

HepPDT Reference Manual
3.01.00

Generated by Doxygen 1.5.1-3

Thu Aug 23 14:01:10 2007

Contents

1 HepPDT Directory Hierarchy	1
1.1 HepPDT Directories	1
2 HepPDT Namespace Index	3
2.1 HepPDT Namespace List	3
3 HepPDT Hierarchical Index	5
3.1 HepPDT Class Hierarchy	5
4 HepPDT Class Index	7
4.1 HepPDT Class List	7
5 HepPDT File Index	9
5.1 HepPDT File List	9
6 HepPDT Directory Documentation	11
6.1 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/examples/ Directory Reference	11
6.2 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/tests/HepPDT/ Directory Reference . . .	12
6.3 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/examples/HepPDT/ Directory Reference	13
6.4 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/include/HepPDT/ Directory Reference .	14
6.5 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/src/HepPDT/ Directory Reference . . .	15
6.6 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/include/HepPID/ Directory Reference .	16
6.7 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/src/HepPID/ Directory Reference . . .	17
6.8 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/tests/HepPID/ Directory Reference . . .	18
6.9 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/examples/HepPID/ Directory Reference	19
6.10 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/include/ Directory Reference	20
6.11 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/src/ Directory Reference	21
6.12 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/tests/ Directory Reference	22
7 HepPDT Namespace Documentation	23

7.1	HepPDT Namespace Reference	23
7.2	HepPDT::detail Namespace Reference	33
7.3	HepPID Namespace Reference	37
7.4	std Namespace Reference	62
8	HepPDT Class Documentation	63
8.1	HepPDT::Constituent Class Reference	63
8.2	HepPDT::DefTable Class Reference	66
8.3	HepPDT::HeavyIonUnknownID Class Reference	69
8.4	HepPDT::Measurement Class Reference	71
8.5	HepPDT::ParticleData Class Reference	74
8.6	HepPDT::ParticleDataTable Class Reference	85
8.7	HepPDT::ParticleDataTableComparison Class Reference	92
8.8	HepPDT::ParticleID Class Reference	93
8.9	HepPID::ParticleNameMap Class Reference	102
8.10	HepPDT::ProcessUnknownID Class Reference	104
8.11	HepPDT::Quarks Struct Reference	106
8.12	HepPDT::ResonanceStructure Class Reference	108
8.13	HepPDT::SimpleProcessUnknownID Class Reference	112
8.14	HepPDT::SpinState Class Reference	113
8.15	HepPDT::TableBuilder Class Reference	116
8.16	HepPDT::TempAliasData Struct Reference	121
8.17	HepPDT::TempConstituent Struct Reference	123
8.18	HepPDT::TempDecayData Struct Reference	124
8.19	HepPDT::TempParticleData Struct Reference	125
9	HepPDT File Documentation	131
9.1	addEvtGenParticles.cc File Reference	131
9.2	addHerwigParticles.cc File Reference	132
9.3	addIsajetParticles.cc File Reference	133
9.4	addParticleTable.cc File Reference	134
9.5	addPDGParticles.cc File Reference	135
9.6	addPythiaParticles.cc File Reference	136
9.7	addQQParticles.cc File Reference	137
9.8	calculateWidthFromLifetime.cc File Reference	138
9.9	Constituent.cc File Reference	139
9.10	Constituent.hh File Reference	140

9.11	convertTemporaryMap.cc File Reference	141
9.12	DefTable.cc File Reference	142
9.13	DefTable.hh File Reference	143
9.14	examListHerwig.cc File Reference	144
9.15	examListIsajet.cc File Reference	146
9.16	examListPythia.cc File Reference	148
9.17	examMyPDT.cc File Reference	150
9.18	getIsajetID.cc File Reference	152
9.19	getPDGpid.cc File Reference	153
9.20	getPythiaid.cc File Reference	154
9.21	hasMethods.cc File Reference	155
9.22	HeavyIonUnknownID.cc File Reference	156
9.23	HeavyIonUnknownID.hh File Reference	157
9.24	lifetime.cc File Reference	158
9.25	list_of_examples.cc File Reference	159
9.26	list_of_tests.cc File Reference	160
9.27	listEvtGenNames.cc.in File Reference	161
9.28	listEvtGenTranslation.cc File Reference	162
9.29	listHerwigTranslation.cc File Reference	163
9.30	listIsajetTranslation.cc File Reference	164
9.31	listParticleNames.cc File Reference	165
9.32	listPDGNames.cc.in File Reference	166
9.33	listPDGTranslation.cc File Reference	167
9.34	listPythiaNames.cc.in File Reference	168
9.35	listPythiaTranslation.cc File Reference	169
9.36	listQQTranslation.cc File Reference	170
9.37	Measurement.hh File Reference	171
9.38	Measurement.icc File Reference	172
9.39	ParticleData.hh File Reference	173
9.40	ParticleData.icc File Reference	174
9.41	ParticleDataTable.cc File Reference	175
9.42	ParticleDataTable.hh File Reference	176
9.43	ParticleDataTable.icc File Reference	177
9.44	ParticleDataTableComparison.hh File Reference	178
9.45	ParticleID.cc File Reference	179
9.46	ParticleID.hh File Reference	180

9.47	ParticleID.icc File Reference	181
9.48	ParticleIDMethods.cc File Reference	182
9.49	ParticleIDMethods.hh File Reference	184
9.50	ParticleIDTranslations.hh File Reference	186
9.51	ParticleName.cc File Reference	188
9.52	ParticleName.hh File Reference	189
9.53	ProcessUnknownID.hh File Reference	190
9.54	quarks.cc File Reference	191
9.55	ResonanceStructure.cc File Reference	192
9.56	ResonanceStructure.hh File Reference	193
9.57	SimpleProcessUnknownID.hh File Reference	194
9.58	spindtoi.cc File Reference	195
9.59	spinitod.cc File Reference	196
9.60	SpinState.hh File Reference	197
9.61	SpinState.icc File Reference	198
9.62	stringtодouble.cc File Reference	199
9.63	stringtодouble.hh File Reference	200
9.64	TableBuilder.hh File Reference	201
9.65	TableBuilder.icc File Reference	203
9.66	TempParticleData.cc File Reference	204
9.67	TempParticleData.hh File Reference	205
9.68	testHepPDT.cc File Reference	206
9.69	testParticleIDMethods.cc File Reference	207
9.70	testPID.cc File Reference	208
9.71	testReadEvtGen.cc.in File Reference	209
9.72	testReadIsajet.cc.in File Reference	210
9.73	testReadParticleTable.cc.in File Reference	211
9.74	testReadQQ.cc.in File Reference	212
9.75	translateEvtGen.cc File Reference	213
9.76	translateGeanttoPDT.cc File Reference	214
9.77	translateHerwig.cc File Reference	215
9.78	translateIsajet.cc File Reference	216
9.79	translatePDG.cc File Reference	217
9.80	translatePDTtoGeant.cc File Reference	218
9.81	translatePythia.cc File Reference	219
9.82	translateQQ.cc File Reference	220

9.83 Version.cc File Reference	222
9.84 Version.cc File Reference	223
9.85 Version.hh File Reference	224
9.86 Version.hh File Reference	225
9.87 write.cc File Reference	226
10 HepPDT Example Documentation	227
10.1 examListHerwig.cc	227
10.2 examListHerwigInterface.F	229
10.3 examListIsajet.cc	231
10.4 examListIsajetInterface.F	233
10.5 examListPythia.cc	235
10.6 examListPythiaInterface.F	237
10.7 examMyPDT.cc	238
10.8 listEvtGenNames.cc.in	240
10.9 listEvtGenTranslation.cc	241
10.10listHerwigTranslation.cc	242
10.11listIsajetTranslation.cc	243
10.12listParticleNames.cc	244
10.13listPDGNames.cc.in	245
10.14listPDGTranslation.cc	246
10.15listPythiaNames.cc.in	247
10.16listPythiaTranslation.cc	248
10.17listQQTranslation.cc	249
10.18testHepPDT.cc	250
10.19testPID.cc	252
10.20testReadEvtGen.cc.in	254
10.21testReadIsajet.cc.in	255
10.22testReadQQ.cc.in	256

Chapter 1

HepPDT Directory Hierarchy

1.1 HepPDT Directories

This directory hierarchy is sorted roughly, but not completely, alphabetically:

examples	11
HepPDT	13
HepPID	19
include	20
HepPDT	14
HepPID	16
src	21
HepPDT	15
HepPID	17
tests	22
HepPDT	12
HepPID	18

Chapter 2

HepPDT Namespace Index

2.1 HepPDT Namespace List

Here is a list of all namespaces with brief descriptions:

HepPDT (HepPDT (p. 23) is a Particle Data Table namespace)	23
HepPDT::detail (HepPDT::detail (p. 33) is for internal use)	33
HepPID (The HepPID (p. 37) namespace has independent particle ID translation methods)	37
std	62

Chapter 3

HepPDT Hierarchical Index

3.1 HepPDT Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

HepPDT::Constituent	63
HepPDT::DefTable	66
HepPDT::Measurement	71
HepPDT::ParticleData	74
HepPDT::ParticleDataTable	85
HepPDT::ParticleDataTableComparison	92
HepPDT::ParticleID	93
HepPID::ParticleNameMap	102
HepPDT::ProcessUnknownID	104
HepPDT::HeavyIonUnknownID	69
HepPDT::SimpleProcessUnknownID	112
HepPDT::Quarks	106
HepPDT::ResonanceStructure	108
HepPDT::SpinState	113
HepPDT::TableBuilder	116
HepPDT::TempAliasData	121
HepPDT::TempConstituent	123
HepPDT::TempDecayData	124
HepPDT::TempParticleData	125

Chapter 4

HepPDT Class Index

4.1 HepPDT Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

HepPDT::Constituent (The Constituent (p. 63) class has information about constituent particles)	63
HepPDT::DefTable (The DefTable (p. 66) class holds EvtGen definitions)	66
HepPDT::HeavyIonUnknownID (The HeavyIonUnknownID (p. 69) class inherits from ProcessUnknownID (p. 104))	69
HepPDT::Measurement (The Measurement (p. 71) class defines a value with its error)	71
HepPDT::ParticleData (The ParticleData (p. 74) class holds basic particle data)	74
HepPDT::ParticleDataTable (The ParticleDataTable (p. 85) class is the core of HepPDT (p. 23))	85
HepPDT::ParticleDataTableComparison (The ParticleDataTableComparison (p. 92) class provides a utility for sorting the PDT)	92
HepPDT::ParticleID (The ParticleID (p. 93) has various utilities to extract information from the particle ID)	93
HepPID::ParticleNameMap	102
HepPDT::ProcessUnknownID (The ProcessUnknownID (p. 104) class is abstract)	104
HepPDT::Quarks (Constituent (p. 63) quarks)	106
HepPDT::ResonanceStructure (The ResonanceStructure (p. 108) class is holds mass and width information)	108
HepPDT::SimpleProcessUnknownID (The SimpleProcessUnknownID (p. 112) class inherits from ProcessUnknownID (p. 104))	112
HepPDT::SpinState (The SpinState (p. 113) class holds spin information)	113
HepPDT::TableBuilder (The TableBuilder (p. 116) class is used to construct a ParticleDataTable (p. 85))	116
HepPDT::TempAliasData (Hold Alias information from EvtGen)	121
HepPDT::TempConstituent (Temporary constituent (e.g., quark) information)	123
HepPDT::TempDecayData (Temporary holder for decay data)	124
HepPDT::TempParticleData (Temporary holder for Particle Data information)	125

Chapter 5

HepPDT File Index

5.1 HepPDT File List

Here is a list of all files with brief descriptions:

<code>addEvtGenParticles.cc</code>	131
<code>addHerwigParticles.cc</code>	132
<code>addIsajetParticles.cc</code>	133
<code>addParticleTable.cc</code>	134
<code>addPDGParticles.cc</code>	135
<code>addPythiaParticles.cc</code>	136
<code>addQQParticles.cc</code>	137
<code>calculateWidthFromLifetime.cc</code>	138
<code>Constituent.cc</code>	139
<code>Constituent.hh</code>	140
<code>convertTemporaryMap.cc</code>	141
<code>DefTable.cc</code>	142
<code>DefTable.hh</code>	143
<code>examListHerwig.cc</code>	144
<code>examListIsajet.cc</code>	146
<code>examListPythia.cc</code>	148
<code>examMyPDT.cc</code>	150
<code>getIsajetID.cc</code>	152
<code>getPDGpid.cc</code>	153
<code>getPythiaid.cc</code>	154
<code>hasMethods.cc</code>	155
<code>HeavyIonUnknownID.cc</code>	156
<code>HeavyIonUnknownID.hh</code>	157
<code>lifetime.cc</code>	158
<code>list_of_examples.cc</code>	159
<code>list_of_tests.cc</code>	160
<code>listEvtGenNames.cc.in</code>	161
<code>listEvtGenTranslation.cc</code>	162
<code>listHerwigTranslation.cc</code>	163
<code>listIsajetTranslation.cc</code>	164
<code>listParticleNames.cc</code>	165
<code>listPDGNames.cc.in</code>	166
<code>listPDGTranslation.cc</code>	167

listPythiaNames.cc.in	168
listPythiaTranslation.cc	169
listQQTranslation.cc	170
Measurement.hh	171
Measurement.icc	172
ParticleData.hh	173
ParticleData.icc	174
ParticleDataTable.cc	175
ParticleDataTable.hh	176
ParticleDataTable.icc	177
ParticleDataTableComparison.hh	178
ParticleID.cc	179
ParticleID.hh	180
ParticleID.icc	181
ParticleIDMethods.cc	182
ParticleIDMethods.hh	184
ParticleIDTranslations.hh	186
ParticleName.cc	188
ParticleName.hh	189
ProcessUnknownID.hh	190
quarks.cc	191
ResonanceStructure.cc	192
ResonanceStructure.hh	193
SimpleProcessUnknownID.hh	194
spindtoi.cc	195
spinitod.cc	196
SpinState.hh	197
SpinState.icc	198
stringtодouble.cc	199
stringtодouble.hh	200
TableBuilder.hh	201
TableBuilder.icc	203
TempParticleData.cc	204
TempParticleData.hh	205
testHepPDT.cc	206
testParticleIDMethods.cc	207
testPID.cc	208
testReadEvtGen.cc.in	209
testReadIsajet.cc.in	210
testReadParticleTable.cc.in	211
testReadQQ.cc.in	212
translateEvtGen.cc	213
translateGeanttoPDT.cc	214
translateHerwig.cc	215
translateIsajet.cc	216
translatePDG.cc	217
translatePDTtoGeant.cc	218
translatePythia.cc	219
translateQQ.cc	220
HepPDT/Version.cc	222
HepPID/Version.cc	223
HepPDT/Version.hh	224
HepPID/Version.hh	225
write.cc	226

Chapter 6

HepPDT Directory Documentation

6.1 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/examples/ Directory Reference

Directories

- directory **HepPDT**
- directory **HepPID**

Files

- file **list_of_examples.cc**

6.2 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/tests/Hep-PDT/ Directory Reference

Files

- file `listEvtGenNames.cc.in`
- file `listPDGNames.cc.in`
- file `listPythiaNames.cc.in`
- file `testHepPDT.cc`
- file `testPID.cc`
- file `testReadEvtGen.cc.in`
- file `testReadIsajet.cc.in`
- file `testReadParticleTable.cc.in`
- file `testReadQQ.cc.in`

6.3 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/examples/HepPDT/ Directory Reference

Files

- file **examMyPDT.cc**

6.4 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/include/HepPDT/ Directory Reference

Files

- file **Constituent.hh**
- file **DefTable.hh**
- file **HeavyIonUnknownID.hh**
- file **Measurement.hh**
- file **Measurement.icc**
- file **ParticleData.hh**
- file **ParticleData.icc**
- file **ParticleDataTable.hh**
- file **ParticleDataTable.icc**
- file **ParticleDataTableComparison.hh**
- file **ParticleID.hh**
- file **ParticleID.icc**
- file **ProcessUnknownID.hh**
- file **ResonanceStructure.hh**
- file **SimpleProcessUnknownID.hh**
- file **SpinState.hh**
- file **SpinState.icc**
- file **stringtodouble.hh**
- file **TableBuilder.hh**
- file **TableBuilder.icc**
- file **TempParticleData.hh**
- file **HepPDT/Version.hh**

6.5 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/src/HepPDT/ Directory Reference

Files

- file **addEvtGenParticles.cc**
- file **addHerwigParticles.cc**
- file **addIsajetParticles.cc**
- file **addParticleTable.cc**
- file **addPDGParticles.cc**
- file **addPythiaParticles.cc**
- file **addQQParticles.cc**
- file **calculateWidthFromLifetime.cc**
- file **Constituent.cc**
- file **convertTemporaryMap.cc**
- file **DefTable.cc**
- file **getIsajetID.cc**
- file **getPDGpid.cc**
- file **getPythiaid.cc**
- file **hasMethods.cc**
- file **HeavyIonUnknownID.cc**
- file **lifetime.cc**
- file **ParticleDataTable.cc**
- file **ParticleID.cc**
- file **quarks.cc**
- file **ResonanceStructure.cc**
- file **spindtoi.cc**
- file **spinitod.cc**
- file **stringtодouble.cc**
- file **TempParticleData.cc**
- file **HepPDT/Version.cc**
- file **write.cc**

6.6 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/include/Hep-PID/ Directory Reference

Files

- file **ParticleIDMethods.hh**
- file **ParticleIDTranslations.hh**
- file **ParticleName.hh**
- file **HepPID/Version.hh**

6.7 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/src/HepPID/ Directory Reference

Files

- file **ParticleIDMethods.cc**
- file **ParticleName.cc**
- file **translateEvtGen.cc**
- file **translateGeanttoPDT.cc**
- file **translateHerwig.cc**
- file **translateIsajet.cc**
- file **translatePDG.cc**
- file **translatePDTtoGeant.cc**
- file **translatePythia.cc**
- file **translateQQ.cc**
- file **HepPID/Version.cc**

6.8 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/tests/Hep-PID/ Directory Reference

Files

- file `listEvtGenTranslation.cc`
- file `listHerwigTranslation.cc`
- file `listIsajetTranslation.cc`
- file `listParticleNames.cc`
- file `listPDGTranslation.cc`
- file `listPythiaTranslation.cc`
- file `listQQTranslation.cc`
- file `testParticleIDMethods.cc`

6.9 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/examples/HepPID/ Directory Reference

Files

- file **examListHerwig.cc**
- file **examListIsajet.cc**
- file **examListPythia.cc**

6.10 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/include/ Directory Reference

Directories

- directory **HepPDT**
- directory **HepPID**

6.11 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/src/ Directory Reference

Directories

- directory **HepPDT**
- directory **HepPID**

6.12 /home/cepa01/garren/lcg/heppdt/HepPDT-3.01.00/tests/ Directory Reference

Directories

- directory **HepPDT**
- directory **HepPID**

Files

- file **list_of_tests.cc**

Chapter 7

HepPDT Namespace Documentation

7.1 HepPDT Namespace Reference

HepPDT (p. 23) is a Particle Data Table namespace.

Classes

- class **Constituent**

*The **Constituent** (p. 63) class has information about constituent particles.*

- class **DefTable**

*The **DefTable** (p. 66) class holds EvtGen definitions.*

- class **HeavyIonUnknownID**

*The **HeavyIonUnknownID** (p. 69) class inherits from **ProcessUnknownID** (p. 104).*

- class **Measurement**

*The **Measurement** (p. 71) class defines a value with its error.*

- class **ParticleData**

*The **ParticleData** (p. 74) class holds basic particle data.*

- class **ParticleDataTable**

*The **ParticleDataTable** (p. 85) class is the core of HepPDT (p. 23).*

- class **ParticleDataTableComparison**

*The **ParticleDataTableComparison** (p. 92) class provides a utility for sorting the PDT.*

- struct **Quarks**

constituent quarks

- class **ParticleID**

*The **ParticleID** (p. 93) has various utilities to extract information from the particle ID.*

- class **ProcessUnknownID**

*The **ProcessUnknownID** (p. 104) class is abstract.*

- class **ResonanceStructure**

*The **ResonanceStructure** (p. 108) class holds mass and width information.*

- class **SimpleProcessUnknownID**

*The **SimpleProcessUnknownID** (p. 112) class inherits from **ProcessUnknownID** (p. 104).*

- class **SpinState**

*The **SpinState** (p. 113) class holds spin information.*

- class **TableBuilder**

*The **TableBuilder** (p. 116) class is used to construct a **ParticleDataTable** (p. 85).*

- struct **TempConstituent**

Temporary constituent (e.g., quark) information.

- struct **TempDecayData**

temporary holder for decay data

- struct **TempAliasData**

Hold Alias information from EvtGen.

- struct **TempParticleData**

temporary holder for Particle Data information

Namespaces

- namespace **detail**

HepPDT::detail (p. 33) is for internal use.

TypeDefs

- typedef std::vector< **TempDecayData** > **TDDlist**

useful typedef

Enumerations

- enum **location** {

 nj = 1, **nq3**, **nq2**, **nq1**,

 nl, **nr**, **n**, **n8**,

 n9, **n10** }

Functions

- void **swap** (**Constituent** &first, **Constituent** &second)
- void **swap** (**Measurement** &first, **Measurement** &second)
- double **NaN** ()
- void **swap** (**ParticleData** &first, **ParticleData** &second)
- bool **writePDGStream** (std::ostream &os, const **ParticleDataTable** &table)
- bool **writePythiaStream** (std::ostream &os, const **ParticleDataTable** &table)
- bool **writeHerwigStream** (std::ostream &os, const **ParticleDataTable** &table)
- bool **writeIsajetStream** (std::ostream &os, const **ParticleDataTable** &table)
- bool **writeQQStream** (std::ostream &os, const **ParticleDataTable** &table)
- bool **writeEvtGenStream** (std::ostream &os, const **ParticleDataTable** &table)
- double **spinitod** (int js)

convert from 2J+1 to the actual spin value
- int **spindtoi** (double spin)

convert an actual spin to 2J+1
- void **swap** (**ParticleID** &first, **ParticleID** &second)
- void **swap** (**ResonanceStructure** &first, **ResonanceStructure** &second)
- void **swap** (**SpinState** &first, **SpinState** &second)
- double **stringtodouble** (std::string &nmb)

extract a double from a string
- bool **addPDGParticles** (std::istream &pdfile, **TableBuilder** &tb)

read PDG input and add particles to the table
- bool **addPythiaParticles** (std::istream &pdfile, **TableBuilder** &tb)

read Pythia input and add particles to the table
- bool **addHerwigParticles** (std::istream &pdfile, **TableBuilder** &tb)
- bool **addIsajetParticles** (std::istream &pdfile, **TableBuilder** &tb)

read Isajet particle input and add particles to the table
- bool **addIsajetDecay** (std::istream &pdfile, **TableBuilder** &tb)

read Isajet decay input and add decay information to the table
- bool **addQQParticles** (std::istream &pdfile, **TableBuilder** &tb)

read QQ input and add particles to the table
- bool **addEvtGenParticles** (std::istream &pdfile, **TableBuilder** &tb)

read EvtGen input and add particles to the table
- bool **addParticleTable** (std::istream &pdfile, **TableBuilder** &tb)

read particle.tbl (or something similar) and add particles to the table
- double **calculateWidthFromLifetime** (double)
- void **swap** (**TempParticleData** &first, **TempParticleData** &second)
- void **version** ()

print HepPDT (p. 23) version

- void **writeVersion** (std::ostream &os)
write HepPDT (p. 23) version to os
- std::string **versionName** ()
return HepPDT (p. 23) version
- bool **getEvtGenLineType** (std::string <ype, int &id, std::string &name, const std::string &pdline)
- void **parseEvtGenLine** (TempParticleData &tpd, const std::string &pdline)
- void **parseEvtGenAlias** (TempAliasData &tad, const std::string &pdline)
- bool **parseEvtGenDecayLine** (TempParticleData &tpd, const std::string &pdline)
- bool **parseEvtGenAliasDecayLine** (TempAliasData &tad, const std::string &pdline)
- void **parseEvtGenConj** (std::string &cname, const std::string &pdline)
- void **parseEvtGenDefinition** (std::string &def, double &val, const std::string &pdline)
- bool **getQQLineType** (std::string <ype, int &id, std::string &name, const std::string &pdline)
- bool **parseQQDecayLine** (const std::string &pdline)
- void **parseQQParticle** (TempParticleData &tpd, const std::string &pdline)

7.1.1 Detailed Description

HepPDT (p. 23) is a Particle Data Table namespace.

The HepPDT (p. 23) classes are used to create a Particle Data Table.

7.1.2 Typedef Documentation

7.1.2.1 `typedef std::vector<TempDecayData> HepPDT::TDDlist`

useful typedef

Definition at line 45 of file TempParticleData.hh.

7.1.3 Enumeration Type Documentation

7.1.3.1 `enum HepPDT::location`

PID digits (base 10) are: n nr nl nq1 nq2 nq3 nj The location enum provides a convenient index into the PID.

Enumerator:

```
nj
nq3
nq2
nq1
nl
nr
n
n8
```

n9

n10

Definition at line 36 of file ParticleID.hh.

7.1.4 Function Documentation

7.1.4.1 **bool HepPDT::addEvtGenParticles (std::istream & *pdfile*, TableBuilder & *tb*)**

read EvtGen input and add particles to the table

Examples:

listEvtGenNames.cc.in, and **testReadEvtGen.cc.in**.

Definition at line 29 of file addEvtGenParticles.cc.

References HepPDT::TableBuilder::addAlias(), HepPDT::DefTable::addDefinition(), HepPDT::TableBuilder::addParticle(), HepPDT::TableBuilder::aliasData(), HepPDT::TableBuilder::aliasSize(), HepPDT::TableBuilder::definitions(), getEvtGenLineType(), HepPDT::TableBuilder::getParticleData(), HepPDT::TableBuilder::hasAlias(), HepPDT::TableBuilder::hasParticleData(), parseEvtGenAlias(), parseEvtGenAliasDecayLine(), parseEvtGenConj(), parseEvtGenDecayLine(), parseEvtGenDefinition(), parseEvtGenLine(), HepPDT::TableBuilder::size(), HepPDT::TempAliasData::tempChargeConj, and HepPID::translateEvtGentoPDT().

Referenced by main().

7.1.4.2 **bool HepPDT::addHerwigParticles (std::istream & *pdfile*, TableBuilder & *tb*)**

Definition at line 13 of file addHerwigParticles.cc.

7.1.4.3 **bool HepPDT::addIsajetDecay (std::istream & *pdfile*, TableBuilder & *tb*)**

read Isajet decay input and add decay information to the table

7.1.4.4 **bool HepPDT::addIsajetParticles (std::istream & *pdfile*, TableBuilder & *tb*)**

read Isajet particle input and add particles to the table

Examples:

testReadIsajet.cc.in.

Definition at line 14 of file addIsajetParticles.cc.

References HepPDT::detail::getIsajetID(), HepPDT::TableBuilder::getParticleData(), HepPDT::detail::parseIsajetLine(), HepPDT::TableBuilder::size(), and HepPID::translateIsajettoPDT().

Referenced by main().

7.1.4.5 **bool HepPDT::addParticleTable (std::istream & *pdfile*, TableBuilder & *tb*)**

read particle.tbl (or something similar) and add particles to the table

Definition at line 21 of file addParticleTable.cc.

References HepPDT::TableBuilder::getParticleData(), HepPDT::detail::getParticleID(), HepPDT::ParticleID::isValid(), HepPDT::detail::parseParticleLine(), and HepPDT::TableBuilder::size().

Referenced by main().

7.1.4.6 **bool HepPDT::addPDGParticles (std::istream & *pdfile*, TableBuilder & *tb*)**

read PDG input and add particles to the table

Examples:

listPDGNames.cc.in, and **testHepPDT.cc**.

Definition at line 22 of file addPDGParticles.cc.

References HepPDT::TableBuilder::getParticleData(), HepPDT::detail::getPDGnames(), HepPDT::detail::getPDGpid(), HepPDT::detail::parsePDGline(), HepPDT::TableBuilder::size(), and HepPDT::TempParticleData::tempSource.

Referenced by main().

7.1.4.7 **bool HepPDT::addPythiaParticles (std::istream & *pdfile*, TableBuilder & *tb*)**

read Pythia input and add particles to the table

Examples:

listPythiaNames.cc.in.

Definition at line 20 of file addPythiaParticles.cc.

References HepPDT::TableBuilder::getAntiParticle(), HepPDT::TableBuilder::getParticleData(), HepPDT::detail::getPythiaid(), HepPDT::detail::parsePythiaDecayLine(), HepPDT::detail::parsePythiaLine(), HepPDT::TableBuilder::size(), HepPDT::TempParticleData::tempMass, HepPDT::TempParticleData::tempOriginalID, HepPDT::TempParticleData::tempSource, and HepPID::translatePythiaToPDT().

Referenced by main().

7.1.4.8 **bool HepPDT::addQQParticles (std::istream & *pdfile*, TableBuilder & *tb*)**

read QQ input and add particles to the table

Examples:

testReadQQ.cc.in.

Definition at line 27 of file addQQParticles.cc.

References HepPDT::TableBuilder::addParticle(), HepPDT::TableBuilder::getParticleData(), getQQLineType(), HepPDT::TableBuilder::hasParticleData(), parseQQDecayLine(), parseQQParticle(),

HepPDT::TableBuilder::size(), HepPDT::TempParticleData::tempOriginalID, HepPDT::TempParticleData::tempParticleName, HepPDT::TempParticleData::tempSource, HepPID::translateQQbar(), and HepPID::translateQQtoPDT().

Referenced by main().

7.1.4.9 double HepPDT::calculateWidthFromLifetime (double)

free function Given the lifetime, calculate the width.

Definition at line 13 of file calculateWidthFromLifetime.cc.

Referenced by parseEvtGenLine(), HepPDT::detail::parseParticleLine(), HepPDT::detail::parsePythiaLine(), and parseQQParticle().

7.1.4.10 bool HepPDT::getEvtGenLineType (std::string & ltype, int & id, std::string & name, const std::string & pdline)

Definition at line 104 of file addEvtGenParticles.cc.

Referenced by addEvtGenParticles().

7.1.4.11 bool HepPDT::getQQLineType (std::string & ltype, int & id, std::string & name, const std::string & pdline)

Definition at line 76 of file addQQParticles.cc.

Referenced by addQQParticles().

7.1.4.12 double HepPDT::NaN () [inline]

Definition at line 11 of file Measurement.icc.

7.1.4.13 void HepPDT::parseEvtGenAlias (TempAliasData & tad, const std::string & pdline)

Definition at line 222 of file addEvtGenParticles.cc.

References HepPDT::TempAliasData::tempAlias, and HepPDT::TempAliasData::tempAliasedParticle.

Referenced by addEvtGenParticles().

7.1.4.14 bool HepPDT::parseEvtGenAliasDecayLine (TempAliasData & tad, const std::string & pdline)

Definition at line 201 of file addEvtGenParticles.cc.

Referenced by addEvtGenParticles().

7.1.4.15 void HepPDT::parseEvtGenConj (std::string & cname, const std::string & pdline)

Definition at line 239 of file addEvtGenParticles.cc.

Referenced by addEvtGenParticles().

7.1.4.16 bool HepPDT::parseEvtGenDecayLine (TempParticleData & *tpd*, const std::string & *pdline*)

Definition at line 172 of file addEvtGenParticles.cc.

References HepPDT::ParticleID::pid(), and HepPDT::TempParticleData::tempID.

Referenced by addEvtGenParticles().

7.1.4.17 void HepPDT::parseEvtGenDefinition (std::string & *def*, double & *val*, const std::string & *pdline*)

Definition at line 254 of file addEvtGenParticles.cc.

Referenced by addEvtGenParticles().

7.1.4.18 void HepPDT::parseEvtGenLine (TempParticleData & *tpd*, const std::string & *pdline*)

Definition at line 136 of file addEvtGenParticles.cc.

References calculateWidthFromLifetime(), HepPDT::ParticleID::pid(), HepPDT::SpinState::setTotalSpin(), HepPDT::TempParticleData::tempCharge, HepPDT::TempParticleData::tempHighCutoff, HepPDT::TempParticleData::tempID, HepPDT::TempParticleData::tempMass, HepPDT::TempParticleData::tempOriginalID, HepPDT::TempParticleData::tempParticleName, HepPDT::TempParticleData::tempSource, HepPDT::TempParticleData::tempSpin, HepPDT::TempParticleData::tempWidth, and HepPDT::SpinState::totalSpin().

Referenced by addEvtGenParticles().

7.1.4.19 bool HepPDT::parseQQDecayLine (const std::string & *pdline*)

Definition at line 158 of file addQQParticles.cc.

Referenced by addQQParticles().

7.1.4.20 void HepPDT::parseQQParticle (TempParticleData & *tpd*, const std::string & *pdline*)

Definition at line 117 of file addQQParticles.cc.

References calculateWidthFromLifetime(), HepPDT::ParticleID::pid(), HepPDT::SpinState::setTotalSpin(), HepPDT::TempParticleData::tempCharge, HepPDT::TempParticleData::tempHighCutoff, HepPDT::TempParticleData::tempID, HepPDT::TempParticleData::tempLowCutoff, HepPDT::TempParticleData::tempMass, HepPDT::TempParticleData::tempOriginalID, HepPDT::TempParticleData::tempParticleName, HepPDT::TempParticleData::tempSpin, HepPDT::TempParticleData::tempWidth, and HepPDT::SpinState::totalSpin().

Referenced by addQQParticles().

7.1.4.21 int HepPDT::spindtoi (double *spin*)

convert an actual spin to 2J+1

Definition at line 13 of file spindtoi.cc.

7.1.4.22 double HepPDT::spinitod (int *js*)

convert from 2J+1 to the actual spin value

Examples:

testPID.cc.

Definition at line 13 of file spinitod.cc.

Referenced by main(), and HepPDT::TempParticleData::processPID().

7.1.4.23 double HepPDT::stringtodouble (std::string & *numb*)

extract a double from a string

Definition at line 15 of file stringtodouble.cc.

7.1.4.24 void HepPDT::swap (TempParticleData & *first*, TempParticleData & *second*) [inline]

Definition at line 106 of file TempParticleData.hh.

References HepPDT::TempParticleData::swap().

Referenced by HepPDT::TempParticleData::swap(), HepPDT::SpinState::swap(), HepPDT::ResonanceStructure::swap(), HepPDT::ParticleID::swap(), HepPDT::ParticleData::swap(), HepPDT::Measurement::swap(), and HepPDT::Constituent::swap().

7.1.4.25 void HepPDT::swap (SpinState & *first*, SpinState & *second*) [inline]

Definition at line 69 of file SpinState.hh.

References HepPDT::SpinState::swap().

7.1.4.26 void HepPDT::swap (ResonanceStructure & *first*, ResonanceStructure & *second*) [inline]

Definition at line 83 of file ResonanceStructure.hh.

References HepPDT::ResonanceStructure::swap().

7.1.4.27 void HepPDT::swap (ParticleID & *first*, ParticleID & *second*) [inline]

Definition at line 159 of file ParticleID.hh.

References HepPDT::ParticleID::swap().

7.1.4.28 void HepPDT::swap (ParticleData & *first*, ParticleData & *second*) [inline]

Definition at line 169 of file ParticleData.hh.

References HepPDT::ParticleData::swap().

7.1.4.29 void HepPDT::swap (Measurement & *first*, Measurement & *second*) [inline]

Definition at line 45 of file Measurement.hh.

References HepPDT::Measurement::swap().

7.1.4.30 void HepPDT::swap (Constituent & *first*, Constituent & *second*) [inline]

Definition at line 77 of file Constituent.hh.

References HepPDT::Constituent::swap().

7.1.4.31 void HepPDT::version ()

print **HepPDT** (p. 23) version

Definition at line 20 of file HepPDT/Version.cc.

References versionName().

Referenced by HepPDT::ParticleDataTable::ParticleDataTable().

7.1.4.32 std::string HepPDT::versionName ()

return **HepPDT** (p. 23) version

Definition at line 15 of file HepPDT/Version.cc.

Referenced by version(), and writeVersion().

7.1.4.33 bool HepPDT::writeEvtGenStream (std::ostream & *os*, const ParticleDataTable & *table*)**7.1.4.34 bool HepPDT::writeHerwigStream (std::ostream & *os*, const ParticleDataTable & *table*)****7.1.4.35 bool HepPDT::writeIsajetStream (std::ostream & *os*, const ParticleDataTable & *table*)****7.1.4.36 bool HepPDT::writePDGStream (std::ostream & *os*, const ParticleDataTable & *table*)****7.1.4.37 bool HepPDT::writePythiaStream (std::ostream & *os*, const ParticleDataTable & *table*)****7.1.4.38 bool HepPDT::writeQQStream (std::ostream & *os*, const ParticleDataTable & *table*)****7.1.4.39 void HepPDT::writeVersion (std::ostream & *os*)**

write **HepPDT** (p. 23) version to *os*

Definition at line 26 of file HepPDT/Version.cc.

References versionName().

Referenced by HepPDT::ParticleDataTable::writeParticleData(), and HepPDT::ParticleDataTable::writeParticleInfo().

7.2 HepPDT::detail Namespace Reference

HepPDT::detail (p. 33) is for internal use.

Functions

- void **getPDGpid** (std::vector< int > &idlist, std::string &pdline)
for internal use
- void **getPDGnames** (std::vector< std::string > &namelst, std::string &pdline)
for internal use
- void **parsePDGline** (TempParticleData &tpd, std::string &pdline)
for internal use
- bool **CheckPDGEntry** (TempParticleData &tpd, const std::string &, double, double)
for internal use
- bool **getPythiaid** (int &id, const std::string &pdline)
for internal use
- void **parsePythiaLine** (TempParticleData &tpd, int &anti, std::string &aname, const std::string &pdline)
for internal use
- void **parsePythiaDecayLine** (TempParticleData &tpd, const std::string &pdline)
for internal use
- TempDecayData **getPythiaDecay** (const std::string &pdline)
for internal use
- bool **getIsajetID** (int &id, const std::string &pdline)
for internal use
- void **parseIsajetLine** (TempParticleData &tpd, const std::string &pdline)
for internal use
- void **parseIsajetDecayLine** (TempParticleData &tpd, const std::string &pdline, TableBuilder &tb)
for internal use
- bool **getParticleID** (int &id, const std::string &pdline)
for internal use
- void **parseParticleLine** (TempParticleData &tpd, const std::string &pdline)
for internal use

7.2.1 Detailed Description

HepPDT::detail (p. 33) is for internal use.

This namespace encapsulates free functions used when parsing various input streams.

7.2.2 Function Documentation

7.2.2.1 **bool HepPDT::detail::CheckPDGEntry (TempParticleData & *tpd*, const std::string &, double, double)**

for internal use

Definition at line 67 of file addPDGParticles.cc.

References HepPDT::TempParticleData::tempMass, and HepPDT::TempParticleData::tempWidth.

Referenced by parsePDGline().

7.2.2.2 **bool HepPDT::detail::getIsajetID (int & *id*, const std::string & *pdline*)**

for internal use

Definition at line 17 of file getIsajetID.cc.

Referenced by HepPDT::addIsajetParticles().

7.2.2.3 **bool HepPDT::detail::getParticleID (int & *id*, const std::string & *pdline*)**

for internal use

Definition at line 42 of file addParticleTable.cc.

Referenced by HepPDT::addParticleTable().

7.2.2.4 **void HepPDT::detail::getPDGnames (std::vector< std::string > & *namelist*, std::string & *pdline*)**

for internal use

Definition at line 40 of file getPDGpid.cc.

Referenced by HepPDT::addPDGParticles().

7.2.2.5 **void HepPDT::detail::getPDGpid (std::vector< int > & *idlist*, std::string & *pdline*)**

for internal use

Definition at line 20 of file getPDGpid.cc.

Referenced by HepPDT::addPDGParticles().

7.2.2.6 **TempDecayData HepPDT::detail::getPythiaDecay (const std::string & *pdline*)**

for internal use

7.2.2.7 bool HepPDT::detail::getPythiaid (int & id, const std::string & pdline)

for internal use

Definition at line 20 of file getPythiaid.cc.

Referenced by HepPDT::addPythiaParticles().

7.2.2.8 void HepPDT::detail::parseIsajetDecayLine (TempParticleData & tpd, const std::string & pdline, TableBuilder & tb)

for internal use

7.2.2.9 void HepPDT::detail::parseIsajetLine (TempParticleData & tpd, const std::string & pdline)

for internal use

Definition at line 33 of file addIsajetParticles.cc.

References HepPDT::ParticleID::pid(), HepPDT::TempParticleData::tempCharge, HepPDT::TempParticleData::tempID, HepPDT::TempParticleData::tempMass, HepPDT::TempParticleData::tempName, and HepPDT::TempParticleData::tempSpin.

Referenced by HepPDT::addIsajetParticles().

7.2.2.10 void HepPDT::detail::parseParticleLine (TempParticleData & tpd, const std::string & pdline)

for internal use

Definition at line 62 of file addParticleTable.cc.

References HepPDT::calculateWidthFromLifetime(), HepPID::particleName(), HepPDT::ParticleID::pid(), HepPDT::TempParticleData::tempCharge, HepPDT::TempParticleData::tempID, HepPDT::TempParticleData::tempMass, HepPDT::TempParticleData::tempOriginalID, HepPDT::TempParticleData::tempParticleName, HepPDT::TempParticleData::tempSource, and HepPDT::TempParticleData::tempWidth.

Referenced by HepPDT::addParticleTable().

7.2.2.11 void HepPDT::detail::parsePDGline (TempParticleData & tpd, std::string & pdline)

for internal use

Definition at line 51 of file addPDGParticles.cc.

References CheckPDGEntry().

Referenced by HepPDT::addPDGParticles().

7.2.2.12 void HepPDT::detail::parsePythiaDecayLine (TempParticleData & tpd, const std::string & pdline)

for internal use

Definition at line 98 of file addPythiaParticles.cc.

References HepPDT::ParticleID::pid(), and HepPDT::TempParticleData::tempID.

Referenced by HepPDT::addPythiaParticles().

7.2.2.13 void HepPDT::detail::parsePythiaLine (TempParticleData & *tpd*, int & *anti*, std::string & *aname*, const std::string & *ppline*)

for internal use

Definition at line 56 of file addPythiaParticles.cc.

References HepPDT::calculateWidthFromLifetime(), HepPDT::ParticleID::pid(), HepPDT::TempParticleData::tempCharge, HepPDT::TempParticleData::tempColorCharge, HepPDT::TempParticleData::tempHighCutoff, HepPDT::TempParticleData::tempID, HepPDT::TempParticleData::tempMass, HepPDT::TempParticleData::tempOriginalID, HepPDT::TempParticleData::tempParticleName, HepPDT::TempParticleData::tempSource, and HepPDT::TempParticleData::tempWidth.

Referenced by HepPDT::addPythiaParticles().

7.3 HepPID Namespace Reference

The **HepPID** (p. 37) namespace has independent particle ID translation methods.

Classes

- class **ParticleNameMap**

TypeDefs

- typedef std::map< int, std::string > **ParticleIdMap**
- typedef std::map< std::string, int > **ParticleLookupMap**
- typedef std::map< int, int > **EvtGenPDTMap**
- typedef std::map< int, int > **PDTEvtGenMap**
- typedef std::map< int, int > **HerwigPDTMap**
- typedef std::map< int, int > **PDTHerwigMap**
- typedef std::map< int, int > **IsajetPDTMap**
- typedef std::map< int, int > **PDTIsajetMap**
- typedef std::map< int, int > **PDGtoPDTMap**
- typedef std::map< int, int > **PDTtoPDGMap**
- typedef std::map< int, int > **PythiaPDTMap**
- typedef std::map< int, int > **PDTPythiaMap**
- typedef std::map< int, int > **QQPDTMap**
- typedef std::map< int, int > **PDTQQMap**
- typedef std::map< int, int > **QQbarMap**
- typedef std::map< int, int > **InverseQQbarMap**

Enumerations

- enum **location** {
 nj = 1, **nq3**, **nq2**, **nq1**,
nl, **nr**, **n**, **n8**,
n9, **n10** }

Functions

- unsigned short **digit** (**location** loc, const int &pid)
return the digit at a named location in the PID
- int **A** (const int &pid)
- int **Z** (const int &pid)
- int **lambda** (const int &pid)
- int **abspid** (const int &pid)
absolute value of particle ID
- int **fundamentalID** (const int &pid)
extract fundamental ID (1-100) if this is a "fundamental" particle

- bool **hasFundamentalAnti** (const int &pid)
if this is a fundamental particle, does it have a valid antiparticle?
- int **extraBits** (const int &pid)
- bool **isValid** (const int &pid)
is this a valid ID?
- bool **isMeson** (const int &pid)
is this a valid meson ID?
- bool **isBaryon** (const int &pid)
is this a valid baryon ID?
- bool **isDiQuark** (const int &pid)
is this a valid diquark ID?
- bool **isHadron** (const int &pid)
is this a valid hadron ID?
- bool **isLepton** (const int &pid)
is this a valid lepton ID?
- bool **isNucleus** (const int &pid)
is this a valid ion ID?
- bool **isPentaquark** (const int &pid)
is this a valid pentaquark ID?
- bool **isSUSY** (const int &pid)
is this a valid SUSY ID?
- bool **isRhadron** (const int &pid)
is this a valid R-hadron ID?
- bool **hasUp** (const int &pid)
does this particle contain an up quark?
- bool **hasDown** (const int &pid)
does this particle contain a down quark?
- bool **hasStrange** (const int &pid)
does this particle contain a strange quark?
- bool **hasCharm** (const int &pid)
does this particle contain a charm quark?
- bool **hasBottom** (const int &pid)
does this particle contain a bottom quark?

- bool **hasTop** (const int &pid)
does this particle contain a top quark?
- int **jSpin** (const int &pid)
jSpin returns $2J+1$, where J is the total spin
- int **sSpin** (const int &pid)
sSpin returns $2S+1$, where S is the spin
- int **lSpin** (const int &pid)
lSpin returns $2L+1$, where L is the orbital angular momentum
- int **threeCharge** (const int &pid)
return 3 times the charge (3 x quark charge is an int)
- int **translateHerwigtoPDT** (const int herwigID)
translate Herwig to PDG standard
- int **translatePDTtoHerwig** (const int pid)
translate PDG standard to Herwig
- void **writeHerwigTranslation** (std::ostream &os)
output the translation list
- int **translateIsajettoPDT** (const int isajetID)
translate Isajet to PDG standard
- int **translatePDTtoIsajet** (const int pid)
translate PDG standard to Isajet
- void **writeIsajetTranslation** (std::ostream &os)
output the translation list
- int **translatePythiatoPDT** (const int pythiaID)
translate Pythia to PDG standard
- int **translatePDTtoPythia** (const int pid)
translate PDG standard to Pythia
- void **writePythiaTranslation** (std::ostream &os)
output the translation list
- int **translateEvtGentoPDT** (const int evtGenID)
translate EvtGen to PDG standard
- int **translatePDTtoEvtGen** (const int pid)
translate PDG standard to EvtGen
- void **writeEvtGenTranslation** (std::ostream &os)
output the translation list

- int **translatePDGtabletoPDT** (const int pdgID)
translate PDG table to PDG standard
- int **translatePDTtoPDGtable** (const int pid)
translate PDG standard to PDG table
- void **writePDGTranslation** (std::ostream &os)
output the translation list
- int **translateQQtoPDT** (const int qqID)
translate QQ to PDG standard
- int **translatePDTtoQQ** (const int pid)
translate PDG standard to QQ
- int **translateQQbar** (const int id)
QQ helper function.
- int **translateInverseQQbar** (const int id)
QQ helper function.
- void **writeQQTranslation** (std::ostream &os)
output the translation list
- int **translateGeanttoPDT** (const int geantID)
translate Geant3 to PDG standard
- int **translatePDTtoGeant** (const int pid)
translate PDG standard to Geant3
- std::string **particleName** (const int &)
get a known HepPID (p. 37) Particle name
- int **particleName** (const std::string &)
lookup a known ID
- void **listParticleNames** (std::ostream &os)
list all known names
- bool **validParticleName** (const int &)
verify that this number has a valid name
- bool **validParticleName** (const std::string &)
verify that this string has a valid id
- **ParticleNameMap** const & **getParticleNameMap** ()
access the ParticleNameMap (p. 102) for other purposes
- void **version** ()

print HepPID (p. 37) version

- void **writeVersion** (std::ostream &os)
write HepPID (p. 37) version to os
- std::string **versionName** ()
return HepPID (p. 37) version
- **ParticleNameMap** const & **ParticleNameInit** ()
- void **writeParticleNameLine** (int i, std::ostream &os)
- **EvtGenPDTMap** const & **getEvtGenPDTMap** ()
- **PDTEvtGenMap** const & **getPDTEvtGenMap** ()
- **EvtGenPDTMap** const & **EvtGenPDTMapInit** ()
- **PDTEvtGenMap** const & **PDTEvtGenMapInit** ()
- **EvtGenPDTMap** const & **getEvtGenPDTMap** ()
- **PDTEvtGenMap** const & **getPDTEvtGenMap** ()
- void **writeEvtGenTranslationLine** (int i, std::ostream &os)
- **HerwigPDTMap** const & **getHerwigPDTMap** ()
- **PDTHerwigMap** const & **getPDTHerwigMap** ()
- **HerwigPDTMap** const & **HerwigPDTMapInit** ()
- **PDTHerwigMap** const & **PDTHerwigMapInit** ()
- **HerwigPDTMap** const & **getHerwigPDTMap** ()
- **PDTHerwigMap** const & **getPDTHerwigMap** ()
- void **writeHerwigTranslationLine** (int i, std::ostream &os)
- **IsajetPDTMap** const & **getIsajetPDTMap** ()
- **PDTIsajetMap** const & **getPDTIsajetMap** ()
- **IsajetPDTMap** const & **IsajetPDTMapInit** ()
- **PDTIsajetMap** const & **PDTIsajetMapInit** ()
- int **convIsajettoPDT** (const int id)
- int **convPDTtoIsajet** (const int id)
- **IsajetPDTMap** const & **getIsajetPDTMap** ()
- **PDTIsajetMap** const & **getPDTIsajetMap** ()
- void **writeIsajetTranslationLine** (int i, std::ostream &os)
- **PDGtoPDTMap** const & **getPDGtoPDTMap** ()
- **PDTtoPDGMap** const & **getPDTtoPDGMap** ()
- **PDGtoPDTMap** const & **PDGtoPDTMapInit** ()
- **PDTtoPDGMap** const & **PDTtoPDGMapInit** ()
- **PDGtoPDTMap** const & **getPDGtoPDTMap** ()
- **PDTtoPDGMap** const & **getPDTtoPDGMap** ()
- void **writePDGTranslationLine** (int i, std::ostream &os)
- **PythiaPDTMap** const & **getPythiaPDTMap** ()
- **PDTPythiaMap** const & **getPDTPythiaMap** ()
- **PythiaPDTMap** const & **PythiaPDTMapInit** ()
- **PDTPythiaMap** const & **PDTPythiaMapInit** ()
- **PythiaPDTMap** const & **getPythiaPDTMap** ()
- **PDTPythiaMap** const & **getPDTPythiaMap** ()
- void **writePythiaTranslationLine** (int i, std::ostream &os)
- **QQPDTMap** const & **getQQPDTMap** ()
- **PDTQQMap** const & **getPDTQQMap** ()
- **QQbarMap** const & **getQQbarMap** ()

- `InverseQQbarMap const & getInverseQQbarMap ()`
- `QPDTMap const & QQPDTMapInit ()`
- `QQbarMap const & QQbarMapInit ()`
- `PDTQQMap const & PDTQQMapInit ()`
- `InverseQQbarMap const & InverseQQbarMapInit ()`
- `QPDTMap const & getQPDTMap ()`
- `PDTQQMap const & getPDTQQMap ()`
- `QQbarMap const & getQQbarMap ()`
- `InverseQQbarMap const & getInverseQQbarMap ()`

7.3.1 Detailed Description

The **HepPID** (p. 37) namespace has independent particle ID translation methods.

The **HepPID** (p. 37) namespace contains a set of independent particle ID translation methods

7.3.2 Typedef Documentation

7.3.2.1 `typedef std::map< int, int > HepPID::EvtGenPDTMap`

Definition at line 33 of file translateEvtGen.cc.

7.3.2.2 `typedef std::map< int, int > HepPID::HerwigPDTMap`

Definition at line 33 of file translateHerwig.cc.

7.3.2.3 `typedef std::map< int, int > HepPID::InverseQQbarMap`

Definition at line 40 of file translateQQ.cc.

7.3.2.4 `typedef std::map< int, int > HepPID::IsajetPDTMap`

Definition at line 36 of file translateIsajet.cc.

7.3.2.5 `typedef std::map< int, std::string > HepPID::ParticleIdMap`

Definition at line 33 of file ParticleName.cc.

7.3.2.6 `typedef std::map< std::string, int > HepPID::ParticleLookupMap`

Definition at line 34 of file ParticleName.cc.

7.3.2.7 `typedef std::map< int, int > HepPID::PDGtoPDTMap`

Definition at line 33 of file translatePDG.cc.

7.3.2.8 `typedef std::map< int, int > HepPID::PDTEvtGenMap`

Definition at line 34 of file translateEvtGen.cc.

7.3.2.9 `typedef std::map< int, int > HepPID::PDTHerwigMap`

Definition at line 34 of file translateHerwig.cc.

7.3.2.10 `typedef std::map< int, int > HepPID::PDTIsajetMap`

Definition at line 37 of file translateIsajet.cc.

7.3.2.11 `typedef std::map< int, int > HepPID::PDTPythiaMap`

Definition at line 34 of file translatePythia.cc.

7.3.2.12 `typedef std::map< int, int > HepPID::PDTQQMap`

Definition at line 38 of file translateQQ.cc.

7.3.2.13 `typedef std::map< int, int > HepPID::PDTtoPDGMap`

Definition at line 34 of file translatePDG.cc.

7.3.2.14 `typedef std::map< int, int > HepPID::PythiaPDTMap`

Definition at line 33 of file translatePythia.cc.

7.3.2.15 `typedef std::map< int, int > HepPID::QQbarMap`

Definition at line 39 of file translateQQ.cc.

7.3.2.16 `typedef std::map< int, int > HepPID::QQPDTMap`

Definition at line 37 of file translateQQ.cc.

7.3.3 Enumeration Type Documentation

7.3.3.1 `enum HepPID::location`

PID digits (base 10) are: n nr nl nq1 nq2 nq3 nj The location enum provides a convenient index into the PID.

Enumerator:

nj

nq3

nq2
nq1
nl
nr
n
n8
n9
n10

Definition at line 24 of file ParticleIDMethods.hh.

7.3.4 Function Documentation

7.3.4.1 int HepPID::A (const int & pid)

if this is a nucleus (ion), get A Ion numbers are +/- 10LZZZAAAI.

Definition at line 59 of file ParticleIDMethods.cc.

References abspid(), and isNucleus().

Referenced by main().

7.3.4.2 int HepPID::abspid (const int & pid)

absolute value of particle ID

Definition at line 15 of file ParticleIDMethods.cc.

Referenced by A(), convIsajettoPDT(), convPDTtoIsajet(), digit(), extraBits(), fundamentalID(), isBaryon(), isDiQuark(), isMeson(), isNucleus(), jSpin(), lambda(), threeCharge(), and Z().

7.3.4.3 int HepPID::@63::convIsajettoPDT (const int id) [static]

Definition at line 689 of file translateIsajet.cc.

References abspid(), digit(), nj, nl, nq1, nq2, and nq3.

Referenced by translateIsajettoPDT().

7.3.4.4 int HepPID::@63::convPDTtoIsajet (const int id) [static]

Definition at line 790 of file translateIsajet.cc.

References abspid(), digit(), fundamentalID(), nj, nl, nq1, nq2, and nq3.

Referenced by translatePDTtoIsajet().

7.3.4.5 unsigned short HepPID::digit (location loc, const int & pid)

return the digit at a named location in the PID

Definition at line 27 of file ParticleIDMethods.cc.

References abspid().

Referenced by convIsajettoPDT(), convPDTtoIsajet(), fundamentalID(), hasBottom(), hasCharm(), hasDown(), hasStrange(), hasTop(), hasUp(), isBaryon(), isDiQuark(), isMeson(), isNucleus(), isPentaquark(), isRhadron(), isSUSY(), lambda(), lSpin(), main(), sSpin(), and threeCharge().

7.3.4.6 EvtGenPDTMap const& HepPID::@60::EvtGenPDTMapInit () [static]

Definition at line 41 of file translateEvtGen.cc.

Referenced by getEvtGenPDTMap().

7.3.4.7 int HepPID::extraBits (const int & pid)

returns everything beyond the 7th digit (e.g. outside the standard numbering scheme)

Definition at line 21 of file ParticleIDMethods.cc.

References abspid().

Referenced by fundamentalID(), hasBottom(), hasCharm(), hasDown(), hasStrange(), hasTop(), hasUp(), isBaryon(), isDiQuark(), isHadron(), isLepton(), isMeson(), isPentaquark(), isRhadron(), isSUSY(), is-Valid(), jSpin(), main(), and threeCharge().

7.3.4.8 int HepPID::fundamentalID (const int & pid)

extract fundamental ID (1-100) if this is a "fundamental" particle

Definition at line 37 of file ParticleIDMethods.cc.

References abspid(), digit(), extraBits(), nq1, and nq2.

Referenced by convPDTtoIsajet(), hasBottom(), hasCharm(), hasDown(), hasFundamentalAnti(), hasStrange(), hasTop(), hasUp(), isBaryon(), isDiQuark(), isLepton(), isMeson(), isSUSY(), isValid(), jSpin(), main(), and threeCharge().

7.3.4.9 EvtGenPDTMap const& HepPID::@60::getEvtGenPDTMap () [static]

Definition at line 608 of file translateEvtGen.cc.

References EvtGenPDTMapInit().

7.3.4.10 EvtGenPDTMap const& HepPID::@60::getEvtGenPDTMap () [static]

Referenced by PDTEvtGenMapInit(), translateEvtGentoPDT(), and writeEvtGenTranslationLine().

7.3.4.11 HerwigPDTMap const& HepPID::@62::getHerwigPDTMap () [static]

Definition at line 500 of file translateHerwig.cc.

References HerwigPDTMapInit().

7.3.4.12 HerwigPDTMap const& HepPID::@62::getHerwigPDTMap () [static]

Referenced by PDTHerwigMapInit(), translateHerwigtoPDT(), and writeHerwigTranslationLine().

7.3.4.13 InverseQQbarMap const& HepPID::@67::getInverseQQbarMap () [static]

Definition at line 560 of file translateQQ.cc.

References InverseQQbarMapInit().

7.3.4.14 InverseQQbarMap const& HepPID::@67::getInverseQQbarMap () [static]

Referenced by translateInverseQQbar().

7.3.4.15 IsajetPDTMap const& HepPID::@63::getIsajetPDTMap () [static]

Definition at line 891 of file translateIsajet.cc.

References IsajetPDTMapInit().

7.3.4.16 IsajetPDTMap const& HepPID::@63::getIsajetPDTMap () [static]

Referenced by PDTIsajetMapInit(), translateIsajettoPDT(), and writeIsajetTranslationLine().

7.3.4.17 ParticleNameMap const & HepPID::getParticleNameMap ()

access the **ParticleNameMap** (p. 102) for other purposes

Definition at line 1631 of file ParticleName.cc.

References ParticleNameInit().

Referenced by particleName(), and validParticleName().

7.3.4.18 PDGtoPDTMap const& HepPID::@64::getPDGtoPDTMap () [static]

Definition at line 369 of file translatePDG.cc.

References PDGtoPDTMapInit().

7.3.4.19 PDGtoPDTMap const& HepPID::@64::getPDGtoPDTMap () [static]

Referenced by PDTtoPDGMapInit(), translatePDGtabletoPDT(), and writePDGTranslationLine().

7.3.4.20 PDTEvtGenMap const& HepPID::@60::getPDTEvtGenMap () [static]

Definition at line 617 of file translateEvtGen.cc.

References PDTEvtGenMapInit().

7.3.4.21 PDTEvtGenMap const& HepPID::@60::getPDTEvtGenMap () [static]

Referenced by translatePDTtoEvtGen().

7.3.4.22 PDTHerwigMap const& HepPID::@62::getPDTHerwigMap () [static]

Definition at line 509 of file translateHerwig.cc.

References PDTHerwigMapInit().

7.3.4.23 PDTHerwigMap const& HepPID::@62::getPDTHerwigMap () [static]

Referenced by translatePDTtoHerwig().

7.3.4.24 PDTIsajetMap const& HepPID::@63::getPDTIsajetMap () [static]

Definition at line 900 of file translateIsajet.cc.

References PDTIsajetMapInit().

7.3.4.25 PDTIsajetMap const& HepPID::@63::getPDTIsajetMap () [static]

Referenced by translatePDTtoIsajet().

7.3.4.26 PDTPythiaMap const& HepPID::@66::getPDTPythiaMap () [static]

Definition at line 625 of file translatePythia.cc.

References PDTPythiaMapInit().

7.3.4.27 PDTPythiaMap const& HepPID::@66::getPDTPythiaMap () [static]

Referenced by translatePDTtoPythia().

7.3.4.28 PDTQQMap const& HepPID::@67::getPDTQQMap () [static]

Definition at line 543 of file translateQQ.cc.

References PDTQQMapInit().

7.3.4.29 PDTQQMap const& HepPID::@67::getPDTQQMap () [static]

Referenced by translatePDTtoQQ().

7.3.4.30 PDTtoPDGMap const& HepPID::@64::getPDTtoPDGMap () [static]

Definition at line 378 of file translatePDG.cc.

References PDTtoPDGMapInit().

7.3.4.31 PDTtoPDGMap const& HepPID::@64::getPDTtoPDGMap () [static]

Referenced by translatePDTtoPDGtable().

7.3.4.32 PythiaPDTMap const& HepPID::@66::getPythiaPDTMap () [static]

Definition at line 616 of file translatePythia.cc.

References PythiaPDTMapInit().

7.3.4.33 PythiaPDTMap const& HepPID::@66::getPythiaPDTMap () [static]

Referenced by PDTPythiaMapInit(), translatePythiatoPDT(), and writePythiaTranslationLine().

7.3.4.34 QQbarMap const& HepPID::@67::getQQbarMap () [static]

Definition at line 551 of file translateQQ.cc.

References QQbarMapInit().

7.3.4.35 QQbarMap const& HepPID::@67::getQQbarMap () [static]

Referenced by InverseQQbarMapInit(), and translateQQbar().

7.3.4.36 QQPDTMap const& HepPID::@67::getQPDTMap () [static]

Definition at line 534 of file translateQQ.cc.

References QQPDTMapInit().

7.3.4.37 QQPDTMap const& HepPID::@67::getQPDTMap () [static]

Referenced by PDTQQMapInit(), and translateQQtPDT().

7.3.4.38 bool HepPID::hasBottom (const int & pid)

does this particle contain a bottom quark?

Definition at line 297 of file ParticleIDMethods.cc.

References digit(), extraBits(), fundamentalID(), nq1, nq2, and nq3.

7.3.4.39 bool HepPID::hasCharm (const int & pid)

does this particle contain a charm quark?

Definition at line 289 of file ParticleIDMethods.cc.

References digit(), extraBits(), fundamentalID(), nq1, nq2, and nq3.

7.3.4.40 bool HepPID::hasDown (const int & pid)

does this particle contain a down quark?

Definition at line 273 of file ParticleIDMethods.cc.

References digit(), extraBits(), fundamentalID(), nq1, nq2, and nq3.

7.3.4.41 bool HepPID::hasFundamentalAnti (const int & pid)

if this is a fundamental particle, does it have a valid antiparticle?

Definition at line 112 of file ParticleIDMethods.cc.

References fundamentalID(), and validParticleName().

Referenced by isValid().

7.3.4.42 bool HepPID::hasStrange (const int & pid)

does this particle contain a strange quark?

Definition at line 281 of file ParticleIDMethods.cc.

References digit(), extraBits(), fundamentalID(), nq1, nq2, and nq3.

7.3.4.43 bool HepPID::hasTop (const int & pid)

does this particle contain a top quark?

Definition at line 305 of file ParticleIDMethods.cc.

References digit(), extraBits(), fundamentalID(), nq1, nq2, and nq3.

7.3.4.44 bool HepPID::hasUp (const int & pid)

does this particle contain an up quark?

Definition at line 265 of file ParticleIDMethods.cc.

References digit(), extraBits(), fundamentalID(), nq1, nq2, and nq3.

7.3.4.45 HerwigPDTMap const& HepPID::@62::HerwigPDTMapInit () [static]

Definition at line 41 of file translateHerwig.cc.

Referenced by getHerwigPDTMap().

7.3.4.46 InverseQQbarMap const& HepPID::@67::InverseQQbarMapInit () [static]

Definition at line 520 of file translateQQ.cc.

References getQQbarMap().

Referenced by getInverseQQbarMap().

7.3.4.47 IsajetPDTMap const& HepPID::@63::IsajetPDTMapInit () [static]

Definition at line 44 of file translateIsajet.cc.

Referenced by getIsajetPDTMap().

7.3.4.48 bool HepPID::isBaryon (const int & pid)

is this a valid baryon ID?

Definition at line 148 of file ParticleIDMethods.cc.

References abspid(), digit(), extraBits(), fundamentalID(), nj, nq1, nq2, and nq3.

Referenced by isHadron(), isValid(), main(), and threeCharge().

7.3.4.49 bool HepPID::isDiQuark (const int & pid)

is this a valid diquark ID?

Definition at line 160 of file ParticleIDMethods.cc.

References abspid(), digit(), extraBits(), fundamentalID(), nj, nq1, nq2, and nq3.

Referenced by isValid(), main(), and threeCharge().

7.3.4.50 bool HepPID::isHadron (const int & pid)

is this a valid hadron ID?

Definition at line 179 of file ParticleIDMethods.cc.

References extraBits(), isBaryon(), isMeson(), and isPentaquark().

Referenced by main().

7.3.4.51 bool HepPID::isLepton (const int & pid)

is this a valid lepton ID?

Definition at line 188 of file ParticleIDMethods.cc.

References extraBits(), and fundamentalID().

Referenced by main().

7.3.4.52 bool HepPID::isMeson (const int & pid)

is this a valid meson ID?

Definition at line 124 of file ParticleIDMethods.cc.

References abspid(), digit(), extraBits(), fundamentalID(), nj, nq1, nq2, and nq3.

Referenced by isHadron(), isValid(), lSpin(), main(), sSpin(), and threeCharge().

7.3.4.53 bool HepPID::isNucleus (const int & pid)

is this a valid ion ID?

Definition at line 202 of file ParticleIDMethods.cc.

References abspid(), digit(), n10, and n9.

Referenced by A(), isValid(), lambda(), main(), and Z().

7.3.4.54 bool HepPID::isPentaquark (const int & pid)

is this a valid pentaquark ID?

Definition at line 216 of file ParticleIDMethods.cc.

References digit(), extraBits(), n, nj, nl, nq1, nq2, nq3, and nr.

Referenced by isHadron(), and isValid().

7.3.4.55 bool HepPID::isRhadron (const int & pid)

is this a valid R-hadron ID?

Definition at line 248 of file ParticleIDMethods.cc.

References digit(), extraBits(), isSUSY(), n, nj, nq2, nq3, and nr.

Referenced by isValid().

7.3.4.56 bool HepPID::isSUSY (const int & pid)

is this a valid SUSY ID?

Definition at line 236 of file ParticleIDMethods.cc.

References digit(), extraBits(), fundamentalID(), n, and nr.

Referenced by isRhadron(), and isValid().

7.3.4.57 bool HepPID::isValid (const int & pid)

is this a valid ID?

Examples:

examListPythia.cc.

Definition at line 82 of file ParticleIDMethods.cc.

References extraBits(), fundamentalID(), hasFundamentalAnti(), isBaryon(), isDiQuark(), isMeson(), isNucleus(), isPentaquark(), isRhadron(), and isSUSY().

Referenced by main(), translateEvtGentoPDT(), translateHerwigtoPDT(), translatePDGtabletoPDT(), translatePDTtoEvtGen(), translatePDTtoHerwig(), translatePDTtoPDGtable(), translatePDTtoPythia(), and translatePythiatoPDT().

7.3.4.58 int HepPID::jSpin (const int & pid)

jSpin returns $2J+1$, where J is the total spin

Definition at line 316 of file ParticleIDMethods.cc.

References abspid(), extraBits(), and fundamentalID().

Referenced by main().

7.3.4.59 int HepPID::lambda (const int & pid)

if this is a nucleus (ion), get nLambda Ion numbers are +/- 10LZZZAAAI.

Definition at line 69 of file ParticleIDMethods.cc.

References abspid(), digit(), isNucleus(), and n8.

Referenced by main().

7.3.4.60 void HepPID::listParticleNames (std::ostream & os)

list all known names

Examples:

listParticleNames.cc.

Definition at line 1678 of file ParticleName.cc.

References n, writeParticleNameLine(), and writeVersion().

Referenced by main().

7.3.4.61 int HepPID::lSpin (const int & pid)

lSpin returns $2L+1$, where L is the orbital angular momentum

Definition at line 357 of file ParticleIDMethods.cc.

References digit(), isMeson(), n, nj, and nl.

Referenced by main().

7.3.4.62 int HepPID::particleName (const std::string &)

lookup a known ID

Definition at line 1666 of file ParticleName.cc.

References HepPID::ParticleNameMap::endLookupMap(), HepPID::ParticleNameMap::findString(), and getParticleNameMap().

7.3.4.63 std::string HepPID::particleName (const int &)

get a known **HepPID** (p. 37) Particle name

Examples:

examListHerwig.cc, **examListIsajet.cc**, and **examListPythia.cc**.

Definition at line 1656 of file ParticleName.cc.

References HepPID::ParticleNameMap::end(), HepPID::ParticleNameMap::find(), and getParticleNameMap().

Referenced by main(), HepPDT::detail::parseParticleLine(), HepPDT::ParticleID::PDTname(), writeEvtGenTranslationLine(), writeHerwigTranslationLine(), writeIsajetTranslationLine(), writeParticleNameLine(), writePDGTranslationLine(), writePythiaTranslationLine(), and writeQQTranslation().

7.3.4.64 ParticleNameMap const& HepPID::@59::ParticleNameInit () [static]

Definition at line 75 of file ParticleName.cc.

Referenced by getParticleNameMap().

7.3.4.65 PDGtoPDTMap const& HepPID::@64::PDGtoPDTMapInit () [static]

Definition at line 41 of file translatePDG.cc.

Referenced by getPDGtoPDTMap().

7.3.4.66 PDTEvtGenMap const& HepPID::@60::PDTEvtGenMapInit () [static]

Definition at line 594 of file translateEvtGen.cc.

References getEvtGenPDTMap().

Referenced by getPDTEvtGenMap().

7.3.4.67 PDTHerwigMap const& HepPID::@62::PDTHerwigMapInit () [static]

Definition at line 486 of file translateHerwig.cc.

References getHerwigPDTMap().

Referenced by getPDTHerwigMap().

7.3.4.68 PDTIsajetMap const& HepPID::@63::PDTIsajetMapInit () [static]

Definition at line 677 of file translateIsajet.cc.

References getIsajetPDTMap().

Referenced by getPDTIsajetMap().

7.3.4.69 PDTPythiaMap const& HepPID::@66::PDTPythiaMapInit () [static]

Definition at line 602 of file translatePythia.cc.

References getPythiaPDTMap().

Referenced by getPDTPythiaMap().

7.3.4.70 PDTQQMap const& HepPID::@67::PDTQQMapInit () [static]

Definition at line 509 of file translateQQ.cc.

References getQPDTMap().

Referenced by getPDTQQMap().

7.3.4.71 PDTtoPDGMap const& HepPID::@64::PDTtoPDGMapInit () [static]

Definition at line 355 of file translatePDG.cc.

References getPDGtoPDTMap().

Referenced by getPDTtoPDGMap().

7.3.4.72 PythiaPDTMap const& HepPID::@66::PythiaPDTMapInit () [static]

Definition at line 41 of file translatePythia.cc.

Referenced by getPythiaPDTMap().

7.3.4.73 QQbarMap const& HepPID::@67::QQbarMapInit () [static]

Definition at line 455 of file translateQQ.cc.

Referenced by getQQbarMap().

7.3.4.74 QQPDTMap const& HepPID::@67::QQPDTMapInit () [static]

Definition at line 49 of file translateQQ.cc.

Referenced by getQPDTMap().

7.3.4.75 int HepPID::sSpin (const int & pid)

sSpin returns $2S+1$, where S is the spin

Definition at line 332 of file ParticleIDMethods.cc.

References digit(), isMeson(), n, nj, and nl.

Referenced by main().

7.3.4.76 int HepPID::threeCharge (const int & pid)

return 3 times the charge (3 x quark charge is an int)

Definition at line 406 of file ParticleIDMethods.cc.

References abspid(), digit(), extraBits(), fundamentalID(), isBaryon(), isDiQuark(), isMeson(), nj, nq1, nq2, and nq3.

Referenced by main().

7.3.4.77 int HepPID::translateEvtGentoPDT (const int *evtGenID*)

translate EvtGen to PDG standard

Definition at line 625 of file translateEvtGen.cc.

References getEvtGenPDTMap(), and isValid().

Referenced by HepPDT::addEvtGenParticles().

7.3.4.78 int HepPID::translateGeanttoPDT (const int *geantID*)

translate Geant3 to PDG standard

Definition at line 20 of file translateGeanttoPDT.cc.

References IDMAX.

7.3.4.79 int HepPID::translateHerwigtoPDT (const int *herwigID*)

translate Herwig to PDG standard

Examples:

examListHerwig.cc.

Definition at line 517 of file translateHerwig.cc.

References getHerwigPDTMap(), and isValid().

Referenced by main().

7.3.4.80 int HepPID::translateInverseQQbar (const int *id*)

QQ helper function.

Definition at line 579 of file translateQQ.cc.

References getInverseQQbarMap().

Referenced by writeQQTranslation().

7.3.4.81 int HepPID::translateIsajettoPDT (const int *isajetID*)

translate Isajet to PDG standard

Examples:

examListIsajet.cc.

Definition at line 908 of file translateIsajet.cc.

References convIsajettoPDT(), and getIsajetPDTMap().

Referenced by HepPDT::addIsajetParticles(), and main().

7.3.4.82 int HepPID::translatePDGtabletoPDT (const int *pdgID*)

translate PDG table to PDG standard

Definition at line 386 of file translatePDG.cc.

References getPDGtoPDTMap(), and isValid().

7.3.4.83 int HepPID::translatePDTtoEvtGen (const int *pid*)

translate PDG standard to EvtGen

Definition at line 638 of file translateEvtGen.cc.

References getPDEvtGenMap(), and isValid().

Referenced by writeEvtGenTranslationLine().

7.3.4.84 int HepPID::translatePDTtoGeant (const int *pid*)

translate PDG standard to Geant3

Definition at line 22 of file translatePDTtoGeant.cc.

References IDMAX.

7.3.4.85 int HepPID::translatePDTtoHerwig (const int *pid*)

translate PDG standard to Herwig

Definition at line 530 of file translateHerwig.cc.

References getPDTHerwigMap(), and isValid().

Referenced by writeHerwigTranslationLine().

7.3.4.86 int HepPID::translatePDTtoIsajet (const int *pid*)

translate PDG standard to Isajet

Definition at line 919 of file translateIsajet.cc.

References convPDTtoIsajet(), and getPDTIsajetMap().

Referenced by writeIsajetTranslationLine().

7.3.4.87 int HepPID::translatePDTtoPDGtable (const int *pid*)

translate PDG standard to PDG table

Definition at line 399 of file translatePDG.cc.

References getPDTtoPDGMap(), and isValid().

Referenced by writePDGTranslationLine().

7.3.4.88 int HepPID::translatePDTtoPythia (const int *pid*)

translate PDG standard to Pythia

Definition at line 646 of file translatePythia.cc.

References getPDTPythiaMap(), and isValid().

Referenced by writePythiaTranslationLine().

7.3.4.89 int HepPID::translatePDTtoQQ (const int *pid*)

translate PDG standard to QQ

Definition at line 601 of file translateQQ.cc.

References getPDTQQMap().

Referenced by writeQQTranslation().

7.3.4.90 int HepPID::translatePythiaToPDT (const int *pythiaID*)

translate Pythia to PDG standard

Examples:

examListPythia.cc.

Definition at line 633 of file translatePythia.cc.

References getPythiaPDTMap(), and isValid().

Referenced by HepPDT::addPythiaParticles(), and main().

7.3.4.91 int HepPID::translateQQbar (const int *id*)

QQ helper function.

Definition at line 568 of file translateQQ.cc.

References getQQbarMap().

Referenced by HepPDT::addQQParticles(), and writeQQTranslation().

7.3.4.92 int HepPID::translateQQtoPDT (const int *qqID*)

translate QQ to PDG standard

Definition at line 590 of file translateQQ.cc.

References getQQPDTMap().

Referenced by HepPDT::addQQParticles(), and writeQQTranslation().

7.3.4.93 bool HepPID::validParticleName (const std::string &)

verify that this string has a valid id

Definition at line 1647 of file ParticleName.cc.

References HepPID::ParticleNameMap::endLookupMap(), HepPID::ParticleNameMap::findString(), and getParticleNameMap().

7.3.4.94 bool HepPID::validParticleName (const int &)

verify that this number has a valid name

Definition at line 1637 of file ParticleName.cc.

References HepPID::ParticleNameMap::end(), HepPID::ParticleNameMap::find(), and getParticleNameMap().

Referenced by hasFundamentalAnti(), and writeParticleNameLine().

7.3.4.95 void HepPID::version ()

print **HepPID** (p. 37) version

Definition at line 19 of file HepPID/Version.cc.

References versionName().

7.3.4.96 std::string HepPID::versionName ()

return **HepPID** (p. 37) version

Definition at line 14 of file HepPID/Version.cc.

Referenced by version(), and writeVersion().

7.3.4.97 void HepPID::writeEvtGenTranslation (std::ostream & os)

output the translation list

Examples:

listEvtGenTranslation.cc.

Definition at line 675 of file translateEvtGen.cc.

References writeEvtGenTranslationLine(), and writeVersion().

Referenced by main().

7.3.4.98 void HepPID::writeEvtGenTranslationLine (int i, std::ostream & os)

Definition at line 651 of file translateEvtGen.cc.

References getEvtGenPDTMap(), particleName(), and translatePDTtoEvtGen().

Referenced by writeEvtGenTranslation().

7.3.4.99 void HepPID::writeHerwigTranslation (std::ostream & os)

output the translation list

Examples:

listHerwigTranslation.cc.

Definition at line 567 of file translateHerwig.cc.

References n, writeHerwigTranslationLine(), and writeVersion().

Referenced by main().

7.3.4.100 void HepPID::writeHerwigTranslationLine (int i, std::ostream & os)

Definition at line 543 of file translateHerwig.cc.

References getHerwigPDTMap(), particleName(), and translatePDTtoHerwig().

Referenced by writeHerwigTranslation().

7.3.4.101 void HepPID::writeIsajetTranslation (std::ostream & os)

output the translation list

Examples:

listIsajetTranslation.cc.

Definition at line 954 of file translateIsajet.cc.

References writeIsajetTranslationLine(), and writeVersion().

Referenced by main().

7.3.4.102 void HepPID::writeIsajetTranslationLine (int i, std::ostream & os)

Definition at line 930 of file translateIsajet.cc.

References getIsajetPDTMap(), particleName(), and translatePDTtoIsajet().

Referenced by writeIsajetTranslation().

7.3.4.103 void HepPID::@59::writeParticleNameLine (int i, std::ostream & os) [static]

Definition at line 1614 of file ParticleName.cc.

References particleName(), and validParticleName().

Referenced by listParticleNames().

7.3.4.104 void HepPID::writePDGTranslation (std::ostream & os)

output the translation list

Examples:**listPDGTranslation.cc.**

Definition at line 436 of file translatePDG.cc.

References writePDGTranslationLine(), and writeVersion().

Referenced by main().

7.3.4.105 void HepPID::writePDGTranslationLine (int *i*, std::ostream & *os*)

Definition at line 412 of file translatePDG.cc.

References getPDGtoPDTMap(), particleName(), and translatePDTtoPDGtable().

Referenced by writePDGTranslation().

7.3.4.106 void HepPID::writePythiaTranslation (std::ostream & *os*)

output the translation list

Examples:**listPythiaTranslation.cc.**

Definition at line 683 of file translatePythia.cc.

References n, writePythiaTranslationLine(), and writeVersion().

Referenced by main().

7.3.4.107 void HepPID::writePythiaTranslationLine (int *i*, std::ostream & *os*)

Definition at line 659 of file translatePythia.cc.

References getPythiaPDTMap(), particleName(), and translatePDTtoPythia().

Referenced by writePythiaTranslation().

7.3.4.108 void HepPID::writeQQTranslation (std::ostream & *os*)

output the translation list

Examples:**listQQTranslation.cc.**

Definition at line 612 of file translateQQ.cc.

References particleName(), translateInverseQQbar(), translatePDTtoQQ(), translateQQbar(), translateQQtoPDT(), and writeVersion().

Referenced by main().

7.3.4.109 void HepPID::writeVersion (std::ostream & os)

write **HepPID** (p. 37) version to os

Examples:

examListHerwig.cc, **examListIsajet.cc**, and **examListPythia.cc**.

Definition at line 25 of file HepPID/Version.cc.

References `versionName()`.

Referenced by `listParticleNames()`, `main()`, `writeEvtGenTranslation()`, `writeHerwigTranslation()`, `writeIsajetTranslation()`, `writePDGTranslation()`, `writePythiaTranslation()`, and `writeQQTranslation()`.

7.3.4.110 int HepPID::Z (const int & pid)

if this is a nucleus (ion), get Z Ion numbers are +/- 10LZZZAAAI.

Definition at line 50 of file ParticleIDMethods.cc.

References `abspid()`, and `isNucleus()`.

Referenced by `main()`.

7.4 std Namespace Reference

Chapter 8

HepPDT Class Documentation

8.1 HepPDT::Constituent Class Reference

The **Constituent** (p. 63) class has information about constituent particles.

```
#include <Constituent.hh>
```

Public Member Functions

- **Constituent** (**ParticleID** p=**ParticleID**(0), int m=-1)
- **Constituent** (**Constituent** const &orig)
- **Constituent & operator=** (**Constituent** const &rhs)
- void **swap** (**Constituent** &other)
- int **multiplicity** () const
how many of this constituent are there?
- **ParticleID pid** () const
ParticleID (p. 93) *of this constituent.*
- bool **isUp** () const
is this an up quark?
- bool **isDown** () const
is this a down quark?
- bool **isStrange** () const
is this a strange quark?
- bool **isCharm** () const
is this a charm quark?
- bool **isBottom** () const
is this a bottom quark?
- bool **isTop** () const
is this a top quark?

8.1.1 Detailed Description

The **Constituent** (p. 63) class has information about constituent particles.

Author:

Lynn Garren

Holds a particle constituent (e.g. quark type and number of quarks of this type)

Definition at line 26 of file Constituent.hh.

8.1.2 Constructor & Destructor Documentation

8.1.2.1 HepPDT::Constituent::Constituent (ParticleID *p* = ParticleID(0), int *m* = -1) [inline]

Definition at line 31 of file Constituent.hh.

8.1.2.2 HepPDT::Constituent::Constituent (Constituent const & *orig*) [inline]

Definition at line 36 of file Constituent.hh.

8.1.3 Member Function Documentation

8.1.3.1 Constituent& HepPDT::Constituent::operator= (Constituent const & *rhs*) [inline]

Definition at line 38 of file Constituent.hh.

References swap().

8.1.3.2 void HepPDT::Constituent::swap (Constituent & *other*) [inline]

Definition at line 43 of file Constituent.hh.

References itsMultiplicity, itsPid, and HepPDT::swap().

Referenced by operator=(), and HepPDT::swap().

8.1.3.3 int HepPDT::Constituent::multiplicity () const [inline]

how many of this constituent are there?

Definition at line 50 of file Constituent.hh.

8.1.3.4 ParticleID HepPDT::Constituent::pid () const [inline]

ParticleID (p. 93) of this constituent.

Definition at line 52 of file Constituent.hh.

8.1.3.5 bool HepPDT::Constituent::isUp () const

is this an up quark?

Definition at line 12 of file Constituent.cc.

References HepPDT::ParticleID::fundamentalID().

8.1.3.6 bool HepPDT::Constituent::isDown () const

is this a down quark?

Definition at line 18 of file Constituent.cc.

References HepPDT::ParticleID::fundamentalID().

8.1.3.7 bool HepPDT::Constituent::isStrange () const

is this a strange quark?

Definition at line 24 of file Constituent.cc.

References HepPDT::ParticleID::fundamentalID().

8.1.3.8 bool HepPDT::Constituent::isCharm () const

is this a charm quark?

Definition at line 30 of file Constituent.cc.

References HepPDT::ParticleID::fundamentalID().

8.1.3.9 bool HepPDT::Constituent::isBottom () const

is this a bottom quark?

Definition at line 36 of file Constituent.cc.

References HepPDT::ParticleID::fundamentalID().

8.1.3.10 bool HepPDT::Constituent::isTop () const

is this a top quark?

Definition at line 42 of file Constituent.cc.

References HepPDT::ParticleID::fundamentalID().

The documentation for this class was generated from the following files:

- **Constituent.hh**
- **Constituent.cc**

8.2 HepPDT::DefTable Class Reference

The **DefTable** (p. 66) class holds EvtGen definitions.

```
#include <DefTable.hh>
```

Public Types

- `typedef TempDefMap::const_iterator const_iterator`
- `typedef TempDefMap::iterator iterator`

Public Member Functions

- **DefTable ()**
add a definition to the map
- **~DefTable ()**
- **void addDefinition (std::string const &def, double val)**
is this definition already defined?
- **bool hasDefinition (std::string const &def) const**
get the size of the definition map
- **int size () const**
return the definition of this parameter
- **double definition (std::string const &def) const**
use for diagnostics
- **iterator begin ()**
begin iterating over the definition map
- **const_iterator begin () const**
begin iterating over the definition map
- **iterator end ()**
end iterating over the definition map
- **const_iterator end () const**
end iterating over the definition map

8.2.1 Detailed Description

The **DefTable** (p. 66) class holds EvtGen definitions.

Author:

Lynn Garren

This is temporary information storage used when reading EvtGen input.

Definition at line 23 of file DefTable.hh.

8.2.2 Member Typedef Documentation

8.2.2.1 `typedef TempDefMap::const_iterator HepPDT::DefTable::const_iterator`

Definition at line 27 of file DefTable.hh.

8.2.2.2 `typedef TempDefMap::iterator HepPDT::DefTable::iterator`

Definition at line 28 of file DefTable.hh.

8.2.3 Constructor & Destructor Documentation

8.2.3.1 `HepPDT::DefTable::DefTable()`

Definition at line 15 of file DefTable.cc.

8.2.3.2 `HepPDT::DefTable::~DefTable() [inline]`

Definition at line 31 of file DefTable.hh.

8.2.4 Member Function Documentation

8.2.4.1 `void HepPDT::DefTable::addDefinition (std::string const & def, double val) [inline]`

add a definition to the map

Definition at line 36 of file DefTable.hh.

Referenced by HepPDT::addEvtGenParticles().

8.2.4.2 `bool HepPDT::DefTable::hasDefinition (std::string const & def) const`

is this definition already defined?

Definition at line 31 of file DefTable.cc.

Referenced by HepPDT::TableBuilder::hasDefinition().

8.2.4.3 `int HepPDT::DefTable::size () const [inline]`

get the size of the definition map

Definition at line 46 of file DefTable.hh.

8.2.4.4 double HepPDT::DefTable::definition (std::string const & def)

return the definition of this parameter

Definition at line 18 of file DefTable.cc.

Referenced by HepPDT::TableBuilder::definition().

8.2.4.5 void HepPDT::DefTable::writeDefinitions () const

use for diagnostics

Definition at line 41 of file DefTable.cc.

References begin(), and end().

8.2.4.6 iterator HepPDT::DefTable::begin () [inline]

begin iterating over the definition map

Definition at line 53 of file DefTable.hh.

Referenced by writeDefinitions().

8.2.4.7 const_iterator HepPDT::DefTable::begin () const [inline]

begin iterating over the definition map

Definition at line 55 of file DefTable.hh.

8.2.4.8 iterator HepPDT::DefTable::end () [inline]

end iterating over the definition map

Definition at line 58 of file DefTable.hh.

Referenced by writeDefinitions().

8.2.4.9 const_iterator HepPDT::DefTable::end () const [inline]

end iterating over the definition map

Definition at line 60 of file DefTable.hh.

The documentation for this class was generated from the following files:

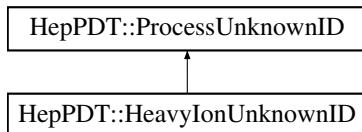
- **DefTable.hh**
- **DefTable.cc**

8.3 HepPDT::HeavyIonUnknownID Class Reference

The **HeavyIonUnknownID** (p. 69) class inherits from **ProcessUnknownID** (p. 104).

```
#include <HeavyIonUnknownID.hh>
```

Inheritance diagram for HepPDT::HeavyIonUnknownID:::



Public Member Functions

- **HeavyIonUnknownID ()**
- virtual **ParticleData * processUnknownID (ParticleID, const ParticleData *, const ParticleData *)**

8.3.1 Detailed Description

The **HeavyIonUnknownID** (p. 69) class inherits from **ProcessUnknownID** (p. 104).

Author:

Lynn Garren

Create and return a pointer to a **ParticleData** (p. 74) object if invoked by a valid Heavy Ion. HeavyIonUnknownID::processUnknownID(ParticleID) calculates both charge and an approximate mass for the particle.

Examples:

testHepPDT.cc.

Definition at line 25 of file HeavyIonUnknownID.hh.

8.3.2 Constructor & Destructor Documentation

8.3.2.1 HepPDT::HeavyIonUnknownID::HeavyIonUnknownID () [inline]

Definition at line 27 of file HeavyIonUnknownID.hh.

8.3.3 Member Function Documentation

8.3.3.1 ParticleData * HepPDT::HeavyIonUnknownID::processUnknownID (ParticleID, const ParticleData *, const ParticleData *) [virtual]

Implements **HepPDT::ProcessUnknownID** (p. 104).

Definition at line 12 of file HeavyIonUnknownID.cc.

References HepPDT::ParticleID::isNucleus(), and HepPDT::ParticleData::mass().

The documentation for this class was generated from the following files:

- **HeavyIonUnknownID.hh**
- **HeavyIonUnknownID.cc**

8.4 HepPDT::Measurement Class Reference

The **Measurement** (p. 71) class defines a value with its error.

```
#include <Measurement.hh>
```

Public Member Functions

- **Measurement ()**
- **Measurement (double value, double sigma)**
- **Measurement (const Measurement &m)**
- **void swap (Measurement &other)**
- **Measurement & operator= (Measurement const &rhs)**
- **bool operator< (Measurement const &other) const**
compare the value, ignore the error
- **bool operator== (Measurement const &other) const**
compare the value, ignore the error
- **double value () const**
- **double sigma () const**
- **operator double () const**

8.4.1 Detailed Description

The **Measurement** (p. 71) class defines a value with its error.

Author:

Mark Fischler

Contain a value with associated errors. Provide simple comparisons.

Examples:

examMyPDT.cc.

Definition at line 20 of file Measurement.hh.

8.4.2 Constructor & Destructor Documentation

8.4.2.1 HepPDT::Measurement::Measurement () [inline]

Definition at line 33 of file Measurement.icc.

8.4.2.2 HepPDT::Measurement::Measurement (double *value*, double *sigma*) [inline]

Definition at line 30 of file Measurement.icc.

8.4.2.3 HepPDT::Measurement::Measurement (const Measurement & *m*) [inline]

Definition at line 35 of file Measurement.icc.

8.4.3 Member Function Documentation**8.4.3.1 void HepPDT::Measurement::swap (Measurement & *other*) [inline]**

Definition at line 38 of file Measurement.icc.

References sig, HepPDT::swap(), and val.

Referenced by operator=(), HepPDT::ResonanceStructure::swap(), and HepPDT::swap().

8.4.3.2 Measurement & HepPDT::Measurement::operator= (Measurement const & *rhs*) [inline]

Definition at line 44 of file Measurement.icc.

References swap().

8.4.3.3 bool HepPDT::Measurement::operator< (Measurement const & *other*) const [inline]

compare the value, ignore the error

Definition at line 51 of file Measurement.icc.

References val.

8.4.3.4 bool HepPDT::Measurement::operator== (Measurement const & *other*) const [inline]

compare the value, ignore the error

Definition at line 56 of file Measurement.icc.

References val.

8.4.3.5 double HepPDT::Measurement::value () const [inline]

Definition at line 61 of file Measurement.icc.

Referenced by HepPDT::ResonanceStructure::lifetime(), operator double(), HepPDT::ResonanceStructure::setTotalWidthFromLifetime(), and HepPDT::ParticleData::write().

8.4.3.6 double HepPDT::Measurement::sigma () const [inline]

Definition at line 62 of file Measurement.icc.

Referenced by HepPDT::ResonanceStructure::lifetime(), HepPDT::ResonanceStructure::setTotalWidthFromLifetime(), and HepPDT::ParticleData::write().

8.4.3.7 HepPDT::Measurement::operator double () const [inline]

Definition at line 64 of file Measurement.icc.

References value().

The documentation for this class was generated from the following files:

- **Measurement.hh**
- **Measurement.icc**

8.5 HepPDT::ParticleData Class Reference

The **ParticleData** (p. 74) class holds basic particle data.

```
#include <ParticleData.hh>
```

Public Member Functions

- **ParticleData** (const **TempParticleData** &tpd)
create ParticleData (p. 74) from the temporary information
- **~ParticleData** ()
- void **swap** (**ParticleData** &rhs)
- **ParticleData** (const **ParticleData** &orig)
- **ParticleData** & **operator=** (const **ParticleData** &rhs)
- const std::string & **name** () const
return particle name as defined by user input
- const std::string **PDTname** () const
return PDG particle name
- const std::string & **source** () const
the name of the input source
- **ParticleID** **ID** () const
return the ParticleID (p. 93)
- int **pid** () const
return the integer ID
- int **originalID** () const
return untranslated integer ID
- double **charge** () const
return charge
- double **color** () const
color information
- **SpinState** **spin** () const
spin information
- **Measurement** **mass** () const
mass
- **Measurement** **totalWidth** () const
return the total width
- double **lowerCutoff** () const
lower cutoff of allowed width values

- double **upperCutoff** () const
upper cutoff of allowed width values
- **Measurement lifetime** () const
calculate the lifetime
- int **numConstituents** () const
number of constituent particles (e.g., quarks)
- **Constituent constituent** (unsigned int i) const
constituent information
- **ParticleID constituentParticle** (unsigned int i) const
ParticleID (p. 93) *for a constituent particle.*
- **ResonanceStructure** const **resonance** () const
resonance (width) information
- void **write** (std::ostream &os) const
output information about this particle
- void **writeParticleInfo** (std::ostream &os) const
- void **writeParticleTranslation** (std::ostream &os) const
output the translation information for this particle
- bool **isMeson** () const
is this a valid meson?
- bool **isBaryon** () const
is this a valid baryon?
- bool **isDiQuark** () const
is this a valid diquark?
- bool **isHadron** () const
is this a valid hadron?
- bool **isLepton** () const
is this a valid lepton?
- bool **isNucleus** () const
is this a valid ion?
- bool **hasUp** () const
does this particle contain an up quark?
- bool **hasDown** () const
does this particle contain a down quark?

- **bool hasStrange () const**
does this particle contain a strange quark?
- **bool hasCharm () const**
does this particle contain a charm quark?
- **bool hasBottom () const**
does this particle contain a bottom quark?
- **bool hasTop () const**
does this particle contain a top quark?
- **bool operator< (const ParticleData &other) const**
compare masses
- **bool operator== (const ParticleData &other) const**
use PID and ignore everything else
- **void setCharge (double chg)**
change the charge
- **void setColor (double col)**
change color information
- **void setSpin (const SpinState &spin)**
change spin information
- **void addConstituent (Constituent c)**
add a constituent particle
- **void setMass (Measurement const &mass)**
change the mass
- **void setTotalWidth (Measurement const &width)**
change the total width
- **void setTotalWidthFromLifetime (Measurement const <)**
change the total width using a lifetime
- **void setLowerCutoff (double cut)**
change the total width lower cutoff
- **void setUpperCutoff (double cut)**
change the total width upper cutoff

8.5.1 Detailed Description

The **ParticleData** (p. 74) class holds basic particle data.

Author:

Lynn Garren

This class holds the basic particle data - name, ID, spin, etc. Methods are provided to get various information about the particle.

Examples:

examMyPDT.cc, and **testHepPDT.cc**.

Definition at line 31 of file ParticleData.hh.

8.5.2 Constructor & Destructor Documentation

8.5.2.1 HepPDT::ParticleData::ParticleData (const TempParticleData & *tpd*) [inline]

create **ParticleData** (p. 74) from the temporary information

Definition at line 12 of file ParticleData.icc.

8.5.2.2 HepPDT::ParticleData::~ParticleData () [inline]

Definition at line 26 of file ParticleData.icc.

8.5.2.3 HepPDT::ParticleData::ParticleData (const ParticleData & *orig*) [inline]

Definition at line 42 of file ParticleData.icc.

8.5.3 Member Function Documentation

8.5.3.1 void HepPDT::ParticleData::swap (ParticleData & *rhs*) [inline]

Definition at line 29 of file ParticleData.icc.

References itsCharge, itsColorCharge, itsID, itsOriginalID, itsParticleName, itsQuarks, itsResonance, itsSource, itsSpin, HepPDT::SpinState::swap(), HepPDT::ParticleID::swap(), and HepPDT::swap().

Referenced by operator=(), and HepPDT::swap().

8.5.3.2 ParticleData & HepPDT::ParticleData::operator= (const ParticleData & *rhs*) [inline]

Definition at line 54 of file ParticleData.icc.

References swap().

8.5.3.3 const std::string& HepPDT::ParticleData::name () const [inline]

return particle name as defined by user input

Examples:

examMyPDT.cc.

Definition at line 50 of file ParticleData.hh.

Referenced by main().

8.5.3.4 const std::string HepPDT::ParticleData::PDTname () const [inline]

return PDG particle name

Definition at line 52 of file ParticleData.hh.

References HepPDT::ParticleID::PDTname().

Referenced by writeParticleTranslation().

8.5.3.5 const std::string& HepPDT::ParticleData::source () const [inline]

the name of the input source

Definition at line 54 of file ParticleData.hh.

8.5.3.6 ParticleID HepPDT::ParticleData::ID () const [inline]

return the **ParticleID** (p. 93)

Definition at line 56 of file ParticleData.hh.

8.5.3.7 int HepPDT::ParticleData::pid () const [inline]

return the integer ID

Definition at line 58 of file ParticleData.hh.

References HepPDT::ParticleID::pid().

Referenced by writeParticleTranslation().

8.5.3.8 int HepPDT::ParticleData::originalID () const [inline]

return untranslated integer ID

Definition at line 60 of file ParticleData.hh.

8.5.3.9 double HepPDT::ParticleData::charge () const [inline]

return charge

Definition at line 62 of file ParticleData.hh.

8.5.3.10 double HepPDT::ParticleData::color () const [inline]

color information

Definition at line 64 of file ParticleData.hh.

8.5.3.11 SpinState HepPDT::ParticleData::spin () const [inline]

spin information

Definition at line 66 of file ParticleData.hh.

Referenced by setSpin().

8.5.3.12 Measurement HepPDT::ParticleData::mass () const [inline]

mass

Definition at line 68 of file ParticleData.hh.

References HepPDT::ResonanceStructure::mass().

Referenced by operator<(), HepPDT::HeavyIonUnknownID::processUnknownID(), setMass(), and write().

8.5.3.13 Measurement HepPDT::ParticleData::totalWidth () const [inline]

return the total width

Examples:

testHepPDT.cc.

Definition at line 70 of file ParticleData.hh.

References HepPDT::ResonanceStructure::totalWidth().

Referenced by main().

8.5.3.14 double HepPDT::ParticleData::lowerCutoff () const [inline]

lower cutoff of allowed width values

Examples:

testHepPDT.cc.

Definition at line 72 of file ParticleData.hh.

References HepPDT::ResonanceStructure::lowerCutoff().

Referenced by main(), and write().

8.5.3.15 double HepPDT::ParticleData::upperCutoff () const [inline]

upper cutoff of allowed width values

Examples:**testHepPDT.cc.**

Definition at line 74 of file ParticleData.hh.

References HepPDT::ResonanceStructure::upperCutoff().

Referenced by main(), and write().

8.5.3.16 Measurement HepPDT::ParticleData::lifetime () const [inline]

calculate the lifetime

Definition at line 76 of file ParticleData.hh.

References HepPDT::ResonanceStructure::lifetime().

Referenced by write().

8.5.3.17 int HepPDT::ParticleData::numConstituents () const [inline]

number of constituent particles (e.g., quarks)

Definition at line 78 of file ParticleData.hh.

8.5.3.18 Constituent HepPDT::ParticleData::constituent (unsigned int *i*) const [inline]

constituent information

return this constituent if it exists

Definition at line 91 of file ParticleData.icc.

8.5.3.19 ParticleID HepPDT::ParticleData::constituentParticle (unsigned int *i*) const [inline]

ParticleID (p. 93) for a constituent particle.

Definition at line 102 of file ParticleData.icc.

8.5.3.20 ResonanceStructure const HepPDT::ParticleData::resonance () const [inline]

resonance (width) information

Definition at line 84 of file ParticleData.hh.

8.5.3.21 void HepPDT::ParticleData::write (std::ostream & *os*) const

output information about this particle

Examples:**testHepPDT.cc.**

Definition at line 18 of file write.cc.

References HepPDT::ParticleID::isValid(), lifetime(), lowerCutoff(), HepPDT::ResonanceStructure::mass(), mass(), HepPDT::SpinState::orbAngMom(), HepPDT::ParticleID::pid(), HepPDT::Measurement::sigma(), HepPDT::SpinState::spin(), HepPDT::SpinState::totalSpin(), HepPDT::ResonanceStructure::totalWidth(), upperCutoff(), and HepPDT::Measurement::value().

Referenced by main(), and writeParticleInfo().

8.5.3.22 void HepPDT::ParticleData::writeParticleInfo (std::ostream & os) const

output all information about a particle EXCEPT its decays This method is redundant with **write()** (p. 80), but retained for backwards compatibility.

Definition at line 68 of file write.cc.

References write().

8.5.3.23 void HepPDT::ParticleData::writeParticleTranslation (std::ostream & os) const

output the translation information for this particle

Definition at line 73 of file write.cc.

References PDTname(), and pid().

8.5.3.24 bool HepPDT::ParticleData::isMeson () const [inline]

is this a valid meson?

Definition at line 97 of file ParticleData.hh.

References HepPDT::ParticleID::isMeson().

8.5.3.25 bool HepPDT::ParticleData::isBaryon () const [inline]

is this a valid baryon?

Definition at line 99 of file ParticleData.hh.

References HepPDT::ParticleID::isBaryon().

8.5.3.26 bool HepPDT::ParticleData::isDiQuark () const [inline]

is this a valid diquark?

Definition at line 101 of file ParticleData.hh.

References HepPDT::ParticleID::isDiQuark().

8.5.3.27 bool HepPDT::ParticleData::isHadron () const [inline]

is this a valid hadron?

Definition at line 103 of file ParticleData.hh.

References HepPDT::ParticleID::isHadron().

8.5.3.28 bool HepPDT::ParticleData::isLepton () const [inline]

is this a valid lepton?

Definition at line 105 of file ParticleData.hh.

References HepPDT::ParticleID::isLepton().

8.5.3.29 bool HepPDT::ParticleData::isNucleus () const [inline]

is this a valid ion?

Definition at line 107 of file ParticleData.hh.

References HepPDT::ParticleID::isNucleus().

8.5.3.30 bool HepPDT::ParticleData::hasUp () const

does this particle contain an up quark?

Definition at line 16 of file hasMethods.cc.

8.5.3.31 bool HepPDT::ParticleData::hasDown () const

does this particle contain a down quark?

Definition at line 26 of file hasMethods.cc.

8.5.3.32 bool HepPDT::ParticleData::hasStrange () const

does this particle contain a strange quark?

Definition at line 36 of file hasMethods.cc.

8.5.3.33 bool HepPDT::ParticleData::hasCharm () const

does this particle contain a charm quark?

Definition at line 46 of file hasMethods.cc.

8.5.3.34 bool HepPDT::ParticleData::hasBottom () const

does this particle contain a bottom quark?

Definition at line 56 of file hasMethods.cc.

8.5.3.35 bool HepPDT::ParticleData::hasTop () const

does this particle contain a top quark?

Definition at line 66 of file hasMethods.cc.

8.5.3.36 bool HepPDT::ParticleData::operator< (const ParticleData & *other*) const [inline]

compare masses

Definition at line 61 of file ParticleData.icc.

References mass().

8.5.3.37 bool HepPDT::ParticleData::operator== (const ParticleData & *other*) const [inline]

use PID and ignore everything else

Definition at line 66 of file ParticleData.icc.

References itsID.

8.5.3.38 void HepPDT::ParticleData::setCharge (double *chg*) [inline]

change the charge

Definition at line 129 of file ParticleData.hh.

8.5.3.39 void HepPDT::ParticleData::setColor (double *col*) [inline]

change color information

Definition at line 131 of file ParticleData.hh.

8.5.3.40 void HepPDT::ParticleData::setSpin (const SpinState & *spin*) [inline]

change spin information

Definition at line 133 of file ParticleData.hh.

References spin().

8.5.3.41 void HepPDT::ParticleData::addConstituent (Constituent *c*) [inline]

add a constituent particle

Definition at line 135 of file ParticleData.hh.

8.5.3.42 void HepPDT::ParticleData::setMass (Measurement const & *mass*) [inline]

change the mass

Definition at line 137 of file ParticleData.hh.

References mass(), and HepPDT::ResonanceStructure::setMass().

8.5.3.43 void HepPDT::ParticleData::setTotalWidth (Measurement const & *width*) [inline]

change the total width

Definition at line 139 of file ParticleData.hh.

References HepPDT::ResonanceStructure::setTotalWidth().

8.5.3.44 void HepPDT::ParticleData::setTotalWidthFromLifetime (Measurement const & *lt*) [inline]

change the total width using a lifetime

Definition at line 141 of file ParticleData.hh.

References HepPDT::ResonanceStructure::setTotalWidthFromLifetime().

8.5.3.45 void HepPDT::ParticleData::setLowerCutoff (double *cut*) [inline]

change the total width lower cutoff

Definition at line 143 of file ParticleData.hh.

References HepPDT::ResonanceStructure::setLowerCutoff().

8.5.3.46 void HepPDT::ParticleData::setUpperCutoff (double *cut*) [inline]

change the total width upper cutoff

Definition at line 145 of file ParticleData.hh.

References HepPDT::ResonanceStructure::setUpperCutoff().

The documentation for this class was generated from the following files:

- **ParticleData.hh**
- **ParticleData.icc**
- **hasMethods.cc**
- **write.cc**

8.6 HepPDT::ParticleDataTable Class Reference

The **ParticleDataTable** (p. 85) class is the core of **HepPDT** (p. 23).

```
#include <ParticleDataTable.hh>
```

Public Types

- `typedef ParticleData CPD`
- `typedef std::list< ParticleData > CPDlist`
- `typedef CPDlist::const_iterator CPDID`
- `typedef std::map< ParticleID, TempParticleData > TempMap`
- `typedef std::map< ParticleID, ParticleData, ParticleDataTableComparison > PDTMap`
- `typedef std::map< std::string, ParticleID > PDTNameMap`
- `typedef PDTMap::const_iterator const_iterator`
- `typedef PDTNameMap::const_iterator const_iteratorByName`

Public Member Functions

- **ParticleDataTable** (std::string name= " ", **ProcessUnknownID** *=`new SimpleProcessUnknownID`)
- **~ParticleDataTable** ()
- int **size** () const
size of the particle data table
- **const_iterator begin** () const
begin iterating over the particle data table
- **const_iterator end** () const
end iterating over the particle data table
- int **sizeNameMap** () const
size of the map of particle names
- **const_iteratorByName beginNameMap** () const
begin iterating over the map of particle names
- **const_iteratorByName endNameMap** () const
end iterating over the map of particle names
- std::string **tableName** () const
return the name of this particle data table
- **ParticleData const * particle (ParticleID)** const
access particle information via ParticleID (p. 93)
- **ParticleData * particle (ParticleID)**
access particle information via ParticleID (p. 93)
- **ParticleData const * particle (std::string)** const

access particle information via a particle name

- **ParticleData * particle (std::string)**

access particle information via a particle name

- **ParticleData * operator[] (ParticleID)**

access particle information via ParticleID (p. 93)

- **ParticleData const * operator[] (ParticleID) const**

access particle information via ParticleID (p. 93)

- **ParticleData * operator[] (std::string)**

access particle information via a particle name

- **ParticleData const * operator[] (std::string) const**

access particle information via a particle name

- **void writeParticleData (std::ostream &outstr)**

output all information in the PDT

- **void writeParticleInfo (std::ostream &outstr)**

output all information about a particle EXCEPT its decays

- **void writeParticleTranslation (std::ostream &outstr)**

output a list of original IDs and their translations

- **void convertTemporaryMap (TempMap &tempPDT, std::ostream &err)**

used by the TableBuilder (p. 116) destructor to fill the PDT

8.6.1 Detailed Description

The **ParticleDataTable** (p. 85) class is the core of **HepPDT** (p. 23).

Author:

Lynn Garren, Walter Brown

This is the table of particle data information. This table is created once at the beginning of a job and referenced as needed. We expect that the table will be saved as part of the standard job output. Methods exist to fill the table from a variety of input formats. The user may fill the table from as many different input streams as desired. See the examples.

Examples:

examMyPDT.cc, **listEvtGenNames.cc.in**, **listPDGNames.cc.in**, **listPythiaNames.cc.in**, **testHepPDT.cc**, **testReadEvtGen.cc.in**, **testReadIsajet.cc.in**, and **testReadQQ.cc.in**.

Definition at line 45 of file ParticleDataTable.hh.

8.6.2 Member Typedef Documentation

8.6.2.1 `typedef ParticleData HepPDT::ParticleDataTable::CPD`

Definition at line 48 of file ParticleDataTable.hh.

8.6.2.2 `typedef std::list<ParticleData> HepPDT::ParticleDataTable::CPDlist`

Definition at line 50 of file ParticleDataTable.hh.

8.6.2.3 `typedef CPDlist::const_iterator HepPDT::ParticleDataTable::CPDID`

Definition at line 52 of file ParticleDataTable.hh.

8.6.2.4 `typedef std::map<ParticleID,TempParticleData> HepPDT::ParticleDataTable::TempMap`

Definition at line 54 of file ParticleDataTable.hh.

8.6.2.5 `typedef std::map<ParticleID,ParticleData,ParticleDataTableComparison> HepPDT::ParticleDataTable::PDTMap`

Definition at line 55 of file ParticleDataTable.hh.

8.6.2.6 `typedef std::map<std::string,ParticleID> HepPDT::ParticleDataTable::PDTNameMap`

Definition at line 56 of file ParticleDataTable.hh.

8.6.2.7 `typedef PDTMap::const_iterator HepPDT::ParticleDataTable::const_iterator`

Definition at line 58 of file ParticleDataTable.hh.

8.6.2.8 `typedef PDTNameMap::const_iterator HepPDT::ParticleDataTable::const_iteratorByName`

Definition at line 59 of file ParticleDataTable.hh.

8.6.3 Constructor & Destructor Documentation

8.6.3.1 `HepPDT::ParticleDataTable::ParticleDataTable (std::string name = " ", ProcessUnknownID * = new SimpleProcessUnknownID)`

Construct a particle data table with an identifying name. Require a method to deal with unknown PID's.

Definition at line 19 of file ParticleDataTable.cc.

References HepPDT::version().

8.6.3.2 HepPDT::ParticleDataTable::~ParticleDataTable () [inline]

Definition at line 10 of file ParticleDataTable.icc.

8.6.4 Member Function Documentation**8.6.4.1 int HepPDT::ParticleDataTable::size () const [inline]**

size of the particle data table

Definition at line 71 of file ParticleDataTable.hh.

Referenced by writeParticleData(), and writeParticleInfo().

8.6.4.2 const_iterator HepPDT::ParticleDataTable::begin () const [inline]

begin iterating over the particle data table

Definition at line 73 of file ParticleDataTable.hh.

Referenced by writeParticleData(), writeParticleInfo(), and writeParticleTranslation().

8.6.4.3 const_iterator HepPDT::ParticleDataTable::end () const [inline]

end iterating over the particle data table

Definition at line 75 of file ParticleDataTable.hh.

Referenced by writeParticleData(), writeParticleInfo(), and writeParticleTranslation().

8.6.4.4 int HepPDT::ParticleDataTable::sizeNameMap () const [inline]

size of the map of particle names

Definition at line 77 of file ParticleDataTable.hh.

8.6.4.5 const_iteratorByName HepPDT::ParticleDataTable::beginNameMap () const [inline]

begin iterating over the map of particle names

Definition at line 79 of file ParticleDataTable.hh.

8.6.4.6 const_iteratorByName HepPDT::ParticleDataTable::endNameMap () const [inline]

end iterating over the map of particle names

Definition at line 81 of file ParticleDataTable.hh.

8.6.4.7 std::string HepPDT::ParticleDataTable::tableName () const [inline]

return the name of this particle data table

Definition at line 83 of file ParticleDataTable.hh.

Referenced by writeParticleData(), and writeParticleInfo().

8.6.4.8 ParticleData const * HepPDT::ParticleDataTable::particle (ParticleID) const

access particle information via **ParticleID** (p. 93)

Examples:

examMyPDT.cc, and **testHepPDT.cc**.

Definition at line 101 of file ParticleDataTable.cc.

Referenced by main(), operator[](), and particle().

8.6.4.9 ParticleData * HepPDT::ParticleDataTable::particle (ParticleID)

access particle information via **ParticleID** (p. 93)

Definition at line 89 of file ParticleDataTable.cc.

8.6.4.10 ParticleData const * HepPDT::ParticleDataTable::particle (std::string) const

access particle information via a particle name

Definition at line 150 of file ParticleDataTable.cc.

References particle().

8.6.4.11 ParticleData * HepPDT::ParticleDataTable::particle (std::string)

access particle information via a particle name

Definition at line 139 of file ParticleDataTable.cc.

References particle().

8.6.4.12 ParticleData * HepPDT::ParticleDataTable::operator[] (ParticleID) [inline]

access particle information via **ParticleID** (p. 93)

Definition at line 13 of file ParticleDataTable.icc.

References particle().

8.6.4.13 ParticleData const * HepPDT::ParticleDataTable::operator[] (ParticleID) const [inline]

access particle information via **ParticleID** (p. 93)

Definition at line 18 of file ParticleDataTable.icc.

References particle().

8.6.4.14 ParticleData * HepPDT::ParticleDataTable::operator[] (std::string) [inline]

access particle information via a particle name

Definition at line 23 of file ParticleDataTable.icc.

References particle().

8.6.4.15 ParticleData const * HepPDT::ParticleDataTable::operator[] (std::string) const [inline]

access particle information via a particle name

Definition at line 28 of file ParticleDataTable.icc.

References particle().

8.6.4.16 void HepPDT::ParticleDataTable::writeParticleData (std::ostream & outstr)

output all information in the PDT

Examples:

examMyPDT.cc, testHepPDT.cc, testReadEvtGen.cc.in, testReadIsajet.cc.in, and testReadQQ.cc.in.

Definition at line 26 of file ParticleDataTable.cc.

References begin(), end(), size(), tableName(), and HepPDT::writeVersion().

Referenced by main().

8.6.4.17 void HepPDT::ParticleDataTable::writeParticleInfo (std::ostream & outstr)

output all information about a particle EXCEPT its decays

Examples:

testHepPDT.cc, and testReadIsajet.cc.in.

Definition at line 52 of file ParticleDataTable.cc.

References begin(), end(), size(), tableName(), and HepPDT::writeVersion().

Referenced by main().

8.6.4.18 void HepPDT::ParticleDataTable::writeParticleTranslation (std::ostream & outstr)

output a list of original IDs and their translations

Examples:

listEvtGenNames.cc.in, listPDGNames.cc.in, listPythiaNames.cc.in, and testReadQQ.cc.in.

Definition at line 78 of file ParticleDataTable.cc.

References begin(), and end().

Referenced by main().

8.6.4.19 void HepPDT::ParticleDataTable::convertTemporaryMap (TempMap & *tempPDT*, std::ostream & *err*)

used by the **TableBuilder** (p. 116) destructor to fill the PDT

Definition at line 17 of file convertTemporaryMap.cc.

Referenced by HepPDT::TableBuilder::~TableBuilder().

The documentation for this class was generated from the following files:

- **ParticleDataTable.hh**
- **ParticleDataTable.icc**
- **convertTemporaryMap.cc**
- **ParticleDataTable.cc**

8.7 HepPDT::ParticleDataTableComparison Class Reference

The **ParticleDataTableComparison** (p. 92) class provides a utility for sorting the PDT.

```
#include <ParticleDataTableComparison.hh>
```

Public Member Functions

- **ParticleDataTableComparison ()**
stateless class
- **bool operator() (const ParticleID &, const ParticleID &) const**

8.7.1 Detailed Description

The **ParticleDataTableComparison** (p. 92) class provides a utility for sorting the PDT.

Author:

Lynn Garren

Sort the PDT by absolute value of the particle ID. If the absolute values are equal, the positive number comes first.

Definition at line 23 of file ParticleDataTableComparison.hh.

8.7.2 Constructor & Destructor Documentation

8.7.2.1 HepPDT::ParticleDataTableComparison::ParticleDataTableComparison () [inline]

stateless class

Definition at line 27 of file ParticleDataTableComparison.hh.

8.7.3 Member Function Documentation

8.7.3.1 bool HepPDT::ParticleDataTableComparison::operator() (const ParticleID &, const ParticleID &) const [inline]

Comparison method sorts by absolute value. If the absolute values are the same, the positive entry comes first.

Definition at line 36 of file ParticleDataTableComparison.hh.

References HepPDT::ParticleID::abspid(), and HepPDT::ParticleID::pid().

The documentation for this class was generated from the following file:

- **ParticleDataTableComparison.hh**

8.8 HepPDT::ParticleID Class Reference

The **ParticleID** (p. 93) has various utilities to extract information from the particle ID.

```
#include <ParticleID.hh>
```

Public Member Functions

- **ParticleID** (int pid=0)
create from an integer ID
- **ParticleID** (const **ParticleID** &orig)
- **ParticleID** & **operator=** (const **ParticleID** &)
- void **swap** (**ParticleID** &other)
- bool **operator<** (**ParticleID** const &other) const
- bool **operator==** (**ParticleID** const &other) const
- int **pid** () const
get the integer ID
- int **abspid** () const
get the absolute value
- bool **isValid** () const
is this a valid ID?
- bool **isMeson** () const
is this a valid meson ID?
- bool **isBaryon** () const
is this a valid baryon ID?
- bool **isDiQuark** () const
is this a valid diquark ID?
- bool **isHadron** () const
is this a valid hadron ID?
- bool **isLepton** () const
is this a valid lepton ID?
- bool **isNucleus** () const
is this a valid ion ID?
- bool **isPentaquark** () const
is this a valid pentaquark ID?
- bool **isSUSY** () const
is this a valid SUSY ID?
- bool **isRhadron** () const

is this a valid R-hadron ID?

- **bool hasUp () const**
does this particle contain an up quark?
- **bool hasDown () const**
does this particle contain a down quark?
- **bool hasStrange () const**
does this particle contain a strange quark?
- **bool hasCharm () const**
does this particle contain a charm quark?
- **bool hasBottom () const**
does this particle contain a bottom quark?
- **bool hasTop () const**
does this particle contain a top quark?
- **int jSpin () const**
jSpin returns $2J+1$, where J is the total spin
- **int sSpin () const**
sSpin returns $2S+1$, where S is the spin
- **int lSpin () const**
lSpin returns $2L+1$, where L is the orbital angular momentum
- **int fundamentalID () const**
return the first two digits if this is a "fundamental" particle
- **int extraBits () const**
- **Quarks quarks () const**
returns a list of 3 constituent quarks
- **int threeCharge () const**
- **int A () const**
if this is a nucleus (ion), get A
- **int Z () const**
if this is a nucleus (ion), get Z
- **int lambda () const**
if this is a nucleus (ion), get nLambda
- **unsigned short digit (location) const**
return the digit at a named location in the PID
- **const std::string PDTname () const**
standard particle name

8.8.1 Detailed Description

The **ParticleID** (p. 93) has various utilities to extract information from the particle ID.

Author:

Lynn Garren

In the standard numbering scheme, the PID digits (base 10) are: +/- n nr nl nq1 nq2 nq3 nj It is expected that any 7 digit number used as a PID will adhere to the Monte Carlo numbering scheme documented by the PDG. Note that particles not already explicitly defined can be expressed within this numbering scheme.

Examples:

examMyPDT.cc, **testHepPDT.cc**, and **testPID.cc**.

Definition at line 64 of file ParticleID.hh.

8.8.2 Constructor & Destructor Documentation

8.8.2.1 HepPDT::ParticleID::ParticleID (int *pid* = 0) [inline]

create from an integer ID

Definition at line 10 of file ParticleID.icc.

8.8.2.2 HepPDT::ParticleID::ParticleID (const ParticleID & *orig*) [inline]

Definition at line 14 of file ParticleID.icc.

8.8.3 Member Function Documentation

8.8.3.1 ParticleID & HepPDT::ParticleID::operator= (const ParticleID &) [inline]

Definition at line 18 of file ParticleID.icc.

References swap().

8.8.3.2 void HepPDT::ParticleID::swap (ParticleID & *other*) [inline]

Definition at line 25 of file ParticleID.icc.

References itsPID, and HepPDT::swap().

Referenced by operator=(), HepPDT::swap(), and HepPDT::ParticleData::swap().

8.8.3.3 bool HepPDT::ParticleID::operator< (ParticleID const & *other*) const [inline]

Definition at line 30 of file ParticleID.icc.

References itsPID.

8.8.3.4 bool HepPDT::ParticleID::operator== (ParticleID const & *other*) const [inline]

Definition at line 35 of file ParticleID.icc.

References itsPID.

8.8.3.5 int HepPDT::ParticleID::pid () const [inline]

get the integer ID

Definition at line 85 of file ParticleID.hh.

Referenced by HepPDT::TempParticleData::antiparticle(), HepPDT::TableBuilder::getAntiParticle(), isMeson(), HepPDT::ParticleDataTableComparison::operator()(), HepPDT::parseEvtGenDecayLine(), HepPDT::parseEvtGenLine(), HepPDT::detail::parseIsajetLine(), HepPDT::detail::parseParticleLine(), HepPDT::detail::parsePythiaDecayLine(), HepPDT::detail::parsePythiaLine(), HepPDT::parseQQParticle(), HepPDT::ParticleData::pid(), HepPDT::TempParticleData::processPID(), threeCharge(), and HepPDT::ParticleData::write().

8.8.3.6 int HepPDT::ParticleID::abspid () const [inline]

get the absolute value

return a value greater than or equal to zero

Definition at line 40 of file ParticleID.icc.

Referenced by A(), digit(), extraBits(), fundamentalID(), hasBottom(), hasCharm(), hasDown(), hasStrange(), hasTop(), hasUp(), isBaryon(), isDiQuark(), isMeson(), isNucleus(), jSpin(), lambda(), lSpin(), HepPDT::ParticleDataTableComparison::operator()(), quarks(), sSpin(), threeCharge(), and Z().

8.8.3.7 bool HepPDT::ParticleID::isValid () const

is this a valid ID?

Definition at line 100 of file ParticleID.cc.

References extraBits(), fundamentalID(), isBaryon(), isDiQuark(), isMeson(), isNucleus(), isPentaquark(), isRhadron(), and isSUSY().

Referenced by HepPDT::addParticleTable(), and HepPDT::ParticleData::write().

8.8.3.8 bool HepPDT::ParticleID::isMeson () const

is this a valid meson ID?

Definition at line 219 of file ParticleID.cc.

References abspid(), digit(), extraBits(), fundamentalID(), HepPDT::nj, HepPDT::nq1, HepPDT::nq2, HepPDT::nq3, and pid().

Referenced by isHadron(), HepPDT::ParticleData::isMeson(), isValid(), lSpin(), and sSpin().

8.8.3.9 bool HepPDT::ParticleID::isBaryon () const

is this a valid baryon ID?

Definition at line 279 of file ParticleID.cc.

References abspid(), digit(), extraBits(), fundamentalID(), HepPDT::nj, HepPDT::nq1, HepPDT::nq2, and HepPDT::nq3.

Referenced by HepPDT::ParticleData::isBaryon(), isHadron(), and isValid().

8.8.3.10 bool HepPDT::ParticleID::isDiQuark () const

is this a valid diquark ID?

Definition at line 261 of file ParticleID.cc.

References abspid(), digit(), extraBits(), fundamentalID(), HepPDT::nj, HepPDT::nq1, HepPDT::nq2, and HepPDT::nq3.

Referenced by HepPDT::ParticleData::isDiQuark(), and isValid().

8.8.3.11 bool HepPDT::ParticleID::isHadron () const

is this a valid hadron ID?

Definition at line 251 of file ParticleID.cc.

References extraBits(), isBaryon(), isMeson(), and isPentaquark().

Referenced by HepPDT::ParticleData::isHadron().

8.8.3.12 bool HepPDT::ParticleID::isLepton () const

is this a valid lepton ID?

Definition at line 243 of file ParticleID.cc.

References extraBits(), and fundamentalID().

Referenced by HepPDT::ParticleData::isLepton().

8.8.3.13 bool HepPDT::ParticleID::isNucleus () const

is this a valid ion ID?

Definition at line 179 of file ParticleID.cc.

References A(), abspid(), digit(), HepPDT::n10, HepPDT::n9, and Z().

Referenced by HepPDT::ParticleData::isNucleus(), isValid(), lambda(), HepPDT::HeavyIonUnknown-ID::processUnknownID(), and threeCharge().

8.8.3.14 bool HepPDT::ParticleID::isPentaquark () const

is this a valid pentaquark ID?

Definition at line 125 of file ParticleID.cc.

References digit(), extraBits(), HepPDT::n, HepPDT::nj, HepPDT::nl, HepPDT::nq1, HepPDT::nq2, HepPDT::nq3, and HepPDT::nr.

Referenced by isHadron(), and isValid().

8.8.3.15 bool HepPDT::ParticleID::isSUSY () const

is this a valid SUSY ID?

Definition at line 162 of file ParticleID.cc.

References digit(), extraBits(), fundamentalID(), HepPDT::n, and HepPDT::nr.

Referenced by isRhadron(), and isValid().

8.8.3.16 bool HepPDT::ParticleID::isRhadron () const

is this a valid R-hadron ID?

Definition at line 145 of file ParticleID.cc.

References digit(), extraBits(), isSUSY(), HepPDT::n, HepPDT::nj, HepPDT::nq2, HepPDT::nq3, and HepPDT::nr.

Referenced by isValid().

8.8.3.17 bool HepPDT::ParticleID::hasUp () const

does this particle contain an up quark?

Definition at line 45 of file ParticleID.cc.

References abspid(), digit(), extraBits(), fundamentalID(), HepPDT::nq1, HepPDT::nq2, and HepPDT::nq3.

8.8.3.18 bool HepPDT::ParticleID::hasDown () const

does this particle contain a down quark?

Definition at line 54 of file ParticleID.cc.

References abspid(), digit(), extraBits(), fundamentalID(), HepPDT::nq1, HepPDT::nq2, and HepPDT::nq3.

8.8.3.19 bool HepPDT::ParticleID::hasStrange () const

does this particle contain a strange quark?

Definition at line 63 of file ParticleID.cc.

References abspid(), digit(), extraBits(), fundamentalID(), HepPDT::nq1, HepPDT::nq2, and HepPDT::nq3.

8.8.3.20 bool HepPDT::ParticleID::hasCharm () const

does this particle contain a charm quark?

Definition at line 72 of file ParticleID.cc.

References abspid(), digit(), extraBits(), fundamentalID(), HepPDT::nq1, HepPDT::nq2, and HepPDT::nq3.

8.8.3.21 bool HepPDT::ParticleID::hasBottom () const

does this particle contain a bottom quark?

Definition at line 81 of file ParticleID.cc.

References abspid(), digit(), extraBits(), fundamentalID(), HepPDT::nq1, HepPDT::nq2, and HepPDT::nq3.

8.8.3.22 bool HepPDT::ParticleID::hasTop () const

does this particle contain a top quark?

Definition at line 90 of file ParticleID.cc.

References abspid(), digit(), extraBits(), fundamentalID(), HepPDT::nq1, HepPDT::nq2, and HepPDT::nq3.

8.8.3.23 int HepPDT::ParticleID::jSpin () const

jSpin returns $2J+1$, where J is the total spin

Definition at line 290 of file ParticleID.cc.

References abspid(), extraBits(), and fundamentalID().

Referenced by HepPDT::TempParticleData::processPID().

8.8.3.24 int HepPDT::ParticleID::sSpin () const

sSpin returns $2S+1$, where S is the spin

Definition at line 357 of file ParticleID.cc.

References abspid(), isMeson(), and HepPDT::nl.

Referenced by HepPDT::TempParticleData::processPID().

8.8.3.25 int HepPDT::ParticleID::lSpin () const

lSpin returns $2L+1$, where L is the orbital angular momentum

Definition at line 307 of file ParticleID.cc.

References abspid(), isMeson(), and HepPDT::nl.

Referenced by HepPDT::TempParticleData::processPID().

8.8.3.26 int HepPDT::ParticleID::fundamentalID () const

return the first two digits if this is a "fundamental" particle

Definition at line 33 of file ParticleID.cc.

References abspid(), digit(), HepPDT::n10, HepPDT::n9, HepPDT::nq1, and HepPDT::nq2.

Referenced by hasBottom(), hasCharm(), hasDown(), hasStrange(), hasTop(), hasUp(), isBaryon(), HepPDT::Constituent::isBottom(), HepPDT::Constituent::isCharm(), isDiQuark(), HepPDT::Constituent::is-

Down(), isLepton(), isMeson(), HepPDT::Constituent::isStrange(), isSUSY(), HepPDT::Constituent::isTop(), HepPDT::Constituent::isUp(), isValid(), jSpin(), quarks(), and threeCharge().

8.8.3.27 int HepPDT::ParticleID::extraBits () const

returns everything beyond the 7th digit (e.g. outside the standard numbering scheme)

Definition at line 25 of file ParticleID.cc.

References abspid().

Referenced by hasBottom(), hasCharm(), hasDown(), hasStrange(), hasTop(), hasUp(), isBaryon(), isDiQuark(), isHadron(), isLepton(), isMeson(), isPentaquark(), isRhadron(), isSUSY(), isValid(), jSpin(), quarks(), and threeCharge().

8.8.3.28 Quarks HepPDT::ParticleID::quarks () const

returns a list of 3 constituent quarks

Definition at line 14 of file quarks.cc.

References abspid(), digit(), extraBits(), fundamentalID(), HepPDT::nq1, HepPDT::Quarks::nq1, HepPDT::nq2, HepPDT::Quarks::nq2, HepPDT::nq3, and HepPDT::Quarks::nq3.

Referenced by HepPDT::TempParticleData::processPID().

8.8.3.29 int HepPDT::ParticleID::threeCharge () const

this is mostly for use by functions like addPDGParticles that have to figure out the charge from the PID

Definition at line 382 of file ParticleID.cc.

References abspid(), digit(), extraBits(), fundamentalID(), isNucleus(), HepPDT::nj, HepPDT::nq1, HepPDT::nq2, HepPDT::nq3, pid(), and Z().

Referenced by HepPDT::TempParticleData::processPID().

8.8.3.30 int HepPDT::ParticleID::A () const

if this is a nucleus (ion), get A

Definition at line 192 of file ParticleID.cc.

References abspid(), digit(), HepPDT::n10, and HepPDT::n9.

Referenced by isNucleus().

8.8.3.31 int HepPDT::ParticleID::Z () const

if this is a nucleus (ion), get Z

Definition at line 201 of file ParticleID.cc.

References abspid(), digit(), HepPDT::n10, and HepPDT::n9.

Referenced by isNucleus(), and threeCharge().

8.8.3.32 int HepPDT::ParticleID::lambda () const

if this is a nucleus (ion), get nLambda

Definition at line 210 of file ParticleID.cc.

References abspid(), digit(), isNucleus(), and HepPDT::n8.

8.8.3.33 unsigned short HepPDT::ParticleID::digit (location) const

return the digit at a named location in the PID

Definition at line 16 of file ParticleID.cc.

References abspid().

Referenced by A(), fundamentalID(), hasBottom(), hasCharm(), hasDown(), hasStrange(), hasTop(), hasUp(), isBaryon(), isDiQuark(), isMeson(), isNucleus(), isPentaquark(), isRhadron(), isSUSY(), lambda(), quarks(), threeCharge(), and Z().

8.8.3.34 const std::string HepPDT::ParticleID::PDTname () const [inline]

standard particle name

Definition at line 150 of file ParticleID.hh.

References HepPID::particleName().

Referenced by HepPDT::ParticleData::PDTname().

The documentation for this class was generated from the following files:

- **ParticleID.hh**
- **ParticleID.icc**
- **ParticleID.cc**
- **quarks.cc**

8.9 HepPID::ParticleNameMap Class Reference

Public Types

- `typedef ParticleIdMap::const_iterator idIterator`
- `typedef ParticleLookupMap::const_iterator nameIterator`

Public Member Functions

- `ParticleNameMap (ParticleIdMap m1, ParticleLookupMap m2)`
- `~ParticleNameMap ()`
- `ParticleIdMap nameMap () const`
- `ParticleLookupMap lookupMap () const`
- `idIterator begin () const`
- `idIterator end () const`
- `idIterator find (const int &id) const`
- `nameIterator beginLookupMap () const`
- `nameIterator endLookupMap () const`
- `nameIterator findString (const std::string &s) const`

8.9.1 Detailed Description

Author:

Lynn Garren

Used internally to store the static maps

Definition at line 42 of file ParticleName.cc.

8.9.2 Member Typedef Documentation

8.9.2.1 `typedef ParticleIdMap::const_iterator HepPID::ParticleNameMap::idIterator`

Definition at line 46 of file ParticleName.cc.

8.9.2.2 `typedef ParticleLookupMap::const_iterator HepPID::ParticleNameMap::nameIterator`

Definition at line 47 of file ParticleName.cc.

8.9.3 Constructor & Destructor Documentation

8.9.3.1 `HepPID::ParticleNameMap::ParticleNameMap (ParticleIdMap m1, ParticleLookupMap m2) [inline]`

Definition at line 49 of file ParticleName.cc.

8.9.3.2 `HepPID::ParticleNameMap::~ParticleNameMap () [inline]`

Definition at line 51 of file ParticleName.cc.

8.9.4 Member Function Documentation

8.9.4.1 ParticleIdMap HepPID::ParticleNameMap::nameMap () const [inline]

Definition at line 53 of file ParticleName.cc.

8.9.4.2 ParticleLookupMap HepPID::ParticleNameMap::lookupMap () const [inline]

Definition at line 54 of file ParticleName.cc.

8.9.4.3 idIterator HepPID::ParticleNameMap::begin () const [inline]

Definition at line 55 of file ParticleName.cc.

8.9.4.4 idIterator HepPID::ParticleNameMap::end () const [inline]

Definition at line 56 of file ParticleName.cc.

Referenced by HepPID::particleName(), and HepPID::validParticleName().

8.9.4.5 idIterator HepPID::ParticleNameMap::find (const int & id) const [inline]

Definition at line 57 of file ParticleName.cc.

Referenced by HepPID::particleName(), and HepPID::validParticleName().

8.9.4.6 nameIterator HepPID::ParticleNameMap::beginLookupMap () const [inline]

Definition at line 58 of file ParticleName.cc.

8.9.4.7 nameIterator HepPID::ParticleNameMap::endLookupMap () const [inline]

Definition at line 59 of file ParticleName.cc.

Referenced by HepPID::particleName(), and HepPID::validParticleName().

8.9.4.8 nameIterator HepPID::ParticleNameMap::findString (const std::string & s) const [inline]

Definition at line 60 of file ParticleName.cc.

Referenced by HepPID::particleName(), and HepPID::validParticleName().

The documentation for this class was generated from the following file:

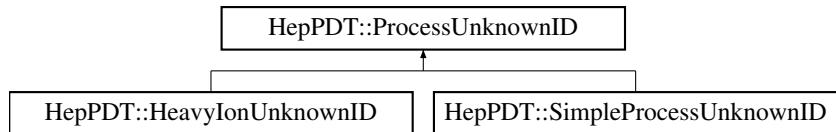
- **ParticleName.cc**

8.10 HepPDT::ProcessUnknownID Class Reference

The **ProcessUnknownID** (p. 104) class is abstract.

```
#include <ProcessUnknownID.hh>
```

Inheritance diagram for HepPDT::ProcessUnknownID::



Public Member Functions

- virtual **ParticleData * processUnknownID (ParticleID, const ParticleData *, const ParticleData *)=0**

Protected Member Functions

- **ProcessUnknownID ()**

8.10.1 Detailed Description

The **ProcessUnknownID** (p. 104) class is abstract.

Author:

Lynn Garren

This is an abstract class which allows you to define your own methods for handling undefined particle lookups.

Definition at line 25 of file ProcessUnknownID.hh.

8.10.2 Constructor & Destructor Documentation

8.10.2.1 HepPDT::ProcessUnknownID::ProcessUnknownID () [inline, protected]

Definition at line 34 of file ProcessUnknownID.hh.

8.10.3 Member Function Documentation

8.10.3.1 virtual ParticleData* HepPDT::ProcessUnknownID::processUnknownID (ParticleID, const ParticleData *, const ParticleData *) [pure virtual]

Implemented in **HepPDT::HeavyIonUnknownID** (p. 69), and **HepPDT::SimpleProcessUnknownID** (p. 112).

The documentation for this class was generated from the following file:

- [ProcessUnknownID.hh](#)

8.11 HepPDT::Quarks Struct Reference

constituent quarks

```
#include <ParticleID.hh>
```

Public Member Functions

- **Quarks ()**
- **Quarks (short q1, short q2, short q3)**

Public Attributes

- short **nq1**
- short **nq2**
- short **nq3**

8.11.1 Detailed Description

constituent quarks

Examples:

testPID.cc.

Definition at line 39 of file ParticleID.hh.

8.11.2 Constructor & Destructor Documentation

8.11.2.1 HepPDT::Quarks::Quarks () [inline]

Definition at line 42 of file ParticleID.hh.

8.11.2.2 HepPDT::Quarks::Quarks (short *q1*, short *q2*, short *q3*) [inline]

Definition at line 43 of file ParticleID.hh.

8.11.3 Member Data Documentation

8.11.3.1 short HepPDT::Quarks::nq1

Examples:

testPID.cc.

Definition at line 46 of file ParticleID.hh.

Referenced by main(), HepPDT::TempParticleData::processPID(), and HepPDT::ParticleID::quarks().

8.11.3.2 short HepPDT::Quarks::nq2**Examples:****testPID.cc.**

Definition at line 47 of file ParticleID.hh.

Referenced by main(), HepPDT::TempParticleData::processPID(), and HepPDT::ParticleID::quarks().

8.11.3.3 short HepPDT::Quarks::nq3**Examples:****testPID.cc.**

Definition at line 48 of file ParticleID.hh.

Referenced by main(), HepPDT::TempParticleData::processPID(), and HepPDT::ParticleID::quarks().

The documentation for this struct was generated from the following file:

- **ParticleID.hh**

8.12 HepPDT::ResonanceStructure Class Reference

The **ResonanceStructure** (p. 108) class holds mass and width information.

```
#include <ResonanceStructure.hh>
```

Public Member Functions

- **ResonanceStructure** (**Measurement** mass=**Measurement**()
width=**Measurement**(), double min=0., double max=0.) **Measurement**
construct from mass and total width
- virtual ~**ResonanceStructure** ()
- **ResonanceStructure** (const **ResonanceStructure** &orig)
- **ResonanceStructure** & operator= (const **ResonanceStructure** &rhs)
- void swap (**ResonanceStructure** &other)
- **Measurement** const & **mass** () const
get the mass
- **Measurement** const & **totalWidth** () const
get the total width
- **Measurement** lifetime () const
calculate lifetime from total width
- double **lowerCutoff** () const
lower cutoff of allowed width values
- double **upperCutoff** () const
upper cutoff of allowed width values
- void **setMass** (**Measurement** const &mass)
change the mass
- void **setTotalWidth** (**Measurement** const &width)
change the total width
- void **setTotalWidthFromLifetime** (**Measurement** const <)
change the total width using a lifetime
- void **setLowerCutoff** (double cut)
change the lower cutoff of allowed width values
- void **setUpUpperCutoff** (double cut)
change the upper cutoff of allowed width values

8.12.1 Detailed Description

The **ResonanceStructure** (p. 108) class is holds mass and width information.

Author:

Lynn Garren

ResonanceStructure (p. 108) contains the minimum information for a Breit-Wigner distribution about a given mass.

Definition at line 27 of file ResonanceStructure.hh.

8.12.2 Constructor & Destructor Documentation

8.12.2.1 HepPDT::ResonanceStructure::ResonanceStructure (**Measurement mass = Measurement()**, **Measurement width = Measurement()**, **double min = 0 .**, **double max = 0 .**)

construct from mass and total width

Definition at line 13 of file ResonanceStructure.cc.

8.12.2.2 HepPDT::ResonanceStructure::~ResonanceStructure () [virtual]

Definition at line 44 of file ResonanceStructure.cc.

8.12.2.3 HepPDT::ResonanceStructure::ResonanceStructure (**const ResonanceStructure & orig**)

Definition at line 21 of file ResonanceStructure.cc.

8.12.3 Member Function Documentation

8.12.3.1 ResonanceStructure & HepPDT::ResonanceStructure::operator= (**const ResonanceStructure & rhs**)

Definition at line 28 of file ResonanceStructure.cc.

References swap().

8.12.3.2 void HepPDT::ResonanceStructure::swap (**ResonanceStructure & other**)

Definition at line 35 of file ResonanceStructure.cc.

References itsLowerCutoff, itsMass, itsTotalWidth, itsUpperCutoff, HepPDT::swap(), and HepPDT::Measurement::swap().

Referenced by operator=(), and HepPDT::swap().

8.12.3.3 Measurement const& HepPDT::ResonanceStructure::mass () const [inline]

get the mass

Definition at line 48 of file ResonanceStructure.hh.

Referenced by HepPDT::ParticleData::mass(), setMass(), and HepPDT::ParticleData::write().

8.12.3.4 Measurement const& HepPDT::ResonanceStructure::totalWidth () const [inline]

get the total width

Definition at line 50 of file ResonanceStructure.hh.

Referenced by HepPDT::ParticleData::totalWidth(), and HepPDT::ParticleData::write().

8.12.3.5 Measurement HepPDT::ResonanceStructure::lifetime () const

calculate lifetime from total width

Definition at line 13 of file lifetime.cc.

References HepPDT::Measurement::sigma(), and HepPDT::Measurement::value().

Referenced by HepPDT::ParticleData::lifetime().

8.12.3.6 double HepPDT::ResonanceStructure::lowerCutoff () const [inline]

lower cutoff of allowed width values

Definition at line 54 of file ResonanceStructure.hh.

Referenced by HepPDT::ParticleData::lowerCutoff().

8.12.3.7 double HepPDT::ResonanceStructure::upperCutoff () const [inline]

upper cutoff of allowed width values

Definition at line 56 of file ResonanceStructure.hh.

Referenced by HepPDT::ParticleData::upperCutoff().

8.12.3.8 void HepPDT::ResonanceStructure::setMass (Measurement const & mass) [inline]

change the mass

Definition at line 61 of file ResonanceStructure.hh.

References mass().

Referenced by HepPDT::ParticleData::setMass().

8.12.3.9 void HepPDT::ResonanceStructure::setTotalWidth (Measurement const & width) [inline]

change the total width

Definition at line 63 of file ResonanceStructure.hh.

Referenced by HepPDT::ParticleData::setTotalWidth().

8.12.3.10 void HepPDT::ResonanceStructure::setTotalWidthFromLifetime (Measurement const & *lt*)

change the total width using a lifetime

Definition at line 47 of file ResonanceStructure.cc.

References HepPDT::Measurement::sigma(), and HepPDT::Measurement::value().

Referenced by HepPDT::ParticleData::setTotalWidthFromLifetime().

8.12.3.11 void HepPDT::ResonanceStructure::setLowerCutoff (double *cut*) [inline]

change the lower cutoff of allowed width values

Definition at line 67 of file ResonanceStructure.hh.

Referenced by HepPDT::ParticleData::setLowerCutoff().

8.12.3.12 void HepPDT::ResonanceStructure::setUpperCutoff (double *cut*) [inline]

change the upper cutoff of allowed width values

Definition at line 69 of file ResonanceStructure.hh.

Referenced by HepPDT::ParticleData::setUpperCutoff().

The documentation for this class was generated from the following files:

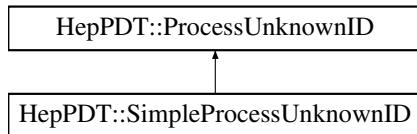
- **ResonanceStructure.hh**
- **lifetime.cc**
- **ResonanceStructure.cc**

8.13 HepPDT::SimpleProcessUnknownID Class Reference

The **SimpleProcessUnknownID** (p. 112) class inherits from **ProcessUnknownID** (p. 104).

```
#include <SimpleProcessUnknownID.hh>
```

Inheritance diagram for HepPDT::SimpleProcessUnknownID:::



Public Member Functions

- **SimpleProcessUnknownID ()**
- virtual **ParticleData * processUnknownID (ParticleID, const ParticleData *, const ParticleData *)**

8.13.1 Detailed Description

The **SimpleProcessUnknownID** (p. 112) class inherits from **ProcessUnknownID** (p. 104).

Author:

Lynn Garren

This provides the default **HepPDT** (p. 23) behaviour and will be invoked by default unless you specify a different behaviour when creating your **ParticleDataTable** (p. 85) object.

Definition at line 24 of file SimpleProcessUnknownID.hh.

8.13.2 Constructor & Destructor Documentation

8.13.2.1 HepPDT::SimpleProcessUnknownID::SimpleProcessUnknownID () [inline]

Definition at line 26 of file SimpleProcessUnknownID.hh.

8.13.3 Member Function Documentation

8.13.3.1 virtual ParticleData* HepPDT::SimpleProcessUnknownID::processUnknownID (ParticleID, const ParticleData *, const ParticleData *) [inline, virtual]

Implements **HepPDT::ProcessUnknownID** (p. 104).

Definition at line 28 of file SimpleProcessUnknownID.hh.

The documentation for this class was generated from the following file:

- **SimpleProcessUnknownID.hh**

8.14 HepPDT::SpinState Class Reference

The **SpinState** (p. 113) class holds spin information.

```
#include <SpinState.hh>
```

Public Member Functions

- **SpinState** (double ts=0., double spin=0., double oam=0.)
(default) constructor
- **SpinState** (const **SpinState** &orig)
- **SpinState** & **operator=** (const **SpinState** &rhs)
- void **swap** (**SpinState** &other)
- bool **operator==** (const **SpinState** &rhs) const
all three spins must match
- double **totalSpin** () const
return the total spin
- double **spin** () const
return the spin
- double **orbAngMom** () const
return the orbital angular momentum
- void **setTotalSpin** (double spin)
change the total spin
- void **setSpin** (double spin)
change the spin
- void **setOrbAngMom** (double ang)
change the orbital angular momentum

8.14.1 Detailed Description

The **SpinState** (p. 113) class holds spin information.

Author:

Lynn Garren

SpinState (p. 113) contains total spin, spin, and orbital angular momentum.

Examples:

examMyPDT.cc.

Definition at line 20 of file SpinState.hh.

8.14.2 Constructor & Destructor Documentation

8.14.2.1 HepPDT::SpinState::SpinState (double *ts* = 0., double *spin* = 0., double *oam* = 0.) [inline]

(default) constructor

Definition at line 11 of file SpinState.icc.

8.14.2.2 HepPDT::SpinState::SpinState (const SpinState & *orig*) [inline]

Definition at line 17 of file SpinState.icc.

8.14.3 Member Function Documentation

8.14.3.1 SpinState & HepPDT::SpinState::operator= (const SpinState & *rhs*) [inline]

Definition at line 23 of file SpinState.icc.

References swap().

8.14.3.2 void HepPDT::SpinState::swap (SpinState & *other*) [inline]

Definition at line 30 of file SpinState.icc.

References itsOrbAngMom, itsSpin, itsTotalSpin, and HepPDT::swap().

Referenced by operator=(), HepPDT::swap(), and HepPDT::ParticleData::swap().

8.14.3.3 bool HepPDT::SpinState::operator== (const SpinState & *rhs*) const [inline]

all three spins must match

Definition at line 36 of file SpinState.icc.

References itsOrbAngMom, itsSpin, and itsTotalSpin.

8.14.3.4 double HepPDT::SpinState::totalSpin () const [inline]

return the total spin

Definition at line 43 of file SpinState.hh.

Referenced by HepPDT::parseEvtGenLine(), HepPDT::parseQQParticle(), and HepPDT::ParticleData::write().

8.14.3.5 double HepPDT::SpinState::spin () const [inline]

return the spin

Definition at line 45 of file SpinState.hh.

Referenced by HepPDT::ParticleData::write().

8.14.3.6 double HepPDT::SpinState::orbAngMom () const [inline]

return the orbital angular momentum

Definition at line 47 of file SpinState.hh.

Referenced by HepPDT::ParticleData::write().

8.14.3.7 void HepPDT::SpinState::setTotalSpin (double *spin*) [inline]

change the total spin

Definition at line 52 of file SpinState.hh.

Referenced by HepPDT::parseEvtGenLine(), HepPDT::parseQQParticle(), and HepPDT::TempParticleData::processPID().

8.14.3.8 void HepPDT::SpinState::setSpin (double *spin*) [inline]

change the spin

Definition at line 54 of file SpinState.hh.

Referenced by HepPDT::TempParticleData::processPID().

8.14.3.9 void HepPDT::SpinState::setOrbAngMom (double *ang*) [inline]

change the orbital angular momentum

Definition at line 56 of file SpinState.hh.

Referenced by HepPDT::TempParticleData::processPID().

The documentation for this class was generated from the following files:

- **SpinState.hh**
- **SpinState.icc**

8.15 HepPDT::TableBuilder Class Reference

The **TableBuilder** (p. 116) class is used to construct a **ParticleDataTable** (p. 85).

```
#include <TableBuilder.hh>
```

Public Member Functions

- **TableBuilder (ParticleDataTable &table, std::ostream &str=std::cerr)**
create TableBuilder (p. 116) from a ParticleDataTable (p. 85)
- **~TableBuilder ()**
call the ParticleDataTable (p. 85) conversion method upon destruction
- **TempParticleData & getParticleData (ParticleID pid)**
create a TempParticleData (p. 125) from a ParticleID (p. 93)
- **TempParticleData & getParticleData (std::string const &name)**
create a TempParticleData (p. 125) from a particle name
- **TempParticleData & getAntiParticle (ParticleID pid, const std::string &aname)**
create an antiparticle TempParticleData (p. 125) from a ParticleID (p. 93)
- **void addParticle (TempParticleData const &pd)**
add a TempParticleData (p. 125) to the map
- **void removeParticle (ParticleID pid)**
remove a TempParticleData (p. 125) from the map
- **void addAlias (TempAliasData const &ad)**
add alias information to the alias map
- **bool hasParticleData (std::string const &name)**
check to see if this particle is already defined
- **bool hasAlias (std::string const &alias)**
check to see if this alias is already defined
- **bool hasDefinition (std::string const &def)**
check to see if this particle name is already defined
- **int size () const**
get size of particle data map
- **int aliasSize () const**
get size of alias map
- **DefTable & definitions ()**
get the list of definitions (for EvtGen)

- double **definition** (std::string const &def)
return a parameter definition (for EvtGen)
- **TempAliasData & aliasData** (std::string const &alias)
find an entry in the alias map

8.15.1 Detailed Description

The **TableBuilder** (p. 116) class is used to construct a **ParticleDataTable** (p. 85).

Author:

Marc Paterno, Walter Brown, Lynn Garren

Define this class and use the add methods to define a **ParticleDataTable** (p. 85). The destructor fills **ParticleDataTable** (p. 85) from the information in **TableBuilder** (p. 116). See the examples for user code.

Examples:

`examMyPDT.cc`, `listEvtGenNames.cc.in`, `listPDGNames.cc.in`, `listPythiaNames.cc.in`, `testHepPDT.cc`, `testReadEvtGen.cc.in`, `testReadIsajet.cc.in`, and `testReadQQ.cc.in`.

Definition at line 42 of file TableBuilder.hh.

8.15.2 Constructor & Destructor Documentation

8.15.2.1 HepPDT::TableBuilder::TableBuilder (ParticleDataTable & table, std::ostream & str = std::cerr) [inline, explicit]

create **TableBuilder** (p. 116) from a **ParticleDataTable** (p. 85)

Definition at line 49 of file TableBuilder.hh.

8.15.2.2 HepPDT::TableBuilder::~TableBuilder () [inline]

call the **ParticleDataTable** (p. 85) conversion method upon destruction

Definition at line 53 of file TableBuilder.hh.

References HepPDT::ParticleDataTable::convertTemporaryMap().

8.15.3 Member Function Documentation

8.15.3.1 TempParticleData & HepPDT::TableBuilder::getParticleData (ParticleID pid) [inline]

create a **TempParticleData** (p. 125) from a **ParticleID** (p. 93)

Examples:

`examMyPDT.cc`.

Definition at line 21 of file TableBuilder.icc.

Referenced by addData(), HepPDT::addEvtGenParticles(), HepPDT::addIsajetParticles(), addParticle(), HepPDT::addParticleTable(), HepPDT::addPDGParticles(), HepPDT::addPythiaParticles(), HepPDT::addQQParticles(), getAntiParticle(), and getParticleData().

8.15.3.2 TempParticleData & HepPDT::TableBuilder::getParticleData (std::string const & name) [inline]

create a **TempParticleData** (p. 125) from a particle name

Definition at line 35 of file TableBuilder.icc.

References getParticleData().

8.15.3.3 TempParticleData & HepPDT::TableBuilder::getAntiParticle (ParticleID pid, const std::string & aname) [inline]

create an antiparticle **TempParticleData** (p. 125) from a **ParticleID** (p. 93)

Definition at line 47 of file TableBuilder.icc.

References HepPDT::TempParticleData::antiparticle(), getParticleData(), HepPDT::ParticleID::pid(), and HepPDT::TempParticleData::tempParticleName.

Referenced by HepPDT::addPythiaParticles().

8.15.3.4 void HepPDT::TableBuilder::addParticle (TempParticleData const & pd) [inline]

add a **TempParticleData** (p. 125) to the map

Examples:

examMyPDT.cc.

Definition at line 83 of file TableBuilder.icc.

References getParticleData(), hasParticleData(), HepPDT::TempParticleData::tempID, and HepPDT::TempParticleData::tempParticleName.

Referenced by addData(), HepPDT::addEvtGenParticles(), and HepPDT::addQQParticles().

8.15.3.5 void HepPDT::TableBuilder::removeParticle (ParticleID pid) [inline]

remove a **TempParticleData** (p. 125) from the map

Examples:

examMyPDT.cc.

Definition at line 68 of file TableBuilder.hh.

Referenced by main().

8.15.3.6 void HepPDT::TableBuilder::addAlias (TempAliasData const & *ad*) [inline]

add alias information to the alias map

Definition at line 91 of file TableBuilder.icc.

References HepPDT::TempAliasData::tempAlias.

Referenced by HepPDT::addEvtGenParticles().

8.15.3.7 bool HepPDT::TableBuilder::hasParticleData (std::string const & *name*) [inline]

check to see if this particle is already defined

Definition at line 71 of file TableBuilder.icc.

Referenced by HepPDT::addEvtGenParticles(), addParticle(), and HepPDT::addQQParticles().

8.15.3.8 bool HepPDT::TableBuilder::hasAlias (std::string const & *alias*) [inline]

check to see if this alias is already defined

Definition at line 77 of file TableBuilder.icc.

Referenced by HepPDT::addEvtGenParticles().

8.15.3.9 bool HepPDT::TableBuilder::hasDefinition (std::string const & *def*) [inline]

check to see if this particle name is already defined

Definition at line 80 of file TableBuilder.hh.

References HepPDT::DefTable::hasDefinition().

8.15.3.10 int HepPDT::TableBuilder::size () const [inline]

get size of particle data map

Definition at line 85 of file TableBuilder.hh.

Referenced by HepPDT::addEvtGenParticles(), HepPDT::addIsajetParticles(), HepPDT::addParticleTable(), HepPDT::addPDGParticles(), HepPDT::addPythiaParticles(), and HepPDT::addQQParticles().

8.15.3.11 int HepPDT::TableBuilder::aliasSize () const [inline]

get size of alias map

Definition at line 87 of file TableBuilder.hh.

Referenced by HepPDT::addEvtGenParticles().

8.15.3.12 DefTable& HepPDT::TableBuilder::definitions () [inline]

get the list of definitions (for EvtGen)

Definition at line 89 of file TableBuilder.hh.

Referenced by HepPDT::addEvtGenParticles().

8.15.3.13 double HepPDT::TableBuilder::definition (std::string const & *def*) [inline]

return a parameter definition (for EvtGen)

Definition at line 91 of file TableBuilder.hh.

References HepPDT::DefTable::definition().

8.15.3.14 TempAliasData & HepPDT::TableBuilder::aliasData (std::string const & *alias*) [inline]

find an entry in the alias map

Definition at line 97 of file TableBuilder.icc.

Referenced by HepPDT::addEvtGenParticles().

The documentation for this class was generated from the following files:

- **TableBuilder.hh**
- **TableBuilder.icc**

8.16 HepPDT::TempAliasData Struct Reference

Hold Alias information from EvtGen.

```
#include <TempParticleData.hh>
```

Public Member Functions

- **TempAliasData ()**

used in a map<>

Public Attributes

- std::string **tempAlias**

the alias

- std::string **tempAliasedParticle**

the "real" particle

- std::string **tempChargeConj**

set if there is a charge conjugate alias

- TDDlist **tempAliasDecayList**

decay list for the alias

8.16.1 Detailed Description

Hold Alias information from EvtGen.

Definition at line 48 of file TempParticleData.hh.

8.16.2 Constructor & Destructor Documentation

8.16.2.1 HepPDT::TempAliasData::TempAliasData ()

used in a map<>

Definition at line 176 of file TempParticleData.cc.

8.16.3 Member Data Documentation

8.16.3.1 std::string HepPDT::TempAliasData::tempAlias

the alias

Definition at line 54 of file TempParticleData.hh.

Referenced by HepPDT::TableBuilder::addAlias(), and HepPDT::parseEvtGenAlias().

8.16.3.2 std::string HepPDT::TempAliasData::tempAliasedParticle

the "real" particle

Definition at line 55 of file TempParticleData.hh.

Referenced by HepPDT::parseEvtGenAlias().

8.16.3.3 std::string HepPDT::TempAliasData::tempChargeConj

set if there is a charge conjugate alias

Definition at line 56 of file TempParticleData.hh.

Referenced by HepPDT::addEvtGenParticles().

8.16.3.4 TDDlist HepPDT::TempAliasData::tempAliasDecayList

decay list for the alias

Definition at line 57 of file TempParticleData.hh.

The documentation for this struct was generated from the following files:

- **TempParticleData.hh**
- **TempParticleData.cc**

8.17 HepPDT::TempConstituent Struct Reference

Temporary constituent (e.g., quark) information.

```
#include <TempParticleData.hh>
```

Public Member Functions

- **TempConstituent (ParticleID p=ParticleID(0), int m=-1)**

Public Attributes

- **ParticleID tempConstituentPID**
- **int tempMultiplicity**

8.17.1 Detailed Description

Temporary constituent (e.g., quark) information.

Definition at line 25 of file TempParticleData.hh.

8.17.2 Constructor & Destructor Documentation

8.17.2.1 HepPDT::TempConstituent::TempConstituent (ParticleID *p* = ParticleID(0), int *m* = -1) [inline]

Definition at line 26 of file TempParticleData.hh.

8.17.3 Member Data Documentation

8.17.3.1 ParticleID HepPDT::TempConstituent::tempConstituentPID

Definition at line 28 of file TempParticleData.hh.

Referenced by HepPDT::TempParticleData::processPID().

8.17.3.2 int HepPDT::TempConstituent::tempMultiplicity

Definition at line 29 of file TempParticleData.hh.

Referenced by HepPDT::TempParticleData::processPID().

The documentation for this struct was generated from the following file:

- **TempParticleData.hh**

8.18 HepPDT::TempDecayData Struct Reference

temporary holder for decay data

```
#include <TempParticleData.hh>
```

Public Attributes

- std::string **tempDecayName**
- double **tempBranchingFraction**
- std::vector< std::string > **tempDaughterList**
list of decay particles
- std::vector< double > **tempDecayParameters**
other decay parameters

8.18.1 Detailed Description

temporary holder for decay data

Definition at line 33 of file TempParticleData.hh.

8.18.2 Member Data Documentation

8.18.2.1 std::string HepPDT::TempDecayData::tempDecayName

Type of decay. Use string to hold int if necessary.

Definition at line 36 of file TempParticleData.hh.

8.18.2.2 double HepPDT::TempDecayData::tempBranchingFraction

Definition at line 37 of file TempParticleData.hh.

8.18.2.3 std::vector<std::string> HepPDT::TempDecayData::tempDaughterList

list of decay particles

Definition at line 39 of file TempParticleData.hh.

8.18.2.4 std::vector<double> HepPDT::TempDecayData::tempDecayParameters

other decay parameters

Definition at line 41 of file TempParticleData.hh.

The documentation for this struct was generated from the following file:

- **TempParticleData.hh**

8.19 HepPDT::TempParticleData Struct Reference

temporary holder for Particle Data information

```
#include <TempParticleData.hh>
```

Public Member Functions

- **TempParticleData ()**
- **TempParticleData (int id)**

*construct a basic **TempParticleData** (p. 125) from just the particle ID*

- **TempParticleData (ParticleID pid)**

*construct a basic **TempParticleData** (p. 125) from just the **ParticleID** (p. 93)*

- **TempParticleData (int id, std::string const &name, std::string const &source, int oid, double charge, SpinState const &Spin, Measurement const &mass, Measurement const &wid)**

*given all the information, construct a **TempParticleData** (p. 125)*

- **TempParticleData (TempParticleData const &orig)**
- **TempParticleData & operator= (TempParticleData const &rhs)**
- **void swap (TempParticleData &other)**
- **TempParticleData antiparticle (std::string const &name)**

given a particle definition, create an antiparticle

- **bool processPID ()**

– mutator - get spin state and constituent list from PID

Public Attributes

- **ParticleID tempID**
- **std::string tempParticleName**
- **std::string tempSource**
- **int tempOriginalID**
- **double tempCharge**
- **double tempColorCharge**
- **SpinState tempSpin**
- **Measurement tempMass**
- **Measurement tempWidth**
- **double tempLowCutoff**
- **double tempHighCutoff**
- **std::vector< TempConstituent > tempQuarks**
- **TDDlist tempDecayList**

8.19.1 Detailed Description

temporary holder for Particle Data information

Examples:

examMyPDT.cc.

Definition at line 61 of file TempParticleData.hh.

8.19.2 Constructor & Destructor Documentation

8.19.2.1 HepPDT::TempParticleData::TempParticleData ()

used in a map<>

Definition at line 15 of file TempParticleData.cc.

8.19.2.2 HepPDT::TempParticleData::TempParticleData (int *id*) [explicit]

construct a basic **TempParticleData** (p. 125) from just the particle ID

Definition at line 32 of file TempParticleData.cc.

References processPID().

8.19.2.3 HepPDT::TempParticleData::TempParticleData (ParticleID *pid*) [explicit]

construct a basic **TempParticleData** (p. 125) from just the **ParticleID** (p. 93)

Definition at line 50 of file TempParticleData.cc.

References processPID().

8.19.2.4 HepPDT::TempParticleData::TempParticleData (int *id*, std::string const & *name*, std::string const & *source*, int *oid*, double *charge*, SpinState const & *Spin*, Measurement const & *mass*, Measurement const & *wid*)

given all the information, construct a **TempParticleData** (p. 125)

Definition at line 68 of file TempParticleData.cc.

8.19.2.5 HepPDT::TempParticleData::TempParticleData (**TempParticleData** const & *orig*)

Definition at line 106 of file TempParticleData.cc.

8.19.3 Member Function Documentation

8.19.3.1 TempParticleData & HepPDT::TempParticleData::operator= (TempParticleData const & *rhs*)

Definition at line 122 of file TempParticleData.cc.

References swap().

8.19.3.2 void HepPDT::TempParticleData::swap (TempParticleData & other)

Definition at line 129 of file TempParticleData.cc.

References HepPDT::swap(), tempCharge, tempColorCharge, tempDecayList, tempHighCutoff, tempID, tempLowCutoff, tempMass, tempOriginalID, tempParticleName, tempQuarks, tempSource, tempSpin, and tempWidth.

Referenced by operator=(), and HepPDT::swap().

8.19.3.3 TempParticleData HepPDT::TempParticleData::antiparticle (std::string const & name)

given a particle definition, create an antiparticle

Definition at line 87 of file TempParticleData.cc.

References HepPDT::ParticleID::pid(), processPID(), tempCharge, tempColorCharge, tempDecayList, tempHighCutoff, tempID, tempLowCutoff, tempMass, tempOriginalID, tempParticleName, tempSource, and tempWidth.

Referenced by HepPDT::TableBuilder::getAntiParticle().

8.19.3.4 bool HepPDT::TempParticleData::processPID ()

– mutator - get spin state and constituent list from PID

Definition at line 146 of file TempParticleData.cc.

References HepPDT::ParticleID::jSpin(), HepPDT::ParticleID::lSpin(), HepPDT::Quarks::nq1, HepPDT::Quarks::nq2, HepPDT::Quarks::nq3, HepPDT::ParticleID::pid(), HepPDT::ParticleID::quarks(), HepPDT::SpinState::setOrbAngMom(), HepPDT::SpinState::setSpin(), HepPDT::SpinState::setTotalSpin(), HepPDT::spinitod(), HepPDT::ParticleID::sSpin(), tempCharge, HepPDT::TempConstituent::tempConstituentPID, tempID, HepPDT::TempConstituent::tempMultiplicity, tempQuarks, tempSpin, and HepPDT::ParticleID::threeCharge().

Referenced by antiparticle(), and TempParticleData().

8.19.4 Member Data Documentation**8.19.4.1 ParticleID HepPDT::TempParticleData::tempID**

Definition at line 89 of file TempParticleData.hh.

Referenced by HepPDT::TableBuilder::addParticle(), antiparticle(), HepPDT::parseEvtGenDecayLine(), HepPDT::parseEvtGenLine(), HepPDT::detail::parseIsajetLine(), HepPDT::detail::parseParticleLine(), HepPDT::detail::parsePythiaDecayLine(), HepPDT::detail::parsePythiaLine(), HepPDT::parseQQParticle(), processPID(), and swap().

8.19.4.2 std::string HepPDT::TempParticleData::tempParticleName

Examples:

examMyPDT.cc.

Definition at line 90 of file TempParticleData.hh.

Referenced by addData(), HepPDT::TableBuilder::addParticle(), HepPDT::addQQParticles(), antiparticle(), HepPDT::TableBuilder::getAntiParticle(), HepPDT::parseEvtGenLine(), HepPDT::detail::parseIsajetLine(), HepPDT::detail::parseParticleLine(), HepPDT::detail::parsePythiaLine(), HepPDT::parseQQParticle(), and swap().

8.19.4.3 std::string HepPDT::TempParticleData::tempSource

Definition at line 91 of file TempParticleData.hh.

Referenced by HepPDT::addPDGParticles(), HepPDT::addPythiaParticles(), HepPDT::addQQParticles(), antiparticle(), HepPDT::parseEvtGenLine(), HepPDT::detail::parseParticleLine(), HepPDT::detail::parsePythiaLine(), and swap().

8.19.4.4 int HepPDT::TempParticleData::tempOriginalID

Definition at line 92 of file TempParticleData.hh.

Referenced by HepPDT::addPythiaParticles(), HepPDT::addQQParticles(), antiparticle(), HepPDT::parseEvtGenLine(), HepPDT::detail::parseParticleLine(), HepPDT::detail::parsePythiaLine(), HepPDT::parseQQParticle(), and swap().

8.19.4.5 double HepPDT::TempParticleData::tempCharge

Examples:

examMyPDT.cc.

Definition at line 93 of file TempParticleData.hh.

Referenced by addData(), antiparticle(), HepPDT::parseEvtGenLine(), HepPDT::detail::parseIsajetLine(), HepPDT::detail::parseParticleLine(), HepPDT::detail::parsePythiaLine(), HepPDT::parseQQParticle(), processPID(), and swap().

8.19.4.6 double HepPDT::TempParticleData::tempColorCharge

Definition at line 94 of file TempParticleData.hh.

Referenced by antiparticle(), HepPDT::detail::parsePythiaLine(), and swap().

8.19.4.7 SpinState HepPDT::TempParticleData::tempSpin

Examples:

examMyPDT.cc.

Definition at line 95 of file TempParticleData.hh.

Referenced by addData(), HepPDT::parseEvtGenLine(), HepPDT::detail::parseIsajetLine(), HepPDT::parseQQParticle(), processPID(), and swap().

8.19.4.8 Measurement HepPDT::TempParticleData::tempMass

Examples:

examMyPDT.cc.

Definition at line 96 of file TempParticleData.hh.

Referenced by addData(), HepPDT::addPythiaParticles(), antiparticle(), HepPDT::detail::CheckPDGEntry(), HepPDT::parseEvtGenLine(), HepPDT::detail::parseIsajetLine(), HepPDT::detail::parseParticleLine(), HepPDT::detail::parsePythiaLine(), HepPDT::parseQQParticle(), and swap().

8.19.4.9 Measurement HepPDT::TempParticleData::tempWidth

Examples:

examMyPDT.cc.

Definition at line 97 of file TempParticleData.hh.

Referenced by addData(), antiparticle(), HepPDT::detail::CheckPDGEntry(), HepPDT::parseEvtGenLine(), HepPDT::detail::parseParticleLine(), HepPDT::detail::parsePythiaLine(), HepPDT::parseQQParticle(), and swap().

8.19.4.10 double HepPDT::TempParticleData::tempLowCutoff

Definition at line 98 of file TempParticleData.hh.

Referenced by antiparticle(), HepPDT::parseQQParticle(), and swap().

8.19.4.11 double HepPDT::TempParticleData::tempHighCutoff

Definition at line 99 of file TempParticleData.hh.

Referenced by antiparticle(), HepPDT::parseEvtGenLine(), HepPDT::detail::parsePythiaLine(), HepPDT::parseQQParticle(), and swap().

8.19.4.12 std::vector<TempConstituent> HepPDT::TempParticleData::tempQuarks

Definition at line 100 of file TempParticleData.hh.

Referenced by processPID(), and swap().

8.19.4.13 TDDlist HepPDT::TempParticleData::tempDecayList

Definition at line 101 of file TempParticleData.hh.

Referenced by antiparticle(), and swap().

The documentation for this struct was generated from the following files:

- **TempParticleData.hh**
- **TempParticleData.cc**

Chapter 9

HepPDT File Documentation

9.1 addEvtGenParticles.cc File Reference

```
#include <string>
#include <algorithm>
#include <iostream>
#include <sstream>
#include "HepPDT/defs.h"
#include "HepPDT/TableBuilder.hh"
#include "HepPDT/TempParticleData.hh"
```

Namespaces

- namespace **HepPDT**

Functions

- bool **HepPDT::getEvtGenLineType** (std::string <ype, int &id, std::string &name, const std::string &pdline)
- void **HepPDT::parseEvtGenLine** (TempParticleData &tpd, const std::string &pdline)
- void **HepPDT::parseEvtGenAlias** (TempAliasData &tad, const std::string &pdline)
- bool **HepPDT::parseEvtGenDecayLine** (TempParticleData &tpd, const std::string &pdline)
- bool **HepPDT::parseEvtGenAliasDecayLine** (TempAliasData &tad, const std::string &pdline)
- void **HepPDT::parseEvtGenConj** (std::string &cname, const std::string &pdline)
- void **HepPDT::parseEvtGenDefinition** (std::string &def, double &val, const std::string &pdline)
- bool **HepPDT::addEvtGenParticles** (std::istream &pdfile, TableBuilder &tb)

read EvtGen input and add particles to the table

9.2 addHerwigParticles.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/TableBuilder.hh"
```

Namespaces

- namespace **HepPDT**

Functions

- bool **HepPDT::addHerwigParticles** (std::istream &pdffile, TableBuilder &tb)

9.3 addIsajetParticles.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/TempParticleData.hh"
#include "HepPDT/TableBuilder.hh"
```

Namespaces

- namespace **HepPDT**
- namespace **HepPDT::detail**

Functions

- bool **HepPDT::addIsajetParticles** (std::istream &pdffile, TableBuilder &tb)
read Isajet particle input and add particles to the table
- void **HepPDT::detail::parseIsajetLine** (TempParticleData &tpd, const std::string &pdline)
for internal use

9.4 addParticleTable.cc File Reference

```
#include <iostream>
#include <string>
#include <sstream>
#include "HepPDT/defs.h"
#include "HepPDT/TempParticleData.hh"
#include "HepPDT/TableBuilder.hh"
#include "HepPID/ParticleName.hh"
```

Namespaces

- namespace **HepPDT**
- namespace **HepPDT::detail**

Functions

- bool **HepPDT::addParticleTable** (std::istream &pdffile, TableBuilder &tb)
read particle.tbl (or something similar) and add particles to the table
- bool **HepPDT::detail::getParticleID** (int &id, const std::string &pdline)
for internal use
- void **HepPDT::detail::parseParticleLine** (TempParticleData &tpd, const std::string &pdline)
for internal use

9.5 addPDGParticles.cc File Reference

```
#include <iostream>
#include <string>
#include <vector>
#include <cmath>
#include <sstream>
#include "HepPDT/defs.h"
#include "HepPDT/TempParticleData.hh"
#include "HepPDT/TableBuilder.hh"
```

Namespaces

- namespace **HepPDT**
- namespace **HepPDT::detail**

Functions

- bool **HepPDT::addPDGParticles** (std::istream &pdf, TableBuilder &tb)
read PDG input and add particles to the table
- void **HepPDT::detail::parsePDGline** (TempParticleData &tpd, std::string &pdline)
for internal use
- bool **HepPDT::detail::CheckPDGEntry** (TempParticleData &tpd, const std::string &, double, double)
for internal use

9.6 addPythiaParticles.cc File Reference

```
#include <iostream>
#include <string>
#include <sstream>
#include "HepPDT/defs.h"
#include "HepPDT/TempParticleData.hh"
#include "HepPDT/TableBuilder.hh"
```

Namespaces

- namespace **HepPDT**
- namespace **HepPDT::detail**

Functions

- bool **HepPDT::addPythiaParticles** (std::istream &pdf, TableBuilder &tb)
read Pythia input and add particles to the table
- void **HepPDT::detail::parsePythiaLine** (TempParticleData &tpd, int &anti, std::string &aname, const std::string &pdline)
for internal use
- void **HepPDT::detail::parsePythiaDecayLine** (TempParticleData &tpd, const std::string &pdline)
for internal use

9.7 addQQParticles.cc File Reference

```
#include <sstream>
#include <string>
#include <iostream>
#include "HepPDT/defs.h"
#include "HepPDT/TableBuilder.hh"
#include "HepPDT/TempParticleData.hh"
#include "HepPID/ParticleIDTranslations.hh"
```

Namespaces

- namespace **HepPDT**

Functions

- bool **HepPDT::getQQLineType** (std::string <ype, int &id, std::string &name, const std::string &pdline)
- bool **HepPDT::parseQQDecayLine** (const std::string &pdline)
- void **HepPDT::parseQQParticle** (TempParticleData &tpd, const std::string &pdline)
- bool **HepPDT::addQQParticles** (std::istream &pdf, TableBuilder &tb)

read QQ input and add particles to the table

9.8 calculateWidthFromLifetime.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/TempParticleData.hh"
```

Namespaces

- namespace **HepPDT**

Functions

- double **HepPDT::calculateWidthFromLifetime** (double)

9.9 Constituent.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/Constituent.hh"
```

Namespaces

- namespace **HepPDT**

9.10 Constituent.hh File Reference

```
#include <algorithm>
#include "HepPDT/ParticleID.hh"
```

Namespaces

- namespace **HepPDT**

Classes

- class **HepPDT::Constituent**

*The **Constituent** (p. 63) class has information about constituent particles.*

Functions

- void **HepPDT::swap** (Constituent &first, Constituent &second)

9.11 convertTemporaryMap.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/ParticleID.hh"
#include "HepPDT/ParticleDataTable.hh"
#include "HepPDT/ParticleData.hh"
#include "HepPDT/TempParticleData.hh"
#include "HepPDT/Version.hh"
```

Namespaces

- namespace **HepPDT**

9.12 DefTable.cc File Reference

```
#include <iostream>
#include "HepPDT/defs.h"
#include "HepPDT/DefTable.hh"
```

Namespaces

- namespace **HepPDT**

9.13 DefTable.hh File Reference

```
#include <string>
#include <map>
```

Namespaces

- namespace **HepPDT**

Classes

- class **HepPDT::DefTable**

*The **DefTable** (p. 66) class holds EvtGen definitions.*

9.14 examListHerwig.cc File Reference

```
#include <fstream>
#include <iostream>
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPID/ParticleName.hh"
#include "HepPID/Version.hh"
```

Functions

- void **list_herwig_init_** (int *nevt)
- void **list_herwig_end_** ()
- void **get_list_size_** (int *)
- void **get_herwig_name_** (int *ihwg, int *id, char *name)
- int **main** ()

9.14.1 Function Documentation

9.14.1.1 void **get_herwig_name_** (int * *ihwg*, int * *id*, char * *name*)

Examples:

examListHerwig.cc.

Referenced by main().

9.14.1.2 void **get_list_size_** (int *)

Examples:

examListHerwig.cc.

Referenced by main().

9.14.1.3 void **list_herwig_end_** ()

Examples:

examListHerwig.cc.

Referenced by main().

9.14.1.4 void **list_herwig_init_** (int * *nevt*)

Author:

Lynn Garren

list Herwig particle ID translations Get ID list directly from Herwig

When mixing C++ and Fortran, the main program must be C++

Examples:

examListHerwig.cc.

Referenced by main().

9.14.1.5 int main ()

Definition at line 29 of file examListHerwig.cc.

References `get_herwig_name_()`, `get_list_size_()`, `list_herwig_end_()`, `list_herwig_init_()`, `HepPID::particleName()`, `HepPID::translateHerwigtoPDT()`, and `HepPID::writeVersion()`.

9.15 examListIsajet.cc File Reference

```
#include <fstream>
#include <iostream>
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPID/ParticleName.hh"
#include "HepPID/Version.hh"
```

Functions

- void **list_isajet_init_()**
- void **flavor_**(int *, int *, int *, int *, int *)
- void **get_label_**(int *id, char *name)
- int **main()**

9.15.1 Function Documentation

9.15.1.1 void flavor_ (int *, int *, int *, int *, int *)

Examples:

examListIsajet.cc.

Referenced by main().

9.15.1.2 void get_label_ (int * *id*, char * *name*)

Examples:

examListIsajet.cc.

Referenced by main().

9.15.1.3 void list_isajet_init_()

Author:

Lynn Garren

List the isajet particle ID translations Get ID list directly from isajet

When mixing C++ and Fortran, the main program must be C++

Examples:

examListIsajet.cc.

Referenced by main().

9.15.1.4 int main()

Definition at line 27 of file examListIsajet.cc.

References flavor_(), get_label_(), list_isajet_init_(), HepPID::particleName(), HepPID::translateIsajetto-PDT(), and HepPID::writeVersion().

9.16 examListPythia.cc File Reference

```
#include <fstream>
#include <string>
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPID/ParticleIDMethods.hh"
#include "HepPID/ParticleName.hh"
#include "HepPID/Version.hh"
```

Functions

- void **list_pythia_()**
- void **getkf_(int *, int *)**
- void **getpname_(int *, int *, char *name)**
- void **writeLine(int &i, int &kf, int &id, std::string &name, std::string &pn, std::ofstream &os)**
- int **main()**

9.16.1 Function Documentation

9.16.1.1 void **getkf_(int *, int *)**

Examples:

examListPythia.cc.

Referenced by main().

9.16.1.2 void **getpname_(int *, int *, char * name)**

Examples:

examListPythia.cc.

Referenced by main().

9.16.1.3 void **list_pythia_()**

Author:

Lynn Garren

list Pythia particle ID translations Get ID list directly from Pythia

When mixing C++ and Fortran, the main program must be C++

Examples:

examListPythia.cc.

Referenced by main().

9.16.1.4 int main ()

Definition at line 31 of file examListPythia.cc.

References `getkf_()`, `getpyname_()`, `HepPID::isValid()`, `list_pythia_()`, `HepPID::particleName()`, `HepPID::translatePythiaToPDT()`, `writeLine()`, and `HepPID::writeVersion()`.

9.16.1.5 void writeLine (int & i, int & kf, int & id, std::string & name, std::string & pn, std::ofstream & os)

Examples:

examListPythia.cc.

Definition at line 78 of file examListPythia.cc.

Referenced by `main()`.

9.17 examMyPDT.cc File Reference

```
#include "HepPDT/defs.h"
#include <fstream>
#include <string>
#include "HepPDT/TableBuilder.hh"
#include "HepPDT/ParticleDataTable.hh"
#include "HepPDT/TempParticleData.hh"
```

Functions

- void **addData** (**HepPDT::TableBuilder** &tb, std::string const &name, int const id, double const mass, double const charge, double const width, double const tspin)
- int **main** ()

9.17.1 Function Documentation

9.17.1.1 void addData (**HepPDT::TableBuilder** & *tb*, std::string const & *name*, int const *id*, double const *mass*, double const *charge*, double const *width*, double const *tspin*)

Author:

Lynn Garren

create a custom PDT using our own definitions and write out the resulting PDT

Examples:

examMyPDT.cc.

Definition at line 67 of file examMyPDT.cc.

References HepPDT::TableBuilder::addParticle(), HepPDT::TableBuilder::getParticleData(), HepPDT::TempParticleData::tempCharge, HepPDT::TempParticleData::tempMass, HepPDT::TempParticleData::tempParticleName, HepPDT::TempParticleData::tempSpin, and HepPDT::TempParticleData::tempWidth.

Referenced by main().

9.17.1.2 int main ()

Examples:

examListHerwig.cc, **examListIsajet.cc**, **examListPythia.cc**, **examMyPDT.cc**, **listEvtGenNames.cc.in**, **listEvtGenTranslation.cc**, **listHerwigTranslation.cc**, **listIsajetTranslation.cc**, **listParticleNames.cc**, **listPDGNames.cc.in**, **listPDGTranslation.cc**, **listPythiaNames.cc.in**, **listPythiaTranslation.cc**, **listQQTranslation.cc**, **testHepPDT.cc**, **testPID.cc**, **testReadEvtGen.cc.in**, **testReadIsajet.cc.in**, and **testReadQQ.cc.in**.

Definition at line 23 of file examMyPDT.cc.

References addData(), HepPDT::ParticleData::name(), HepPDT::ParticleDataTable::particle(), HepPDT::TableBuilder::removeParticle(), and HepPDT::ParticleDataTable::writeParticleData().

9.18 getIsajetID.cc File Reference

```
#include <iostream>
#include <string>
#include <sstream>
```

Namespaces

- namespace **HepPDT**
- namespace **HepPDT::detail**

Functions

- bool **HepPDT::detail::getIsajetID** (int &id, const std::string &pdline)
for internal use

9.19 getPDGpid.cc File Reference

```
#include <string>
#include <vector>
#include <sstream>
#include "HepPDT/defs.h"
#include "HepPDT/TableBuilder.hh"
```

Namespaces

- namespace **HepPDT**
- namespace **HepPDT::detail**

Functions

- void **HepPDT::detail::getPDGpid** (std::vector< int > &idlist, std::string &pdline)
for internal use
- void **HepPDT::detail::getPDGnames** (std::vector< std::string > &namelst, std::string &pdline)
for internal use

9.20 getPythiaid.cc File Reference

```
#include <iostream>
#include <string>
#include <sstream>
#include "HepPDT/defs.h"
#include "HepPDT/TableBuilder.hh"
```

Namespaces

- namespace **HepPDT**
- namespace **HepPDT::detail**

Functions

- bool **HepPDT::detail::getPythiaid** (int &id, const std::string &pdline)
for internal use

9.21 hasMethods.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/ParticleData.hh"
```

Namespaces

- namespace **HepPDT**

9.22 HeavyIonUnknownID.cc File Reference

```
#include "HepPDT/HeavyIonUnknownID.hh"
```

Namespaces

- namespace **HepPDT**

9.23 HeavyIonUnknownID.hh File Reference

```
#include "HepPDT/ProcessUnknownID.hh"
#include "HepPDT/ParticleData.hh"
```

Namespaces

- namespace **HepPDT**

Classes

- class **HepPDT::HeavyIonUnknownID**

*The **HeavyIonUnknownID** (p. 69) class inherits from **ProcessUnknownID** (p. 104).*

9.24 lifetime.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/ResonanceStructure.hh"
```

Namespaces

- namespace **HepPDT**

9.25 list_of_examples.cc File Reference

9.26 list_of_tests.cc File Reference

9.27 listEvtGenNames.cc.in File Reference

```
#include <fstream>
#include <iostream>
#include "HepPDT/TableBuilder.hh"
#include "HepPDT/ParticleDataTable.hh"
```

Functions

- int **main** ()

9.27.1 Function Documentation

9.27.1.1 int main ()

Definition at line 15 of file listEvtGenNames.cc.in.

References HepPDT::addEvtGenParticles(), and HepPDT::ParticleDataTable::writeParticleTranslation().

9.28 listEvtGenTranslation.cc File Reference

```
#include <fstream>
#include <iostream>
#include "HepPID/ParticleIDTranslations.hh"
```

Functions

- int **main ()**

9.28.1 Function Documentation

9.28.1.1 int main ()

Definition at line 14 of file listEvtGenTranslation.cc.

References HepPID::writeEvtGenTranslation().

9.29 listHerwigTranslation.cc File Reference

```
#include <fstream>
#include <iostream>
#include "HepPID/ParticleIDTranslations.hh"
```

Functions

- int **main ()**

9.29.1 Function Documentation

9.29.1.1 int main ()

Definition at line 14 of file listHerwigTranslation.cc.

References HepPID::writeHerwigTranslation().

9.30 listIsajetTranslation.cc File Reference

```
#include <fstream>
#include <iostream>
#include "HepPID/ParticleIDTranslations.hh"
```

Functions

- int **main ()**

9.30.1 Function Documentation

9.30.1.1 int main ()

Definition at line 14 of file listIsajetTranslation.cc.

References HepPID::writeIsajetTranslation().

9.31 listParticleNames.cc File Reference

```
#include <fstream>
#include <iostream>
#include "HepPID/ParticleName.hh"
```

Functions

- int **main ()**

9.31.1 Function Documentation

9.31.1.1 int main ()

Definition at line 16 of file listParticleNames.cc.

References HepPID::listParticleNames().

9.32 listPDGNames.cc.in File Reference

```
#include <fstream>
#include <iostream>
#include "HepPDT/TableBuilder.hh"
#include "HepPDT/ParticleDataTable.hh"
```

Functions

- int **main ()**

9.32.1 Function Documentation

9.32.1.1 int main ()

Definition at line 15 of file listPDGNames.cc.in.

References HepPDT::addPDGParticles(), and HepPDT::ParticleDataTable::writeParticleTranslation().

9.33 listPDGTranslation.cc File Reference

```
#include <fstream>
#include <iostream>
#include "HepPID/ParticleIDTranslations.hh"
```

Functions

- int **main ()**

9.33.1 Function Documentation

9.33.1.1 int main ()

Definition at line 14 of file listPDGTranslation.cc.

References HepPID::writePDGTranslation().

9.34 listPythiaNames.cc.in File Reference

```
#include <fstream>
#include <iostream>
#include "HepPDT/TableBuilder.hh"
#include "HepPDT/ParticleDataTable.hh"
```

Functions

- int **main ()**

9.34.1 Function Documentation

9.34.1.1 int main ()

Definition at line 15 of file listPythiaNames.cc.in.

References HepPDT::addPythiaParticles(), and HepPDT::ParticleDataTable::writeParticleTranslation().

9.35 listPythiaTranslation.cc File Reference

```
#include <fstream>
#include <iostream>
#include "HepPID/ParticleIDTranslations.hh"
```

Functions

- int **main ()**

9.35.1 Function Documentation

9.35.1.1 int main ()

Definition at line 14 of file listPythiaTranslation.cc.

References HepPID::writePythiaTranslation().

9.36 listQQTranslation.cc File Reference

```
#include <fstream>
#include <iostream>
#include "HepPID/ParticleIDTranslations.hh"
```

Functions

- int **main ()**

9.36.1 Function Documentation

9.36.1.1 int main ()

Definition at line 14 of file listQQTranslation.cc.

References HepPID::writeQQTranslation().

9.37 Measurement.hh File Reference

```
#include "HepPDT/Measurement.icc"
```

Namespaces

- namespace **HepPDT**

Classes

- class **HepPDT::Measurement**

*The **Measurement** (p. 71) class defines a value with its error.*

Functions

- void **HepPDT::swap** (Measurement &first, Measurement &second)

9.38 Measurement.icc File Reference

```
#include <algorithm>
```

Namespaces

- namespace **HepPDT**

Functions

- double **HepPDT::NaN ()**

9.39 ParticleData.hh File Reference

```
#include <string>
#include <vector>
#include "HepPDT/ParticleID.hh"
#include "HepPDT/SpinState.hh"
#include "HepPDT/Constituent.hh"
#include "HepPDT/ResonanceStructure.hh"
#include "HepPDT/TempParticleData.hh"
#include "HepPDT/ParticleData.icc"
```

Namespaces

- namespace **HepPDT**

Classes

- class **HepPDT::ParticleData**

*The **ParticleData** (p. 74) class holds basic particle data.*

Functions

- void **HepPDT::swap** (ParticleData &first, ParticleData &second)

9.40 ParticleData.icc File Reference

```
#include <algorithm>
```

Namespaces

- namespace **HepPDT**

9.41 ParticleDataTable.cc File Reference

```
#include <iostream>
#include <string>
#include <map>
#include <list>
#include "HepPDT/defs.h"
#include "HepPDT/ParticleDataTable.hh"
```

Namespaces

- namespace **HepPDT**

9.42 ParticleDataTable.hh File Reference

```
#include <iostream>
#include <string>
#include <map>
#include <list>
#include "HepPDT/ParticleID.hh"
#include "HepPDT/ParticleData.hh"
#include "HepPDT/ParticleDataTableComparison.hh"
#include "HepPDT/ProcessUnknownID.hh"
#include "HepPDT/SimpleProcessUnknownID.hh"
#include "HepPDT/Version.hh"
#include "HepPDT/ParticleDataTable.icc"
```

Namespaces

- namespace **HepPDT**

Classes

- class **HepPDT::ParticleDataTable**

The *ParticleDataTable* (p. 85) class is the core of **HepPDT** (p. 23).

Functions

- bool **HepPDT::writePDGStream** (std::ostream &os, const ParticleDataTable &table)
- bool **HepPDT::writePythiaStream** (std::ostream &os, const ParticleDataTable &table)
- bool **HepPDT::writeHerwigStream** (std::ostream &os, const ParticleDataTable &table)
- bool **HepPDT::writeIsajetStream** (std::ostream &os, const ParticleDataTable &table)
- bool **HepPDT::writeQQStream** (std::ostream &os, const ParticleDataTable &table)
- bool **HepPDT::writeEvtGenStream** (std::ostream &os, const ParticleDataTable &table)

9.43 ParticleDataTable.icc File Reference

Namespaces

- namespace **HepPDT**

9.44 ParticleDataTableComparison.hh File Reference

```
#include "HepPDT/ParticleID.hh"
```

Namespaces

- namespace **HepPDT**

Classes

- class **HepPDT::ParticleDataTableComparison**

*The **ParticleDataTableComparison** (p. 92) class provides a utility for sorting the PDT.*

9.45 ParticleID.cc File Reference

```
#include <stdlib.h>
#include <cmath>
#include "HepPDT/defs.h"
#include "HepPDT/ParticleID.hh"
```

Namespaces

- namespace **HepPDT**

9.46 ParticleID.hh File Reference

```
#include <string>
#include <algorithm>
#include "HepPID/ParticleName.hh"
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPDT/ParticleID.icc"
```

Namespaces

- namespace **HepPDT**

Classes

- struct **HepPDT::Quarks**
constituent quarks
- class **HepPDT::ParticleID**
*The **ParticleID** (p. 93) has various utilities to extract information from the particle ID.*

Enumerations

- enum **HepPDT::location** {
 HepPDT::nj = 1, **HepPDT::nq3**, **HepPDT::nq2**, **HepPDT::nq1**,
 HepPDT::nl, **HepPDT::nr**, **HepPDT::n**, **HepPDT::n8**,
 HepPDT::n9, **HepPDT::n10** }

Functions

- double **HepPDT::spinitod** (int js)
convert from 2J+1 to the actual spin value
- int **HepPDT::spindtoi** (double spin)
convert an actual spin to 2J+1
- void **HepPDT::swap** (ParticleID &first, ParticleID &second)

9.47 ParticleID.icc File Reference

Namespaces

- namespace **HepPDT**

9.48 ParticleIDMethods.cc File Reference

```
#include <cmath>
#include "HepPID/ParticleIDMethods.hh"
#include "HepPID/ParticleName.hh"
```

Namespaces

- namespace **HepPID**

Functions

- int **HepPID::abspid** (const int &pid)
absolute value of particle ID
- int **HepPID::extraBits** (const int &pid)
- unsigned short **HepPID::digit** (location loc, const int &pid)
return the digit at a named location in the PID
- int **HepPID::fundamentalID** (const int &pid)
extract fundamental ID (1-100) if this is a "fundamental" particle
- int **HepPID::Z** (const int &pid)
- int **HepPID::A** (const int &pid)
- int **HepPID::lambda** (const int &pid)
- bool **HepPID::isValid** (const int &pid)
is this a valid ID?
- bool **HepPID::hasFundamentalAnti** (const int &pid)
if this is a fundamental particle, does it have a valid antiparticle?
- bool **HepPID::isMeson** (const int &pid)
is this a valid meson ID?
- bool **HepPID::isBaryon** (const int &pid)
is this a valid baryon ID?
- bool **HepPID::isDiQuark** (const int &pid)
is this a valid diquark ID?
- bool **HepPID::isHadron** (const int &pid)
is this a valid hadron ID?
- bool **HepPID::isLepton** (const int &pid)
is this a valid lepton ID?
- bool **HepPID::isNucleus** (const int &pid)
is this a valid ion ID?

- bool **HepPID::isPentaquark** (const int &pid)
is this a valid pentaquark ID?
- bool **HepPID::isSUSY** (const int &pid)
is this a valid SUSY ID?
- bool **HepPID::isRhadron** (const int &pid)
is this a valid R-hadron ID?
- bool **HepPID::hasUp** (const int &pid)
does this particle contain an up quark?
- bool **HepPID::hasDown** (const int &pid)
does this particle contain a down quark?
- bool **HepPID::hasStrange** (const int &pid)
does this particle contain a strange quark?
- bool **HepPID::hasCharm** (const int &pid)
does this particle contain a charm quark?
- bool **HepPID::hasBottom** (const int &pid)
does this particle contain a bottom quark?
- bool **HepPID::hasTop** (const int &pid)
does this particle contain a top quark?
- int **HepPID::jSpin** (const int &pid)
jSpin returns $2J+1$, where J is the total spin
- int **HepPID::sSpin** (const int &pid)
sSpin returns $2S+1$, where S is the spin
- int **HepPID::lSpin** (const int &pid)
lSpin returns $2L+1$, where L is the orbital angular momentum
- int **HepPID::threeCharge** (const int &pid)
return 3 times the charge (3 x quark charge is an int)

9.49 ParticleIDMethods.hh File Reference

Namespaces

- namespace **HepPID**

Enumerations

- enum **HepPID::location** {

 HepPID::nj = 1, **HepPID::nq3**, **HepPID::nq2**, **HepPID::nq1**,

 HepPID::nl, **HepPID::nr**, **HepPID::n**, **HepPID::n8**,

 HepPID::n9, **HepPID::n10** }

Functions

- unsigned short **HepPID::digit** (location loc, const int &pid)

return the digit at a named location in the PID
- int **HepPID::A** (const int &pid)
- int **HepPID::Z** (const int &pid)
- int **HepPID::lambda** (const int &pid)
- int **HepPID::abspid** (const int &pid)

absolute value of particle ID
- int **HepPID::fundamentalID** (const int &pid)

extract fundamental ID (1-100) if this is a "fundamental" particle
- bool **HepPID::hasFundamentalAnti** (const int &pid)

if this is a fundamental particle, does it have a valid antiparticle?
- int **HepPID::extraBits** (const int &pid)
- bool **HepPID::isValid** (const int &pid)

is this a valid ID?
- bool **HepPID::isMeson** (const int &pid)

is this a valid meson ID?
- bool **HepPID::isBaryon** (const int &pid)

is this a valid baryon ID?
- bool **HepPID::isDiQuark** (const int &pid)

is this a valid diquark ID?
- bool **HepPID::isHadron** (const int &pid)

is this a valid hadron ID?
- bool **HepPID::isLepton** (const int &pid)

is this a valid lepton ID?

- bool **HepPID::isNucleus** (const int &pid)
is this a valid ion ID?
- bool **HepPID::isPentaquark** (const int &pid)
is this a valid pentaquark ID?
- bool **HepPID::isSUSY** (const int &pid)
is this a valid SUSY ID?
- bool **HepPID::isRhadron** (const int &pid)
is this a valid R-hadron ID?
- bool **HepPID::hasUp** (const int &pid)
does this particle contain an up quark?
- bool **HepPID::hasDown** (const int &pid)
does this particle contain a down quark?
- bool **HepPID::hasStrange** (const int &pid)
does this particle contain a strange quark?
- bool **HepPID::hasCharm** (const int &pid)
does this particle contain a charm quark?
- bool **HepPID::hasBottom** (const int &pid)
does this particle contain a bottom quark?
- bool **HepPID::hasTop** (const int &pid)
does this particle contain a top quark?
- int **HepPID::jSpin** (const int &pid)
jSpin returns $2J+1$, where J is the total spin
- int **HepPID::sSpin** (const int &pid)
sSpin returns $2S+1$, where S is the spin
- int **HepPID::lSpin** (const int &pid)
lSpin returns $2L+1$, where L is the orbital angular momentum
- int **HepPID::threeCharge** (const int &pid)
return 3 times the charge (3 x quark charge is an int)

9.50 ParticleIDTranslations.hh File Reference

```
#include <iostream>
```

Namespaces

- namespace **HepPID**

Functions

- int **HepPID::translateHerwigtoPDT** (const int herwigID)
translate Herwig to PDG standard
- int **HepPID::translatePDTtoHerwig** (const int pid)
translate PDG standard to Herwig
- void **HepPID::writeHerwigTranslation** (std::ostream &os)
output the translation list
- int **HepPID::translateIsajettoPDT** (const int isajetID)
translate Isajet to PDG standard
- int **HepPID::translatePDTtoIsajet** (const int pid)
translate PDG standard to Isajet
- void **HepPID::writeIsajetTranslation** (std::ostream &os)
output the translation list
- int **HepPID::translatePythiatoPDT** (const int pythiaID)
translate Pythia to PDG standard
- int **HepPID::translatePDTtoPythia** (const int pid)
translate PDG standard to Pythia
- void **HepPID::writePythiaTranslation** (std::ostream &os)
output the translation list
- int **HepPID::translateEvtGentoPDT** (const int evtGenID)
translate EvtGen to PDG standard
- int **HepPID::translatePDTtoEvtGen** (const int pid)
translate PDG standard to EvtGen
- void **HepPID::writeEvtGenTranslation** (std::ostream &os)
output the translation list
- int **HepPID::translatePDGtabletoPDT** (const int pdgID)
translate PDG table to PDG standard

- int **HepPID::translatePDTtoPDGtable** (const int pid)
translate PDG standard to PDG table
- void **HepPID::writePDGTranslation** (std::ostream &os)
output the translation list
- int **HepPID::translateQQtoPDT** (const int qqID)
translate QQ to PDG standard
- int **HepPID::translatePDTtoQQ** (const int pid)
translate PDG standard to QQ
- int **HepPID::translateQQbar** (const int id)
QQ helper function.
- int **HepPID::translateInverseQQbar** (const int id)
QQ helper function.
- void **HepPID::writeQQTranslation** (std::ostream &os)
output the translation list
- int **HepPID::translateGeanttoPDT** (const int geantID)
translate Geant3 to PDG standard
- int **HepPID::translatePDTtoGeant** (const int pid)
translate PDG standard to Geant3

9.51 ParticleName.cc File Reference

```
#include <string>
#include <map>
#include <iostream>
#include <iomanip>
#include <utility>
#include "HepPID/ParticleName.hh"
#include "HepPID/Version.hh"
```

Namespaces

- namespace **HepPID**

Classes

- class **HepPID::ParticleNameMap**

Typedefs

- typedef std::map< int, std::string > **HepPID::ParticleIdMap**
- typedef std::map< std::string, int > **HepPID::ParticleLookupMap**

Functions

- ParticleNameMap const & **HepPID::ParticleNameInit** ()
 - void **HepPID::writeParticleNameLine** (int i, std::ostream &os)
 - ParticleNameMap const & **HepPID::getParticleNameMap** ()
 - access the ParticleNameMap (p. 102) for other purposes*
- bool **HepPID::validParticleName** (const int &)
 - verify that this number has a valid name*
- bool **HepPID::validParticleName** (const std::string &)
 - verify that this string has a valid id*
- std::string **HepPID::particleName** (const int &)
 - get a known HepPID (p. 37) Particle name*
- int **HepPID::particleName** (const std::string &)
 - lookup a known ID*
- void **HepPID::listParticleNames** (std::ostream &os)
 - list all known names*

9.52 ParticleName.hh File Reference

```
#include <string>
#include <map>
#include <iostream>
```

Namespaces

- namespace **HepPID**

Functions

- std::string **HepPID::particleName** (const int &)
get a known HepPID (p. 37) Particle name
- int **HepPID::particleName** (const std::string &)
lookup a known ID
- void **HepPID::listParticleNames** (std::ostream &os)
list all known names
- bool **HepPID::validParticleName** (const int &)
verify that this number has a valid name
- bool **HepPID::validParticleName** (const std::string &)
verify that this string has a valid id
- ParticleNameMap const & **HepPID::getParticleNameMap** ()
access the ParticleNameMap (p. 102) for other purposes

9.53 ProcessUnknownID.hh File Reference

```
#include "HepPDT/ParticleID.hh"
#include "HepPDT/ParticleData.hh"
```

Namespaces

- namespace **HepPDT**

Classes

- class **HepPDT::ProcessUnknownID**

*The **ProcessUnknownID** (p. 104) class is abstract.*

9.54 quarks.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/ParticleID.hh"
```

Namespaces

- namespace **HepPDT**

9.55 ResonanceStructure.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/ResonanceStructure.hh"
```

Namespaces

- namespace **HepPDT**

9.56 ResonanceStructure.hh File Reference

```
#include <algorithm>
#include "HepPDT/Measurement.hh"
```

Namespaces

- namespace **HepPDT**

Classes

- class **HepPDT::ResonanceStructure**

*The **ResonanceStructure** (p. 108) class is holds mass and width information.*

Functions

- void **HepPDT::swap** (ResonanceStructure &first, ResonanceStructure &second)

9.57 SimpleProcessUnknownID.hh File Reference

```
#include "HepPDT/ProcessUnknownID.hh"  
#include "HepPDT/ParticleData.hh"
```

Namespaces

- namespace **HepPDT**

Classes

- class **HepPDT::SimpleProcessUnknownID**

*The **SimpleProcessUnknownID** (p. 112) class inherits from **ProcessUnknownID** (p. 104).*

9.58 spindtoi.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/ParticleID.hh"
```

Namespaces

- namespace **HepPDT**

Functions

- int **HepPDT::spindtoi** (double spin)
convert an actual spin to 2J+I

9.59 spinitod.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/ParticleID.hh"
```

Namespaces

- namespace **HepPDT**

Functions

- double **HepPDT::spinitod** (int js)
convert from 2J+1 to the actual spin value

9.60 SpinState.hh File Reference

```
#include "HepPDT/SpinState.icc"
```

Namespaces

- namespace **HepPDT**

Classes

- class **HepPDT::SpinState**

The SpinState (p. 113) class holds spin information.

Functions

- void **HepPDT::swap** (SpinState &first, SpinState &second)

9.61 SpinState.icc File Reference

```
#include <algorithm>
```

Namespaces

- namespace **HepPDT**

9.62 stringtodouble.cc File Reference

```
#include <sstream>
#include "HepPDT/defs.h"
#include "HepPDT/stringtodouble.hh"
```

Namespaces

- namespace **HepPDT**

Functions

- double **HepPDT::stringtodouble** (std::string &numb)
extract a double from a string

9.63 stringtodouble.hh File Reference

```
#include <string>
```

Namespaces

- namespace **HepPDT**

Functions

- double **HepPDT::stringtodouble** (std::string &numb)
extract a double from a string

9.64 TableBuilder.hh File Reference

```
#include <iostream>
#include <string>
#include <map>
#include "HepPDT/ParticleDataTable.hh"
#include "HepPDT/TempParticleData.hh"
#include "HepPDT/DefTable.hh"
#include "HepPDT/stringtodouble.hh"
#include "HepPDT/TableBuilder.icc"
```

Namespaces

- namespace **HepPDT**
- namespace **HepPDT::detail**

Classes

- class **HepPDT::TableBuilder**

*The **TableBuilder** (p. 116) class is used to construct a **ParticleDataTable** (p. 85).*

Functions

- bool **HepPDT::addPDGParticles** (std::istream &pdfile, TableBuilder &tb)
read PDG input and add particles to the table
- bool **HepPDT::addPythiaParticles** (std::istream &pdfile, TableBuilder &tb)
read Pythia input and add particles to the table
- bool **HepPDT::addHerwigParticles** (std::istream &pdfile, TableBuilder &tb)
- bool **HepPDT::addIsajetParticles** (std::istream &pdfile, TableBuilder &tb)
read Isajet particle input and add particles to the table
- bool **HepPDT::addIsajetDecay** (std::istream &pdfile, TableBuilder &tb)
read Isajet decay input and add decay information to the table
- bool **HepPDT::addQQParticles** (std::istream &pdfile, TableBuilder &tb)
read QQ input and add particles to the table
- bool **HepPDT::addEvtGenParticles** (std::istream &pdfile, TableBuilder &tb)
read EvtGen input and add particles to the table
- bool **HepPDT::addParticleTable** (std::istream &pdfile, TableBuilder &tb)
read particle.tbl (or something similar) and add particles to the table

- void **HepPDT::detail::getPDGpid** (std::vector< int > &idlist, std::string &pdline)
for internal use
- void **HepPDT::detail::getPDGnames** (std::vector< std::string > &namelst, std::string &pdline)
for internal use
- void **HepPDT::detail::parsePDGLine** (TempParticleData &tpd, std::string &pdline)
for internal use
- bool **HepPDT::detail::CheckPDGEntry** (TempParticleData &tpd, const std::string &, double, double)
for internal use
- bool **HepPDT::detail::getPythiaid** (int &id, const std::string &pdline)
for internal use
- void **HepPDT::detail::parsePythiaLine** (TempParticleData &tpd, int &anti, std::string &aname, const std::string &pdline)
for internal use
- void **HepPDT::detail::parsePythiaDecayLine** (TempParticleData &tpd, const std::string &pdline)
for internal use
- TempDecayData **HepPDT::detail::getPythiaDecay** (const std::string &pdline)
for internal use
- bool **HepPDT::detail::getIsajetID** (int &id, const std::string &pdline)
for internal use
- void **HepPDT::detail::parseIsajetLine** (TempParticleData &tpd, const std::string &pdline)
for internal use
- void **HepPDT::detail::parseIsajetDecayLine** (TempParticleData &tpd, const std::string &pdline, TableBuilder &tb)
for internal use
- bool **HepPDT::detail::getParticleID** (int &id, const std::string &pdline)
for internal use
- void **HepPDT::detail::parseParticleLine** (TempParticleData &tpd, const std::string &pdline)
for internal use

9.65 TableBuilder.icc File Reference

```
#include <sstream>
```

Namespaces

- namespace **HepPDT**

9.66 TempParticleData.cc File Reference

```
#include <algorithm>
#include "HepPDT/defs.h"
#include "HepPDT/TempParticleData.hh"
```

Namespaces

- namespace **HepPDT**

9.67 TempParticleData.hh File Reference

```
#include <string>
#include <vector>
#include "HepPDT/SpinState.hh"
#include "HepPDT/ParticleID.hh"
#include "HepPDT/Measurement.hh"
```

Namespaces

- namespace **HepPDT**

Classes

- struct **HepPDT::TempConstituent**
Temporary constituent (e.g., quark) information.
- struct **HepPDT::TempDecayData**
temporary holder for decay data
- struct **HepPDT::TempAliasData**
Hold Alias information from EvtGen.
- struct **HepPDT::TempParticleData**
temporary holder for Particle Data information

Typedefs

- typedef std::vector<TempDecayData> **HepPDT::TDDlist**
useful typedef

Functions

- double **HepPDT::calculateWidthFromLifetime** (double)
- void **HepPDT::swap** (TempParticleData &first, TempParticleData &second)

9.68 testHepPDT.cc File Reference

```
#include <fstream>
#include "HepPDT/defs.h"
#include "HepPDT/TableBuilder.hh"
#include "HepPDT/ParticleDataTable.hh"
#include "HepPDT/HeavyIonUnknownID.hh"
```

Functions

- int **main ()**

9.68.1 Function Documentation

9.68.1.1 int main ()

Definition at line 23 of file testHepPDT.cc.

References HepPDT::addPDGParticles(), HepPDT::ParticleData::lowerCutoff(), HepPDT::ParticleDataTable::particle(), HepPDT::ParticleData::totalWidth(), HepPDT::ParticleData::upperCutoff(), HepPDT::ParticleData::write(), HepPDT::ParticleDataTable::writeParticleData(), and HepPDT::ParticleDataTable::writeParticleInfo().

9.69 testParticleIDMethods.cc File Reference

```
#include <iostream>
#include <iomanip>
#include "HepPID/ParticleIDMethods.hh"
```

Functions

- int **main ()**

9.69.1 Function Documentation

9.69.1.1 int main ()

Definition at line 19 of file testParticleIDMethods.cc.

References HepPID::A(), HepPID::digit(), HepPID::extraBits(), HepPID::fundamentalID(), HepPID::isBaryon(), HepPID::isDiQuark(), HepPID::isHadron(), HepPID::isLepton(), HepPID::isMeson(), HepPID::isNucleus(), HepPID::isValid(), HepPID::jSpin(), HepPID::lambda(), HepPID::lSpin(), HepPID::n, HepPID::n10, HepPID::nj, HepPID::nl, HepPID::nq1, HepPID::nq2, HepPID::nq3, HepPID::nr, HepPID::sSpin(), HepPID::threeCharge(), and HepPID::Z().

9.70 testPID.cc File Reference

```
#include <iostream>
#include <iomanip>
#include "HepPDT/defs.h"
#include "HepPDT/ParticleID.hh"
```

Functions

- int **main ()**

9.70.1 Function Documentation

9.70.1.1 int main ()

Definition at line 20 of file testPID.cc.

References HepPDT::n, HepPDT::n10, HepPDT::nj, HepPDT::nl, HepPDT::Quarks::nq1, HepPDT::nq1, HepPDT::Quarks::nq2, HepPDT::nq2, HepPDT::Quarks::nq3, HepPDT::nq3, HepPDT::nr, HepPID::nr, and HepPDT::spinitod().

9.71 testReadEvtGen.cc.in File Reference

```
#include <fstream>
#include "HepPDT/defs.h"
#include "HepPDT/TableBuilder.hh"
#include "HepPDT/ParticleDataTable.hh"
```

Functions

- int **main** ()

9.71.1 Function Documentation

9.71.1.1 int main ()

Definition at line 15 of file testReadEvtGen.cc.in.

References HepPDT::addEvtGenParticles(), and HepPDT::ParticleDataTable::writeParticleData().

9.72 testReadIsajet.cc.in File Reference

```
#include <iostream>
#include "HepPDT/defs.h"
#include "HepPDT/TableBuilder.hh"
#include "HepPDT/ParticleDataTable.hh"
```

Functions

- int **main** ()

9.72.1 Function Documentation

9.72.1.1 int main ()

Definition at line 22 of file testReadIsajet.cc.in.

References HepPDT::addIsajetParticles(), HepPDT::ParticleDataTable::writeParticleData(), and HepPDT::ParticleDataTable::writeParticleInfo().

9.73 testReadParticleTable.cc.in File Reference

```
#include <fstream>
#include "HepPDT/defs.h"
#include "HepPDT/TableBuilder.hh"
#include "HepPDT/ParticleDataTable.hh"
```

Functions

- int **main** ()

9.73.1 Function Documentation

9.73.1.1 int main ()

Definition at line 15 of file testReadParticleTable.cc.in.

References HepPDT::addParticleTable(), HepPDT::ParticleDataTable::particle(), HepPDT::ParticleDataTable::write(), and HepPDT::ParticleDataTable::writeParticleData().

9.74 testReadQQ.cc.in File Reference

```
#include <fstream>
#include "HepPDT/defs.h"
#include "HepPDT/TableBuilder.hh"
#include "HepPDT/ParticleDataTable.hh"
```

Functions

- int **main** ()

9.74.1 Function Documentation

9.74.1.1 int main ()

Definition at line 15 of file testReadQQ.cc.in.

References HepPDT::addQQParticles(), HepPDT::ParticleDataTable::writeParticleData(), and HepPDT::ParticleDataTable::writeParticleTranslation().

9.75 translateEvtGen.cc File Reference

```
#include <map>
#include <utility>
#include "HepPID/Version.hh"
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPID/ParticleIDMethods.hh"
#include "HepPID/ParticleName.hh"
```

Namespaces

- namespace **HepPID**

Typedefs

- typedef std::map< int, int > **HepPID::EvtGenPDTMap**
- typedef std::map< int, int > **HepPID::PDTEvtGenMap**

Functions

- EvtGenPDTMap const & **HepPID::getEvtGenPDTMap** ()
- PDTEvtGenMap const & **HepPID::getPDTEvtGenMap** ()
- EvtGenPDTMap const & **HepPID::EvtGenPDTMapInit** ()
- PDTEvtGenMap const & **HepPID::PDTEvtGenMapInit** ()
- EvtGenPDTMap const & **HepPID::getEvtGenPDTMap** ()
- PDTEvtGenMap const & **HepPID::getPDTEvtGenMap** ()
- int **HepPID::translateEvtGentoPDT** (const int evtGenID)
translate EvtGen to PDG standard
- int **HepPID::translatePDTtoEvtGen** (const int pid)
translate PDG standard to EvtGen
- void **HepPID::writeEvtGenTranslationLine** (int i, std::ostream &os)
- void **HepPID::writeEvtGenTranslation** (std::ostream &os)
output the translation list

9.76 translateGeanttoPDT.cc File Reference

```
#include <iostream>
#include "HepPID/Version.hh"
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPID/ParticleIDMethods.hh"
```

Namespaces

- namespace **HepPID**

Defines

- #define **IDMAX** 49

Functions

- int **HepPID::translateGeanttoPDT** (const int geantID)
translate Geant3 to PDG standard

9.76.1 Define Documentation

9.76.1.1 #define IDMAX 49

Definition at line 16 of file translateGeanttoPDT.cc.

Referenced by HepPID::translateGeanttoPDT(), and HepPID::translatePDTtoGeant().

9.77 translateHerwig.cc File Reference

```
#include <map>
#include <utility>
#include "HepPID/Version.hh"
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPID/ParticleIDMethods.hh"
#include "HepPID/ParticleName.hh"
```

Namespaces

- namespace **HepPID**

Typedefs

- typedef std::map< int, int > **HepPID::HerwigPDTMap**
- typedef std::map< int, int > **HepPID::PDTHerwigMap**

Functions

- HerwigPDTMap const & **HepPID::getHerwigPDTMap** ()
- PDTHerwigMap const & **HepPID::getPDTHerwigMap** ()
- HerwigPDTMap const & **HepPID::HerwigPDTMapInit** ()
- PDTHerwigMap const & **HepPID::PDTHerwigMapInit** ()
- HerwigPDTMap const & **HepPID::getHerwigPDTMap** ()
- PDTHerwigMap const & **HepPID::getPDTHerwigMap** ()
- int **HepPID::translateHerwigtoPDT** (const int herwigID)

translate Herwig to PDG standard
- int **HepPID::translatePDTtoHerwig** (const int pid)

translate PDG standard to Herwig
- void **HepPID::writeHerwigTranslationLine** (int i, std::ostream &os)
- void **HepPID::writeHerwigTranslation** (std::ostream &os)

output the translation list

9.78 translateIsajet.cc File Reference

```
#include <map>
#include <utility>
#include "HepPID/Version.hh"
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPID/ParticleIDMethods.hh"
#include "HepPID/ParticleName.hh"
```

Namespaces

- namespace **HepPID**

Typedefs

- typedef std::map< int, int > **HepPID::IsajetPDTMap**
- typedef std::map< int, int > **HepPID::PDTIsajetMap**

Functions

- IsajetPDTMap const & **HepPID::getIsajetPDTMap** ()
- PDTIsajetMap const & **HepPID::getPDTIsajetMap** ()
- IsajetPDTMap const & **HepPID::IsajetPDTMapInit** ()
- PDTIsajetMap const & **HepPID::PDTIsajetMapInit** ()
- int **HepPID::convIsajettoPDT** (const int id)

translate Isajet to PDG standard
- int **HepPID::convPDTtoIsajet** (const int id)
- IsajetPDTMap const & **HepPID::getIsajetPDTMap** ()
- PDTIsajetMap const & **HepPID::getPDTIsajetMap** ()
- int **HepPID::translateIsajettoPDT** (const int isajetID)

translate Isajet to PDG standard
- int **HepPID::translatePDTtoIsajet** (const int pid)

translate PDG standard to Isajet
- void **HepPID::writeIsajetTranslationLine** (int i, std::ostream &os)
- void **HepPID::writeIsajetTranslation** (std::ostream &os)

output the translation list

9.79 translatePDG.cc File Reference

```
#include <map>
#include <utility>
#include "HepPID/Version.hh"
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPID/ParticleIDMethods.hh"
#include "HepPID/ParticleName.hh"
```

Namespaces

- namespace **HepPID**

Typedefs

- typedef std::map< int, int > **HepPID::PDGtoPDTMap**
- typedef std::map< int, int > **HepPID::PDTtoPDGMap**

Functions

- PDGtoPDTMap const & **HepPID::getPDGtoPDTMap()**
- PDTtoPDGMap const & **HepPID::getPDTtoPDGMap()**
- PDGtoPDTMap const & **HepPID::PDGtoPDTMapInit()**
- PDTtoPDGMap const & **HepPID::PDTtoPDGMapInit()**
- PDGtoPDTMap const & **HepPID::getPDGtoPDTMap()**
- PDTtoPDGMap const & **HepPID::getPDTtoPDGMap()**
- int **HepPID::translatePDGtabletoPDT** (const int pdgID)

translate PDG table to PDG standard
- int **HepPID::translatePDTtoPDGtable** (const int pid)

translate PDG standard to PDG table
- void **HepPID::writePDGTranslationLine** (int i, std::ostream &os)
- void **HepPID::writePDGTranslation** (std::ostream &os)

output the translation list

9.80 translatePDTtoGeant.cc File Reference

```
#include <iostream>
#include "HepPID/Version.hh"
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPID/ParticleIDMethods.hh"
```

Namespaces

- namespace **HepPID**

Defines

- #define **IDMAX** 49

Functions

- int **HepPID::translatePDTtoGeant** (const int pid)
translate PDG standard to Geant3

9.80.1 Define Documentation

9.80.1.1 #define IDMAX 49

Definition at line 18 of file translatePDTtoGeant.cc.

9.81 translatePythia.cc File Reference

```
#include <map>
#include <utility>
#include "HepPID/Version.hh"
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPID/ParticleIDMethods.hh"
#include "HepPID/ParticleName.hh"
```

Namespaces

- namespace **HepPID**

Typedefs

- typedef std::map< int, int > **HepPID::PythiaPDTMap**
- typedef std::map< int, int > **HepPID::PDTPythiaMap**

Functions

- PythiaPDTMap const & **HepPID::getPythiaPDTMap** ()
- PDTPythiaMap const & **HepPID::getPDTPythiaMap** ()
- PythiaPDTMap const & **HepPID::PythiaPDTMapInit** ()
- PDTPythiaMap const & **HepPID::PDTPythiaMapInit** ()
- PythiaPDTMap const & **HepPID::getPythiaPDTMap** ()
- PDTPythiaMap const & **HepPID::getPDTPythiaMap** ()
- int **HepPID::translatePythiaToPDT** (const int pythiaID)

translate Pythia to PDG standard
- int **HepPID::translatePDTToPythia** (const int pid)

translate PDG standard to Pythia
- void **HepPID::writePythiaTranslationLine** (int i, std::ostream &os)
- void **HepPID::writePythiaTranslation** (std::ostream &os)

output the translation list

9.82 translateQQ.cc File Reference

```
#include <map>
#include <utility>
#include "HepPID/Version.hh"
#include "HepPID/ParticleIDTranslations.hh"
#include "HepPID/ParticleIDMethods.hh"
#include "HepPID/ParticleName.hh"
```

Namespaces

- namespace **HepPID**

Typedefs

- typedef std::map< int, int > **HepPID::QQPDTMap**
- typedef std::map< int, int > **HepPID::PDTQQMap**
- typedef std::map< int, int > **HepPID::QQbarMap**
- typedef std::map< int, int > **HepPID::InverseQQbarMap**

Functions

- QQPDTMap const & **HepPID::getQQPDTMap** ()
- PDTQQMap const & **HepPID::getPDTQQMap** ()
- QQbarMap const & **HepPID::getQQbarMap** ()
- InverseQQbarMap const & **HepPID::getInverseQQbarMap** ()
- QQPDTMap const & **HepPID::QQPDTMapInit** ()
- QQbarMap const & **HepPID::QQbarMapInit** ()
- PDTQQMap const & **HepPID::PDTQQMapInit** ()
- InverseQQbarMap const & **HepPID::InverseQQbarMapInit** ()
- QQPDTMap const & **HepPID::getQQPDTMap** ()
- PDTQQMap const & **HepPID::getPDTQQMap** ()
- QQbarMap const & **HepPID::getQQbarMap** ()
- InverseQQbarMap const & **HepPID::getInverseQQbarMap** ()
- int **HepPID::translateQQbar** (const int id)
QQ helper function.
- int **HepPID::translateInverseQQbar** (const int id)
QQ helper function.
- int **HepPID::translateQQtoPDT** (const int qqID)
translate QQ to PDG standard
- int **HepPID::translatePDTtoQQ** (const int pid)
translate PDG standard to QQ
- void **HepPID::writeQQTranslation** (std::ostream &os)

output the translation list

9.83 Version.cc File Reference

```
#include "HepPDT/defs.h"
#include "HepPDT/Version.hh"
```

Namespaces

- namespace **HepPDT**

Functions

- std::string **HepPDT::versionName ()**
return HepPDT (p. 23) version
- void **HepPDT::version ()**
print HepPDT (p. 23) version
- void **HepPDT::writeVersion (std::ostream &os)**
write HepPDT (p. 23) version to os

9.84 Version.cc File Reference

```
#include "HepPID/Version.hh"
```

Namespaces

- namespace **HepPID**

Functions

- std::string **HepPID::versionName** ()
return HepPID (p. 37) version
- void **HepPID::version** ()
print HepPID (p. 37) version
- void **HepPID::writeVersion** (std::ostream &os)
write HepPID (p. 37) version to os

9.85 Version.hh File Reference

```
#include <string>
#include <iostream>
```

Namespaces

- namespace **HepPDT**

Functions

- void **HepPDT::version ()**
print HepPDT (p. 23) version
- void **HepPDT::writeVersion (std::ostream &os)**
write HepPDT (p. 23) version to os
- std::string **HepPDT::versionName ()**
return HepPDT (p. 23) version

9.86 Version.hh File Reference

```
#include <string>
#include <iostream>
```

Namespaces

- namespace **HepPID**

Functions

- void **HepPID::version ()**
print HepPID (p. 37) version
- void **HepPID::writeVersion (std::ostream &os)**
write HepPID (p. 37) version to os
- std::string **HepPID::versionName ()**
return HepPID (p. 37) version

9.87 write.cc File Reference

```
#include <iomanip>
#include <sstream>
#include "HepPDT/defs.h"
#include "HepPDT/ParticleData.hh"
```

Namespaces

- namespace **HepPDT**
- namespace **std**

Chapter 10

HepPDT Example Documentation

10.1 examListHerwig.cc

list Herwig particle ID translations

```
1 // -----
2 // examListHerwig.cc
3 //
4 // -----
13
14 #include <fstream>
15 #include <iostream>
16
17 #include "HepPID/ParticleIDTranslations.hh"
18 #include "HepPID/ParticleName.hh"
19 #include "HepPID/Version.hh"
20
21 extern "C" {
22 // these functions are defined in examListHerwigInterface.F
23     void list_herwig_init_ ( int * nevt );
24     void list_herwig_end_ ( );
25     void get_list_size_ ( int * );
26     void get_herwig_name_( int * ihwg, int * id, char *name );
27 }
28
29 int main()
30 {
31     int nevt=20;
32     int i, j, iend, isize;
33     int hid, id;
34     char cname[10];
35     std::string hname;
36     std::string pn;
37     static char outfile[] = "examListHerwig.out";
38     std::string title = "HepPID listing of Herwig translations";
39
40     // initialize herwig
41     list_herwig_init_ ( & nevt );
42
43     // open the output stream
44     std::ofstream os( outfile );
45     if( !os ) {
46         std::cout << "error opening output file" << std::endl;
47         exit(1);
48     }
49     HepPID::writeVersion(os);
```

```
50     get_list_size_( & isize );
51     os << "      " << title << std::endl;
52     os << " number of Herwig particles: " << isize << std::endl;
53
54     for( i=1, iend=isize+1; i<iend; ++i ) {
55         // get info from herwig
56         for( j=0; j<10; ++j ) { cname[j] = '\0'; }
57         get_herwig_name_( & i, & hid, cname );
58         hname = std::string( cname );
59         id = HepPID::translateHerwigtoPDT( hid );
60         pn = HepPID::particleName( id );
61         os << "Herwig: ";
62         os.width(7);
63         os << i ;
64         os.width(12);
65         os << hid << " " << hname;
66         os << " HepPID: " ;
67         os.width(12);
68         os << id << " " << pn << std::endl;
69     }
70 }
71 list_herwig_end_();
72
73 return 0;
74 }
75 }
76
```

10.2 examListHerwigInterface.F

interface to some Herwig Fortran routines

```

1      subroutine list_herwig_init(nevt)
2
3 c
4 c  initialization for the herwig C++ listing
5 c
6 #include "herwig65.inc"
7     integer lnhwrt,lnhrd,lnhout,lnhdcy
8     common/heplun/lnhwrt,lnhrd,lnhout,lnhdcy
9
10    external hwudat
11    integer n
12    integer istr,nevt
13 C
14 C initialize HEP logical units
15    lnhwrt=0
16    lnhrd=0
17    lnhdcy=0
18    lnhout=22
19    lhwout=lnhout
20 C      open(unit=lnhout,file='examHerwigToStdHep.lpt',status='new')
21 C
22 c      call hptrlsth
23 C
24    return
25    end
26
27    subroutine get_list_size( isize )
28 c return the maximum size of herwig's particle list
29 #include "herwig65.inc"
30    integer isize
31    isize = NRES
32    return
33    end
34
35    subroutine get_herwig_name( ihwg, id, name )
36 c ihwg is the index into herwig's short list
37 #include "herwig65.inc"
38    integer id, ihwg
39    character*8 name
40    id = 0
41    call HWUIDT(2,id,ihwg,name)
42    return
43    end
44
45    subroutine list_herwig_end
46    integer lnhwrt,lnhrd,lnhout,lnhdcy
47    common/heplun/lnhwrt,lnhrd,lnhout,lnhdcy
48 C---terminate elementary process
49 c      call hwefin
50 C      close(unit=lnhout)
51    return
52    end
53
54 C-----
55    subroutine hwabeg
56 C... user's routine for initialization
57    end
58    subroutine hwaend
59 C... user's routine for terminal calculations, histogram output, etc
60    end
61    subroutine hwanal
62 C... user's routine to analyse data from event

```

```
63      end
64 C-----
```

10.3 examListIsajet.cc

List the Isajet particle ID translations

```

1 // -----
2 // examListIsajet.cc
3 //
4 // -----
13
14 #include <fstream>
15 #include <iostream>
16
17 #include "HepPID/ParticleIDTranslations.hh"
18 #include "HepPID/ParticleName.hh"
19 #include "HepPID/Version.hh"
20
21 extern "C" {
22     void list_isajet_init_ ( );
23     void flavor_( int *, int *, int *, int *, int *, int * );
24     void get_label_( int * id, char *name );
25 }
26
27 int main()
28 {
29     static char outfile[] = "examListIsajet.out";
30     std::string title = "HepPID listing of Isajet translations";
31
32     // initialize isajet
33     list_isajet_init_ ();
34
35     // open the output stream
36     std::ofstream os( outfile );
37     if( !os ) {
38         std::cout << "error opening output file" << std::endl;
39         exit(1);
40     }
41     HepPID::writeVersion(os);
42
43     os << "      " << title << std::endl;
44
45     int i, j;
46     int id, aid, f11, f12, f13, js, indx;
47     int pid;
48     char cname[10];
49     char acname[10];
50     std::string hname;
51     std::string pn;
52     for( i=1; i<100005; ++i ) {
53         // make sure names are empty
54         for( j=0; j<10; ++j) { cname[j] = '\0'; }
55         for( j=0; j<10; ++j) { acname[j] = '\0'; }
56         // get info from isajet
57         id = i;
58         aid = 0;
59         flavor_(&id, &f11, &f12, &f13, &js, &indx );
60         // we need both a valid index and a valid label
61         // check the label only if there is a valid translation
62         if ( indx > 0 ) {
63             get_label_(&id,cname);
64             aid = -id;
65             get_label_(&aid,acname);
66         } else {
67             id = aid = 0;
68         }
69     }
70     // print particle

```

```
71      if( id != 0 ) {
72          pid = HepPID::translateIsajettoPDT( id );
73          hname = std::string( cname );
74          if ( pid != 0 ) {
75              pn = HepPID::particleName( pid );
76              os << "Isajet: ";
77              os.width(10);
78              os << id << " " << hname;
79              os << " HepPID: ";
80              os.width(12);
81              os << pid << " " << pn << std::endl;
82          } else if ( strncmp( cname, "ERR", 3 ) != 0 ) {
83              os << "Isajet: ";
84              os.width(10);
85              os << id << " with name \\" " << hname;
86              os << "\\" has no HepPID translation " << std::endl;
87          }
88      }
89 // print antiparticle
90 if( aid != 0 ) {
91     hname = std::string( acname );
92     pid = HepPID::translateIsajettoPDT( aid );
93     if ( pid != 0 ) {
94         pn = HepPID::particleName( pid );
95         os << "Isajet: ";
96         os.width(10);
97         os << aid << " " << hname;
98         os << " HepPID: ";
99         os.width(12);
100        os << pid << " " << pn << std::endl;
101    } else if ( strncmp( acname, "ERR", 3 ) != 0 ) {
102        os << "Isajet: ";
103        os.width(10);
104        os << aid << " with name \\" " << hname;
105        os << "\\" has no HepPID translation " << std::endl;
106    }
107 }
108 }
109
110 return 0;
111 }
```

10.4 examListIsajetInterface.F

interface to some Isajet Fortran routines

```

1      subroutine list_isajet_init
2 C
3 C      JTDKY = +/- unit number for decay table file.
4 C          If it is negative, decay table is not printed.
5 C      JTEVT = +/- unit number for output event file.
6 C          If it is negative, only stable particles are written on it.
7 C      JTCOM =     unit number for command file.
8 C      JTLIS =     unit number for listing.
9 C
10     IMPLICIT NONE
11
12 C
13     integer istr,nevt,itotal
14     INTEGER JTDKY,JTEVT,JTCOM,JTLIS,IFL,ILOOP,IPRT,LOK,ILOOP2
15     INTEGER INDEC,INDEC2
16     CHARACTER*132 ISADEC
17     LOGICAL OK,DONE
18     SAVE ILOOP,JTDKY,JTEVT,JTCOM,JTLIS
19
20     EXTERNAL ALDATA
21 C
22 C          Initialize ISAJET
23 C
24     JTDKY=-1
25     JTEVT=23
26     JTCOM=21
27     JTLIS=22
28     nevt=1000
29     itotal=0
30 C  point to standard decay tables
31     CALL GETENV( 'ISAJET_DIR', ISADEC )
32     IF ( ISADEC .EQ. ' ' ) THEN
33         ISADEC = 'isadecay.dat'
34     ELSE
35         INDEC   = INDEX ( ISADEC , ' ' )
36         ISADEC(INDEC:INDEC+13) = '/isadecay.dat'
37         INDEC2  = INDEX ( ISADEC , ' ' ) - 1
38 C         print *,'looking for ',ISADEC(:INDEC2)
39     ENDIF
40     OPEN(UNIT=1,FILE=ISADEC,STATUS='OLD')
41     OPEN(UNIT=JTLIS,FILE='examListIsajet.lpt',STATUS='NEW')
42     CALL ISAINI(JTDKY,JTEVT,JTCOM,JTLIS)
43     IPRT=1
44 C
45 C print list of defined particles and their translations
46 C      (you have to call ISAINI first)
47 C use bogus masses for 4th generation quarks so we can see their definitions
48 C
49     CALL PRTLST(JTLIS,200.,220.)
50 C
51
52     return
53 END
54
55     subroutine get_label( id, lb)
56
57     integer id
58     character*8 LB,LABEL
59
60     lb = ''
61
62     if ( id.ne.0) then

```

```
63      lb = LABEL(id)
64      endif
65
66      return
67      end
```

10.5 examListPythia.cc

list Pythia particle ID translations

```

1 // -----
2 // examListPythia.cc
3 //
4 // -----
13
14 #include <fstream>
15 #include <string>
16
17 #include "HepPID/ParticleIDTranslations.hh"
18 #include "HepPID/ParticleIDMethods.hh"
19 #include "HepPID/ParticleName.hh"
20 #include "HepPID/Version.hh"
21
22 extern "C" {
23     void list_pythia_( );
24     void getkf_( int *, int * );
25     void getpyname_( int * , int *, char *name );
26 }
27
28 void writeLine( int & i, int & kf, int & id,
29                 std::string & name, std::string & pn, std::ofstream & os );
30
31 int main()
32 {
33     int kf,akf,pid,apid,iok;
34     char cname[17],caname[17];
35     std::string name, aname, pn, apn;
36     const char outfile[] = "examListPythia.out";
37     std::string title = "HepPID listing of Pythia translations";
38     // open the output file
39     std::ofstream os( outfile );
40     if( !os ) {
41         std::cerr << "cannot open " << outfile << std::endl;
42         exit(-1);
43     }
44     HepPID::writeVersion(os);
45     // get Pythia listing using the fortran pylist function
46     // write the output of pylist to examListPythia.lpt
47     list_pythia_();
48
49     os << "      " << title << std::endl;
50
51     for(int i=1; i<501; ++i) {
52         getkf_( &i, &kf );
53         if( kf != 0 ) {
54             getpyname_( &kf, &iok, cname );
55             if( iok == 1 ) {
56                 cname[16]='\0';
57                 name = std::string( cname );
58                 pid = HepPID::translatePythiaToPDT( kf );
59                 pn = HepPID::particleName( pid );
60                 writeLine( i, kf, pid, name, pn, os );
61                 if( HepPID::isValid( -pid ) ) {
62                     akf=-kf;
63                     getpyname_( &akf, &iok, caname );
64                     if( iok == 1 ) {
65                         apid = HepPID::translatePythiaToPDT( akf );
66                         apn = HepPID::particleName( apid );
67                         caname[16]='\0';
68                         aname = std::string( caname );
69                         writeLine( i, akf, apid, aname, apn, os );
70                 }
71             }
72         }
73     }
74 }
```

```
71             }
72         }
73     }
74 }
75 return 0;
76 }
77
78 void writeLine( int & i, int & kf, int & id,
79                 std::string & name, std::string & pn, std::ofstream & os )
80 {
81     os << "Pythia: ";
82     os.width(7);
83     os << i ;
84     os.width(12);
85     os << kf << " " << name;
86     os << " HepPID: " ;
87     os.width(12);
88     os << id << " " << pn << std::endl;
89 }
```

10.6 examListPythiaInterface.F

interface to some Pythia Fortran routines

```

1      subroutine list_pythia
2 C
3 C      list jetset particle definitions
4 C
5 #include "pydat1.inc"
6
7      INTEGER LNHOUT
8 C
9      lnhout=22
10     MSTU(11)=LNHOUT
11     OPEN(UNIT=LNHOUT,FILE='examListPythia.lpt',STATUS='NEW')
12 C
13 C...
14     call pylist(11)
15 C...enable pylist(12) if you need a new pythia decay table
16 C     call pylist(12)
17 C
18 C
19 C...close output file
20     CLOSE(UNIT=LNHOUT)
21     return
22     END
23
24     subroutine getkf(kc,kf)
25 #include "pydat2.inc"
26     integer kf,kc
27     kf = KCHG(KC,4)
28     return
29     END
30
31     subroutine getpyname(kf,idef,chap)
32 #include "pydat2.inc"
33     integer kf
34     character*16 chap
35     integer idef
36     idef = 1
37     CALL PYNAME(KF,CHAP)
38     if(CHAP.EQ.' ') idef=0
39     return
40     END
41
42

```

10.7 examMyPDT.cc

create a custom PDT using our own definitions and write out the resulting PDT

```

1 // -----
2 // examMyPDT.cc
3 //
9
10 #include "HepPDT/defs.h"
11 #include <fstream>
12
13 #include <string>
14
15 #include "HepPDT/TableBuilder.hh"
16 #include "HepPDT/ParticleDataTable.hh"
17 #include "HepPDT/TempParticleData.hh"
18
19 void addData( HepPDT::TableBuilder& tb, std::string const & name, int const id,
20               double const mass, double const charge, double const width,
21               double const tspin );
22
23 int main()
24 {
25     const char outfile[] = "examMyPDT.out";
26     // construct empty PDT
27     HepPDT::ParticleDataTable datacol;
28     {
29         // Construct table builder
30         HepPDT::TableBuilder tb(datacol);
31         // create my own particles here
32         addData( tb, "p+", 2212, 0.938, +1.0, -1, .5 );
33         addData( tb, "d", 1, 0., -2./3, -1, .5 );
34         addData( tb, "u~", -2, 0., -1./3, -1, .5 );
35         addData( tb, "W-", -24, 80.396, -1.0, 2.06, 1.0 );
36         addData( tb, "gamma", 22, 0., 0., -1, 1.0 );
37         addData( tb, "badgamma", 122, 0., 0., -1, 1.0 );
38         tb.removeParticle( 122 );
39     } // the tb destructor fills datacol
40     std::ofstream wpdfile( outfile );
41     if( !wpdfile ) {
42         std::cerr << "cannot open " << outfile << std::endl;
43         exit(-1);
44     }
45     datacol.writeParticleData(wpdfile);
46     // access a particle
47     // you get a null pointer if you request an undefined particle
48     HepPDT::ParticleData * pd = datacol.particle( HepPDT::ParticleID(22) );
49     if( pd ) {
50         std::cout << "particle " << pd->name() << " is defined" << std::endl;
51     } else {
52         std::cout << "ERROR: particle is not in particle data table" << std::endl;
53     }
54     pd = datacol[ HepPDT::ParticleID(-24) ];
55     // we expect this next line to produce an error
56     std::cout << "the error is expected" << std::endl;
57     if( datacol[ HepPDT::ParticleID(111) ] ) {
58         std::cout << "particle " << datacol[ HepPDT::ParticleID(111) ]->name() << " is defined" << std::endl;
59     } else {
60         std::cout << "ERROR: particle " << HepPDT::ParticleID(111).pid()
61             << " is not in particle data table" << std::endl;
62     }
63
64     return 0;
65 }
66
67 void addData( HepPDT::TableBuilder& tb, std::string const & name, int const id,
```

```
68         double const mass, double const charge, double const width,
69         double const tspin )
70 {
71     HepPDT::TempParticleData& tpd = tb.getParticleData( HepPDT::ParticleID( id ) );
72     tpd.tempParticleName = name;
73     tpd.tempCharge = charge;
74     tpd.tempMass = HepPDT::Measurement( mass, 0. );
75     tpd.tempSpin = HepPDT::SpinState( tspin, 0., 0. );
76     tpd.tempWidth = HepPDT::Measurement( width, 0. );
77     tb.addParticle( tpd );
78 }
```

10.8 listEvtGenNames.cc.in

The **HepPDT** (p. 23) tests are also useful examples

read EvtGen table and write out translation from EvtGen to **HepPDT** (p. 23)

```

1 // -----
2 // listEvtGenNames.cc
3 // Author: Lynn Garren
4 //
5 // read EvtGen table and write out translation from EvtGen to HepPDT
6 //
7 // -----
8
9 #include <fstream>
10 #include <iostream>
11
12 #include "HepPDT/TableBuilder.hh"
13 #include "HepPDT/ParticleDataTable.hh"
14
15 int main()
16 {
17     const char infile1[] = "@top_srcdir@/data/pdt.table";
18     const char infile2[] = "@top_srcdir@/data/DECAY.DEC";
19     const char outfile[] = "listEvtGenNames.out";
20     // open input files
21     std::ifstream pdfile1( infile1 );
22     if( !pdfile1 ) {
23         std::cerr << "cannot open " << infile1 << std::endl;
24         exit(-1);
25     }
26     std::ifstream pdfile2( infile2 );
27     if( !pdfile2 ) {
28         std::cerr << "cannot open " << infile2 << std::endl;
29         exit(-1);
30     }
31     // construct PDT
32     HepPDT::ParticleDataTable datacol( "EvtGen Table" );
33     {
34         // Construct table builder
35         HepPDT::TableBuilder tb(datacol);
36         // read the input - put as many here as you want
37         if( !addEvtGenParticles( pdfile1, tb ) ) { std::cout << "error reading EvtGen pdt file " << std::endl;
38         if( !addEvtGenParticles( pdfile2, tb ) ) { std::cout << "error reading EvtGen decay file " << std::endl;
39     } // the tb destructor fills datacol
40     // open output file
41     std::ofstream wpdfile( outfile );
42     if( !wpdfile ) {
43         std::cerr << "cannot open " << outfile << std::endl;
44         exit(-1);
45     }
46     // write a translation list
47     datacol.writeParticleTranslation( wpdfile );
48
49     return 0;
50 }
```

10.9 listEvtGenTranslation.cc

The **HepPDT** (p. 23) tests are also useful examples
write the **HepPID** (p. 37) EvtGen translations

```
1 // -----
2 // listEvtGenTranslation.cc
3 // Author: Lynn Garren
4 //
5 // Usage: listEvtGenTranslation
6 //
7 // -----
8
9 #include <fstream>
10 #include <iostream>
11
12 #include "HepPID/ParticleIDTranslations.hh"
13
14 int main()
15 {
16     const char outfile[] = "listEvtGenTranslation.out";
17     // open the output file
18     std::ofstream wpdfile( outfile );
19     if( !wpdfile ) {
20         std::cerr << "cannot open " << outfile << std::endl;
21         exit(-1);
22     }
23     // write the particle names
24     HepPID::writeEvtGenTranslation( wpdfile );
25 }
```

10.10 listHerwigTranslation.cc

The **HepPDT** (p. 23) tests are also useful examples
write the **HepPID** (p. 37) Herwig translations

```
1 // -----
2 // listHerwigTranslation.cc
3 // Author: Lynn Garren
4 //
5 // Usage: listHerwigTranslation
6 //
7 // -----
8
9 #include <fstream>
10 #include <iostream>
11
12 #include "HepPID/ParticleIDTranslations.hh"
13
14 int main()
15 {
16     const char outfile[] = "listHerwigTranslation.out";
17     // open the output file
18     std::ofstream wpdfile( outfile );
19     if( !wpdfile ) {
20         std::cerr << "cannot open " << outfile << std::endl;
21         exit(-1);
22     }
23     // write the particle names
24     HepPID::writeHerwigTranslation( wpdfile );
25 }
```

10.11 listIsajetTranslation.cc

The **HepPDT** (p. 23) tests are also useful examples

write the **HepPID** (p. 37) Isajet translations

```
1 // -----
2 // listIsajetTranslation.cc
3 // Author: Lynn Garren
4 //
5 // Usage: listIsajetTranslation
6 //
7 // -----
8
9 #include <fstream>
10 #include <iostream>
11
12 #include "HepPID/ParticleIDTranslations.hh"
13
14 int main()
15 {
16     const char outfile[] = "listIsajetTranslation.out";
17     // open the output file
18     std::ofstream wpdfile( outfile );
19     if( !wpdfile ) {
20         std::cerr << "cannot open " << outfile << std::endl;
21         exit(-1);
22     }
23     // write the particle names
24     HepPID::writeIsajetTranslation( wpdfile );
25 }
```

10.12 listParticleNames.cc

The **HepPDT** (p. 23) tests are also useful examples

list all known **HepPID** (p. 37) particle names

```
1 // -----
2 // listParticleNames.cc
3 // Author: Lynn Garren
4 //
5 // list all known HepPID particle names
6 //
7 // Usage: listParticleNames
8 //
9 // -----
10
11 #include <fstream>
12 #include <iostream>
13
14 #include "HepPID/ParticleName.hh"
15
16 int main()
17 {
18     const char outfile[] = "listParticleNames.out";
19     // open the output file
20     std::ofstream wpdfile( outfile );
21     if( !wpdfile ) {
22         std::cerr << "cannot open " << outfile << std::endl;
23         exit(-1);
24     }
25     // write the particle names
26     HepPID::listParticleNames( wpdfile );
27 }
```

10.13 listPDGNames.cc.in

The **HepPDT** (p. 23) tests are also useful examples
read PDG table and write out translation to **HepPDT** (p. 23)

```
1 // -----
2 // listPDGNames.cc
3 // Author: Lynn Garren
4 //
5 // read PDG table and write out translation to HepPDT
6 //
7 // -----
8
9 #include <fstream>
10 #include <iostream>
11
12 #include "HepPDT/TableBuilder.hh"
13 #include "HepPDT/ParticleDataTable.hh"
14
15 int main()
16 {
17     const char infile[] = "@top_srcdir@/data/mass_width_2006.mc";
18     const char outfile[] = "listPDGNames.out";
19     // open input file
20     std::ifstream pdfile( infile );
21     if( !pdfile ) {
22         std::cerr << "cannot open " << infile << std::endl;
23         exit(-1);
24     }
25     // construct empty PDT
26     HepPDT::ParticleDataTable datacol( "PDG Table" );
27     {
28         // Construct table builder
29         HepPDT::TableBuilder tb(datacol);
30         // read the input - put as many here as you want
31         if( !addPDGParticles( pdfile, tb ) )
32             { std::cout << "error reading PDG file " << std::endl; }
33     } // the tb destructor fills datacol
34     // open output file
35     std::ofstream wpdfile( outfile );
36     if( !wpdfile ) {
37         std::cerr << "cannot open " << outfile << std::endl;
38         exit(-1);
39     }
40     // write a translation list
41     datacol.writeParticleTranslation( wpdfile );
42
43     return 0;
44 }
```

10.14 listPDGTranslation.cc

The **HepPDT** (p. 23) tests are also useful examples
write the **HepPID** (p. 37) PDG translations

```
1 // -----
2 // listPDGTranslation.cc
3 // Author: Lynn Garren
4 //
5 // Usage: listPDGTranslation
6 //
7 // -----
8
9 #include <fstream>
10 #include <iostream>
11
12 #include "HepPID/ParticleIDTranslations.hh"
13
14 int main()
15 {
16     const char outfile[] = "listPDGTranslation.out";
17     // open the output file
18     std::ofstream wpdfile( outfile );
19     if( !wpdfile ) {
20         std::cerr << "cannot open " << outfile << std::endl;
21         exit(-1);
22     }
23     // write the particle names
24     HepPID::writePDGTranslation( wpdfile );
25 }
```

10.15 listPythiaNames.cc.in

The **HepPDT** (p. 23) tests are also useful examples

read Pythia table and write out translation from Pythia to **HepPDT** (p. 23)

```

1 // -----
2 // listPythiaNames.cc
3 // Author: Lynn Garren
4 //
5 // read Pythia table and write out translation from pythia to HepPDT
6 //
7 // -----
8
9 #include <fstream>
10 #include <iostream>
11
12 #include "HepPDT/TableBuilder.hh"
13 #include "HepPDT/ParticleDataTable.hh"
14
15 int main()
16 {
17     const char infile[] = "@srcdir@/listPythia.tbl";
18     const char outfile[] = "listPythiaNames.out";
19     // open input file
20     std::ifstream pdfile( infile );
21     if( !pdfile ) {
22         std::cerr << "cannot open " << infile << std::endl;
23         exit(-1);
24     }
25     // construct empty PDT
26     HepPDT::ParticleDataTable datacol( "Pythia Table" );
27     {
28         // Construct table builder
29         HepPDT::TableBuilder tb(datacol);
30         // read the input - put as many here as you want
31         if( !addPythiaParticles( pdfile, tb ) )
32             { std::cout << "error reading pythia file " << std::endl; }
33     } // the tb destructor fills datacol
34     // open output file
35     std::ofstream wpdfile( outfile );
36     if( !wpdfile ) {
37         std::cerr << "cannot open " << outfile << std::endl;
38         exit(-1);
39     }
40     // write a translation list
41     datacol.writeParticleTranslation( wpdfile );
42
43     return 0;
44 }
```

10.16 listPythiaTranslation.cc

The **HepPDT** (p. 23) tests are also useful examples

write the **HepPID** (p. 37) Pythia translations

```
1 // -----
2 // listPythiaTranslation.cc
3 // Author: Lynn Garren
4 //
5 // Usage: listPythiaTranslation
6 //
7 // -----
8
9 #include <fstream>
10 #include <iostream>
11
12 #include "HepPID/ParticleIDTranslations.hh"
13
14 int main()
15 {
16     const char outfile[] = "listPythiaTranslation.out";
17     // open the output file
18     std::ofstream wpdfile( outfile );
19     if( !wpdfile ) {
20         std::cerr << "cannot open " << outfile << std::endl;
21         exit(-1);
22     }
23     // write the particle names
24     HepPID::writePythiaTranslation( wpdfile );
25 }
```

10.17 listQQTranslation.cc

The **HepPDT** (p. 23) tests are also useful examples

write the **HepPID** (p. 37) QQ translations

```
1 // -----
2 // listQQTranslation.cc
3 // Author: Lynn Garren
4 //
5 // Usage: listQQTranslation
6 //
7 // -----
8
9 #include <fstream>
10 #include <iostream>
11
12 #include "HepPID/ParticleIDTranslations.hh"
13
14 int main()
15 {
16     const char outfile[] = "listQQTranslation.out";
17     // open the output file
18     std::ofstream wpdfile( outfile );
19     if( !wpdfile ) {
20         std::cerr << "cannot open " << outfile << std::endl;
21         exit(-1);
22     }
23     // write the particle names
24     HepPID::writeQQTranslation( wpdfile );
25 }
```

10.18 testHepPDT.cc

The **HepPDT** (p. 23) tests are also useful examples

Test by reading the PDG table. Get filename and location of PDG table from input stream.

```

1 // -----
2 // testHepPDT.cc
3 // Author: Lynn Garren
4 //
5 // test by reading the PDG table
6 // get filename and location of PDG table from input stream
7 //
8 // Usage: testHepPDT
9 //
10 // -----
11
12 #include <fstream>
13
14 #include "HepPDT/defs.h"
15 #include "HepPDT/TableBuilder.hh"
16 #include "HepPDT/ParticleDataTable.hh"
17 #include "HepPDT/HeavyIonUnknownID.hh"
18
19 using std::cout;
20 using std::cerr;
21 using std::endl;
22
23 int main()
24 {
25     char pdgfile[300] = "";
26     const char outfile[] = "testHepPDT.out";
27     std::cin >> pdgfile;
28     // open input file
29     std::ifstream pdfile( pdgfile );
30     if( !pdfile ) {
31         cerr << "cannot open " << pdgfile << endl;
32         exit(-1);
33     }
34     // construct empty PDT
35     HepPDT::ParticleDataTable datacol( "2006 PDG Table" );
36     {
37         // Construct table builder
38         HepPDT::TableBuilder tb(datacol);
39         // read the input - put as many here as you want
40         if( !HepPDT::addPDGParticles( pdfile, tb ) ) { cout << "error reading PDG file " << endl; }
41     } // the tb destructor fills datacol
42     // done with pdfile, so close it
43     pdfile.close();
44     // open output file
45     std::ofstream wpdfile( outfile );
46     if( !wpdfile ) {
47         cerr << "cannot open " << outfile << endl;
48         exit(-1);
49     }
50     datacol.writeParticleData(wpdfile);
51     wpdfile << std::endl;
52
53     // output some pion information
54     HepPDT::ParticleData * pd;
55     pd=datacol.particle(HepPDT::ParticleID(111));
56     // test the ResonanceStructure cutoff methods here
57     if(pd) {
58         pd->write(wpdfile);
59         wpdfile << "Resonance info for 111 "
60             << pd->totalWidth().value() << " "
61             << pd->totalWidth().sigma() << " "

```

```
62             << pd->lowerCutoff() << " "
63             << pd->upperCutoff() << std::endl;
64     }
65     // -111 is an illegal particle, no info will be written
66     pd=datacol.particle(HepPDT::ParticleID(-111));
67     if(pd) pd->write(wpdfile);
68     pd=datacol.particle(HepPDT::ParticleID(211));
69     if(pd) pd->write(wpdfile);
70     // string lookup
71     pd=datacol.particle(std::string("pi0"));
72     if(pd) pd->write(wpdfile);
73
74     // reopen input file
75     std::ifstream pdfile2( pdgfile );
76     if( !pdfile2 ) {
77         cerr << "cannot open " << pdgfile << endl;
78         exit(-1);
79     }
80     // construct another PDT instance that knows how to deal with unknown heavy ions
81     // NOTE: normally you would construct a single ParticleDataTable with this option
82     HepPDT::ParticleDataTable pdt2( "Handle Heavy Ions", new HepPDT::HeavyIonUnknownID );
83     {
84         // Construct table builder
85         HepPDT::TableBuilder tb2(pdt2);
86         // read the input - put as many here as you want
87         if( !HepPDT::addPDGParticles( pdfile2, tb2 ) ) { cout << "error reading PDG file " << endl; }
88     }
89     // done with pdfile, so close it
90     pdfile2.close();
91
92     // try a heavy ion
93     pd=pdt2.particle(HepPDT::ParticleID(1000020040));
94     if(pd) pd->write(wpdfile);
95
96     // particle info
97     datacol.writeParticleInfo(wpdfile);
98
99     return 0;
100 }
```

10.19 testPID.cc

The **HepPDT** (p. 23) tests are also useful examples

test ParticleID methods

```

1 // -----
2 // TestPID.cc
3 // Author: Lynn Garren
4 //
5 // test ParticleID
6 //
7 // -----
8
9 #include <iostream>
10 #include <iomanip>
11
12 #include "HepPDT/defs.h"
13 #include "HepPDT/ParticleID.hh"
14
15 using std::cout;
16 using std::cerr;
17 using std::endl;
18 using std::setw;
19
20 int main()
21 {
22     int id[16] = { 5, 25, 15, 213, -3214, 10213, 9050225, -200543, 129050225,
23                   2000025, 3101, 3301, -2212, 1000020040, -1000060120, 555 };
24
25     int it;
26     int nr, nx;
27     int chg, sid, extra;
28     int js, ls;
29     for( it=0; it < 16; it++ ) {
30         HepPDT::ParticleID pid( id[it] );
31         nx = pid.digit(HepPDT::n);
32         nr = pid.digit(HepPDT::nr);
33         extra = pid.extraBits();
34         cout << endl;
35         cout << setw(18) << id[it] << ":" << nx << " " << nr
36             << " " << pid.digit(HepPDT::nl)
37             << " " << pid.digit(HepPDT::nq1) << " "
38             << pid.digit(HepPDT::nq2) << " " << pid.digit(HepPDT::nq3)
39             << " " << pid.digit(HepPDT::nj)
40             << " extra bits " << extra << endl;
41         js = pid.jSpin();
42         HepPDT::Quarks cqks = pid.quarks( );
43         ls = pid.lSpin();
44         sid = pid.fundamentalID();
45         chg = pid.threeCharge();
46         if( !pid.isValid() ) {
47             cout << "**** Invalid PID: " << pid.pid() << " ****" << endl;
48         } else {
49             if( pid.isHadron() ) {
50                 if( pid.isMeson() ) {
51                     cout << "meson" << setw(10) << id[it] << ":" << nx
52                         << " " << nr << " " << ls << " "
53                         << cqks.nq1 << " " << cqks.nq2 << " " << cqks.nq3
54                         << " " << js << " " << sid << " " << chg
55                         << " extra bits " << extra << endl;
56             } else if( pid.isBaryon() ) {
57                 cout << "baryon" << setw(10) << id[it] << ":" << nx
58                     << " " << nr << " " << ls << " "
59                     << cqks.nq1 << " " << cqks.nq2 << " " << cqks.nq3
60                     << " " << js << " " << sid << " " << chg
61                     << " extra bits " << extra << endl;
62     } else {

```

```

62             cout << "***** undefined hadron: " << pid.pid()
63             << " ****" << endl;
64         }
65     }
66     if( pid.isLepton() ) {
67         cout << "lepton " << setw(10) << id[it] << ":" << nx
68         << " " << nr << " " << ls << " "
69         << cqks.nq1 << " " << cqks.nq2 << " " << cqks.nq3
70         << " " << js << " " << sid << " " << chg
71         << " extra bits " << extra << endl;
72     }
73     if( pid.isDiQuark() ) {
74         cout << "diquark " << setw(10) << id[it] << ":" << nx
75         << " " << nr << " " << ls << " "
76         << cqks.nq1 << " " << cqks.nq2 << " " << cqks.nq3
77         << " " << js << " " << sid << " " << chg
78         << " extra bits " << extra << endl;
79     }
80     if( pid.isNucleus() ) {
81         cout << "ion " << setw(11) << id[it]
82         << ":" << pid.digit(HepPDT::n10)
83         << " " << setw(3) << pid.A()
84         << " " << setw(3) << pid.Z()
85         << " " << setw(3) << pid.lambdas()
86         << " " << js << " " << sid << " " << chg
87         << " extra bits " << extra << endl;
88     }
89     if( pid.isHadron() || pid.isLepton() || pid.isDiQuark() || pid.isNucleus() ) {
90     } else {
91         cout << "unknown " << setw(10) << id[it] << ":" << nx
92         << " " << nr << " " << ls << " "
93         << cqks.nq1 << " " << cqks.nq2 << " " << cqks.nq3
94         << " " << js << " " << sid << " " << chg
95         << " extra bits " << extra << endl;
96     }
97 }
98 if( pid.isValid() ) {
99     js = pid.jSpin();
100    chg = pid.threeCharge();
101    cout << "total spin: " << js << " " << HepPDT::spinitod(js) ;
102    cout << "orbital angular momentum: " << pid.lSpin() ;
103    cout << "spin: " << pid.sSpin() ;
104    cout << "charge: " << chg << " " << double(chg)/3.0 << endl;
105    HepPDT::Quarks qlist = pid.quarks( );
106    cout << "quarks: " << qlist.nq1 << " " << qlist.nq2 << " " << qlist.nq3 << endl;
107 }
108 }
109 }
```

10.20 testReadEvtGen.cc.in

The **HepPDT** (p. 23) tests are also useful examples

read EvtGen table and write it out

```

1 // $Id: testReadEvtGen.cc.in,v 1.1 2007/05/22 22:12:23 garren Exp $
2 // -----
3 // testReadEvtGen.cc
4 //
5 // read EvtGen table and write it out
6 //
7 // -----
8
9 #include <fstream>
10
11 #include "HepPDT/defs.h"
12 #include "HepPDT/TableBuilder.hh"
13 #include "HepPDT/ParticleDataTable.hh"
14
15 int main()
16 {
17     const char infile1[] = "@top_srcdir@/data/pdt.table";
18     const char infile2[] = "@top_srcdir@/data/DECAY.DEC";
19     const char outfile[] = "testReadEvtGen.out";
20     // open input files
21     std::ifstream pdfile1( infile1 );
22     if( !pdfile1 ) {
23         std::cerr << "cannot open " << infile1 << std::endl;
24         exit(-1);
25     }
26     // construct empty PDT
27     std::ifstream pdfile2( infile2 );
28     if( !pdfile2 ) {
29         std::cerr << "cannot open " << infile2 << std::endl;
30         exit(-1);
31     }
32     HepPDT::ParticleDataTable datacol( "EvtGen Table" );
33     {
34         // Construct table builder
35         HepPDT::TableBuilder tb(datacol);
36         // read the input - put as many here as you want
37         if( !addEvtGenParticles( pdfile1, tb ) ) { std::cout << "error reading EvtGen pdt file " << std::endl;
38         if( !addEvtGenParticles( pdfile2, tb ) ) { std::cout << "error reading EvtGen decay file " << std::endl;
39     } // the tb destructor fills datacol
40     std::ofstream wfile( outfile );
41     if( !wfile ) {
42         std::cerr << "cannot open " << outfile << std::endl;
43         exit(-1);
44     }
45     datacol.writeParticleData(wfile);
46
47     return 0;
48 }
```

10.21 testReadIsajet.cc.in

The **HepPDT** (p. 23) tests are also useful examples

read the isajet particle and decay tables for testing purposes Note that isaparticles.dat was created with PRTLST(...)

```

1 // -----
2 // readIsajet.cc
3 // Author: Lynn Garren
4 //
5 // read the isajet particle table for testing purposes
6 // note that isaparticles.dat was created with PRTLST(....)
7 //
8 // Usage:  readIsajet
9 //
10 // -----
11
12 #include <fstream>
13
14 #include "HepPDT/defs.h"
15 #include "HepPDT/TableBuilder.hh"
16 #include "HepPDT/ParticleDataTable.hh"
17
18 using std::cout;
19 using std::cerr;
20 using std::endl;
21
22 int main()
23 {
24     char pdgfile[300] = "@top_srcdir@/examples/data/isaparticles.dat";
25     const char outfile[] = "testReadIsajet.out";
26     // construct empty PDT
27     HepPDT::ParticleDataTable datacol( "Isajet Table" );
28     {
29         // open input files
30         std::ifstream pdfile( pdgfile );
31         if( !pdfile ) {
32             cerr << "cannot open " << pdgfile << endl;
33             exit(-1);
34         }
35         // Construct table builder
36         HepPDT::TableBuilder tb(datacol);
37         // read the input - put as many here as you want
38         if( !HepPDT::addIsajetParticles( pdfile, tb ) ) { cout << "error reading " << pdgfile << endl;
39     } // the tb destructor fills datacol
40     std::ofstream wpdfile( outfile );
41     if( !wpdfile ) {
42         cerr << "cannot open " << outfile << endl;
43         exit(-1);
44     }
45     datacol.writeParticleData(wpdfile);
46     wpdfile << std::endl;
47
48     // particle info
49     datacol.writeParticleInfo(wpdfile);
50
51     return 0;
52 }
```

10.22 testReadQQ.cc.in

The **HepPDT** (p. 23) tests are also useful examples

read QQ table and write it out

```

1 // $Id: testReadQQ.cc.in,v 1.1 2007/05/22 22:12:23 garren Exp $
2 // -----
3 // testReadQQ.cc
4 //
5 // read QQ table and write it out
6 //
7 // -----
8
9 #include <fstream>
10
11 #include "HepPDT/defs.h"
12 #include "HepPDT/TableBuilder.hh"
13 #include "HepPDT/ParticleDataTable.hh"
14
15 int main()
16 {
17     const char infile[] = "@srcdir@/listQQ.dec";
18     const char outfile[] = "testReadQQ.out";
19     // open input file
20     std::ifstream pdfile( infile );
21     if( !pdfile ) {
22         std::cerr << "cannot open " << infile << std::endl;
23         exit(-1);
24     }
25     // construct empty PDT
26     HepPDT::ParticleDataTable datacol( "QQ Table" );
27     {
28         // Construct table builder
29         HepPDT::TableBuilder tb(datacol);
30         // read the input - put as many here as you want
31         if( !addQQParticles( pdfile, tb ) )
32             { std::cout << "error reading QQ table file " << std::endl; }
33     } // the tb destructor fills the PDT
34     std::ofstream wpdfile( outfile );
35     if( !wpdfile ) {
36         std::cerr << "cannot open " << outfile << std::endl;
37         exit(-1);
38     }
39     // write a translation list
40     datacol.writeParticleTranslation( wpdfile );
41     // write the particle and decay info
42     datacol.writeParticleData( wpdfile );
43
44     return 0;
45 }
```

Index

/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/examples/ Directory Reference, 11
/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/examples/HepPDT/ Directory Reference, 13
/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/examples/HepPID/ Directory Reference, 19
/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/include/ Directory Reference, 20
/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/include/HepPDT/ Directory Reference, 14
/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/include/HepPID/ Directory Reference, 16
/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/src/ Directory Reference, 21
/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/src/HepPDT/ Directory Reference, 15
/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/src/HepPID/ Directory Reference, 17
/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/tests/ Directory Reference, 22
/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/tests/HepPDT/ Directory Reference, 12
/home/cepa01/garren/lcg/heppdt/HepPDT-
3.01.00/tests/HepPID/ Directory Reference, 18
~DefTable
 HepPDT::DefTable, 67
~ParticleData
 HepPDT::ParticleData, 77
~ParticleDataTable
 HepPDT::ParticleDataTable, 87
~ParticleNameMap
 HepPID::ParticleNameMap, 102
~ResonanceStructure
 HepPDT::ResonanceStructure, 109
~TableBuilder
 HepPDT::TableBuilder, 117

A

- HepPDT::ParticleID, 100
- HepPID, 44

abspid
 HepPDT::ParticleID, 96

HepPID, 44

addAlias
 HepPDT::TableBuilder, 118

addConstituent
 HepPDT::ParticleData, 83

addData
 examMyPDT.cc, 150

addDefinition
 HepPDT::DefTable, 67

addEvtGenParticles
 HepPDT, 27

addEvtGenParticles.cc, 131

addHerwigParticles
 HepPDT, 27

addHerwigParticles.cc, 132

addIsajetDecay
 HepPDT, 27

addIsajetParticles
 HepPDT, 27

addIsajetParticles.cc, 133

addParticle
 HepPDT::TableBuilder, 118

addParticleTable
 HepPDT, 27

addParticleTable.cc, 134

addPDGParticles
 HepPDT, 28

addPDGParticles.cc, 135

addPythiaParticles
 HepPDT, 28

addPythiaParticles.cc, 136

addQQParticles
 HepPDT, 28

addQQParticles.cc, 137

aliasData
 HepPDT::TableBuilder, 120

aliasSize
 HepPDT::TableBuilder, 119
 antiparticle
 HepPDT::TempParticleData, 127
 begin
 HepPDT::DefTable, 68
 HepPDT::ParticleDataTable, 88
 HepPID::ParticleNameMap, 103
 beginLookupMap
 HepPID::ParticleNameMap, 103
 beginNameMap
 HepPDT::ParticleDataTable, 88
 calculateWidthFromLifetime
 HepPDT, 29
 calculateWidthFromLifetime.cc, 138
 charge
 HepPDT::ParticleData, 78
 CheckPDGEntry
 HepPDT::detail, 34
 color
 HepPDT::ParticleData, 78
 const_iterator
 HepPDT::DefTable, 67
 HepPDT::ParticleDataTable, 87
 const_iteratorByName
 HepPDT::ParticleDataTable, 87
 Constituent
 HepPDT::Constituent, 64
 constituent
 HepPDT::ParticleData, 80
 Constituent.cc, 139
 Constituent.hh, 140
 constituentParticle
 HepPDT::ParticleData, 80
 convertTemporaryMap
 HepPDT::ParticleDataTable, 91
 convertTemporaryMap.cc, 141
 convIsajettoPDT
 HepPID, 44
 convPDTtoIsajet
 HepPID, 44
 CPD
 HepPDT::ParticleDataTable, 87
 CPDID
 HepPDT::ParticleDataTable, 87
 CPDlist
 HepPDT::ParticleDataTable, 87
 definition
 HepPDT::DefTable, 67
 HepPDT::TableBuilder, 120
 definitions
 HepPDT::TableBuilder, 119
 DefTable
 HepPDT::DefTable, 67
 DefTable.cc, 142
 DefTable.hh, 143
 digit
 HepPDT::ParticleID, 101
 HepPID, 44
 end
 HepPDT::DefTable, 68
 HepPDT::ParticleDataTable, 88
 HepPID::ParticleNameMap, 103
 endLookupMap
 HepPID::ParticleNameMap, 103
 endNameMap
 HepPDT::ParticleDataTable, 88
 EvtGenPDTMap
 HepPID, 42
 EvtGenPDTMapInit
 HepPID, 45
 examListHerwig.cc, 144
 examListHerwig.cc
 get_herwig_name_, 144
 get_list_size_, 144
 list_herwig_end_, 144
 list_herwig_init_, 144
 main, 145
 examListIsajet.cc, 146
 examListIsajet.cc
 flavor_, 146
 get_label_, 146
 list_isajet_init_, 146
 main, 146
 examListPythia.cc, 148
 examListPythia.cc
 getkf_, 148
 getpyname_, 148
 list_pythia_, 148
 main, 148
 writeLine, 149
 examMyPDT.cc, 150
 examMyPDT.cc
 addData, 150
 main, 150
 extraBits
 HepPDT::ParticleID, 100
 HepPID, 45
 find
 HepPID::ParticleNameMap, 103
 findString
 HepPID::ParticleNameMap, 103
 flavor_

```

examListIsajet.cc, 146
fundamentalID
    HepPDT::ParticleID, 99
    HepPID, 45

get_herwig_name_
    examListHerwig.cc, 144
get_label_
    examListIsajet.cc, 146
get_list_size_
    examListHerwig.cc, 144
getAntiParticle
    HepPDT::TableBuilder, 118
getEvtGenLineType
    HepPDT, 29
getEvtGenPDTMap
    HepPID, 45
getHerwigPDTMap
    HepPID, 45
getInverseQQbarMap
    HepPID, 46
getIsajetID
    HepPDT::detail, 34
getIsajetID.cc, 152
getIsajetPDTMap
    HepPID, 46
getkf_
    examListPythia.cc, 148
getParticleData
    HepPDT::TableBuilder, 117, 118
getParticleID
    HepPDT::detail, 34
getParticleNameMap
    HepPID, 46
getPDGnames
    HepPDT::detail, 34
getPDGpid
    HepPDT::detail, 34
getPDGpid.cc, 153
getPDGtoPDTMap
    HepPID, 46
getPDTEvtGenMap
    HepPID, 46
getPDTHerwigMap
    HepPID, 47
getPDTIsajetMap
    HepPID, 47
getPDTPythiaMap
    HepPID, 47
getPDTQQMap
    HepPID, 47
getPDTtoPDGMap
    HepPID, 47
getpyname_
    HepPDT, 23

examListPythia.cc, 148
getPythiaDecay
    HepPDT::detail, 34
getPythiaid
    HepPDT::detail, 34
getPythiaid.cc, 154
getPythiaPDTMap
    HepPID, 48
getQQbarMap
    HepPID, 48
getQQLineType
    HepPDT, 29
getQQPDTMap
    HepPID, 48

hasAlias
    HepPDT::TableBuilder, 119
hasBottom
    HepPDT::ParticleData, 82
    HepPDT::ParticleID, 98
    HepPID, 48
hasCharm
    HepPDT::ParticleData, 82
    HepPDT::ParticleID, 98
    HepPID, 48
hasDefinition
    HepPDT::DefTable, 67
    HepPDT::TableBuilder, 119
hasDown
    HepPDT::ParticleData, 82
    HepPDT::ParticleID, 98
    HepPID, 48
hasFundamentalAnti
    HepPID, 49
hasMethods.cc, 155
hasParticleData
    HepPDT::TableBuilder, 119
hasStrange
    HepPDT::ParticleData, 82
    HepPDT::ParticleID, 98
    HepPID, 49
hasTop
    HepPDT::ParticleData, 82
    HepPDT::ParticleID, 99
    HepPID, 49
hasUp
    HepPDT::ParticleData, 82
    HepPDT::ParticleID, 98
    HepPID, 49
HeavyIonUnknownID
    HepPDT::HeavyIonUnknownID, 69
HeavyIonUnknownID.cc, 156
HeavyIonUnknownID.hh, 157
HepPDT, 23

```

n, 26
 n10, 27
 n8, 26
 n9, 26
 nj, 26
 nl, 26
 nq1, 26
 nq2, 26
 nq3, 26
 nr, 26
HepPDT
 addEvtGenParticles, 27
 addHerwigParticles, 27
 addIsajetDecay, 27
 addIsajetParticles, 27
 addParticleTable, 27
 addPDGParticles, 28
 addPythiaParticles, 28
 addQQParticles, 28
 calculateWidthFromLifetime, 29
 getEvtGenLineType, 29
 getQQLineType, 29
 location, 26
 NaN, 29
 parseEvtGenAlias, 29
 parseEvtGenAliasDecayLine, 29
 parseEvtGenConj, 29
 parseEvtGenDecayLine, 29
 parseEvtGenDefinition, 30
 parseEvtGenLine, 30
 parseQQDecayLine, 30
 parseQQParticle, 30
 spindtoi, 30
 spinitod, 30
 stringtoDouble, 31
 swap, 31, 32
 TDDlist, 26
 version, 32
 versionName, 32
 writeEvtGenStream, 32
 writeHerwigStream, 32
 writeIsajetStream, 32
 writePDGStream, 32
 writePythiaStream, 32
 writeQQStream, 32
 writeVersion, 32
HepPDT::Constituent, 63
HepPDT::Constituent
 Constituent, 64
 isBottom, 65
 isCharm, 65
 isDown, 65
 isStrange, 65
 isTop, 65
 isUp, 64
 multiplicity, 64
 operator=, 64
 pid, 64
 swap, 64
HepPDT::DefTable, 66
HepPDT::DefTable
 ~DefTable, 67
 addDefinition, 67
 begin, 68
 const_iterator, 67
 definition, 67
 DefTable, 67
 end, 68
 hasDefinition, 67
 iterator, 67
 size, 67
 writeDefinitions, 68
HepPDT::detail, 33
HepPDT::detail
 CheckPDGEntry, 34
 getIsajetID, 34
 getParticleID, 34
 getPDGnames, 34
 getPDGpid, 34
 getPythiaDecay, 34
 getPythiaid, 34
 parseIsajetDecayLine, 35
 parseIsajetLine, 35
 parseParticleLine, 35
 parsePDGline, 35
 parsePythiaDecayLine, 35
 parsePythiaLine, 36
HepPDT::HeavyIonUnknownID, 69
HepPDT::HeavyIonUnknownID
 HeavyIonUnknownID, 69
 processUnknownID, 69
HepPDT::Measurement, 71
HepPDT::Measurement
 Measurement, 71
 operator double, 72
 operator<, 72
 operator=, 72
 operator==, 72
 sigma, 72
 swap, 72
 value, 72
HepPDT::ParticleData, 74
HepPDT::ParticleData
 ~ParticleData, 77
 addConstituent, 83
 charge, 78
 color, 78
 constituent, 80

constituentParticle, 80
hasBottom, 82
hasCharm, 82
hasDown, 82
hasStrange, 82
hasTop, 82
hasUp, 82
ID, 78
isBaryon, 81
isDiQuark, 81
isHadron, 81
isLepton, 81
isMeson, 81
isNucleus, 82
lifetime, 80
lowerCutoff, 79
mass, 79
name, 77
numConstituents, 80
operator<, 82
operator=, 77
operator==, 83
originalID, 78
ParticleData, 77
PDTname, 78
pid, 78
resonance, 80
setCharge, 83
setColor, 83
setLowerCutoff, 84
setMass, 83
setSpin, 83
setTotalWidth, 83
setTotalWidthFromLifetime, 84
setUpperCutoff, 84
source, 78
spin, 79
swap, 77
totalWidth, 79
upperCutoff, 79
write, 80
writeParticleInfo, 81
writeParticleTranslation, 81
HepPDT::ParticleDataTable, 85
HepPDT::ParticleDataTable
 ~ParticleDataTable, 87
 begin, 88
 beginNameMap, 88
 const_iterator, 87
 const_iteratorByName, 87
 convertTemporaryMap, 91
 CPD, 87
 CPDID, 87
 CPDlist, 87
 end, 88
 endNameMap, 88
 operator[], 89, 90
 particle, 89
 ParticleDataTable, 87
 PDTMap, 87
 PDTNameMap, 87
 size, 88
 sizeNameMap, 88
 tableName, 88
 TempMap, 87
 writeParticleData, 90
 writeParticleInfo, 90
 writeParticleTranslation, 90
 HepPDT::ParticleDataTableComparison, 92
 HepPDT::ParticleDataTableComparison
 operator(), 92
 ParticleDataTableComparison, 92
 HepPDT::ParticleID, 93
 HepPDT::ParticleID
 A, 100
 abspid, 96
 digit, 101
 extraBits, 100
 fundamentalID, 99
 hasBottom, 98
 hasCharm, 98
 hasDown, 98
 hasStrange, 98
 hasTop, 99
 hasUp, 98
 isBaryon, 96
 isDiQuark, 97
 isHadron, 97
 isLepton, 97
 isMeson, 96
 isNucleus, 97
 isPentaquark, 97
 isRhadron, 98
 isSUSY, 97
 isValid, 96
 jSpin, 99
 lambda, 100
 lSpin, 99
 operator<, 95
 operator=, 95
 operator==, 95
 ParticleID, 95
 PDTname, 101
 pid, 96
 quarks, 100
 sSpin, 99
 swap, 95
 threeCharge, 100

Z, 100
HepPDT::ProcessUnknownID, 104
HepPDT::ProcessUnknownID
 ProcessUnknownID, 104
 processUnknownID, 104
HepPDT::Quarks, 106
HepPDT::Quarks
 nq1, 106
 nq2, 106
 nq3, 107
 Quarks, 106
HepPDT::ResonanceStructure, 108
HepPDT::ResonanceStructure
 ~ResonanceStructure, 109
 lifetime, 110
 lowerCutoff, 110
 mass, 109
 operator=, 109
 ResonanceStructure, 109
 setLowerCutoff, 111
 setMass, 110
 setTotalWidth, 110
 setTotalWidthFromLifetime, 110
 setUpperCutoff, 111
 swap, 109
 totalWidth, 110
 upperCutoff, 110
HepPDT::SimpleProcessUnknownID, 112
HepPDT::SimpleProcessUnknownID
 processUnknownID, 112
 SimpleProcessUnknownID, 112
HepPDT::SpinState, 113
HepPDT::SpinState
 operator=, 114
 operator==, 114
 orbAngMom, 114
 setOrbAngMom, 115
 setSpin, 115
 setTotalSpin, 115
 spin, 114
 SpinState, 114
 swap, 114
 totalSpin, 114
HepPDT::TableBuilder, 116
HepPDT::TableBuilder
 ~TableBuilder, 117
 addAlias, 118
 addParticle, 118
 aliasData, 120
 aliasSize, 119
 definition, 120
 definitions, 119
 getAntiParticle, 118
 getParticleData, 117, 118
 hasAlias, 119
 hasDefinition, 119
 hasParticleData, 119
 removeParticle, 118
 size, 119
 TableBuilder, 117
HepPDT::TempAliasData, 121
HepPDT::TempAliasData
 tempAlias, 121
 TempAliasData, 121
 tempAliasDecayList, 122
 tempAliasedParticle, 121
 tempChargeConj, 122
HepPDT::TempConstituent, 123
HepPDT::TempConstituent
 TempConstituent, 123
 tempConstituentPID, 123
 tempMultiplicity, 123
HepPDT::TempDecayData, 124
HepPDT::TempDecayData
 tempBranchingFraction, 124
 tempDaughterList, 124
 tempDecayName, 124
 tempDecayParameters, 124
HepPDT::TempParticleData, 125
HepPDT::TempParticleData
 antiparticle, 127
 operator=, 126
 processPID, 127
 swap, 126
 tempCharge, 128
 tempColorCharge, 128
 tempDecayList, 129
 tempHighCutoff, 129
 tempID, 127
 tempLowCutoff, 129
 tempMass, 128
 tempOriginalID, 128
 TempParticleData, 126
 tempParticleName, 127
 tempQuarks, 129
 tempSource, 128
 tempSpin, 128
 tempWidth, 129
HepPID, 37
 n, 44
 n10, 44
 n8, 44
 n9, 44
 nj, 43
 nl, 44
 nq1, 44
 nq2, 43
 nq3, 43

nr, 44
HepPID
A, 44
abspid, 44
convIsajettoPDT, 44
convPDTtoIsajet, 44
digit, 44
EvtGenPDTMap, 42
EvtGenPDTMapInit, 45
extraBits, 45
fundamentalID, 45
getEvtGenPDTMap, 45
getHerwigPDTMap, 45
getInverseQQbarMap, 46
getIsajetPDTMap, 46
getParticleNameMap, 46
getPDGtoPDTMap, 46
getPDTEvtGenMap, 46
getPDTHerwigMap, 47
getPDTIsajetMap, 47
getPDTPythiaMap, 47
getPDTQQMap, 47
getPDTtoPDGMap, 47
getPythiaPDTMap, 48
getQQbarMap, 48
getQQPDTMap, 48
hasBottom, 48
hasCharm, 48
hasDown, 48
hasFundamentalAnti, 49
hasStrange, 49
hasTop, 49
hasUp, 49
HerwigPDTMap, 42
HerwigPDTMapInit, 49
InverseQQbarMap, 42
InverseQQbarMapInit, 49
IsajetPDTMap, 42
IsajetPDTMapInit, 49
isBaryon, 50
isDiQuark, 50
isHadron, 50
isLepton, 50
isMeson, 50
isNucleus, 50
isPentaquark, 51
isRhadron, 51
isSUSY, 51
isValid, 51
jSpin, 51
lambda, 52
listParticleNames, 52
location, 43
lSpin, 52
ParticleIdMap, 42
ParticleLookupMap, 42
particleName, 52
ParticleNameInit, 53
PDGtoPDTMap, 42
PDGtoPDTMapInit, 53
PDTEvtGenMap, 42
PDTEvtGenMapInit, 53
PDTHerwigMap, 43
PDTHerwigMapInit, 53
PDTIsajetMap, 43
PDTIsajetMapInit, 53
PDTPythiaMap, 43
PDTPythiaMapInit, 53
PDTQQMap, 43
PDTQQMapInit, 53
PDTtoPDGMap, 43
PDTtoPDGMapInit, 54
PythiaPDTMap, 43
PythiaPDTMapInit, 54
QQbarMap, 43
QQbarMapInit, 54
QQPDTMap, 43
QQPDTMapInit, 54
sSpin, 54
threeCharge, 54
translateEvtGentoPDT, 54
translateGeanttoPDT, 55
translateHerwigtoPDT, 55
translateInverseQQbar, 55
translateIsajettoPDT, 55
translatePDGtabletoPDT, 55
translatePDTtoEvtGen, 56
translatePDTtoGeant, 56
translatePDTtoHerwig, 56
translatePDTtoIsajet, 56
translatePDTtoPDGtable, 56
translatePDTtoPythia, 56
translatePDTtoQQ, 57
translatePythiatoPDT, 57
translateQQbar, 57
translateQQtoPDT, 57
validParticleName, 57, 58
version, 58
versionName, 58
writeEvtGenTranslation, 58
writeEvtGenTranslationLine, 58
writeHerwigTranslation, 58
writeHerwigTranslationLine, 59
writeIsajetTranslation, 59
writeIsajetTranslationLine, 59
writeParticleNameLine, 59
writePDGTranslation, 59
writePDGTranslationLine, 60

writePythiaTranslation, 60
 writePythiaTranslationLine, 60
 writeQQTranslation, 60
 writeVersion, 60
 Z, 61
 HepPID::ParticleNameMap, 102
 HepPID::ParticleNameMap
 ~ParticleNameMap, 102
 begin, 103
 beginLookupMap, 103
 end, 103
 endLookupMap, 103
 find, 103
 findString, 103
 idIterator, 102
 lookupMap, 103
 nameIterator, 102
 nameMap, 103
 ParticleNameMap, 102
 HerwigPDTMap
 HepPID, 42
 HerwigPDTMapInit
 HepPID, 49

 ID
 HepPDT::ParticleData, 78
 idIterator
 HepPID::ParticleNameMap, 102
 IDMAX
 translateGeanttoPDT.cc, 214
 translatePDTtoGeant.cc, 218
 InverseQQbarMap
 HepPID, 42
 InverseQQbarMapInit
 HepPID, 49
 IsajetPDTMap
 HepPID, 42
 IsajetPDTMapInit
 HepPID, 49
 isBaryon
 HepPDT::ParticleData, 81
 HepPDT::ParticleID, 96
 HepPID, 50
 isBottom
 HepPDT::Constituent, 65
 isCharm
 HepPDT::Constituent, 65
 isDiQuark
 HepPDT::ParticleData, 81
 HepPDT::ParticleID, 97
 HepPID, 50
 isDown
 HepPDT::Constituent, 65
 isHadron

 HepPDT::ParticleData, 81
 HepPDT::ParticleID, 97
 HepPID, 50
 isLepton
 HepPDT::ParticleData, 81
 HepPDT::ParticleID, 97
 HepPID, 50
 isMeson
 HepPDT::ParticleData, 81
 HepPDT::ParticleID, 96
 HepPID, 50
 isNucleus
 HepPDT::ParticleData, 82
 HepPDT::ParticleID, 97
 HepPID, 50
 isPentaquark
 HepPDT::ParticleID, 97
 HepPID, 51
 isRhadron
 HepPDT::ParticleID, 98
 HepPID, 51
 isStrange
 HepPDT::Constituent, 65
 isSUSY
 HepPDT::ParticleID, 97
 HepPID, 51
 isTop
 HepPDT::Constituent, 65
 isUp
 HepPDT::Constituent, 64
 isValid
 HepPDT::ParticleID, 96
 HepPID, 51
 iterator
 HepPDT::DefTable, 67

 jSpin
 HepPDT::ParticleID, 99
 HepPID, 51

 lambda
 HepPDT::ParticleID, 100
 HepPID, 52
 lifetime
 HepPDT::ParticleData, 80
 HepPDT::ResonanceStructure, 110
 lifetime.cc, 158
 list_herwig_end_
 examListHerwig.cc, 144
 list_herwig_init_
 examListHerwig.cc, 144
 list_isajet_init_
 examListIsajet.cc, 146
 list_of_examples.cc, 159

list_of_tests.cc, 160
 list_pythia_
 examListPythia.cc, 148
 listEvtGenNames.cc.in, 161
 listEvtGenNames.cc.in
 main, 161
 listEvtGenTranslation.cc, 162
 listEvtGenTranslation.cc
 main, 162
 listHerwigTranslation.cc, 163
 listHerwigTranslation.cc
 main, 163
 listIsajetTranslation.cc, 164
 listIsajetTranslation.cc
 main, 164
 listParticleNames
 HepPID, 52
 listParticleNames.cc, 165
 listParticleNames.cc
 main, 165
 listPDGNames.cc.in, 166
 listPDGNames.cc.in
 main, 166
 listPDGTranslation.cc, 167
 listPDGTranslation.cc
 main, 167
 listPythiaNames.cc.in, 168
 listPythiaNames.cc.in
 main, 168
 listPythiaTranslation.cc, 169
 listPythiaTranslation.cc
 main, 169
 listQQTranslation.cc, 170
 listQQTranslation.cc
 main, 170
 location
 HepPDT, 26
 HepPID, 43
 lookupMap
 HepPID::ParticleNameMap, 103
 lowerCutoff
 HepPDT::ParticleData, 79
 HepPDT::ResonanceStructure, 110
 lSpin
 HepPDT::ParticleID, 99
 HepPID, 52
 main
 examListHerwig.cc, 145
 examListIsajet.cc, 146
 examListPythia.cc, 148
 examMyPDT.cc, 150
 listEvtGenNames.cc.in, 161
 listEvtGenTranslation.cc, 162
 listHerwigTranslation.cc, 163
 listIsajetTranslation.cc, 164
 listParticleNames.cc, 165
 listPDGNames.cc.in, 166
 listPDGTranslation.cc, 167
 listPythiaNames.cc.in, 168
 listPythiaTranslation.cc, 169
 listQQTranslation.cc, 170
 testHepPDT.cc, 206
 testParticleIDMethods.cc, 207
 testPID.cc, 208
 testReadEvtGen.cc.in, 209
 testReadIsajet.cc.in, 210
 testReadParticleTable.cc.in, 211
 testReadQQ.cc.in, 212
 mass
 HepPDT::ParticleData, 79
 HepPDT::ResonanceStructure, 109
 Measurement
 HepPDT::Measurement, 71
 Measurement.hh, 171
 Measurement.icc, 172
 multiplicity
 HepPDT::Constituent, 64
 n
 HepPDT, 26
 HepPID, 44
 n10
 HepPDT, 27
 HepPID, 44
 n8
 HepPDT, 26
 HepPID, 44
 n9
 HepPDT, 26
 HepPID, 44
 name
 HepPDT::ParticleData, 77
 nameIterator
 HepPID::ParticleNameMap, 102
 nameMap
 HepPID::ParticleNameMap, 103
 NaN
 HepPDT, 29
 nj
 HepPDT, 26
 HepPID, 43
 nl
 HepPDT, 26
 HepPID, 44
 nq1
 HepPDT, 26
 HepPDT::Quarks, 106

HepPID, 44
 nq2
 HepPDT, 26
 HepPDT::Quarks, 106
 HepPID, 43
 nq3
 HepPDT, 26
 HepPDT::Quarks, 107
 HepPID, 43
 nr
 HepPDT, 26
 HepPID, 44
 numConstituents
 HepPDT::ParticleData, 80

 operator double
 HepPDT::Measurement, 72
 operator()
 HepPDT::ParticleDataTableComparison,
 92
 operator<
 HepPDT::Measurement, 72
 HepPDT::ParticleData, 82
 HepPDT::ParticleID, 95
 operator=
 HepPDT::Constituent, 64
 HepPDT::Measurement, 72
 HepPDT::ParticleData, 77
 HepPDT::ParticleID, 95
 HepPDT::ResonanceStructure, 109
 HepPDT::SpinState, 114
 HepPDT::TempParticleData, 126
 operator==
 HepPDT::Measurement, 72
 HepPDT::ParticleData, 83
 HepPDT::ParticleID, 95
 HepPDT::SpinState, 114
 operator[]
 HepPDT::ParticleDataTable, 89, 90
 orbAngMom
 HepPDT::SpinState, 114
 originalID
 HepPDT::ParticleData, 78

 parseEvtGenAlias
 HepPDT, 29
 parseEvtGenAliasDecayLine
 HepPDT, 29
 parseEvtGenConj
 HepPDT, 29
 parseEvtGenDecayLine
 HepPDT, 29
 parseEvtGenDefinition
 HepPDT, 30

 parseEvtGenLine
 HepPDT, 30
 parseIsajetDecayLine
 HepPDT::detail, 35
 parseIsajetLine
 HepPDT::detail, 35
 parseParticleLine
 HepPDT::detail, 35
 parsePDGline
 HepPDT::detail, 35
 parsePythiaDecayLine
 HepPDT::detail, 35
 parsePythiaLine
 HepPDT::detail, 36
 parseQQDecayLine
 HepPDT, 30
 parseQQParticle
 HepPDT, 30
 ParticleIdMap
 HepPID, 42
 particle
 HepPDT::ParticleDataTable, 89
 ParticleData
 HepPDT::ParticleData, 77
 ParticleData.hh, 173
 ParticleData.icc, 174
 ParticleDataTable
 HepPDT::ParticleDataTable, 87
 ParticleDataTable.cc, 175
 ParticleDataTable.hh, 176
 ParticleDataTable.icc, 177
 ParticleDataTableComparison
 HepPDT::ParticleDataTableComparison,
 92
 ParticleDataTableComparison.hh, 178
 ParticleID
 HepPDT::ParticleID, 95
 ParticleID.cc, 179
 ParticleID.hh, 180
 ParticleID.icc, 181
 ParticleIDMethods.cc, 182
 ParticleIDMethods.hh, 184
 ParticleIDTranslations.hh, 186
 ParticleLookupMap
 HepPID, 42
 particleName
 HepPID, 52
 ParticleName.cc, 188
 ParticleName.hh, 189
 ParticleNameInit
 HepPID, 53
 ParticleNameMap
 HepPID::ParticleNameMap, 102
 PDGtoPDTMap

HepPID, 42
 PDGtoPDTMapInit
 HepPID, 53
 PDTEvtGenMap
 HepPID, 42
 PDTEvtGenMapInit
 HepPID, 53
 PDTHerwigMap
 HepPID, 43
 PDTHerwigMapInit
 HepPID, 53
 PDTIsajetMap
 HepPID, 43
 PDTIsajetMapInit
 HepPID, 53
 PDTMap
 HepPDT::ParticleDataTable, 87
 PDTname
 HepPDT::ParticleData, 78
 HepPDT::ParticleID, 101
 PDTNameMap
 HepPDT::ParticleDataTable, 87
 PDTPythiaMap
 HepPID, 43
 PDTPythiaMapInit
 HepPID, 53
 PDTQQMap
 HepPID, 43
 PDTQQMapInit
 HepPID, 53
 PDTtoPDGMap
 HepPID, 43
 PDTtoPDGMapInit
 HepPID, 54
 pid
 HepPDT::Constituent, 64
 HepPDT::ParticleData, 78
 HepPDT::ParticleID, 96
 processPID
 HepPDT::TempParticleData, 127
 ProcessUnknownID
 HepPDT::ProcessUnknownID, 104
 processUnknownID
 HepPDT::HeavyIonUnknownID, 69
 HepPDT::ProcessUnknownID, 104
 HepPDT::SimpleProcessUnknownID, 112
 ProcessUnknownID.hh, 190
 PythiaPDTMap
 HepPID, 43
 PythiaPDTMapInit
 HepPID, 54
 QQbarMap
 HepPID, 43
 QQbarMapInit
 HepPID, 54
 QQPDTMap
 HepPID, 43
 QQPDTMapInit
 HepPID, 54
 Quarks
 HepPDT::Quarks, 106
 quarks
 HepPDT::ParticleID, 100
 quarks.cc, 191
 removeParticle
 HepPDT::TableBuilder, 118
 resonance
 HepPDT::ParticleData, 80
 ResonanceStructure
 HepPDT::ResonanceStructure, 109
 ResonanceStructure.cc, 192
 ResonanceStructure.hh, 193
 setCharge
 HepPDT::ParticleData, 83
 setColor
 HepPDT::ParticleData, 83
 setLowerCutoff
 HepPDT::ParticleData, 84
 HepPDT::ResonanceStructure, 111
 setMass
 HepPDT::ParticleData, 83
 HepPDT::ResonanceStructure, 110
 setOrbAngMom
 HepPDT::SpinState, 115
 setSpin
 HepPDT::ParticleData, 83
 HepPDT::SpinState, 115
 setTotalSpin
 HepPDT::SpinState, 115
 setTotalWidth
 HepPDT::ParticleData, 83
 HepPDT::ResonanceStructure, 110
 setTotalWidthFromLifetime
 HepPDT::ParticleData, 84
 HepPDT::ResonanceStructure, 110
 setUpperCutoff
 HepPDT::ParticleData, 84
 HepPDT::ResonanceStructure, 111
 sigma
 HepPDT::Measurement, 72
 SimpleProcessUnknownID
 HepPDT::SimpleProcessUnknownID, 112
 SimpleProcessUnknownID.hh, 194
 size
 HepPDT::DefTable, 67

HepPDT::ParticleDataTable, 88
 HepPDT::TableBuilder, 119
 sizeNameMap
 HepPDT::ParticleDataTable, 88
 source
 HepPDT::ParticleData, 78
 spin
 HepPDT::ParticleData, 79
 HepPDT::SpinState, 114
 spindtoi
 HepPDT, 30
 spindtoi.cc, 195
 spinitod
 HepPDT, 30
 spinitod.cc, 196
 SpinState
 HepPDT::SpinState, 114
 SpinState.hh, 197
 SpinState.icc, 198
 sSpin
 HepPDT::ParticleID, 99
 HepPID, 54
 std, 62
 stringtoDouble
 HepPDT, 31
 stringtoDouble.cc, 199
 stringtoDouble.hh, 200
 swap
 HepPDT, 31, 32
 HepPDT::Constituent, 64
 HepPDT::Measurement, 72
 HepPDT::ParticleData, 77
 HepPDT::ParticleID, 95
 HepPDT::ResonanceStructure, 109
 HepPDT::SpinState, 114
 HepPDT::TempParticleData, 126

TableBuilder
 HepPDT::TableBuilder, 117
 TableBuilder.hh, 201
 TableBuilder.icc, 203
 tableName
 HepPDT::ParticleDataTable, 88
 TDDlist
 HepPDT, 26
 tempAlias
 HepPDT::TempAliasData, 121
 TempAliasData
 HepPDT::TempAliasData, 121
 tempAliasDecayList
 HepPDT::TempAliasData, 122
 tempAliasedParticle
 HepPDT::TempAliasData, 121
 tempBranchingFraction

HepPDT::TempDecayData, 124
 tempCharge
 HepPDT::TempParticleData, 128
 tempChargeConj
 HepPDT::TempAliasData, 122
 tempColorCharge
 HepPDT::TempParticleData, 128
 TempConstituent
 HepPDT::TempConstituent, 123
 tempConstituentPID
 HepPDT::TempConstituent, 123
 tempDaughterList
 HepPDT::TempDecayData, 124
 tempDecayList
 HepPDT::TempParticleData, 129
 tempDecayName
 HepPDT::TempDecayData, 124
 tempDecayParameters
 HepPDT::TempDecayData, 124
 tempHighCutoff
 HepPDT::TempParticleData, 129
 tempID
 HepPDT::TempParticleData, 127
 tempLowCutoff
 HepPDT::TempParticleData, 129
 TempMap
 HepPDT::ParticleDataTable, 87
 tempMass
 HepPDT::TempParticleData, 128
 tempMultiplicity
 HepPDT::TempConstituent, 123
 tempOriginalID
 HepPDT::TempParticleData, 128
 TempParticleData
 HepPDT::TempParticleData, 126
 TempParticleData.cc, 204
 TempParticleData.hh, 205
 tempParticleName
 HepPDT::TempParticleData, 127
 tempQuarks
 HepPDT::TempParticleData, 129
 tempSource
 HepPDT::TempParticleData, 128
 tempSpin
 HepPDT::TempParticleData, 128
 tempWidth
 HepPDT::TempParticleData, 129
 testHepPDT.cc, 206
 testHepPDT.cc
 main, 206
 testParticleIDMethods.cc, 207
 testParticleIDMethods.cc
 main, 207
 testPID.cc, 208

testPID.cc
 main, 208
testReadEvtGen.cc.in, 209
testReadEvtGen.cc.in
 main, 209
testReadIsajet.cc.in, 210
testReadIsajet.cc.in
 main, 210
testReadParticleTable.cc.in, 211
testReadParticleTable.cc.in
 main, 211
testReadQQ.cc.in, 212
testReadQQ.cc.in
 main, 212
threeCharge
 HepPDT::ParticleID, 100
 HepPID, 54
totalSpin
 HepPDT::SpinState, 114
totalWidth
 HepPDT::ParticleData, 79
 HepPDT::ResonanceStructure, 110
translateEvtGen.cc, 213
translateEvtGentoPDT
 HepPID, 54
translateGeanttoPDT
 HepPID, 55
translateGeanttoPDT.cc, 214
translateGeanttoPDT.cc
 IDMAX, 214
translateHerwig.cc, 215
translateHerwigtoPDT
 HepPID, 55
translateInverseQQbar
 HepPID, 55
translateIsajet.cc, 216
translateIsajettoPDT
 HepPID, 55
translatePDG.cc, 217
translatePDGtabletoPDT
 HepPID, 55
translatePDTtoEvtGen
 HepPID, 56
translatePDTtoGeant
 HepPID, 56
translatePDTtoGeant.cc, 218
translatePDTtoGeant.cc
 IDMAX, 218
translatePDTtoHerwig
 HepPID, 56
translatePDTtoIsajet
 HepPID, 56
translatePDTtoPDGtable
 HepPID, 56
translatePDTtoPythia
 HepPID, 56
translatePDTtoQQ
 HepPID, 57
translatePythia.cc, 219
translatePythiatoPDT
 HepPID, 57
translateQQ.cc, 220
translateQQbar
 HepPID, 57
translateQQtoPDT
 HepPID, 57
upperCutoff
 HepPDT::ParticleData, 79
 HepPDT::ResonanceStructure, 110
validParticleName
 HepPID, 57, 58
value
 HepPDT::Measurement, 72
version
 HepPDT, 32
 HepPID, 58
Version.cc, 222, 223
Version.hh, 224, 225
versionName
 HepPDT, 32
 HepPID, 58
write
 HepPDT::ParticleData, 80
write.cc, 226
writeDefinitions
 HepPDT::DefTable, 68
writeEvtGenStream
 HepPDT, 32
writeEvtGenTranslation
 HepPID, 58
writeEvtGenTranslationLine
 HepPID, 58
writeHerwigStream
 HepPDT, 32
writeHerwigTranslation
 HepPID, 58
writeHerwigTranslationLine
 HepPID, 59
writeIsajetStream
 HepPDT, 32
writeIsajetTranslation
 HepPID, 59
writeIsajetTranslationLine
 HepPID, 59
writeLine

examListPythia.cc, 149
writeParticleData
 HepPDT::ParticleDataTable, 90
writeParticleInfo
 HepPDT::ParticleData, 81
 HepPDT::ParticleDataTable, 90
writeParticleNameLine
 HepPID, 59
writeParticleTranslation
 HepPDT::ParticleData, 81
 HepPDT::ParticleDataTable, 90
writePDGStream
 HepPDT, 32
writePDGTranslation
 HepPID, 59
writePDGTranslationLine
 HepPID, 60
writePythiaStream
 HepPDT, 32
writePythiaTranslation
 HepPID, 60
writePythiaTranslationLine
 HepPID, 60
writeQQStream
 HepPDT, 32
writeQQTranslation
 HepPID, 60
writeVersion
 HepPDT, 32
 HepPID, 60

Z
 HepPDT::ParticleID, 100
 HepPID, 61