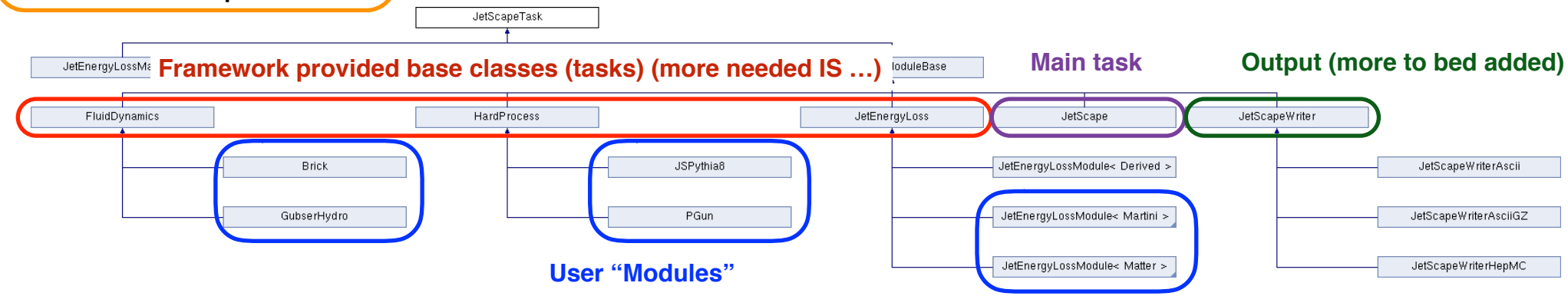
Parton-Shower Generation in JetScape

Joern Putschke (WSU)

```
[Info] *-------------------------------------------------------------*
[Info] |                                                             |
[Info] |                  /\                                         |
[Info] |                 /  \                                        |
[Info] |                / |  \                                       |
[Info] |               /  |   \            /\                        |
[Info] |              /   |    \         /\/  \                      |
[Info] |             / \  |  /  \     /\/   |  \/\                  |
[Info] |            /   \ | /    \   /     % | %   \                 |
[Info] |         __/     \|/     \__/      %|%     \/\__             |
[Info] |                                                             |
[Info] |                JETSCAPE beta release 0.1                    |
[Info] |                                                             |
[Info] |     The Jet Energy-loss Tomography with a Statistically     |
[Info] |       and Computationally Advanced Program Envelope         |
[Info] |               http://jetscape.wayne.edu                     |
[Info] |                                                             |
[Info] | Please cite xxx if you use this package for scientific work. |
[Info] |                                                             |
[Info] | JETSCAPE is provided without warranty under the terms       |
[Info] | of the GNU GPLv3. It uses xxx code(s).                      |
[Info] | See COPYING file for details.                               |
[Info] |                                                             |
[Info] *-------------------------------------------------------------*
```

Task-based Framework Implementation

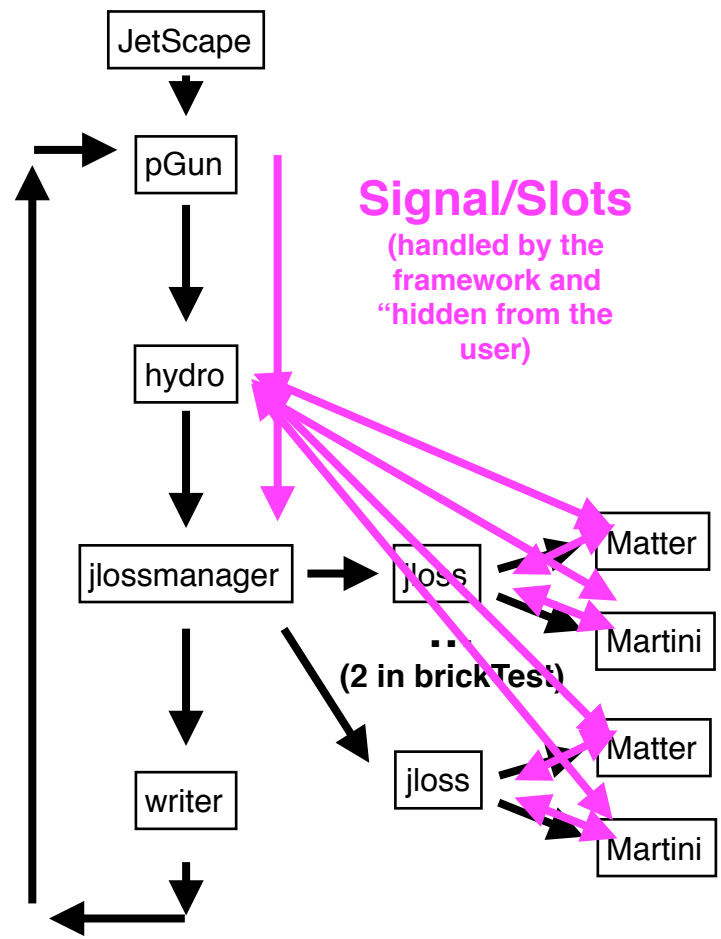
The JetScapeTask:



Member Functions

	JetScapeTask ()
virtual	~JetScapeTask ()
virtual void	Init ()
virtual void	Exec ()
virtual void	Finish ()
virtual void	Clear ()
virtual void	ExecuteTasks ()
virtual void	ExecuteTask ()
virtual void	InitTask ()
virtual void	InitTasks ()
virtual void	ClearTasks ()
virtual void	ClearTask ()
virtual void	FinishTask ()
virtual void	FinishTasks ()
virtual void	WriteTasks (weak_ptr< JetScapeWriter > w)
virtual void	WriteTask (weak_ptr< JetScapeWriter > w)
virtual void	Add (shared_ptr< JetScapeTask > m_tasks)
ctor< shared_ptr< JetScapeTask > >	GetTaskList () const
shared_ptr< JetScapeTask >	GetTaskAt (int i)
void	EraseTaskLast ()
void	EraseTaskAt (int i)
void	ResizeTaskList (int i)
void	ClearTaskList ()
int	GetNumberOfTasks ()
const bool	GetActive () const
void	SetActive (bool m_active_exec)
void	SetId (string m_id)
const string	GetId () const

All of theses functions will be called recursively in a task based framework, so once added there is no problem that it will get initialized, executed (if specified) … Of course a task in itself is a natural unit which can be easily parallelized if necessary

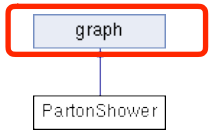


JetScape Parton Showers as Graph “Containers” (more later)

PartonShower Class Reference

```
#include <PartonShower.h>
```

Inheritance diagram for PartonShower:



GTL: Graph Template Library

Public Member Functions

	PartonShower ()
virtual	~PartonShower ()
node	new_vertex (shared_ptr< VertexBase > v)
void	new_parton (node s, node t, shared_ptr< Parton > p)
shared_ptr< VertexBase >	GetVertex (node n)
shared_ptr< Parton >	GetParton (edge e)
shared_ptr< Parton >	GetPartonAt (int n)
shared_ptr< VertexBase >	GetVertexAt (int n)
node	GetNodeAt (int n)
edge	GetEdgeAt (int n)
int	GetNumberOfParents (int n)
int	GetNumberOfChilds (int n)
vector< shared_ptr< Parton > >	GetFinalPartons ()
vector< Parton >	GetFinalPartonsForFastJet ()
int	GetNumberOfPartons () const
int	GetNumberOfVertices () const
void	save_node_info_handler (ostream *o, node n) const
void	save_edge_info_handler (ostream *o, edge n) const
void	load_edge_info_handler (edge e, GML_pair *read)
void	load_node_info_handler (node n, GML_pair *read)
void	pre_clear_handler ()
void	PrintVertices ()
void	PrintPartons ()
void	PrintNodes (bool verbose=true)
void	PrintEdges (bool verbose=true)
void	SaveAsGML (string fName)
void	SaveAsGV (string fName)
void	SaveAsGraphML (string fName)

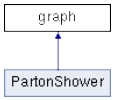
▸ Public Member Functions inherited from graph

graph Class Reference

A directed or undirected graph. More...

```
#include <graph.h>
```

Inheritance diagram for graph:



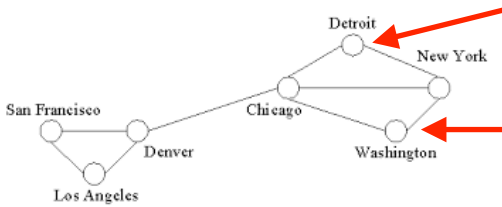
Public Types

typedef list< node >::const_iterator	node_iterator
typedef list< edge >::const_iterator	edge_iterator

Public Member Functions

	graph ()
	graph (const graph &G)
	graph (const graph &G, const list< node > &nodes)
	graph (const graph &G, list< node >::const_iterator it, list< node >::const_iterator end)
virtual	~graph ()
void	make_directed ()
void	make_undirected ()
bool	is_directed () const
bool	is_undirected () const
bool	is_bidirected (edge_map< edge > &rev) const
bool	is_connected () const
bool	is_acyclic () const
int	number_of_nodes () const
int	number_of_edges () const
node	center () const
virtual node	new_node ()
virtual edge	new_edge (node s, node t)
virtual edge	new_edge (const list< node > &sources, const list< node > &targets)
void	del_node (node n)
void	del_all_nodes ()

...



City=Node—> Vertex

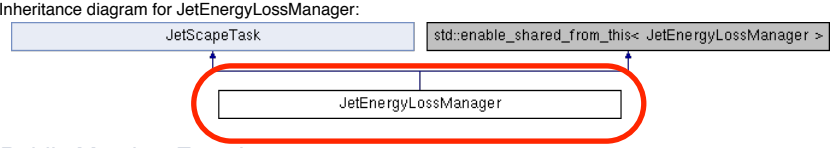
Route=Edge—> Parton

JetEnergyLoss and JetEnergyLossManager

JetEnergyLossManager Class Reference

```
#include <JetEnergyLossManager.h>
```

Inheritance diagram for JetEnergyLossManager:



Public Member Functions

	JetEnergyLossManager ()
virtual	**~JetEnergyLossManager** ()
virtual void	**Init** ()
virtual void	**Exec** ()
virtual void	**Clear** ()
virtual void	**WriteTask** (weak_ptr< **JetScapeWriter** > w)
int	**GetNumSignals** ()
void	**CreateSignalSlots** ()
void	**SetGetHardPartonListConnected** (bool m_GetHardPartonListConnected)
const bool	**GetGetHardPartonListConnected** ()

▸ Public Member Functions inherited from **JetScapeTask**

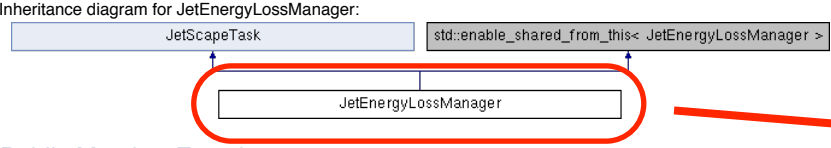
jlossmanager

JetEnergyLoss and JetEnergyLossManager

JetEnergyLossManager Class Reference

`#include <JetEnergyLossManager.h>`

Inheritance diagram for JetEnergyLossManager:

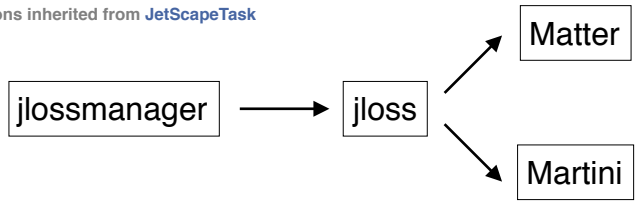


Public Member Functions

	JetEnergyLossManager ()
virtual	~JetEnergyLossManager ()
virtual void	Init ()
virtual void	Exec ()
virtual void	Clear ()
virtual void	WriteTask (weak_ptr< JetScapeWriter > w)
int	GetNumSignals ()
void	CreateSignalSlots ()
void	SetGetHardPartonListConnected (bool m_GetHardP
const bool	GetGetHardPartonListConnected ()

Public Member Functions inherited from JetScapeTask

Creates as many copies as needed (based on input from IS/HardProcess). Including all relevant signal/ slots. Simple to parallelize, checked with C++11 <thread>. Works fine!



shared_ptr< Parton >	GetShowerInitiatingParton ()
void	PrintShowerInitiatingParton ()
double	GetDeltaT ()
double	GetMaxT ()
shared_ptr< PartonShower >	GetShower ()

Public Member Functions inherited from JetScapeModuleBase

Public Member Functions inherited from JetScapeTask

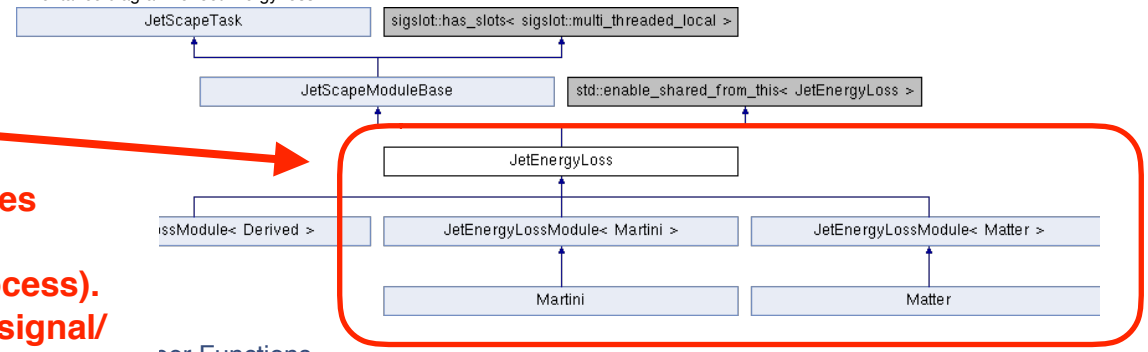
Public Attributes

sigslot::signal2< int, double, multi_threaded_local >	jetSignal
sigslot::signal2< int, double &, multi_threaded_local >	edensitySignal
sigslot::signal5< double, double, double, double, JetSource, multi_threaded_local >	AddJetSourceSignal
sigslot::signal5< double, double, double, double, double &, multi_threaded_local >	GetTemperatureSignal
sigslot::signal5< double, double, double, double, FluidCellInfo *, multi_threaded_local >	GetHydroCellSignal
sigslot::signal4< double, double, vector< Parton > &, vector< Parton > &, multi_threaded_local >	SentInPartons
sigslot::signal1< vector< Parton > &, multi_threaded_local >	GetOutPartons

JetEnergyLoss Class Reference

`#include <JetEnergyLoss.h>`

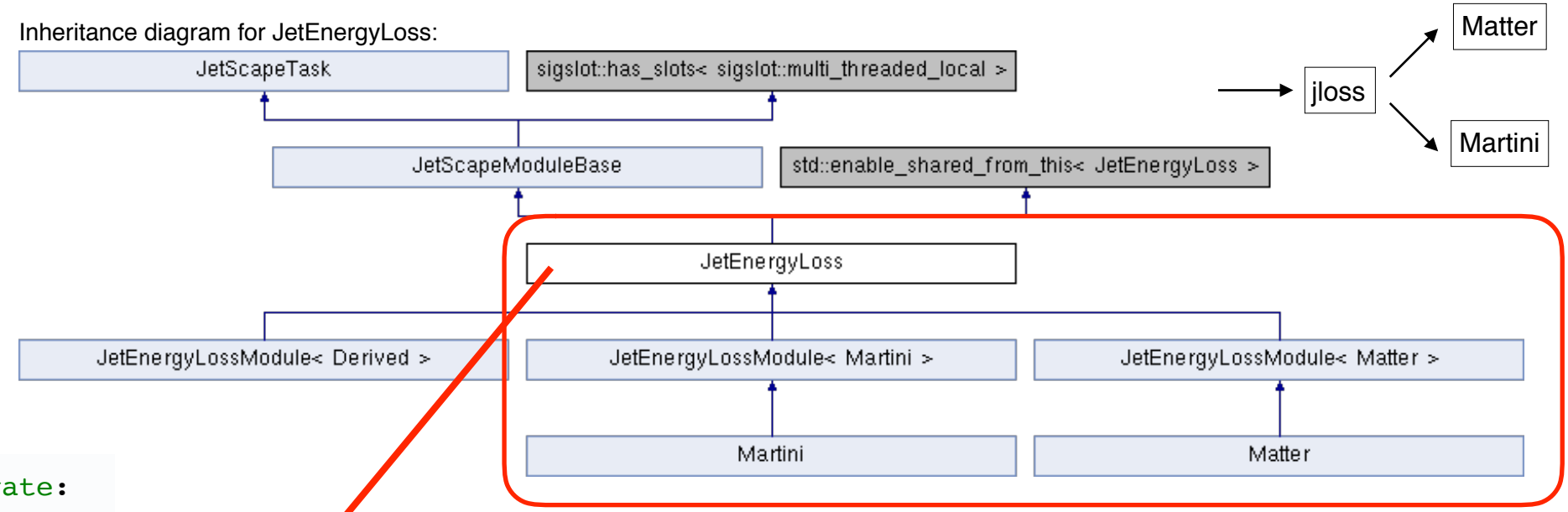
Inheritance diagram for JetEnergyLoss:



Public Member Functions

	JetEnergyLoss ()
	JetEnergyLoss (string m_name)
	JetEnergyLoss (const JetEnergyLoss &j)
virtual	~JetEnergyLoss ()
virtual shared_ptr< JetEnergyLoss >	Clone () const
virtual void	Init ()
virtual void	Exec () final
virtual void	WriteTask (weak_ptr< JetScapeWriter > w)
virtual void	Clear ()
virtual void	DoEnergyLoss (double deltaT, double Q2, vector< Parton > &pIn, vector< Parton > &pOut)
void	SetQhat (double m_qhat)
const double	GetQhat () const
void	SetJetSignalConnected (bool m_jetSignalConnected)
const bool	GetJetSignalConnected () const
void	SetEdensitySignalConnected (bool m_edensitySignalConnected)
const bool	GetEdensitySignalConnected () const
void	SetAddJetSourceSignalConnected (bool m_AddJetSourceSignalConnected)
const bool	GetAddJetSourceSignalConnected ()
void	SetGetTemperatureSignalConnected (bool m_GetTemperatureSignalConnected)
const bool	GetGetTemperatureSignalConnected ()
void	SetGetHydroCellSignalConnected (bool m_GetHydroCellSignalConnected)
const bool	GetGetHydroCellSignalConnected ()
void	SetSentInPartonsConnected (bool m_SentInPartonsConnected)
const bool	GetSentInPartonsConnected ()
void	SetGetOutPartonsConnected (bool m_GetOutPartonsConnected)
const bool	GetGetOutPartonsConnected ()
void	AddShowerInitiatingParton (shared_ptr< Parton > p)

The “Do” Parton Shower in JetScapeEnergyLoss

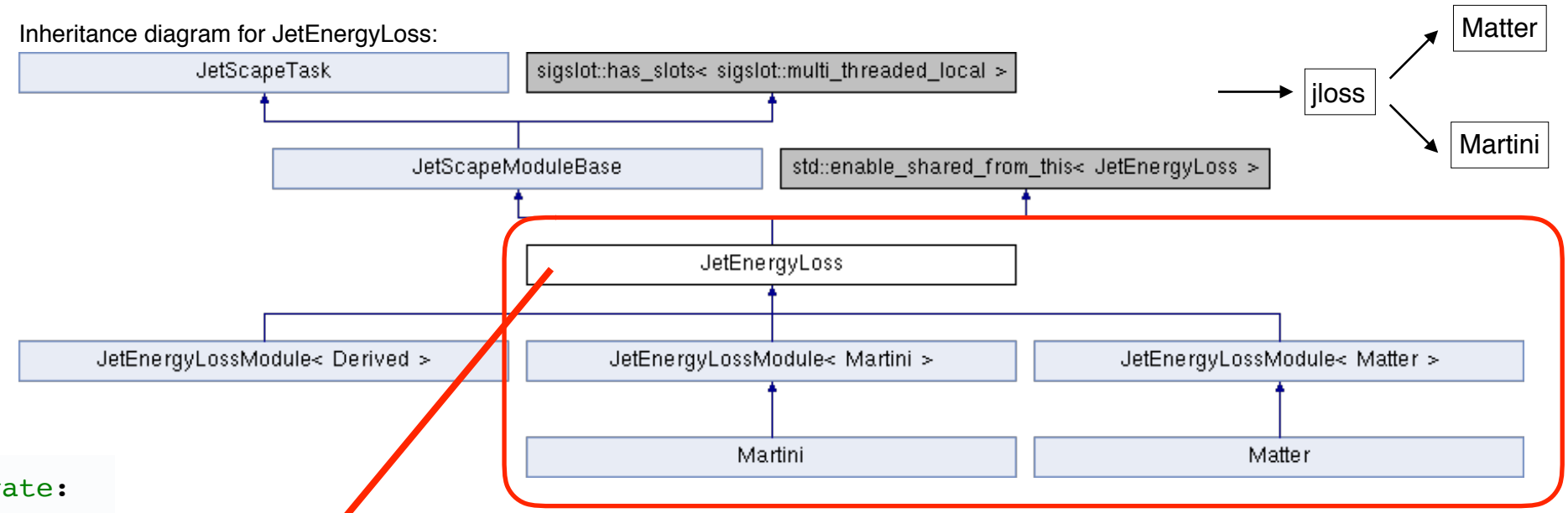



```
private:
...
shared_ptr<PartonShower> pShower;
node vStart;
node vEnd;
void DoShower();
...
```

Recursive
(n) —> (m) parton “splitting”
routine in time steps deltaT

The “Do” Parton Shower in JetScapeEnergyLoss

Inheritance diagram for JetEnergyLoss:




```
private:
...
shared_ptr<PartonShower> pShower;
node vStart;
node vEnd;
void DoShower();
...
```

Recursive
(n) —> (m) parton “splitting”
routine in time steps deltaT

DoShower() sends a Signal

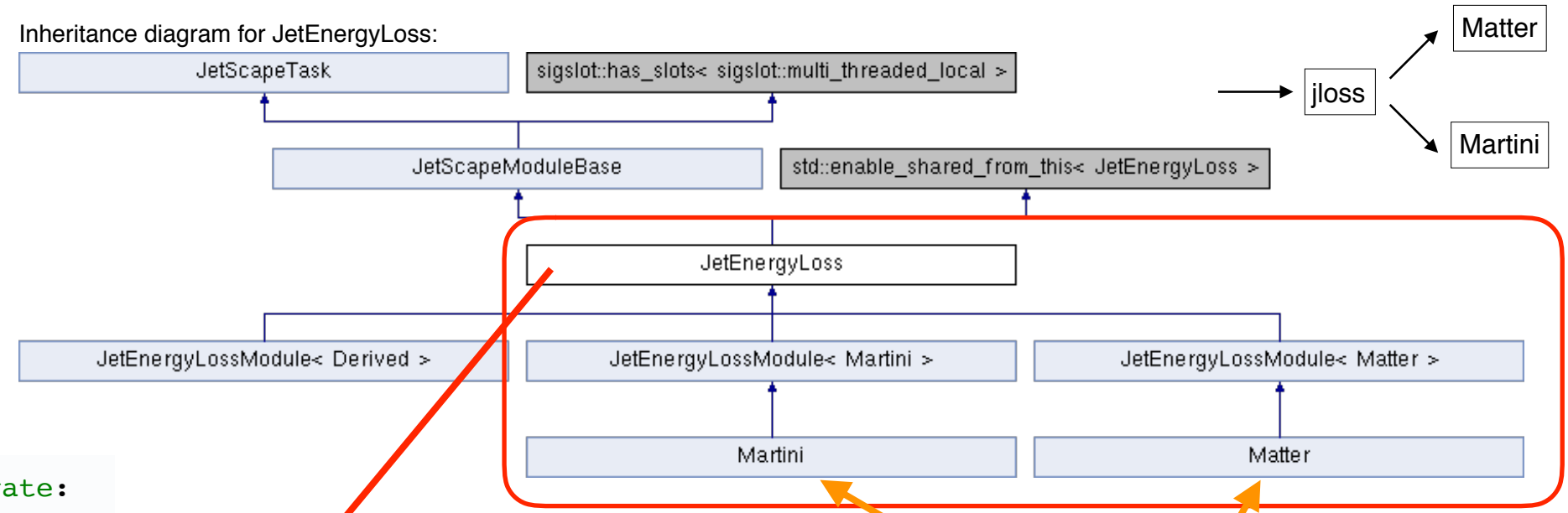
◆ SentInPartons

sigslot::signal4<double, double, vector<**Parton**>&, vector<**Parton**>&, multi_threaded_local> JetEnergyLoss::SentInPartons

Definition at line **56** of file **JetEnergyLoss.h**.

The “Do” Parton Shower in JetScapeEnergyLoss

Inheritance diagram for JetEnergyLoss:




```
private:
...
shared_ptr<PartonShower> pShower;
node vStart;
node vEnd;
void DoShower();
...
```

Recursive
(n) —> (m) parton “splitting”
routine in time steps deltaT

DoShower() sends a Signal

◆ SentInPartons

sigslot::signal4<double, double, vector<**Parton**>&, vector<**Parton**>&, multi_threaded_local> JetEnergyLoss::SentInPartons

Definition at line **56** of file **JetEnergyLoss.h**.

virtual void JetEnergyLoss::DoEnergyLoss (double deltaT,
double Q2,
vector< **Parton** > & pIn,
vector< **Parton** > & pOut
)

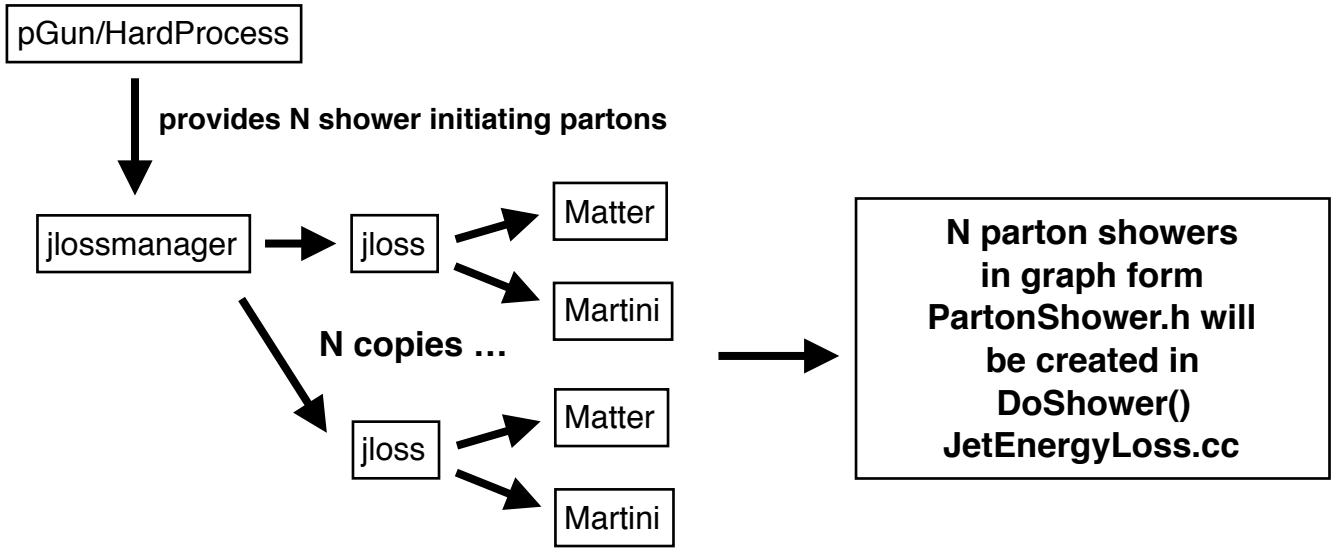
Slot(s)

Reimplemented in **Martini**, and **Matter**.

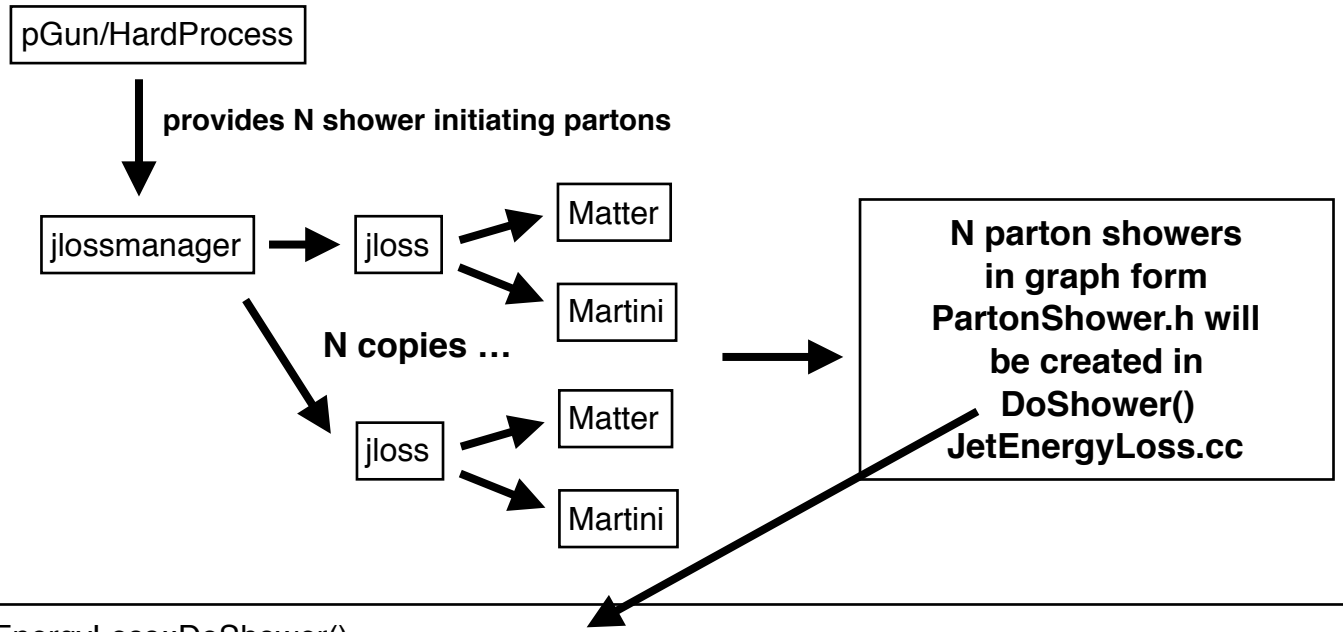
Definition at line **41** of file **JetEnergyLoss.h**.

JetEnergyLoss has signal/slots with Hydro: get energy density and sent energy deposition

DoShower() and Graph Generation in more detail …



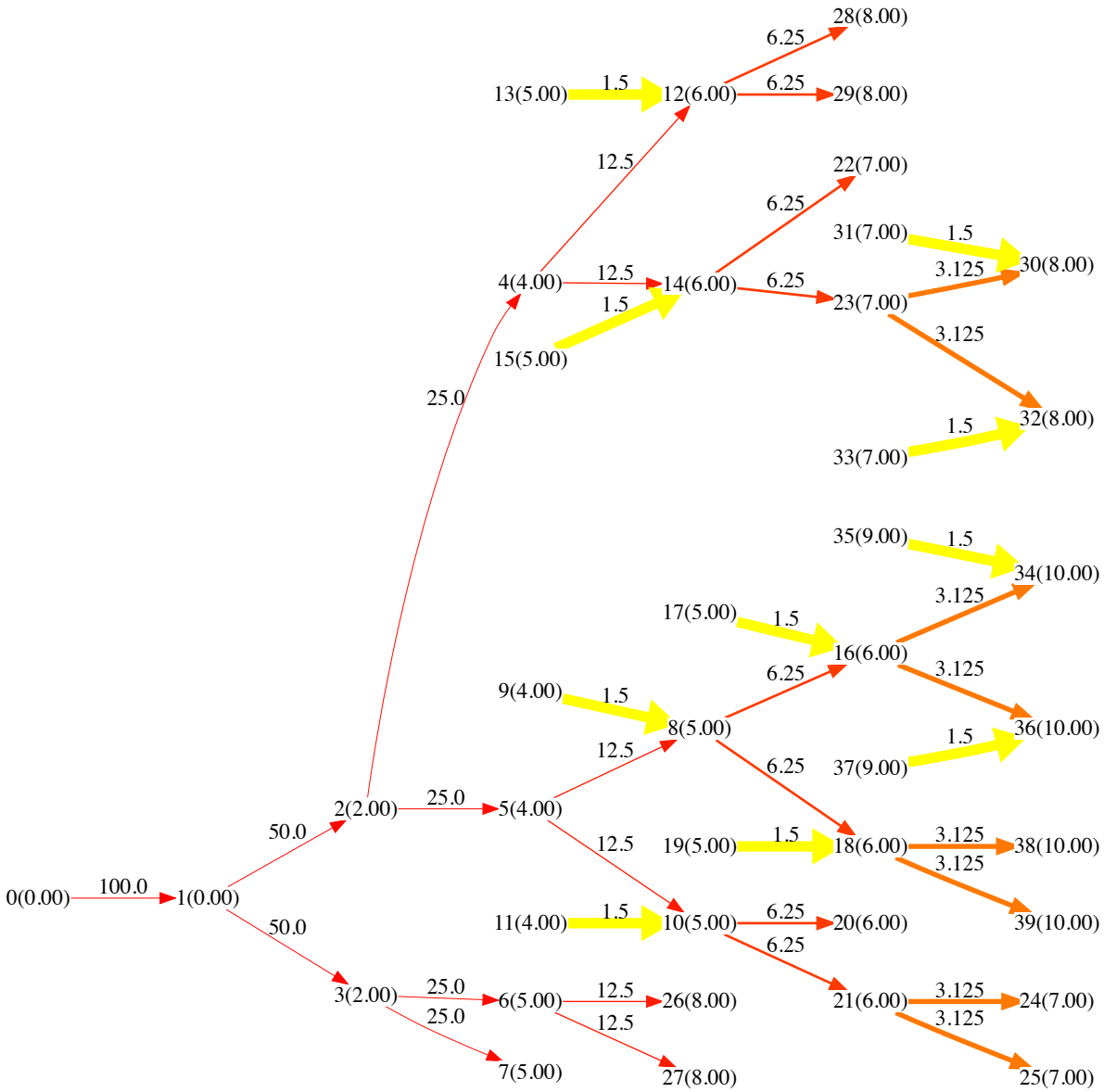
DoShower() and Graph Generation in more detail …




```
void::JetEnergyLoss::DoShower()
{
 vector<Parton> pIn; vector<Parton> pOut;
  …
  do
   {
    for (int i=0;i<pIn.size();i++)
    {
     …
     SentInPartons(currentTime,pIn[i].pt(),pInTempModule,pOutTemp); //Signal
     …
      for (int k=0;k<pOutTemp.size();k++)
      {
       vEnd=pShower->new_vertex(make_shared<VertexBase>(0,0,0,currentTime));
       pShower->new_parton(vStart,vEnd,make_shared<Parton>(pOutTemp[k]));
        …
      }
    }
  while (currentTime<maxT);
}
```

vector of partons used in DoShower() to recursively create the shower (Graph structure not directly used, "just" for storage)

Only vector of partons as input/output for loss modules

Dynamic generation of graph structure (Parton Shower.h) if something "happened" in an loss module …

A JetScape (non-physical) Parton Shower as a Graph …