FINANCÉ PAR ANR

IPHC Institut Pluridisciplinaire Hubert CURIEN STRASBOURG

GRPHE UHA

MadAnalysis 5 *status and news*

Eric CONTE, Benjamin FUKS

GDR Terascale @ Annecy
28-30 October 2013

Outlines

Overview of MadAnalysis 5 v1.1.9

The normal mode and its metalanguage

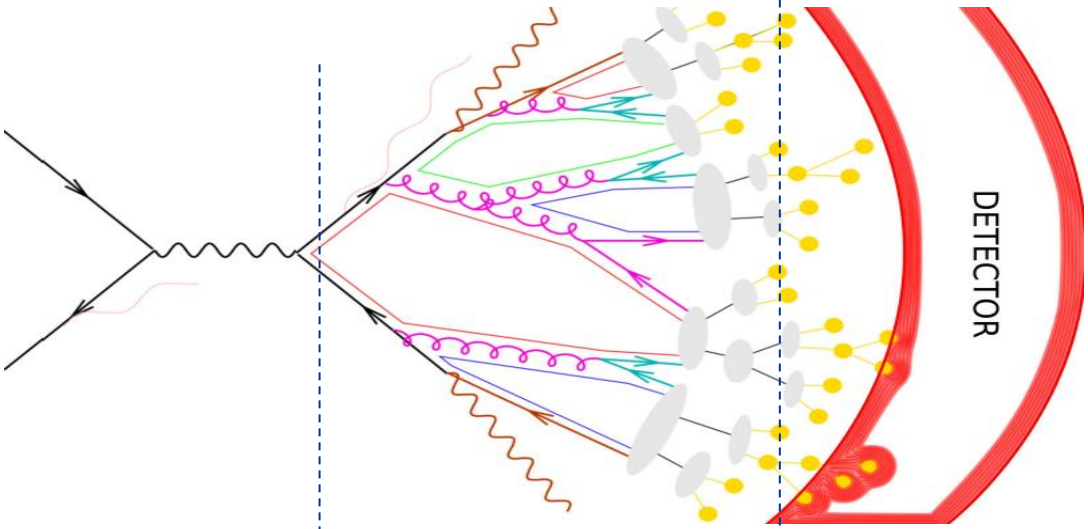
Fast-simulation packages

Summary & perspectives

Overview

Starting points of the project:

Several levels of sophistication for phenomenological analyses



Parton level	Hadron level	Reconstructed objects level
LHE files	STDHEP/HEPMC files	LHCO, ROOT files

Relevant features of design

- ☑ User-friendly
- ☑ Flexible
- ☑ Efficient
- ☑ Easy to maintain

A unique framework : MadAnalysis 5

Overview

Scope:

Defining an analysis with a dedicated metalanguage

Reading of signal and background event files

Supporting any event file format (STDHEP, HEPMC, LHE, LHCO, ROOT)

optional

Processing data

- Clustering
- Fast-simulation

Analyzing data

- Selection cuts
- Filling histograms

Analysis report

histograms, selection efficiencies, S/B-like ratio table

optional

Storing events into output files

LHE and LHCO file formats

Overview

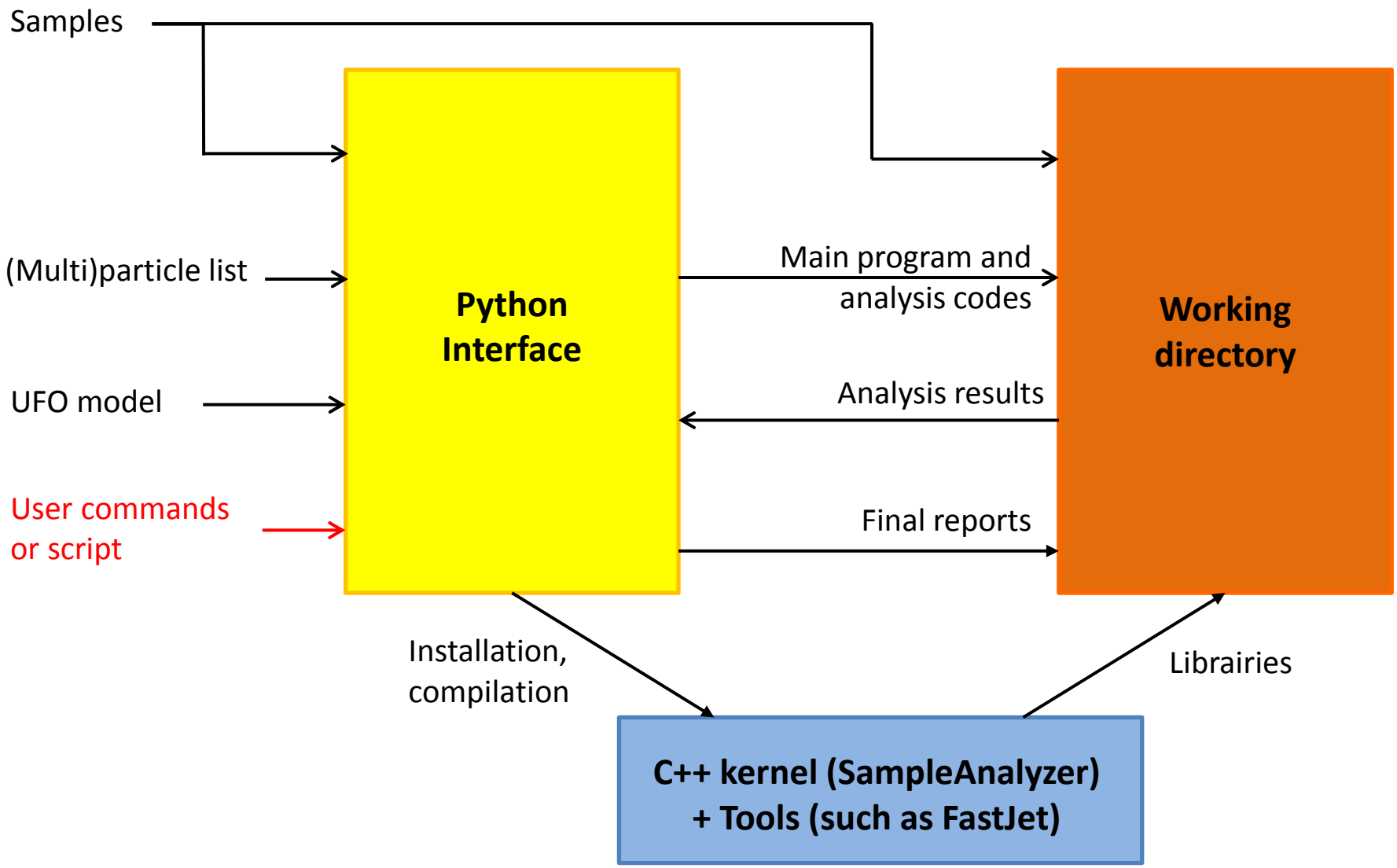
Requirements:

Mandatory	Optional
Python 2.6 or a more recent version (but not the 3.X series) **GNU GCC** compiler **ROOT** 5.27 or a more recent version	**zlib** **Latex** / PDFLatex **FastJet** 3.0 or a more recent version **Delphes** 3.0

Where MadAnalysis can be downloaded ?

- From the official website https://launchpad.net/madanalysis5 **(new address)**
- From MadGraph 5 interface **(available soon)**

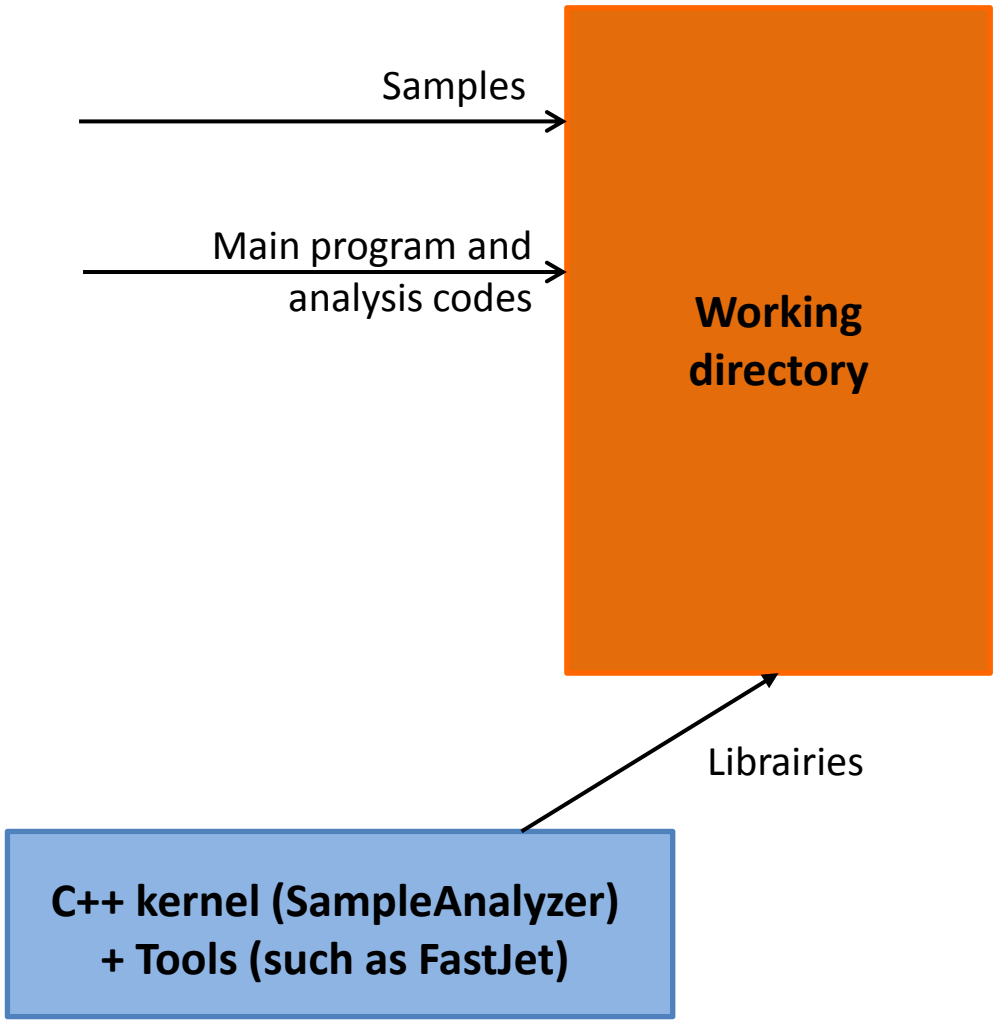
Overview


Samples
(Multi)particle list
UFO model
User commands
or script
Python
Interface
Main program and
analysis codes
Analysis results
Final reports
Working
directory
Installation,
compilation
Librairies
C++ kernel (SampleAnalyzer)
+ Tools (such as FastJet)

Overview

MadAnalysis has an **expert mode** (developer-friendly) :

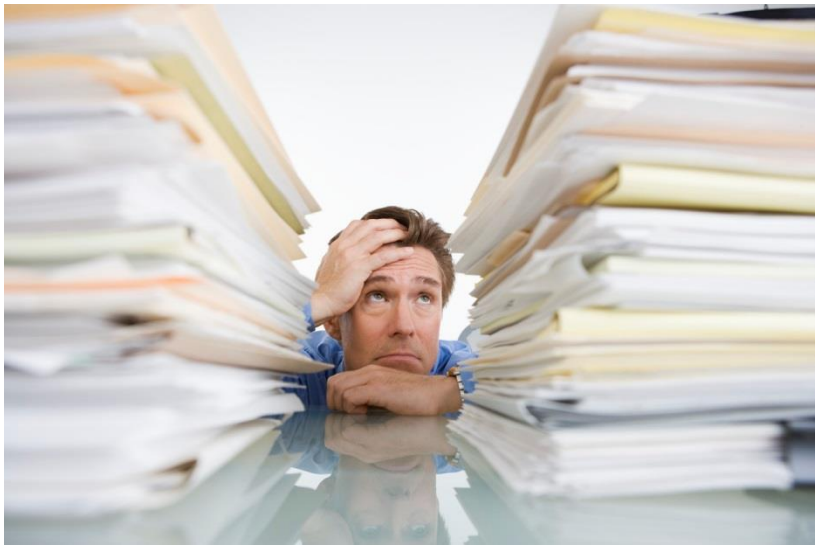
- C++ programming within the SampleAnalyzer framework.
- The Python interface creates a blank analysis as a starting point.



Overview

Validation suite:

- Series of automated tests targeting to check all the functionalities of the program. Last ~ 3 hours.
- Launched for each compilation configuration of the kernel.
- One bug report → creation of a dedicated automated test.



Documentation:

- **Manual** published in January 2013:
  Comput. Phys. Commun. 184 (2013) 222
- **First tutorials** are available on the website
  https://madanalysis.irmp.ucl.ac.be/wiki/tutorials
  → Firsts steps with the Expert Mode

Normal mode and its metalanguage

First start of MadAnalysis 5:

- **Execution**

Parton level	Hadron level	Reconstructed objects level
bin/ma5 or bin/ma5 –P	bin/ma5 –H	bin/ma5 -R

- **Initial sequence:**
  - **Step 1: Testing all dependencies.**
  - **Step 2: Compiling (if necessary) the C++ library.**
  - **Step 3: Importing the list of particles and multiparticles (from MadGraph if this program is found on your system).**

Normal mode and its metalanguage

Defining new particles and multiparticles

- Particles are defined by **labels**, which could point to one or several **PDG-id**.
- SM and MSSM labels are automatically loaded at the starting of MadAnalysis.
- The user can define his own labels :

```
ma5> define mu = mu+ mu-
```

- All labels defined in a UFO model can be loaded too.

Importing datasets

- For MadAnalysis, a **dataset** is a collection of samples which will be merged.
- All sample files are stored in a dataset.

```
ma5> import tt*.lhe
```

```
ma5> import tt*.lhe as ttbar
ma5> import Wj*.lhe as Wjets
```

- Possibility to tag datasets as **signal** or **background**.

Normal mode and its metalanguage

Defining an analysis: plots and/or cuts

- **Histograms**
  - Observable can be related to the event or the properties of a particle
  - Plethora of observables: N, E, ET, M, MT, P, PT, PX, PY, PZ, THETA, ETA, …, ALPHAT
  - Combining particles

```
ma5> plot MET
ma5> plot PT(mu)
```

```
ma5> plot M(mu+ mu-)
```

- **Cuts : selecting / rejecting events**

```
ma5> reject MHT < 50
ma5> select N(mu) >= 2
```

- **Cuts : selecting / rejecting a particle or a combination**

```
ma5> select (mu) PT > 50
ma5> select 80 < M (mu+ mu-) < 100
```

Normal mode and its metalanguage

Defining an analysis: plots and/or cuts

Several options or syntaxes allow to extend the potential of MadAnalysis.
Some examples:

- By default, a combination is interpreted as the vector sum of momenta.
  This interpretation can be changed by adding a prefix to the observable label.
  For instance : vPT, sPT, dsPT, dvPT, rPT

- List of observables specific to the reconstructed object level : ISOL, HE_EE, NTRACKS, …

- Selecting a particle according to its rank in energy (or to other observables)

```
ma5> plot PT(mu+[1])
```

- Selecting a particle according to its history (requirements on mother, grand-mother …)

```
ma5> plot PT(mu+ < w+ < t~)
```

Normal mode and its metalanguage

Launching the analysis:

This can be done by the command **`submit`**

- Creating a working directory (with a default name if no name is specified)
- Compiling the C++ job
- Launching the analysis over the different samples contained in the datasets

```
* SampleAnalyzer 2.0 for MadAnalysis 5 - Welcome.
* Option choices: selecting analysis = 'MadAnalysis5job'.
* Extracting the following sample files:
* 1/4  ~/samples/ttbar_sl_1.lhe.gz
    => file size : 107.09 Mo
    => sample produced by MadGraph.
    => progress [======================>          ]
```

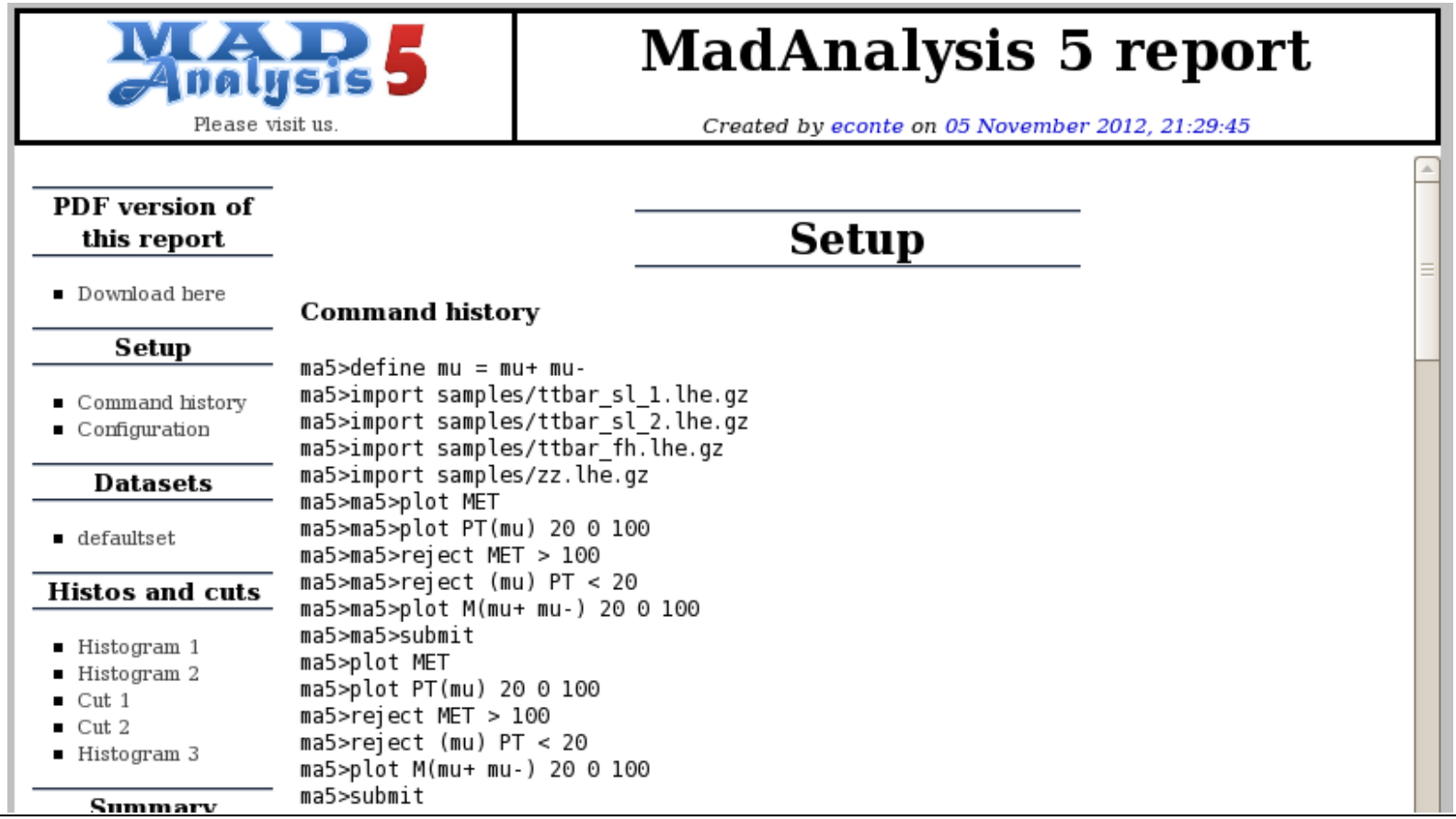
If you modify, after the submission, the analysis or the layout of the plots , the results can be updated in an optimized way by the command **`resubmit`**.

Normal mode and its metalanguage

Opening a generated report:

The command **open** displays the HTML report of the last job created.

Reports in PDF and DVI format are also available.



Fast-simulation

MadAnalysis 5 = multi-purpose interface

External static or shared libraries are linked to the built executable file.

Technical packages

- **ZLIB**: Reading and writing gzip-compressed files.

Physics packages

- **FASTJET**: M. Cacciari, G.P. Salam and G. Soyez, Eur.Phys.J. C72 (2012) 1896 [arXiv:1111.6097]
  - Generation of the ME/PS merging validation plots.
  - Applying an ideal detector simulation (jet clustering algorithm) to hadronic events.
- **DELPHES 3.0.10**: arXiv:1307.6346
  - Reading ROOT files.
  - Applying a realistic detector simulation to hadronic events.

Assisted installation

```
ma5>  install fastjet
ma5>  install delphes
ma5>  install zlib
```

Fast-simulation

How to go from hadronic events to reconstructed events ?

MadAnalysis 5 recipe based on FastJet

Delphes

Jet-clustering and identification algorithms
+ efficiencies on (mis)identification

- Selecting the fast-simulation package:

```
ma5>  set main.fastsim.package =
delphes          fastjet          none
```

Fast-simulation

(Almost) ideal detector simulation based on FastJet

Thanks to A. Alloul for validation

- Selecting the fast-simulation package called: `fastjet`

```
ma5>  set main.fastsim.package = fastjet
```

- Adopting a fast-simulation package → new options:
  - **Large selection of jet clustering algorithms**

```
ma5>  set  main.fastsim.algorithm  =

antikt            cdfjetclu         genkt    kt       siscone
cambridge         cdfmidpoint       gridjet  none
```

  - **algorithm** & **object-identification** parameters. For instance:

```
ma5>  set  main.fastsim.algorithm  =  antikt
ma5>  set  main.fastsim.ptmin   =  5
ma5>  set  main.fastsim.radius  =  0.5

ma5>  set  main.fastsim.exclusive = true
ma5>  set  main.fastsim.bjet_id.efficiency  =  0.6
ma5>  set  main.fastsim.bjet_id.misid_cjet  =  0.
ma5>  set  main.fastsim.bjet_id.misid_ljet  =  0.
```

Fast-simulation

Realistic detector simulation based on Delphes 3

- Selecting the fast-simulation package called: `delphes`

```
ma5>  set main.fastsim.package = delphes
```

- Adopting a fast-simulation package → new options:
  - **Choice of the default Delphes cards**

```
ma5>  set  main.fastsim.detector  =
              cms            atlas
```

  - **Pile-up input:**

```
ma5>  set  main.fastsim.pileup  = "MinBias.pileup"
```

- Once the analysis is submitted, MadAnalysis 5 invites the user to edit the configuration card with your favorite text editor (set the EDITOR shell variable).

```
ma5>  Would you like to edit the configuration card used by
Delphes ? yes/no [default = no]
```

- Delphes fast-simulation is launched event-by-event. No intermediate file is produced.

Summary and perspectives

- **MadAnalysis 5 = a unique framework with two ways to use it:**
  - **Normal mode:** python interface with intuitive commands.
  - **Expert mode:** requiring programming skills (C++, ROOT).

- **Relevant features of MadAnalysis 5 design:**
  - **User-friendly** → professional analyses in a simple way.
  - **Flexible:** no limit on the analysis complexity.
  - **Easy** to maintain and to validate.

- **Interface to physics-relevant tools:**
  - **Fast-Jet** for clustering and for generating ME/PS merging validation.
  - **Delphes 3** for fast-simulation.
  - Expected soon: shower programs.

MadAnalysis 5

ma5team@iphc.cnrs.fr
https://launchpad.net/madanalysis5
Comput. Phys. Commun. 184 (2013) 222