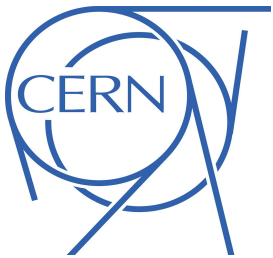




MDI Software Integration

G Ganis, C Helsens, V Volkl
For the FCC Software team



The context



- Purpose of this workshop is to *progress on the mechanical design and integration of the interaction region including the MDI area*
- Ultimately the goal is to find a solution that provides acceptable beam- and MDI- related backgrounds
- For the physicist this is a source of systematics and need to be controlled as precisely as possible
- A common software framework is the best way to achieve this goal

Beam- & MDI- related backgrounds



- [A Andrezza](#) has discussed the main backgrounds for the detectors
 - Multiple scattering in beampipe
 - Synchrotron radiation
 - Beamstrahlung
 - Pair production
 - Off-momentum particles
- Critical aspects
 - For performance: detector occupancy
 - Possibly also radiation damage

Available codes



Software Packages to calculate all those backgrounds exist

- Non exhaustive list
 - MDISim: SR, Single beam induced backgrounds
 - SYNC_BKG, SYNRAD+: SR
 - GuineaPig++: IP backgrounds
 - (In)coherent Pairs Creation, $\gamma\gamma$ to hadrons
 - Pythia8: $\gamma\gamma$ to hadrons
 - BBBrem+SAD: Radiative Bhabhas

SR: Synchrotron Radiation

FCCSW



- FCCSW is the offline software framework for FCC
- FCCSW aims to be a flexible tool to
 - Describe a detector concept (DD4hep), simulate its response to particles, reconstruct the collisions
 - To evaluate its physics potential
- Detector description includes relevant IR elements
 - E.g. beampipe
- However, the only beam-related effects included in the simulation of a IP induced process are beam spreads in energy and position
- Interoperability with “MDI” codes required

Ways to interface codes



Interoperability I

- Level 0 - Common Data Formats
 - Allows interoperability between different programs, even running on different hardware
 - E.g., HepMC event records, LCIO, GDML, ALFA messages
- Level 1 - Callable Interfaces
 - Basic calling interfaces defined by the programming language
 - Cross language calls are, of course, possible
 - Can be dependent on the compiler and language version (C++ in particular)
 - Details are important
 - how to handle errors and exceptions, is it thread safe, are objects const, dependent libraries and runtime setup
 - E.g., FastJet, Eigen, Boost

G A Stewart, FCC week 2019

Common Data Formats

- Widely used by physics Monte Carlo generators w/ standard data formats such as HepMC or LHE
- E.g. the [Les Houches Event format](#) (LHE)
 - XML-like structure

```
<LesHouchesEvents version="1.0">
<init>
  compulsory initialization information
</init>
<event>
  compulsory event#1 information
</event>
(further <event> ... </event> blocks, one for each event)
</LesHouchesEvents>
```

- Support for *compulsory* and *optional* information
- Simpler (and solid) solution for 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, ...)
 - This may include the interaction in the beam-pipe (as in MDISim)
- FCCSW simulates the interaction of these particles in the the detector to evaluate occupancies
- FCCSW may overlay these events to “signal” events for a more detailed background simulation
 - This may also be done with a weighted mixture of MDI processes

Workflow - current status



- FCCSW *cannot* read MDISim, GuineaPig++ output formats
 - Some private work started in this direction to be continued and integrated
 - ILCSoft (Marlin) can read GP format
 - FCCSW can read standard formats events, e.g. LHE
- FCCSW *can* overlay events
 - Same technology / modules used to simulated pile-up and FCC-hh can be used

Wish list



- Provide easier access to (source) codes
 - (Almost) always required to contact the authors
 - Exception is GuineaPig++
 - Code on CERN GitLab/clic-software
 - Nice repository with scripts and instructions setup by G Voutsinas
- Use an established data format for outputs, or at least use the same data format for all the MDI programs
- Use same language (DD4hep?) for common geometry elements

Outlook



- A common software framework would be the best way to achieve a fruitful collaboration between the machine and the detector
 - Need to understand in what extend this can be done
 - Need to identify clear deliverables
 - Need to define a detailed schedule