

# The FCC software - how to keep SW experiment independent

*Monday 10 October 2016 11:45 (15 minutes)*

The Future Circular Collider (FCC) software effort is supporting the different experiment design studies for the three future collider options, hadron-hadron, electron-electron or electron-hadron. The software framework used by data processing applications has to be independent of the detector layout and the collider configuration. The project starts from the premise of using existing software packages that are experiment independent and of leveraging other packages, such as the LHCb simulation framework or the ATLAS tracking software, that can be easily modified to factor out any experiment dependency. At the same time, new components are being developed with a view to allowing usage outside of the FCC software project; for example, the data analysis front-end, which is written in Python, is decoupled from the main software stack and is only dependent on the FCC event data model. The event data model itself is generated from configuration files, allowing customisation, and enables parallelisation by supporting a corresponding data layout. A concise overview of the FCC software project will be presented and developments that can be of use to the HEP community highlighted, including the experiment-independent event data model library, the integrated simulation framework that supports Fast and Full simulation and the Tracking Software package.

## Tertiary Keyword (Optional)

## Secondary Keyword (Optional)

Analysis tools and techniques

## Primary Keyword (Mandatory)

Data processing workflows and frameworks/pipelines

**Authors:** ROBSON, Alice (Universite de Geneve (CH)); DELL'ACQUA, Andrea (CERN); ZABOROWSKA, Anna (Warsaw University of Technology (PL)); HEGNER, Benedikt (CERN); HELSENS, Clement (CERN); BERNET, Colin (IPNL/CNRS (Lyon)); LINGEMANN, Joschka (CERN); HRDINKA, Julia (Vienna University of Technology (AT)); VOLKL, Valentin (University of Innsbruck (AT)); DRASAL, Zbynek (CERN)

**Session Classification:** Track 2: Offline Computing

**Track Classification:** Track 2: Offline Computing