FCC software: Status and Plans

Clement Helsens and Gerardo Ganis, CERN-EP
FCC week 2019, Brussels
Basic Software Requirements

• Conceptual detector design studies
  • Flexibility
  • Ideal detector descriptions
  • Open to evolution

• Broad range of event complexity
  • $e^+e^- \text{ vs } pp \text{ vs } e^-p$

• 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
Basic Software Requirements

We have achieved most of this with the publication of the FCC CDRs
Ingredients

Event Data Model  Detector Description  Simulation  Reconstruction  Analysis

Software Framework

Workload and Data Management

Software Infrastructure
Ingredients

- PODIO
- DD4HEP
- tkLayout
- Geant4
- Delphes
- TrickTrack
- ACTS
- HEPPY
- GAUDI
- HTCondor
- Git
- ProjectTemplate
- CernVM-FS
- Docker
- SWAN

Software Framework

Event Data Model

Detector Description

Simulation

Reconstruction

Analysis

Workload and Data Management

Software Infrastructure
Outline

• Status

• Plans
Event Data Model

- Solutions adopted by LHC experiments and for LC studies proved to work but partly suffered from:
  - Overly complex data models with deep object-hierarchies
  - Unfavorable I/O performance
- **PODIO**: EDM toolkit
  - As much as possible based on Plain Old Data
    - Keeps memory model simple
    - Enables fast I/O operations and efficient vectorization
  - Automatic code generation
    - Consistent and homogeneous implementation, minimizes mistakes
- Support different backend
- Developed in the context of AIDA2020 EU programme
Detector description

- **DD4Hep generic detector description:**
  - Support: Simulation, Reconstruction, Analysis

- **Design goals:**
  - Complete detector description
  - Supports all stages of the experiment
  - Single source of information
  - Easy to use

- Part of AIDA2020 EU programme
- Used by CLIC, CMS, FCC, ILC, LHCb
DD4Hep @ FCC

- FCC-hh
  - complete

- FCC-ee
  - IDEA concept
    - Beam pipe, instrumentation
    - Vertex detector, drift chamber
    - DREAM calorimeter (under dev)
  - LAr+Tile calo. concept (under dev)
Software Framework

• **Provides:**
  • Vocabulary: Algorithm, Tool, Service, etc...
  • Well defined, stable interfaces
  • Plug-in based, evolving implementations
  • Homogeneous configuration, logging, error reporting

• **GAUDI: data processing framework**
  • Designed to manage generic experiment workflows
  • Main concepts:
    • Separate data and algorithm; well defined interfaces
    • User code encapsulated in “Algorithms” and “Tools”
    • Different persistent and transient views of data
  • Originating from LHCb; also used by ATLAS, Daya Bay, LZ
    • Support for concurrent event processing, reduce memory footprint
GAUDI @ FCCSW

• **Currently integrates:**
  • Generation
  • Fast/Full Simulation
  • Parameterized simulation with Delphes and PAPAS
  • Track seeding
  • Calorimeter reconstruction

• **Generation:**
  • Pileup overlay
  • HepMC, HepEVT readers
  • Pythia8 as main generator, or for hadronisation of LHE/HepMC
    • All generators available in LCG_releases can in principle be used

Identical outputs, makes it easy to compare
Fast/Full simulation interplay

Example:
Higgs self-coupling
@ FCC-hh
Reconstruction

• **Challenges**
  • Reconstruction Algorithm as much as possible independent of the detector concept: full flexibility and avoid duplication

• **Tracking**
  • Track seeding (TrickTrack)
  • Hough transform for drift chambers
  • Conformal tracking (being implemented)
  • ACTS integrated in FCCS

• **Calorimeters**
  • Sliding window (rectangle/ellipse), also versus depth
  • 3D Topo-clustering
Analysis

• Order of 20-30 analyses carried out for the CDRs within the same analysis code in python
• Based on 100TB of Delphes events produced with an home made production system (used to produce and validate several billions of events)
• Heppy used to process the FCC EDM events and produce flat and light ROOT ntuples
• Other python codes to run on the flat ntuples
• Flexible but relatively slow
  • New analyses are being developed in DataFrames (orders of magnitude faster)
Software infrastructure

- **Code repository**
  - On GitHub: https://github.com/HEP-FCC
  - Development workflow: PR reviews, nightlies, Continous Integration
  - Coding guidelines, Automatic code formatting, static code analysis

- **Deliverables**
  - FCCSW: FCC software framework
  - Externals: FCCSW dependencies
  - Based on LCG releases

- **Builds managed by Spack (HSF)**
  - Configuration, build and installation
  - Actually run on LCG SFT resources

- **Deployment: dedicated CVMFS repository**
  - source /cvmfs/fcc.cern.ch/sw/views/releases/externals/94.2.0/x86_64-centos7-gcc62-opt/setup.sh
Documentation

- Users Forum: [https://fccsw-forum.web.cern.ch/](https://fccsw-forum.web.cern.ch/)
Outline

• Status
• Plans
Mandate for the after CDR FCCSW (in view of ESPP 2026)

- Support more detailed studies, in particular for FCC-ee, focusing on:
  - Completeness
    - State-of-the-art generators, beam/detector simulation, reconstruction/analysis algorithms,…
  - Flexible detector description
    - Easy switch/replace sub-detectors, change dimensions/layout
  - Low usability thresholds and fast/easy learning curve
  - Adequate computing support and CPU/storage resources
  - Extensive documentation and regular training

- Ensure that SW is a part-and-parcels of the Turnkey Software Stack Key4HEP

- Foster development and use in physics studies, detector optimization, machine-detector interface

- Support substantial participation from FCC institutes worldwide
Future plans

• **Address missing/lacking/weak parts**
  • Reconstruction tools
  • FCC-ee detector description, fast/full simulation
  • Simulation of beam related effects (ongoing with GUINEAPIG)
  • Workload & Data management
  • Analysis
  • ...

• **Foster the Turnkey Software Stack (Key4HEP)**
  • Contribute with feedback and proposals to the development of already used components
  • Increase/usage integration of common components
    • DIRAC, RUCIO
    • ROOT new solutions
    • ...

FCC Software: Status and Plans
26/06/19
Future plans

• **Follow/participate AIDA++ software projects**
  • After meeting last week, few proposals have been agreed upon

• **Follow relevant HSF working group activities**
  • Simulation, Reconstruction, Analysis

• **Main items for next year**
  • Consolidate physics generators interface
  • Allow for an easy introduction of new detector concepts described in DD4HEP
  • Interface-to / Synergies-with Key4HEP
  • Develop analysis workflow based on RDataFrame
  • Define a well established platform for detector performance comparison
  • Enable DIRAC / Rucio ...

• **FCC-ee Software Tutorial hands on session during next FCC-ee WS Sep 9-12**
Preliminary agenda of the WS

• Half a day of parallel session
  • Intro and Status of Things
  • Key4HEP
  • Status/Plans of IDEA detector implementation
  • Status/Plans of Physics generators and their availability in FCCSW
  • Status/plans of MDI simulation and their availability in FCCSW
  • Status and plans of physics analyses framework

• Half day hands-on tutorial session
  • IDEA detector concept in Delphes
  • Full simulation (at least) of the vertex
  • Example analysis in HEPPY and RDataFrame
Outlook

• The FCCSW stack has been assembled using as much as possible existing components
• Served well the purposes of the CDRs
• Next phase
  • Further developments to support more detailed studies, in particular for FCC-ee
  • Follow closely, participate and collaborate
    • Turnkey (Key4Hep) software Stack R&D
    • HSF activities
    • Other new common activities