

# Effects of heavy Higgs bosons

## 1.0

Generated by Doxygen 1.8.6

Wed Apr 1 2015 11:24:43



# Contents

<b>1</b>	<b>Hierarchical Index</b>	<b>1</b>
1.1	Class Hierarchy . . . . .	1
<b>2</b>	<b>Class Index</b>	<b>3</b>
2.1	Class List . . . . .	3
<b>3</b>	<b>Class Documentation</b>	<b>5</b>
3.1	AmplitudePrefactors Struct Reference . . . . .	5
3.1.1	Detailed Description . . . . .	6
3.2	CanvasPtr Struct Reference . . . . .	6
3.3	DoubleCanvasPtr Struct Reference . . . . .	6
3.4	eps_entry Struct Reference . . . . .	6
3.5	FV Class Reference . . . . .	7
3.6	gsl_monte_vegas_state Struct Reference . . . . .	7
3.7	HiggsBoson Class Reference . . . . .	8
3.7.1	Detailed Description . . . . .	9
3.7.2	Member Function Documentation . . . . .	9
3.7.2.1	SetFormFactors . . . . .	9
3.7.2.2	SetPropagator . . . . .	10
3.8	HiggsModel Class Reference . . . . .	10
3.8.1	Detailed Description . . . . .	11
3.8.2	Member Function Documentation . . . . .	11
3.8.2.1	AddBoson . . . . .	11
3.8.2.2	AddPseudoscalar . . . . .	11
3.8.2.3	AddScalar . . . . .	11
3.8.2.4	SetPrefactors . . . . .	12
3.9	HiggsPrefactors Struct Reference . . . . .	12
3.9.1	Detailed Description . . . . .	13
3.10	HistArray Class Reference . . . . .	13
3.11	Integral Class Reference . . . . .	14
3.12	integrand_par Class Reference . . . . .	14
3.13	Integrator Class Reference . . . . .	15

---

3.14 LT Class Reference . . . . .	15
3.15 opt Struct Reference . . . . .	16
3.16 PS_2 Class Reference . . . . .	16
3.17 PS_2_1 Class Reference . . . . .	17
3.18 PS_2_2 Class Reference . . . . .	18
3.19 PS_2_3 Class Reference . . . . .	19
3.20 teebuf Class Reference . . . . .	20
3.21 teestream Class Reference . . . . .	21
3.22 vegas_par Class Reference . . . . .	21
<b>Index</b>	<b>22</b>

# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

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

AmplitudePrefactors . . . . .	5
CanvasPtr . . . . .	6
DoubleCanvasPtr . . . . .	6
eps_entry . . . . .	6
FV . . . . .	7
gsl_monte_vegas_state . . . . .	7
HiggsBoson . . . . .	8
HiggsModel . . . . .	10
HiggsPrefactors . . . . .	12
HistArray . . . . .	13
Integral . . . . .	14
integrand_par . . . . .	14
Integrator . . . . .	15
LT . . . . .	15
opt . . . . .	16
ostream	
teestream . . . . .	21
PS_2 . . . . .	16
PS_2_1 . . . . .	17
PS_2_2 . . . . .	18
PS_2_3 . . . . .	19
streambuf	
teebuf . . . . .	20
vegas_par . . . . .	21



## Chapter 2

# Class Index

### 2.1 Class List

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

AmplitudePrefactors	5
CanvasPtr	6
DoubleCanvasPtr	6
eps_entry	6
FV	7
gsl_monte_vegas_state	7
HiggsBoson	8
HiggsModel	10
HiggsPrefactors	12
HistArray	13
Integral	14
integrand_par	14
Integrator	15
LT	15
opt	16
PS_2	16
PS_2_1	17
PS_2_2	18
PS_2_3	19
teebuf	20
teestream	21
vegas_par	21





## Chapter 3

# Class Documentation

### 3.1 AmplitudePrefactors Struct Reference

```
#include <HiggsModel.h>
```

#### Public Member Functions

- void **Print** (std::ostream &ost)

#### Public Attributes

- double **PREF\_B\_PHI $\times$ PHI** = 0.0
- double **PREF\_B\_PHI $\times$ QCD** = 0.0
- double **PREF\_B\_QCD $\times$ QCD** = 0.0
- double **PREF\_B\_QCD $\times$ QCD\_CF** = 0.0
- double **PREF\_B\_QCD $\times$ QCD\_CA** = 0.0
- double **PREF\_B\_QCD $\times$ QCD\_CFCA** = 0.0
- double **PREF\_V\_PHI $\times$ QCD** = 0.0
- double **PREF\_V\_PHI $\times$ QCD\_CF** = 0.0
- double **PREF\_V\_PHI $\times$ QCD\_CA** = 0.0
- double **PREF\_V\_PHI $\times$ QCD\_CFCA2** = 0.0
- double **PREF\_V\_PHI $\times$ QCD\_Nf** = 0.0
- double **PREF\_V\_PHI $\times$ QCD\_CT** = 0.0
- double **PREF\_V\_PHI $\times$ PHI** = 0.0
- double **PREF\_V\_PHI $\times$ PHI\_CA** = 0.0
- double **PREF\_V\_PHI $\times$ PHI\_CF** = 0.0
- double **PREF\_R\_PHI $\times$ QCD** = 0.0
- double **PREF\_R\_PHI $\times$ QCD\_CF** = 0.0
- double **PREF\_R\_PHI $\times$ QCD\_CA** = 0.0
- double **PREF\_R\_PHI $\times$ QCD\_CFCA2** = 0.0
- double **PREF\_R\_PHI $\times$ PHI** = 0.0
- double **PREF\_R\_PHI $\times$ PHI\_CA** = 0.0
- double **PREF\_R\_PHI $\times$ PHI\_CF** = 0.0
- double **PREF\_UID\_TF** = 0.0
- double **PREF\_UID\_CA** = 0.0
- double **PREF\_UID\_CF** = 0.0

### 3.1.1 Detailed Description

This structure contains the prefactors used in the amplitudes. The prefactors depend on AlphaS and have to be reset whenever it changes (usually only once for each run). This is done via the [HiggsModel](#) class.

See Also

[HiggsModel::SetAmpPrefactors\(\)](#)

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

- inc/HiggsModel.h
- src/HiggsModel.cpp

## 3.2 CanvasPtr Struct Reference

### Public Attributes

- TCanvas \* **c**
- TPad \* **p1\_1**
- TPad \* **p1\_2**

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

- inc/Functions\_Shared.h

## 3.3 DoubleCanvasPtr Struct Reference

### Public Attributes

- [CanvasPtr](#) **c1**
- [CanvasPtr](#) **c2**

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

- inc/Functions\_Shared.h

## 3.4 eps\_entry Struct Reference

### Public Attributes

- int **indices** [4]
- int **sign**

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

- src/Functions\_Shared.cpp

## 3.5 FV Class Reference

### Public Member Functions

- **FV** (double const &a=0.0)
- **FV** (**FV** const &rhs)
- **FV** (**FV** &&rhs) noexcept
- **FV** (std::initializer\_list< double > rhs)
- double & **operator[]** (int const &i)
- double const & **operator[]** (int const &i) const
- **FV** & **operator=** (std::initializer\_list< double > L)
- **FV** & **operator=** (**FV** const &other)
- **FV** & **operator=** (**FV** &&other) noexcept
- **FV** & **operator+=** (**FV** const &other)
- **FV** & **operator-=** (**FV** const &other)
- **FV** & **operator\*=** (double const &a)
- **FV** & **operator/=** (double const &a)

### Protected Attributes

- double **v** [4]

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

- inc/Lorentz.h
- src/Lorentz.cpp

## 3.6 gsl\_monte\_vegas\_state Struct Reference

### Public Attributes

- double **alpha**
- int **mode**
- int **verbose**
- unsigned int **iterations**
- int **stage**
- size\_t **dim**
- size\_t **bins\_max**
- double **vol**
- double \* **delx**
- unsigned int **bins**
- unsigned int **boxes**
- double \* **xi**
- double \* **d**
- double \* **xin**
- double \* **weight**
- double \* **x**
- int \* **bin**
- int \* **box**
- double **bin\_vol**
- double **wgt**
- double **jac**
- double **wtd\_int\_sum**

- double **sum\_wgts**
- double **chi\_sum**
- double **chisq**
- double **result**
- double **sigma**
- unsigned int **it\_start**
- unsigned int **it\_num**
- unsigned int **samples**
- unsigned int **calls\_per\_box**
- FILE \* **ostream**

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

- inc/VEGAS.h

### 3.7 HiggsBoson Class Reference

```
#include <HiggsModel.h>
```

#### Public Member Functions

- **HiggsBoson** (double m, double g, double at, double bt, double ab=0.0, double bb=0.0)
- void **Set** (double m, double g, double at, double bt, double ab=0.0, double bb=0.0)
- void **SetCoupling** (double at, double bt, double ab=0.0, double bb=0.0)
- void **CmpFormFactors** (double &S)
- double const & **M** () const
- double const & **G** () const
- double const & **M2** () const
- double const & **G2** () const
- double const & **At** () const
- double const & **Bt** () const
- double const & **Ab** () const
- double const & **Bb** () const
- double const & **FH** () const
- double const & **FA** () const
- double const & **FH2** () const
- double const & **FA2** () const
- c\_double const & **F\_ggH\_s** () const
- c\_double const & **F\_ggH\_p** () const
- **HiggsBoson** (double const &M, double const &G, double const &Vh, double const &At, double const &Bt, double const &Ab=0.0, double const &Bb=0.0)
- double const & **M** () const
- double const & **G** () const
- double const & **At** () const
- double const & **Bt** () const
- double const & **Ab** () const
- double const & **Bb** () const
- c\_double const & **GetFH0** () const
- c\_double const & **GetFA0** () const
- void **SetFormFactors** (double S, double mt2, double mb2)
- *compute 1-loop form factors for given c.m.e., top- and bottom mass and store values in member variables*
- c\_double const & **GetFs** () const
- c\_double const & **GetFp** () const

- `c_double` const & [GetFH](#) (bool EFF) const  
*returns effective gg-scalar coupling if EFF=true, full 1-loop form factor otherwise*
- `c_double` const & [GetFA](#) (bool EFF) const  
*returns effective gg-pseudoscalar coupling if EFF=true, full 1-loop form factor otherwise*
- void [SetPropagator](#) (double const &S)  
*compute propagator value for given c.m.e. and store values in member variables*
- double const & **GetPropagatorSq** () const
- `c_double` const & **GetPropagator** () const

### Static Public Attributes

- static double **Vh** = 246.0/RunParameters::mScale

### Protected Attributes

- double [d\\_M](#)  
*mass*
- double [d\\_G](#)  
*width*
- double [d\\_M2](#)  
*mass squared*
- double [d\\_G2](#)  
*width squared*
- double [d\\_At](#)  
*reduced scalar-top coupling divided by combined vacuum expectation value*
- double [d\\_Bt](#)  
*reduced pseudoscalar-top coupling divided by combined vacuum expectation value*
- double [d\\_Ab](#)  
*reduced scalar-bottom coupling divided by combined vacuum expectation value*
- double [d\\_Bb](#)  
*reduced pseudoscalar-bottom coupling divided by combined vacuum expectation value*
- double **d\_FH**
- double **d\_FA**
- double **d\_FH2**
- double **d\_FA2**
- `c_double` **d\_F\_ggH\_s**
- `c_double` **d\_F\_ggH\_p**

#### 3.7.1 Detailed Description

This class contains the parameters that describe a single neutral Higgs boson.

See Also

[HiggsModel](#)

#### 3.7.2 Member Function Documentation

##### 3.7.2.1 void HiggsBoson::SetFormFactors ( double S, double mt2, double mb2 )

compute 1-loop form factors for given c.m.e., top- and bottom mass and store values in member variables

## Parameters

$S$	c.m.e.
$mt2$	top-mass squared
$mb2$	bottom-mass squared

3.7.2.2 void HiggsBoson::SetPropagator ( double const &  $S$  )

compute propagator value for given c.m.e. and store values in member variables

## Parameters

$S$	momentum squared in the Higgs propagators
-----	-------------------------------------------

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

- inc/Higgs.h
- inc/HiggsModel.h
- src/Higgs.cpp
- src/HiggsModel.cpp

## 3.8 HiggsModel Class Reference

```
#include <HiggsModel.h>
```

## Public Member Functions

- **HiggsModel** (std::string const &name="noname")
- std::string const & **Name** () const
- double const & **AlphaS** () const
- double const & **AlphaS2** () const
- double const & **MUR** () const
- double const & **MUR2** () const
- double const & **MUF** () const
- double const & **MUF2** () const
- double const & **mt** () const
- double const & **mt2** () const
- double const & **mb** () const
- double const & **mb2** () const
- double const & **VH** () const
- void **SetAlphaS** (double const &val)
 

*Use this member to change AlphaS. It automatically resets the values of the amplitude prefactors.*
- void **SetMUR** (double const &val)
- void **SetMUF** (double const &val)
- void **SetMt** (double const &val)
- void **SetMb** (double const &val)
- void **SetVH** (double const &val)
- void **SetPrefactors** (double const &S, bool EFF)
 

*Reset values of the Higgs prefactors. All Higgs bosons in the vector d\_Bosons will be taken into account.*
- **HiggsPrefactors** const & **GetHiggsPrefactors** () const
- void **SetAmpPrefactors** ()
 

*Reset values of the amplitude prefactors.*
- **AmplitudePrefactors** const & **GetAmpPrefactors** () const

- void [AddBoson](#) (double const &M, double const &G, double const &a\_t=1.0, double const &b\_t=1.0, double const &a\_b=0.0, double const &b\_b=0.0)  
*Add a Higgs boson to the vector d\_Bosons.*
- void [AddScalar](#) (double const &M, double const &G, double const &a\_t=1.0, double const &a\_b=0.0)  
*Add a scalar Higgs boson to the vector d\_Bosons.*
- void [AddPseudoscalar](#) (double const &M, double const &G, double const &b\_t=1.0, double const &b\_b=0.0)  
*Add a pseudoscalar Higgs boson to the vector d\_Bosons.*
- void [PopBoson](#) ()  
*Remove the last Higgs boson added to the vector d\_Bosons.*
- void [Print](#) (std::ostream &ost, double const &mScale)

### 3.8.1 Detailed Description

This class contains all the physical, model specific parameters, i.e. the strong coupling AlphaS, renormalization and factorization scales MUR, MUF, the third generation quark masses mt and mb as well as the combined Higgs vacuum expectation value VH and the individual Higgs boson parameters. It also provides appropriate setter functions. It contains instances of the [HiggsPrefactors](#) and [Amprefactors](#) structures that are needed for the evaluation of the amplitudes. Take care to provide numerical values consistently in the same units.

#### See Also

[AmpPrefactors](#), [HiggsPrefactors](#), [HiggsBoson](#)

### 3.8.2 Member Function Documentation

**3.8.2.1** void [HiggsModel::AddBoson](#) ( double const & *M*, double const & *G*, double const & *a\_t* = 1.0, double const & *b\_t* = 1.0, double const & *a\_b* = 0.0, double const & *b\_b* = 0.0 )

Add a Higgs boson to the vector d\_Bosons.

#### Parameters

<i>M</i>	mass
<i>G</i>	width
<i>a_t</i>	reduced scalar coupling to the top-quark
<i>b_t</i>	reduced pseudoscalar coupling to the top-quark
<i>a_b</i>	reduced scalar coupling to the bottom-quark
<i>b_b</i>	reduced pseudoscalar coupling to the bottom-quark

**3.8.2.2** void [HiggsModel::AddPseudoscalar](#) ( double const & *M*, double const & *G*, double const & *b\_t* = 1.0, double const & *b\_b* = 0.0 )

Add a pseudoscalar Higgs boson to the vector d\_Bosons.

#### Parameters

<i>M</i>	mass
<i>G</i>	width
<i>b_t</i>	reduced pseudoscalar coupling to the top-quark
<i>b_b</i>	reduced pseudoscalar coupling to the bottom-quark

**3.8.2.3** void [HiggsModel::AddScalar](#) ( double const & *M*, double const & *G*, double const & *a\_t* = 1.0, double const & *a\_b* = 0.0 )

Add a scalar Higgs boson to the vector d\_Bosons.

## Parameters

$M$	mass
$G$	width
$a_t$	reduced scalar coupling to the top-quark
$a_b$	reduced scalar coupling to the bottom-quark

3.8.2.4 void HiggsModel::SetPrefactors ( double const &  $S$ , bool  $EFF$  )

Reset values of the Higgs prefactors. All Higgs bosons in the vector `d_Bosons` will be taken into account.

## Parameters

$S$	momentum squared in the Higgs propagators
$EFF$	use effective Higgs-top coupling (large $m_t$ limit) if true. Couplings to the bottom-quark have no effect in this case. The full one-loop form factors are used otherwise.

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

- inc/HiggsModel.h
- src/HiggsModel.cpp

## 3.9 HiggsPrefactors Struct Reference

```
#include <HiggsModel.h>
```

## Public Member Functions

- void **Reset** ()
- void **Print** (std::ostream &ost)

## Public Attributes

- double **At\_fH\_re** = 0.0
- double **At\_fA\_re** = 0.0
- double **Bt\_fH\_re** = 0.0
- double **Bt\_fA\_re** = 0.0
- double **At\_fH\_im** = 0.0
- double **At\_fA\_im** = 0.0
- double **Bt\_fH\_im** = 0.0
- double **Bt\_fA\_im** = 0.0
- double **At2\_fH2\_De** = 0.0
- double **At2\_fA2\_De** = 0.0
- double **Bt2\_fH2\_De** = 0.0
- double **Bt2\_fA2\_De** = 0.0
- double **At\_Bt\_fH2\_De** = 0.0
- double **At\_Bt\_fA2\_De** = 0.0
- double **At\_Bt\_fH2\_DeIM** = 0.0
- double **At\_Bt\_fA2\_DeIM** = 0.0



### 3.9.1 Detailed Description

This structure contains the Higgs specific prefactors, i.e. couplings and propagator denominator. The prefactors depend on the momentum of the Higgs boson and have to be reset whenever it changes (usually for every call of the integrand). This is done via the [HiggsModel](#) class.

See Also

[HiggsModel::SetPrefactors\(\)](#)

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

- inc/HiggsModel.h
- src/HiggsModel.cpp

## 3.10 HistArray Class Reference

### Public Member Functions

- **HistArray** (int nbinsx, double xlow, double xup, int mass\_dim, std::string const &label="", bool SUMW2=false)
- TH1D \* **operator[]** (unsigned i)
- bool **IsActive** (unsigned i)
- void **SetActive** (unsigned i)
- void **Pause** ()
- void **Resume** ()
- void **SetLabel** (std::string const &label)
- const char \* **GetLabel** ()
- void **FillAll** (double const &x, double const &wgt)
- void **FillOne** (unsigned i, double const &x, double const &wgt)
- void **Draw** (const char \*opt="")
- void **Scale** (double c)
- void **Normalize** (bool MASS=false)

### Protected Attributes

- TH1D **d\_histograms** [NHIST]
- unsigned **d\_active**
- unsigned **d\_active\_t**
- std::string **d\_label**
- int **d\_mass\_dim**

### Static Protected Attributes

- static int **d\_ID** = 0

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

- inc/HistArray.h
- src/HistArray.cpp

## 3.11 Integral Class Reference

### Public Member Functions

- **Integral** (size\_t dim)
- **Integral** (size\_t dim, double IntLimitLo[], double IntLimitUp[])
- **Integral** & **operator=** ([Integral](#) const &rhs)
- bool **operator>** (size\_t rhs)
- bool **operator<** (size\_t rhs)
- bool **operator>=** (size\_t rhs)
- bool **operator<=** (size\_t rhs)
- bool **operator==** (size\_t rhs)
- bool **operator!=** (size\_t rhs)
- bool **InRange** (size\_t i)
- double **Lo** (size\_t i)
- double **Up** (size\_t i)
- double \* **Lo** ()
- double \* **Up** ()
- unsigned **GetDim** ()
- void **SetIntegrand** (GSLIFnc Integrand)
- GSLIFnc **GetIntegrand** ()
- void **PrintLimits** (std::ostream &ost=std::cout)
- void **Print** (std::ostream &ost=std::cout)

### Public Attributes

- [vegas\\_par](#) **LastRun**

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

- inc/Integrator.h
- src/Integrator.cpp

## 3.12 integrand\_par Class Reference

### Public Member Functions

- **integrand\_par** (std::ostream &os=std::cout)
- double **cmp\_v\_weight** ()
- void **SetPS** ([PS\\_2](#) \*psn)

### Public Attributes

- double **s\_hadr**
- [PS\\_2](#) \* **ps**
- LHAPDF::PDF \* **pdf**
- [gsl\\_monte\\_vegas\\_state](#) \* **v\_state**
- std::vector< [HistArray](#) \* > \* **distributions**
- bool **collect\_dist**
- bool **tDecay**
- std::ostream & **ost**
- ulong **eval\_flags**

- double **K**
- bool **cleanup**

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

- inc/Integrator.h
- src/Integrator.cpp

## 3.13 Integrator Class Reference

### Public Member Functions

- **Integrator** (std::ostream &ost=std::cout)
- void **SetState** ([gsl\\_monte\\_vegas\\_state](#) \*extGSLState)
- void **DropState** ()
- void **Integrate** ([Integral](#) &integral, [integrand\\_par](#) &ip, [vegas\\_par](#) &vp)
- void **Reset** ()

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

- inc/Integrator.h
- src/Integrator.cpp

## 3.14 LT Class Reference

### Public Member Functions

- void **invert** ()
- void **transpose** ()
- void **apply** ([FV](#) &v)
- void **apply\_G** ([FV](#) &v)
- int **set\_FF** ([FV](#) const &p1, [FV](#) const &p2)
- int **set\_II** ([FV](#) const &K, [FV](#) const &Kb)
- int **set\_boost** ([FV](#) const &P, bool INV=false)
- int **set\_boost\_inv** ([FV](#) const &P)
- int **set\_wigner** ([FV](#) const &P1, [FV](#) const &P2)
- void **print** ()

### Protected Attributes

- double **M** [4][4]

### Static Protected Attributes

- static const double **G** [4] = {1.0,-1.0,-1.0,-1.0}

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

- inc/Lorentz.h
- src/Lorentz.cpp

### 3.15 opt Struct Reference

#### Public Attributes

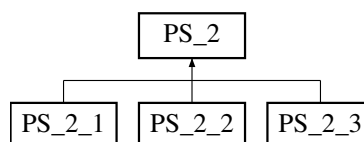
- int **int\_flags**
- int **n\_calls**
- double **ren\_scale**
- double **cme**
- double **mH**
- double **GammaH**
- double **At**
- double **Bt**
- double **Ab**
- double **Bb**
- int **tech\_cut**
- int **precision**
- bool **dist**
- bool **tdecay**
- bool **logfile**
- bool **rootfile**
- int **verb\_level**

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

- inc/Functions\_Shared.h

### 3.16 PS\_2 Class Reference

Inheritance diagram for PS\_2:



#### Public Member Functions

- **PS\_2** (const std::string &nm="")
- **FV** & **p1** ()
- **FV** & **p2** ()
- **FV** const & **p1** () const
- **FV** const & **p2** () const
- **FV** & **P1** ()
- **FV** & **P2** ()
- **FV** const & **P1** () const
- **FV** const & **P2** () const
- double const & **get\_rs** () const
- double **get\_s** () const
- void **set\_rs** (double const &rs)
- void **set\_name** (std::string const &name)
- bool **toggle\_decay** ()

- double const & **get\_wgt** ()
- int **set\_parent** (PS\_2 \*parent)
- void **swap** ()
- virtual double const & **get\_msq** (int i) const =0
- virtual PS\_2 \* **get\_child** (int i) const =0
- virtual int **set\_child** (int i, PS\_2 \*child)=0
- virtual FV const & **get\_k** (int i) const =0
- virtual int **whattype** () const =0
- virtual void **print** () const
- virtual void **FillDistributions** (std::vector< HistArray \* > &dist, int id, double const &wgt)=0

### Protected Member Functions

- void **set\_initial\_state** (double const &s)

### Protected Attributes

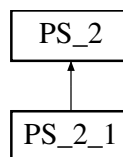
- double **d\_rs**
- std::string **d\_name**
- FV **p** [2]
- FV **P** [2]
- double **d\_wgt**
- bool **d\_decay**
- PS\_2 \* **d\_parent**

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

- inc/PhaseSpace.h
- src/PhaseSpace.cpp

## 3.17 PS\_2\_1 Class Reference

Inheritance diagram for PS\_2\_1:



### Public Member Functions

- **PS\_2\_1** (const std::string &nm="2->2")
- **PS\_2\_1** (double const &msq, const std::string &nm="2->2")
- void **set\_x** (double const &x)
- double const & **get\_x** () const
- FV const & **get\_k** (int i) const
- PS\_2 \* **get\_child** (int i) const
- double const & **get\_msq** (int i) const
- void **set\_msq** (double const &msq)
- FV & **k1** ()

- **FV** const & **k1** () const
- int **set** ()
- int **set\_child** (int i, **PS\_2** \*child)
- int **whattype** () const
- void **print** () const
- void **FillDistributions** (std::vector< **HistArray** \* > &dist, int id, double const &wgt)

### Protected Attributes

- **FV** **k**
- **FV** **S**
- double **d\_msq**
- **PS\_2** \* **d\_child**
- double **d\_x**

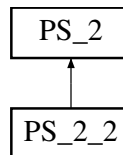
### Additional Inherited Members

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

- inc/PhaseSpace.h
- src/PhaseSpace.cpp

## 3.18 PS\_2\_2 Class Reference

Inheritance diagram for PS\_2\_2:



### Public Member Functions

- **PS\_2\_2** (const std::string &nm="2->2")
- **PS\_2\_2** (double const &m1sq, double const &m2sq, const std::string &nm="2->2")
- **FV** const & **get\_k** (int i) const
- **PS\_2** \* **get\_child** (int i) const
- double const & **get\_msq** (int i) const
- void **set\_msq** (double const &m1sq, double const &m2sq)
- double const & **get\_beta** (int i) const
- double const & **get\_beta** () const
- double const & **get\_y** () const
- double const & **get\_phi** () const
- double const & **get\_t11** () const
- double const & **get\_t12** () const
- double const & **get\_x** () const
- double const & **get\_beta\_y** () const
- void **set\_x** (double const &x)
- double **cmp\_wgt** ()
- **FV** & **k1** ()

- [FV](#) & **k2** ()
- [FV](#) & **s1** ()
- [FV](#) & **s2** ()
- [FV](#) & **s1\_r** ()
- [FV](#) & **s2\_r** ()
- [FV](#) const & **k1** () const
- [FV](#) const & **k2** () const
- [FV](#) const & **s1** () const
- [FV](#) const & **s2** () const
- [FV](#) const & **s1\_r** () const
- [FV](#) const & **s2\_r** () const
- int **boost\_initial\_state** ()
- int **boost\_initial\_state** (double const &x)
- int **boost\_final\_state** ()
- int **set** (double const &rs, double const &y, double const &phi=0.0)
- void **set** ()
- int **set\_child** (int i, [PS\\_2](#) \*child)
- int **whattype** () const
- void **print** () const
- void **FillDistributions** (std::vector< [HistArray](#) \* > &dist, int id, double const &wgt)

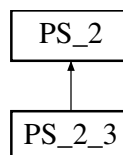
### Additional Inherited Members

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

- inc/PhaseSpace.h
- src/PhaseSpace.cpp

## 3.19 PS\_2\_3 Class Reference

Inheritance diagram for PS\_2\_3:



### Public Member Functions

- **PS\_2\_3** (double const &m1sq, double const &m2sq, double const &m3sq, const std::string &nm="2->3")
- **PS\_2\_3** (const std::string &nm="2->3")
- [FV](#) const & **get\_k** (int i) const
- [PS\\_2](#) \* **get\_child** (int i) const
- double const & **get\_msq** (int i) const
- void **set\_msq** (double const &m1sq, double const &m2sq, double const &m3sq)
- double const & **get\_beta** (int i) const
- double const & **get\_y\_cm** () const
- double const & **get\_M12** () const
- double const & **get\_y\_12** () const
- double const & **get\_phi\_12** () const
- [FV](#) & **k1** ()

- [FV](#) & **k2** ()
- [FV](#) & **k3** ()
- [FV](#) & **p3** ()
- [FV](#) & **s1** ()
- [FV](#) & **s2** ()
- [FV](#) & **s1\_r** ()
- [FV](#) & **s2\_r** ()
- [FV](#) const & **k1** () const
- [FV](#) const & **k2** () const
- [FV](#) const & **k3** () const
- [FV](#) const & **p3** () const
- [FV](#) const & **s1** () const
- [FV](#) const & **s2** () const
- [FV](#) const & **s1\_r** () const
- [FV](#) const & **s2\_r** () const
- int **boost\_to\_parent** ()
- int **set** (double const &rs, double const &y\_cm, double const &phi\_cm, double const &M12, double const &y\_12, double const &phi\_12)
- int **set\_child** (int i, [PS\\_2](#) \*child)
- int **whattype** () const
- void **print** () const
- void **FillDistributions** (std::vector< [HistArray](#) \* > &dist, int id, double const &wgt)

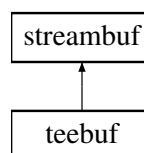
### Additional Inherited Members

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

- inc/PhaseSpace.h
- src/PhaseSpace.cpp

## 3.20 teebuf Class Reference

Inheritance diagram for teebuf:



### Public Member Functions

- **teebuf** (std::streambuf \*sb1, std::streambuf \*sb2)

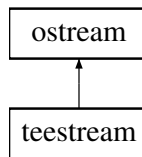
The documentation for this class was generated from the following file:

- inc/Global.h



## 3.21 teestream Class Reference

Inheritance diagram for teestream:



### Public Member Functions

- **teestream** (std::ostream &o1, std::ostream &o2)

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

- inc/Global.h

## 3.22 vegas\_par Class Reference

### Public Member Functions

- void **Print** (std::ostream &ost=std::cout)

### Public Attributes

- int **verbose**
- int **calls**
- int **iterations**
- int **max\_runs**
- int **num\_runs**
- double **chisq\_limit**
- double **chisq**
- bool **do\_warmup**
- bool **grid\_fixed**
- double **result**
- double **error**

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

- inc/Integrator.h
- src/Integrator.cpp

# Index

- AddBoson
  - HiggsModel, [11](#)
- AddPseudoscalar
  - HiggsModel, [11](#)
- AddScalar
  - HiggsModel, [11](#)
- AmplitudePrefactors, [5](#)
- CanvasPtr, [6](#)
- DoubleCanvasPtr, [6](#)
- eps\_entry, [6](#)
- FV, [7](#)
- gsl\_monte\_vegas\_state, [7](#)
- HiggsBoson, [8](#)
  - SetFormFactors, [9](#)
  - SetPropagator, [10](#)
- HiggsModel, [10](#)
  - AddBoson, [11](#)
  - AddPseudoscalar, [11](#)
  - AddScalar, [11](#)
  - SetPrefactors, [12](#)
- HiggsPrefactors, [12](#)
- HistArray, [13](#)
- Integral, [14](#)
- integrand\_par, [14](#)
- Integrator, [15](#)
- LT, [15](#)
- opt, [16](#)
- PS\_2, [16](#)
- PS\_2\_1, [17](#)
- PS\_2\_2, [18](#)
- PS\_2\_3, [19](#)
- SetFormFactors
  - HiggsBoson, [9](#)
- SetPrefactors
  - HiggsModel, [12](#)
- SetPropagator
  - HiggsBoson, [10](#)
- teebuf, [20](#)
- teestream, [21](#)
- vegas\_par, [21](#)