

PyCos

1.x

Generated by Doxygen 1.7.6.1

Tue Feb 11 2014 13:52:43

Contents

1	Class Index	1
1.1	Class List	1
2	File Index	3
2.1	File List	3
3	Class Documentation	5
3.1	Cosmology Class Reference	5
3.1.1	Detailed Description	6
3.1.2	Constructor & Destructor Documentation	6
3.1.2.1	Cosmology	6
3.1.2.2	~Cosmology	6
3.1.3	Member Function Documentation	6
3.1.3.1	age	6
3.1.3.2	age_now	6
3.1.3.3	ang_dist	6
3.1.3.4	ang_dist_z1_z2	6
3.1.3.5	comoving_volume_0_z	7
3.1.3.6	comoving_volume_z1_z2	7
3.1.3.7	conformal_time	7
3.1.3.8	D_C	7
3.1.3.9	D_M	7
3.1.3.10	lookback_time	7

3.1.3.11	lum_dist	7
3.1.3.12	unit_comoving_volume	7
3.1.4	Member Data Documentation	7
3.1.4.1	h	7
3.1.4.2	omega_k	8
3.1.4.3	omega_m	8
3.1.4.4	omega_x	8
3.1.4.5	w_x	8
4	File Documentation	9
4.1	Cosmology.h File Reference	9
4.1.1	Function Documentation	10
4.1.1.1	double_schechter	10
4.1.1.2	E_z	10
4.1.1.3	luminosity_double_schechter	10
4.1.1.4	luminosity_schechter	10
4.1.1.5	number_double_schechter	10
4.1.1.6	number_schechter	10
4.1.1.7	schechter	10
4.1.1.8	time_integral	11
4.1.1.9	total_luminosity_double_schechter	11
4.1.1.10	total_luminosity_schechter	11
4.1.1.11	total_number_schechter	11

Chapter 1

Class Index

1.1 Class List

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

Cosmology	5
----------------------------	---

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Cosmology.h	9
------------------------------	---

Chapter 3

Class Documentation

3.1 Cosmology Class Reference

```
#include <Cosmology.h>
```

Public Member Functions

- **Cosmology** (double o_m=0.260, double **omega_x**=0.740, double **omega_k**=0.-0, double **w_x**=-1.0, double **h**=0.72)
- **~Cosmology** ()
- double **D_C** (double z1, double z2)
- double **D_M** (double z1, double z2)
- double **ang_dist** (double z)
- double **ang_dist_z1_z2** (double z1, double z2)
- double **lum_dist** (double z)
- double **unit_comoving_volume** (double z, void *params)
- double **comoving_volume_z1_z2** (double z1, double z2)
- double **comoving_volume_0_z** (double z)
- double **lookback_time** (double z)
- double **age** (double z)
- double **age_now** (void)
- double **conformal_time** (double z)

Private Attributes

- double **omega_m**
- double **omega_x**

- double **omega_k**
- double **w_x**
- double **h**

3.1.1 Detailed Description

Definition at line 9 of file Cosmology.h.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 Cosmology::Cosmology (double *o_m* = 0.260, double *omega_x* = 0.740, double *omega_k* = 0.0, double *w_x* = -1.0, double *h* = 0.72)

Constructs a cosmological model. Deafulat values are the concordance LCDM.

3.1.2.2 Cosmology::~~Cosmology ()

The destructor does nothing.

3.1.3 Member Function Documentation

3.1.3.1 double Cosmology::age (double *z*)

Cosmic age at *z* in years

3.1.3.2 double Cosmology::age_now (void)

Cosmic age at *z*=0 in years

3.1.3.3 double Cosmology::ang_dist (double *z*)

Angular size distance to a given redshift *z*. It returns distance in Mpc.

3.1.3.4 double Cosmology::ang_dist_z1_z2 (double *z1*, double *z2*)

Angular size distance between *z1* and *z2*. It returns distance in Mpc.

3.1.3.5 `double Cosmology::comoving_volume_0_z(double z)`

Comoving volume between $z=0$ and z in Mpc^3 per unit solid angle.

3.1.3.6 `double Cosmology::comoving_volume_z1_z2(double z1, double z2)`

Comoving volume between $z1$ and $z2$ in Mpc^3 per unit solid angle.

3.1.3.7 `double Cosmology::conformal_time(double z)`

Conformal time at z in years

3.1.3.8 `double Cosmology::D_C(double z1, double z2)`

(Radial) comoving distance. It returns comoving distance between $z1$ and $z2$ in Mpc.

3.1.3.9 `double Cosmology::D_M(double z1, double z2)`

Proper distance (aka. transverse comoving distance). It returns proper distance between $z1$ and $z2$ in Mpc.

3.1.3.10 `double Cosmology::lookback_time(double z)`

Lookback time to z in years

3.1.3.11 `double Cosmology::lum_dist(double z)`

Luminosity distance to a given redshift z . It returns distance in Mpc.

3.1.3.12 `double Cosmology::unit_comoving_volume(double z, void * params)`

Unit comoving volume at a given redshift z . It returns volume in Mpc^3 per unit solid angle and unit change of redshift.

3.1.4 Member Data Documentation

3.1.4.1 `double Cosmology::h [private]`

parametrized Hubble constant, $H = 100h \text{ km s}^{-1} \text{ Mpc}^{-1}$

Definition at line 16 of file Cosmology.h.

3.1.4.2 `double Cosmology::omega_k` `[private]`

Omega curvature

Definition at line 14 of file Cosmology.h.

3.1.4.3 `double Cosmology::omega_m` `[private]`

Omega mass

Definition at line 12 of file Cosmology.h.

3.1.4.4 `double Cosmology::omega_x` `[private]`

Omega dark energy

Definition at line 13 of file Cosmology.h.

3.1.4.5 `double Cosmology::w_x` `[private]`

EOS of dark energy

Definition at line 15 of file Cosmology.h.

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

- **Cosmology.h**

Chapter 4

File Documentation

4.1 Cosmology.h File Reference

Classes

- class **Cosmology**

Functions

- double **E_z** (double x, void *params)
- double **time_integral** (double x, void *params)
- double **schechter** (double L, double phi_star, double alpha, double L_star)
- double **number_schechter** (double L, double phi_star, double alpha, double L_star)
- double **total_number_schechter** (double phi_star, double alpha)
- double **luminosity_schechter** (double L, double phi_star, double alpha, double L_star)
- double **total_luminosity_schechter** (double phi_star, double alpha, double L_star)
- double **double_schechter** (double L, double phi_star1, double alpha1, double L_star1, double phi_star2, double alpha2, double L_star2)
- double **number_double_schechter** (double L, double phi_star1, double alpha1, double L_star1, double phi_star2, double alpha2, double L_star2)
- double **luminosity_double_schechter** (double L, double phi_star1, double alpha1, double L_star1, double phi_star2, double alpha2, double L_star2)
- double **total_luminosity_double_schechter** (double phi_star1, double alpha1, double L_star1, double phi_star2, double alpha2, double L_star2)

4.1.1 Function Documentation

4.1.1.1 `double double_schechter(double L, double phi_star1, double alpha1, double L_star1, double phi_star2, double alpha2, double L_star2)`

General double Schechter function $\phi(L)$ $dL = \phi_star1 * (L/L_star1)^\alpha * \exp(-L/L_star1) dL / L_star1 + \phi_star2 * (L/L_star2)^\alpha * \exp(-L/L_star2) dL / L_star2$ The unit of $\phi(L)$ is same as that of ϕ_star1 and ϕ_star2 .

4.1.1.2 `double E_z(double x, void * params)`

4.1.1.3 `double luminosity_double_schechter(double L, double phi_star1, double alpha1, double L_star1, double phi_star2, double alpha2, double L_star2)`

the luminosity integration of double Schechter function $L(>L)$ $L(>L) = \text{Integration of } L * \phi(L) \text{ from } L \text{ to infinity}$ The unit of the returned value depends on the unit of ϕ_star .

4.1.1.4 `double luminosity_schechter(double L, double phi_star, double alpha, double L_star)`

the luminosity integration of Schechter function $L(>L)$ $L(>L) = \text{Integration of } L * \phi(L) \text{ from } L \text{ to infinity}$ The unit of the returned value depends on the unit of ϕ_star .

4.1.1.5 `double number_double_schechter(double L, double phi_star1, double alpha1, double L_star1, double phi_star2, double alpha2, double L_star2)`

the number integration of double Schechter function $N(>L)$ $N(>L) = \text{Integration of } \phi(L) \text{ from } L \text{ to infinity}$ The unit of the returned value depends on the unit of ϕ_star .

4.1.1.6 `double number_schechter(double L, double phi_star, double alpha, double L_star)`

the number integration of Schechter function $N(>L)$ $N(>L) = \text{Integration of } \phi(L) \text{ from } L \text{ to infinity}$ The unit of the returned value depends on the unit of ϕ_star .

4.1.1.7 `double schechter(double L, double phi_star, double alpha, double L_star)`

Schechter function $\phi(L)$ $dL = \phi_star * (L/L_star)^\alpha * \exp(-L/L_star) dL / L_star$ The unit of $\phi(L)$ is same as that of ϕ_star .

4.1.1.8 `double time_integral (double x, void * params)`

4.1.1.9 `double total_luminosity_double_schechter (double phi_star1, double alpha1,
double L_star1, double phi_star2, double alpha2, double L_star2)`

the luminosity integration of double Schechter function $L(>0)$ $L(>0)$ = Integration of $L \cdot \phi(L)$ from 0 to infinity The unit of the returned value depends on the unit of *phi_star*.

4.1.1.10 `double total_luminosity_schechter (double phi_star, double alpha, double L_star
)`

the luminosity integration of Schechter function $L(>0)$ $L(>0)$ = Integration of $L \cdot \phi(L)$ from 0 to infinity The unit of the returned value depends on the unit of *phi_star*.

4.1.1.11 `double total_number_schechter (double phi_star, double alpha)`

the number integration of Schechter function $N(>0)$ $N(>0)$ = Integration of $\phi(L)$ from 0 to infinity The unit of the returned value depends on the unit of *phi_star*.

Index

- ~Cosmology
 - Cosmology, 6
- Cosmology, 5
 - ~Cosmology, 6
 - Cosmology, 6
 - D_C, 7
 - D_M, 7
 - age, 6
 - age_now, 6
 - ang_dist, 6
 - ang_dist_z1_z2, 6
 - comoving_volume_0_z, 6
 - comoving_volume_z1_z2, 7
 - conformal_time, 7
 - Cosmology, 6
 - h, 7
 - lookback_time, 7
 - lum_dist, 7
 - omega_k, 8
 - omega_m, 8
 - omega_x, 8
 - unit_comoving_volume, 7
 - w_x, 8
- Cosmology.h, 9
 - E_z, 10
 - double_schechter, 10
 - luminosity_double_schechter, 10
 - luminosity_schechter, 10
 - number_double_schechter, 10
 - number_schechter, 10
 - schechter, 10
 - time_integral, 10
 - total_luminosity_double_schechter, 11
 - total_luminosity_schechter, 11
 - total_number_schechter, 11
- D_C
 - Cosmology, 7
- D_M
 - Cosmology, 7
- E_z
 - Cosmology.h, 10
- age
 - Cosmology, 6
- age_now
 - Cosmology, 6
- ang_dist
 - Cosmology, 6
- ang_dist_z1_z2
 - Cosmology, 6
- comoving_volume_0_z
 - Cosmology, 6
- comoving_volume_z1_z2
 - Cosmology, 7
- conformal_time
 - Cosmology, 7
- double_schechter
 - Cosmology.h, 10
- h
 - Cosmology, 7
- lookback_time
 - Cosmology, 7
- lum_dist
 - Cosmology, 7
- luminosity_double_schechter
 - Cosmology.h, 10
- luminosity_schechter
 - Cosmology.h, 10
- number_double_schechter

Cosmology.h, 10
number_schechter
 Cosmology.h, 10

omega_k
 Cosmology, 8
omega_m
 Cosmology, 8
omega_x
 Cosmology, 8

schechter
 Cosmology.h, 10

time_integral
 Cosmology.h, 10
total_luminosity_double_schechter
 Cosmology.h, 11
total_luminosity_schechter
 Cosmology.h, 11
total_number_schechter
 Cosmology.h, 11

unit_comoving_volume
 Cosmology, 7

w_x
 Cosmology, 8