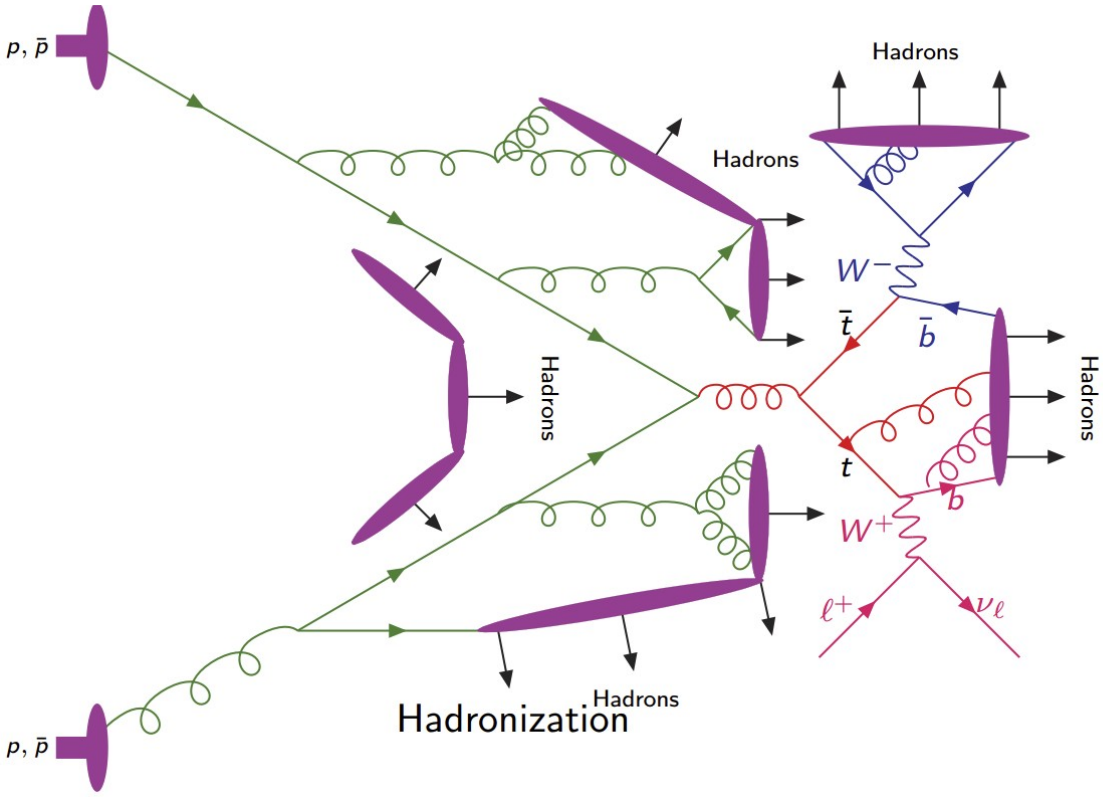
Improved Pythia-EvtGen interface

Jihyun Bhom, Marcin Chrzaszcz

CERN, FCC Software Meeting
2021-02-26

Heavy flavor simulation



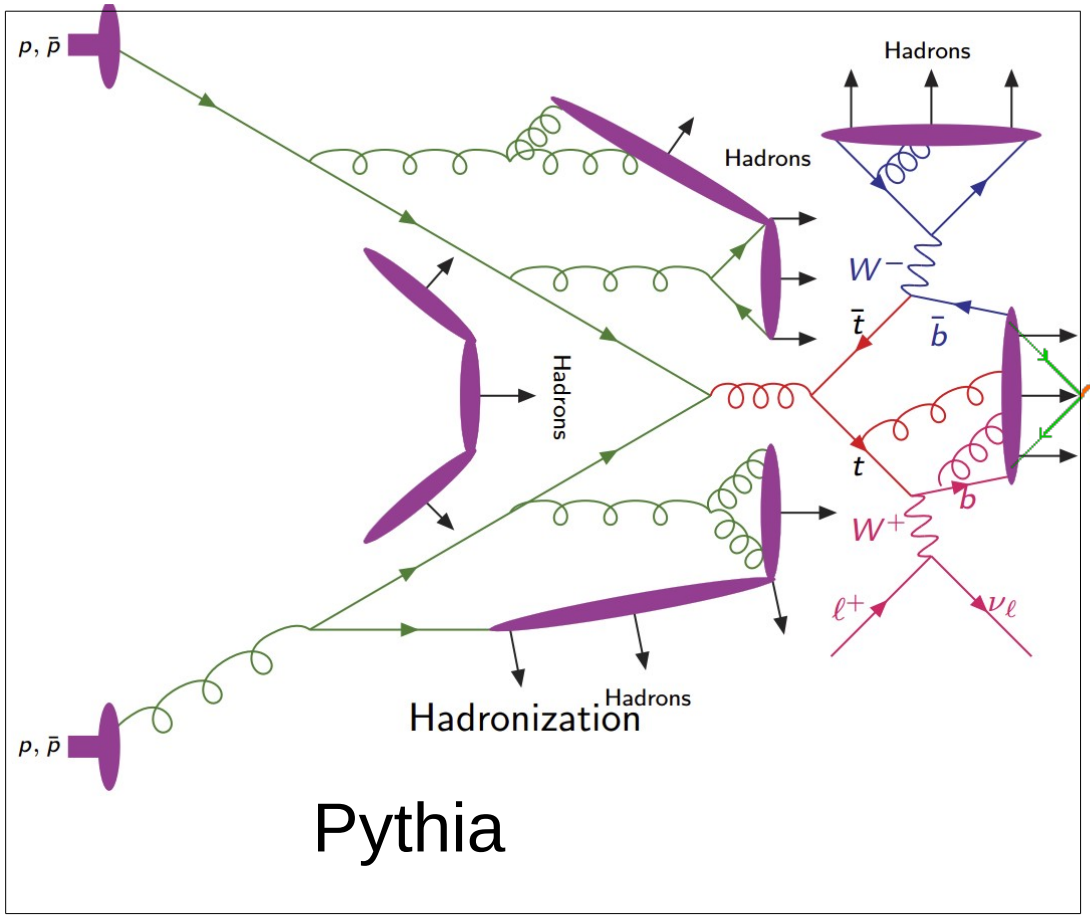
- **Pythia**

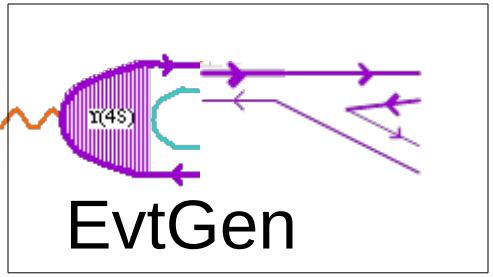
- Responsible for ISR, FSR and hadronization.
- We end up with B flavoured particles.
- Pythia has the possibility to decay then, but the simulation is limited to couple of decay channels.
- Far from state of the art.

- **EvtGen**

- Dedicated only for b-flavoured hadrons
- Constantly updated to match the PDG branching fractions.
- Dedicated models with Form factors, decay constants, etc.
- State of the art. Used in ALL Bfactories.

Heavy flavor simulation



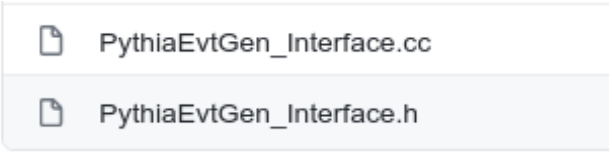


Current interface

- **In Pythia 8.3+ the interface to EvtGen was introduced. Unfortunately it has several disadvantages:**
  - It allows events with no signal to pass. Eg:

1. We are want a decay B+ → K K K. However in the hadronization process B0 B0bar was created. In that case we are storing an event on disk that has no signal inside.
2. On the other hand if in the hadronization we created B+ and B- both will decay to 3 Kaon final state (bug reported to the Pythia developers).

- **Unfortunately both "features" make this interface pretty much useless**
- **In key4hep/k4SimDelphes4hep we created our own interface that solves this issues:**



- **And one can use it:**
  ./DelphesPythia8EvtGen_EDM4HEP_k4Interface $DELPHES/cards/delphes_card_IDEAtrkCov.tcl example_Clement/edm4hep_output_config.tcl example_Clement/ee_Z_bbbar_my.cmd test.root $EVTGEN/DECAY.DEC $EVTGEN/evt.pdl example_Clement/Bc2TauNuTAUHADNU.dec 541 Bc_SIGNAL 1

Features, Usage

- **One needs to turn off B decays in pythia (example_Clement/ee_Z_bbbar_my.cmd):**

```
!WeakSingleBoson:ffbar2ffbar(s:gmZ) = on
!WeakZ0:gmZmode = 2
!23:onMode = off
!23:onIfMatch = 5 5

WeakSingleBoson:ffbar2gmZ = on
23:onMode = off
23:onIfAny = 5
! turn off hadronization
HadronLevel:all = off
! The hadronization is forced by evtgen interface

! turn off all B hadron decays
511:onMode = off
521:onMode = off
531:onMode = off
541:onMode = off
5122:onMode = off
5132:onMode = off
5142:onMode = off
5232:onMode = off
5242:onMode = off
5332:onMode = off
5342:onMode = off
5412:onMode = off
5414:onMode = off
5422:onMode = off
```

This comes with EvtGen and is based on PDG

And providing the inclusive list of decays ($EVTGEN/DECAY.DEC):

```
#
Decay anti-B0
# Updated to PDG 2008
#                  b -> c semileptonic
#
0.0493   D*+     e-      anti-nu_e      PHOTOS  HQET2 1.207 0.920 1.406 0.853; # 1.0 * PDG2014
0.0219   D+      e-      anti-nu_e      PHOTOS  HQET2 1.185 1.081;              # 1.0 * PDG2014
0.0042   D_1+    e-   anti-nu_e         PHOTOS  ISGW2;                # PDG2014+D_J below
0.0045   D_0*+   e-    anti-nu_e        PHOTOS  ISGW2;                # PDG2014+D_J below
0.0046   D'_1+   e-    anti-nu_e        PHOTOS  ISGW2;                # PDG2014+D_J below
0.0033   D_2*+   e-    anti-nu_e        PHOTOS  ISGW2;                # PDG2014+D_J below
0.00045  D*+  pi0   e-    anti-nu_e      PHOTOS  GOITY_ROBERTS;        # Increase by 1.5 compared to 2011
0.00490  D*0   pi+  e-    anti-nu_e      PHOTOS  GOITY_ROBERTS;        # Same as 2011
0.0015   D+   pi0   e-    anti-nu_e      PHOTOS  GOITY_ROBERTS;        # 1.5 * of 2011 table
```

User Input:

- **Decfile (eg. example_Clement/Bc2TauNuTAUHADNU.dec):**

```
Define TauolaCurrentOption 0
Define TauolaBR1 1.0
#
Alias Bc_SIGNAL     B_c+
Alias Bcbar_SIGNAL  B_c-
ChargeConj Bc_SIGNAL Bcbar_SIGNAL

Alias        MyTau+    tau+
Alias        MyTau-    tau-
ChargeConj    MyTau+    MyTau-
#
Decay Bc_SIGNAL
      1.0    MyTau+    nu_tau    PHSP;
Enddecay
CDecay Bcbar_SIGNAL
#
Decay MyTau-
      1.0    pi-    pi-     pi+     nu_tau                TAUHADNU -0.108 0.775 0.149 1.364 0.400 1.23 0.
Enddecay
CDecay MyTau+
#
End
```

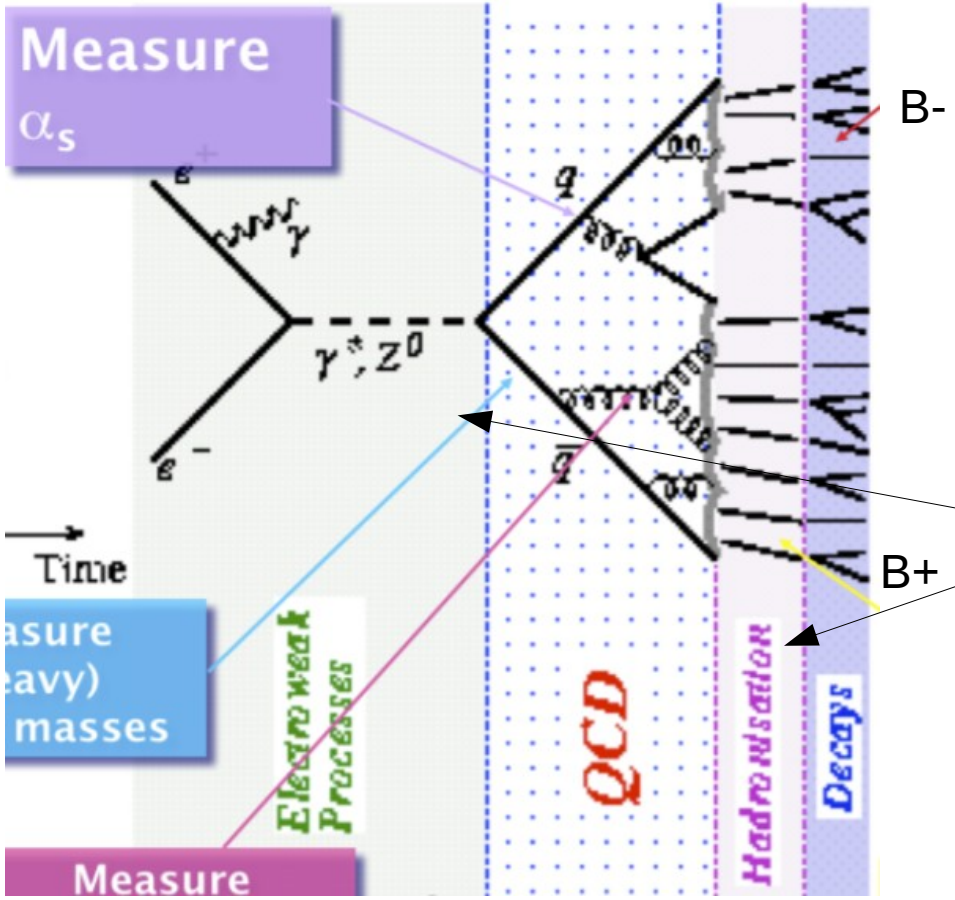
- **Last two inputs:**

541 Bc_SIGNAL 1 → Next slide

Pythia ID of Bc meson

Bc_SIGNAL is the name of particle in the user decfile

The Need for Speed :)



The new interface accepts the even from Pythia ONLY if it contains the correct species of b-flavoured hadron

If the generated event does not have the correct b-flavoured hadron:
- Repeat the hadronization (FAST)
- Regenerate whole event (“1” from previous slide) - SLOW

The slow option is needed for “rare” hadrons (eg. Bc). The Bc cannot be produced in a “standard hadronization”. The c quark needs to come from the hard part.

Many THANKS to Clement for testing this!!!

Conclusion

- The in-house FCC Pythia-EvtGen interface we solved the issues with the build one in Pythia
- We have a possibility to speed up the simulation by “re simulating” the hadronizatio.
- Interface is more flexible for many signal production (eg. we can ask B+ and B- to decay to specific final state).

The interface can be already found in the k4SimDelphes:

https://github.com/key4hep/k4SimDelphes

Many thanks to Clement, Gerardo, Valentin and Thomas for all the help getting this done


THANK YOU