Sharing the workload in collider Monte Carlo simulations

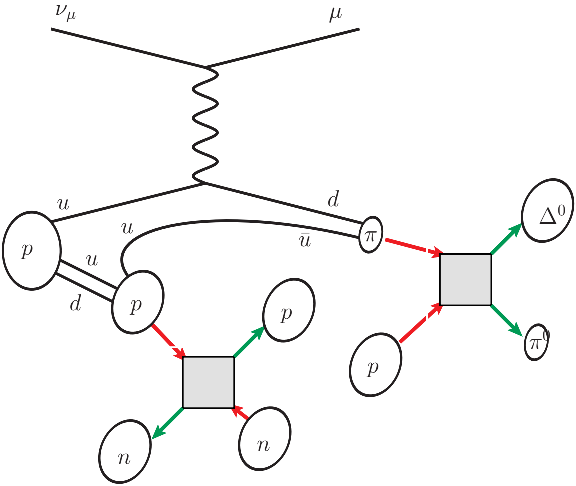
Neutrino + Theory WG, FNAL

October 25, 2017

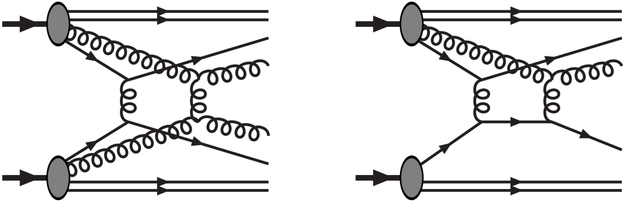
Stefan Prestel, John Campbell (Fermilab)

input from Nadine Fischer, Holger Schulz, Johannes Bellm

My naive view of neutrino scattering



arXiv:0911.1909 [hep]



(a) (b)

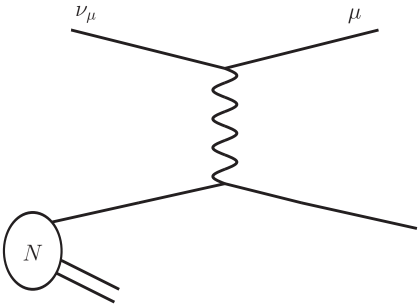
Figure 1: (a) Two $2 \to 2$ scatterings, (b) a $2 \to 2$ scattering followed by a rescattering

...where we face similar problems, but with partons.

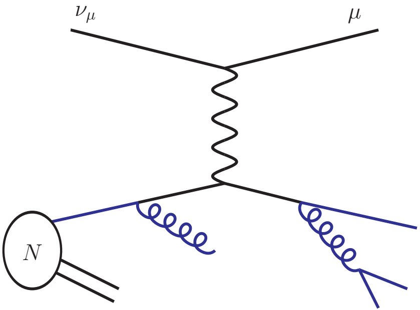
Collider event generator development is a theorist's playground:

- ... lots of perturbation theory
  Tool chains $\mathcal{L}$(my fav. model)$\rightarrow \sigma \rightarrow$ detector
  High precision for parton distribution functions (PDFs), cross sections;
- ... lots of non-perturbative modelling
  for multiple interactions between nucleon constituents
  to convert partons to hadrons;
- + lots of data to beat down uncertainties in non-perturbative modelling;
- + **lots of software tools.**

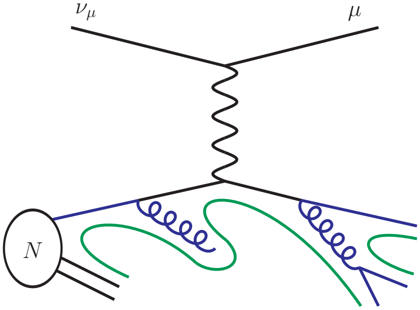
...but is probably less challenging than neutrino event generation.



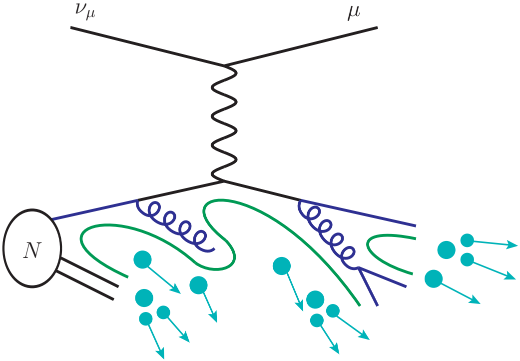
Start with hard scattering of partons...
usually multi-parton states with one/two loops
$\rightarrow$ complicated, lots of theory, software & interfacing.



Produce radiative cascade (quarks/gluons/photons...)...

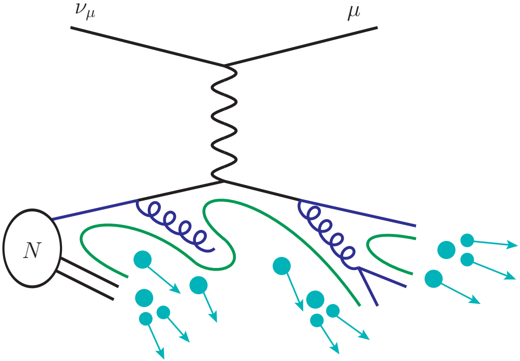


Form proto-hadrons (colour strings, colour clusters)...
Not shown: Multiple interactions, parton rescattering, diffractions, since usually handled internally (exception: DIPSY code)

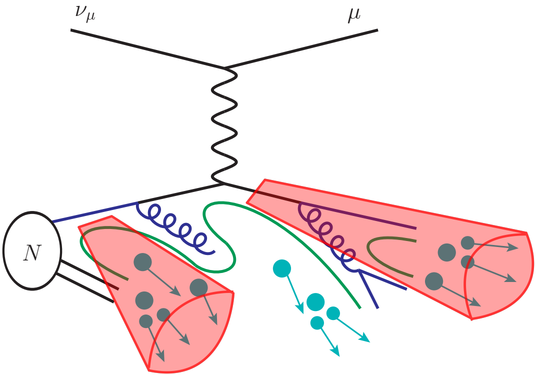


…and decay to primary hadron (resonances) and secondary hadrons/photons et cetera, including e.g. hadron rescattering

A word on calculability



Fact: Distribution of hadrons in detector not calculable in QCD.
If LHC would rely on multiplicities alone, progress would be hopeless.



Fact: Distribution of hadrons in detector not calculable in QCD.

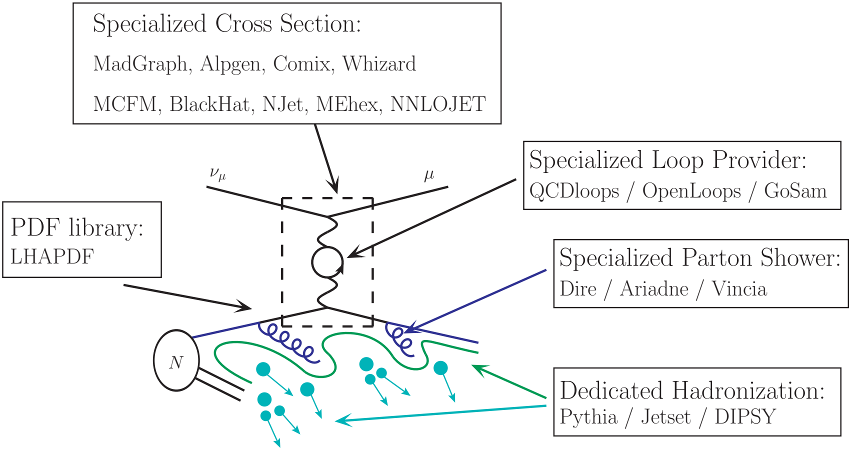
Claim: LHC is only successful because we use “safe” observables:

Coarse-grained and averaged, i.e. jets.

Event generation is broadly factorized into

- Precision hard scattering cross section (most "hard-core theory" $\rightarrow$ John, Walter)
- Other event generation (infrared physics, non-perturbative aspects $\rightarrow$ Steve, Stefan)
- Analysis object (jet) definition, analysis and/or detector simulation (analysis $\rightarrow$ Holger Schulz)

Factorized of collider event generation



Jet Definition:	Analysis:	Detector Simulation:
FastJet	Rivet	Geant / Delphes

More factorized – i.e. more codes contributing – where it matters most (precision hard scattering)

What do we interface? How do we interface?

	Downstream	*Upstream*
Hard cross section	PDF lib, loop integrals	parton shower
Parton shower	PDF lib, hard cross section	MPI, hadronization
Hadronization	parton shower, MPI	analysis, detector sim.
Analysis	data repo^s, ob^s def. tools	results

Need interfaces that are generic and allow cross-talk. We mostly use

- File-based interfaces
- Run-time interfaces

What do we interface? How do we interface?

	Downstream	*Upstream*
Hard cross section	PDF lib, loop integrals	parton shower
Parton shower	PDF lib, hard cross section	MPI, hadronization
Hadronization	parton shower, MPI	analysis, detector sim.
Analysis	data repo^s, ob^s def. tools	results

Need interfaces that are generic and allow cross-talk. We mostly use

- File-based interfaces
- Run-time interfaces

	Downstream	*Upstream*
Hard cross section	PDF lib, loop integrals	parton shower
Parton shower	PDF lib, hard cross section	MPI, hadronization
Hadronization	parton shower, MPI	analysis, detector sim.
Analysis	data repo^s, ob^s def. tools	results

Need interfaces that are generic and allow cross-talk. We mostly use

- File-based interfaces
- Run-time interfaces

What do we interface? How do we interface?

	Downstream	*Upstream*
Hard cross section	PDF lib, loop integrals	parton shower
Parton shower	PDF lib, hard cross section	MPI, hadronization
Hadronization	parton shower, MPI	analysis, detector sim.
Analysis	data repo^s, ob^s def. tools	results

Need interfaces that are generic and allow cross-talk. We mostly use

- File-based interfaces
- Run-time interfaces

File-based interfaces

arXiv:hep-ph/0311123, arXiv:0801.0045

- Exchange parameters, particle id's and four-vectors
- Many providers, many users → Avoid specialized interfaces.
- Computationally precious intermediate results.
- Interface: One centralized reader/writer for few languages.

```
###################################
## INFORMATION FOR MASS
###################################
Block mass
    6 1.732000e+02 # MT
   15 1.777000e+00 # MTA
   23 9.118760e+01 # MZ
   24 8.039800e+01 # MW
   25 1.200000e+02 # MH
   82 0.000000 # gh : 0.0

###################################
## INFORMATION FOR SMINPUTS
###################################
Block sminputs
    1 1.323384e+02 # aEWM1
    3 1.180000e-01 # aS

###################################
## INFORMATION FOR DECAY
###################################
DECAY   6 1.501700e+00 # WT Gamma_t_LO
DECAY  23 2.495200e+00 # WZ
DECAY  24 0.000000e+00 # WW
DECAY  25 5.753088e-03 # WH
DECAY  82 0.000000 # gh : 0.0

#===========================================================
# QUANTUM NUMBERS OF NEW STATE(S) (NON SM PDG CODE)
#===========================================================
Block QNUMBERS 82  # gh
        1 0  # 3 times electric charge
        2 1  # number of spin states (2S+1)
        3 8  # colour rep (1: singlet, 3: triplet, 8: octet)
        4 1  # Particle/Antiparticle distinction (0=own anti)
```

e.g. Susy Les Houches Accord files (SLHA, generically for BSM models)

File-based interfaces

arXiv:hep-ph/0109068, arXiv:hep-ph/0609017, arXiv:1405.1067

e.g. exchange four-momenta, flavours, cross sections, uncertainties for hard scattering with Les Houches Event Files (LHEF) exploiting *Cross section* $=$ *sum over events*.

```
<event>
5     66 0.50109093E+02 0.14137688E+03 0.75563862E-02 0.12114027E+00
        5 -1    0    0  501    0 0.00000000E+00 0.00000000E+00 0.14322906E+03 0.14330946E+03 0.48000000E+01 0.0000E+00 0.0000E+00
        2 -1    0    0  502    0 0.00000000E+00 0.00000000E+00 -.93544317E+03 0.93544323E+03 0.33000000E+00 0.0000E+00 0.0000E+00
       24  1    1    2    0    0 -.84258804E+02 -.15708566E+03 -.10629600E+03 0.22257162E+03 0.80398000E+02 0.0000E+00 0.0000E+00
        5  1    1    2  501    0 -.13668073E+03 -.36307424E+02 -.40614473E+02 0.14721558E+03 0.48000000E+01 0.0000E+00 0.0000E+00
        1  1    1    2  502    0 0.22093954E+03 0.19339308E+03 -.64530364E+03 0.70896548E+03 0.33000000E+00 0.0000E+00 0.0000E+00
<rwgt>
 <wgt id="1001"> 0.50109E+02 </wgt>
 <wgt id="1002"> 0.45746E+02 </wgt>
 <wgt id="1003"> 0.52581E+02 </wgt>
 <wgt id="1004"> 0.50109E+02 </wgt>
 <wgt id="1005"> 0.45746E+02 </wgt>
 <wgt id="1006"> 0.52581E+02 </wgt>
 <wgt id="1007"> 0.50109E+02 </wgt>
 <wgt id="1008"> 0.45746E+02 </wgt>
 <wgt id="1009"> 0.52581E+02 </wgt>
</rwgt>
<weights> 1.000e+00 0.204e+00 1.564e+00 </weights>
<scales muf="90.1" mur="90.2" mups="90.3" newscale="90.4"> random stuff </scales>
</event>
```

Often used between hard scattering cross section generator (MadGraph) and event generator (Pythia) Not possible if up- or downstream code cannot be run stand-alone.

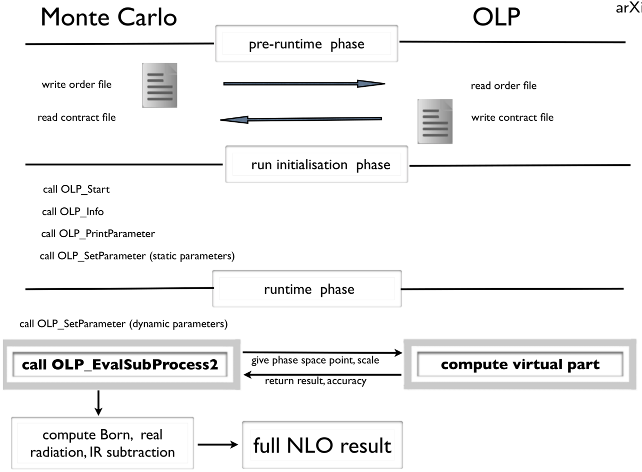
Comput.Phys.Commun. 134 (2001) 41-46

e.g. store four-momenta, flavours, cross sections, uncertainties after event generation in HepMC event files *before* detector simulation & for analysis prototyping

```
HepMC::Version 2.06.09
HepMC::IO_GenEvent-START_EVENT_LISTING
E 0 -1 -1.0000000000000000e+00 -1.0000000000000000e+00 -1.0000000000000000e+00 0 0 8 1 2 0 1 5.0727818584935023e-16
N 1 "0"
U GEV MM
C 5.0727818584935028e-07 2.8641748972593421e-02
V -1 0 0 0 0 0 1 1 0
P 1 14 -6.1062266354383610e-16 0 1.0000000000000000e+01 1.0000000000000002e+01 0 4 0 0 -1 0
P 3 14 -6.1062266354383610e-16 0 1.0000000000000000e+01 1.0000000000000002e+01 0 21 0 0 -3 0
V -2 0 0 0 0 0 0 1 0
P 4 1 1.1102230246251565e-16 2.7755575615628914e-17 -3.3786624711333191e-01 3.3786624711333263e-01 0 21 0 0 -3 1 1 101
V -3 0 0 0 0 0 0 2 0
P 5 13 1.5436629084424718e+00 -1.4642943199919159e-01 7.4852688365925690e+00 7.6449166342819295e+00 1.0566000000000000e-01 1 0 0 0 0
P 6 2 -1.5436629084424731e+00 1.4642943199919156e-01 2.1768649162940950e+00 2.6929496128313986e+00 3.3000000000000002e-01 23 0 0 -5 1 1 101
V -4 0 0 0 0 0 1 2 0
P 2 2212 -2.2204460492503131e-16 5.5511151231257827e-17 -3.4590375409931440e-01 1.0000000000000011e+00 9.3827000000000005e-01 4 0 0 -4 0
P 7 1 2.3418389329410927e-01 3.8065298354292854e-01 -5.8804961543564516e-02 4.5077357205215524e-01 0 61 0 0 -2 1 1 101
P 9 2203 -4.7481444848931020e-01 -3.1949722919432438e-01 1.1083686479487653e-01 9.6683091564896451e-01 7.7132999999999996e-01 63 0 0 -6 1 2 101
V -5 0 0 0 0 0 0 1 0
P 8 2 -1.0688484599531634e+00 4.6592666119351611e-01 2.0579905445132347e+00 2.3882524500691016e+00 3.3000000000000002e-01 62 0 0 -6 1 1 101
V -6 0 0 0 0 0 0 2 0
P 10 211 -1.0645551661431243e+00 -8.9555571034242387e-02 1.7567327618510375e+00 2.0607978269413185e+00 1.3957000000000000e-01 1 0 0 0 0
P 11 2214 -4.7910774229934994e-01 2.3598500303343378e-01 4.1209464745707314e-01 1.2942855387767489e+00 1.1045903797498771e+00 2 0 0 -7 0
V -7 0 0 0 0 0 0 2 0
P 12 2212 -3.9016256981713637e-01 2.3432010072537812e-01 4.3706067676524718e-01 1.1307101169463198e+00 9.3827000000000005e-01 1 0 0 0 0
P 13 111 -8.8945172482213411e-02 1.6649023080556186e-03 -2.4966029308174188e-02 1.6357542183042872e-01 1.3497999999999999e-01 2 0 0 -8 0
V -8 0 -1.6789351900021416e-06 3.1426810414801768e-08 -4.7126048542432794e-07 3.0876609068971146e-06 0 2 0
P 14 22 1.3119240933915144e-02 -4.5803387936032995e-02 2.4978883742856808e-03 4.7710630633342535e-02 0 1 0 0 0 0
P 15 22 -1.0206441341612857e-01 4.7468290244088614e-02 -2.7463917682459871e-02 1.1586479119708619e-01 0 1 0 0 0 0
```

Runtime interfaces

arXiv:1308.3462 [hep]



... necessary when software not stand-alone (e.g. if divergent)

... to avoid large files

e.g. interface between "one-loop providers" and x-section generators via Binoth Les Houches Accord (BLHA)

How to get there?

- Prepare a wishlist.
- Lock relevant experimenters & theorists away in the Alps for 2 weeks. Fuel with cheese & wine.
- Discussions on site, try to agree, enforce detailed proceedings afterwards.
- Implement, rethink, repeat after two years.

⇒ Les Houches Accords

Other run-time interface models (Pythia bias!)

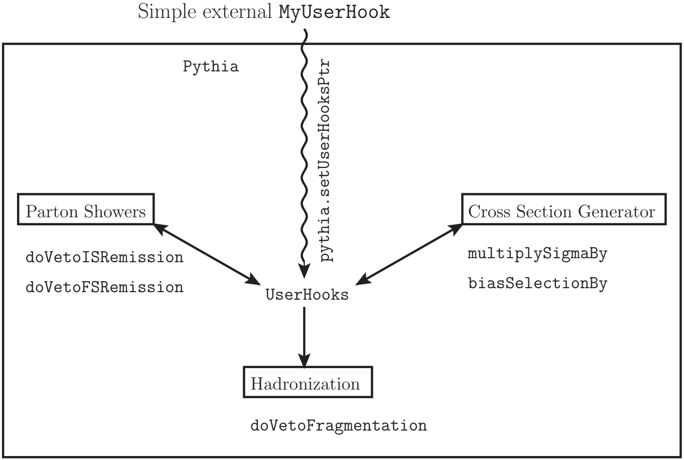
Most LHC code is C++ ⟶ Use power of common language!
Pythia 8: Allow to inherit from & replace most physics modules.

```

class DireTimes : public Pythia8::TimeShower {
  // some code
  virtual double pTnext(Pythia8::Event&,double,do
  // some more code
};

class DireSpace : public Pythia8::SpaceShower {
  // some code
  virtual double pTnext(Pythia8::Event&,double,do
  // some more code
};
```

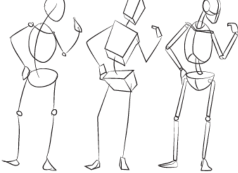
```
// DIRE includes.
#include "DireTimes.h"
#include "DireSpace.h"
// Pythia includes.
#include "Pythia8/Pythia.h"
using namespace Pythia8;

// Example main programm.
int main(){
  // Declare Pythia generator.
  Pythia pythia;
  // Declare new showers.
  DireTimes times(&pythia);
  DireSpace space(&pythia);
  DireTimes timesDec(&pythia);
  // Feed new showers to Pythia.
  pythia.setShowerPtr( &timesDec, &times, &space);
  // Initialize.
  pythia.init();
  // Produce an event, using the new showers.
  pythia.next();
  // Done
  return 0;
}
```

- Mostly used for BSM cross sections and new showers.
- Rather high threshold, but gives extreme flexibility.

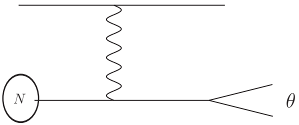
`UserHooks` model: Allow user to overwrite important decisions
→ Use power of statistics to bend probabities/rates to your will:
Strawman implementation + easy external user re-weighting!



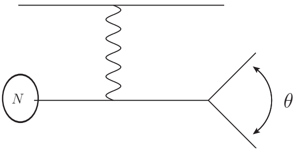
Other run-time interface models (Pythia bias!)



`UserHooks` model: Allow user to overwrite important decisions
$\rightarrow$ Use power of statistics to bend probabities/rates to your will:
Strawman implementation + easy external user re-weighting!



Produce splitting with rate $p = 1/\theta^4$

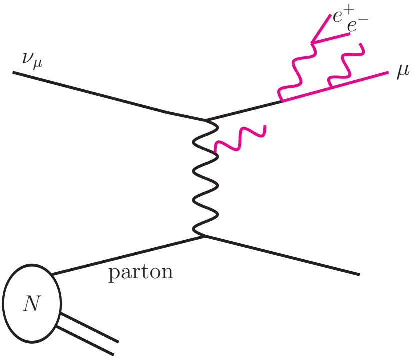


Overwrite `UserHooks::doVetoFSREmission`:
to reject emission with probability $\frac{1/\theta^2}{p}$
$\Rightarrow$ Average angle becomes larger. Rate now given by $p_{new} = 1/\theta^2$
$\Rightarrow$ Replaced Pythia's rates by your own calculation

Crucial for LHC: **Define** data objects that are "insensitive"

- ... to phenomena we do not understand well enough,
- ... to corrections we cannot calculate well enough.

Analysis objects: What's a muon anyway?

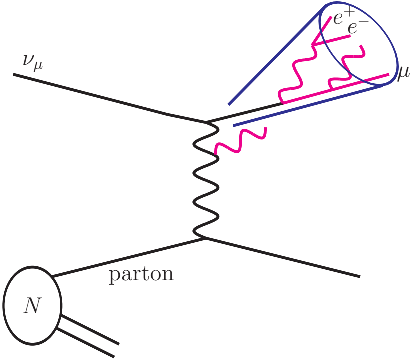


Charged particles always come with a photon cloud

...or with even more charged particles.

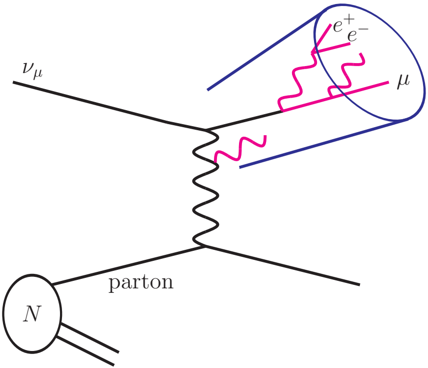
So how should we "define" what we mean by e.g. "muon"?

Analysis objects: What's a muon anyway?



Need to combine charged particle + photon momenta to extract ν energy loss / momentum that probes the nucleus!
But how? Should we use small cones?

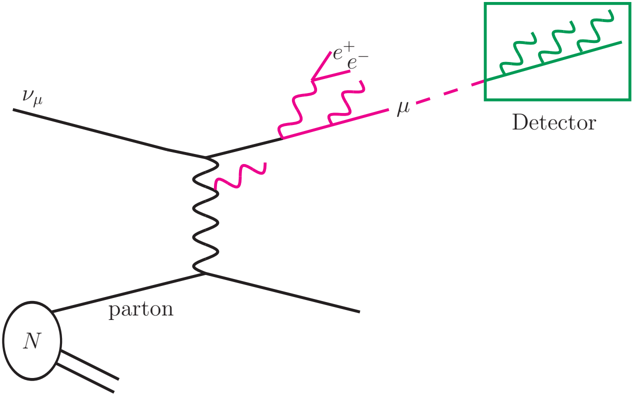
Analysis objects: What's a muon anyway?



Need to combine charged particle + photon momenta to extract ν energy loss / momentum that probes the nucleus!

But how? Should we use small cones or **big cones**?

Analysis objects: What's a muon anyway?



Can "vacuum radiation" and detector smearing / transition radiation ever be separated? Is it okay to correct for one, but not the other?

Theory:

- Make event generators public and linked at a common resource (see e.g. `hepforge.org`)
- Decouple analysis from event generation.
- Make analysis prototype code public (see e.g. RIVET)

Experiment:

- Agree on common analysis objects ("hey, let's all use infrared-insensitive jets and QED-dressed leptons")
- Make data public (see e.g. `opendata.cern.ch`, `hepdata.net`)
- Make analysis code public (see e.g. RIVET)

Easy data & analysis access $\Rightarrow$ Theory+software progress

- Should certainly make sure that LHC event generators work, at a technical level, together with neutrino generators
- Can supply detailed DIS-like calculations to help define "safe" observables (easy example in the talk: QED corrections).
- Can supply analysis tools. Example: Rivet (LHC) inspired by H1/Zeus (HERA) efforts.

Question: Is it reasonable to first want to get DIS right, then nucleon scattering, and learn while doing that, before moving to the nucleus?