FCC Software Workshop

Introduction and Overview

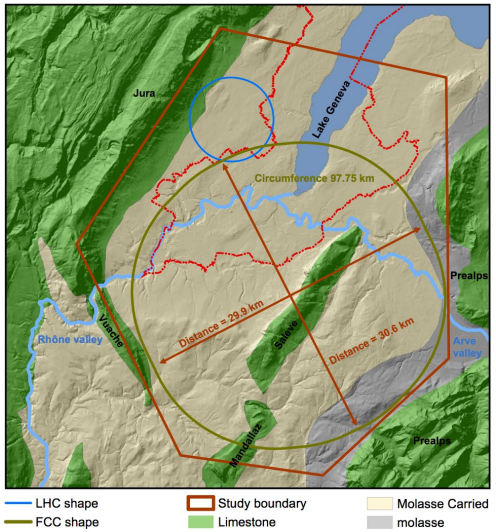
FCCSW Workshop October 2019

Oct 02, 2019

G Ganis, C Helsens for the FCC Software Group
CERN

The FCC@CERN project

- Circular Collider of 97.75 km of circumference
- Phase 1 : e+e-
  - {~91,~160,240,~360} GeV
  - Data taking period: 2039-2054
- Phase 2: pp, ep
  - ~100 TeV pp; possibly ep in parallel
  - Data taking period: 2065-2090
- Conceptual Design Report Dec 2018
  - Used as input for ESPP
  - Based on 150 TB of full simulated data and 100 TB of Delphes events



FCC Software

- Set of software packages, tools and standards at the service of FCC studies
- What exists is largely based on what has been used to prepare of the FCC CDR
- A new phase started to continue and expand to more detailed studies, in particular for the e^+e^- machine
- Closely follows the ongoing R&D activities about a Common Software Stack (Key4HEP)

Goals of the FCC Software Workshop / Tutorials

- Make participants able to use what exists
- Make interested participants able to contribute to the evolution / development
- Make participant to provide feedback on
    - The format
    - The topics
    - Expectations

Structure of the event

- Day 1 (today): talks
  - Overview
  - Specific components
    - Framework, Geometry, Reconstruction, Generators, Analysis
- Day 2 (tomorrow) Hands-on tutorials
  - Full analysis chain example (ttbar)
  - ee->HZ analysis
  - Calorimeter example
  - Drift Chamber
  - AoB: could possibly adapt to people wishes and help them getting started

Word of Caution

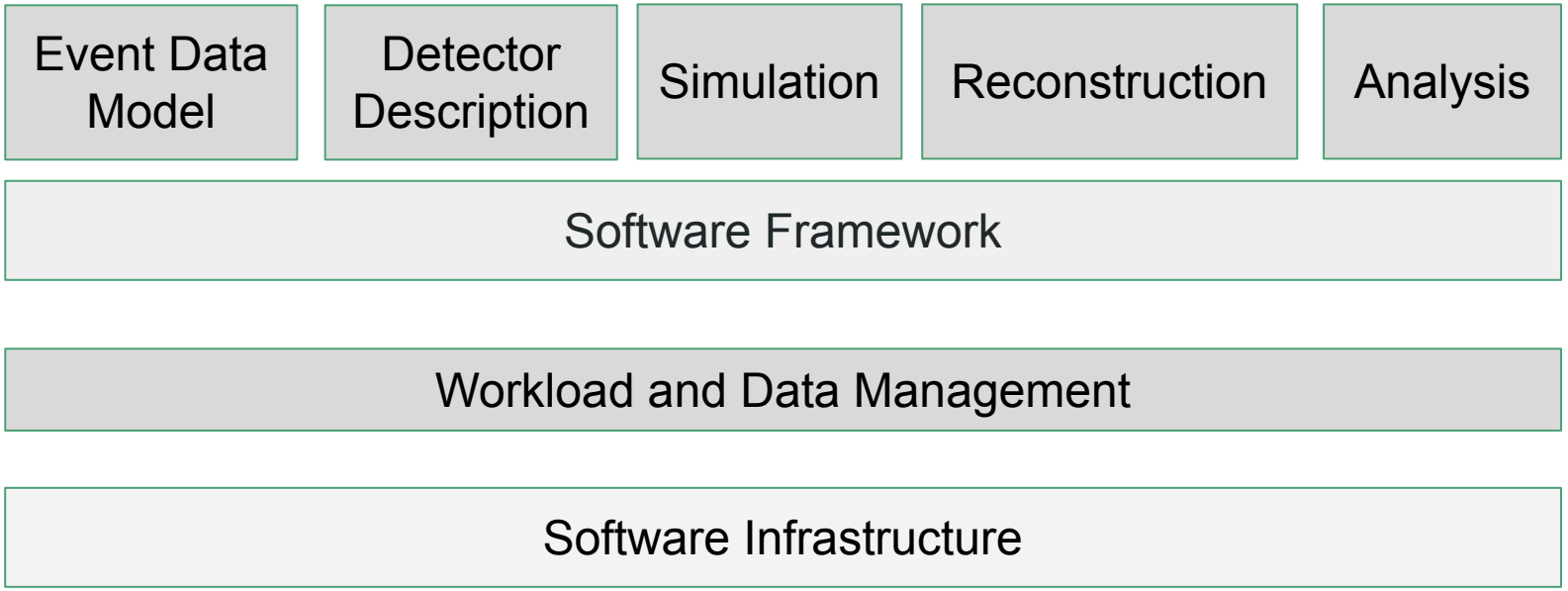
- Pilot tutorial session
  - Plan to have regular editions (next during January Physics Workshop)
  - Your feedback highly appreciated
    - Questionnaire circulated after the workshop
- FCC software CERN centric
  - Intensive use requires a CERN account
  - Temporary access for the workshop, not a long term solution

Overview of the existing FCC Software components

Basic Software Requirements

- Conceptual / Technical Detector Design studies
  - Flexibility
  - Ideal and detailed detector description
  - Open to evolution
- Broad range of event complexity
  - e+e- vs pp vs e-p
- Need to support physics and detector studies
  - Parametrised, fast and full simulation (and mixture of the 3)
- Aim to de-duplicate efforts
  - One software stack to support all the cases, all the detector concepts, and future (proto-)collaborations

Ingredients



Software Framework

- A software framework provides
  - Vocabulary (‘algorithm’, ‘tool’, ‘service’, ...)
  - Well defined, stable interfaces
  - Plug-in based, evolving implementations
  - Homogeneous configuration, logging, error reporting
- GAUDI: data processing framework
  - Designed to managed generic experiment workflows
    - Separate data and algorithms; well defined interfaces
    - User’s code encapsulated in Algorithm’s and Tool’s
    - Different persistent and transient views of data
  - Originating from LHCb, adopted by ATLAS, Daya Bay, LZ, ...
  - Actively developed to face LHC Run 3 and Run 4 challenges (high PU)

GAUDI and FCCSW

Talk by J. Faltova

- Currently integrates
  - Generation
  - Parametrized/Fast/Full simulation
  - Reconstruction
- Generation
  - Pythia8 as main general purpose generator (other Gen studied)
    - Hadronization and/or LHE reader
    - SingleGun shooter
  - HepMC, HepEVT readers
  - Pileup overlay technology
  - All generators available in LCG_releases can be used

Event Data Model

- Heart of any HEP experiment software
- Complete description of observables produced by detectors
- Basis for derived quantities described in analysis
- Follows needs and phases of the experiment
- Each experiment had its own EDM, though communalities, inside HEP, are sometimes evident

EDM4HEP

- Ongoing project to define an EDM to serve all HEP experiments
  - FCC, ILC, CLIC, CEPC
  - LHC OpenData, Data preservation
- Tightly connected to Key4HEP

Current FCC Event Data Model

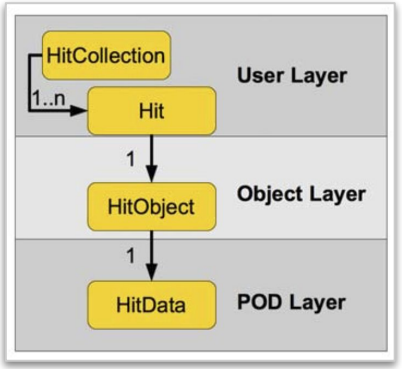
- *Components*
  - Point, LorentzVectors, BareHit, BareCluster, BareParticle, BareJet
- *Particles*
  - MCParticle, ParticleMCParticleAssociation, Particle, TaggedParticle
- *Vertices*
  - Vertex, GenVertex
- *Track Detector Types*
  - TrackHit, PositionedTrackHit, DigiTrackHitAssociation, TrackCluster, TrackState, Track, WeightedTrack
- *Calorimeter Types*
  - CaloHit, CaloCluster, CaloHitMCParticleAssociation, CaloHitAssociation

Current FCC Event Data Model (2)

- *High Level Reco objects*
  - MET, Jet, TaggedJet, ResolvedJet, GenJet, TaggedGenJet, ResolvedGenJet
- *Event / Run Types*
  - EventInfo

PODIO: an EDM toolkit

- Plain Old Data
  - Keep memory model simple, enabling fast I/O and efficient vectorization
- Automatic code generation
  - Consistent and homogeneous implementation, minimizes mistakes
- Support for different backends
- Input specified in YAML format
- Three-layers
  - User: handles objects and collections
  - Object: transient, relations between objects
  - POD: actual data structures



- Developed in the context of AIDA 2020 (git repo)

FCC EDM with PODIO

- Dedicated repository: https://github.com/HEP-FCC/fcc-edm
- YAML: https://raw.githubusercontent.com/HEP-FCC/fcc-edm/master/edm.yaml

```
fcc::Track:
      Description: "Track reconstructed from tracker clusters "
      Author : "C. Bernet, B. Hegner"
Members:
      - float chi2 // chi2 returned by the track fit
      - unsigned ndf // Number of deg of freedom of fit
      - unsigned bits // Stores flags
OneToManyRelations:
      - fcc::PositionedTrackHit hits // Hits used
      - fcc::TrackState states // States along the track
```

FCC EDM at work

- PODIO data service interface at $FCCVIEW/include/podio

```
lxplus $ source /cvmfs/fcc.cern.ch/sw/latest/setup.sh
lxplus $ ls $FCCVIEW/include/podio
ASCIIWriter.h CollectionIDTable.h  ICollectionProvider.h  ObjBase.h   PythonEventStore.h  ROOTWriter.h
CollectionBase.h  EventStore.h  IReader.h  ObjectID.h  ROOTReader.h podioVersion.h
```

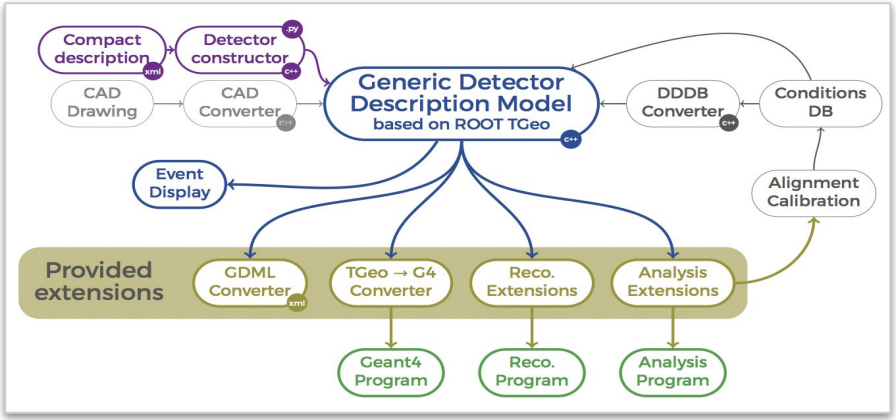
- EDM interface at $FCCVIEW/include/datamodel

```
lxplus $ ls $FCCVIEW/include/datamodel/Particle*
Particle.h ParticleConst.h  ParticleMCParticleAssociation.h  ParticleMCParticleAssociationConst.h
ParticleMCParticleAssociationObj.h ParticleCollection.h  ParticleData.h
ParticleMCParticleAssociationCollection.h  ParticleMCParticleAssociationData.h ParticleObj.h
```

Detector Description

Talk by M Petric

- DD4HEP: generic detector view appropriate to support
  - simulation, reconstruction, analysis, …
- Design goals
  - Complete
  - Single source of information
  - Easy of use

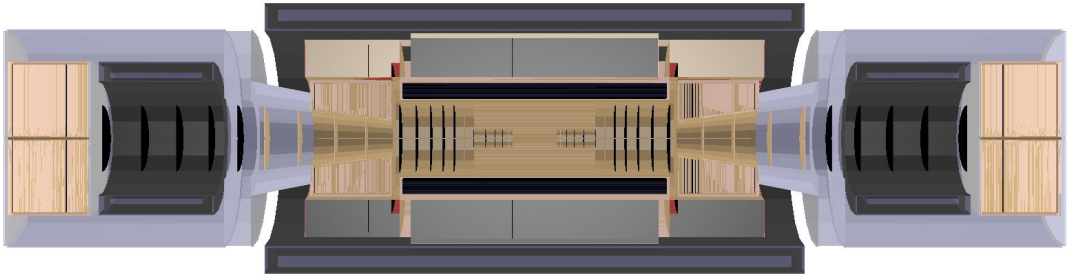


- Part of AIDA2020
- Used by CLIC, ILC, FCC, LHCb, CMS, SCT

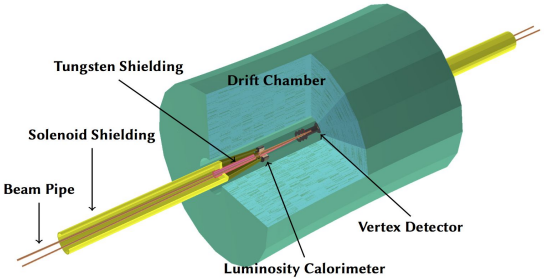
DD4HEP and FCCSW

Talk by V. Volkl

- FCC-hh
  - Complete for reference detector

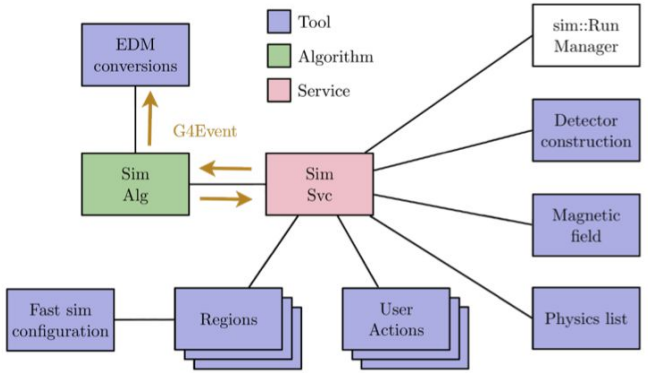


- FCC-ee
  - IDEA Concept
    - Beam Pipe, instrumentation
    - Vertex Detector, Drift Chamber
    - DREAM Calorimeter (under dev)
  - LAr+Tile calorimeter (under dev)



Simulation

- GAUDI components exists to create
  - User Actions
  - Regions
  - Sensitive detectors
  - saveOutput



- Mixing fast and full G4 simulation possible
  - SimG4Full / SimG4Fast

Parametric Simulation Tool : Delphes

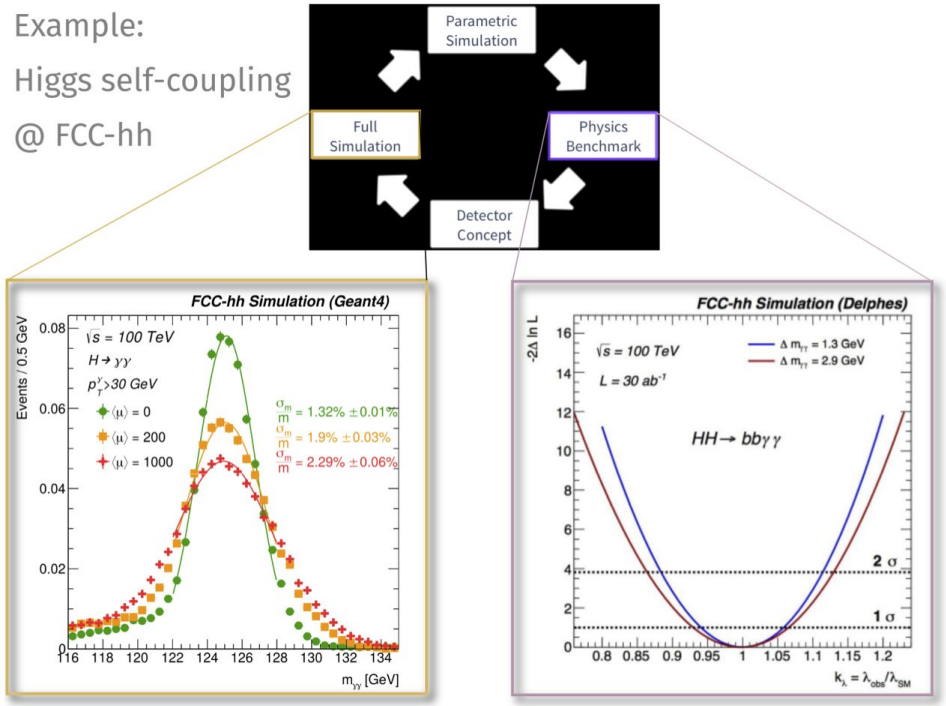
- Very fast simulation for every detector
- Widely used for phenomenology
    - Also for CMS phase 2, LHCb, ...
- ID, particle/energy flow, pile-up, jet sub-structure, ...
- EDM Output in ROOT format
- Important tool for physics analysis, but need to”tune” the parametrisation to the detector response in full simulation
- In FCCSW
    - Sim/SimDelphesInterface

Fast / Full Simulation Interplay

Example:

Higgs self-coupling

@ FCC-hh



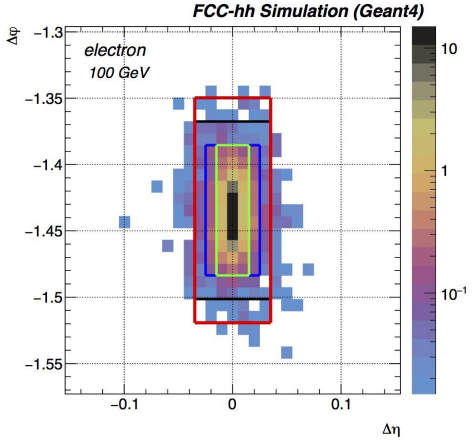
Reconstruction

Talk by V. Volkl

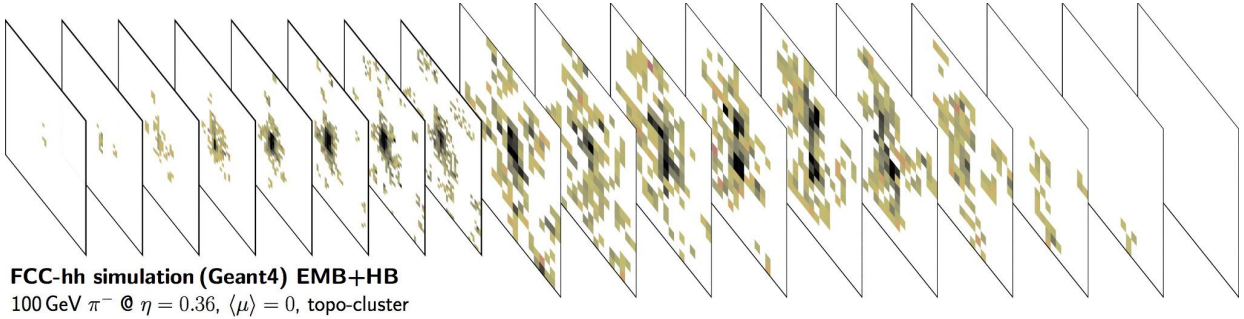
- Challenges: algorithm detector concept independent
  - Fully flexibility, avoid duplication
- Tracking
  - Track seeding (TrickTrack)
  - Hough Transform for drift chambers
  - Under implementation
    - ACTS integration
    - Conformal tracking
- Calorimeters
  - Sliding window (rectangular/ellipse)
  - Topo-clustering

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



Analysis,
Workload and Data Management,
Software infrastructure

Analysis Workflow typical example

Talk by C. Helsens

- LHE producer
  - E.g., for FCC-hh, from GridPacks produced using MadGraph
  - E.g., for FCC-ee, Whizard with output format ‘lhef’
- FCCSW
  - Run Pythia8 for parton shower + hadronization (also as generator)
  - Run Delphes parametrization (output EDM in ROOT format)
- HEPPY
  - Analysis Pre-Selection (output light flat ROOT Trees)
- Analysis Final Selection
  - ROOT, PyROOT, HEPPY
- Analysis Final Results
  - Limits, cross-sections, couplings, ...

HEPPY

- High Energy Physics with PYthon: modular python framework for the analysis of collision events
- Developed and still used for CMS
- In FCC 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 a C++-based analysis framework, e.g. RDataFrame
    - At least for the most time-consuming parts

Workload and Data Management

- Homemade production system
  - Developed and used for the CDR data samples
  - Bookkeeping / Monitoring of the events produced on web pages
- Interfaced to CERN batch system
  - FCC dedicated CPU and storage (EOS) resources
  - Access controlled via MoU signature
- Plan to adopt a production option (DIRAC, RUCIO, etc.) in the medium term
  - Including distributed computing (not needed so far, though VO exists)

Software Infrastructure

- Code repositories on GitHub, project HEP-FCC
  https://github.com/HEP-FCC/FCCSW, ...
- Development workflow
  - PR reviews, nightlies, Continuous Integration, Coding guidelines, Automatic code formatting, Static code analysis
- Builds (nightlies, releases) managed by Spack (HSF)
  - Actually run on EP-SFT resources

Software Infrastructure (2)

- Deliverables
  - FCCSW
  - Externals: FCCSW specific dependencies
  - Based on LCG releases

FCCSW - Main package

FCC externals
fcc-edm fcc-physics tricktrack
heppy podio ...

LCG release
Gaudi dd4hep ROOT Pythia8 ...

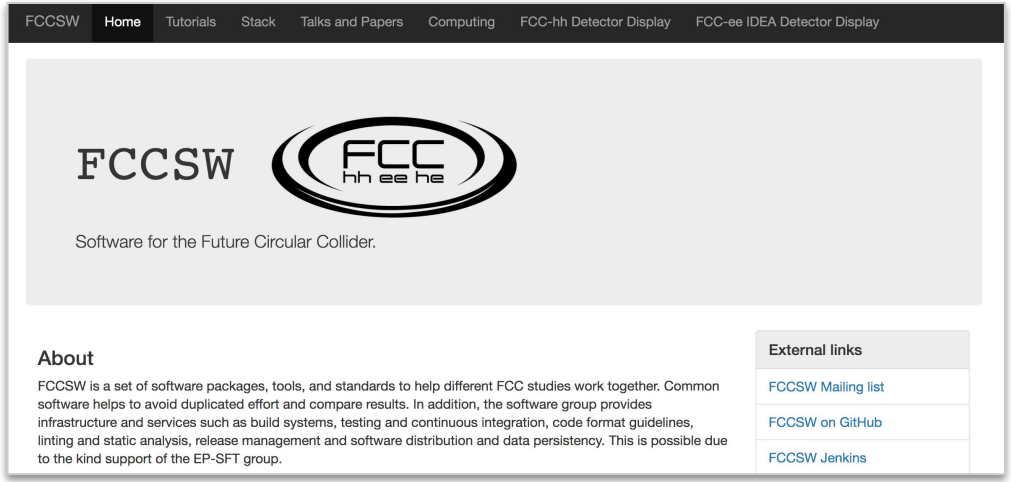
- Deployment on dedicated CernVM-FS repositories

  /cvmfs/fcc.cern.ch, /cvmfs/fcc-nightlies.cern.ch
- On lxplus

  $ source /cvmfs/fcc.cern.ch/sw/latest/setup.sh

Documentation

- http://hep-fcc.github.io/FCCSW/



Future Plans

Address missing / lacking / weak parts

- FCC-ee detector description, fast/full simulation
- Reconstruction tools
- Beam-related backgrounds
- Coherent MDI
- Complete palette of physics generators
- Adopt new solutions for analysis (RDataFrame, ...)
- Workload and Data management
- ...

Follow up on Key4HEP

- Contribute to the definition of EDM4HEP
  - Adopt EDM4HEP
- Contribute feedback, proposals .. on existing components
- Increase integration / usage of common components
- Ideally, rebase FCCSW on top of Key4HEP

Summary

- The FCC software stack assembled using as much as possible existing components
- Served well the purposes of the CDRs
- Started a new phase to further develop to support more detailed studies, in particular for e+e-
- Follow closely, participate and collaborate new common activities within HSF
    - Key4HEP, EDM4HEP

Workshop Practicalities

- All sessions in room 593/R-010 Salle 11
- VidyoConference available (see Indico page), however not ideal for hands-on
  - Priority will be given to the room
  - Also possible to use dedicated Mattermost channel to discuss
- Computing resources
  - FCC Software is CERN centric: CERN account required
    - Temporary credentials for those who do not have (locally only)
  - Exercises and demos prepared for lxplus (CentOS7)
  - Laptops connected at CERN can use a Specialized CernVM Virtual Machine
  - We can provide access to temporary resources if needed
  - Tutorials can also be run on SWAN within a web browser
  - More details during hands-on tomorrow
- Dinner tonight at Piatto D'Oro (see next slide) at 19h30

Workshop Dinner at Piatto d’Oro

- Reserved for 19h30
- People who have not signed up can still do it until the coffee break (by mail or voice).
- Tram 18 stop ‘Bois-du-Lan’ or ‘Jardin Alpin-Vivarium’