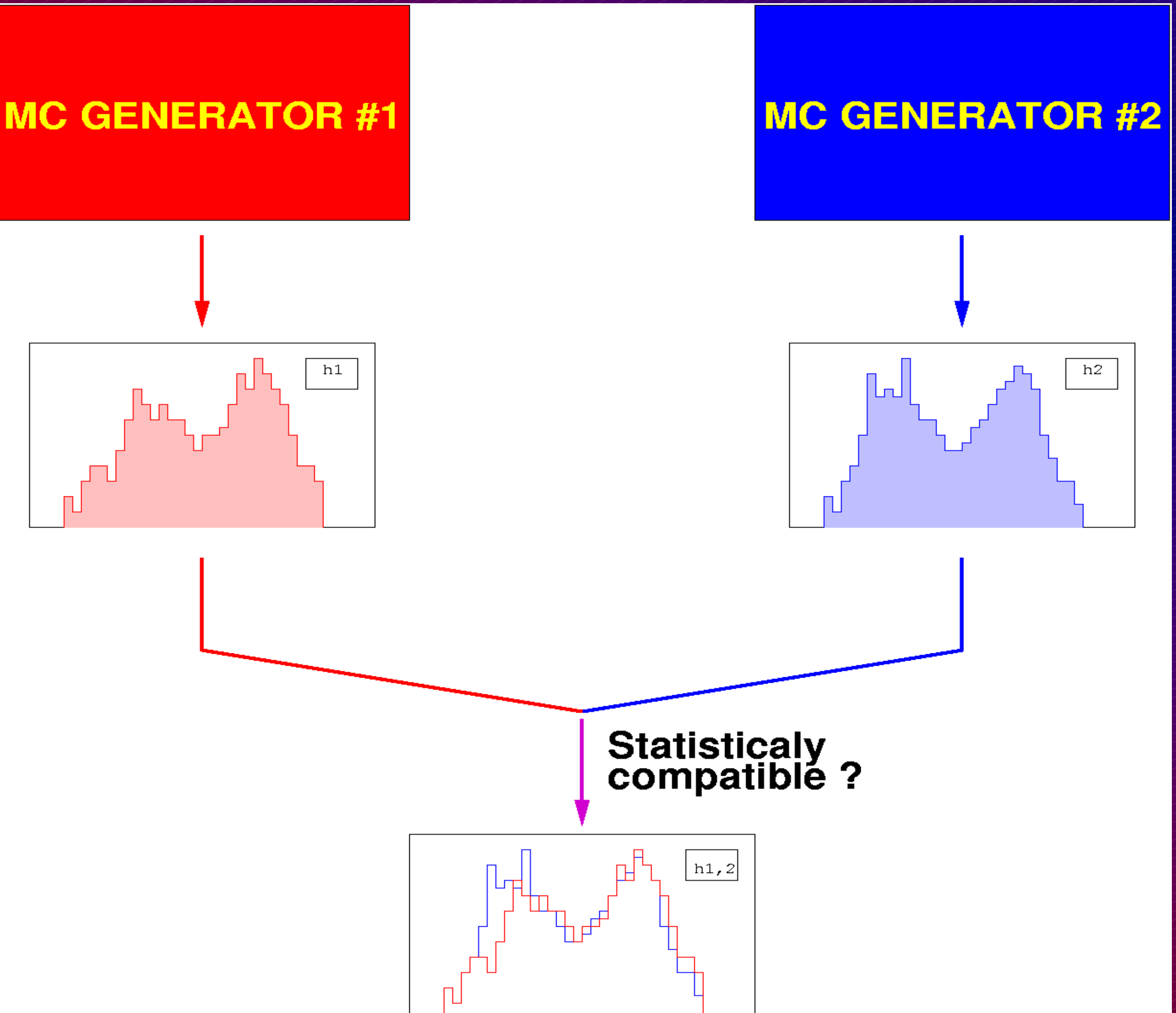
MC-TESTER:

a universal tool for comparisons of Monte Carlo predictions in High Energy Physics

Piotr Golonka
Piotr.Golonka@CERN.CH
CERN EP/ATR, AGH/INP Kraków

Zbigniew Wąs (CERN/TH, INP Kraków)
Tomasz Pierzchała (UŚI Katowice)

MC-TESTER: why?



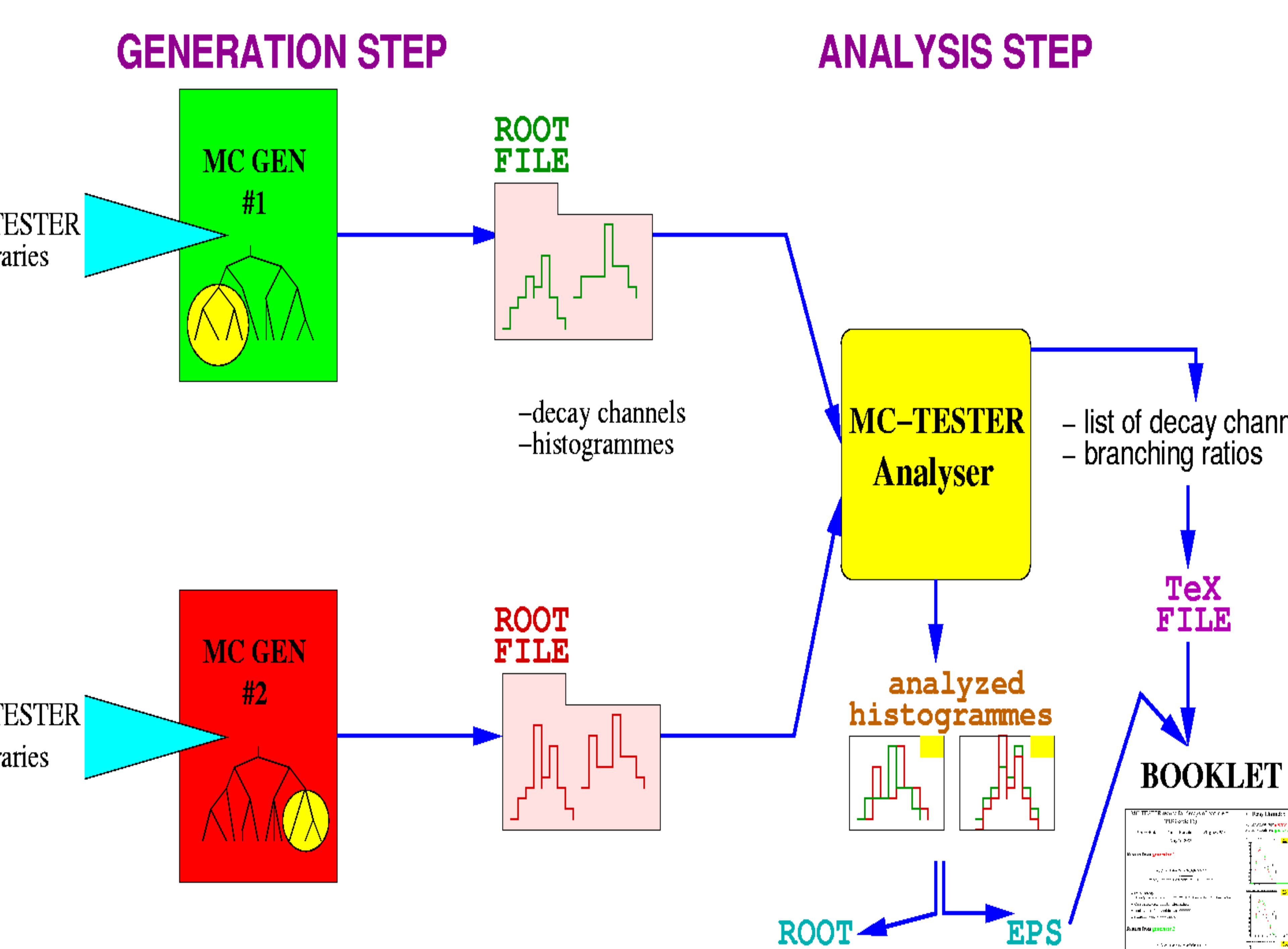
- Comparisons of physics content
- Debuging, development
- Library version maintainance
- Automated tests of installations of large MC

Areas of use:

- ***Exploration tool***
  for new (versions/implementations of) generators
- **Comparisons of physics in generators:**
  - ***Particles' decays***
  - ***Parton-level (2 → n processes)***
- **Debuging:**
  - ***Generator versions ("flavours")***
  - ***Porting the code (F77→C++ transition)***
  - ***Phase-space inefficiencies***
  - ***Event record formats***

MC-TESTER analysis:

- *Generation step:*
  - For every event produced by tested generator, certain (decay) processes are searched for and identified
  - For identified process, all possible ***invariant masses*** are histogrammed
  - At the end of the run, histogrammes are stored in root file
- *Analysis step:*
  - Creates a table of all processes, matches the corresponding processes from two runs
  - Calculates branching ratios (and statistical errors)
  - Performs statistical analysis all histograms from two generated root files, produces EPS files (and root file)


GENERATION STEP
ANALYSIS STEP
MC GEN #1
ESTER
aries
ROOT FILE
MC GEN #2
ESTER
aries
ROOT FILE
–decay channels
–histogrammes
MC–TESTER Analyser
– list of decay chann
– branching ratios
TeX FILE
analyzed histogrammes
BOOKLET
ROOT
EPS

Features:

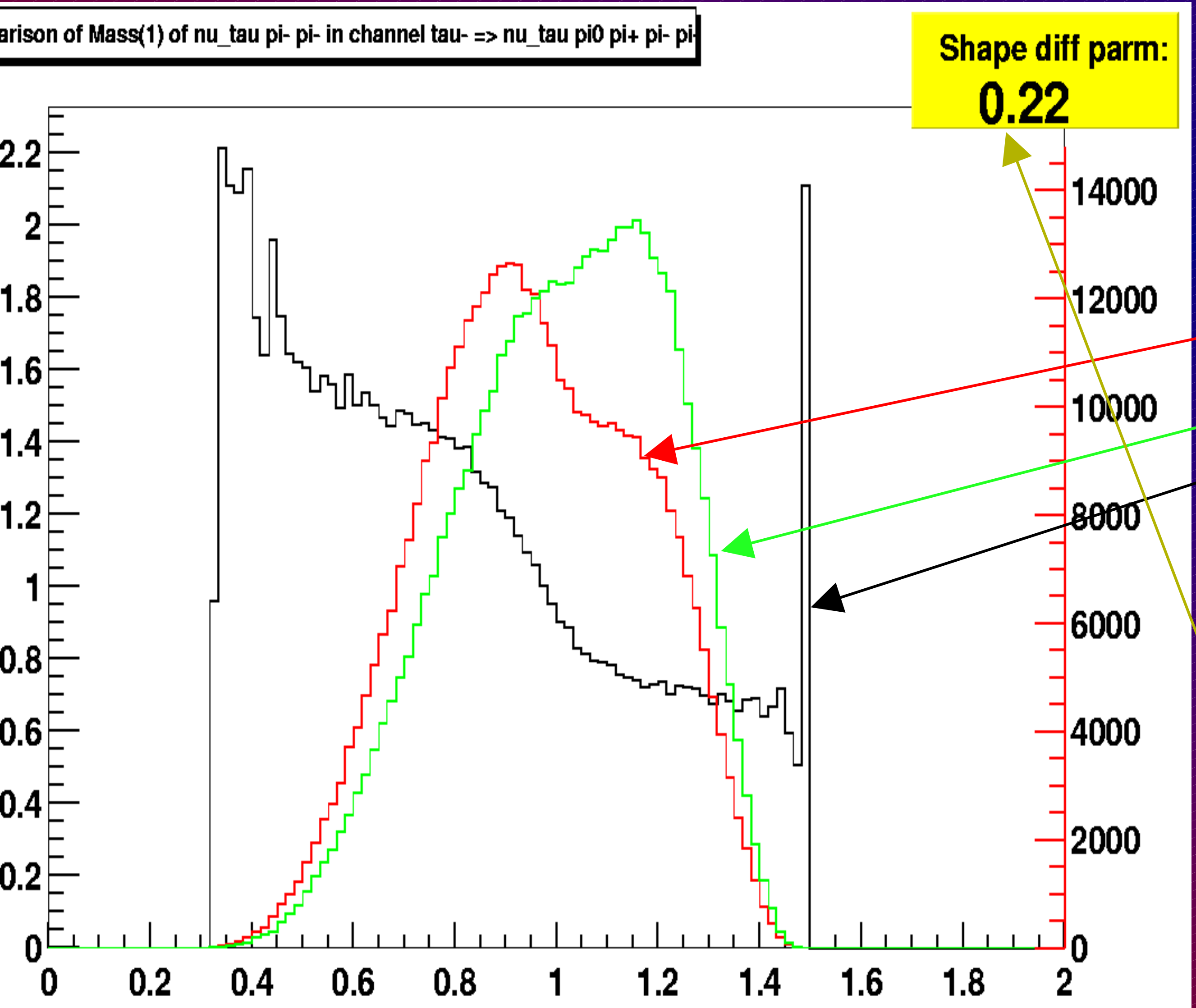
- Written completely in **C++** (compiled code and ROOT macros)
- Directly usable from **F77** code
- **Simple**: two F77 routines to be called in the code
- Support for **HEPEVT,LUJETS,PYJETS**
- **Runtime** parameters specified by **root macros** (flexibility! e.g. read parameters from external file!)
- **Easy to integrate**/use with existing MC installations (2 .so/.a libraries + root libs)
- **Examples** of use provided (TAUOLA, PYTHIA)

Table of decay modes:

Decay channel	Branching Ratio ± Rough Errors Generator #1	Generator #2	Max. shape dif. param.
$\tau^- \to \nu_\tau \pi^0 \pi^-$	25.3683 ± 0.0159%	25.3085 ± 0.0159%	0.04375
$\tau^- \to e^- \tilde{\nu}_e \nu_\tau$	17.8479 ± 0.0134%	18.1093 ± 0.0135%	0.00000
$\tau^- \to \mu^- \tilde{\nu}_\mu \nu_\tau$	17.3866 ± 0.0132%	17.6326 ± 0.0133%	0.00000
$\tau^- \to \nu_\tau \pi^-$	11.0768 ± 0.0105%	11.1765 ± 0.0106%	0.00000
$\tau^- \to \nu_\tau \pi^0 \pi^0 \pi^-$	9.1865 ± 0.0096%	9.1171 ± 0.0095%	0.09413
$\tau^- \to \nu_\tau \pi^+ \pi^- \pi^-$	8.9837 ± 0.0095%	8.8828 ± 0.0094%	0.09368
$\tau^- \to \nu_\tau \pi^0 \pi^+ \pi^- \pi^-$	4.2973 ± 0.0066%	4.5319 ± 0.0067%	0.30310
$\tau^- \to \nu_\tau \pi^0 \pi^0 \pi^0 \pi^-$	1.0765 ± 0.0033%	1.0090 ± 0.0032%	0.00724
$\tau^- \to \nu_\tau K^-$	0.7202 ± 0.0027%	0.7138 ± 0.0027%	0.00000
$\to \nu_\tau \pi^0 \pi^0 \pi^+ \pi^- \pi^-$	0.4990 ± 0.0022%	0.0897 ± 0.0009%	0.00000
$\tau^- \to \nu_\tau \pi^0 K^-$	0.4785 ± 0.0022%	0.46[illegible] ± 0.0021%	0.00000
$\tau^- \to \nu_\tau K^0_L \pi^-$	0.4624 ± 0.0022%	0.4444 ± 0.0021%	0.00000
$\tau^- \to \nu_\tau \pi^- K^0_S$	0.4610 ± 0.0021%	0.4449 ± 0.0021%	0.00000
$\tau^- \to \nu_\tau \pi^+ \pi^- K^-$	0.3902 ± 0.0020%	0.5051 ± 0.0022%	0.52330
$\tau^- \to \nu_\tau \pi^0 \pi^- \eta$	0.1707 ± 0.0013%	0.1696 ± 0.0013%	0.00000
$\tau^- \to \nu_\tau \pi^- K^+ K^-$	0.1704 ± 0.0013%	0.1509 ± 0.0012%	0.07360
$\tau^- \to \nu_\tau \pi^0 K^0_L \pi^-$	0.1605 ± 0.0013%	0.2745 ± 0.0017%	0.92850
$\tau^- \to \nu_\tau \pi^0 \pi^- K^0_S$	0.1592 ± 0.0013%	0.2734 ± 0.0017%	0.93657
$\tau^- \to \nu_\tau \gamma \pi^0 \pi^-$	0.1559 ± 0.0012%	0.1303 ± 0.0011%	0.00000
$\tau^- \to \nu_\tau K^0_L \pi^- K^0_S$	0.1510 ± 0.0012%	0.0763 ± 0.0009%	0.00270
$\tau^- \to \nu_\tau K^0_L K^-$	0.1289 ± 0.0011%	0.0508 ± 0.0007%	0.00000
$\tau^- \to \nu_\tau K^0_S K^-$	0.1287 ± 0.0011%	0.0507 ± 0.0007%	0.00000
$\to \nu_\tau \pi^0 \pi^0 \pi^0 \pi^+ \pi^- \pi^-$	0.1094 ± 0.0010%	0.0506 ± 0.0007%	0.00000
$\to \nu_\tau \pi^+ \pi^+ \pi^- \pi^- \pi^-$	0.0803 ± 0.0009%	0.0401 ± 0.0006%	0.00000
$\tau^- \to \nu_\tau \pi^0 \pi^0 K^-$	0.0792 ± 0.0009%	0.0504 ± 0.0007%	0.291[illegible]
$\tau^- \to \nu_\tau K^0_L K^0_L \pi^-$	0.0760 ± 0.0009%	0.0372 ± 0.0006%	0.00854
$\tau^- \to \nu_\tau \pi^- K^0_S K^0_S$	0.0756 ± 0.0009%	0.0378 ± 0.0006%	0.01189
$\tau^- \to \nu_\tau \pi^0 K^0_L K^-$	0.0507 ± 0.0007%	0.0763 ± 0.0009%	0.85321
$\tau^- \to \nu_\tau \pi^0 K^0_S K^-$	0.0498 ± 0.0007%	0.0746 ± 0.0009%	0.87506
$\to \nu_\tau \pi^0 \pi^+ \pi^+ \pi^- \pi^- \pi^-$	0.0186 ± 0.0004%	0.0293 ± 0.0005%	0.00000

- Process (decay channel)
- Branching ratio for generator #1 and #2
- Rough statistical errors of branching ratios
- Maximal "Shape Difference Parameter"
- Similarity Coefficients (combined: for all decay modes)

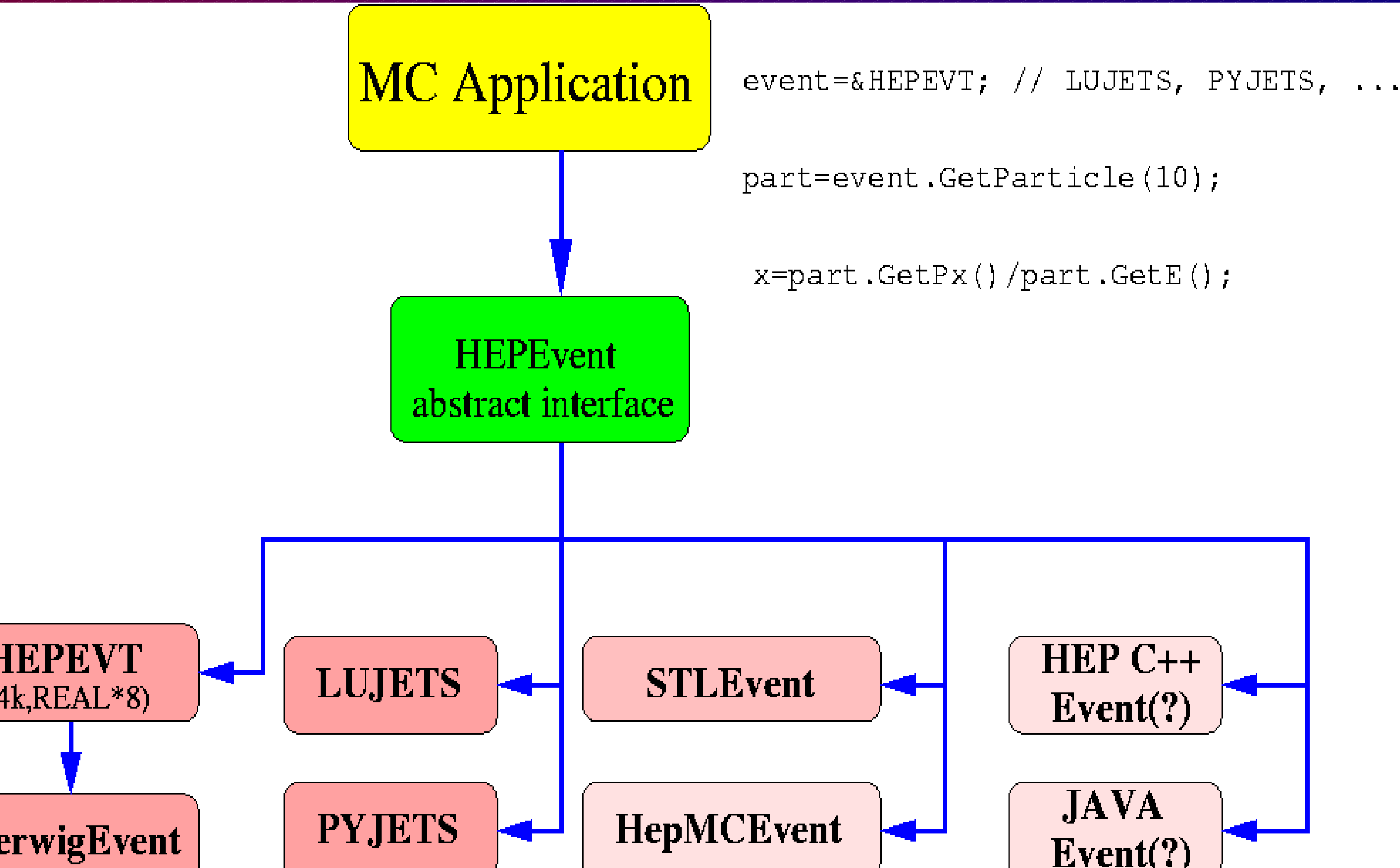
Example of histogrammes:



- Histogramme of invariant mas from generator #1 and #2
- Ratio of the tw histogrammes
- ***Shape Difference Parameter*** valu

HEPEvent library:

a unified interface to event record formats



MC-TESTER as librarian tool:

- Automated comparison tests for changing versions of code
- Differences (e.g. new features) visible at a glance!
- Verification of compatibility with other version
- Example: ***TAUOLA*** generator
    - 3 versions of code to be managed: CPC, CLEO, ALEPH

MC-TESTER as debuging tool:

Detection of integrity errors in generator systems:

- Event gramatics problems

- cannot interpret data – reports the cause in debug mode

- Configuration clashes and overwrites of private data:

  Detects places where one generators “overwrite” the internal da
  of the other -> differences in distributions (example: BaBar)

- Invariant mass distribution – good variable for finding phase-space inefficiencies

Event Record debuging:

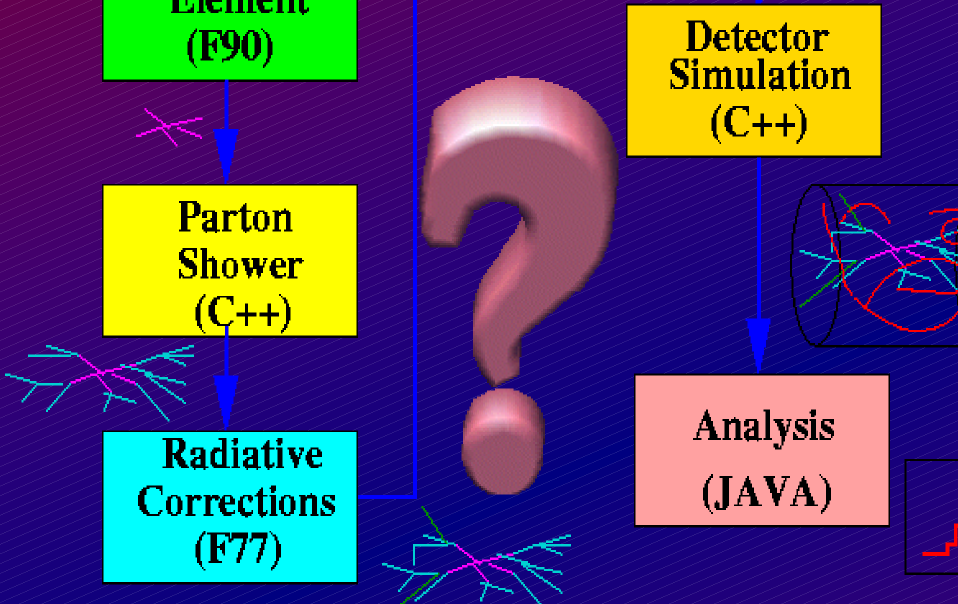
- Support for various formats of event record:
  HEPEvent library: easily extensible!

- Compare the same events stored in
  various formats of event records

- Debug event persistency methods
  and event record I/O

Event records over pipes...

n general, Monte Carlo modules
coded in different languages
(F77,F90,C++, Java,Perl...)



Hard to link single main program (needs wrappers)

One may export an event to a file in one module
executable), then read it in another

Possibly large executable files

why not use UNIX FIFO queues instead?

Synchronized parallelism!

MC-TESTER example: reading output text files from
adGraph ME generator

Requirements:

- We support ***Linux*** (RedHat 6/7/8/9/10beta) on i386 (Tested also on AMD64)
- Gcc 2.95/2.96/3.x: gcc,g++,g77,make
- root 3.X
- latex, (+dvips,gv)
- Unweighted events to be stored in one of supported event records (or text file)
- Work in progress:
  - Support for weighted events
  - support for HEPMC C++ event record
  - Support for MacOS X

Status and availability:

Project's homepage:

http://cern.ch/Piotr.Golonka/MC/MC-TESTER

Current version: 1.112, 1.1p1 (src on the web)

Documentation: (on the web)

CERN, *LANL*, *LC* preprints, to be published in *CPC*

Contact:

Piotr.Golonka@CERN.CH
Zbigniew.Was@CERN.CH
tomek@friend**.phys.us.edu.pl**

We are open to discussions about:

- possible extensions and areas of use
- event record formats to be supported (F77,C++,...)
- hardware/software platforms to be supported