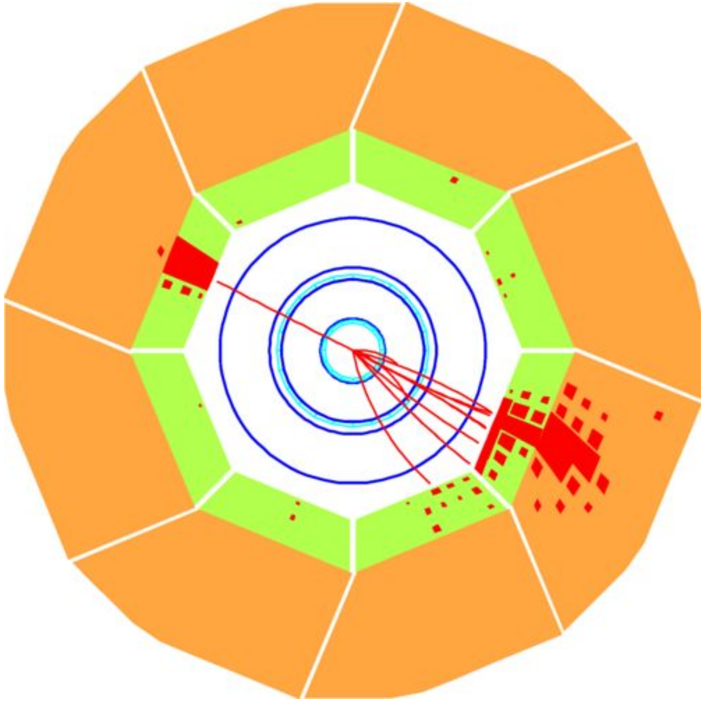
Jet and heavy flavour at the EIC

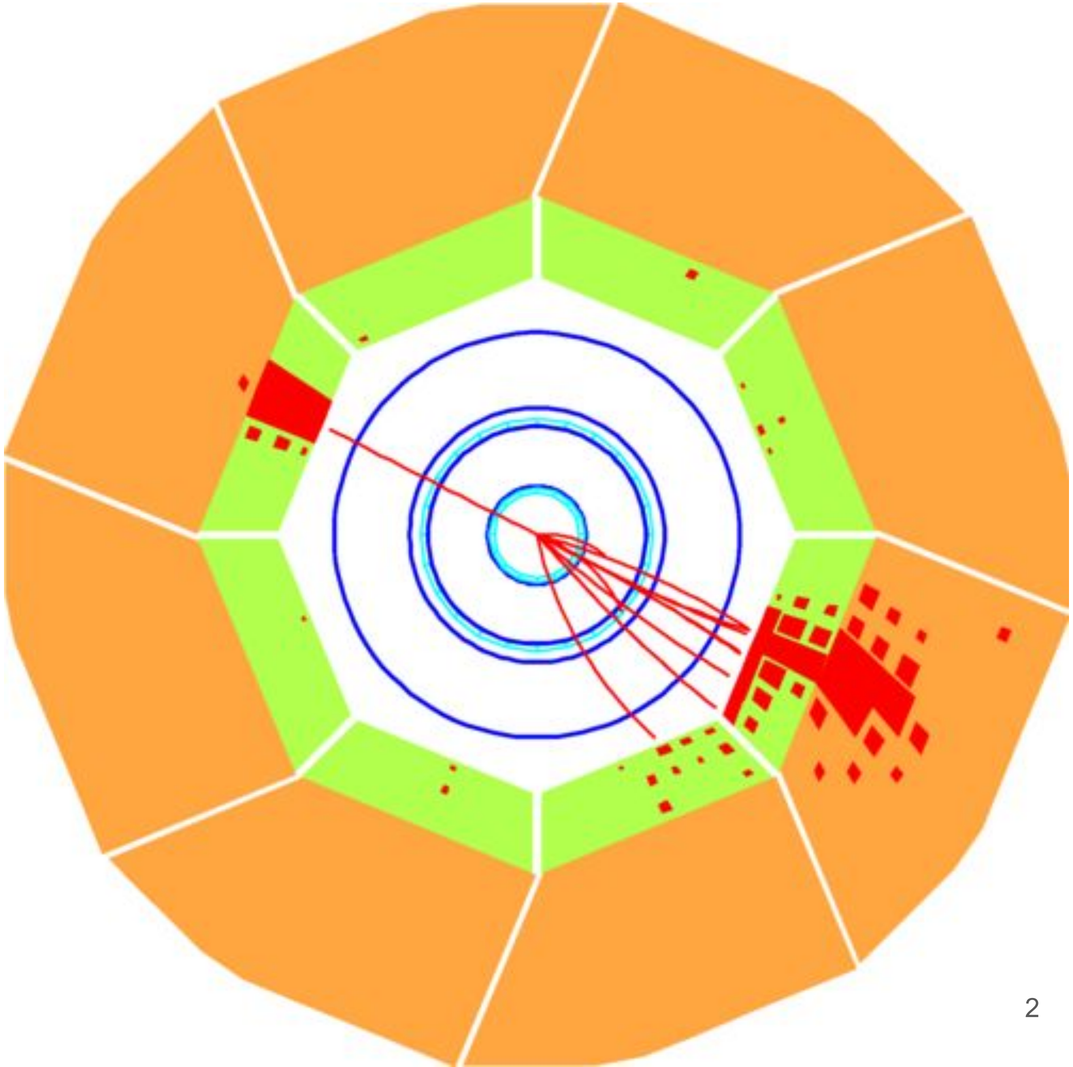
CFNS EIC Summer School 2022

Miguel Arratia

UC RIVERSIDE

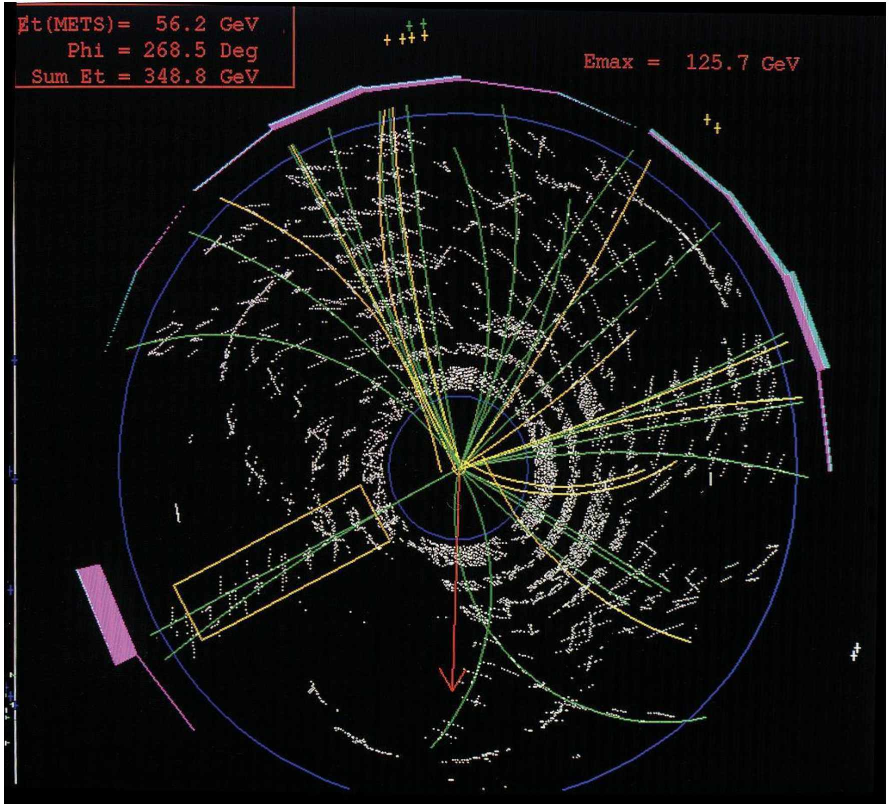


What is a jet?

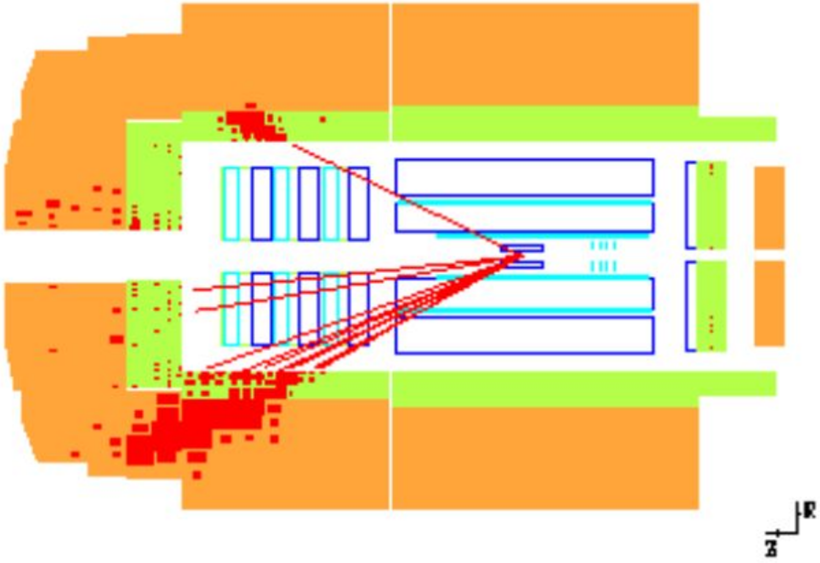


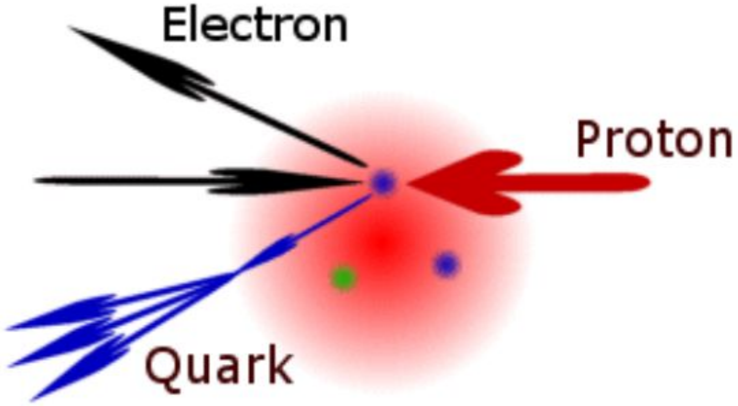
"A jet is a <u>narrow cone of hadrons</u> and other particles produced by the hadronization of a quark or gluon in a particle physics or heavy ion experiment."

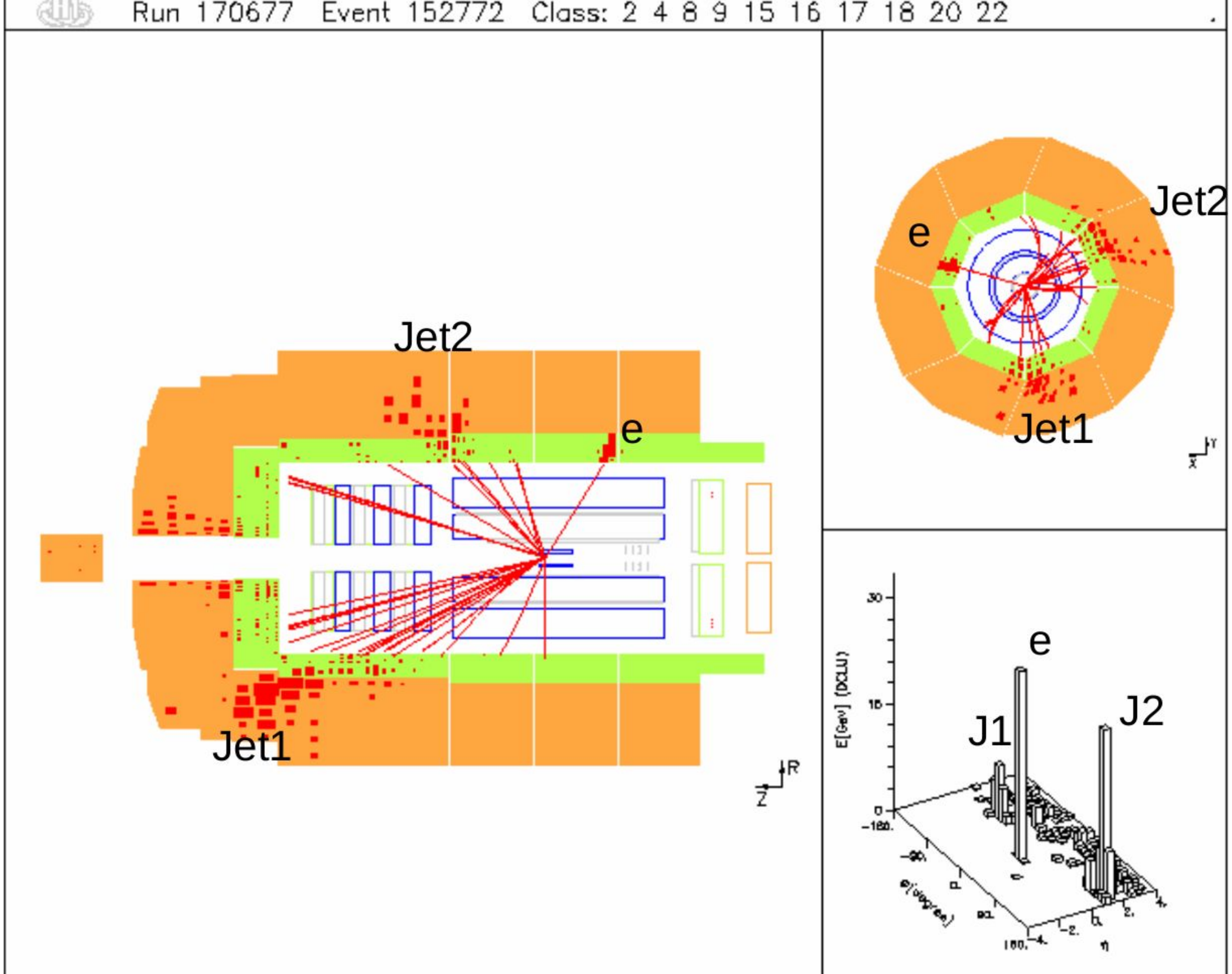
Wikipedia



Sometimes “*narrow cone of hadrons*” seems intuitive:

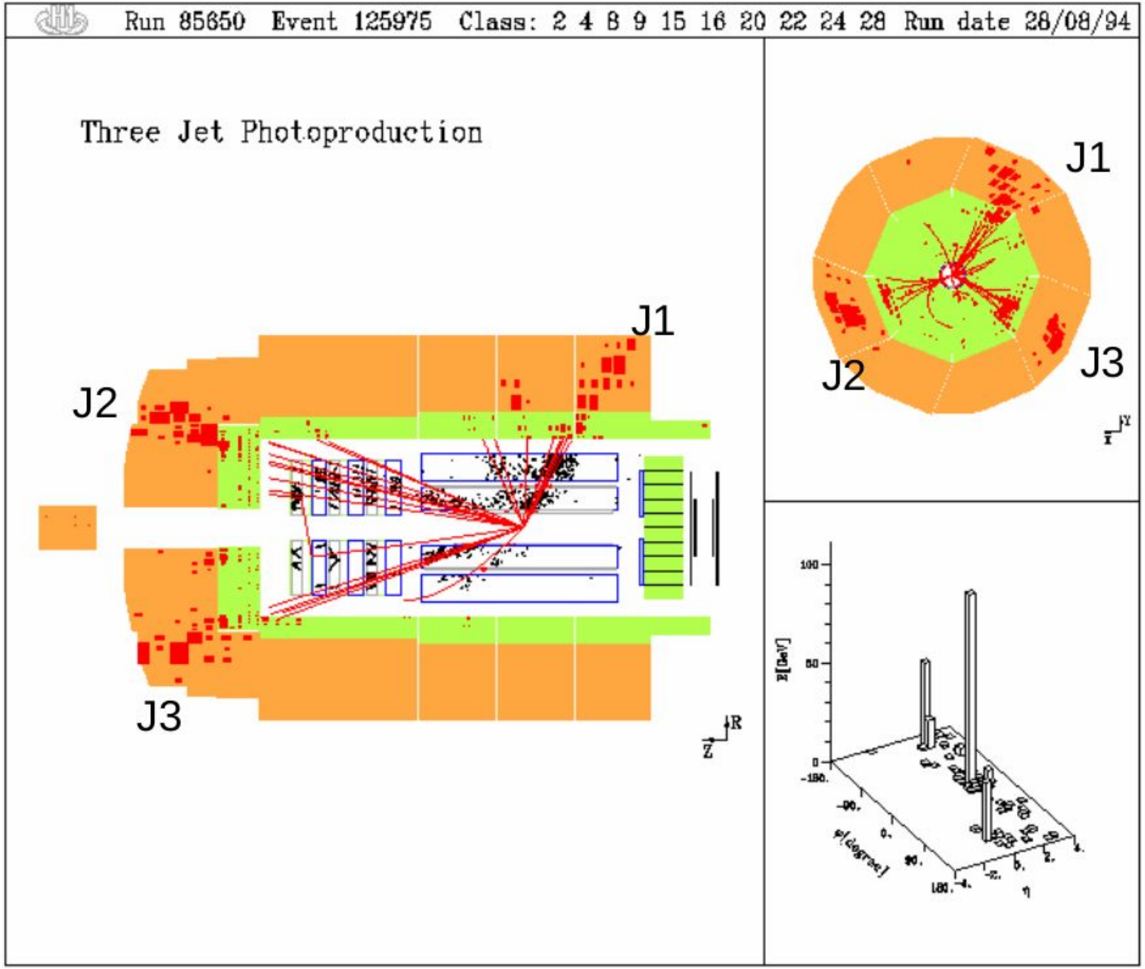






Run 170677 Event 152772 Class: 2 4 8 9 15 16 17 18 20 22
Jet2
e
Jet1
Jet2
e
Jet1
e
J2
J1
E[GeV] (DCLU)

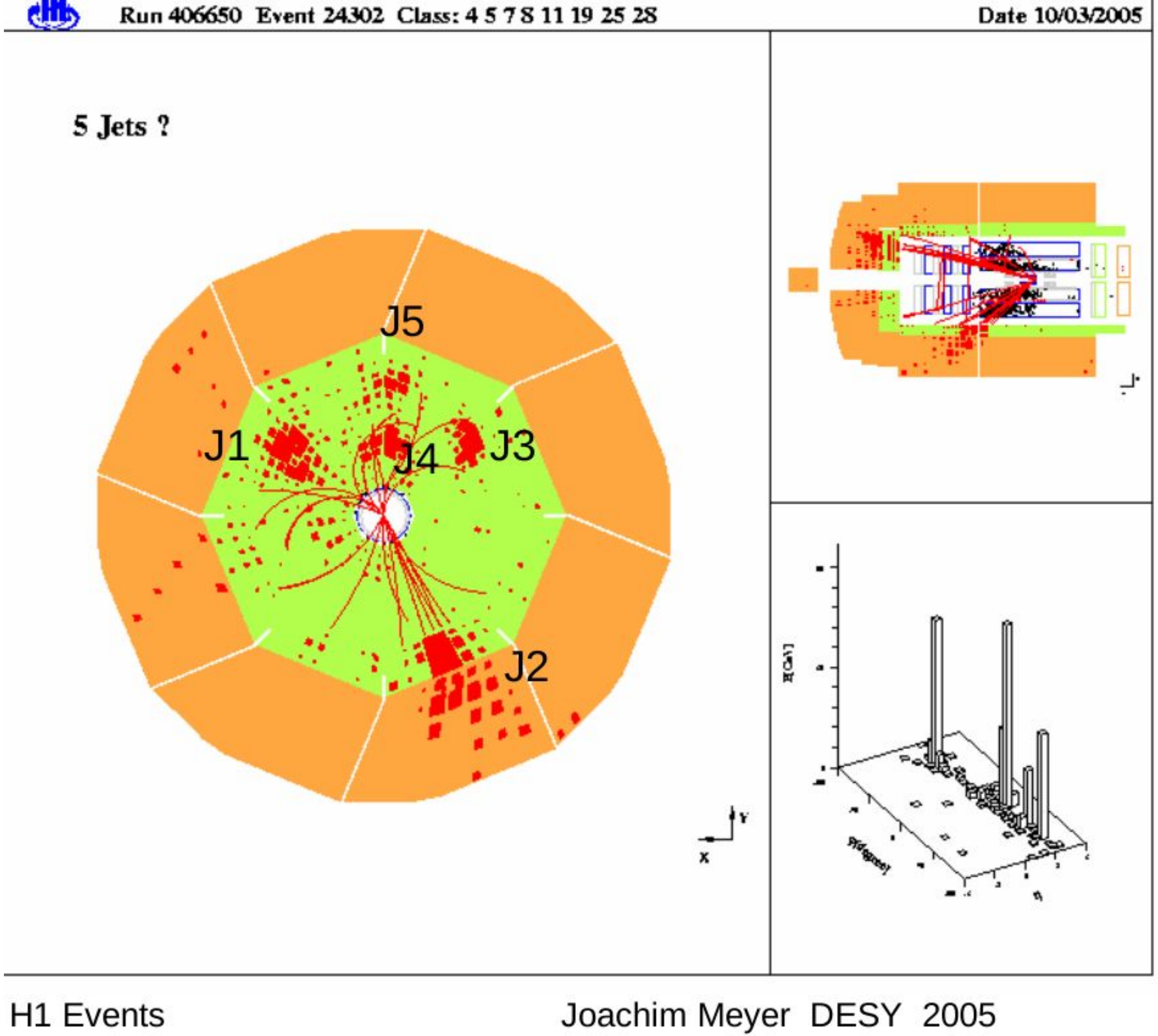
Sometimes such
A definition is a bit
More tricky

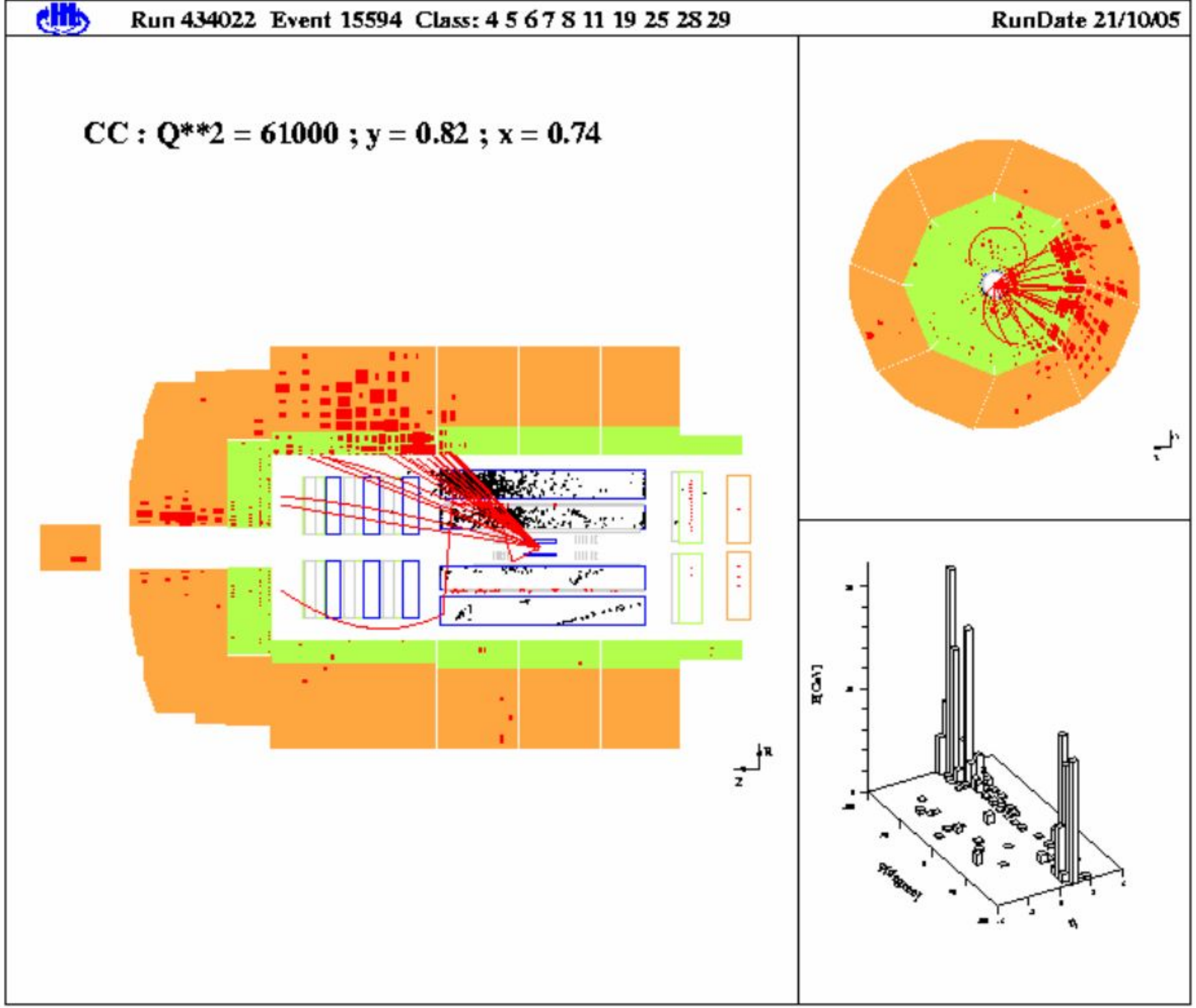


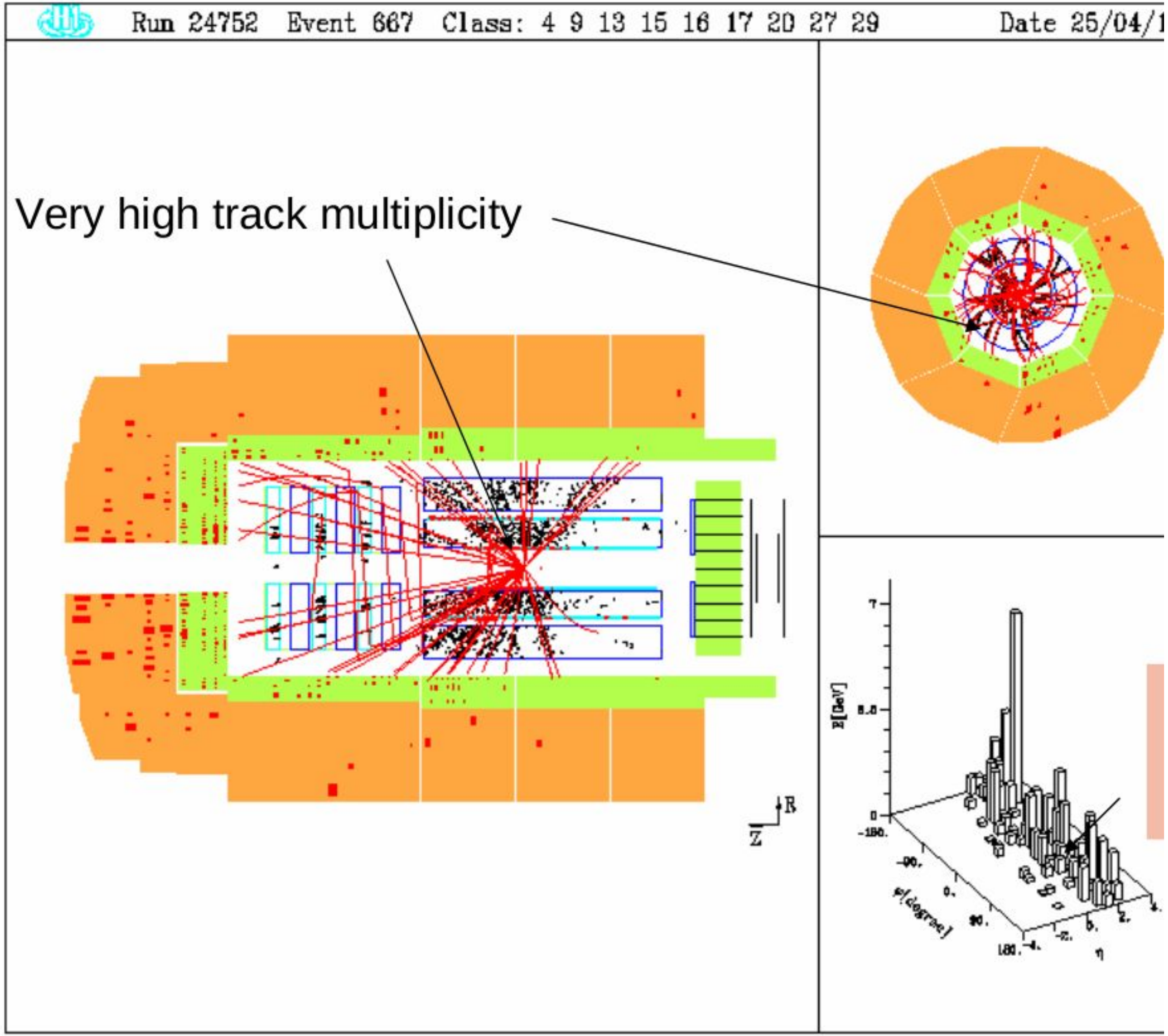
H1 Events

Joachim Meyer DESY 2005

Sometimes even more tricky

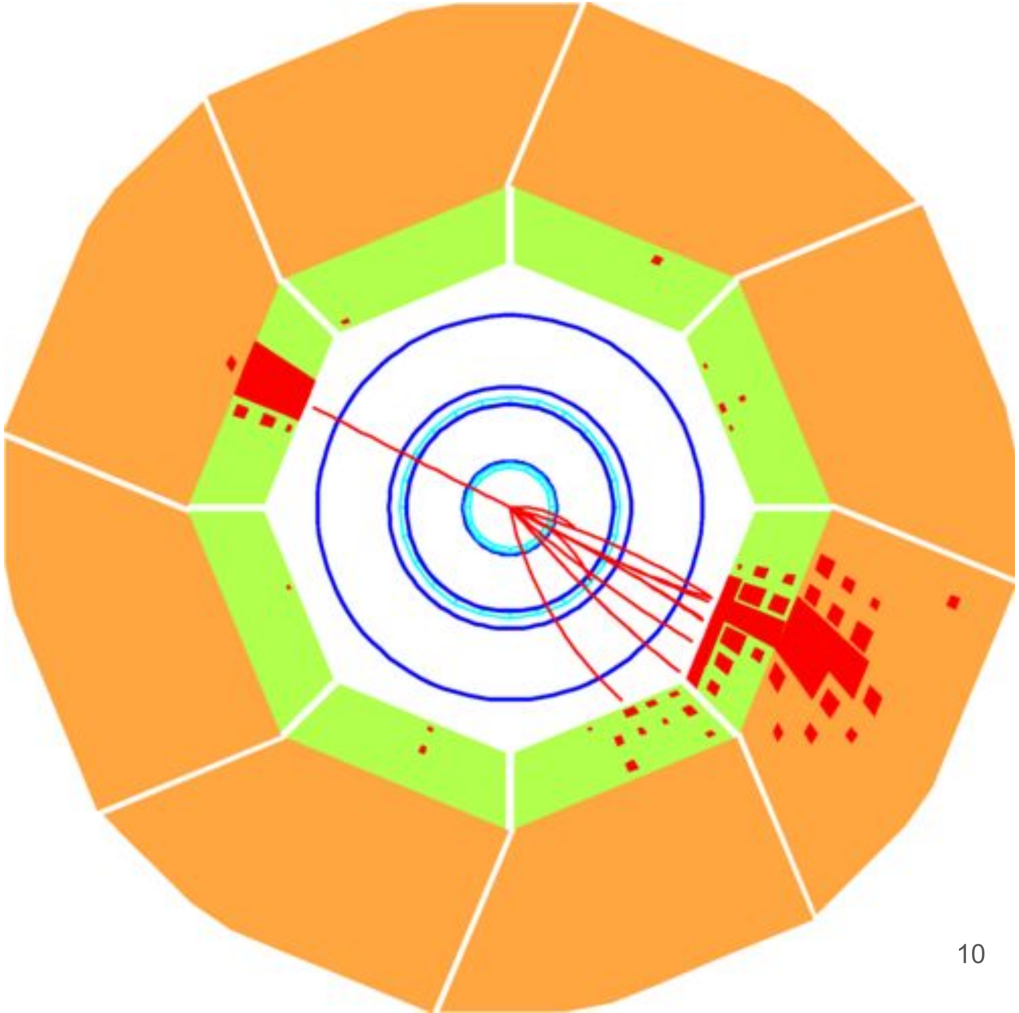



Run 434022 Event 15594 Class: 4 5 6 7 8 11 19 25 28 29
RunDate 21/10/05
CC : Q**2 = 61000 ; y = 0.82 ; x = 0.74


Run 24752 Event 667 Class: 4 9 13 15 16 17 20 27 29
Date 25/04/1
Very high track multiplicity
R
Z
E[GeV]
φ[degree]
η

What is a jet?

The output of a jet algorithm



Why are jets algorithms useful?

- Link theory and in experiment.
- Good jet algorithms have nice theory properties
  (like “infrared and collinear safety”)
- Good jet algorithms get you closer to “parton level”
  (minimize “hadronization corrections” and sensitivity to “underlying event”)
- Good jet algorithms work well in the real world of experiments
  (Nice smooth conical shapes, robust to noise and pileup, easy to calibrate)

“Jet algorithms” started in 1975

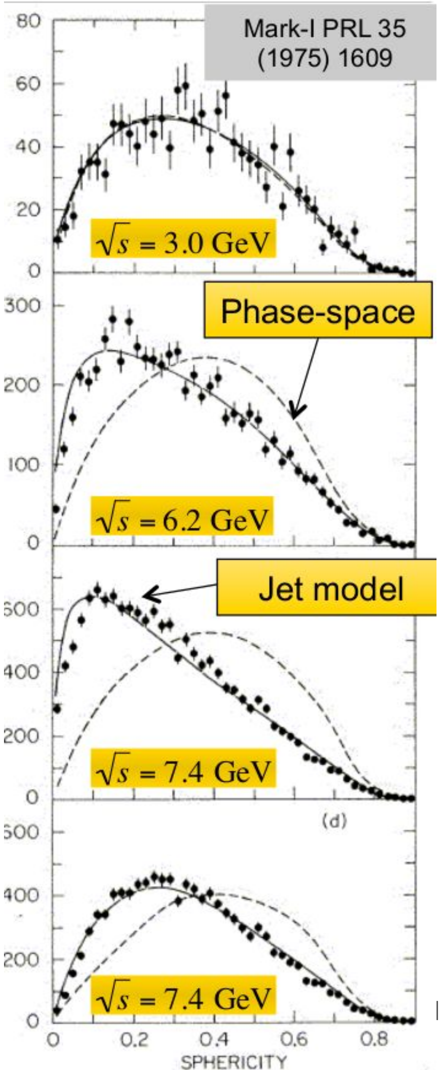
Evidence for Jet Structure in Hadron Production by e^+e^- Annihilation

G. Hanson *et al.*
Phys. Rev. Lett. **35**, 1609 – Published 15 December 1975

$$S = \frac{3\left(\sum_i p_{T,i}^2\right)}{2\left(\sum_i p_i^2\right)}$$

“Event shape” variables

Tha define “Pencil” vs “sphere” like events



After even shapes, the
Next big-thing were “Cone algorithms”,

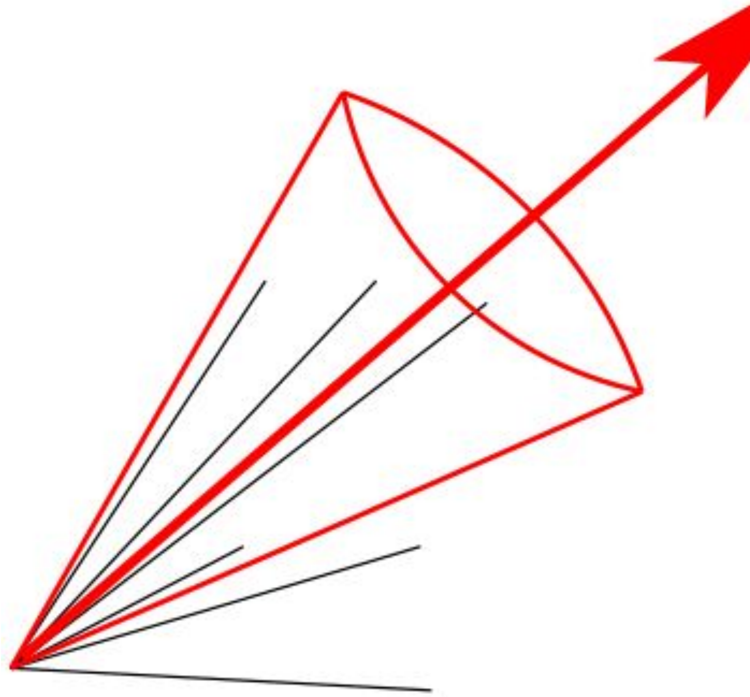


Figure by G. Salam

But the problem of finding "stable cones" was found to be pretty complicated, specially in real world

Moreover, conical algos and their tweaks not always had nice theory properties.

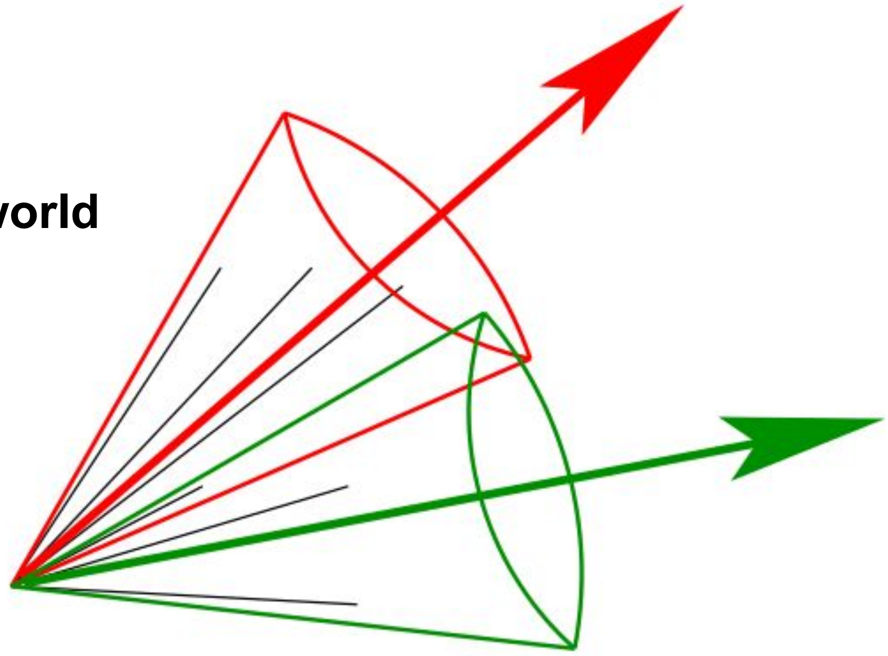


Figure by G. Salam

“Cone algorithms” got pretty complicated over time...

SISCone part 2: finding stable cones

For any group of collinear particles, merge them into a single particle.
for particle $i = 1 \ldots N$ **do**
  Find all particles j within a distance $2R$ of i. If there are no such particles, i forms a stable cone of its own.
  Otherwise for each j identify the two circles for which i and j lie on the circumference. For each circle, compute the angle of its centre C relative to i, $\zeta = \arctan \frac{\Delta\phi_{iC}}{\Delta y_{iC}}$.
  Sort the circles into increasing angle ζ.
  Take the first circle in this order, and call it the current circle. Calculate the total momentum and checkxor for the cones that it defines. Consider all 4 permutations of edge points being included or excluded. Call these the “current cones”.
  repeat
    for each of the 4 current cones **do**
      If this cone has not yet been found, add it to the list of distinct cones.
      If this cone has not yet been labelled as unstable, establish if the in/out status of the edge particles (with respect to the cone momentum axis) is the same as when defining the cone; if it is not, label the cone as unstable.
    end for
    Move to the next circle in order. It differs from the previous one either by a particle entering the circle, or one leaving the circle. Calculate the momentum for the new circle and corresponding new current cones by adding (or removing) the momentum of the particle that has entered (left); the checkxor can be updated by XORing with the label of that particle.
  until all circles considered.
end for
for each of the cones not labelled as unstable **do**
  Explicitly check its stability, and if it is stable, add it to the list of stable cones (protojets).
end for

SISCone part 3: split–merge

1: **repeat**

Remove all protojets with $p_t < p_{t,\min}$.

Identify the protojet (i) with the highest $\tilde{p}_t$ ($\tilde{p}_{t,\mathrm{jet}} = \sum_{i\in\mathrm{jet}} |p_{t,i}|$).

Among the remaining protojets identify the one (j) with highest $\tilde{p}_t$ that shares particles (overlaps) with i.

5: **if** there is such an overlapping jet **then**

6: Determine the total $\tilde{p}_{t,\mathrm{shared}} = \sum_{k\in i\&j} |p_{t,k}|$ of the particles shared between i and j.

7: **if** $\tilde{p}_{t,\mathrm{shared}} < f\,\tilde{p}_{t,j}$ **then**

Each particle that is shared between the two protojets is assigned to the one to whose axis it is closest. The protojet momenta are then recalculated.

9: **else**

Merge the two protojets into a single new protojet (added to the list of protojets, while the two original ones are removed).

11: **end if**

12: If steps 7–11 produced a protojet that coincides with an existing one, maintain the new protojet as distinct from the existing copy(ies).

13: **else**

Add i to the list of final jets, and remove it from the list of protojets.

15: **end if**

16: **until** no protojets are left.

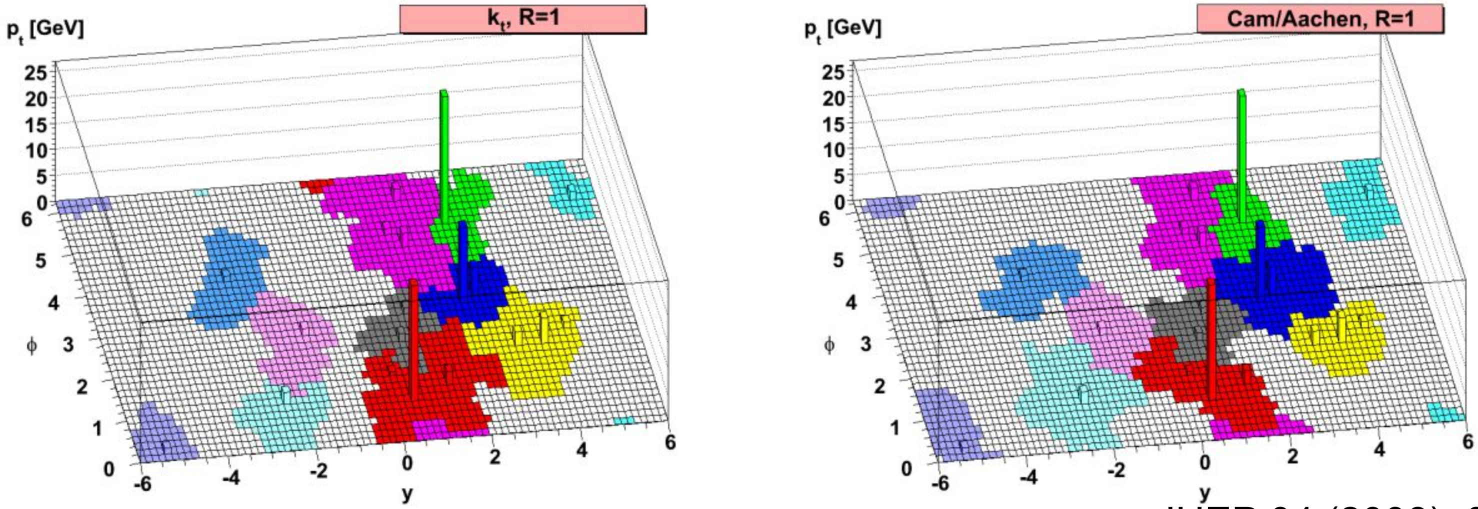
Sequential recombination algorithms

Defined with a metric and simple steps.
k=1 defines “kT algorithm” and k=0 the “Cambridge/Achen algorithm”

$$d_{ij}(p_i, p_j) = \min(p_{T,i}^{2k}, p_{T,j}^{2k}) \frac{\Delta R^2}{R^2}$$

- Calculate pairwise distance between all possible pair of 4-vectors
- Merge the closest two to define a new 4-vector
- Repeat

These had good theory properties but did not yield nice stable cones in presence of soft radiation (an issue in hadron colliders)



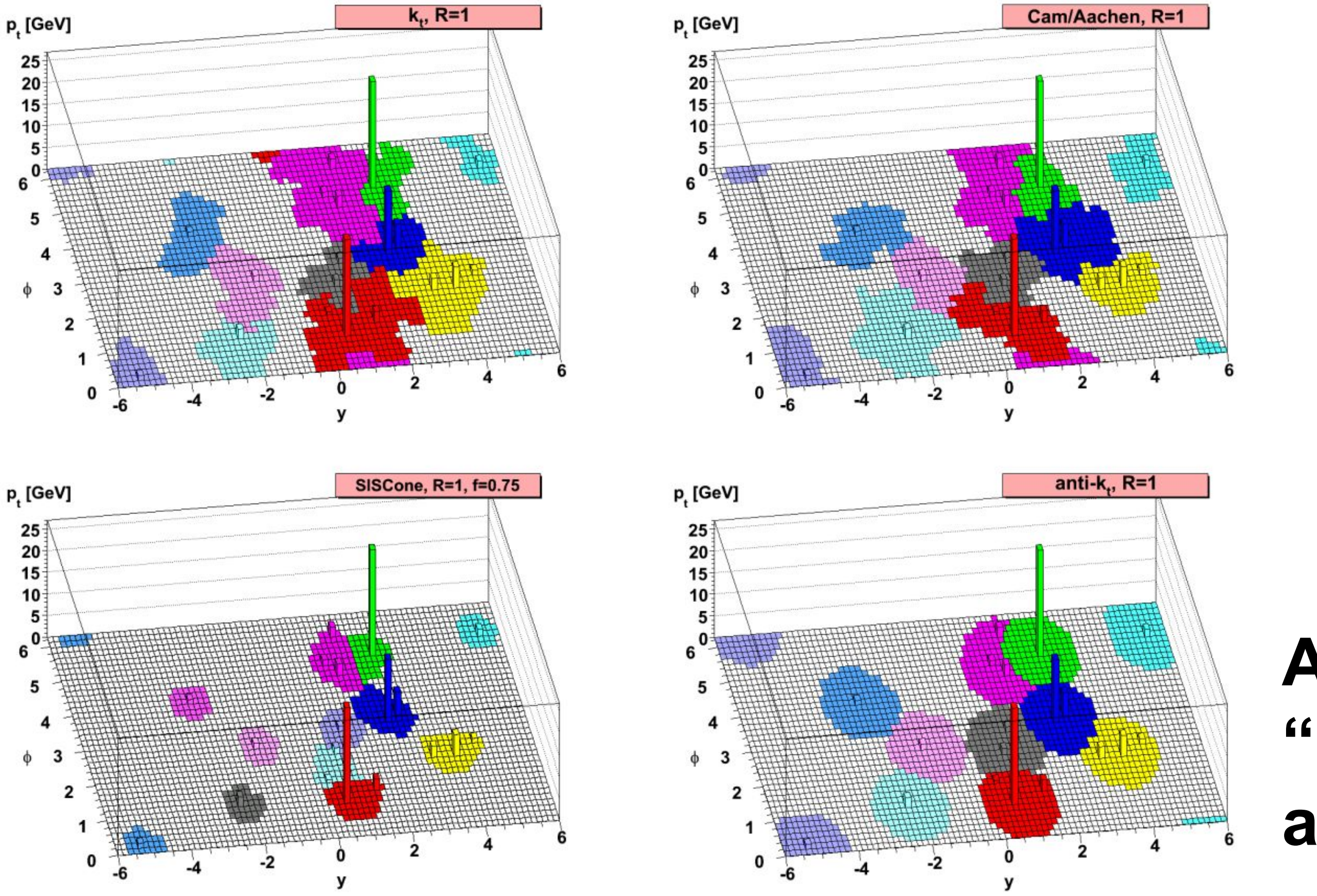
JHEP 04 (2008) 063

One event that contains some particles in azimuth and rapidity space
+ a large number of “ghost” particles.
Colors represent the boundaries defined by different particle algorithms

The “anti-kT” algorithm JHEP 04 (2008) 063

- The exponent in metric can be negative and yield an IRC safe algorithm with other sensible properties. The k=-1 case defines “anti-kT”.

$$d_{ij}(p_i, p_j) = \min(p_{T,i}^{2k}, p_{T,j}^{2k}) \frac{\Delta R^2}{R^2}$$



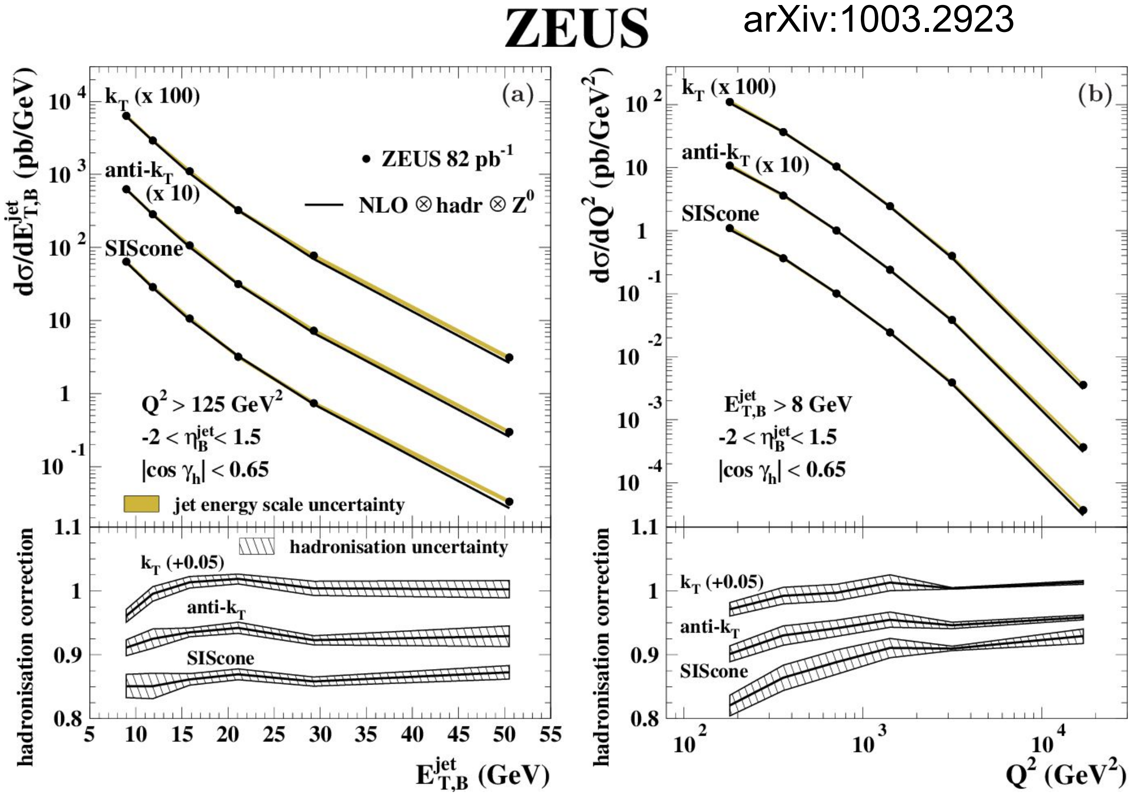
Anti-kT a "perfect cone algorithm"

JHEP 04 (2008) 063

Figure 1: A sample parton-level event (generated with Herwig [8]), together with many random soft "ghosts", clustered with four different jets algorithms, illustrating the "active" catchment areas of

Historical fun fact: anti-kT was invented at the dawn of the LHC era.
Now the default and original paper has 8810 citations

How about at in electron-proton DIS at at HERA?



arXiv:1003.2923

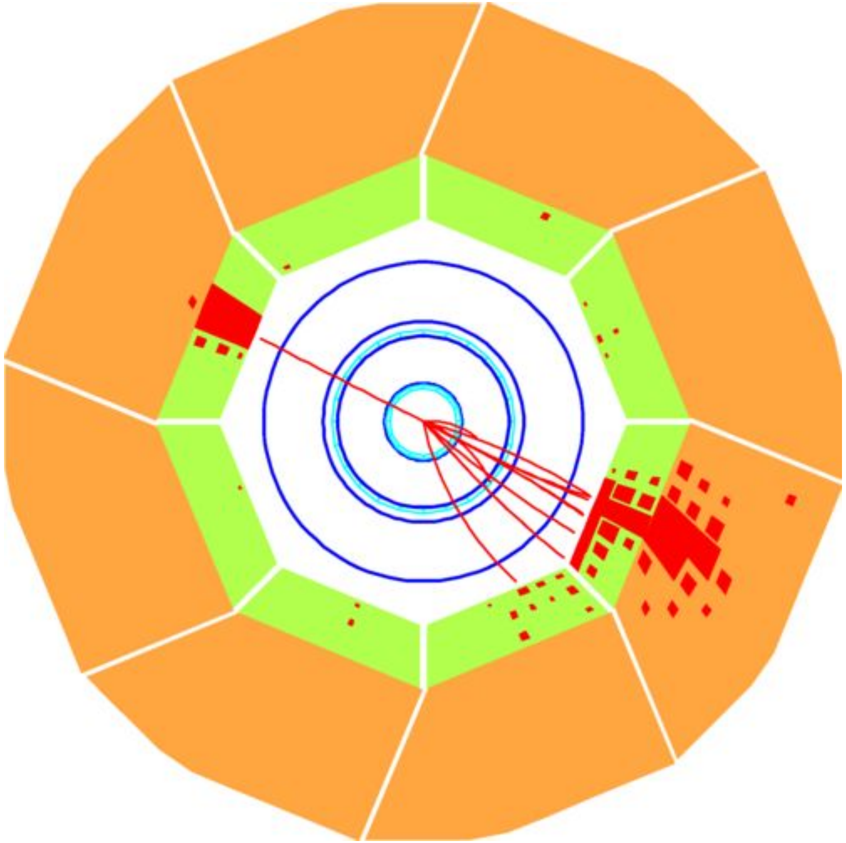
- Most HERA jet measurements use the kT algo.
- Quality of data/theory agreement and hadronization corrections seem rather comparable for kT and anti-kT in DIS

An example of running a jet algorithm

https://fastjet.fr/quickstart.html

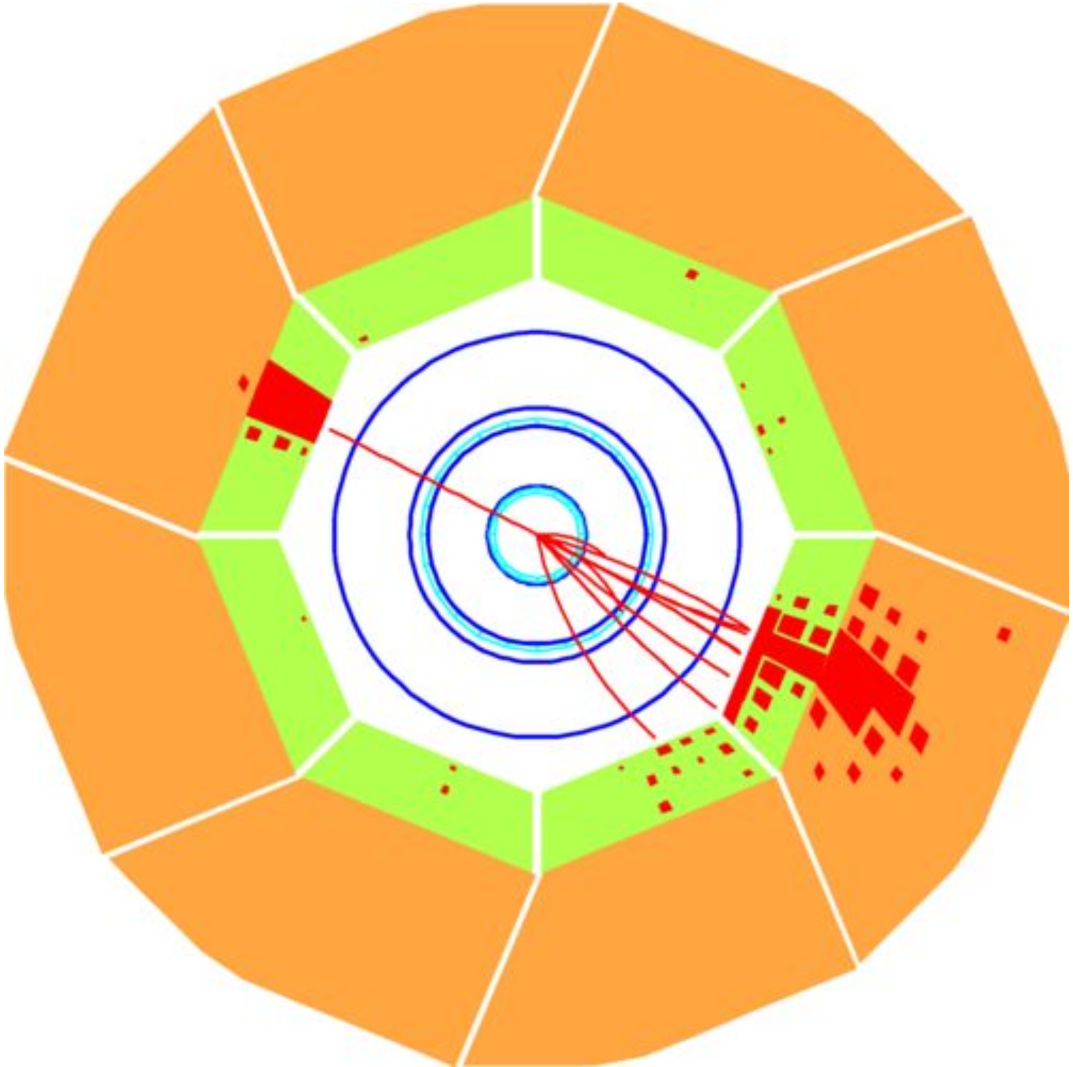
```
#include "fastjet/ClusterSequence.hh"
#include <iostream>
using namespace fastjet;
using namespace std;

int main () {
  vector<PseudoJet> particles;
  // an event with three particles:   px    py  pz      E
  particles.push_back( PseudoJet(   99.0,  0.1,  0, 100.0) );
  particles.push_back( PseudoJet(    4.0, -0.1,  0,   5.0) );
  particles.push_back( PseudoJet(  -99.0,    0,  0,  99.0) );

  // choose a jet definition
  double R = 0.7;
  JetDefinition jet_def(antikt_algorithm, R);

  // run the clustering, extract the jets
  ClusterSequence cs(particles, jet_def);
  vector<PseudoJet> jets = sorted_by_pt(cs.inclusive_jets());

  // print out some infos
  cout << "Clustering with " << jet_def.description() << endl;

  // print the jets
  cout <<   "        pt y phi" << endl;
  for (unsigned i = 0; i < jets.size(); i++) {
    cout << "jet " << i << ": "<< jets[i].pt() << " "
                 << jets[i].rap() << " " << jets[i].phi() << endl;
    vector<PseudoJet> constituents = jets[i].constituents();
    for (unsigned j = 0; j < constituents.size(); j++) {
      cout << "    constituent " << j << "'s pt: " << constituents[j].pt()
           << endl;
    }
  }
}
```

Why are jets useful?

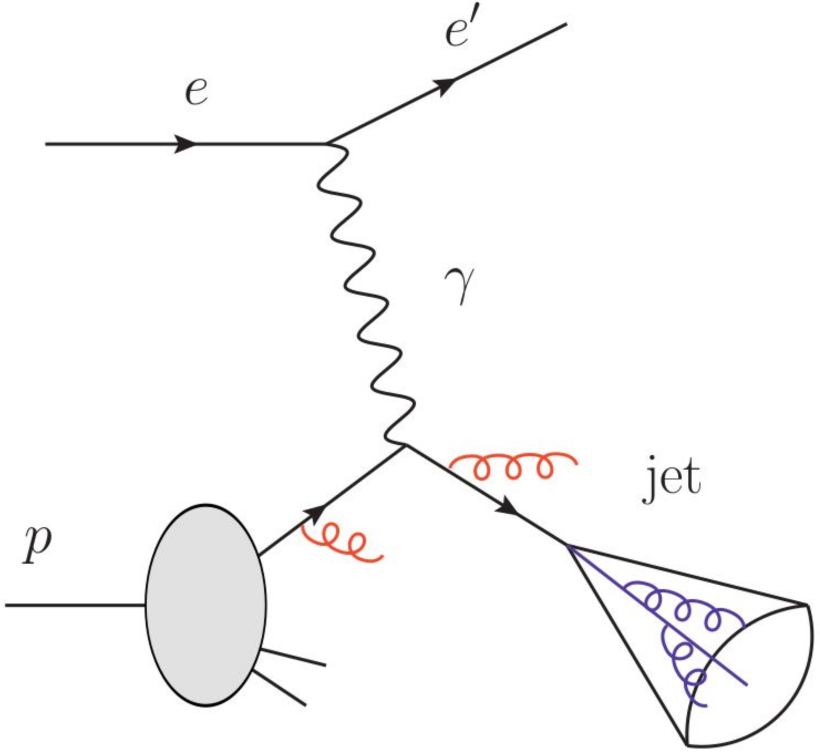


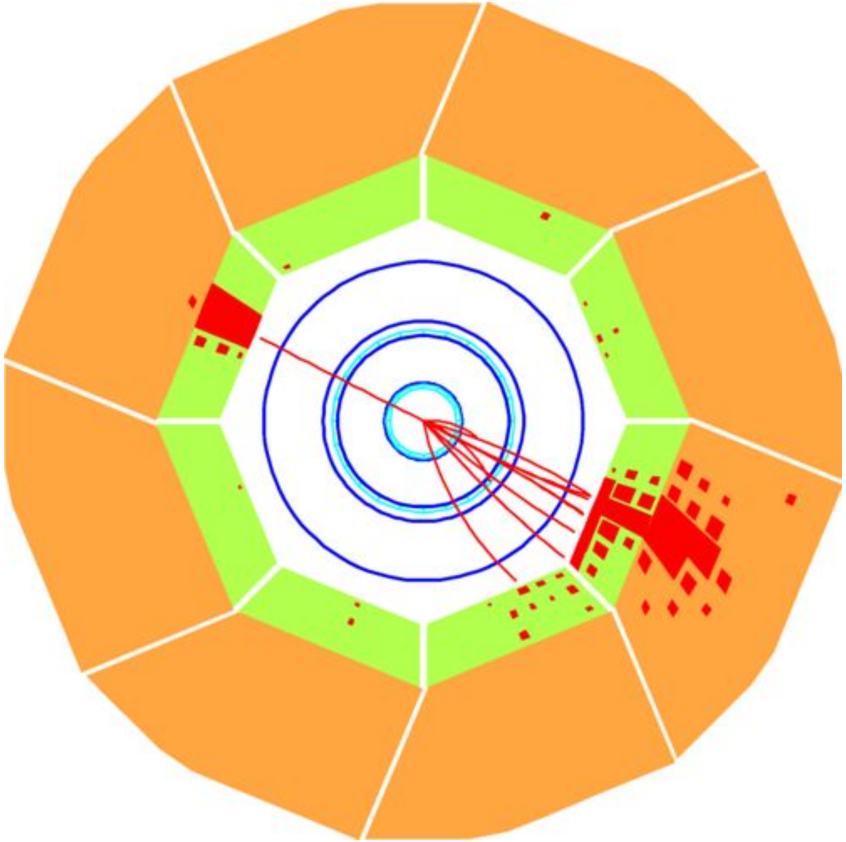
“Jets are measured in particle detectors and studied in order to determine the properties of the original quarks”

Wikipedia.

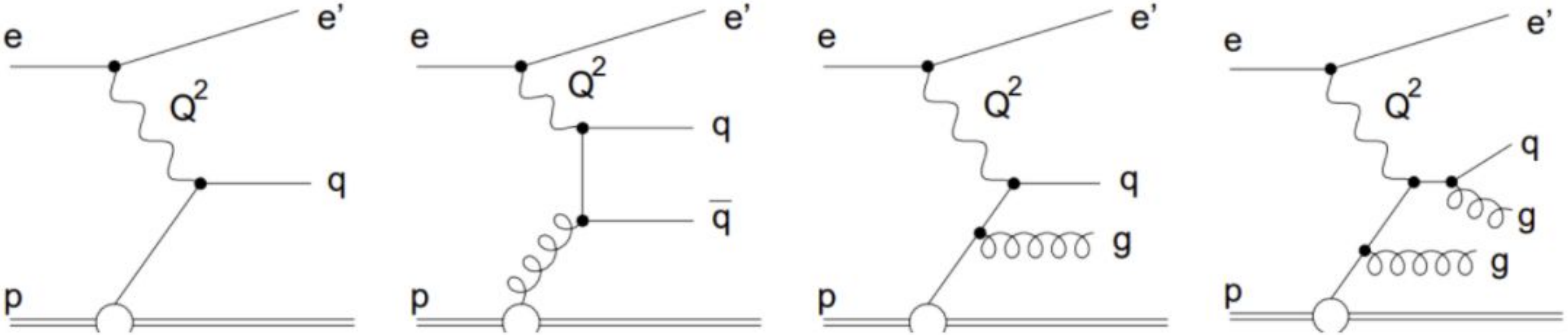


Jets are good proxies for quarks (gluons)





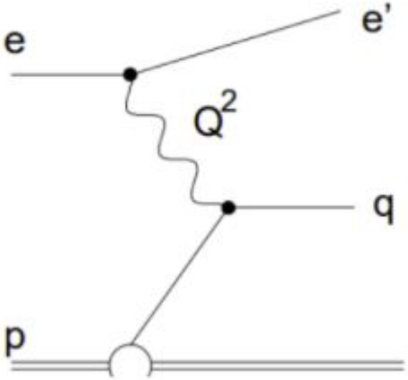
Jet production in DIS

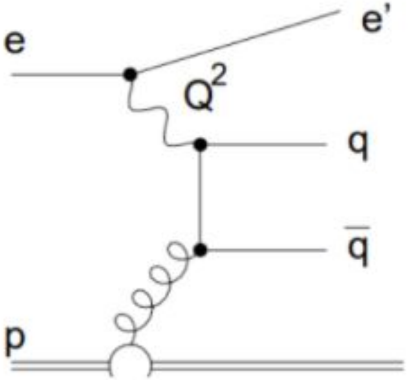


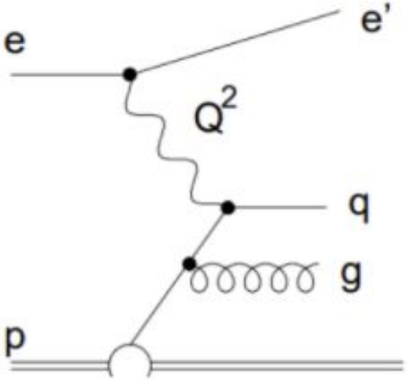
For most HERA studies:

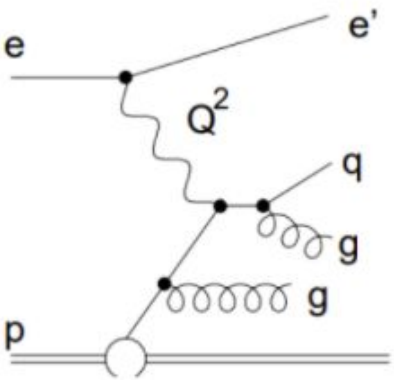
~ 0 p_T in Breit frame
Background

High p_T in Breit frame
Signal (gluon PDFs, alpha_s)



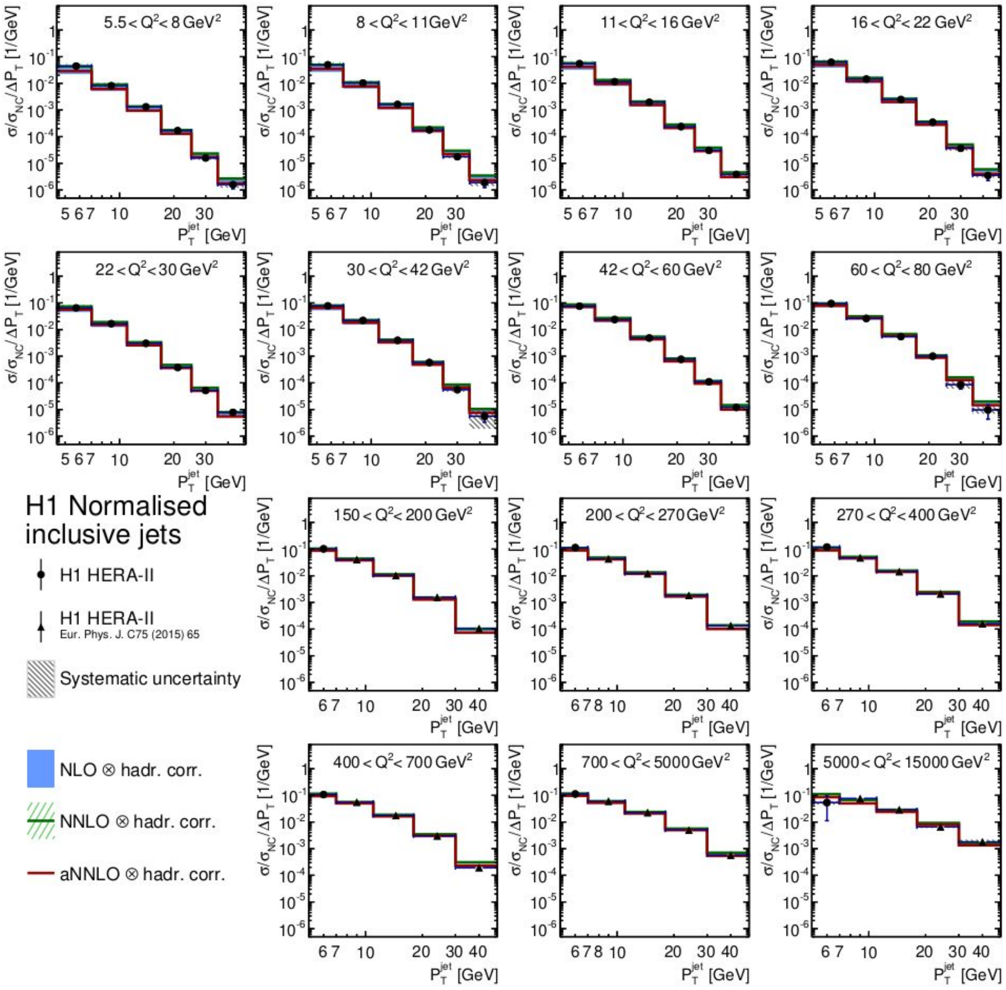






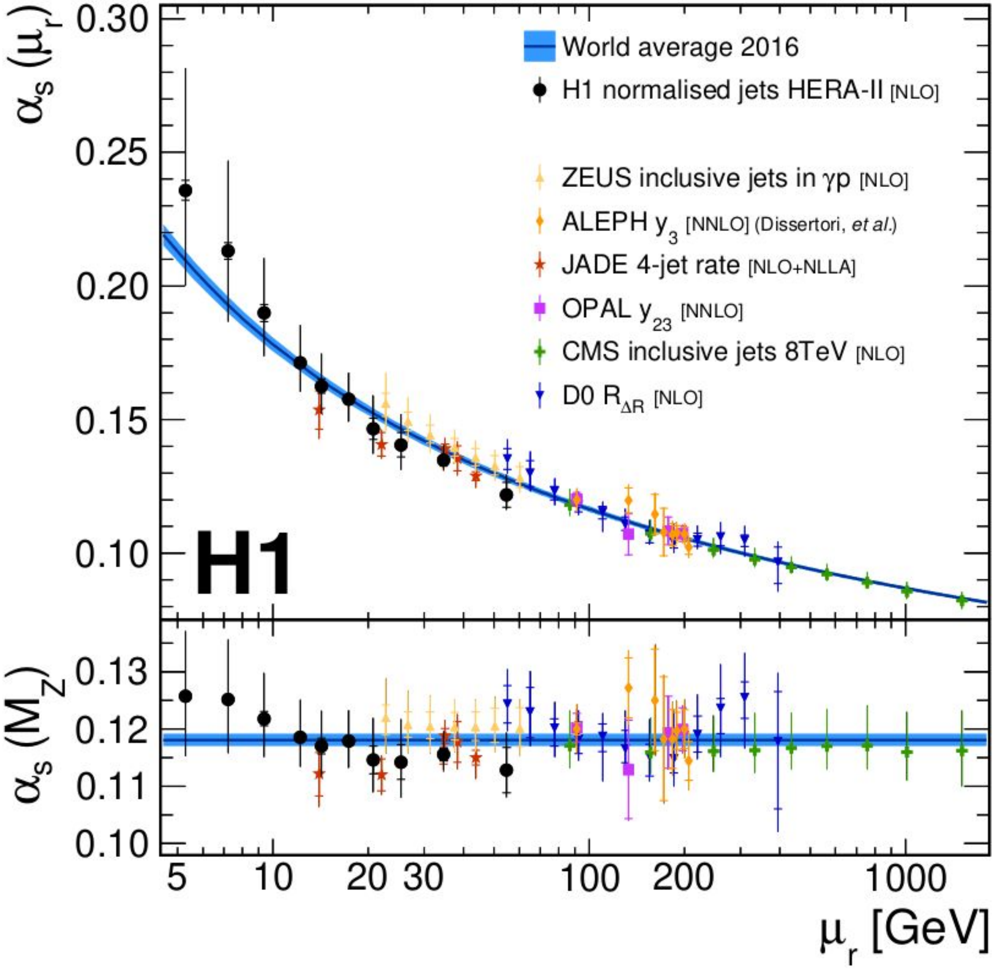
HERA experiments did many measurements like this

arXiv:1611.03421



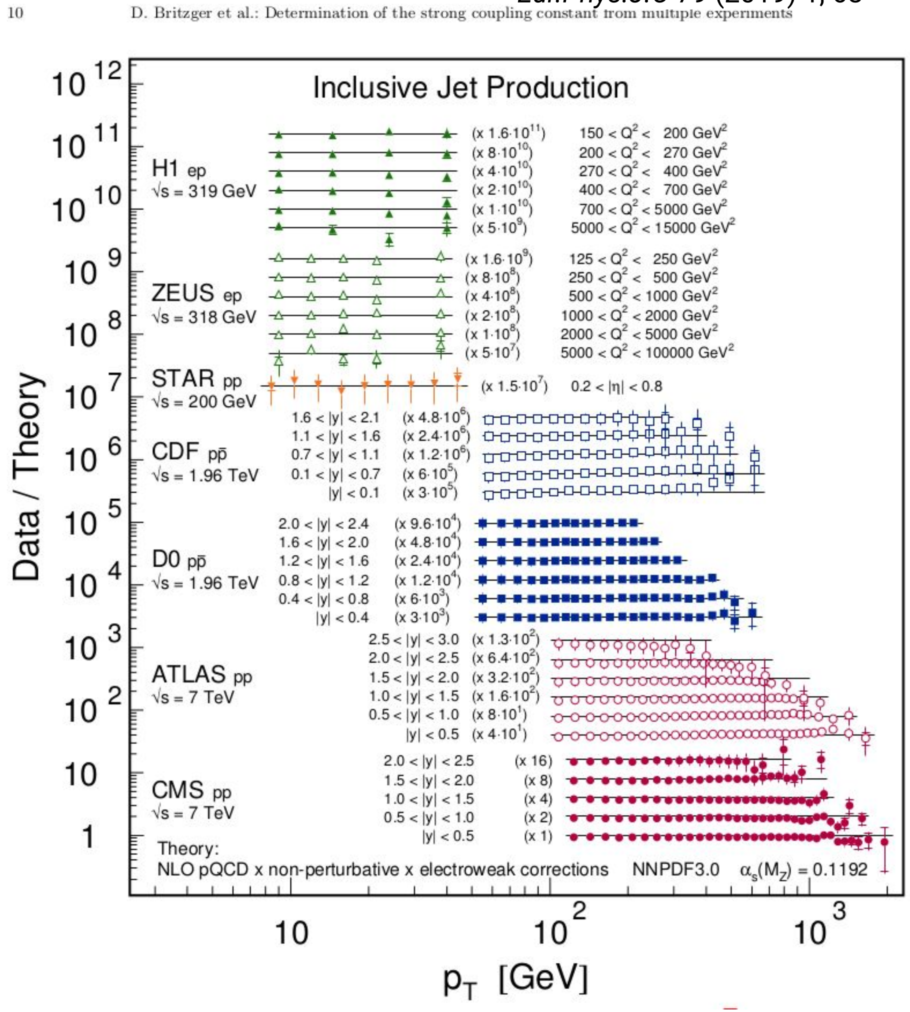
arXiv:1611.03421

Which yielded Nice legacy “textbook” plots

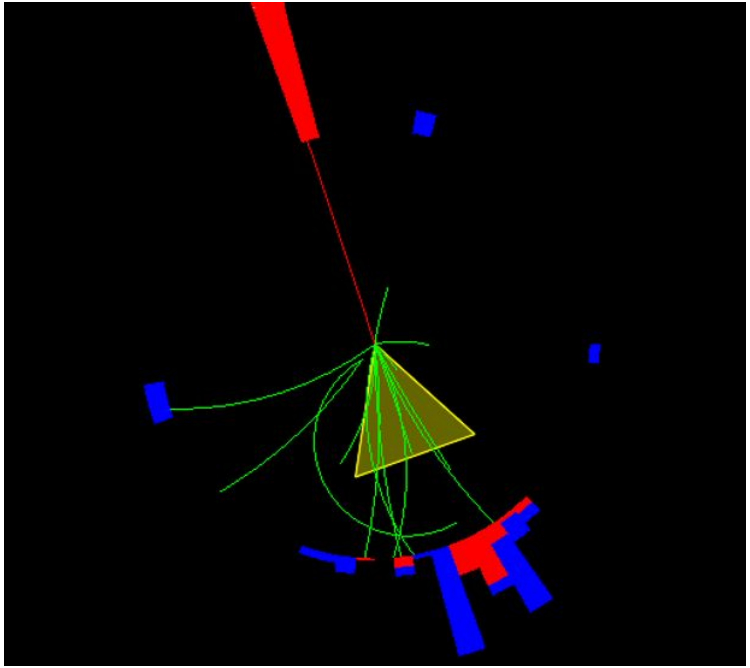


Eur.Phys.J.C 79 (2019) 1, 68

Similar measurements were performed In hadron collisions (Tevatron, RICH and LHC) These constrain strong-coupling and gluon PDFs.



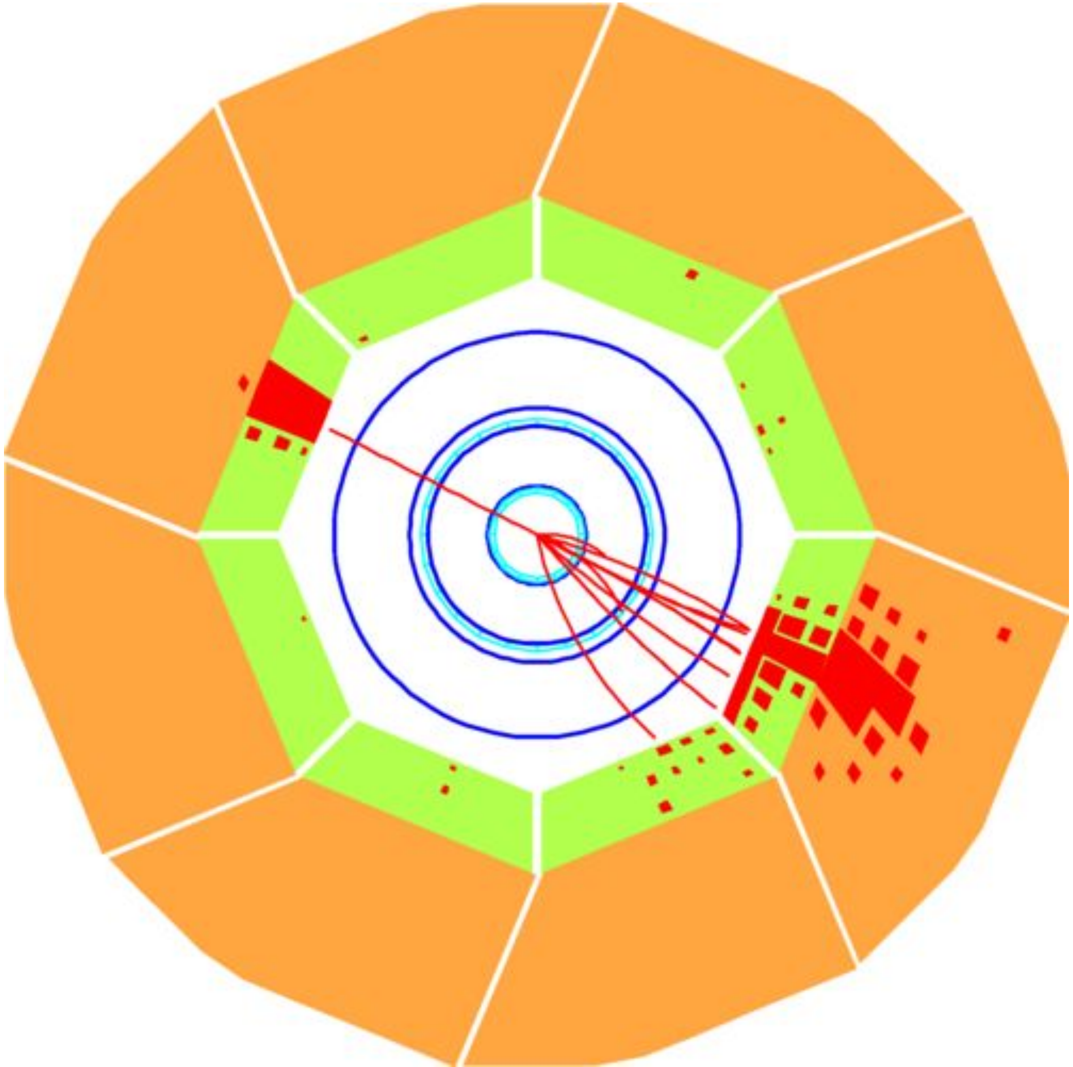
Is the point of jets@EIC to repeat jets@HERA?



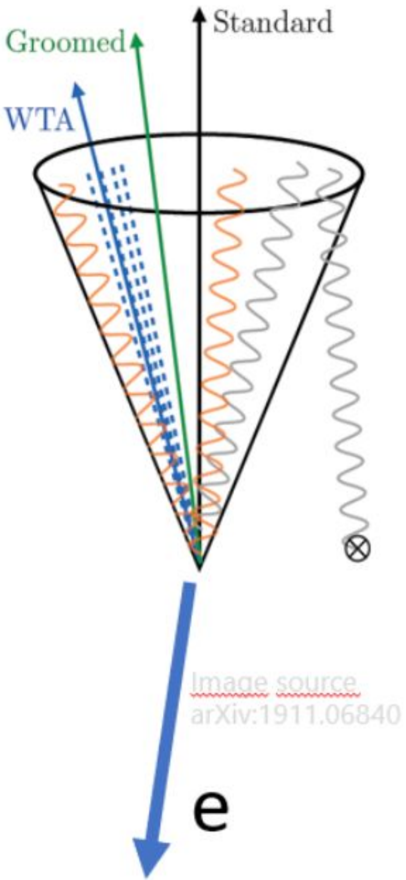
No, that is not the point.
(this is a very common misconception!)
Rather, we will explore jets in polarized DIS and nuclear DIS, which have never done before → Discovery potential

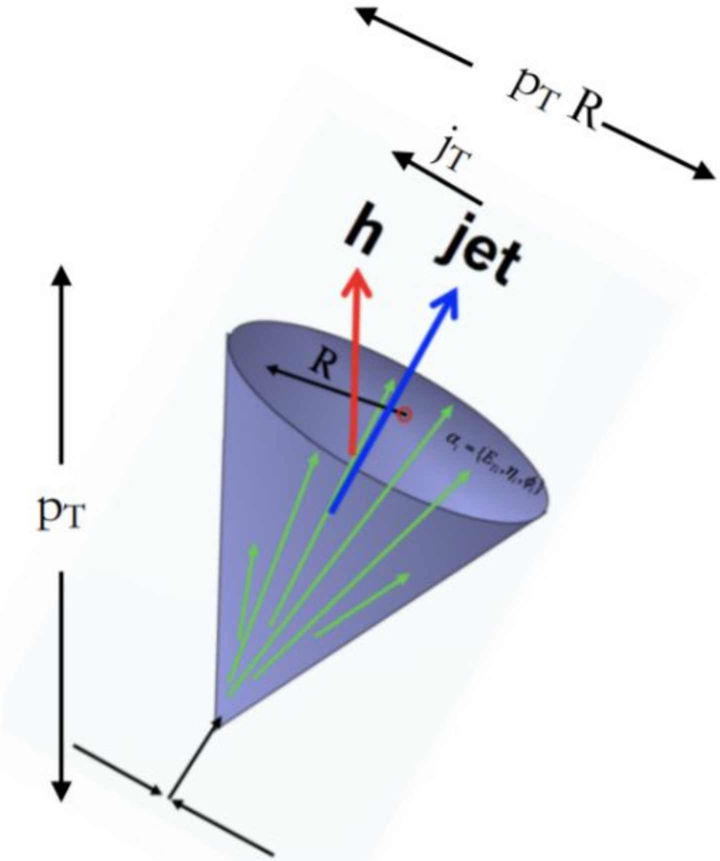
Why are jets useful?

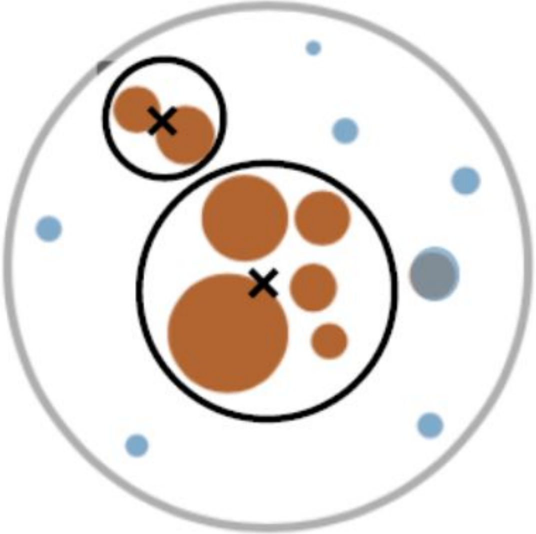
Classic answer:
Good proxies for quarks
Modern addition:
Jets have substructure



Jets have rich substructure, which encodes rich dynamics







Jet Substructure studies have exploded at the LHC over the last decade and is still hot topic. It will surely influence EIC studies

Jet Substructure at the Large Hadron Collider: A Review of Recent Advances in Theory and Machine Learning #1

Andrew J. Larkoski (Reed Coll.), Ian Moult (UC, Berkeley and LBNL, Berkeley), Benjamin Nachman (LBL, Berkeley) (Sep 13, 2017)

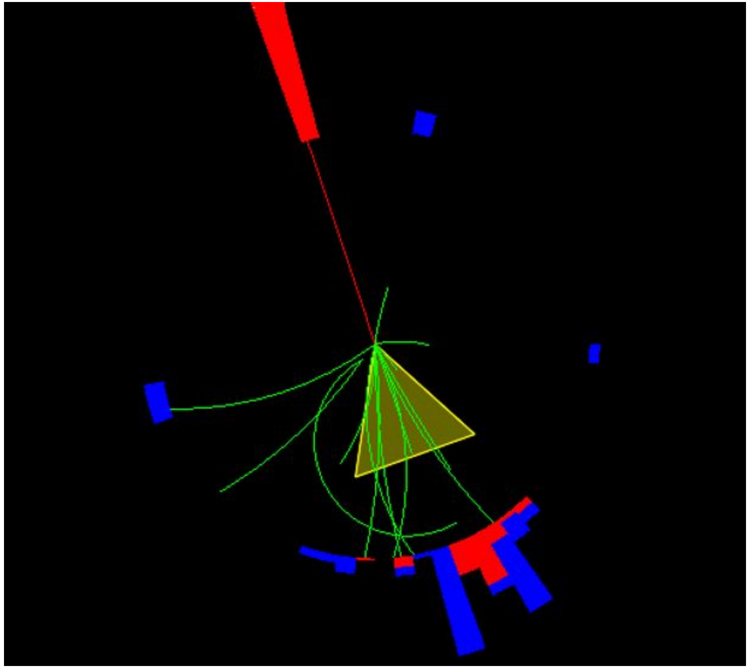
Published in: *Phys.Rept.* 841 (2020) 1-63 • e-Print: 1709.04464 [hep-ph]

pdf DOI cite 366 citations

Why is studying jet substructure useful?

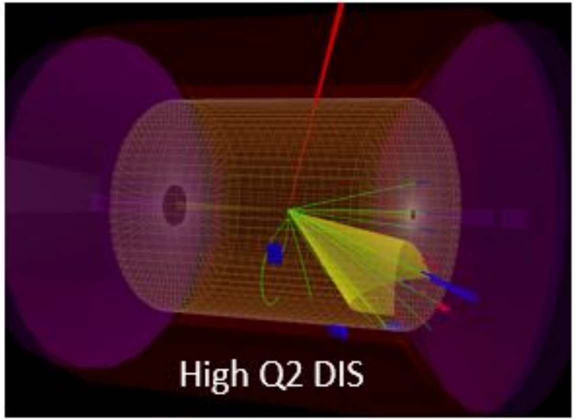
- Substructure encodes **much more information** (QCD) than a single 4 vector→ many more studies possible.
- Can be used as **tool** to better control theory or experiment.

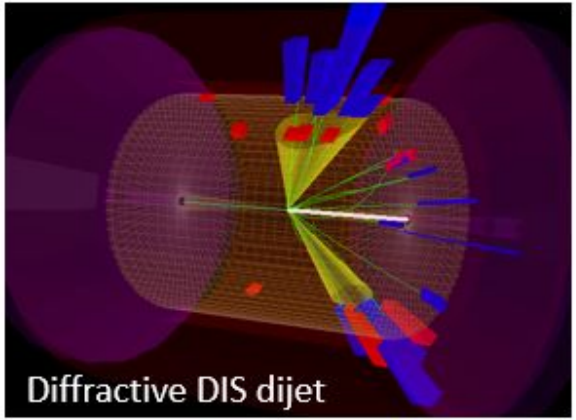
Is the point of jets@EIC to repeat jets@HERA?

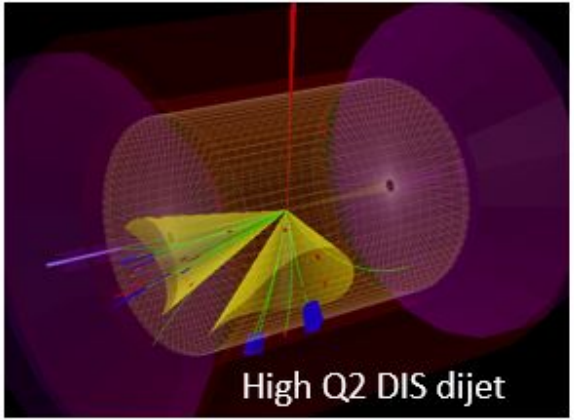


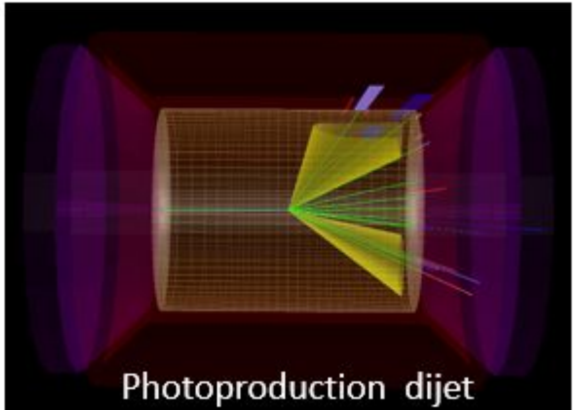
No, and jet substructure was in its infancy at HERA times!

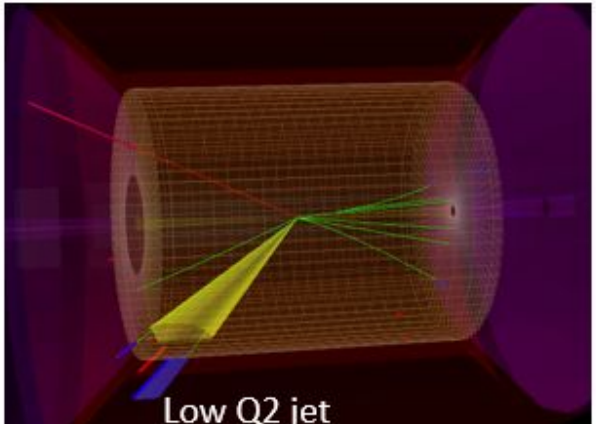
The EIC, a jet factory, will make the first jets in nuclear DIS and proton-polarized DIS

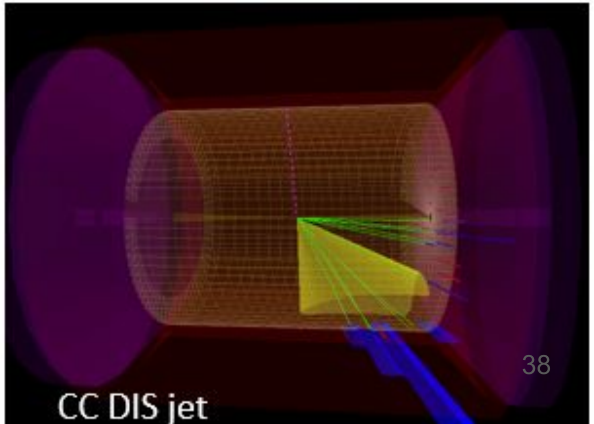


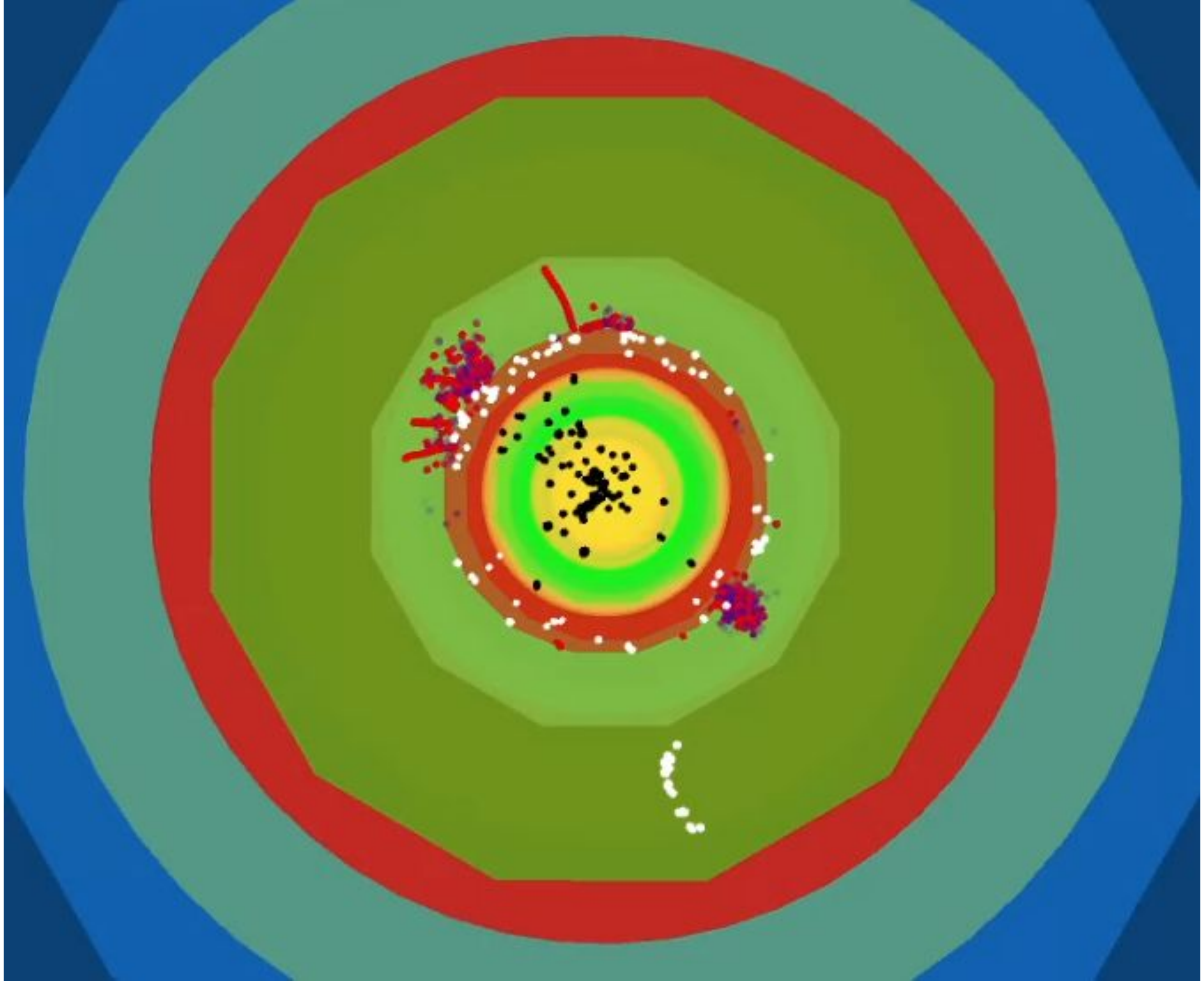






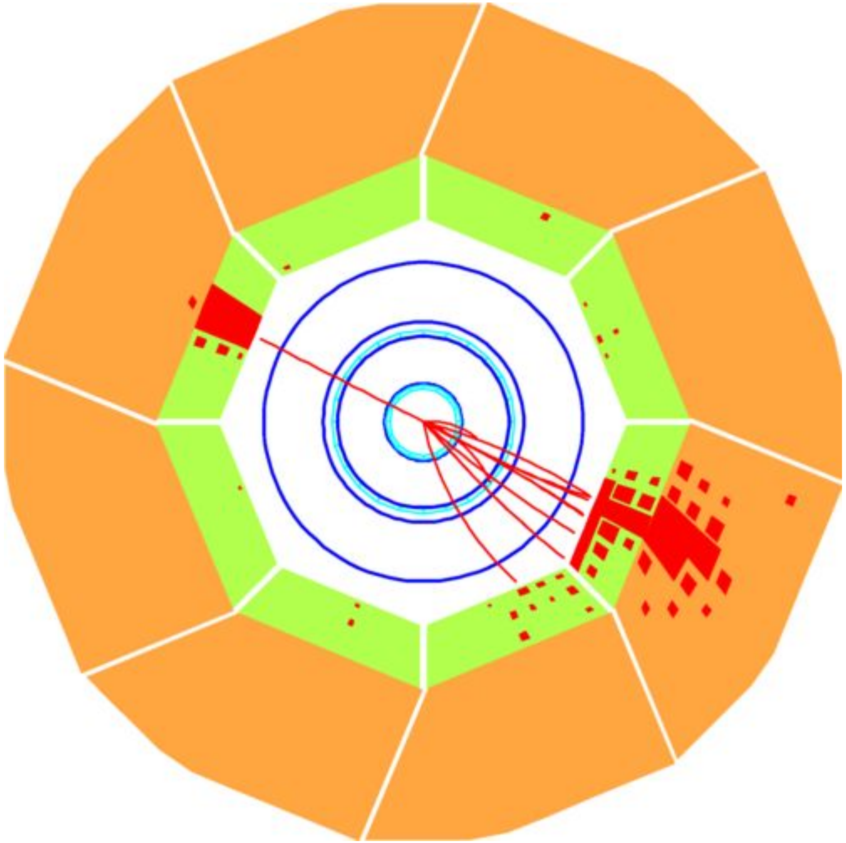






More next lecture…

Summary



What is a jet?
Output of a jet algo

Why are jets useful?
Proxies to partons and their substructure encodes rich, useful info