

SpartyJet

Code organisation

Main Classes

InputMaker

```
fillInput(int eventn,  
          Jet::jet_list_t &inputList)
```

Reads a input collection of 4-vectors
and convert it into an **initial jets list**

Example InputMaker implementation :

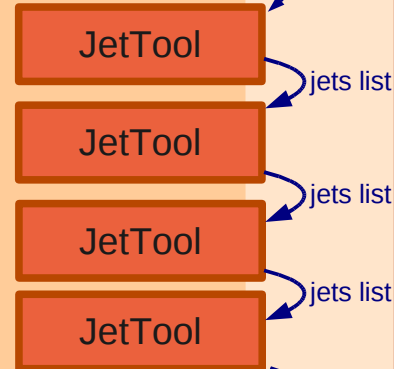
NtupleInputMaker

TextInputMaker

JetAlgorithm

```
addTool(JetTool * tool);  
execute(Jet::jet_list_t &inputJets,  
        JetCollection &outputJets);
```

'execute' just passes **jets list**
through a list of JetTools



JetTool

a base class. Every concrete
tools inherit from it

```
execute(JetCollection &inputJets)
```

'execute' modifies the input jet list

Example sequence :

JetSelectorTool

MinBiasInserterTool

JetKtFinderTool

JetSelectorTool

NtupleMaker

```
addJetVar(std::string jetname);  
set_data(std::string jetname,  
          JetCollection &theJets);
```

Handle ntuple creation for arbitrary
number of jet collection identified by names.

The only place where jet finding algs
are actually implemented.
Simply inherit from JetTool and plug
implementation in execute()

Basic objects

CLHEP::HepLorentzVector

Lorentzvector

Jet

holds :
list<Jet*> constituents

Other Classes

std::vector<Jet*>
(typedef as Jet::jet_list_t)

JetCollection

Is a vector of **Jet*** by inheritance

holds a **JetMomentMap**

shortcuts to access jet moment

```
get_jet_moment(std::string mom_name,  
               int jet_index);  
get_jet_moment_array(std::string mom_name,  
                     int jet_index);
```

JetMomentMap

Associated to a jet collection

- holds arbitrary quantities calculated for jets 'moment'

- holds variables associated to the jet alg

```
void schedule_jet_moment(std::string name);  
void set_jet_moment(int pos, Jet* jet, float value);  
  
void schedule_event_moment(std::string name);  
void set_Event_moment(std::string name, float value);
```

JetBuilder

Combine every classes together to perform
jet finfding and write results
Simple settings applied to all jet algs

holds :

- InputMaker
- Ntuplemaker
- list<JetAlgorithm*>

```
void configure_input(InputMaker *input, bool saveInput = true);  
void configure_output(std::string treename, std::string filename);  
  
void add_default_alg(JetTool *jetfinder , bool withIndex = false);  
  
void process_events(int nevent, int start =0);
```

ROOT script example

```
gSystem->Load("libs/libJetCore.so");
gSystem->Load("libs/libCDFJet.so");
gSystem->Load("libs/libATLASJet.so");

// configure an interface to the tree -----
NtupleInputMaker input(NtupleInputMaker.PxPyPzE_vector_double);
input.set_variables("px","py","pz","E");
input.set_n_name("N");
input.set_prefix("Clusters_");
input.setFileTree('myFile.root', "FullRec0");
// -----

JetBuilder builder;
builder.configure_input((InputMaker*)&input);

builder.add_default_alg( new MidPoint("myMidPoint"));

atlas::FastKtTool * fastkt = new atlas::FastKtTool("FastKt");
fastkt->simple_config("Standard",0.7);
builder.add_default_alg(fastkt , true);

builder.set_default_cut(0,5000);

builder.configure_output("myTree","out.root");

EtaPhiMomentTool * mom = new EtaPhiMomentTool("EtaPhiMom");
builder.add_moments(mom);
builder.do_time_measure();

// -----
builder.silent_mode();
builder.process_events(300);
```

1

2

3

4

5

6

7

SpartyJet code location

`JetCore/`

Core classes (Jet, JetTool, JetAlgorithm, etc..)

`atlas/`
`celljet/`
`fastjet/`
`cdf/`
`D0/`

Various implementations of jet finding algorithms

`programs/`
`python/`
`scripts/`

Example code for executable, ROOT scripts and python scripts

`userjet/`
`extras/`

Template code for a jet finder
Everything else including experimental code

