

# SPHERLS Python Scripts

## 1.0

Generated by Doxygen 1.7.6.1

Tue Jan 1 2013 12:54:27



# Contents

<b>1</b>	<b>Class Index</b>	<b>1</b>
1.1	Class List . . . . .	1
<b>2</b>	<b>File Index</b>	<b>3</b>
2.1	File List . . . . .	3
<b>3</b>	<b>Class Documentation</b>	<b>5</b>
3.1	plot_file.Axis Class Reference . . . . .	5
3.1.1	Detailed Description . . . . .	5
3.1.2	Constructor & Destructor Documentation . . . . .	5
3.1.2.1	__init__ . . . . .	5
3.1.3	Member Function Documentation . . . . .	6
3.1.3.1	load . . . . .	6
3.2	plot_profile.Axis Class Reference . . . . .	6
3.2.1	Detailed Description . . . . .	6
3.2.2	Constructor & Destructor Documentation . . . . .	7
3.2.2.1	__init__ . . . . .	7
3.2.3	Member Function Documentation . . . . .	7
3.2.3.1	load . . . . .	7
3.3	plot_file.Curve Class Reference . . . . .	7
3.3.1	Detailed Description . . . . .	8
3.3.2	Constructor & Destructor Documentation . . . . .	8
3.3.2.1	__init__ . . . . .	8
3.3.3	Member Function Documentation . . . . .	9
3.3.3.1	load . . . . .	9
3.4	plot_profile.Curve Class Reference . . . . .	9

3.4.1	Detailed Description	10
3.4.2	Constructor & Destructor Documentation	10
3.4.2.1	__init__	10
3.4.3	Member Function Documentation	10
3.4.3.1	load	10
3.5	datafile.DataFile Class Reference	10
3.5.1	Detailed Description	11
3.5.2	Member Function Documentation	11
3.5.2.1	readFile	11
3.5.2.2	readFileFixed	11
3.5.2.3	readFileUnFixed	11
3.6	plot_file.DataSet Class Reference	11
3.6.1	Detailed Description	12
3.6.2	Constructor & Destructor Documentation	12
3.6.2.1	__init__	12
3.6.3	Member Function Documentation	12
3.6.3.1	getCurve	12
3.6.3.2	load	12
3.7	plot_profile.DataSet Class Reference	12
3.7.1	Detailed Description	13
3.7.2	Constructor & Destructor Documentation	13
3.7.2.1	__init__	13
3.7.3	Member Function Documentation	13
3.7.3.1	getCurve	13
3.7.3.2	load	13
3.8	dump.dump Class Reference	14
3.8.1	Constructor & Destructor Documentation	15
3.8.1.1	__init__	15
3.8.2	Member Function Documentation	15
3.8.2.1	printHeader	15
3.8.2.2	printVar	15
3.8.2.3	read	15
3.8.2.4	readAsciiVar	16
3.8.2.5	readBinaryVar	16

3.8.2.6	<a href="#">readHeader</a>	16
3.8.2.7	<a href="#">readHeaderAscii</a>	16
3.8.2.8	<a href="#">readHeaderBinary</a>	17
3.8.2.9	<a href="#">setVarIDs</a>	17
3.9	<a href="#">eos_interp.eosTable Class Reference</a>	17
3.9.1	<a href="#">Detailed Description</a>	18
3.9.2	<a href="#">Constructor &amp; Destructor Documentation</a>	18
3.9.2.1	<a href="#">__init__</a>	18
3.9.3	<a href="#">Member Function Documentation</a>	18
3.9.3.1	<a href="#">interpolate</a>	18
3.9.3.2	<a href="#">load</a>	18
3.9.3.3	<a href="#">plotLogE</a>	19
3.9.3.4	<a href="#">plotLogP</a>	19
3.9.3.5	<a href="#">write</a>	19
3.10	<a href="#">eos_interp.eosTableManager Class Reference</a>	19
3.10.1	<a href="#">Detailed Description</a>	20
3.10.2	<a href="#">Constructor &amp; Destructor Documentation</a>	20
3.10.2.1	<a href="#">__init__</a>	20
3.10.3	<a href="#">Member Function Documentation</a>	20
3.10.3.1	<a href="#">getTableFromComp</a>	20
3.10.3.2	<a href="#">interpComp</a>	20
3.10.3.3	<a href="#">load</a>	21
3.10.3.4	<a href="#">plotGrid</a>	21
3.11	<a href="#">plot_2DSlices.File2DSlice Class Reference</a>	21
3.11.1	<a href="#">Member Function Documentation</a>	22
3.11.1.1	<a href="#">load</a>	22
3.12	<a href="#">make_hdf.fileSet Class Reference</a>	22
3.12.1	<a href="#">Constructor &amp; Destructor Documentation</a>	22
3.12.1.1	<a href="#">__init__</a>	22
3.12.2	<a href="#">Member Function Documentation</a>	23
3.12.2.1	<a href="#">convertDumpToHDF</a>	23
3.12.2.2	<a href="#">makeHDFFiles</a>	23
3.13	<a href="#">make_hdf2.fileSet Class Reference</a>	23
3.13.1	<a href="#">Constructor &amp; Destructor Documentation</a>	24

3.13.1.1	<code>__init__</code>	24
3.13.2	Member Function Documentation	24
3.13.2.1	<code>convertDumpToHDF</code>	24
3.13.2.2	<code>makeHDFFiles</code>	24
3.14	<code>make_hdf.hdfFile</code> Class Reference	25
3.14.1	Member Function Documentation	25
3.14.1.1	<code>write</code>	25
3.15	<code>eos_interp.interpTable</code> Class Reference	25
3.15.1	Detailed Description	26
3.15.2	Constructor & Destructor Documentation	26
3.15.2.1	<code>__init__</code>	26
3.15.3	Member Function Documentation	27
3.15.3.1	<code>interpolate</code>	27
3.15.3.2	<code>plotLogE</code>	27
3.15.3.3	<code>plotLogK</code>	27
3.15.3.4	<code>plotLogP</code>	28
3.15.3.5	<code>read</code>	28
3.16	<code>eos_interp.interpTableManager</code> Class Reference	28
3.16.1	Constructor & Destructor Documentation	29
3.16.1.1	<code>__init__</code>	29
3.16.2	Member Function Documentation	29
3.16.2.1	<code>createTables</code>	29
3.17	<code>make_hdf.interpVar</code> Class Reference	29
3.18	<code>light_curve.LightCurve</code> Class Reference	30
3.18.1	Member Function Documentation	30
3.18.1.1	<code>calculateCurve</code>	30
3.18.1.2	<code>readBoloCorr</code>	31
3.18.1.3	<code>readProfiles</code>	31
3.18.1.4	<code>write</code>	31
3.19	<code>eos_interp.opacityTable</code> Class Reference	31
3.19.1	Detailed Description	32
3.19.2	Constructor & Destructor Documentation	32
3.19.2.1	<code>__init__</code>	32
3.19.3	Member Function Documentation	32

3.19.3.1	<a href="#">fillInDepNans</a>	32
3.19.3.2	<a href="#">interpolate</a>	32
3.19.3.3	<a href="#">load</a>	33
3.19.3.4	<a href="#">plotLogK</a>	33
3.20	<a href="#">eos_interp.opacityTableManager Class Reference</a>	33
3.20.1	<a href="#">Detailed Description</a>	34
3.20.2	<a href="#">Constructor &amp; Destructor Documentation</a>	34
3.20.2.1	<a href="#">__init__</a>	34
3.20.3	<a href="#">Member Function Documentation</a>	34
3.20.3.1	<a href="#">getTableFromComp</a>	34
3.20.3.2	<a href="#">interpComp</a>	35
3.20.3.3	<a href="#">load</a>	35
3.20.3.4	<a href="#">plotGrids</a>	35
3.21	<a href="#">work_plot.PdVPlotSettings Class Reference</a>	35
3.22	<a href="#">plot_profile.Plot Class Reference</a>	36
3.22.1	<a href="#">Detailed Description</a>	36
3.22.2	<a href="#">Constructor &amp; Destructor Documentation</a>	36
3.22.2.1	<a href="#">__init__</a>	36
3.22.3	<a href="#">Member Function Documentation</a>	37
3.22.3.1	<a href="#">load</a>	37
3.23	<a href="#">plot_file.Plot Class Reference</a>	37
3.23.1	<a href="#">Detailed Description</a>	37
3.23.2	<a href="#">Constructor &amp; Destructor Documentation</a>	37
3.23.2.1	<a href="#">__init__</a>	37
3.23.3	<a href="#">Member Function Documentation</a>	38
3.23.3.1	<a href="#">load</a>	38
3.24	<a href="#">work_plot.Settings Class Reference</a>	38
3.24.1	<a href="#">Constructor &amp; Destructor Documentation</a>	38
3.24.1.1	<a href="#">__init__</a>	38
3.24.2	<a href="#">Member Function Documentation</a>	39
3.24.2.1	<a href="#">parseXML</a>	39
3.25	<a href="#">plot_file.Text Class Reference</a>	39
3.25.1	<a href="#">Detailed Description</a>	39
3.25.2	<a href="#">Constructor &amp; Destructor Documentation</a>	39

3.25.2.1	<code>__init__</code>	39
3.26	<code>make_hdf.variable</code> Class Reference	40
3.27	<code>work_plot.WorkPlotSettings</code> Class Reference	40
<b>4</b>	<b>File Documentation</b>	<b>41</b>
4.1	<code>average_PKE.py</code> File Reference	41
4.1.1	Detailed Description	41
4.2	<code>combine_bins.py</code> File Reference	41
4.2.1	Detailed Description	41
4.3	<code>combine_bins_persistent.py</code> File Reference	41
4.3.1	Detailed Description	41
4.4	<code>compare_sedov_blasts.py</code> File Reference	42
4.4.1	Detailed Description	42
4.5	<code>cp_files.py</code> File Reference	42
4.5.1	Detailed Description	42



# Chapter 1

## Class Index

### 1.1 Class List

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

<a href="#">plot_file.Axis</a>	5
<a href="#">plot_profile.Axis</a>	6
<a href="#">plot_file.Curve</a>	7
<a href="#">plot_profile.Curve</a>	9
<a href="#">datafile.DataFile</a>	10
<a href="#">plot_file.DataSet</a>	11
<a href="#">plot_profile.DataSet</a>	12
<a href="#">dump.dump</a>	14
<a href="#">eos_interp.eosTable</a>	17
<a href="#">eos_interp.eosTableManager</a>	19
<a href="#">plot_2DSlices.File2DSlice</a>	21
<a href="#">make_hdf.fileSet</a>	22
<a href="#">make_hdf2.fileSet</a>	23
<a href="#">make_hdf.hdfFile</a>	25
<a href="#">eos_interp.interpTable</a>	25
<a href="#">eos_interp.interpTableManager</a>	28
<a href="#">make_hdf.interpVar</a>	29
<a href="#">light_curve.LightCurve</a>	30
<a href="#">eos_interp.opacityTable</a>	31
<a href="#">eos_interp.opacityTableManager</a>	33
<a href="#">work_plot.PdVPlotSettings</a>	35
<a href="#">plot_profile.Plot</a>	36
<a href="#">plot_file.Plot</a>	37
<a href="#">work_plot.Settings</a>	38
<a href="#">plot_file.Text</a>	39
<a href="#">make_hdf.variable</a>	40
<a href="#">work_plot.WorkPlotSettings</a>	40



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all documented files with brief descriptions:

<a href="#">average_PKE.py</a>	41
<a href="#">combine_bins.py</a>	41
<a href="#">combine_bins_persistent.py</a>	41
<a href="#">compare_sedov_blasts.py</a>	42
<a href="#">cp_files.py</a>	42



## Chapter 3

# Class Documentation

### 3.1 `plot_file.Axis` Class Reference

#### Public Member Functions

- `def \_\_init\_\_`
- `def load`

#### Public Attributes

- `plots`
- `xlabel`
- `limits`
- `bMinorTics`
- `ticks`
- `grid`
- `plotHeightWeights`

#### 3.1.1 Detailed Description

This class holds all the information needed for a particular x-axis.

#### 3.1.2 Constructor & Destructor Documentation

##### 3.1.2.1 `def plot_file.Axis.__init__( self, element, options )`

This function initializes the axis object.

References `plot_file.Plot.bMinorTics`, `plot_file.Axis.bMinorTics`, `plot_profile.Curve.bTime`, `plot_profile.Axis.bTime`, `plot_file.Plot.grid`, `plot_file.Axis.grid`, `plot_file.Plot.limits`,

plot\_file.Axis.limits, plot\_profile.Axis.period, plot\_file.Axis.plotHeightWeights, plot\_file.-Axis.plots, plot\_file.Plot.ticks, plot\_file.Axis.ticks, and plot\_file.Axis.xlabel.

### 3.1.3 Member Function Documentation

#### 3.1.3.1 `def plot_file.Axis.load( self, files, options )`

This function loads the values needed for the x-axis data from the `fileData` argument

References `plot_file.Axis.plots`.

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

- `plot_file.py`

## 3.2 `plot_profile.Axis` Class Reference

### Public Member Functions

- `def \_\_init\_\_`
- `def load`

### Public Attributes

- `plots`
- `bTime`
- `period`
- `nColumn`
- `xlabel`
- `x`
- `formulaOrig`
- `formula`
- `phase`
- `code`
- `limits`
- `bMinorTics`
- `grid`

### 3.2.1 Detailed Description

This class holds all the information needed for a particular x-axis. An axis can either be either of time, or of some column in the data files.

### 3.2.2 Constructor & Destructor Documentation

#### 3.2.2.1 `def plot_profile.Axis.__init__( self, element, options )`

This function initializes the axis object.

References `plot_file.Plot.bMinorTics`, `plot_profile.Plot.bMinorTics`, `plot_file.Axis.bMinorTics`, `plot_profile.Axis.bMinorTics`, `plot_profile.Curve.bTime`, `plot_profile.Axis.bTime`, `plot_profile.Curve.code`, `plot_profile.Axis.code`, `plot_profile.Axis.formula`, `make_hdf.variable.formula`, `make_hdf.interpVar.formula`, `plot_profile.Curve.formulaOrig`, `plot_profile.Axis.formulaOrig`, `plot_file.Plot.grid`, `plot_profile.Plot.grid`, `plot_file.Axis.grid`, `plot_profile.Axis.grid`, `plot_file.Plot.limits`, `plot_profile.Plot.limits`, `plot_file.Axis.limits`, `plot_profile.Axis.limits`, `plot_profile.Curve.nColumn`, `plot_profile.Axis.nColumn`, `plot_profile.Axis.period`, `plot_profile.Axis.phase`, `plot_file.Axis.plots`, `plot_profile.Axis.plots`, `plot_file.Text.x`, `plot_file.Curve.x`, `plot_profile.Axis.x`, `plot_file.Axis.xlabel`, and `plot_profile.Axis.xlabel`.

### 3.2.3 Member Function Documentation

#### 3.2.3.1 `def plot_profile.Axis.load( self, fileData, options, dataSet, nFileCount )`

This function loads the values needed for the x-axis data from the `fileData` argument

References `plot_profile.Curve.bTime`, `plot_profile.Axis.bTime`, `plot_profile.Curve.code`, `plot_profile.Axis.code`, `plot_profile.Curve.formulaOrig`, `plot_profile.Axis.formulaOrig`, `plot_profile.Curve.nColumn`, `plot_profile.Axis.nColumn`, `plot_profile.Axis.period`, `plot_file.Axis.plots`, `plot_profile.Axis.plots`, `plot_file.Text.x`, `plot_file.Curve.x`, `plot_profile.Axis.x`, `plot_file.Axis.xlabel`, and `plot_profile.Axis.xlabel`.

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

- `plot_profile.py`

## 3.3 plot\_file.Curve Class Reference

### Public Member Functions

- `def \_\_init\_\_`
- `def load`

### Public Attributes

- `nColumnX`
- `nColumnY`
- `nColumnErr`
- `y`

- `x`
- `error`
- `index`
- `formulaOrigY`
- `formulaOrigX`
- `formulaOrigErr`
- `formulaX`
- `formulaY`
- `formulaErr`
- `codeY`
- `codeX`
- `codeErr`
- `style`
- `color`
- `markerfacecolor`
- `markeredgecolor`
- `markersize`
- `linewidth`
- `label`
- `fileReference`
- `nRowShiftErr`
- `nRowShiftX`
- `nRowShiftY`
- `marker`
- `ecolor`
- `elinewidth`
- `capsize`

### 3.3.1 Detailed Description

This class holds all the information for a curve on a plot.

### 3.3.2 Constructor & Destructor Documentation

#### 3.3.2.1 `def plot_file.Curve.__init__( self, element )`

This method initializes a curve object, the type parameter allows checking curve syntax against axis syntax to see if they match.

References `plot_file.Curve.capsize`, `plot_file.Curve.codeErr`, `plot_file.Curve.codeX`, `plot_file.Curve.codeY`, `plot_file.Curve.color`, `plot_file.Curve.ecolor`, `plot_file.Curve.elinewidth`, `plot_file.Curve.error`, `plot_file.Curve.fileReference`, `plot_file.Curve.formulaErr`, `plot_file.Curve.formulaOrigErr`, `plot_file.Curve.formulaOrigX`, `plot_file.Curve.formulaOrigY`, `plot_file.Curve.formulaX`, `plot_file.Curve.formulaY`, `plot_2DSlices.File2DSlice.index`, `plot_file.Curve.index`, `plot_file.Curve.label`, `plot_file.Curve.linewidth`, `plot_file.Curve.marker`, `plot_file.Curve.markeredgecolor`, `plot_file.Curve.markerfacecolor`,



plot\_file.Curve.markersize, plot\_file.Curve.nColumnErr, plot\_file.Curve.nColumnX, plot\_file.Curve.nColumnY, plot\_file.Curve.nRowShiftErr, plot\_file.Curve.nRowShiftX, plot\_file.Curve.nRowShiftY, plot\_file.Curve.style, plot\_file.Text.x, plot\_file.Curve.x, plot\_file.Text.y, and plot\_file.Curve.y.

### 3.3.3 Member Function Documentation

#### 3.3.3.1 def plot\_file.Curve.load( self, files, options )

This method adds a y value and index to the curve for the current fileData.

References plot\_file.Curve.codeErr, plot\_file.Curve.codeX, plot\_file.Curve.codeY, plot\_file.Curve.fileReference, plot\_file.Curve.nColumnErr, plot\_file.Curve.nColumnX, plot\_file.Curve.nColumnY, plot\_file.Curve.nRowShiftErr, plot\_file.Curve.nRowShiftX, and plot\_file.Curve.nRowShiftY.

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

- plot\_file.py

## 3.4 plot\_profile.Curve Class Reference

### Public Member Functions

- def [\\_\\_init\\_\\_](#)
- def [load](#)

### Public Attributes

- nColumn
- zone
- nCurveIDForZoneRef
- y
- index
- bTime
- formulaOrig
- code
- style
- color
- markersize
- linewidth
- testZoneAdjust
- label
- ID
- indexOfLastFileLoad

### 3.4.1 Detailed Description

This class holds all the information for a curve on a plot.

### 3.4.2 Constructor & Destructor Documentation

#### 3.4.2.1 `def plot_profile.Curve.__init__( self, element, type, curveIndex )`

This method initializes a curve object, the `type` parameter allows checking curve syntax against axis syntax to see if they match.

References `plot_profile.Curve.bTime`, `plot_profile.Curve.code`, `plot_profile.Curve.color`, `plot_file.Curve.color`, `plot_profile.Axis.formula`, `make_hdf.variable.formula`, `make_hdf.interpVar.formula`, `plot_profile.Curve.formulaOrig`, `plot_profile.Curve.ID`, `plot_2DSlices.File2DSlice.index`, `plot_profile.Curve.index`, `plot_file.Curve.index`, `plot_profile.Curve.indexOfLastFileLoad`, `plot_profile.Curve.label`, `plot_file.Curve.label`, `plot_profile.Curve.linewidth`, `plot_file.Curve.linewidth`, `plot_profile.Curve.markersize`, `plot_file.Curve.markersize`, `plot_profile.Curve.nColumn`, `plot_profile.Curve.nCurveIDForZoneRef`, `plot_profile.Curve.style`, `plot_file.Curve.style`, `plot_profile.Curve.testZoneAdjust`, `plot_file.Text.y`, `plot_profile.Curve.y`, `plot_file.Curve.y`, and `plot_profile.Curve.zone`.

### 3.4.3 Member Function Documentation

#### 3.4.3.1 `def plot_profile.Curve.load( self, fileData, options, dataSet, nFileCount )`

This method adds a `y` value and index to the curve for the current `fileData`.

References `plot_profile.Curve.bTime`, `plot_profile.Curve.code`, `plot_profile.Curve.formulaOrig`, `plot_profile.Curve.ID`, `plot_profile.Curve.indexOfLastFileLoad`, `plot_profile.Curve.nColumn`, `plot_profile.Curve.testZoneAdjust`, and `plot_profile.Curve.zone`.

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

- `plot_profile.py`

## 3.5 datafile.DataFile Class Reference

### Public Member Functions

- `def setFileSize`
- `def readFile`
- `def readFileFixed`
- `def readFileUnFixed`

### Public Attributes

- `sFileName`

### Static Public Attributes

- **sColumnNames** = None
- **fColumnValues** = None
- **sHeader** = None

### 3.5.1 Detailed Description

A generic class for holding a file consisting of a header and columns of floats

### 3.5.2 Member Function Documentation

#### 3.5.2.1 def datafile.DataFile.readFile( self, sFileName )

a wrapper to determine which readFile function should be used

References `datafile.DataFile.fColumnValues`, `datafile.DataFile.readFileFixed()`, `datafile.DataFile.readFileUnFixed()`, and `datafile.DataFile.sFileName`.

#### 3.5.2.2 def datafile.DataFile.readFileFixed( self, sFileName )

Reads in a file when the size has already been set using `\ref setFileSize`, or by a previous file read using `\ref readFileUnFixed`.

References `datafile.DataFile.fColumnValues`, `datafile.DataFile.sColumnNames`, and `datafile.DataFile.sHeader`.

Referenced by `datafile.DataFile.readFile()`.

#### 3.5.2.3 def datafile.DataFile.readFileUnFixed( self, sFileName )

Reads in a file when the size is not fixed and needs to be determined from the input file being read in

References `datafile.DataFile.fColumnValues`, `datafile.DataFile.sColumnNames`, and `datafile.DataFile.sHeader`.

Referenced by `datafile.DataFile.readFile()`.

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

- `datafile.py`

## 3.6 plot\_file.DataSet Class Reference

### Public Member Functions

- def `__init__`

- def [load](#)
- def [getCurve](#)

### Public Attributes

- **axes**
- **files**

### 3.6.1 Detailed Description

This class holds all the information for a single dataSet, which includes the baseFileName of the dataset, the range of the dataSet (start-end), the times and phases of the files within the range of the dataSet, and the plots made from the dataSet.

### 3.6.2 Constructor & Destructor Documentation

#### 3.6.2.1 `def plot_file.DataSet.__init__( self, element, options )`

Initilizes the dataSet by setting baseFileName, start, end, and intilizing plots from an xml element

References `plot_file.DataSet.axes`, and `plot_file.DataSet.files`.

### 3.6.3 Member Function Documentation

#### 3.6.3.1 `def plot_file.DataSet.getCurve( self, ID )`

Returns a curve object that has ID, ID

References `plot_file.DataSet.axes`.

#### 3.6.3.2 `def plot_file.DataSet.load( self, options )`

Loads the dataSet, this means that it sets, time, phases, and plots data

References `plot_file.DataSet.axes`, and `plot_file.DataSet.files`.

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

- `plot_file.py`

## 3.7 `plot_profile.DataSet` Class Reference

### Public Member Functions

- def [\\_\\_init\\_\\_](#)

- def [load](#)
- def [getCurve](#)

### Public Attributes

- **baseFileName**
- **start**
- **end**
- **axes**
- **nNumFiles**
- **fileIndices**
- **hasNonTimeAxis**
- **eosFile**

### 3.7.1 Detailed Description

This class holds all the information for a single dataSet, which includes the baseFileName of the dataset, the range of the dataSet (start-end), the times and phases of the files within the range of the dataSet, and the plots made from the dataSet.

### 3.7.2 Constructor & Destructor Documentation

#### 3.7.2.1 def plot\_profile.DataSet.\_\_init\_\_( self, element, options )

Initilizes the dataSet by setting baseFileName, start, end, and intilizing plots from an xml element

References `plot_file.DataSet.axes`, `plot_profile.DataSet.axes`, `plot_profile.DataSet.baseFileName`, `plot_profile.DataSet.end`, `plot_2DSlices.File2DSlice.eosFile`, `light_curve.LightCurve.eosFile`, `plot_profile.DataSet.eosFile`, `plot_profile.DataSet.fileIndices`, `plot_profile.DataSet.hasNonTimeAxis`, `light_curve.LightCurve.nNumFiles`, `plot_profile.DataSet.nNumFiles`, and `plot_profile.DataSet.start`.

### 3.7.3 Member Function Documentation

#### 3.7.3.1 def plot\_profile.DataSet.getCurve( self, ID )

Returns a curve object that has ID, ID

References `plot_file.DataSet.axes`, and `plot_profile.DataSet.axes`.

#### 3.7.3.2 def plot\_profile.DataSet.load( self, options )

Loads the dataSet, this means that it sets, time, phases, and plots data

References `plot_file.DataSet.axes`, `plot_profile.DataSet.axes`, `plot_profile.DataSet.baseFileName`, `plot_profile.DataSet.end`, `plot_2DSlices.File2DSlice.eosFile`, `light_curve.LightCurve.eosFile`, `plot_profile.DataSet.eosFile`, `light_curve.LightCurve.nNumFiles`, `plot_profile.DataSet.nNumFiles`, and `plot_profile.DataSet.start`.

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

- `plot_profile.py`

## 3.8 dump.dump Class Reference

### Public Member Functions

- `def __init__`
- `def read`
- `def setVarIDs`
- `def readHeader`
- `def readHeaderBinary`
- `def readHeaderAscii`
- `def readBinaryVar`
- `def readAsciiVar`
- `def printHeader`
- `def printVar`
- `def printDump`

### Public Attributes

- `fileName`
- `f`
- `type`
- `varIDs`
- `varNames`
- `numDims`
- `version`
- `time`
- `timeStepIndex`
- `delta_t_nm1half`
- `delta_t_np1half`
- `alpha`
- `eosStringLen`
- `eosString`
- `gamma`
- `av`
- `avthreshold`
- `globalDims`

- **boundaryConditions**
- **num1DZones**
- **numGhostCells**
- **numVars**
- **varInfo**
- **varSize**
- **vars**

### 3.8.1 Constructor & Destructor Documentation

#### 3.8.1.1 `def dump.dump.__init__( self, fileName )`

Initilizes the dump by reading in a binary file.

References `dump.dump.read()`.

### 3.8.2 Member Function Documentation

#### 3.8.2.1 `def dump.dump.printHeader( self )`

Prints the header of a binary dump file to the standard output.

References `dump.dump.alpha`, `dump.dump.av`, `dump.dump.avthreshold`, `dump.dump.boundaryConditions`, `dump.dump.delta_t_nm1half`, `dump.dump.delta_t_np1half`, `dump.dump.eosString`, `dump.dump.eosStringLen`, `dump.dump.gamma`, `dump.dump.globalDims`, `dump.dump.num1DZones`, `dump.dump.numDims`, `dump.dump.numGhostCells`, `dump.dump.numVars`, `dump.dump.time`, `dump.dump.timeStepIndex`, `dump.dump.type`, `dump.dump.varInfo`, `dump.dump.varSize`, and `dump.dump.version`.

Referenced by `dump.dump.printVar()`.

#### 3.8.2.2 `def dump.dump.printVar( self, var )`

Prints a variable to the standard output.

References `dump.dump.boundaryConditions`, `dump.dump.num1DZones`, `dump.dump.numGhostCells`, `dump.dump.numVars`, `dump.dump.printHeader()`, `dump.dump.printVar()`, `dump.dump.varInfo`, `dump.dump.vars`, and `dump.dump.varSize`.

Referenced by `dump.dump.printVar()`.

#### 3.8.2.3 `def dump.dump.read( self, fileName )`

Reads in a binary dump file.

References `dump.dump.f`, `dump.dump.fileName`, `dump.dump.numVars`, `dump.dump.readAsciiVar()`, `dump.dump.readBinaryVar()`, `dump.dump.readHeader()`, `dump.dump.setVarIDs()`, and `dump.dump.type`.

Referenced by `dump.dump.__init__()`.

#### 3.8.2.4 `def dump.dump.readAsciiVar( self, var )`

Read in a variable from an ascii dump file. Must be called with `var` increasing from 0 to `self.numVars`.

References `dump.dump.boundaryConditions`, `dump.dump.num1DZones`, `dump.dump.numGhostCells`, `dump.dump.varInfo`, and `dump.dump.varSize`.

Referenced by `dump.dump.read()`.

#### 3.8.2.5 `def dump.dump.readBinaryVar( self, var )`

Read in a variable from a binary dump file. Must be called with `var` increasing from 0 to `self.numVars`.

References `dump.dump.boundaryConditions`, `dump.dump.num1DZones`, `dump.dump.numGhostCells`, `dump.dump.varInfo`, and `dump.dump.varSize`.

Referenced by `dump.dump.read()`.

#### 3.8.2.6 `def dump.dump.readHeader( self )`

Reads header information from binary dump file.

References `dump.dump.readHeaderAscii()`, `dump.dump.readHeaderBinary()`, and `dump.dump.type`.

Referenced by `dump.dump.read()`.

#### 3.8.2.7 `def dump.dump.readHeaderAscii( self )`

Reads a header from a ascii file, after the type has been read in.

References `dump.dump.alpha`, `dump.dump.av`, `dump.dump.avthreshold`, `dump.dump.boundaryConditions`, `dump.dump.delta_t_nm1half`, `dump.dump.delta_t_np1half`, `dump.dump.eosString`, `dump.dump.eosStringLen`, `dump.dump.gamma`, `dump.dump.globalDims`, `dump.dump.num1DZones`, `dump.dump.numDims`, `dump.dump.numGhostCells`, `dump.dump.numVars`, `dump.dump.time`, `dump.dump.timeStepIndex`, `dump.dump.varInfo`, `dump.dump.vars`, `dump.dump.varSize`, and `dump.dump.version`.

Referenced by `dump.dump.readHeader()`.



### 3.8.2.8 def dump.dump.readHeaderBinary( self )

Reads a header from a binary file, after the type has been read in.

References dump.dump.alpha, dump.dump.av, dump.dump.avthreshold, dump.dump.boundaryConditions, dump.dump.delta\_t\_nm1half, dump.dump.delta\_t\_np1half, dump.dump.eosString, dump.dump.eosStringLen, dump.dump.gamma, dump.dump.globalDims, dump.dump.num1DZones, dump.dump.numDims, dump.dump.numGhostCells, dump.dump.numVars, dump.dump.time, dump.dump.timeStepIndex, dump.dump.varInfo, dump.dump.vars, dump.dump.varSize, and dump.dump.version.

Referenced by dump.dump.readHeader().

### 3.8.2.9 def dump.dump.setVarIDs( self )

Sets names for the interger values of the grid variables

References dump.dump.gamma, dump.dump.numDims, dump.dump.varIDs, and dump.dump.varNames.

Referenced by dump.dump.read().

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

- dump.py

## 3.9 eos\_interp.eosTable Class Reference

### Public Member Functions

- def [load](#)
- def [write](#)
- def [plotLogE](#)
- def [plotLogP](#)
- def [interpolate](#)
- def [\\_\\_init\\_\\_](#)

### Public Attributes

- **status**
- **X**
- **Z**
- **logT**
- **logD**
- **logP**
- **logE**
- **sFileName**

### 3.9.1 Detailed Description

Holds equation of state data.

### 3.9.2 Constructor & Destructor Documentation

#### 3.9.2.1 `def eos_interp.eosTable.__init__( self, sFileName = None )`

Returns a new instance of eosTable.

If sFileName is set it will use that to set the filename to load the data from.

References eos\_interp.eosTable.logD, eos\_interp.eosTable.logE, eos\_interp.eosTable.logP, eos\_interp.eosTable.logT, datafile.DataFile.sFileName, eos\_interp.eosTable.sFileName, eos\_interp.eosTable.status, light\_curve.LightCurve.temperature, eos\_interp.eosTable.X, and eos\_interp.eosTable.Z.

### 3.9.3 Member Function Documentation

#### 3.9.3.1 `def eos_interp.eosTable.interpolate( self, gridConfig, setExtrapolatedToNan = True )`

Interpolate from self's table to the gridding specified by:

```
logDMin: first (smallest) logD value of grid
logDDel: spacing in logD
numLogD: number of logD grid points
logTMin: first (smallest) logT value of grid
logTDel: spacing in logT
numLogT: number of logT grid points
```

References eos\_interp.eosTable.\_\_fillDepNans(), eos\_interp.eosTable.logD, eos\_interp.eosTable.logE, eos\_interp.eosTable.logT, eos\_interp.eosTable.X, and eos\_interp.eosTable.Z.

#### 3.9.3.2 `def eos_interp.eosTable.load( self )`

Reads in an OPAL equation of state file.

```
It puts the resulting file info into:
self.X: the hydrogen mass fraction
self.Z: the metal mass fraction
self.logD: numpy array of log density grid points [g/cm^3]
self.logT: numpy array of log tempeature grid points [K]
self.logE: numpy array of log energy [ergs/g]
self.logP: numpy array of log pressure [dynes/cm^2]
```

self.logD, self.logT, self.logE, and self.logP are all the same size numpy arrays, empty emelents have logE and logP as nans.

References eos\_interp.eosTable.\_\_fillInDepNans(), eos\_interp.eosTable.\_\_gmass(), eos\_interp.eosTable.logD, eos\_interp.eosTable.logE, eos\_interp.eosTable.logP, eos\_interp.eosTable.logT, datafile.DataFile.sFileName, eos\_interp.eosTable.sFileName, eos\_interp.eosTable.status, eos\_interp.eosTable.X, and eos\_interp.eosTable.Z.

**3.9.3.3** `def eos_interp.eosTable.plotLogE( self, otherTables=None, logDIndexList=None, wireFrame=True )`

Plots LogE

Keywords:

otherTables: a list of other eosTables to include in the plot

logDIndexList: a list of integers corresponding to which densities to plot the tables at

wireFrame: if set to true (the default) and logDIndexList is set to None it will plot a 3D wireframe of logE.

References eos\_interp.eosTable.logD, eos\_interp.eosTable.logE, and eos\_interp.eosTable.logT.

**3.9.3.4** `def eos_interp.eosTable.plotLogP( self, otherTables=None, logDIndexList=None, wireFrame=True )`

Plots LogP

Keywords:

otherTables: a list of other eosTables to include in the plot

logDIndexList: a list of integers corresponding to which densities to plot the tables at

wireFrame: if set to true (the default) and logDIndexList is set to None it will plot a 3D wireframe of logP.

References eos\_interp.eosTable.logD, eos\_interp.eosTable.logP, and eos\_interp.eosTable.logT.

**3.9.3.5** `def eos_interp.eosTable.write( self, args )`

Generic write function that calls either writeToScreen, or writeToFiel depending on if a file name is specified or not.

References eos\_interp.eosTable.\_\_writeToFile(), and eos\_interp.eosTable.\_\_writeToScreen().

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

- eos\_interp.py

## 3.10 eos\_interp.eosTableManager Class Reference

### Public Member Functions

- [def load](#)

- def [interpComp](#)
- def [plotGrid](#)
- def [getTableFromComp](#)
- def [\\_\\_init\\_\\_](#)

### Public Attributes

- **Z**
- **X**
- **eosFileName**
- **eosTables**

### 3.10.1 Detailed Description

Manages equation of state files, including how they are interpolated between.

### 3.10.2 Constructor & Destructor Documentation

#### 3.10.2.1 def eos\_interp.eosTableManager.\_\_init\_\_( self, eosFileName = None )

Returns a new instance of eosTableManager.

if eosFileName is set it will call `__initFromFile` to load settings from a file to initialize the new eosTableManager.

References `eos_interp.eosTableManager.__initFromFile()`, `eos_interp.eosTableManager.__quad()`, `eos_interp.eosTableManager.__quadInterpInZ()`, `eos_interp.eosTableManager.eosFileName`, `eos_interp.eosTableManager.eosTables`, `eos_interp.eosTable.X`, `eos_interp.opacityTable.X`, `eos_interp.opacityTableManager.X`, `eos_interp.eosTableManager.X`, `eos_interp.eosTable.Z`, `eos_interp.opacityTable.Z`, `eos_interp.opacityTableManager.Z`, and `eos_interp.eosTableManager.Z`.

### 3.10.3 Member Function Documentation

#### 3.10.3.1 def eos\_interp.eosTableManager.getTableFromComp( self, X, Z )

Returns a shallow copy of the eos table with matching composition. If none found it returns None.

References `eos_interp.eosTableManager.eosTables`.

#### 3.10.3.2 def eos\_interp.eosTableManager.interpComp( self, X, Z )

Interpolates a set of eos files and opacities to the desired X and Z, and returns an eosManager with this new set of files which can then be interpolated to the desired rho and T's.

References eos\_interp.opacityTableManager.\_\_cubicSplineInX(), eos\_interp.eosTableManager.\_\_cubicSplineInX(), eos\_interp.eosTableManager.\_\_quadInterpInZ(), eos\_interp.eosTable.X, eos\_interp.opacityTable.X, eos\_interp.opacityTableManager.X, eos\_interp.eosTableManager.X, eos\_interp.eosTable.Z, eos\_interp.opacityTable.Z, eos\_interp.opacityTableManager.Z, and eos\_interp.eosTableManager.Z.

#### 3.10.3.3 def eos\_interp.eosTableManager.load( self )

Loads eos files.

Sets the following:

self.Z: a list of Z (metal mass fraction) values of the equation of state files  
 self.X: a list of X (hydrogen mass fraction) values of the equation of state files

References eos\_interp.eosTableManager.eosTables, eos\_interp.eosTable.X, eos\_interp.opacityTable.X, eos\_interp.opacityTableManager.X, eos\_interp.eosTableManager.X, eos\_interp.eosTable.Z, eos\_interp.opacityTable.Z, eos\_interp.opacityTableManager.Z, and eos\_interp.eosTableManager.Z.

#### 3.10.3.4 def eos\_interp.eosTableManager.plotGrid( self, eosIndex )

Plot rho and T points that form the grid

References eos\_interp.eosTableManager.eosTables.

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

- eos\_interp.py

## 3.11 plot\_2DSlices.File2DSlice Class Reference

### Public Member Functions

- def [load](#)

### Public Attributes

- **fileName**
- **index**
- **planeType**
- **time**
- **eosFile**
- **gamma**
- **coordinateNames**
- **coordinates**
- **dataNames**
- **data**

### 3.11.1 Member Function Documentation

#### 3.11.1.1 `def plot_2DSlices.File2DSlice.load( self, fileName )`

sets:  
 fileName, file name of the 2D slice  
 planeType, type of the 2D slice ("rt","rp", "tp")  
 eosFile, file name of the equation of state file, if using a gamma-law gas it is None  
 gamma, value of gamma for a gamma-law gas, if using an equation of state table it is None  
 coordinateNames, Names of the coordinates  
 coordinates, values of the coordinates  
 dataNames, names of the data columns  
 data, the data columns

References `plot_2DSlices.File2DSlice.coordinateNames`, `plot_2DSlices.File2DSlice.coordinates`, `plot_2DSlices.File2DSlice.data`, `make_hdf.hdfFile.data`, `make_hdf2.fileSet.data`, `plot_2DSlices.File2DSlice.dataNames`, `make_hdf2.fileSet.dataNames`, `plot_2DSlices.File2DSlice.eosFile`, `light_curve.LightCurve.eosFile`, `dump.dump.fileName`, `plot_2DSlices.File2DSlice.fileName`, `plot_2DSlices.File2DSlice.gamma`, `dump.dump.gamma`, `plot_2DSlices.File2DSlice.index`, `plot_2DSlices.File2DSlice.planeType`, `plot_2DSlices.File2DSlice.time`, `light_curve.LightCurve.time`, and `dump.dump.time`.

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

- `plot_2DSlices.py`

## 3.12 `make_hdf.fileSet` Class Reference

### Public Member Functions

- `def \_\_init\_\_`
- `def makeHDFFiles`
- `def convertDumpToHDF`

### Public Attributes

- `fileRange`
- `timeFile`
- `variables`
- `interpVars`
- `supportedNodeAttributes`

### 3.12.1 Constructor & Destructor Documentation

#### 3.12.1.1 `def make_hdf.fileSet.__init__( self, element )`

Initialize an fileSet from an xml node

References `make_hdf.fileSet.__checkSupportedNodeAttributes()`, `make_hdf.fileSet.__setSupportedNodeAttributes()`, `plot_profile.DataSet.baseFileName`, `plot_profile.DataSet.end`, `make_hdf.fileSet.fileRange`, `make_hdf.hdfFile.interpVars`, `make_hdf.fileSet.interpVars`, `plot_profile.DataSet.start`, `make_hdf.fileSet.supportedNodeAttributes`, `make_hdf.fileSet.timeFile`, `make_hdf.hdfFile.variables`, and `make_hdf.fileSet.variables`.

### 3.12.2 Member Function Documentation

#### 3.12.2.1 `def make_hdf.fileSet.convertDumpToHDF( self, dump )`

Converts a dump ifle to an hdf file formatted in the way sepcified in the xml configuration file

References `make_hdf.hdfFile.interpVars`, `make_hdf.fileSet.interpVars`, `make_hdf.hdfFile.variables`, and `make_hdf.fileSet.variables`.

Referenced by `make_hdf2.fileSet.makeHDFFiles()`, and `make_hdf.fileSet.makeHDFFiles()`.

#### 3.12.2.2 `def make_hdf.fileSet.makeHDFFiles( self, options )`

Makes HDF files specified by settings

References `plot_profile.DataSet.baseFileName`, `make_hdf.fileSet.convertDumpToHDF()`, `plot_profile.DataSet.end`, and `plot_profile.DataSet.start`.

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

- `make_hdf.py`

## 3.13 make\_hdf2.fileSet Class Reference

### Public Member Functions

- `def __init__`
- `def makeHDFFiles`
- `def convertDumpToHDF`
- `def setAdditionalVariables`
- `def getDataFromDump`

### Public Attributes

- `fileRange`
- `timeFile`
- `frequency`
- `outputPath`

- **radialCutZone**
- **includeBoundaries**
- **numRInterp**
- **supportedNodeAttributes**
- **data**
- **dataMax**
- **dataMin**
- **dataShape**
- **dataNames**
- **dataIDs**

### 3.13.1 Constructor & Destructor Documentation

#### 3.13.1.1 `def make_hdf2.fileSet.__init__( self, element )`

Initialize an fileSet from an xml node

References `make_hdf2.fileSet.__checkSuppotedNodeAttributes()`, `make_hdf.fileSet.___checkSuppotedNodeAttributes()`, `make_hdf2.fileSet.__setSupportedNodeAttributes()`, `make_hdf.fileSet.__setSupportedNodeAttributes()`, `plot_profile.DataSet.baseFileName`, `plot_profile.DataSet.end`, `make_hdf2.fileSet.fileRange`, `make_hdf.fileSet.fileRange`, `make_hdf2.fileSet.frequency`, `light_curve.LightCurve.frequency`, `make_hdf2.fileSet.includeBoundaries`, `make_hdf2.fileSet.numRInterp`, `make_hdf2.fileSet.outputPath`, `make_hdf2.fileSet.radialCutZone`, `plot_profile.DataSet.start`, `make_hdf2.fileSet.supportedNodeAttributes`, `make_hdf.fileSet.supportedNodeAttributes`, `make_hdf2.fileSet.timeFile`, and `make_hdf.fileSet.timeFile`.

### 3.13.2 Member Function Documentation

#### 3.13.2.1 `def make_hdf2.fileSet.convertDumpToHDF( self, dump )`

Converts a dump ifle to an hdf file formatted in the way sepcified in the xml configuration file

References `make_hdf.hdfFile.__interpolateLinearln1DI()`, `make_hdf2.fileSet.___interpolateLinearln1DI()`, `make_hdf.hdfFile.data`, `make_hdf2.fileSet.data`, `make_hdf2.fileSet.dataIDs`, `make_hdf.hdfFile.dataMax`, `make_hdf2.fileSet.dataMax`, `make_hdf.hdfFile.dataMin`, `make_hdf2.fileSet.dataMin`, `make_hdf2.fileSet.dataNames`, `make_hdf2.fileSet.dataShape`, `make_hdf2.fileSet.getDataFromDump()`, `make_hdf2.fileSet.includeBoundaries`, `make_hdf2.fileSet.numRInterp`, `make_hdf2.fileSet.outputPath`, `make_hdf2.fileSet.radialCutZone`, and `make_hdf2.fileSet.setAdditionalVariables()`.

Referenced by `make_hdf2.fileSet.makeHDFFiles()`.

#### 3.13.2.2 `def make_hdf2.fileSet.makeHDFFiles( self, options )`

Makes HDF files specified by settings



References `plot_profile.DataSet.baseFileName`, `make_hdf2.fileSet.convertDumpToHDF()`, `make_hdf.fileSet.convertDumpToHDF()`, `plot_profile.DataSet.end`, `make_hdf2.fileSet.frequency`, `light_curve.LightCurve.frequency`, `plot_profile.DataSet.start`, `make_hdf2.fileSet.timeFile`, and `make_hdf.fileSet.timeFile`.

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

- `make_hdf2.py`

## 3.14 make\_hdf.hdfFile Class Reference

### Public Member Functions

- `def __init__`
- `def printVarToScreen`
- `def write`

### Public Attributes

- `variables`
- `interpVars`
- `varNames`
- `varIDs`
- `data`
- `dataMax`
- `dataMin`

### 3.14.1 Member Function Documentation

#### 3.14.1.1 `def make_hdf.hdfFile.write( self )`

this function writes the data specified in the configuration file to a new hdf file. It does this by interpolating where nessacary to get data at the right location

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

- `make_hdf.py`

## 3.15 eos\_interp.interpTable Class Reference

### Public Member Functions

- `def interpolate`
- `def read`

- def [plotLogE](#)
- def [plotLogP](#)
- def [plotLogK](#)
- def [\\_\\_init\\_\\_](#)

### Public Attributes

- **eosAtNewComp**
- **opacityAtNewComp**
- **eosTable**
- **opacityTable**
- **sFileName**
- **numLogR**
- **X**
- **Z**
- **gridConfig**
- **logD**
- **logT**
- **logP**
- **logE**
- **logK**
- **outputFile**
- **plot**
- **setNans**

### 3.15.1 Detailed Description

This class reads in and holds data for an equations of state and opacities from a file format in the same was as read to and written by the class defined in eos.h, and implemented in eos.

### 3.15.2 Constructor & Destructor Documentation

#### 3.15.2.1 `def eos_interp.interpTable.__init__( self, tableElement = None )`

Reads in an interpolation table info from from the xml element tableElement.

References `eos_interp.interpTable.eosTable`, `eos_interp.interpTable.gridConfig`, `eos_interp.interpTable.opacityTable`, `eos_interp.interpTable.outputFile`, `eos_interp.interpTable.plot`, `eos_interp.interpTable.setNans`, `eos_interp.eosTable.X`, `eos_interp.opacityTable.X`, `eos_interp.opacityTableManager.X`, `eos_interp.eosTableManager.X`, `eos_interp.interpTable.X`, `eos_interp.eosTable.Z`, `eos_interp.opacityTable.Z`, `eos_interp.opacityTableManager.Z`, `eos_interp.eosTableManager.Z`, and `eos_interp.interpTable.Z`.

### 3.15.3 Member Function Documentation

**3.15.3.1** `def eos_interp.interpTable.interpolate( self, eosSet, opacitySet, withoutNans=None )`

creates the interpolated table and writes it out

References eos\_interp.interpTable.\_\_writeCompleteEOS(), eos\_interp.interpTable.eosAtNewComp, eos\_interp.interpTable.eosTable, eos\_interp.interpTable.gridConfig, eos\_interp.interpTable.opacityAtNewComp, eos\_interp.interpTable.opacityTable, eos\_interp.interpTable.outputFile, eos\_interp.interpTable.plot, eos\_interp.interpTable.setNans, eos\_interp.eosTable.X, eos\_interp.opacityTable.X, eos\_interp.opacityTableManager.X, eos\_interp.eosTableManager.X, eos\_interp.interpTable.X, eos\_interp.eosTable.Z, eos\_interp.opacityTable.Z, eos\_interp.opacityTableManager.Z, eos\_interp.eosTableManager.Z, and eos\_interp.interpTable.Z.

**3.15.3.2** `def eos_interp.interpTable.plotLogE( self, otherTables=None, logDIndexList=None, logDRangeList=None, wireFrame=True, rstride=1, cstride=1, outputfile=None )`

Plots LogE

Keywords:

otherTables: a list of other eosTables to include in the plot

logDIndexList: a list of integers corresponding to which densities to plot the tables at

wireFrame: if set to true (the default) and logDIndexList is set to None it will plot a 3D wireframe of logE.

References eos\_interp.eosTable.logD, eos\_interp.interpTable.logD, eos\_interp.eosTable.logE, eos\_interp.interpTable.logE, eos\_interp.eosTable.logT, eos\_interp.opacityTable.logT, eos\_interp.interpTable.logT, datafile.DataFile.sFileName, eos\_interp.eosTable.sFileName, eos\_interp.opacityTable.sFileName, and eos\_interp.interpTable.sFileName.

**3.15.3.3** `def eos_interp.interpTable.plotLogK( self, otherTables=None, logDIndexList=None, logDRangeList=None, wireFrame=True, outputfile=None )`

Plots opacity

Keywords:

otherTables: a list of opacity tables to also be plotted

logDIndex: a list of integers used to indicate a specific logR index to plot 2D line plots at.

References eos\_interp.eosTable.logD, eos\_interp.interpTable.logD, eos\_interp.opacityTable.logK, eos\_interp.interpTable.logK, eos\_interp.eosTable.logT, eos\_interp.opacityTable.logT, eos\_interp.interpTable.logT, datafile.DataFile.sFileName, eos\_interp.eosTable.sFileName, eos\_interp.opacityTable.sFileName, and eos\_interp.interpTable.sFileName.

```
3.15.3.4 def eos_interp.interpTable.plotLogP( self, otherTables = None,
      logDIndexList = None, logDRangeList = None, wireFrame = True, outputfile =
      None )
```

Plots LogP

Keywords:

otherTables: a list of other eosTables to include in the plot

logDIndexList: a list of integers corresponding to which densities to plot the tables at

wireFrame: if set to true (the default) and logDIndexList is set to None it will plot a 3D wireframe of logP.

References eos\_interp.eosTable.logD, eos\_interp.interpTable.logD, eos\_interp.eosTable.logP, eos\_interp.interpTable.logP, eos\_interp.eosTable.logT, eos\_interp.opacityTable.logT, eos\_interp.interpTable.logT, datafile.DataFile.sFileName, eos\_interp.eosTable.sFileName, eos\_interp.opacityTable.sFileName, and eos\_interp.interpTable.sFileName.

```
3.15.3.5 def eos_interp.interpTable.read( self, sFilename )
```

Reads in an interpolated table

References eos\_interp.interpTable.gridConfig, eos\_interp.eosTable.logD, eos\_interp.interpTable.logD, eos\_interp.eosTable.logE, eos\_interp.interpTable.logE, eos\_interp.opacityTable.logK, eos\_interp.interpTable.logK, eos\_interp.eosTable.logP, eos\_interp.interpTable.logP, eos\_interp.eosTable.logT, eos\_interp.opacityTable.logT, eos\_interp.interpTable.logT, eos\_interp.interpTable.numLogR, datafile.DataFile.sFileName, eos\_interp.eosTable.sFileName, eos\_interp.opacityTable.sFileName, eos\_interp.interpTable.sFileName, eos\_interp.eosTable.X, eos\_interp.opacityTable.X, eos\_interp.opacityTableManager.X, eos\_interp.eosTableManager.X, eos\_interp.interpTable.X, eos\_interp.eosTable.Z, eos\_interp.opacityTable.Z, eos\_interp.opacityTableManager.Z, eos\_interp.eosTableManager.Z, and eos\_interp.interpTable.Z.

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

- eos\_interp.py

## 3.16 eos\_interp.interpTableManager Class Reference

### Public Member Functions

- def [createTables](#)
- def [\\_\\_init\\_\\_](#)

### Public Attributes

- [configFile](#)
- [eosSet](#)

- **opacitySet**
- **tables**

### 3.16.1 Constructor & Destructor Documentation

3.16.1.1 `def eos_interp.interpTableManager.__init__( self, configFile=None )`

Initializes interpTableManager from the given configuration file.

References eos\_interp.interpTableManager.\_\_readInterpTableConfigs(), eos\_interp.interpTableManager.configFile, eos\_interp.interpTableManager.eosSet, eos\_interp.interpTableManager.opacitySet, and eos\_interp.interpTableManager.tables.

### 3.16.2 Member Function Documentation

3.16.2.1 `def eos_interp.interpTableManager.createTables( self, withoutNans=None )`

Creates interpolated tables and write them out.

References eos\_interp.interpTableManager.eosSet, eos\_interp.interpTableManager.opacitySet, and eos\_interp.interpTableManager.tables.

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

- eos\_interp.py

## 3.17 make\_hdf.interpVar Class Reference

### Public Member Functions

- `def __init__`

### Public Attributes

- **numPoints**
- **name**
- **formula**

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

- make\_hdf.py

## 3.18 light\_curve.LightCurve Class Reference

### Public Member Functions

- def `__init__`
- def `create`
- def `readProfiles`
- def `readBoloCorr`
- def `calculateCurve`
- def `write`

### Public Attributes

- `boloCorrFile`
- `columnBC`
- `withAcceleration`
- `inputFileRange`
- `frequency`
- `eosFile`
- `zonesFromSurf`
- `outputFile`
- `nNumFiles`
- `luminosity`
- `temperature`
- `interiorMass`
- `time`
- `gridVelocity`
- `radius`
- `TMin`
- `loggMin`
- `TDel`
- `loggDel`
- `numLogg`
- `numT`
- `BC`

### 3.18.1 Member Function Documentation

#### 3.18.1.1 `def light_curve.LightCurve.calculateCurve( self )`

Creates the light curve by converting luminosity to bolometric magnitude and then applying a bolometric correction and returns a 2D list of times and light curve magnitudes.

References `light_curve.LightCurve.BC`, `light_curve.LightCurve.gridVelocity`, `light_curve.LightCurve.interiorMass`, `light_curve.LightCurve.loggDel`, `light_curve.LightCurve.loggMin`, `light_curve.LightCurve.luminosity`, `light_curve.LightCurve.numLogg`, `light_curve.LightCurve.numT`, `light_curve.LightCurve.radius`, `light_curve.LightCurve.TDel`, `light_curve.LightCurve.temperature`, `light_curve.LightCurve.time`, `dump.dump.time`, `light_curve.LightCurve.TMin`, and `light_curve.LightCurve.withAcceleration`.

#### 3.18.1.2 `def light_curve.LightCurve.readBoloCorr( self )`

Reads in the bolometric correction table

References `light_curve.LightCurve.BC`, `light_curve.LightCurve.boloCorrFile`, `light_curve.LightCurve.columnBC`, `light_curve.LightCurve.loggDel`, `light_curve.LightCurve.loggMin`, `light_curve.LightCurve.numLogg`, `light_curve.LightCurve.numT`, `light_curve.LightCurve.TDel`, and `light_curve.LightCurve.TMin`.

#### 3.18.1.3 `def light_curve.LightCurve.readProfiles( self, options )`

Reads the needed data to create the light curve from the radial profile files

References `plot_profile.DataSet.baseFileName`, `plot_profile.DataSet.end`, `light_curve.LightCurve.eosFile`, `light_curve.LightCurve.frequency`, `light_curve.LightCurve.gridVelocity`, `light_curve.LightCurve.interiorMass`, `light_curve.LightCurve.luminosity`, `light_curve.LightCurve.nNumFiles`, `light_curve.LightCurve.radius`, `plot_profile.DataSet.start`, `light_curve.LightCurve.temperature`, `light_curve.LightCurve.time`, `dump.dump.time`, and `light_curve.LightCurve.zonesFromSurf`.

#### 3.18.1.4 `def light_curve.LightCurve.write( self, curve )`

Writes out the light curve to the specified output file.

References `light_curve.LightCurve.outputFile`, and `eos_interp.interpTable.outputFile`.

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

- `light_curve.py`

## 3.19 eos\_interp.opacityTable Class Reference

### Public Member Functions

- `def load`
- `def plotLogK`
- `def interpolate`
- `def __init__`
- `def fillInDepNans`

## Public Attributes

- **multitableFile**
- **X**
- **Z**
- **sFileName**
- **logT**
- **logR**
- **logK**

### 3.19.1 Detailed Description

Holds opacity table data.

Initialize with a composition (X,Z), file name and weather the file name contains multiple.

### 3.19.2 Constructor & Destructor Documentation

**3.19.2.1** `def eos_interp.opacityTable.__init__( self, X=None, Z=None, sFileName=None, multitableFile=None )`

Initializes the opacity object.

```
sets:
self.X: the hydrogen mass fraction
self.Z: the metal mass fraction
self.sFileName: the file name to load the table from
self.multitableFile: weather or not the file has more than one table in it
```

References eos\_interp.opacityTable.logK, eos\_interp.opacityTable.logR, eos\_interp.eosTable.logT, eos\_interp.opacityTable.logT, eos\_interp.opacityTable.multitableFile, datafile.DataFile.sFileName, eos\_interp.eosTable.sFileName, eos\_interp.opacityTable.sFileName, eos\_interp.eosTable.X, eos\_interp.opacityTable.X, eos\_interp.eosTable.Z, and eos\_interp.opacityTable.Z.

### 3.19.3 Member Function Documentation

**3.19.3.1** `def eos_interp.opacityTable.fillInDepNans( self )`

Fills in logR and logT values to make a rectangular grid

References eos\_interp.opacityTable.logR, eos\_interp.eosTable.logT, and eos\_interp.opacityTable.logT.

**3.19.3.2** `def eos_interp.opacityTable.interpolate( self, gridConfig, setExtrapolatedToNan=True )`

Interpolate from self's table to the griding specified by:



parameters:

- logDMin: first (smallest) logD value of grid
- logDDel: spacing in logD
- numLogD: number of logD grid points
- logTMin: first (smallest) logT value of grid
- logTDel: spacing in logT
- numLogT: number of logT grid points

keyword:

- setExtrapolatedToNan: controls weather extrapolated points are set to nans (default is True)

returns:

an opacity table interpolated to the specified grid. In addition to the regular members of an opacity table logD is also included.

References eos\_interp.eosTable.\_\_fillDepNans(), eos\_interp.opacityTable.\_\_fillDepNans(), eos\_interp.opacityTable.logK, eos\_interp.opacityTable.logR, eos\_interp.eosTable.logT, eos\_interp.opacityTable.logT, eos\_interp.eosTable.X, eos\_interp.opacityTable.X, eos\_interp.eosTable.Z, and eos\_interp.opacityTable.Z.

### 3.19.3.3 def eos\_interp.opacityTable.load( self )

Load from a file an opacity table for composition of the current opacity object. It does this by advancing a file until the composition is matched and then calls \_\_loadTableFromFile to load the logR, logT, and logK values.

References eos\_interp.opacityTable.\_\_loadTableFromFile(), eos\_interp.opacityTable.\_multitableFile, datafile.DataFile.sFileName, eos\_interp.eosTable.sFileName, eos\_interp.opacityTable.sFileName, eos\_interp.eosTable.X, eos\_interp.opacityTable.X, eos\_interp.eosTable.Z, and eos\_interp.opacityTable.Z.

### 3.19.3.4 def eos\_interp.opacityTable.plotLogK( self, otherTables=None, logRIndex=None, wireFrame=True )

Plots opacity

Keywords:

- otherTables: a list of opacity tables to also be plotted
- logRIndex: a list of integers used to indicate a specific logR index to plot 2D line plots at.

References eos\_interp.opacityTable.logK, eos\_interp.opacityTable.logR, eos\_interp.eosTable.logT, and eos\_interp.opacityTable.logT.

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

- eos\_interp.py

## 3.20 eos\_interp.opacityTableManager Class Reference

## Public Member Functions

- def [load](#)
- def [interpComp](#)
- def [plotGrids](#)
- def [getTableFromComp](#)
- def [\\_\\_init\\_\\_](#)

## Public Attributes

- **opacityConfigFileName**
- **opacityFileNames**
- **opacityTables**
- **Z**
- **X**

### 3.20.1 Detailed Description

Manages opacity files, including how they are interpolated between in composition.

### 3.20.2 Constructor & Destructor Documentation

**3.20.2.1** `def eos_interp.opacityTableManager.__init__( self, opacityConfigFile = None )`

Creates a new instance of opacityTableManager.

If opacityConfigFile is set it will try to parse it for xml settings to get all the file names of the opacity files to include in the opacityTableManager.

References eos\_interp.opacityTableManager.\_\_getCompositions(), eos\_interp.opacityTableManager.\_\_merge2files(), eos\_interp.opacityTableManager.opacityConfigFileName, eos\_interp.opacityTableManager.opacityFileNames, eos\_interp.opacityTableManager.opacityTables, eos\_interp.eosTable.X, eos\_interp.opacityTable.X, eos\_interp.opacityTableManager.X, eos\_interp.eosTable.Z, eos\_interp.opacityTable.Z, and eos\_interp.opacityTableManager.Z.

### 3.20.3 Member Function Documentation

**3.20.3.1** `def eos_interp.opacityTableManager.getTableFromComp( self, X, Z )`

Returns a shallow copy of the opacity table with matching composition.

References eos\_interp.opacityTableManager.opacityTables.

**3.20.3.2 def eos\_interp.opacityTableManager.interpComp( self, X, Z )**

Interpolates a set of opacity files to the desired X and Z, and returns an the interpolated opacityTable.

Parameters:

X: hydrogen mass fraction

Z: metal mass fraction

References eos\_interp.opacityTableManager.\_\_bicubicSplineInXZ(), eos\_interp.eosTable.X, eos\_interp.opacityTable.X, eos\_interp.opacityTableManager.X, eos\_interp.eosTable.Z, eos\_interp.opacityTable.Z, and eos\_interp.opacityTableManager.Z.

**3.20.3.3 def eos\_interp.opacityTableManager.load( self )**

Loads opacity files and merge files at duplicate compositions (i.e. merges low and high temperature opacity tables).

Sets the following:

self.X: list of hydrogen mass fractions covered by opacity tables

self.Z: list of metal mass fractions covered by opacity tables

References eos\_interp.opacityTableManager.\_\_merge(), eos\_interp.opacityTableManager.\_\_setCompLists(), and eos\_interp.opacityTableManager.opacityTables.

**3.20.3.4 def eos\_interp.opacityTableManager.plotGrids( self, opacityIndex )**

Plot LogR and LogT points that form the opacity grid.

Parameters:

opacityIndex: a list of integers used to select which opacity tables will be plotted

References eos\_interp.opacityTableManager.opacityTables.

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

- eos\_interp.py

**3.21 work\_plot.PdVPlotSettings Class Reference****Public Member Functions**

- def **\_\_init\_\_**
- def **parseXML**

**Public Attributes**

- **startZone**

- **points**
- **grid**
- **format**
- **outputFile**
- **lines**
- **show**

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

- `work_plot.py`

## 3.22 plot\_profile.Plot Class Reference

### Public Member Functions

- `def \_\_init\_\_`
- `def load`

### Public Attributes

- **ylabel**
- **curves**
- **limits**
- **grid**
- **bMinorTics**
- **legendloc**

### 3.22.1 Detailed Description

This class holds all the information for a single plot, namely the list of curves for that pl

### 3.22.2 Constructor & Destructor Documentation

#### 3.22.2.1 `def plot_profile.Plot.__init__( self, element, type )`

This method initlizes the plot object

References `plot_file.Plot.bMinorTics`, `plot_profile.Plot.bMinorTics`, `plot_file.Axis.bMinorTics`, `plot_file.Plot.curves`, `plot_profile.Plot.curves`, `plot_file.Plot.grid`, `plot_profile.Plot.grid`, `plot_file.Axis.grid`, `plot_file.Plot.legendloc`, `plot_profile.Plot.legendloc`, `plot_file.Plot.limits`, `plot_profile.Plot.limits`, `plot_file.Axis.limits`, `plot_file.Plot.ylabel`, and `plot_profile.Plot.ylabel`.

### 3.22.3 Member Function Documentation

#### 3.22.3.1 def plot\_profile.Plot.load( self, fileData, options, dataSet, nFileCount )

loads the data for a plot, y-data is stored in the curves, and sets the ylabel from the first file read in

References plot\_file.Plot.curves, plot\_profile.Plot.curves, plot\_file.Plot.ylabel, and plot\_profile.Plot.ylabel.

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

- plot\_profile.py

## 3.23 plot\_file.Plot Class Reference

### Public Member Functions

- def [\\_\\_init\\_\\_](#)
- def [load](#)

### Public Attributes

- ylabel
- curves
- texts
- limits
- grid
- bMinorTics
- legendloc
- numpoints
- weightHeight
- ticks

#### 3.23.1 Detailed Description

This class holds all the information for a single plot, namely the list of curves for that plot.

#### 3.23.2 Constructor & Destructor Documentation

##### 3.23.2.1 def plot\_file.Plot.\_\_init\_\_( self, element )

This method initlizes the plot object

References plot\_file.Plot.bMinorTics, plot\_file.Plot.curves, plot\_file.Plot.grid, plot\_file.Plot.legendloc, plot\_file.Plot.limits, plot\_file.Plot.numpoints, plot\_file.Plot.texts, plot\_file.Plot.ticks, plot\_file.Plot.weightHeight, and plot\_file.Plot.ylabel.

### 3.23.3 Member Function Documentation

#### 3.23.3.1 `def plot_file.Plot.load( self, files, options )`

loads the data for a plot, y-data is stored in the curves, and sets the ylabel from the first file read in

References `plot_file.Plot.curves`.

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

- `plot_file.py`

## 3.24 `work_plot.Settings` Class Reference

### Public Member Functions

- `def \_\_init\_\_`
- `def parseXML`

### Public Attributes

- `pColumn`
- `pColumnHeader`
- `tColumn`
- `tColumnHeader`
- `rhoColumn`
- `rhoColumnHeader`
- `QColumn`
- `QColumnHeader`
- `deltaMColumn`
- `deltaMColumnHeader`
- `AV`
- `outputFile`
- `plotPdVCurves`
- `files`
- `workPlotSettings`
- `PdVPlotSettings`

### 3.24.1 Constructor & Destructor Documentation

#### 3.24.1.1 `def work_plot.Settings.__init__( self, oldColumns = False )`

Initialize settings

References `work_plot.Settings.AV`, `work_plot.Settings.deltaMColumn`, `work_plot.Settings.deltaMColumnHeader`, `work_plot.WorkPlotSettings.outputFile`, `light_curve.LightCurve.outputFile`, `work_plot.PdVPlotSettings.outputFile`, `work_plot.Settings.outputFile`, `eos_interp.interpTable.outputFile`, `work_plot.Settings.pColumn`, `work_plot.Settings.pColumnHeader`, `work_plot.WorkPlotSettings.plotPdVCurves`, `work_plot.Settings.plotPdVCurves`, `work_plot.Settings.QColumn`, `work_plot.Settings.QColumnHeader`, `work_plot.Settings.rhoColumn`, `work_plot.Settings.rhoColumnHeader`, `work_plot.Settings.tColumn`, and `work_plot.Settings.tColumnHeader`.

### 3.24.2 Member Function Documentation

#### 3.24.2.1 `def work_plot.Settings.parseXML( self, fileName )`

Get user settings from XML file

References `work_plot.Settings.AV`, `work_plot.Settings.files`, `plot_file.DataSet.files`, `work_plot.Settings.PdVPlotSettings`, `work_plot.WorkPlotSettings.plotPdVCurves`, `work_plot.Settings.plotPdVCurves`, and `work_plot.Settings.workPlotSettings`.

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

- `work_plot.py`

## 3.25 plot\_file.Text Class Reference

### Public Member Functions

- `def \_\_init\_\_`

### Public Attributes

- `x`
- `y`
- `text`

#### 3.25.1 Detailed Description

This class holds informatin for a text object on a plot.

#### 3.25.2 Constructor & Destructor Documentation

##### 3.25.2.1 `def plot_file.Text.__init__( self, element )`

This method initializest a text object from an xml element

References `plot_file.Text.text`, `plot_file.Text.x`, and `plot_file.Text.y`.

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

- `plot_file.py`

### 3.26 `make_hdf.variable` Class Reference

#### Public Member Functions

- `def __init__`

#### Public Attributes

- `indep`
- `fillValue`
- `formula`

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

- `make_hdf.py`

### 3.27 `work_plot.WorkPlotSettings` Class Reference

#### Public Member Functions

- `def __init__`
- `def parseXML`

#### Public Attributes

- `minTemp`
- `ylim`
- `grid`
- `points`
- `lines`
- `plotPdVCurves`
- `startZone`
- `temperatureProfileFile`
- `format`
- `outputFile`

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

- `work_plot.py`



## Chapter 4

# File Documentation

### 4.1 `average_PKE.py` File Reference

#### Functions

- `def average_PKE.parseOptions`
- `def average_PKE.main`
- `def average_PKE.averagePKE`

#### 4.1.1 Detailed Description

### 4.2 `combine_bins.py` File Reference

#### Functions

- `def combine_bins.main`
- `def combine_bins.combine_bin_files`

#### 4.2.1 Detailed Description

### 4.3 `combine_bins_persistent.py` File Reference

#### Functions

- `def combine_bins_persistent.main`

#### 4.3.1 Detailed Description

## 4.4 `compare_sedov_blasts.py` File Reference

### Functions

- `def compare_sedov_blasts.main`

#### 4.4.1 Detailed Description

## 4.5 `cp_files.py` File Reference

### Functions

- `def cp_files.main`
- `def cp_files.cp_files`

#### 4.5.1 Detailed Description

# Index

## `__init__`

- `dump::dump`, 15
- `eos_interp::eosTable`, 18
- `eos_interp::eosTableManager`, 20
- `eos_interp::interpTable`, 26
- `eos_interp::interpTableManager`, 29
- `eos_interp::opacityTable`, 32
- `eos_interp::opacityTableManager`, 34
- `make_hdf2::fileSet`, 24
- `make_hdf::fileSet`, 22
- `plot_file::Axis`, 5
- `plot_file::Curve`, 8
- `plot_file::DataSet`, 12
- `plot_file::Plot`, 37
- `plot_file::Text`, 39
- `plot_profile::Axis`, 7
- `plot_profile::Curve`, 10
- `plot_profile::DataSet`, 13
- `plot_profile::Plot`, 36
- `work_plot::Settings`, 38

`average_PKE.py`, 41

## `calculateCurve`

- `light_curve::LightCurve`, 30

`combine_bins.py`, 41

`combine_bins_persistent.py`, 41

`compare_sedov_blasts.py`, 42

## `convertDumpToHDF`

- `make_hdf2::fileSet`, 24

- `make_hdf::fileSet`, 23

`cp_files.py`, 42

## `createTables`

- `eos_interp::interpTableManager`, 29

`datafile.DataFile`, 10

## `datafile::DataFile`

- `readFile`, 11

- `readFileFixed`, 11

- `readFileUnFixed`, 11

`dump.dump`, 14

## `dump::dump`

- `__init__`, 15
- `printHeader`, 15
- `printVar`, 15
- `read`, 15
- `readAsciiVar`, 16
- `readBinaryVar`, 16
- `readHeader`, 16
- `readHeaderAscii`, 16
- `readHeaderBinary`, 16
- `setVarIDs`, 17

`eos_interp.eosTable`, 17

`eos_interp.eosTableManager`, 19

`eos_interp.interpTable`, 25

`eos_interp.interpTableManager`, 28

`eos_interp.opacityTable`, 31

`eos_interp.opacityTableManager`, 33

## `eos_interp::eosTable`

- `__init__`, 18
- `interpolate`, 18
- `load`, 18
- `plotLogE`, 19
- `plotLogP`, 19
- `write`, 19

## `eos_interp::eosTableManager`

- `__init__`, 20
- `getTableFromComp`, 20
- `interpComp`, 20
- `load`, 21
- `plotGrid`, 21

## `eos_interp::interpTable`

- `__init__`, 26
- `interpolate`, 27
- `plotLogE`, 27
- `plotLogK`, 27
- `plotLogP`, 27
- `read`, 28

## `eos_interp::interpTableManager`

- `__init__`, 29
- `createTables`, 29

---

- eos\_interp::opacityTable
  - \_\_init\_\_, 32
  - fillInDepNans, 32
  - interpolate, 32
  - load, 33
  - plotLogK, 33
- eos\_interp::opacityTableManager
  - \_\_init\_\_, 34
  - getTableFromComp, 34
  - interpComp, 34
  - load, 35
  - plotGrids, 35
- fillInDepNans
  - eos\_interp::opacityTable, 32
- getCurve
  - plot\_file::DataSet, 12
  - plot\_profile::DataSet, 13
- getTableFromComp
  - eos\_interp::eosTableManager, 20
  - eos\_interp::opacityTableManager, 34
- interpComp
  - eos\_interp::eosTableManager, 20
  - eos\_interp::opacityTableManager, 34
- interpolate
  - eos\_interp::eosTable, 18
  - eos\_interp::interpTable, 27
  - eos\_interp::opacityTable, 32
- light\_curve.LightCurve, 30
- light\_curve::LightCurve
  - calculateCurve, 30
  - readBoloCorr, 31
  - readProfiles, 31
  - write, 31
- load
  - eos\_interp::eosTable, 18
  - eos\_interp::eosTableManager, 21
  - eos\_interp::opacityTable, 33
  - eos\_interp::opacityTableManager, 35
  - plot\_2DSlices::File2DSlice, 22
  - plot\_file::Axis, 6
  - plot\_file::Curve, 9
  - plot\_file::DataSet, 12
  - plot\_file::Plot, 38
  - plot\_profile::Axis, 7
  - plot\_profile::Curve, 10
  - plot\_profile::DataSet, 13
  - plot\_profile::Plot, 37
  - make\_hdf.fileSet, 22
  - make\_hdf.hdfFile, 25
  - make\_hdf.interpVar, 29
  - make\_hdf.variable, 40
  - make\_hdf2.fileSet, 23
  - make\_hdf2::fileSet
    - \_\_init\_\_, 24
    - convertDumpToHDF, 24
    - makeHDFFiles, 24
  - make\_hdf::fileSet
    - \_\_init\_\_, 22
    - convertDumpToHDF, 23
    - makeHDFFiles, 23
  - make\_hdf::hdfFile
    - write, 25
  - makeHDFFiles
    - make\_hdf2::fileSet, 24
    - make\_hdf::fileSet, 23
  - parseXML
    - work\_plot::Settings, 39
  - plot\_2DSlices.File2DSlice, 21
  - plot\_2DSlices::File2DSlice
    - load, 22
  - plot\_file.Axis, 5
  - plot\_file.Curve, 7
  - plot\_file.DataSet, 11
  - plot\_file.Plot, 37
  - plot\_file.Text, 39
  - plot\_file::Axis
    - \_\_init\_\_, 5
    - load, 6
  - plot\_file::Curve
    - \_\_init\_\_, 8
    - load, 9
  - plot\_file::DataSet
    - \_\_init\_\_, 12
    - getCurve, 12
    - load, 12
  - plot\_file::Plot
    - \_\_init\_\_, 37
    - load, 38
  - plot\_file::Text
    - \_\_init\_\_, 39
  - plot\_profile.Axis, 6
  - plot\_profile.Curve, 9
  - plot\_profile.DataSet, 12
  - plot\_profile.Plot, 36

---

---

- plot\_profile::Axis
  - \_\_init\_\_, 7
  - load, 7
- plot\_profile::Curve
  - \_\_init\_\_, 10
  - load, 10
- plot\_profile::DataSet
  - \_\_init\_\_, 13
  - getCurve, 13
  - load, 13
- plot\_profile::Plot
  - \_\_init\_\_, 36
  - load, 37
- plotGrid
  - eos\_interp::eosTableManager, 21
- plotGrids
  - eos\_interp::opacityTableManager, 35
- plotLogE
  - eos\_interp::eosTable, 19
  - eos\_interp::interpTable, 27
- plotLogK
  - eos\_interp::interpTable, 27
  - eos\_interp::opacityTable, 33
- plotLogP
  - eos\_interp::eosTable, 19
  - eos\_interp::interpTable, 27
- printHeader
  - dump::dump, 15
- printVar
  - dump::dump, 15
- read
  - dump::dump, 15
  - eos\_interp::interpTable, 28
- readAsciiVar
  - dump::dump, 16
- readBinaryVar
  - dump::dump, 16
- readBoloCorr
  - light\_curve::LightCurve, 31
- readFile
  - datafile::DataFile, 11
- readFileFixed
  - datafile::DataFile, 11
- readFileUnFixed
  - datafile::DataFile, 11
- readHeader
  - dump::dump, 16
- readHeaderAscii
  - dump::dump, 16
- readHeaderBinary
  - dump::dump, 16
- readProfiles
  - light\_curve::LightCurve, 31
- setVarIDs
  - dump::dump, 17
- work\_plot.PdVPlotSettings, 35
- work\_plot.Settings, 38
- work\_plot.WorkPlotSettings, 40
- work\_plot::Settings
  - \_\_init\_\_, 38
  - parseXML, 39
- write
  - eos\_interp::eosTable, 19
  - light\_curve::LightCurve, 31
  - make\_hdf::hdfFile, 25

---