolution: EDM4HEP

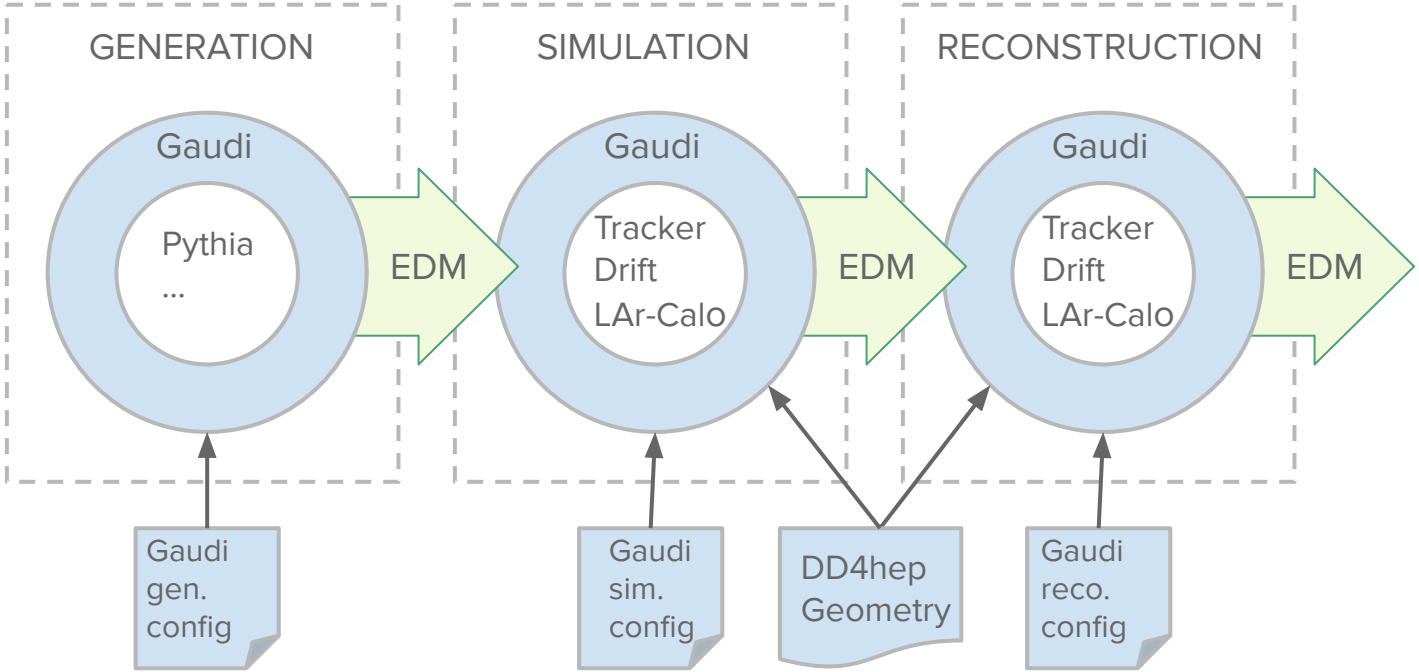
- EDM4HEP: attempt to have a common EDM for FCC, LC, CEPC, OpenData,...
- Task force meeting every 2-4 weeks since July 2019
  - Representatives from all future collider projects and LHC
- Started from comparing/merging FCC-EDM and LCIO (used by LC and CEPC)
  - Git repository available at EDM4hep
- Based on PODIO high-level EDM generator
  - As FCC-EDM
- Goal is to have a usable version by Q1/2020
  - Dedicated session at the Hong-Kong IAS mini-workshop next Friday

Software Framework

Software Framework: Gaudi-based

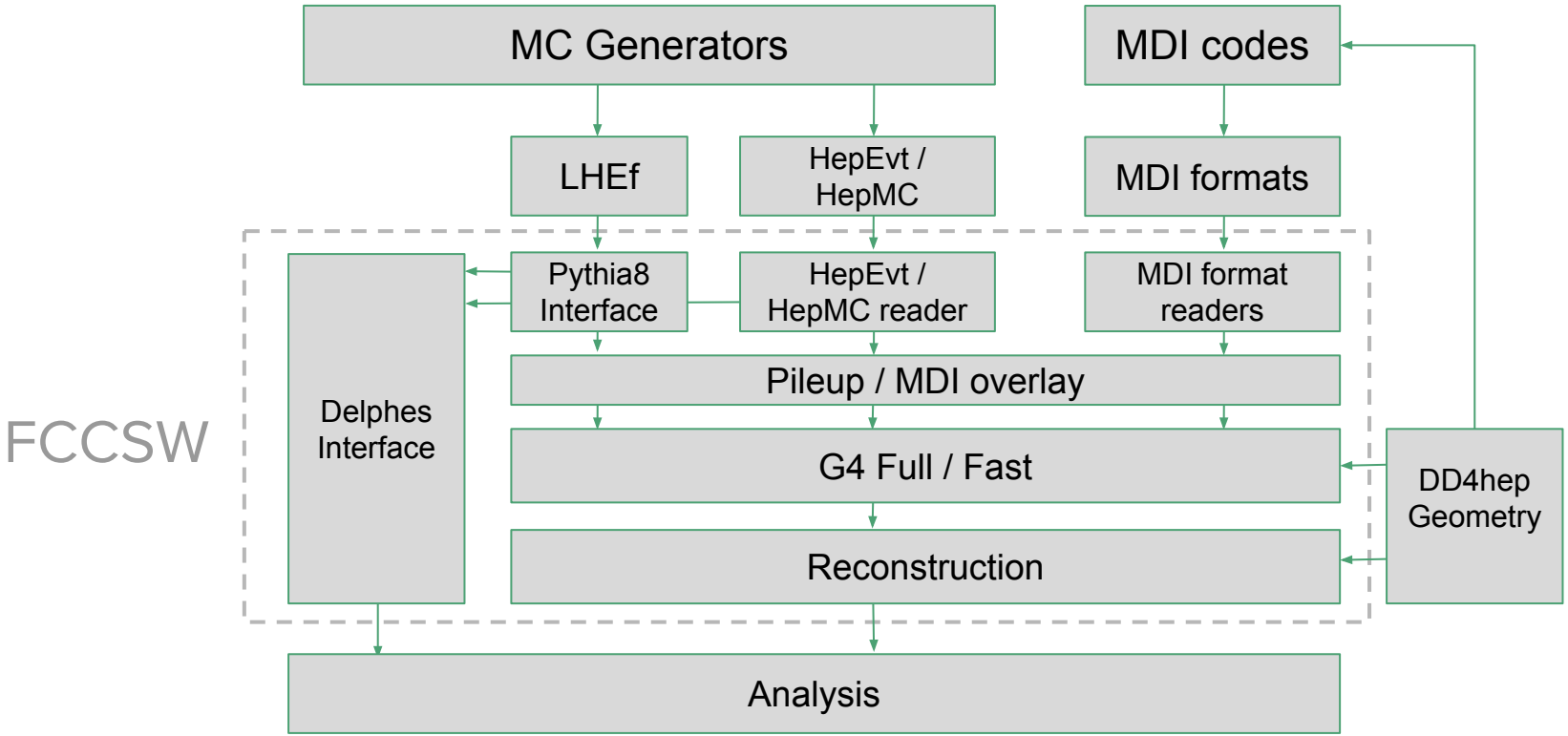
- Framework toolkit to provide required interfaces and services to build HEP experiment frameworks
  - Opensource project and experiment independent
- Data processing framework designed to manage experiment workflows
  - Separate data and algorithms; well defined interfaces
  - User's code encapsulated in Algorithm's, Tool's / Interface's, Service's
  - Different persistent and transient views of data
  - C++, with Python configuration
- Originating from LHCb, Gaudi is adopted also by ATLAS
  - Actively developed to face LHC Run 3 and Run 4 challenges (high PU)
- Using the latest Gaudi version (v32r2).

Gaudi and FCCSW



Typical workflows

Overview


MC Generators
MDI codes
LHEf
HepEvt / HepMC
MDI formats
FCCSW
Pythia8 Interface
HepEvt / HepMC reader
MDI format readers
Delphes Interface
Pileup / MDI overlay
G4 Full / Fast
DD4hep Geometry
Reconstruction
Analysis

Monte Carlo Generators

- MC Generators are an essential ingredient to understand the potential of a detector
  - Need to simulate precisely enough both signal and backgrounds
- Backgrounds
  - Unwanted collisions products / signals
  - Beam-related backgrounds (SR, …)
  - Beam unrelated backgrounds (cosmic rays, ...)
- Many programs exists to simulate the relevant processes
  - Session dedicated to generators for collision products this afternoon in 220/R-001

Monte Carlo Generators and FCCSW

- Generators repository: GenSer @ LCG software stacks
  - Generator Service hosted by EP-SFT

    Collaboration with the authors and with the LHC experiments to prepare validated code for communities at the LHC
  - Actively used by ATLAS, LHCb, SWAN and some SME experiments
  - Deployed via CernVM-FS
- MC generators are typically standalone codes
  - Noticeable exception is Pythia8, which provides a callable interface
- FCCSW interoperates MC generators mostly through common data formats
  - HepMC, LHEF
  - Pythia8 used to read LHEF files

MC Generators: status and areas of work

- GenSer generators palette biased towards LHC
  - Good for FCC-hh, incomplete for FCC-ee
- General purpose generators such as Pythia8, Whizard, MadGraph5 available
  - But we need to get experience on how to use them effectively for FCC-ee
- Integration of KKMC and BHLUMI in GenSer well advanced
  - Wrappers to produce HepMC and/or LHEF output required
- Similar work will be needed for MCSANC, BabaYaga, ...

Contributions welcome/required on interfacing and testing

Experience needed for interfacing: FORTRAN, FORTRAN&C++ interplay

For testing: ability/willingness to understand settings of a given generator

MDI Software Integration

- (Beam- and) MDI- related backgrounds are source of systematics
  - Need to be controlled as precisely as possible
  - Critical aspect, for performance, is detector occupancy
    - Possibly also radiation damage
- Non-exhaustive list of programs to calculate these backgrounds
  - MDISim: *Synchtron Radiation, Single beam induced backgrounds*
  - SYNC_BKG, SYNRAD+: *Synchtron Radiation*
  - GuineaPig++: *IP backgrounds, (In)coherent Pairs Creation, γγ to hadrons*
  - Pythia8: *γγ to hadrons*
  - BBBrem+SAD: *Radiative Bhabhas*
- Simplest and solid solution for integrating the MDI calculations in FCCSW is through shared data formats

MDI @ FCCSW Workflow

- MDI code provides sets of events with the 4-vectors and vertex of the relevant particles
  - γ's for SR; e+e- pairs, hadrons for IP processes, ...
  - May include the interaction in the beam-pipe (as in MDISim)
- Evaluation of detector occupancy in FCCSW
  - Through interaction of MDI particles in the detector
  - Through overlay of MDI events to "signal" events for a more detailed background simulation
  - This may also be done with a weighted mixture of MDI processes

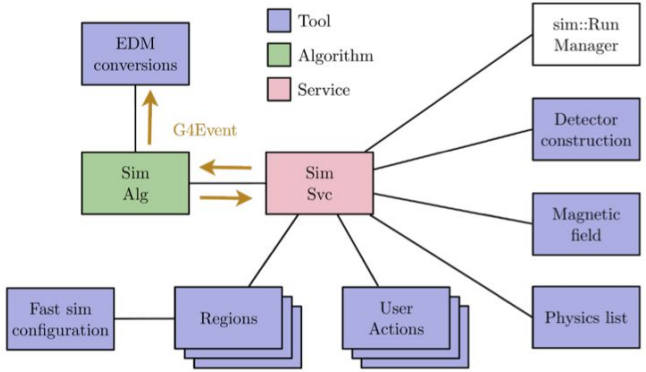
MDI integration: status and areas of work

- Agreement to define shared documented data formats
  - Work not started yet
- GuineaPig++ interfaced to iLCSoft
  - May be used as source of inspiration
- Event overlay in FCCSW
  - Same technology used for pileUP
  - Need to be tested and validated

Contributions required to all the above items
Experience required: file formats, FORTRAN, C++

Simulation

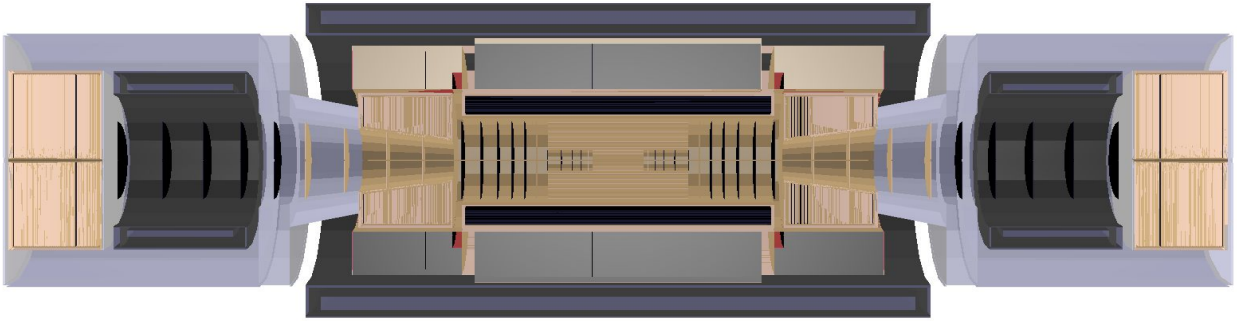
- Geant4
  - Gaudi components exists to create
    - User Actions
    - Regions
    - Sensitive detectors
    - Selective output options
  - Mixing fast and full G4 simulation possible
    - SimG4Full / SimG4Fast



- Delphes
  - Gaudi interface
    - FCC EDM output

FCC detector palette in DD4hep: FCC-hh

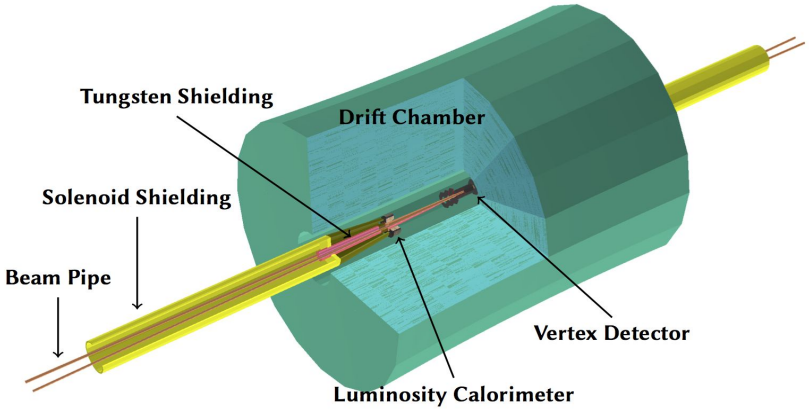
FCC-hh baseline



- Barrel, Endcap, Forward
- Beam Pipe, Shielding, Magnet solenoid
- Silicon Tracker
- LAr ECal, Tile HCal
- Muon System

FCC detector palette in DD4hep: FCC-ee

FCC-ee IDEA



- Beam Pipe, Beam instrumentation
- Lumical, HOM Absorber
- Vertex detector
- Drift Chamber
- Dual Readout Calorimeter
- Muon System

DR calo required support for optical properties in DD4hep available in the latest version shipped with FCCSW

FCC detector palette in DD4hep: FCC-ee

Possible alternatives for FCC-ee

- "IDEA" with reduced version of LAr ECal + Tile HCal
  - First DD4hep description available for testing
- CLD
  - Geometry description in DD4hep exists: https://github.com/iLCSoft/lcgeo
  - Requires some integration for usage in FCCSW

Contributions welcome/required on:
- DD4hep description for the IDEA DR calo, including digitization; muon system
- Enabling of CLD in FCCSW

Experience needed: familiarity/willingness to learn: DD4hep, detector geometry, Geant 4 simulation

FCC detector palette for Delphes

- Validated
  - FCC-hh baseline
  - HL-HELHC baseline
- Available for FCC-ee (need adjustment in the steering code)
  - IDEA
  - CLICDet

Possible contributions: testing, validation, fine tuning of existing cards; scripts or tool to easy variate relevant dimensions

Experience needed: familiarity with Delphes, Gaudi, simulation

Reconstruction

- Challenges: algorithm detector concept independent
  - Full flexibility, avoid duplication
- Tracking
  - Track seeding (Silicon tracker), Hough Transform (drift chambers)
  - Under development / investigation: ACTS integration, Conformal tracking
- Calorimeters
  - Sliding window (rectangular/ellipse), Topo-clustering
  - Under development / investigation: ML techniques

Possible contributions: vertexing, ACTS, ML, particle ID

Experience needed: familiarity with reconstruction algorithms, Gaudi, C++

Considerations for Physics Analyses

- HEPPY: High Energy Physics with PYthon
  - Modular python framework for the analysis of collision events
  - Developed and still used for CMS
- In FCCSW HEPPY is used to
  - Process EDM events, apply-preselection, produce a flat and light ROOT ntuple
  - Analyse the ROOT ntuple
    - Not the only code used for this purpose
- Flexible but slow
  - Plan to move to a C++-based analysis framework, e.g. RDataFrame

Possible contributions: develop HEPPY replacement based on RDataFrame or other fast technology

Experience needed: familiarity with ROOT and RDataFrame; advanced Python

Infrastructure, OS support, resources

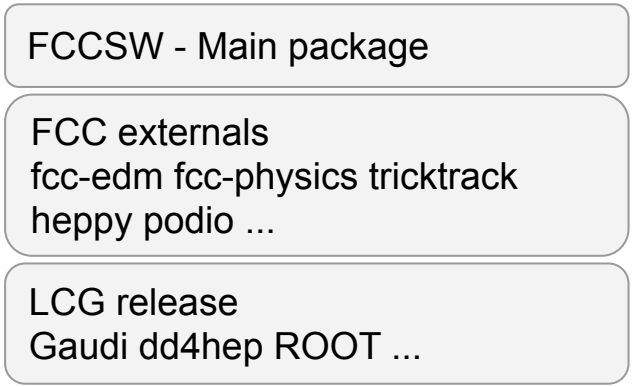
Software Infrastructure

- Typical HEP development workflow

- Deliverables
  - FCCSW
  - Externals: FCCSW specific dependencies
  - Based on LCG releases provided by EP-SFT

- Builds (nightlies, releases) managed by Spack package manager
  - Good feedback to HSF packaging WG

- Deployment on dedicated CernVM-FS repositories
  - /cvmfs/fcc.cern.ch/, /cvmfs/fcc-nightlies.cern.ch/



OS support

- Currently CERN-centric
- Support for the default version running on lxplus
  - CentOS 7, gcc 8
- VM, based on CernVM, available to recreate equivalent environment
  - Works from everywhere but speeds depends on the network
- Access via notebooks (SWAN) also possible
  - Being used for tutorials

Possible contributions: provide support for other OSs (Ubuntu, MacOsX, …)

Experience needed: familiarity with build systems, linux, ...

CERN resources and access policy

- CERN resources are available to member of institutes having signed the Memorandum of Understanding and its addendum
- EOS areas for data or large files: /eos/experiment/fcc
  - Current quota: 400 TB
  - E-group membership: fcc-eos-access (and alike)
  - Dedicated areas for ee, hh, eh, helhc, users
    - Plan to deprecate ‘users’: each CERN user has 1 TB at /eos/user/*u*/*username*
    - Needs to be enabled on Account Management page
- EOS areas for shared files: /eos/project/f/fccsw-web/www
  - Also accessible also via web
- Dedicated queue on LXBATCH
  - AccountingGroup = "group_u_FCC.local_gen" (on HTCondor)
  - E-group membership: fcc-experiments-comp

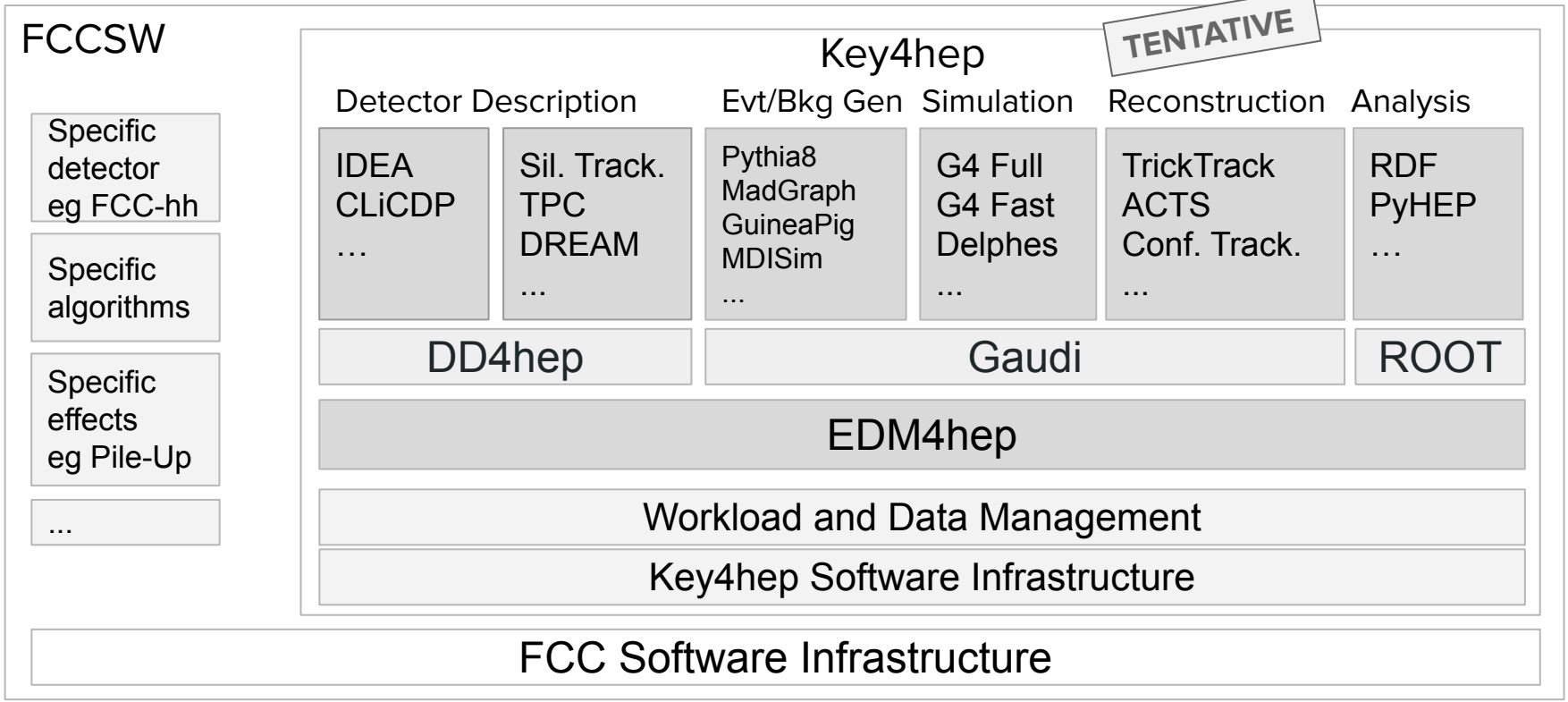
About Key4HEP

Key4HEP

- Depends crucially on EDM4HEP
  - Sets the chronological order
- Full-time fellow just started now (Jan 2020)
  - Another fellow working through CREMIN PLUS soon (Mar 2020)
- Possible contribution from EU projects
  - AIDA++, MSCA-ITN SPIRAL-NET

- Possibly rapid development when EDM4HEP is available
  - Key4HEP core ≈ FCCSW core + EDM4HEP
  - Algorithms (FCCSW, other) adapted to EDM4HEP can then be used
- Deliver early and often approach

Connection with Key4HEP



After-CDR for FCCSW

- Current approach seems adequate to fulfill the requirements
- Contribution from FCC institutes is essential and very welcome
  - Interested people should manifest themselves and pick-up an area of work
  - Public software coordination meetings will be resumed soon (see next)
- Foster activities such as Key4HEP, which formalize and extend FCCSW approach
  - Exchange / share effort within the community

Areas of work summary

- MC generators interfacing
- MC generators testing
- MDI shared formats
- GuineaPig++ integration
- Overlay of MDI/signal events
- IDEA DR Calo full simulation
- IDEA Muon system full sim
- Validation of LAr Ecal for FCC-ee
- Enabling of CLD in FCCSW
- Validation/testing of Delphes cards

- Vertex reconstruction
- ACTS integration
- ML for calo reconstruction
- e, mu, tau, c, b tagging / ID
- RDataFrame based analysis
- Porting to other OSs
- ...

Software Coordination Meetings

Open software coordination meeting will restart on 31 January 2020

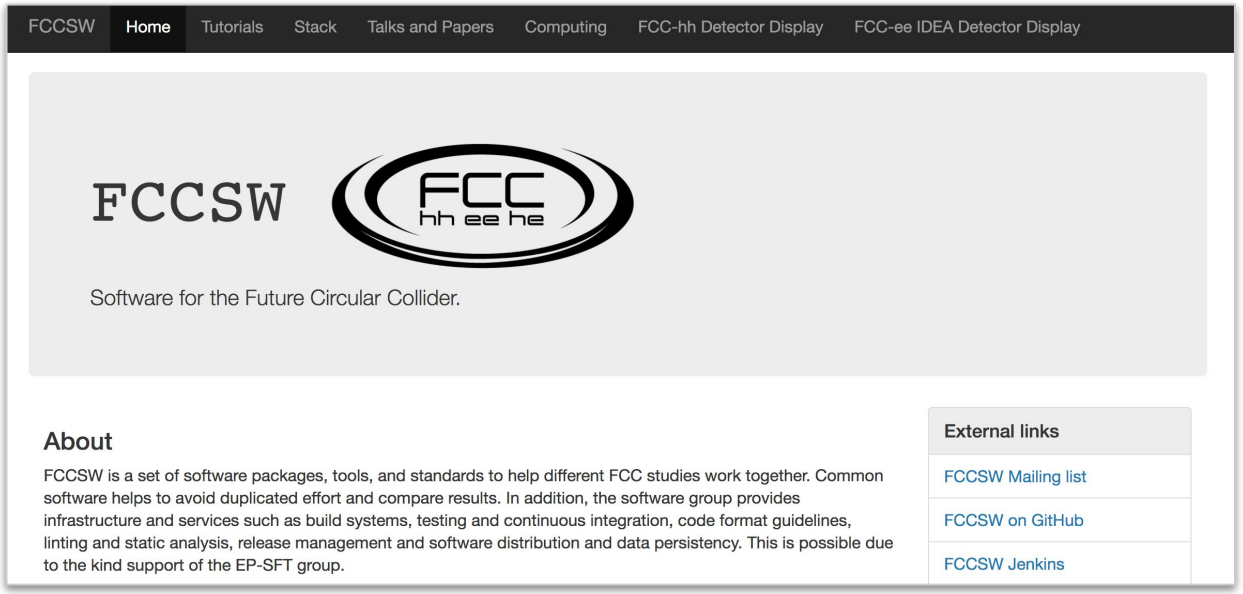
- Frequency: Bi-weekly meetings with remote connection
- Time slot: Friday morning 9h-11h
- Location: 40/R-B10
- Announced on fcc-experiment-sw-dev

Summary

- The FCC software stack has been assembled using as much as possible existing components
  - Served well the purposes of the CDRs
- Started a new phase of development to support more detailed studies in view of the TDRs, in particular for e^+e^-
- Areas of work identified
- Follow closely, participate and collaborate new common activities {Key4hep, EDM4hep}

Thank you!

- Web site https://cern.ch/fccsw



Hands-on tutorial tomorrow!

- Location: 4/3-006 (this room)
- Time slots: 9h-13h, 14h-17h
- Program: Roughly the same as in October SW Workshop

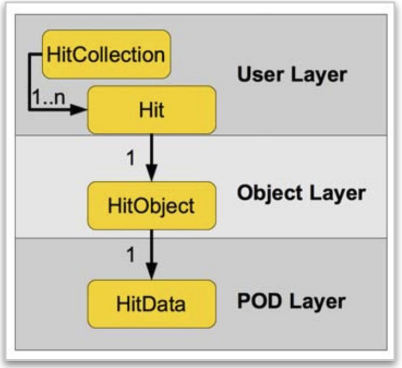
  1. Running FCCSW standalone
  2. Example of Physics Analysi (ZH)
  3. Delphes cards optimization
  4. Tracking with Drift Chamber

- Bring your own laptop!

Backup

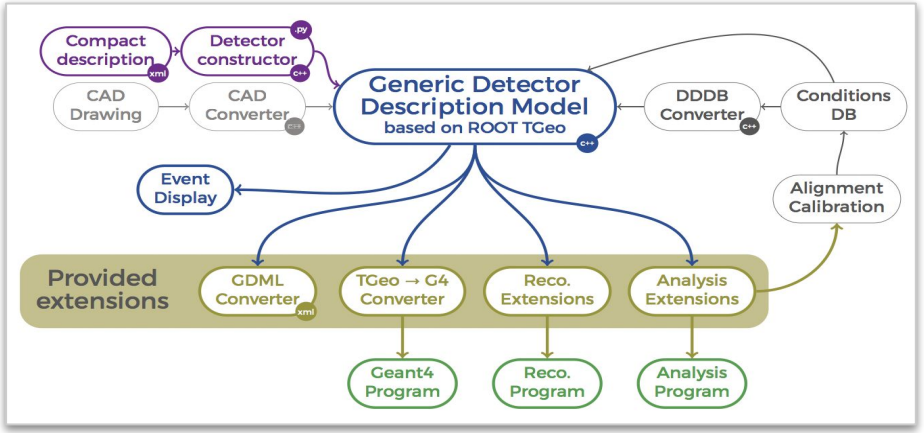
Event Data Model and PODIO

- LHC experiments / LC studies solutions suffering (partly) from
  - Overly complex data models with deep object-hierarchies
  - Unfavorable I/O performance
- **PODIO**: an EDM toolkit (AIDA2020 project)
  - Plain Old Data, automatic code generation, support for different backends
    - Keep memory model simple, enabling fast I/O and efficient vectorization
    - Consistent / homogeneous implementation, minimizes mistakes
  - High-level description in YAML format
  - Three-layers
    - User: handles objects and collections
    - Object: transient, relations between objects
    - POD: actual data structures
  - Follow-up for next AIDA being prepared
    - Schema evolution, memory and I/O optimizations



Detector Description: DD4hep

- Generic detector view appropriate to support
  - Simulation, reconstruction, analysis, …
- Design goals
  - Complete detector description
  - Single source of information
  - Support all stages of the experiment
  - Easy of use
- Part of AIDA2020
- Used by CLIC, ILC, FCC, LHCb, CMS, SCT

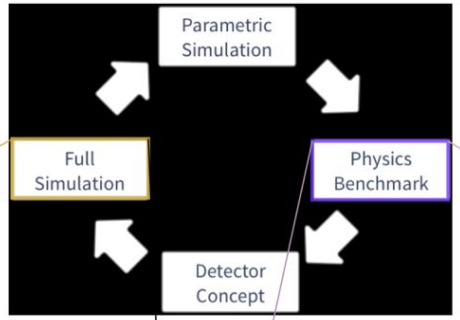


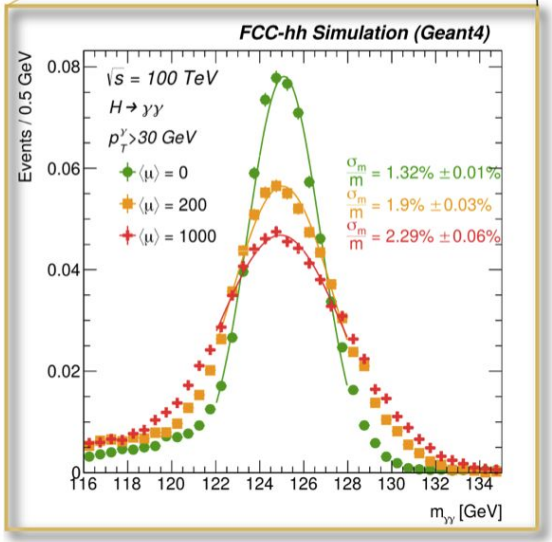
Fast / Full Simulation Interplay

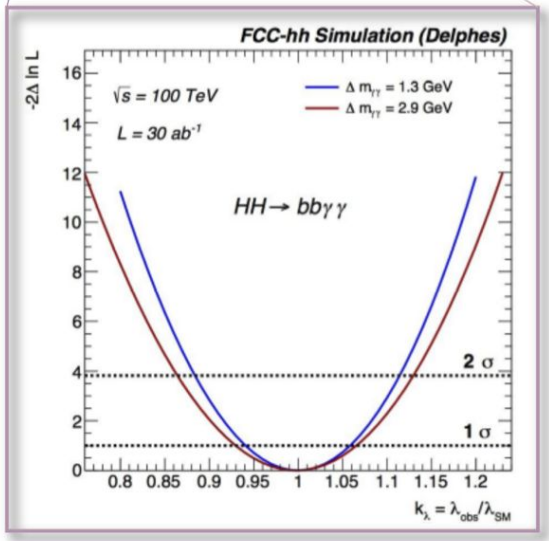
Example:

Higgs self-coupling

@ FCC-hh







Gaudi and FCCSW

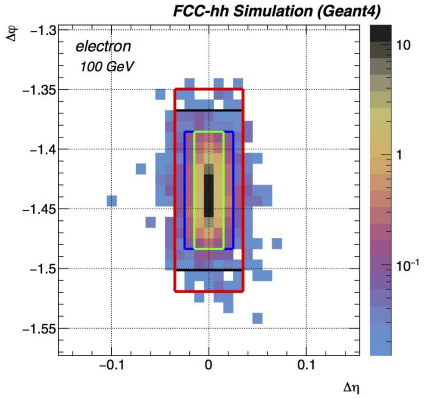
- Python scripts to configure tools and algorithms and define their processing order

```
$ fccrun \                                                  # Generic Gaudi app
      ./Examples/options/geant_fullsim_fccee_pgun.py \      # job definition
      --energyMin=10  --energyMax=10 --particleName="mu-" \ # parameters
      --n=1000 --filename=fccee_idea_mu.root
```

- Each algorithm defines its parameters which can be overwritten of the command line

Calo Reconstruction Example

- Single 100 GeV e- reconstructed by sliding window



- Single 100 GeV pion in 8+10 layers of the E+HCal reconstructed by topo-cluster