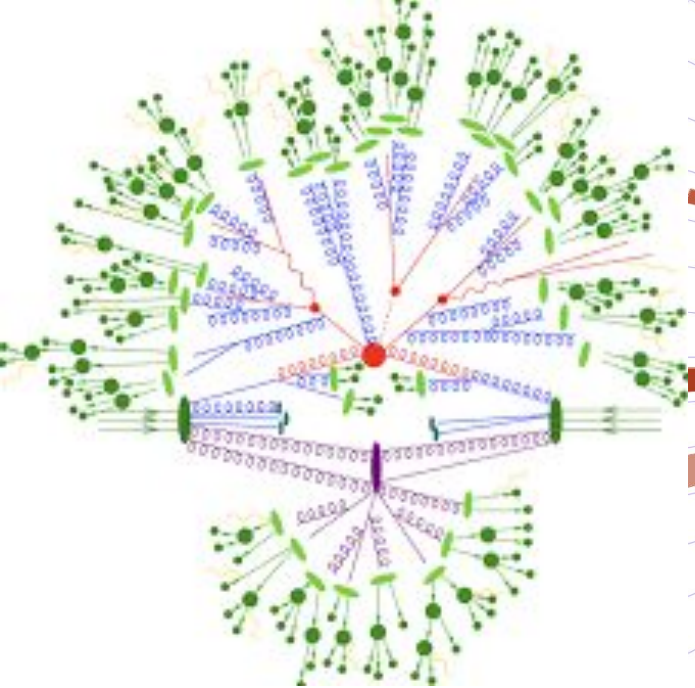
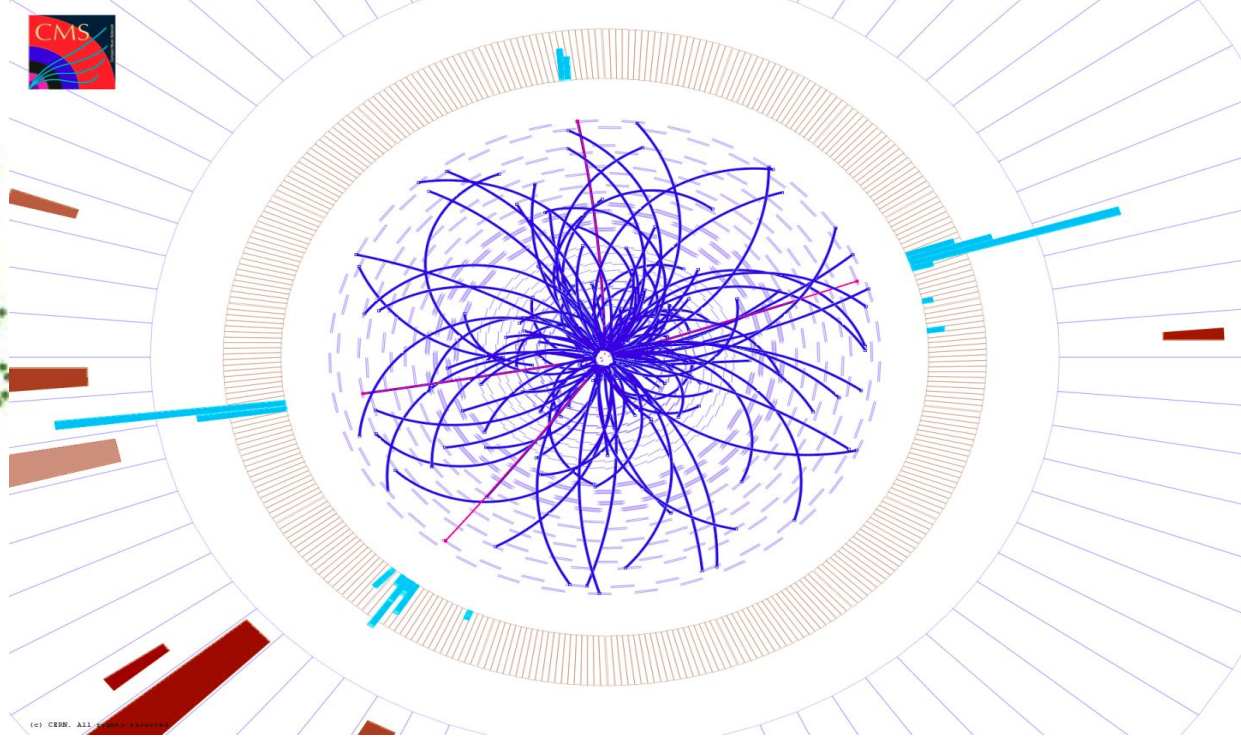
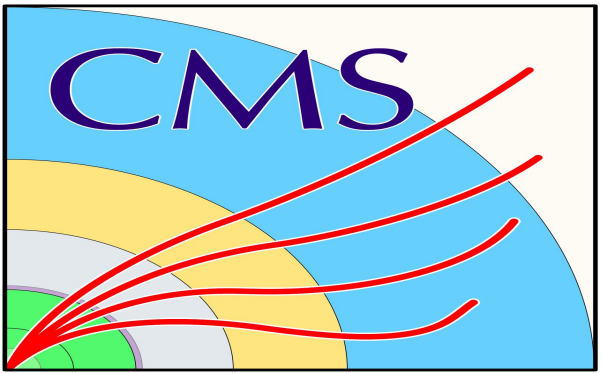






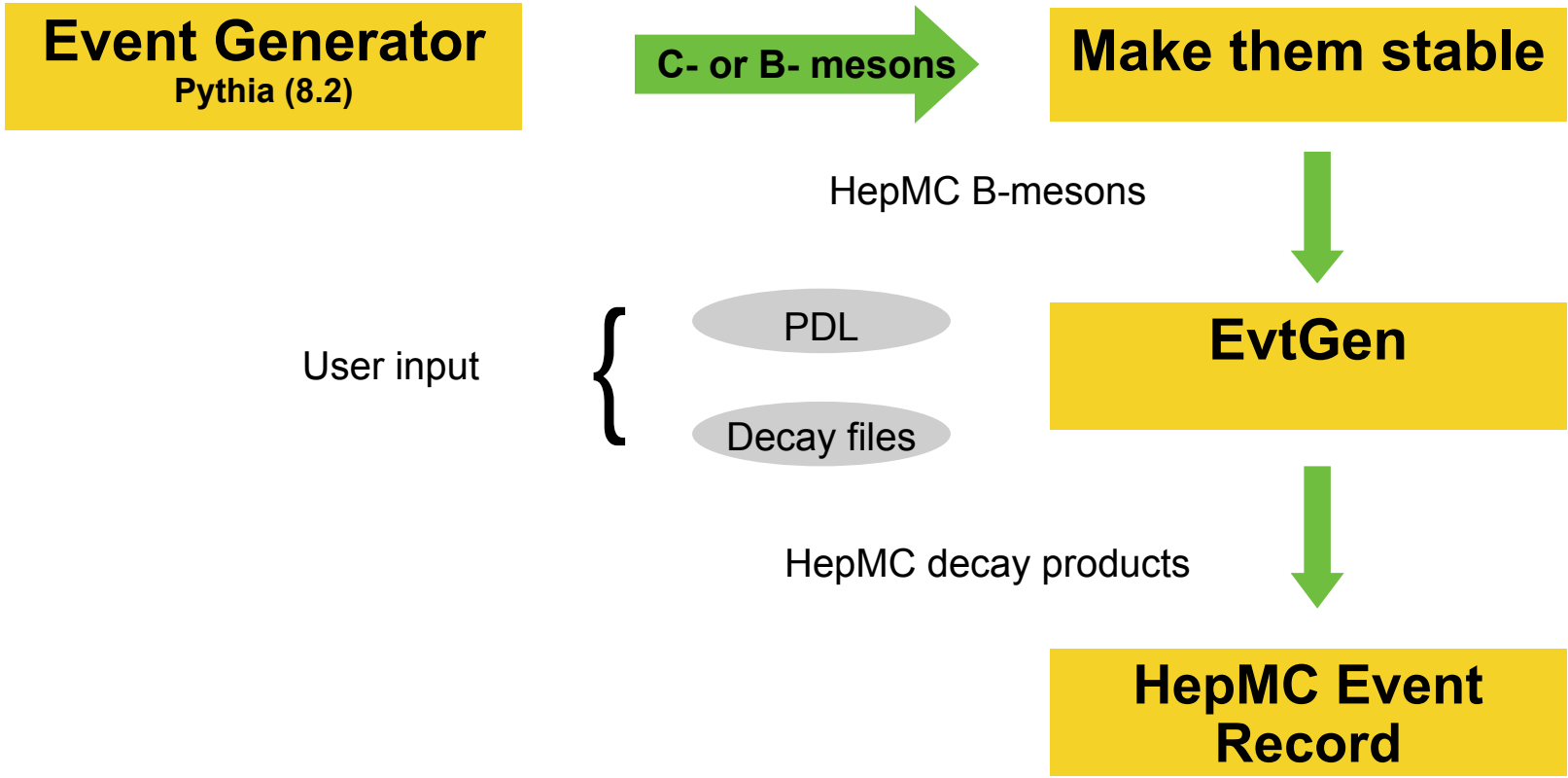
Jhovanny Andres Mejia Guisao

On behalf of the BPH-CMS Group

LHC Heavy Flavour WG topical meeting: Simulation techniques

Friday June 28, 2024

EvtGen in CMSSW – how does it works?



EvtGen examples

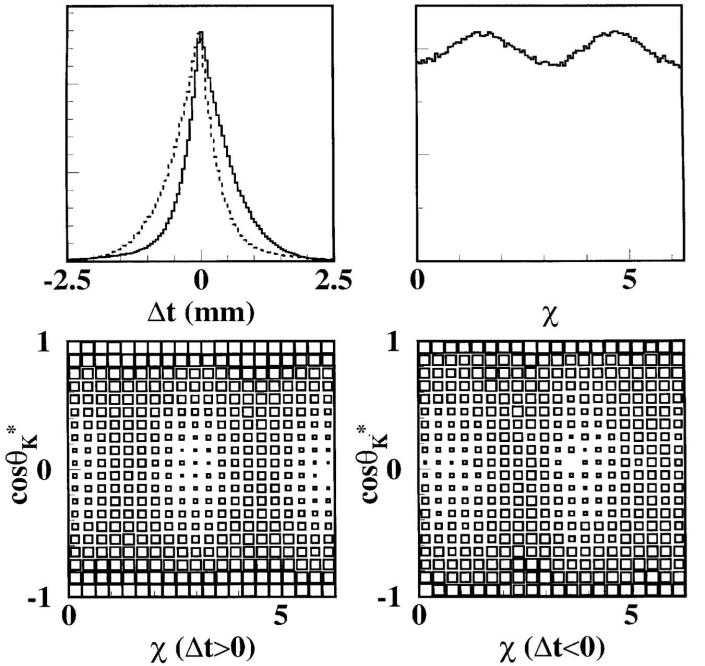


Fig. 1. Distributions for the CP violating decay $B \to J/\psi K^*$ ($J/\psi \to ll$ and $K^* \to K\pi$). The upper-left plot shows the Δt distribution for B^0 and $\bar{B}^0$ (at $t = 0$) decays. χ, shown in the upper-right distribution, is the angle between the decay plane of the J/ψ and that of the K^*. The lower plots show $\cos\theta_{K^*}$ vs. χ for $\Delta t > 0$ and $\Delta t < 0$. $\cos\theta_{K^*}$ is the decay angle of the K^*.

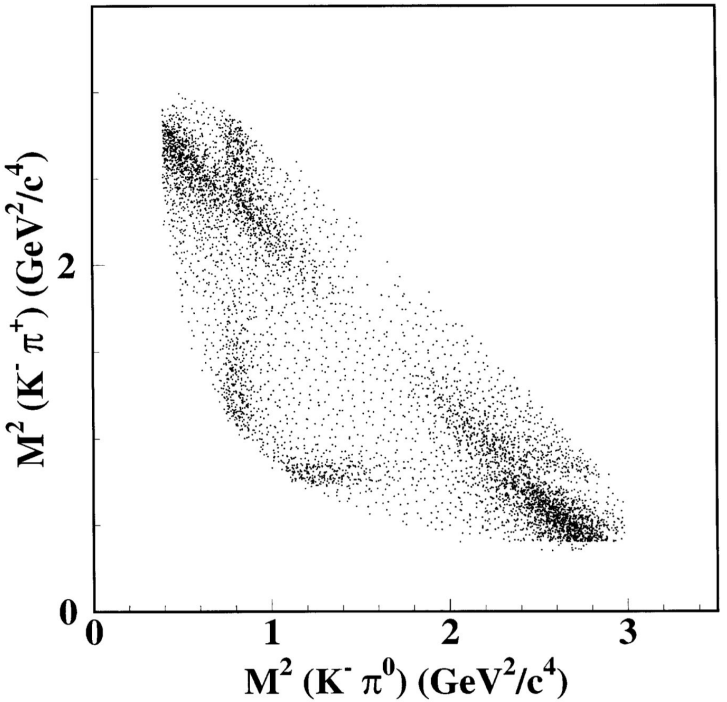


Fig. 2. The Dalitz distribution for the decay $D \to K^-\pi^+\pi^0$ from EvtGen. The resonance parameters used are measurements from the CLEO Collaboration (Ref. [2]).

Nuclear Instruments and Methods in Physics Research A 462 (2001) 152–155

Is EvtGen upto date?

- Over the years we've heard that even the latest public recommended version is not effectively the latest one. And that some experiment has its own version with developments that do not enter regularly in the latest version and with large delay in the public recommended one. [Of course we mention this aspect not to complain at all (!) but to increase among all of us the awareness that synchronization among experiments is essentially lost].

- More in general, we received many internal inquiries asking why the particle properties and decay table (including non-PHSP models for the angular characteristics of 3-body decays and the intermediate resonances; for example a pentaquark model - inspired by LHCb results - for $\Lambda_b \rightarrow J/\psi\, p\, K$) are not synchronized with latest experimental measurements.

- However, since the development was moved to git (HepForge) it seems more active (https://phab.hepforge.org/source/evtgen/)

- A relevant point, beyond Heavy Flavour physics, is that the inclusive Branching Factions have a impact on the b-/c-jets description which is used also in most "high-p_T"analyses.
  For instance CMS and ATLAS have not been always aligned (ex.: CMS moved to v2.0 when ATLAS remained to v1.6 for some time).

Procedures within CMS

- Being a multi-purpose experiment, CMS has to deal with several requirements at once.
- Implementing a new software cycle or a patch in the generators may be not “fast” enough, since several steps need to be met (implementation, validation, approval, etc.) before actually going into production.
- EvtGen 2.0 is the latest version, however it involves (upto run 2)
    - **TAUOLA++ 1.1.8 (1.1.5)**
    - **PHOTOS++ 3.64 (3.61)**
    - **PYTHIA at least 8.243 (8.303)**
    - **HepMC 2.6.10 (2.6.7)**

  all of them has to be validated, not just in the context of BPH, but CMS.

CMS did the update all this packages, in preparation for run 3, couple of years ago. EvtGen has implemented cmake build instead of the old config method.

Points that CMS colleagues would like to discuss

➔ **EvtGen cannot be used in a multithreaded/multicore environment efficiently.**
- This is the future of computing, tools that cannot adapt will be marginalized.
- EvtGen needs to be thread safe.

➔ **A better common strategy is needed for handling both inclusive and exclusive samples.**
- Is it possible find an optimal solution here? (which involves also Pythia)

➔ **To deal with low productions of B_C mesons there is BCVEGPY, but what about strange baryons (Ξ_b, Ω_b) : how to solve issues like this, characterized by a "low generation efficiency"?**

➔ **Is a single state-of-the-art EvtGen setup desired?**
- Can developments in each experiment be offloaded and approved regularly in official EvtGen releases?
  - ✓ All of the community can benefit from developments and results can be compared correctly
  - ✓ Should/could this include PHOTOS and Tauola (or the generators to replace them)?

A shared solution of this kind would be greatly appreciated.

Summary

➔ **Main tool for BPH**

- Event generators: (Pythia8 or BCVEGPY or anything else interfaced to pythia) + EvtGen
- Standard versions, no special modifications
- EvtGen 2.0 is in production since Run 3.

We think that what initially triggered this effort in HFWG is the possibility to have a state-of-art /updated EvtGen version shared among experiments and always aligned along the time.

Benefit: Physics comparisons or even combinations of results would be straightforward.

Backup slides

What does EvtGen 2.0 has in comparison with 1.6

1) Several fixes and improvements, which may impact or not:

   a) EvtSVP, EvtVVP and EvtTVP models
   b) EvtDDalitz, EvtbTosllVectorAm, EvtFlatQ2, EvtBcXMuNu
   c) Usage of same properties in EvtGen and pythia model
   d) improve in EvtTauolaEngine
   e) Reimplement code in EvtBTo3pi.F, EvtBTo3piMPP.F, EvtBTo3piP00.F and EvtBToKpipi.F in C++
   f) fixes to EvtBTo3hCP model via EvtCBTo3piP00 and EvtCBTo3piMPP, EvtLambdacPHH

2) New models

   a) VTOSLL
   b) EvtBcVHad, Evtbs2llGammaMNT, Evtbs2llGammaISRFSR, EvtbTosllMS, EvtbTosllMSExt, EvtLb2Baryonlnu, EvtLb2plnuLCSR, EvtLb2plnuLQCD, EvtFlatSqDalitz and EvtPhspFlatLifetime.
   c) BToDiBaryonlnupQCD, EvtBsMuMuKK (BS_MUMUKK) , EvtLambdacPHH, EvtDToKpienu
   d) add EvtBLLNuL and remove EvtB2MuMuMuNu

3) properties

   a) Update particle properties file evt.pdl to 2019 version of RPP by PDG.

(in blue, changes in 2.0, while in black changes since 1.7).