FCC Software Framework, vision for the future

Javier Cervantes, Pere Mato and Graeme Stewart
EP-SFT

11th FCC-ee workshop
10 January 2018
Introduction

- Support experiments for all collider options: ee, eh & hh
- Support physics and multiple detector studies
- Collaborative approach:
  - Extract and adapt from the LHC experiments if possible
  - Invest into new solutions where necessary
  - Benefit from AIDA and HSF efforts
- One software stack:
  Support all experiments from event generation to physics analysis
Introduction

- Support experiments for all collider options: ee, eh & hh
- Support physics and multiple detector studies
- Collaborative approach:
  - Extract and adapt from the LHC experiments if possible
  - Invest into new solutions where necessary
  - Benefit from AIDA and HSF efforts
- One software stack:
  Support all experiments from *event generation* to *physics analysis*

Design changes can be evaluated for physics impact
Introduction

- Support experiments for all collider options: ee, eh & hh
- Support physics and multiple detector studies
- Collaborative approach:
  - Extract and adapt from the LHC experiments if possible
  - Invest into new solutions where necessary
  - Benefit from AIDA and HSF efforts
- One software stack:
  Support all experiments from event generation to physics analysis
SFT Group Mandate

- The group **develops and maintains common scientific software** for the physics experiments in close collaboration with the EP experimental groups, the IT department and collaborating HEP institutes
  - Geant4, ROOT, Gaudi, CernVM, ...
- The group provides **common infrastructure and expertise** to the experiments
  - select & maintain tools used in the development process
  - manage stack of >300 external software packages
  - provides people to the experiments to fill key roles
  - training (CSC, GridKa, CERN Technical Training, …)
- The group leads and participates actively to **community initiatives** such as [HEP Software Foundation](https://www.hep-software-foundation.org) (HSF)
Why SFT is relevant here?

- SFT is a software support group in the EP department with **long-term commitment**
  - Originally created to support the LHC experiments
  - Can ensure long-term **sustainability** and **innovation** for software for the experiments

- Common ‘turnkey’ software stack
  - We aim to maintain a **common stack** that can serve several experiments with minimal effort
  - Components could be: Gaudi, DD4hep, PODIO, Acts, ... in addition to ROOT, Geant4

- EP R&D initiatives
  - The ‘turnkey’ software is one of the proposed R&D projects
Main lines of development in SFT

- **Simulation**
  - More accurate physics models and extended range of applicability
  - Faster simulation: vector prototype (GeantV), fast simulation techniques (ML), etc.

- **ROOT**
  - Declarative and parallel new way of writing data analyses (RDataFrame)
  - New web-based graphics (re-using libraries and standards widely adopted in industry)
  - Better interoperability (Python, ML tools, etc.)
  - Modularization and modernization of interfaces

- **Web-based Analysis (SWAN)**
  - Portal to dedicated computational resources (clusters, GPU farms,...)

- **CernVM**
  - Faster software publication and delivery

- **LCG software provision**
  - New tools to streamline releases and CI
Interactions with SFT

● Several regular meetings exists
  ○ Architects Forum (AF): SFT and main stakeholders of its deliverables
  ○ Librarian and Integration Meeting (LIM): goals and requirements of the LCG Software stack
  ○ Geant4 Technical Forum: experiments and core Geant4 developers
  ○ ROOT Planning: experiments and core developers

● Users’ Workshops
  ○ CernVM
  ○ ROOT
  ○ Soon SWAN

● Long-term community driven evolution of software
  ○ HSF working groups
Benefits of the FCC & SFT Collaboration

● Input to the common development projects (ROOT, Geant4,...)
  ○ The needs of FCC are very important for the evolution of these codes

● Definition of common turnkey software stack
  ○ FCC would be a main client of the new system

● LCG common software stack
  ○ Same set of tools, standards and procedures as some other LHC experiments
  ○ Speed up FCC-specific software builds
  ○ New developments can be integrated from/by existing projects

● Continuously evolving software stack
  ○ Version maintenance
  ○ Based on user needs and current state of the art
  ○ New packages constantly added
Benefits of the FCC & SFT Collaboration

- Reduce complexity
- Speed up FCC-specific builds

Full graph of FCC software dependencies
Benefits of the FCC & SFT Collaboration

- Reduce complexity
- Speed up FCC-specific builds

Reduction 94%

Reduced graph of FCC software dependencies

- Taken from LCG
- Built by FCC
FCC Software components and collaborations

- Event Data Model - PODIO
- Detector Description - DD4HEP
- Simulation - Geant4, PAPAS
- Reconstruction - Acts, TrickTrack
- Physics Analysis - HEPPY

Strong candidates to become part of the **Turnkey Software Stack**

More details about the current workflow adopted by FCC on [Joschka’s talk](#)
FCC Build infrastructure

- Two main deliverables:
  - **FCCSW**: FCC software, framework common to FCC-hh, -ee, and -eh
  - **Externals**: FCC-specific software dependencies

- Computing resources
  - Shared with LCG infrastructure
  - CERN Openstack virtual machines + LCG Physical nodes
  - CVMFS as main software repository for distribution

- Build services based on **Spack**
  - Common tool with LLNL and FNL, **supported by HSF**
  - Package manager tool: reduce complexity dealing with multiple versiones, compilers, platforms
  - Manages **configuration, build** and **installation** steps
  - Installs new packages reusing LCG installations

---

**FCCSW - Main package**

**FCC Externals**
- fcc-edm
- papas
- podio
- fcc-physics
- acts-core
- gaudi
- tricktrack
- heppy

**LCG Releases - Common experiment software**

source /cvmfs/fcc.cern.ch/sw/views/releases/externals/94.0.0/x86_64-slc6-gcc62-opt/setup.sh

---

Setup up the FCC environment
FCC Interactive analysis with **SWAN**

Provide at CERN “Data Analysis as a Service”

**Interface chosen: Jupyter notebooks:**
- Analysis **only with a web browser**
- **Easy sharing** of scientific results: plots, data, code
- Integration with several **analysis ecosystems**: R, Python, ...
- All software from LCG releases
  - And you can install more too, e.g. with pip

Advantages also in terms of **Hardware**:
- Access to larger computational resources:
  - Interfaced to powerful **clusters**, such as **Spark**
  - Allow off-loading intensive computation

**SWAN** hides complexity from the users
FCC Interactive analysis with **SWAN**

Provide at CERN “Data Analysis as a Service”

Interface chosen: Jupyter notebooks:
- Analysis only with a web browser
- Easy sharing of scientific results: plots, data, code
- Integration with several analysis ecosystems: R, Python, ...
- All software from LCG releases
  - And you can install more too, e.g. with pip

Advantages also in terms of **Hardware**:
- Access to larger computational resources:
  - Interfaced to powerful clusters, such as Spark
  - Allow off-loading intensive computation

**SWAN** hides complexity from the users
Conclusions

- Building blocks, demonstrators and prototypes delivered by R&D programmes
- Solid, tested and flexible software infrastructure will be crucial to validate experimental results
- EP-SFT group committed to make this happen using its expertise in HEP Software
  - Sustainability and flexibility
  - New developments on base packages driven by the community
  - Comfortable environment for developers and users

Immediate Challenges

- Spack for LCG Releases:
  - Easier integration between LCG and FCC software
- FCC Input to the EP R&D project on Turnkey Stack
Backup
Every stage of the FCC Build process is based on the equivalent LCG product.

- **Releases**: 93.0.0 built on top of LCG_93
  - FCC external release
  - LCG Release

- **Nightlies**: Today built on top of Today
  - FCC nightly build
  - LCG nightly build

- **Views**: 93.0.0 / Today
  - FCC release/nightly view
  - LCG_93 / Today

- Continuous Integration
- Development
- SWAN Analyses