DE4xxSolarSystemEphemerides

5.0

Generated by Doxygen 1.8.5

Wed Jun 1 2022 12:06:52

Contents

Chapter 1

Module Index

1.1 Modules

Here is a list of all modules:

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

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

Chapter 4

Data Structure Index

4.1 Data Structures

Here are the data structures with brief descriptions:

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

Chapter 6

Module Documentation

6.1 Models

Modules

- Environment

6.1.1 Detailed Description

6.2 Environment

Modules

- Ephemerides

6.2.1 Detailed Description

6.3 Ephemerides

Modules

- De4xxEphem
- EphemInterface
- EphemItem
- EphemManager
- PropagatedPlanet

6.3.1 Detailed Description

6.4 De4xxEphem

Files

- file de4xx_ephem/include/class_declarations.hh

  Forward declarations of classes defined in the DE4xx model.
- file de4xx_base.hh

  Define data types for JPL ephemeris model.
- file de4xx_ephem.hh

  Define class for the De4xx ephemeris model.
- file de4xx_file.hh

  Define the class responsible for reading the DE4xx ephemeris file.
- file de4xx_ephem.cc

  Define the methods of the classes defined in de4xx_ephem.hh.
- file de4xx_ephem_dynmanager.cc

  Wall off dependencies on the dynamics manager.
- file de4xx_file.cc

  This file defines several utility functions used to read a binary JPL DE405 ephemeris file.
- file de4xx_file_init.cc

  Define De4xx initialization methods.
- file de4xx_file_update.cc

  Define De4xxFile::update.

Namespaces

- jeod

  Namespace jeod.

Macros

- #define __STDC_LIMIT_MACROS

6.4.1 Detailed Description

6.4.2 Macro Definition Documentation

6.4.2.1 #define __STDC_LIMIT_MACROS

Definition at line 60 of file de4xx_file.cc.

6.5 EphemInterface

Files

- file ephem_interface/include/class_declarations.hh

  Forward declarations of classes defined in models/environment/ephemerides/ephem_interface files.
- file ephem_interface.hh

  Define base class for all ephemeris interface models.
- file ephem_messages.hh

  Define the class EphemeridesMessages, the class that specifies the message IDs used in the JEOD ephemerides model.
- file ephem_ref_frame.hh

  Define the class EphemerisRefFrame.
- file simple_ephemerides.hh

  Define classes that define simple ephemeris models.
- file ephem_messages.cc

  Implement the class EphemeridesMessages.
- file ephem_ref_frame.cc

  Define non-inlined member functions for the EphemRefFrame class.
- file simple_ephemerides.cc

  Define member functions for the SinglePointEphemeris class and subclasses.

Namespaces

- jeod

  Namespace jeod.

6.5.1 Detailed Description

6.6 EphemItem

Files

- file ephem_item/include/class_declarations.hh

  Forward declarations of classes defined in models/environment/ephemerides/ephem_item files.
- file ephem_item.hh

  Define classes for items represented in some ephemeris model.
- file ephem_item_inline.hh

  Define inline methods for the EphemerisItem class.
- file ephem_orient.hh

  Define class EphemerisOrientation.
- file ephem_orient_zxz.hh

  Define classes for items represented in some ephemeris model.
- file ephem_point.hh

  Define class EphemerisPoint.
- file ephem_item.cc

  Define member functions for the EphemItem class and subclasses.
- file ephem_orient.cc

  Define member functions for the EphemItem class and subclasses.
- file ephem_orient_zxz.cc

  Define member functions for the EphemItem class and subclasses.
- file ephem_point.cc

  Define member functions for the EphemPoint class.

Namespaces

- jeod

  Namespace jeod.

Macros

- #define EPSILON_TIME 1e-12
- #define TAYLOR_CUTOFF 0.00786

6.6.1 Detailed Description

6.6.2 Macro Definition Documentation

6.6.2.1 #define EPSILON_TIME 1e-12

Definition at line 62 of file ephem_orient_zxz.cc.

Referenced by jeod::EphemerisZXZOrientation::propagate().

6.6.2.2 #define TAYLOR_CUTOFF 0.00786

Definition at line 72 of file ephem_orient_zxz.cc.

Referenced by jeod::EphemerisZXZOrientation::propagate().

6.7 EphemManager

Files

- file base_ephem_manager.hh

  Define the BaseEphemManager class, which defines the interfaces to the class EphemManager.
- file ephem_manager.hh

  Define the EphemManager class, which manages the ephemeris models in a JEOD-based simulation.
- file ephem_manager.cc

  Define EphemeridesManager methods.
- file find_planet.cc

  Define EphemeridesManager::find_planet.

Namespaces

- jeod

  Namespace jeod.

6.7.1 Detailed Description

6.8 PropagatedPlanet

Files

- file propagated_planet.hh

  Define the classes needed to propagate a planet.
- file propagated_planet.cc

  Define the methods of the classes defined in propagated_planet.hh.

Namespaces

- jeod

  Namespace jeod.

6.8.1 Detailed Description

Chapter 7

Namespace Documentation

7.1 jeod Namespace Reference

Namespace jeod.

Namespaces

- De4xxBase

  Defines enumerations used in the DE4xx ephemeris model.

Data Structures

- class De4xxEphemItem

  Describes a point modeled in a DE4xx ephemeris file.
- class De4xxEphemeris

  The S_define-level class that provides planetary ephemerides.
- struct EphemerisDataSetMeta

  Container for the metadata from the DE model header.
- struct EphemerisDataItemMeta

  Structure containing the header metadata for sizing/locating the data entries with the data segments.
- struct EphemerisDataSegmentMeta

  Metadata implied from each data segment.
- class De4xxFileSpec

  Specifies which file to use (user input initialization-time data).
- class De4xxFileIO

  Contains data used directly for reading the ephemeris file.
- class De4xxFileHeader

  Contains data extracted from the ephemeris file header.
- class De4xxFileItem

  Contains data regarding one of the items in a DE ephemeris file.
- class De4xxFileRefTime

  Contains timing reference data.
- class De4xxFileCoef

  Contains Chebychev polynomial coefficients and terms.
- class De4xxFileRestart

  The FILE pointer in a De4xxFileIO cannot be restored by Trick.
- class De4xxFile

Provides the ability to read and interpret a DE4xx ephemeris file.

- class EphemerisInterface

  Interface class that specifies minimal functionality of an ephemeris model.

- class EphemeridesMessages

  Specifies the message IDs used in the Ephemerides model.

- class EphemerisRefFrame

  An EphemerisRefFrame is a RefFrame whose state is set by an ephemeris model.

- class SinglePointEphemeris

  A SinglePointEphemeris has one ephemeris point.

- class EmptySpaceEphemeris

  Empty space has one ephemeris point.

- class SinglePlanetEphemeris

  A space with one gravitation body has one ephemeris point.

- class EphemerisItem

  The EphemerisItem class is the base class for representing an item that is modeled in an ephemeris model.

- class EphemerisOrientation

  An EphemerisOrientation object updates the rotational state of an ephemeris reference frame.

- class EphemerisZXZOrientation

  The EphemerisZXZOrientation is an EphemerisOrientation subclass that updates orientation based on an Z-X-Z Euler sequence and the time derivatives of this sequence.

- class EphemerisPoint

  An EphemerisPoint object updates the translational state of an ephemeris reference frame.

- class BaseEphemeridesManager

  The EphemManager class augments the RefFrameManager with ephemeris-related items.

- class EphemeridesManager

  The EphemeridesManager class manages the ephemeris models in a simulation.

- class PropagatedEphemerisPlanet

  A PropagatedEphemerisPlanet is an EphemerisPoint whose state is coupled with the translational state of a DynBody reference frame.

- class PropagatedEphemerisOrientation

  A PropagatedEphemerisOrientation is an EphemerisOrientation whose state is coupled with the rotational state of a DynBody reference frame.

- class PropagatedPlanet

  The PropagatedPlanet ephemeris model provides planetary state via a DynBody object whose state is propagated using the JEOD state integration techniques.

Functions

- void process_mem_usage (double &vm_usage, double &resident_set)
- static double l1_point (double b1b2_mass_ratio)

  Calculate the location of the L1 point as a ratio.

7.1.1 Detailed Description

Namespace jeod.

7.1.2 Function Documentation

7.1.2.1 static double jeod::l1_point (double *b1b2_mass_ratio*) [static]

Calculate the location of the L1 point as a ratio.

Returns

Ratio of body1 to L1-point distance to body1 to body2 distance

Parameters

in	*b1b2_mass_ratio*	Body1 to body2 mass ratio

Definition at line 288 of file de4xx_file_init.cc.

Referenced by jeod::De4xxFile::initialize().

7.1.2.2 void jeod::process_mem_usage (double & *vm_usage*, double & *resident_set*)

Definition at line 519 of file de4xx_file.cc.

Referenced by jeod::De4xxFile::capture_mem_stats().

7.2 jeod::De4xxBase Namespace Reference

Defines enumerations used in the DE4xx ephemeris model.

Enumerations

- enum De4xxFileEntries {
  De4xx_File_Mercury = 0, De4xx_File_Venus = 1, De4xx_File_EMbary = 2, De4xx_File_Mars = 3,
  De4xx_File_Jupiter = 4, De4xx_File_Saturn = 5, De4xx_File_Uranus = 6, De4xx_File_Neptune = 7,
  De4xx_File_Pluto = 8, De4xx_File_Moon = 9, De4xx_File_Sun = 10, De4xx_File_ENutation = 11,
  De4xx_File_LLibration = 12, De4xx_File_LAngVel = 13, De4xx_File_tt_tdb = 14, De4xx_File_MaxEntries }

  Defines names for planetary body descriptors in the ephemeris file.
- enum De4xxEphemConsts {
  De4xx_Const_DENUM = 0, De4xx_Const_LENUM, De4xx_Const_AU, De4xx_Const_EMRAT,
  De4xx_Const_CLIGHT, De4xx_Const_GM1, De4xx_Const_GM2, De4xx_Const_GMB,
  De4xx_Const_GM4, De4xx_Const_GM5, De4xx_Const_GM6, De4xx_Const_GM7,
  De4xx_Const_GM8, De4xx_Const_GM9, De4xx_Const_GMS, De4xx_Const_MaxConsts }

  Index aliases for the constants listed in the DE header that are used by JEOD.
- enum De4xxEphemBodies {
  De4xx_Ephem_Sun = 0, De4xx_Ephem_Mercury = 1, De4xx_Ephem_Venus = 2, De4xx_Ephem_Earth = 3,
  De4xx_Ephem_Mars = 4, De4xx_Ephem_Jupiter = 5, De4xx_Ephem_Saturn = 6, De4xx_Ephem_Uranus = 7,
  De4xx_Ephem_Neptune = 8, De4xx_Ephem_Pluto = 9, De4xx_Ephem_Moon = 10, De4xx_Ephem_EMbary = 11,
  De4xx_Ephem_SSbary = 12, De4xx_Ephem_EML1 = 13, De4xx_Ephem_ENutation = 14, De4xx_Ephem_LLibration = 15,
  De4xx_Ephem_MaxBodies }

  Defines names for ephemeris items as represented in the JEOD DE4xx model.

Functions

- static const char ∗point_names[32] __attribute__ ((unused))
- static uint32_t number_jeod_items (int de_version_num)

  Total number of items in the JEOD ephemeris.
- static uint32_t number_trans_points (int de_version_num)

  Total number of translational states in the JEOD ephemeris.
- static uint32_t number_grav_models (int de_version_num)

  Number of gravity models in the JEOD ephemeris (Mercury to Sun + implied Earth) Currently only one possibility, but written for extensibility.
- static uint32_t number_physical_bodies (int de_version_num)

  Number of bodies in the JEOD ephemeris (Planets + Pluto + Moon + Sun) Currently only one possibility, but written for extensibility.

7.2.1 Detailed Description

Defines enumerations used in the DE4xx ephemeris model.

7.2.2 Enumeration Type Documentation

7.2.2.1 enum jeod::De4xxBase::De4xxEphemBodies

Defines names for ephemeris items as represented in the JEOD DE4xx model.

NOTA BENE: The Earth-moon barycenter follows the massive bodies as the barycenter is not itself a massive body.

Enumerator

De4xx_Ephem_Sun Sun.

De4xx_Ephem_Mercury Mercury.

De4xx_Ephem_Venus Venus.

De4xx_Ephem_Earth Earth.

De4xx_Ephem_Mars Mars.

De4xx_Ephem_Jupiter Jupiter.

De4xx_Ephem_Saturn Saturn.

De4xx_Ephem_Uranus Uranus.

De4xx_Ephem_Neptune Neptune.

De4xx_Ephem_Pluto Pluto.

De4xx_Ephem_Moon Moon.

De4xx_Ephem_EMbary Earth-moon barycenter.

De4xx_Ephem_SSbary Solar system barycenter.

De4xx_Ephem_EML1 Earth-moon L1 point (disabled)

De4xx_Ephem_ENutation Nutations (disabled)

De4xx_Ephem_LLibration Librations.

De4xx_Ephem_MaxBodies

Definition at line 122 of file de4xx_base.hh.

7.2.2.2 enum jeod::De4xxBase::De4xxEphemConsts

Index aliases for the constants listed in the DE header that are used by JEOD.

This is an incomplete subset of the constants provided by DE, but all of these constants must be present for JEOD ephemeris to perform.

Enumerator

De4xx_Const_DENUM
De4xx_Const_LENUM
De4xx_Const_AU
De4xx_Const_EMRAT
De4xx_Const_CLIGHT
De4xx_Const_GM1
De4xx_Const_GM2
De4xx_Const_GMB
De4xx_Const_GM4
De4xx_Const_GM5
De4xx_Const_GM6
De4xx_Const_GM7
De4xx_Const_GM8
De4xx_Const_GM9
De4xx_Const_GMS
De4xx_Const_MaxConsts

Definition at line 96 of file de4xx_base.hh.

7.2.2.3 enum jeod::De4xxBase::De4xxFileEntries

Defines names for planetary body descriptors in the ephemeris file.

This enum defines names for the bodies as they are represented in the ephemeris file.

Enumerator

De4xx_File_Mercury Mercury XYZ [km].
De4xx_File_Venus Venus XYZ.
De4xx_File_EMbary Earth-moon barycenter XYZ.
De4xx_File_Mars Mars XYZ.
De4xx_File_Jupiter Jupiter XYZ.
De4xx_File_Saturn Saturn XYZ.
De4xx_File_Uranus Uranus XYZ.
De4xx_File_Neptune Neptune XYZ.
De4xx_File_Pluto Pluto XYZ.
De4xx_File_Moon Moon, geocentric coords. XYZ.
De4xx_File_Sun Sun XYZ.
De4xx_File_ENutation Earth Nutations d(psi), d(eps) [rad].
De4xx_File_LLibration Lun mantle Libratns phi,tht,psi [rad].
De4xx_File_LAngVel Lun mantle ang vel omg_{xyz} [rad/day].
De4xx_File_tt_tdb TDB to TT offset @ geocenter [s].
De4xx_File_MaxEntries

Definition at line 69 of file de4xx_base.hh.

7.2.3 Function Documentation

7.2.3.1 static const char* point_names [32] jeod::De4xxBase::__attribute__ ((unused)) [static]

7.2.3.2 static uint32_t jeod::De4xxBase::number_grav_models (int *de_version_num*) [inline], [static]

Number of gravity models in the JEOD ephemeris (Mercury to Sun + implied Earth) Currently only one possibility, but written for extensibility.

Definition at line 191 of file de4xx_base.hh.

Referenced by jeod::De4xxFileHeader::De4xxFileHeader(), and jeod::De4xxFile::initialize().

7.2.3.3 static uint32_t jeod::De4xxBase::number_jeod_items (int *de_version_num*) [inline], [static]

Total number of items in the JEOD ephemeris.

Refer to De4xxEphemBodies for identities of each item. Currently only one possibility, but written for extensibility

Definition at line 171 of file de4xx_base.hh.

Referenced by jeod::De4xxEphemeris::activate_nodes(), jeod::De4xxEphemeris::De4xxEphemeris(), jeod::De4xxEphemeris::ephem_activate(), jeod::De4xxEphemeris::ephem_initialize(), and jeod::De4xxEphemeris::initialize_items().

7.2.3.4 static uint32_t jeod::De4xxBase::number_physical_bodies (int *de_version_num*) [inline], [static]

Number of bodies in the JEOD ephemeris (Planets + Pluto + Moon + Sun) Currently only one possibility, but written for extensibility.

Definition at line 200 of file de4xx_base.hh.

7.2.3.5 static uint32_t jeod::De4xxBase::number_trans_points (int *de_version_num*) [inline], [static]

Total number of translational states in the JEOD ephemeris.

(Sun, Mercury to Pluto, EMBary, SSbary) Currently excludes EML1 Currently only one possibility, but written for extensibility

Definition at line 181 of file de4xx_base.hh.

Referenced by jeod::De4xxEphemeris::activate_nodes(), jeod::De4xxEphemeris::De4xxEphemeris(), jeod::De4xxEphemeris::determine_root_node(), jeod::De4xxEphemeris::ephem_build_tree(), and jeod::De4xxEphemeris::initialize_items().

Chapter 8

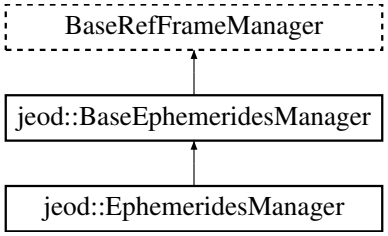
Data Structure Documentation

8.1 jeod::BaseEphemeridesManager Class Reference

The EphemManager class augments the RefFrameManager with ephemeris-related items.

`#include <base_ephem_manager.hh>`

Inheritance diagram for jeod::BaseEphemeridesManager:



Public Member Functions

- virtual ~BaseEphemeridesManager ()

  Destructor.
- virtual void ephem_note_tree_status_change ()=0

  Denote that the tree needs to be rebuilt.
- virtual void add_planet (BasePlanet &planet)=0

  Add a planet to the list of such.
- virtual void add_planet (Planet &planet)=0

  Add a planet to the list of such.
- virtual BasePlanet * find_base_planet (const char *name) const =0

  Find a planet.
- virtual Planet * find_planet (const char *name) const =0

  Find a planet.
- virtual unsigned int get_num_planets (void) const =0

  Return number of registered planets.
- virtual void add_ephemeris (EphemerisInterface &ephem_if)=0

  Add an ephemeris model to the list of such.
- virtual void clear_added_ephemerides (void)=0

  Deactivate all registered ephemeris models.
- virtual void disable_add_ephemeris (void)=0

Disable registration of new ephemeris models.

- virtual void add_ephem_item (EphemerisItem &ephem_item)=0

  Add an ephemeris item to the list of such.
- virtual EphemerisItem * find_ephem_item (const char *name) const =0

  Find an ephemeris item.
- virtual EphemerisOrientation * find_ephem_angle (const char *name) const =0

  Find an ephemeris orientation.
- virtual EphemerisPoint * find_ephem_point (const char *name) const =0

  Find an ephemeris point.
- virtual void add_integ_frame (EphemerisRefFrame &ref_frame)=0

  Add an integration frame to the list of such.
- virtual EphemerisRefFrame * find_integ_frame (const char *name) const =0

  Find an integration frame.
- virtual bool is_integ_frame (const RefFrame &ref_frame) const =0

  Check whether a reference frame is an integration frame.
- virtual unsigned int find_integ_frame_index (const EphemerisRefFrame &ref_frame) const =0

  Find a reference frame's index in the list of integration frames.
- virtual const std::vector < EphemerisRefFrame * > & get_integ_frames (void) const =0

  Get the vector of integration frames.

Friends

- class InputProcessor
- void init_attrjeod__BaseEphemeridesManager ()

8.1.1 Detailed Description

The EphemManager class augments the RefFrameManager with ephemeris-related items.

This class defines the external interfaces to that class.

Definition at line 61 of file base_ephem_manager.hh.

8.1.2 Constructor & Destructor Documentation

8.1.2.1 virtual jeod::BaseEphemeridesManager::~BaseEphemeridesManager ( ) [inline], [virtual]

Destructor.

Definition at line 75 of file base_ephem_manager.hh.

8.1.3 Member Function Documentation

8.1.3.1 virtual void jeod::BaseEphemeridesManager::add_ephem_item (EphemerisItem & *ephem_item*) [pure virtual]

Add an ephemeris item to the list of such.

Parameters

ephem_item	Item to be added.

Implemented in jeod::EphemeridesManager.

8.1.3.2 virtual void jeod::BaseEphemeridesManager::add_ephemeris (EphemerisInterface & *ephem_if*) [pure virtual]

Add an ephemeris model to the list of such.

Parameters

ephem_if	Ephemeris model to be added.

Implemented in jeod::EphemeridesManager.

8.1.3.3 virtual void jeod::BaseEphemeridesManager::add_integ_frame (EphemerisRefFrame & *ref_frame*) [pure virtual]

Add an integration frame to the list of such.

Parameters

ref_frame	Frame to be added.

Implemented in jeod::EphemeridesManager.

8.1.3.4 virtual void jeod::BaseEphemeridesManager::add_planet (BasePlanet & *planet*) [pure virtual]

Add a planet to the list of such.

Parameters

planet	Planet to be added.

Implemented in jeod::EphemeridesManager.

8.1.3.5 virtual void jeod::BaseEphemeridesManager::add_planet (Planet & *planet*) [pure virtual]

Add a planet to the list of such.

Parameters

planet	Planet to be added.

Implemented in jeod::EphemeridesManager.

8.1.3.6 virtual void jeod::BaseEphemeridesManager::clear_added_ephemerides (void) [pure virtual]

Deactivate all registered ephemeris models.

Implemented in jeod::EphemeridesManager.

8.1.3.7 virtual void jeod::BaseEphemeridesManager::disable_add_ephemeris (void) [pure virtual]

Disable registration of new ephemeris models.

Implemented in jeod::EphemeridesManager.

8.1.3.8 virtual void jeod::BaseEphemeridesManager::ephem_note_tree_status_change ( ) [pure virtual]

Denote that the tree needs to be rebuilt.

Implemented in jeod::EphemeridesManager.

Referenced by jeod::EphemerisItem::disable(), jeod::EphemerisItem::enable(), and jeod::EphemerisRefFrame::set_active_status().

8.1.3.9 virtual BasePlanet* jeod::BaseEphemeridesManager::find_base_planet (const char * *name*) const [pure virtual]

Find a planet.

Parameters

name	Planet name.

Returns

Pointer to found planet.

Implemented in jeod::EphemeridesManager.

8.1.3.10 virtual EphemerisOrientation* jeod::BaseEphemeridesManager::find_ephem_angle (const char * *name*) const [pure virtual]

Find an ephemeris orientation.

Parameters

name	Item to be found.

Returns

Found item.

Implemented in jeod::EphemeridesManager.

8.1.3.11 virtual EphemerisItem* jeod::BaseEphemeridesManager::find_ephem_item (const char * *name*) const [pure virtual]

Find an ephemeris item.

Parameters

name	Item to be found.

Returns

Found item.

Implemented in jeod::EphemeridesManager.

8.1.3.12 virtual EphemerisPoint* jeod::BaseEphemeridesManager::find_ephem_point (const char * *name*) const [pure virtual]

Find an ephemeris point.

Parameters

name	Item to be found.

Returns

Found item.

Implemented in jeod::EphemeridesManager.

8.1.3.13 virtual EphemerisRefFrame* jeod::BaseEphemeridesManager::find_integ_frame (const char * *name*) const [pure virtual]

Find an integration frame.

Parameters

name	Frame to be found.

Returns

Found frame.

Implemented in jeod::EphemeridesManager.

8.1.3.14 virtual unsigned int jeod::BaseEphemeridesManager::find_integ_frame_index (const EphemerisRefFrame & *ref_frame*) const [pure virtual]

Find a reference frame's index in the list of integration frames.

Parameters

ref_frame	Frame to be checked.

Returns

Frame index.

Implemented in jeod::EphemeridesManager.

8.1.3.15 virtual Planet* jeod::BaseEphemeridesManager::find_planet (const char * *name*) const [pure virtual]

Find a planet.

Parameters

name	Planet name.

Returns

Pointer to found planet.

Implemented in jeod::EphemeridesManager.

8.1.3.16 virtual const std::vector<EphemerisRefFrame *>& jeod::BaseEphemeridesManager::get_integ_frames (void) const [pure virtual]

Get the vector of integration frames.

Returns

Vector of reference frame pointers.

Implemented in jeod::EphemeridesManager.

8.1.3.17 virtual unsigned int jeod::BaseEphemeridesManager::get_num_planets (void) const [pure virtual]

Return number of registered planets.

Returns

Number of planets.

Implemented in jeod::EphemeridesManager.

8.1.3.18 virtual bool jeod::BaseEphemeridesManager::is_integ_frame (const RefFrame & *ref_frame*) const [pure virtual]

Check whether a reference frame is an integration frame.

Parameters

ref_frame	Frame to be checked.

Returns

True if ref_frame is an integration frame, false otherwise.

Implemented in jeod::EphemeridesManager.

Referenced by jeod::EphemerisItem::set_target_frame().

8.1.4 Friends And Related Function Documentation

8.1.4.1 void init_attrjeod__BaseEphemeridesManager ( ) [friend]

8.1.4.2 friend class InputProcessor [friend]

Definition at line 64 of file base_ephem_manager.hh.

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

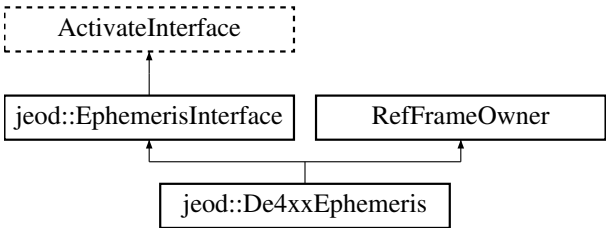
- base_ephem_manager.hh

8.2 jeod::De4xxEphemeris Class Reference

The S_define-level class that provides planetary ephemerides.

`#include <de4xx_ephem.hh>`

Inheritance diagram for jeod::De4xxEphemeris:



Public Member Functions

- De4xxEphemeris (void)

  De4xxEphemeris default constructor.
- ~De4xxEphemeris (void)

  De4xxEphemeris destructor.
- void initialize_model (const TimeManager &time_manager, DynManager &dyn_manager, std::string time_type="TT")

  Initialize the De4xxEphemeris model.
- void initialize_model (const TimeManager &time_manager, EphemeridesManager &ephem_manager, std::string time_type="TT")

  Initialize the De4xxEphemeris model.
- void propagate_lunar_rnp (void)

  Propagate the lunar orientation to the current time.
- void shutdown (void)

  Free resources allocated by the De4xxEphemeris model.
- void activate (void)

  Nominally, activate the object.
- void deactivate (void)

  Deactivate the De4xxEphemeris object.
- double timestamp (void) const

  Return time of last update.
- const char * get_name (void) const

  Return model name.
- void ephem_initialize (EphemeridesManager &ephem_manager)

  Complete the initialization process.
- void ephem_activate (EphemeridesManager &ephem_manager)

  Mark appropriate items in the model as active.
- void ephem_build_tree (EphemeridesManager &ephem_manager)

  Construct the ephemeris model portions of the reference frame tree.
- void ephem_update (void)

  Update ephemerides for subscribed items.
- bool time_is_in_range (void) const

  Check whether the specified time is represented in the JPL ephemeris file.
- void set_model_number (int denum_in)

  Set ephemeris model number.
- uint32_t get_model_number ()

  Get Ephemeris model number.
- const De4xxFileHeader & get_header_data ()

Data Fields

- bool active

  Is the model active? This is set to true by the constructor.
- bool * selected_items

  Used at initialization time only to selectively enable/disable portions of the model.

Protected Attributes

- De4xxFile file

  The ephemeris file model.
- bool force_update

  Is an update needed even if the time hasn't changed?
- unsigned int nactive_items

  Number of items that are currently active.
- De4xxEphemItem * item_data

  Data pertaining to the points for which translational states are calculated.
- char * ident

  Identifier for this model, computed from the supplied file.
- double update_time

  Time of last update, dynamic time seconds.
- EphemerisPoint * points

  The planets and barycenter points, in De4xxEphemBodies FileBodies order.
- EphemerisZXZOrientation lunar_orientation

  Lunar orientation.
- EphemerisRefFrame earth_moon_barycenter_frame

  Earth-Moon barycenter reference frame.
- EphemerisRefFrame solar_system_barycenter_frame

  Solar system barycenter reference frame.
- De4xxEphemItem * root_item

  The root point in the reference frame tree.
- const TimeStandard * time_tt

  The source of ephemeris time information.
- const TimeDyn * time_dyn

  The source of dynamic time information.
- int * body_to_file_idx

  Mapping from De4xxEphemBodies numbers to De4xxFileBodies numbers.

Private Member Functions

- void initialize_time (const TimeManager &time_manager, std::string time_type)

  Initialize De4xxEphemeris timing.
- void initialize_file (void)

  Initialize the De4xxEphemeris file.
- void initialize_items (EphemeridesManager &ephem_manager)

  Initialize the De4xxEphemeris item data.
- unsigned int activate_nodes (void)

  Mark appropriate items in the model as active.
- unsigned int activate_em_nodes (unsigned int tot_active)

  Adjust Earth, Moon, and Earth-Moon barycenter activity.

- void determine_root_node (void)

  Determine which item should be the root of the ref frame tree.
- De4xxEphemeris (const De4xxEphemeris &)

  Not implemented.
- De4xxEphemeris & operator= (const De4xxEphemeris &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__De4xxEphemeris ()

8.2.1 Detailed Description

The S_define-level class that provides planetary ephemerides.

The De4xxEphemeris class constructs the ephemeris reference frame tree and updates the states of the planets based on data from a DE4xx ephemeris model.

Definition at line 169 of file de4xx_ephem.hh.

8.2.2 Constructor & Destructor Documentation

8.2.2.1 jeod::De4xxEphemeris::De4xxEphemeris (void)

De4xxEphemeris default constructor.

Definition at line 128 of file de4xx_ephem.cc.

References body_to_file_idx, jeod::De4xxBase::De4xx_Ephem_Earth, jeod::De4xxBase::De4xx_Ephem_EMbary, jeod::De4xxBase::De4xx_Ephem_EML1, jeod::De4xxBase::De4xx_Ephem_ENutation, jeod::De4xxBase::De4xx_Ephem_Jupiter, jeod::De4xxBase::De4xx_Ephem_LLibration, jeod::De4xxBase::De4xx_Ephem_Mars, jeod::De4xxBase::De4xx_Ephem_Mercury, jeod::De4xxBase::De4xx_Ephem_Moon, jeod::De4xxBase::De4xx_Ephem_Neptune, jeod::De4xxBase::De4xx_Ephem_Pluto, jeod::De4xxBase::De4xx_Ephem_Saturn, jeod::De4xxBase::De4xx_Ephem_SSbary, jeod::De4xxBase::De4xx_Ephem_Sun, jeod::De4xxBase::De4xx_Ephem_Uranus, jeod::De4xxBase::De4xx_Ephem_Venus, jeod::De4xxBase::De4xx_File_EMbary, jeod::De4xxBase::De4xx_File_ENutation, jeod::De4xxBase::De4xx_File_Jupiter, jeod::De4xxBase::De4xx_File_LLibration, jeod::De4xxBase::De4xx_File_Mars, jeod::De4xxBase::De4xx_File_MaxEntries, jeod::De4xxBase::De4xx_File_Mercury, jeod::De4xxBase::De4xx_File_Moon, jeod::De4xxBase::De4xx_File_Neptune, jeod::De4xxBase::De4xx_File_Pluto, jeod::De4xxBase::De4xx_File_Saturn, jeod::De4xxBase::De4xx_File_Sun, jeod::De4xxBase::De4xx_File_Uranus, jeod::De4xxBase::De4xx_File_Venus, earth_moon_barycenter_frame, jeod::EphemerisOrientation::enable(), jeod::EphemerisItem::enable(), jeod::EphemerisItem::get_name(), jeod::De4xxEphemItem::index, jeod::De4xxEphemItem::item, item_data, lunar_orientation, jeod::De4xxEphemItem::name, jeod::De4xxBase::number_jeod_items(), jeod::De4xxBase::number_trans_points(), points, selected_items, jeod::EphemerisItem::set_name(), jeod::EphemerisItem::set_owner(), and solar_system_barycenter_frame.

8.2.2.2 jeod::De4xxEphemeris::~De4xxEphemeris (void)

De4xxEphemeris destructor.

Definition at line 227 of file de4xx_ephem.cc.

References body_to_file_idx, item_data, points, selected_items, and shutdown().

8.2.2.3 jeod::De4xxEphemeris::De4xxEphemeris (const De4xxEphemeris &) [private]

Not implemented.

8.2.3 Member Function Documentation

8.2.3.1 void jeod::De4xxEphemeris::activate (void)

Nominally, activate the object.

In the case of a De4xxEphemeris object, an inactive object cannot be activated.

Definition at line 260 of file de4xx_ephem.cc.

References active, and jeod::EphemeridesMessages::internal_error.

8.2.3.2 unsigned int jeod::De4xxEphemeris::activate_em_nodes (unsigned int *tot_active*) [private]

Adjust Earth, Moon, and Earth-Moon barycenter activity.

Returns

Void

Parameters

in	*tot_active*	Number active translation nodes

Definition at line 587 of file de4xx_ephem.cc.

References jeod::EphemerisItem::activate(), jeod::De4xxEphemItem::Active, jeod::De4xxBase::De4xx_Ephem_Earth, jeod::De4xxBase::De4xx_Ephem_EMbary, jeod::De4xxBase::De4xx_Ephem_Moon, jeod::De4xxEphemItem::Deselected, jeod::EphemerisItem::enable(), jeod::De4xxEphemItem::enabled_item, jeod::De4xxEphemItem::Inactive, jeod::EphemeridesMessages::inconsistent_setup, jeod::De4xxEphemItem::InTree, jeod::De4xxEphemItem::item, item_data, nactive_items, jeod::De4xxEphemItem::name, and jeod::De4xxEphemItem::status.

Referenced by ephem_activate().

8.2.3.3 unsigned int jeod::De4xxEphemeris::activate_nodes (void) [private]

Mark appropriate items in the model as active.

Returns

Void

Definition at line 543 of file de4xx_ephem.cc.

References jeod::De4xxEphemItem::Active, jeod::De4xxEphemItem::Deselected, jeod::De4xxEphemItem::enabled_item, file, jeod::De4xxFile::file_spec, jeod::EphemerisItem::get_enabled_item(), jeod::De4xxFileSpec::get_model_number(), jeod::De4xxEphemItem::Inactive, jeod::De4xxEphemItem::InTree, jeod::EphemerisItem::is_active(), jeod::De4xxEphemItem::item, item_data, nactive_items, jeod::De4xxBase::number_jeod_items(), jeod::De4xxBase::number_trans_points(), and jeod::De4xxEphemItem::status.

Referenced by ephem_activate().

8.2.3.4 void jeod::De4xxEphemeris::deactivate (void)

Deactivate the De4xxEphemeris object.

Definition at line 277 of file de4xx_ephem.cc.

References active.

8.2.3.5 void jeod::De4xxEphemeris::determine_root_node (void) [private]

Determine which item should be the root of the ref frame tree.

Definition at line 655 of file de4xx_ephem.cc.

References jeod::De4xxBase::De4xx_Ephem_Earth, jeod::De4xxBase::De4xx_Ephem_EMbary, jeod::De4xxBase::De4xx_Ephem_Moon, jeod::De4xxBase::De4xx_Ephem_SSbary, jeod::De4xxEphemItem::enabled_item, file, jeod::De4xxFile::file_spec, jeod::De4xxFileSpec::get_model_number(), jeod::De4xxEphemItem::Inactive, jeod::De4xxEphemItem::IsRoot, jeod::De4xxEphemItem::item, item_data, nactive_items, jeod::De4xxBase::number_trans_points(), root_item, and jeod::De4xxEphemItem::status.

Referenced by ephem_activate().

8.2.3.6 void jeod::De4xxEphemeris::ephem_activate (EphemeridesManager & *ephem_manager*) [virtual]

Mark appropriate items in the model as active.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::EphemerisInterface.

Definition at line 721 of file de4xx_ephem.cc.

References activate_em_nodes(), activate_nodes(), jeod::De4xxEphemItem::Active, active, jeod::De4xxFileItem::active, body_to_file_idx, jeod::De4xxBase::De4xx_Ephem_EMbary, jeod::De4xxBase::De4xx_Ephem_LLibration, jeod::De4xxBase::De4xx_Ephem_Sun, jeod::De4xxBase::De4xx_File_ENutation, jeod::De4xxBase::De4xx_File_LLibration, determine_root_node(), file, jeod::De4xxFile::file_spec, force_update, jeod::De4xxFileSpec::get_model_number(), jeod::De4xxFile::item, item_data, nactive_items, jeod::De4xxBase::number_jeod_items(), and jeod::De4xxEphemItem::status.

8.2.3.7 void jeod::De4xxEphemeris::ephem_build_tree (EphemeridesManager & *ephem_manager*) [virtual]

Construct the ephemeris model portions of the reference frame tree.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::EphemerisInterface.

Definition at line 780 of file de4xx_ephem.cc.

References jeod::De4xxEphemItem::Active, active, jeod::De4xxBase::De4xx_Ephem_Earth, jeod::De4xxBase::De4xx_Ephem_EMbary, jeod::De4xxBase::De4xx_Ephem_Moon, jeod::De4xxBase::De4xx_Ephem_SSbary, jeod::De4xxEphemItem::Deselected, jeod::De4xxEphemItem::enabled_item, file, jeod::De4xxFile::file_spec, jeod::De4xxEphemItem::frame, jeod::De4xxFileSpec::get_model_number(), jeod::EphemeridesMessages::inconsistent_setup, jeod::De4xxEphemItem::item, item_data, jeod::De4xxEphemItem::name, jeod::De4xxBase::number_trans_points(), root_item, and jeod::De4xxEphemItem::status.

8.2.3.8 void jeod::De4xxEphemeris::ephem_initialize (EphemeridesManager & *ephem_manager*) [virtual]

Complete the initialization process.

This method should be called after all other ephemeris models have completed their basic initialization and after all planets have registered themselves with the ephemeris manager.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::EphemerisInterface.

Definition at line 503 of file de4xx_ephem.cc.

References active, jeod::De4xxEphemItem::Deselected, file, jeod::De4xxFile::file_spec, jeod::De4xxEphemItem::frame, jeod::De4xxFileSpec::get_model_number(), jeod::EphemerisItem::get_target_frame(), jeod::De4xxEphemItem::item, item_data, jeod::De4xxBase::number_jeod_items(), and jeod::De4xxEphemItem::status.

8.2.3.9 void jeod::De4xxEphemeris::ephem_update (void) [virtual]

Update ephemerides for subscribed items.

Implements jeod::EphemerisInterface.

Definition at line 840 of file de4xx_ephem.cc.

References jeod::De4xxEphemItem::Active, active, jeod::De4xxFileHeader::be_em_dist_ratio, jeod::De4xxFileHeader::bm_em_dist_ratio, body_to_file_idx, jeod::De4xxBase::De4xx_Ephem_Earth, jeod::De4xxBase::De4xx_Ephem_EMbary, jeod::De4xxBase::De4xx_Ephem_LLibration, jeod::De4xxBase::De4xx_Ephem_Moon, jeod::De4xxBase::De4xx_Ephem_Sun, jeod::De4xxBase::De4xx_File_LLibration, jeod::De4xxEphemItem::enabled_item, file, force_update, jeod::De4xxFile::header, jeod::De4xxEphemItem::item, jeod::De4xxFile::item, item_data, lunar_orientation, nactive_items, points, root_item, jeod::De4xxFileItem::state, time_dyn, time_tt, jeod::EphemerisPoint::update(), jeod::EphemerisZXZOrientation::update(), jeod::De4xxFile::update(), jeod::EphemerisPoint::update_scaled(), and update_time.

8.2.3.10 const De4xxFileHeader& jeod::De4xxEphemeris::get_header_data ( ) [inline]

Definition at line 241 of file de4xx_ephem.hh.

References file, and jeod::De4xxFile::header.

8.2.3.11 uint32_t jeod::De4xxEphemeris::get_model_number ( ) [inline]

Get Ephemeris model number.

This number is used to specify the de file to use the pathname is of the form PWD/build/de4xx_lib/libde<denumIn>.so

Definition at line 237 of file de4xx_ephem.hh.

References file, jeod::De4xxFile::file_spec, and jeod::De4xxFileSpec::get_model_number().

8.2.3.12 const char * jeod::De4xxEphemeris::get_name (void) const [virtual]

Return model name.

Returns

Name

Implements jeod::EphemerisInterface.

Definition at line 304 of file de4xx_ephem.cc.

References ident.

8.2.3.13 void jeod::De4xxEphemeris::initialize_file (void) [private]

Initialize the De4xxEphemeris file.

Definition at line 392 of file de4xx_ephem.cc.

References jeod::De4xxBase::De4xx_Const_DENUM, jeod::De4xxBase::De4xx_Const_LENUM, jeod::EphemerisDataSetMeta::de_constants, file, ident, jeod::De4xxFile::initialize(), jeod::De4xxFile::io, jeod::De4xxFileIO::metaData, and time_tt.

Referenced by initialize_model().

8.2.3.14 void jeod::De4xxEphemeris::initialize_items (EphemeridesManager & *ephem_manager*) [private]

Initialize the De4xxEphemeris item data.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Definition at line 428 of file de4xx_ephem.cc.

References jeod::EphemeridesManager::add_ephem_item(), jeod::EphemeridesManager::add_integ_frame(), jeod::De4xxBase::De4xx_Ephem_Earth, jeod::De4xxBase::De4xx_Ephem_EMbary, jeod::De4xxBase::De4xx_Ephem_LLibration, jeod::De4xxBase::De4xx_Ephem_Moon, jeod::De4xxBase::De4xx_Ephem_SSbary, jeod::De4xxEphemItem::Deselected, jeod::EphemerisItem::disable(), earth_moon_barycenter_frame, file, jeod::De4xxFile::file_spec, jeod::De4xxFileSpec::get_model_number(), jeod::De4xxEphemItem::Inactive, jeod::EphemeridesMessages::inconsistent_setup, jeod::De4xxEphemItem::item, item_data, lunar_orientation, jeod::De4xxBase::number_jeod_items(), jeod::De4xxBase::number_trans_points(), points, selected_items, solar_system_barycenter_frame, and jeod::De4xxEphemItem::status.

Referenced by initialize_model().

8.2.3.15 void jeod::De4xxEphemeris::initialize_model (const TimeManager & *time_manager*, DynManager & *dyn_manager*, std::string *time_type* = "TT")

Initialize the De4xxEphemeris model.

Parameters

in	*time_manager*	Time manager
in,out	*dyn_manager*	Dynamics manager
in	*time_type*	time type

Definition at line 53 of file de4xx_ephem_dynmanager.cc.

8.2.3.16 void jeod::De4xxEphemeris::initialize_model (const TimeManager & *time_manager*, EphemeridesManager & *ephem_manager*, std::string *time_type* = "TT")

Initialize the De4xxEphemeris model.

This method is called before the planets have been registered with the reference frame manager, so we don't know whether the ephemeris items should be enabled or disabled.

Parameters

in	*time_manager*	Time manager
in,out	*ephem_manager*	Ephemerides manager

in	*time_type*	optional "tt"||"tdb" "tt" default manager

Definition at line 322 of file de4xx_ephem.cc.

References active, jeod::EphemeridesManager::add_ephemeris(), initialize_file(), initialize_items(), and initialize_time().

8.2.3.17 void jeod::De4xxEphemeris::initialize_time (const TimeManager & *time_manager,* std::string *time_type*) [private]

Initialize De4xxEphemeris timing.

Parameters

in	*time_manager*	Time manager
in	*time_type*	time type

Definition at line 353 of file de4xx_ephem.cc.

References jeod::EphemeridesMessages::inconsistent_setup, time_dyn, and time_tt.

Referenced by initialize_model().

8.2.3.18 De4xxEphemeris& jeod::De4xxEphemeris::operator= (const De4xxEphemeris &) [private]

Not implemented.

8.2.3.19 void jeod::De4xxEphemeris::propagate_lunar_rnp (void)

Propagate the lunar orientation to the current time.

Definition at line 942 of file de4xx_ephem.cc.

References jeod::De4xxEphemItem::Active, active, jeod::De4xxBase::De4xx_Ephem_LLibration, item_data, lunar_orientation, jeod::EphemerisZXZOrientation::propagate(), and time_dyn.

8.2.3.20 void jeod::De4xxEphemeris::set_model_number (int *denum_in*) [inline]

Set ephemeris model number.

This number is used to specify the de file to use the pathname is of the form PWD/build/de4xx_lib/libde<denum_In>.so

Definition at line 228 of file de4xx_ephem.hh.

References file, jeod::De4xxFile::file_spec, and jeod::De4xxFileSpec::set_model_number().

8.2.3.21 void jeod::De4xxEphemeris::shutdown (void)

Free resources allocated by the De4xxEphemeris model.

Definition at line 242 of file de4xx_ephem.cc.

References file, ident, and jeod::De4xxFile::shutdown().

Referenced by ~De4xxEphemeris().

8.2.3.22 bool jeod::De4xxEphemeris::time_is_in_range (void) const

Check whether the specified time is represented in the JPL ephemeris file.

Assumptions and Limitations

- Ephemeris file is open for input
- Ephemeris file is blocked per value set in the ephem_file structure

Returns

True if time is in file

Definition at line 930 of file de4xx_ephem.cc.

References file, jeod::De4xxFile::time_is_in_range(), and time_tt.

8.2.3.23 double jeod::De4xxEphemeris::timestamp (void) const [virtual]

Return time of last update.

Returns

Timestamp
Units: day

Implements jeod::EphemerisInterface.

Definition at line 291 of file de4xx_ephem.cc.

References update_time.

8.2.4 Friends And Related Function Documentation

8.2.4.1 void init_attrjeod__De4xxEphemeris ( ) [friend]

8.2.4.2 friend class InputProcessor [friend]

Definition at line 171 of file de4xx_ephem.hh.

8.2.5 Field Documentation

8.2.5.1 bool jeod::De4xxEphemeris::active

Is the model active? This is set to true by the constructor.

Setting this flag to false prior to initialization time will result in the model never doing anything. Setting this flag to false after the model has been active for some time is not supported.trick_units(–)

Definition at line 255 of file de4xx_ephem.hh.

Referenced by activate(), deactivate(), ephem_activate(), ephem_build_tree(), ephem_initialize(), ephem_update(), initialize_model(), and propagate_lunar_rnp().

8.2.5.2 int* jeod::De4xxEphemeris::body_to_file_idx [protected]

Mapping from De4xxEphemBodies numbers to De4xxFileBodies numbers.

trick_units(–)

Definition at line 349 of file de4xx_ephem.hh.

Referenced by De4xxEphemeris(), ephem_activate(), ephem_update(), and ~De4xxEphemeris().

8.2.5.3 EphemerisRefFrame jeod::De4xxEphemeris::earth_moon_barycenter_frame [protected]

Earth-Moon barycenter reference frame.

trick_units(–)

Definition at line 324 of file de4xx_ephem.hh.

Referenced by De4xxEphemeris(), and initialize_items().

8.2.5.4 De4xxFile jeod::De4xxEphemeris::file [protected]

The ephemeris file model.

The items of interest to the typical user are the data members file.file_spec.ephem_file_name and file.file_spec.denum. The former specifies the name of the file while the latter serves as a sanity check that the right file is being read.trick_units(–)

Definition at line 283 of file de4xx_ephem.hh.

Referenced by activate_nodes(), determine_root_node(), ephem_activate(), ephem_build_tree(), ephem_initialize(), ephem_update(), get_header_data(), get_model_number(), initialize_file(), initialize_items(), set_model_number(), shutdown(), and time_is_in_range().

8.2.5.5 bool jeod::De4xxEphemeris::force_update [protected]

Is an update needed even if the time hasn't changed?

trick_units(–)

Definition at line 288 of file de4xx_ephem.hh.

Referenced by ephem_activate(), and ephem_update().

8.2.5.6 char* jeod::De4xxEphemeris::ident [protected]

Identifier for this model, computed from the supplied file.

trick_units(–)

Definition at line 304 of file de4xx_ephem.hh.

Referenced by get_name(), initialize_file(), and shutdown().

8.2.5.7 De4xxEphemItem* jeod::De4xxEphemeris::item_data [protected]

Data pertaining to the points for which translational states are calculated.

trick_units(–)

Definition at line 299 of file de4xx_ephem.hh.

Referenced by activate_em_nodes(), activate_nodes(), De4xxEphemeris(), determine_root_node(), ephem_activate(), ephem_build_tree(), ephem_initialize(), ephem_update(), initialize_items(), propagate_lunar_rnp(), and ~De4xxEphemeris().

8.2.5.8 EphemerisZXZOrientation jeod::De4xxEphemeris::lunar_orientation [protected]

Lunar orientation.

trick_units(–)

Definition at line 319 of file de4xx_ephem.hh.

Referenced by De4xxEphemeris(), ephem_update(), initialize_items(), and propagate_lunar_rnp().

8.2.5.9 unsigned int jeod::De4xxEphemeris::nactive_items [protected]

Number of items that are currently active.

trick_units(–)

Definition at line 293 of file de4xx_ephem.hh.

Referenced by activate_em_nodes(), activate_nodes(), determine_root_node(), ephem_activate(), and ephem_update().

8.2.5.10 EphemerisPoint* jeod::De4xxEphemeris::points [protected]

The planets and barycenter points, in De4xxEphemBodies FileBodies order.

trick_units(–)

Definition at line 314 of file de4xx_ephem.hh.

Referenced by De4xxEphemeris(), ephem_update(), initialize_items(), and ~De4xxEphemeris().

8.2.5.11 De4xxEphemItem* jeod::De4xxEphemeris::root_item [protected]

The root point in the reference frame tree.

trick_units(–)

Definition at line 334 of file de4xx_ephem.hh.

Referenced by determine_root_node(), ephem_build_tree(), and ephem_update().

8.2.5.12 bool* jeod::De4xxEphemeris::selected_items

Used at initialization time only to selectively enable/disable portions of the model.

The constructor initializes all elements of this array to true. Users can set selected elements to false to disable the corresponding ephemeris items. The intent is to enable the use of multiple ephemeris models. Typical users of the model can leave this member as-is. NOTE: while the container is of length 16, Nutations and lunar librations are not currently supported by JEODtrick_units(–)

Definition at line 267 of file de4xx_ephem.hh.

Referenced by De4xxEphemeris(), initialize_items(), and ~De4xxEphemeris().

8.2.5.13 EphemerisRefFrame jeod::De4xxEphemeris::solar_system_barycenter_frame [protected]

Solar system barycenter reference frame.

trick_units(–)

Definition at line 329 of file de4xx_ephem.hh.

Referenced by De4xxEphemeris(), and initialize_items().

8.2.5.14 const TimeDyn* jeod::De4xxEphemeris::time_dyn [protected]

The source of dynamic time information.

trick_units(–)

Definition at line 344 of file de4xx_ephem.hh.

Referenced by ephem_update(), initialize_time(), and propagate_lunar_rnp().

8.2.5.15 const TimeStandard* jeod::De4xxEphemeris::time_tt [protected]

The source of ephemeris time information.

trick_units(–)

Definition at line 339 of file de4xx_ephem.hh.

Referenced by ephem_update(), initialize_file(), initialize_time(), and time_is_in_range().

8.2.5.16 double jeod::De4xxEphemeris::update_time [protected]

Time of last update, dynamic time seconds.

trick_units(s)

Definition at line 309 of file de4xx_ephem.hh.

Referenced by ephem_update(), and timestamp().

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

- de4xx_ephem.hh
- de4xx_ephem.cc
- de4xx_ephem_dynmanager.cc

8.3 jeod::De4xxEphemItem Class Reference

Describes a point modeled in a DE4xx ephemeris file.

`#include <de4xx_ephem.hh>`

Public Types

- enum Status {
  Deselected = 0, Inactive = 1, IsRoot = 2, InTree = 3,
  Active = 4 }

  Enumerates the status values of a De4xEphemItem.

Public Member Functions

- De4xxEphemItem (void)

  De4xxEphemItem default constructor.
- ~De4xxEphemItem (void)

  De4xxEphemItem destructor.

Protected Attributes

- EphemerisItem * item

  The ephemeris item for this item from this model.
- EphemerisItem * enabled_item

  The enabled ephemeris item for this item, not necessarily from this model.
- EphemerisRefFrame * frame

The reference frame whose state is set by this item.

- const char * name

  Item name; used for reporting errors.

- Status status

  The status for this item.

- unsigned int index

  The node index number, per the De4xxEphemBodies numbering scheme.

Private Member Functions

- De4xxEphemItem (const De4xxEphemItem &)

  Not implemented.

- De4xxEphemItem & operator= (const De4xxEphemItem &)

  Not implemented.

Friends

- class InputProcessor
- class De4xxEphemeris
- void init_attrjeod__De4xxEphemItem ()

8.3.1 Detailed Description

Describes a point modeled in a DE4xx ephemeris file.

This class is only used inside the De4xxEphemeris class as the type of the protected item_data data member.

Definition at line 70 of file de4xx_ephem.hh.

8.3.2 Member Enumeration Documentation

8.3.2.1 enum jeod::De4xxEphemItem::Status

Enumerates the status values of a De4xEphemItem.

Enumerator

Deselected The item is marked as deselected or the corresponding reference frame is not present in the simulation. The corresponding ephemeris item is permanently disabled under such circumstances.

Inactive The item has not been deselected and the corresponding reference frame is present but is inactive.

IsRoot The item has not been deselected and the corresponding reference frame is present and active. However, the item in question is the root of the reference frame tree and hence its state is the trivial state.

InTree The item has not been deselected and the corresponding reference frame is present and active. However, the ephemeris item that updates this frame is not a part of this model.

Active The item has not been deselected, the corresponding reference frame is present and active, and the ephemeris item that updates this frame belongs to this model.

Definition at line 82 of file de4xx_ephem.hh.

8.3.3 Constructor & Destructor Documentation

8.3.3.1 jeod::De4xxEphemItem::De4xxEphemItem (void)

De4xxEphemItem default constructor.

Definition at line 100 of file de4xx_ephem.cc.

8.3.3.2 jeod::De4xxEphemItem::~De4xxEphemItem (void)

De4xxEphemItem destructor.

Definition at line 117 of file de4xx_ephem.cc.

8.3.3.3 jeod::De4xxEphemItem::De4xxEphemItem (const De4xxEphemItem &) [private]

Not implemented.

8.3.4 Member Function Documentation

8.3.4.1 De4xxEphemItem& jeod::De4xxEphemItem::operator= (const De4xxEphemItem &) [private]

Not implemented.

8.3.5 Friends And Related Function Documentation

8.3.5.1 friend class De4xxEphemeris [friend]

Definition at line 73 of file de4xx_ephem.hh.

8.3.5.2 void init_attrjeod__De4xxEphemItem ( ) [friend]

8.3.5.3 friend class InputProcessor [friend]

Definition at line 71 of file de4xx_ephem.hh.

8.3.6 Field Documentation

8.3.6.1 EphemerisItem* jeod::De4xxEphemItem::enabled_item [protected]

The enabled ephemeris item for this item, not necessarily from this model.

trick_units(–)

Definition at line 126 of file de4xx_ephem.hh.

Referenced by jeod::De4xxEphemeris::activate_em_nodes(), jeod::De4xxEphemeris::activate_nodes(), jeod::De4xxEphemeris::determine_root_node(), jeod::De4xxEphemeris::ephem_build_tree(), and jeod::De4xxEphemeris::ephem_update().

8.3.6.2 EphemerisRefFrame* jeod::De4xxEphemItem::frame [protected]

The reference frame whose state is set by this item.

trick_units(–)

Definition at line 131 of file de4xx_ephem.hh.

Referenced by jeod::De4xxEphemeris::ephem_build_tree(), and jeod::De4xxEphemeris::ephem_initialize().

8.3.6.3 unsigned int jeod::De4xxEphemItem::index [protected]

The node index number, per the De4xxEphemBodies numbering scheme.

trick_units(–)

Definition at line 146 of file de4xx_ephem.hh.

Referenced by jeod::De4xxEphemeris::De4xxEphemeris().

8.3.6.4 EphemerisItem* jeod::De4xxEphemItem::item [protected]

The ephemeris item for this item from this model.

trick_units(–)

Definition at line 120 of file de4xx_ephem.hh.

Referenced by jeod::De4xxEphemeris::activate_em_nodes(), jeod::De4xxEphemeris::activate_nodes(), jeod::De4xxEphemeris::De4xxEphemeris(), jeod::De4xxEphemeris::determine_root_node(), jeod::De4xxEphemeris::ephem_build_tree(), jeod::De4xxEphemeris::ephem_initialize(), jeod::De4xxEphemeris::ephem_update(), and jeod::De4xxEphemeris::initialize_items().

8.3.6.5 const char* jeod::De4xxEphemItem::name [protected]

Item name; used for reporting errors.

trick_units(–)

Definition at line 136 of file de4xx_ephem.hh.

Referenced by jeod::De4xxEphemeris::activate_em_nodes(), jeod::De4xxEphemeris::De4xxEphemeris(), and jeod::De4xxEphemeris::ephem_build_tree().

8.3.6.6 Status jeod::De4xxEphemItem::status [protected]

The status for this item.

trick_units(–)

Definition at line 141 of file de4xx_ephem.hh.

Referenced by jeod::De4xxEphemeris::activate_em_nodes(), jeod::De4xxEphemeris::activate_nodes(), jeod::De4xxEphemeris::determine_root_node(), jeod::De4xxEphemeris::ephem_activate(), jeod::De4xxEphemeris::ephem_build_tree(), jeod::De4xxEphemeris::ephem_initialize(), and jeod::De4xxEphemeris::initialize_items().

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

- de4xx_ephem.hh
- de4xx_ephem.cc

8.4 jeod::De4xxFile Class Reference

Provides the ability to read and interpret a DE4xx ephemeris file.

`#include <de4xx_file.hh>`

Public Member Functions

- De4xxFile (void)

  Construct the JPL ephemeris file.
- ~De4xxFile (void)

  Destroy the JPL ephemeris file.
- void pre_initialize (void)

  Pre-initialize a DE4xxFile instance.
- void initialize (double epoch_time, double del_day, double time_offset, double init_time)

  Initialize a DE4xxFile instance.
- bool time_is_in_range (double time) const

  Check whether the specified time is represented in the JPL ephemeris file.
- void update (double time)

  Calcuate the position and velocity states of selected planetary bodies at some point in time.
- void shutdown ()

  Shutdown the JPL ephemeris file.

Data Fields

- De4xxFileSpec file_spec

  File specification.
- De4xxFileHeader header

  File header.
- De4xxFileItem * item

  Item data.
- De4xxFileIO io

  File descriptor.
- De4xxFileRefTime ref_time

  Reference time.
- De4xxFileCoef coef

  Chebychev coefs.
- De4xxFileRestart restart

  Restart handler.
- double update_time

  Time of last update.

Private Member Functions

- void open (void)

  Open the JPL ephemeris file.
- void reopen (void)

  Open the JPL ephemeris file on restart.
- void close (void)

  Close the JPL ephemeris file.
- void interpolate (double time, double fblk)

  Calcuate the position and velocity states of selected planetary bodies at some point in time.
- De4xxFile (const De4xxFile &)

  Not implemented.
- De4xxFile & operator= (const De4xxFile &)

  Not implemented.
- void capture_mem_stats ()

Friends

- class InputProcessor
- class De4xxFileRestart
- void init_attrjeod__De4xxFile ()

8.4.1 Detailed Description

Provides the ability to read and interpret a DE4xx ephemeris file.

Definition at line 593 of file de4xx_file.hh.

8.4.2 Constructor & Destructor Documentation

8.4.2.1 jeod::De4xxFile::De4xxFile (void)

Construct the JPL ephemeris file.

Definition at line 265 of file de4xx_file.cc.

References jeod::De4xxBase::De4xx_File_ENutation, jeod::De4xxBase::De4xx_File_LLibration, jeod::De4xxBase::De4xx_File_MaxEntries, jeod::De4xxBase::De4xx_File_tt_tdb, item, jeod::De4xxFileItem::nitems, jeod::De4xxFileItem::pscale, and restart.

8.4.2.2 jeod::De4xxFile::~De4xxFile (void)

Destroy the JPL ephemeris file.

Definition at line 305 of file de4xx_file.cc.

References close(), item, and restart.

8.4.2.3 jeod::De4xxFile::De4xxFile (const De4xxFile &) [private]

Not implemented.

8.4.3 Member Function Documentation

8.4.3.1 void jeod::De4xxFile::capture_mem_stats ( ) [private]

Definition at line 556 of file de4xx_file.cc.

References jeod::process_mem_usage().

8.4.3.2 void jeod::De4xxFile::close (void) [private]

Close the JPL ephemeris file.

Assumptions and Limitations

- Ephemeris file is open for input
- Errors are fatal

Definition at line 457 of file de4xx_file.cc.

References jeod::De4xxFileCoef::chebyderiv, jeod::De4xxFileCoef::chebypoly, coef, jeod::De4xxFileIO::file, jeod::EphemeridesMessages::file_error, io, and jeod::De4xxFileIO::metaData.

Referenced by shutdown(), and ~De4xxFile().

8.4.3.3 void jeod::De4xxFile::initialize (double *epoch_time,* double *del_day,* double *time_offset,* double *init_time*)

Initialize a DE4xxFile instance.

Parameters

in	*epoch_time*	Julian date Units: day
in	*del_day*	Days from epoch Units: day
in	*time_offset*	Terrestrial Time offet Units: s
in	*init_time*	Seconds from epoch Units: s

Definition at line 190 of file de4xx_file_init.cc.

References jeod::De4xxFileHeader::au, jeod::De4xxFileHeader::b1_em_dist_ratio, jeod::De4xxFileHeader::be_em_dist_ratio, jeod::De4xxFileRefTime::block_no, jeod::De4xxFileHeader::bm_em_dist_ratio, jeod::De4xxFileCoef::chebyderiv, jeod::De4xxFileCoef::chebypoly, coef, jeod::De4xxBase::De4xx_Const_AU, jeod::De4xxBase::De4xx_Const_CLIGHT, jeod::De4xxBase::De4xx_Const_EMRAT, jeod::De4xxBase::De4xx_Const_GM1, jeod::De4xxBase::De4xx_Const_GM2, jeod::De4xxBase::De4xx_Const_GM4, jeod::De4xxBase::De4xx_Const_GM5, jeod::De4xxBase::De4xx_Const_GM6, jeod::De4xxBase::De4xx_Const_GM7, jeod::De4xxBase::De4xx_Const_GM8, jeod::De4xxBase::De4xx_Const_GM9, jeod::De4xxBase::De4xx_Const_GMB, jeod::De4xxBase::De4xx_Const_GMS, jeod::De4xxBase::De4xx_Ephem_Earth, jeod::De4xxBase::De4xx_Ephem_EMbary, jeod::De4xxBase::De4xx_Ephem_Jupiter, jeod::De4xxBase::De4xx_Ephem_Mars, jeod::De4xxBase::De4xx_Ephem_Mercury, jeod::De4xxBase::De4xx_Ephem_Moon, jeod::De4xxBase::De4xx_Ephem_Neptune, jeod::De4xxBase::De4xx_Ephem_Pluto, jeod::De4xxBase::De4xx_Ephem_Saturn, jeod::De4xxBase::De4xx_Ephem_Sun, jeod::De4xxBase::De4xx_Ephem_Uranus, jeod::De4xxBase::De4xx_Ephem_Venus, jeod::EphemerisDataSetMeta::de_constants, jeod::EphemerisDataSetMeta::delta_epoch, jeod::De4xxFileSpec::denum, jeod::De4xxFileHeader::e1_em_dist_ratio, jeod::De4xxFileHeader::em_mass_ratio, jeod::De4xxFileRefTime::epoch_date, jeod::De4xxFileRefTime::fdate, file_spec, jeod::De4xxFileHeader::gmbody, header, jeod::De4xxFileRefTime::init_time, io, jeod::l1_point(), jeod::De4xxFileIO::max_terms, jeod::De4xxFileIO::metaData, jeod::De4xxBase::number_grav_models(), pre_initialize(), ref_time, jeod::De4xxFileIO::segmentData, jeod::EphemerisDataSegmentMeta::start_epoch, jeod::EphemeridesMessages::time_not_in_range, jeod::De4xxFileIO::total_num_recs, update_time, and jeod::De4xxFileHeader::vlight.

Referenced by jeod::De4xxEphemeris::initialize_file().

8.4.3.4 void jeod::De4xxFile::interpolate (double *time,* double *fblk*) [private]

Calcuate the position and velocity states of selected planetary bodies at some point in time.

Parameters

in	*time*	Time since reference Units: s
in	*fblk*	Fractional block

Definition at line 278 of file de4xx_file_update.cc.

References jeod::De4xxFileItem::active, jeod::De4xxFileCoef::chebyderiv, jeod::De4xxFileCoef::chebypoly, jeod::De4xxFileCoef::chebyterms, jeod::De4xxFileCoef::chebyx, jeod::De4xxFileCoef::coef, coef, jeod::EphemerisDataSetMeta::delta_epoch, io, item, jeod::De4xxFileItem::item_idx, itemData, jeod::De4xxFileIO::itemData, jeod::De4xxFileIO::metaData, jeod::De4xxFileItem::nitems, jeod::EphemerisDataItemMeta::npoly, jeod::EphemerisDataItemMeta::nterms, jeod::EphemerisDataSetMeta::number_file_items, jeod::EphemerisDataItemMeta::offset, jeod::De4xxFileItem::pscale, jeod::De4xxFileItem::state, and jeod::De4xxFileItem::update_time.

Referenced by update().

8.4.3.5 void jeod::De4xxFile::open (void) [private]

Open the JPL ephemeris file.

Assumptions and Limitations

- Errors are fatal

Returns

Void

Definition at line 334 of file de4xx_file.cc.

References jeod::De4xxBase::De4xx_File_MaxEntries, jeod::EphemeridesMessages::debug, jeod::De4xxFileSpec::ephem_file_dir, jeod::De4xxFileSpec::ephem_file_name, jeod::De4xxFileIO::file, jeod::EphemeridesMessages::file_error, file_spec, io, jeod::De4xxFileIO::itemData, jeod::De4xxFileIO::metaData, jeod::EphemerisDataSetMeta::number_file_items, jeod::De4xxFileSpec::pathname, and jeod::De4xxFileIO::segmentData.

Referenced by pre_initialize().

8.4.3.6 De4xxFile& jeod::De4xxFile::operator= (const De4xxFile &) [private]

Not implemented.

8.4.3.7 void jeod::De4xxFile::pre_initialize (void)

Pre-initialize a DE4xxFile instance.

Definition at line 83 of file de4xx_file_init.cc.

References jeod::De4xxFileItem::avail, jeod::De4xxFileCoef::coef, coef, jeod::De4xxFileIO::coeffs_segment_starting_addr, jeod::De4xxFileIO::current_record_starting_addr, jeod::De4xxBase::De4xx_Const_DENUM, jeod::De4xxBase::De4xx_File_MaxEntries, jeod::EphemerisDataSetMeta::de_constants, jeod::De4xxFileSpec::denum, jeod::De4xxFileIO::file, jeod::EphemeridesMessages::file_error, file_spec, jeod::EphemeridesMessages::garbage_file, jeod::EphemeridesMessages::internal_error, io, item, jeod::De4xxFileItem::item_idx, itemData, jeod::De4xxFileIO::itemData, jeod::De4xxFileIO::max_terms, jeod::De4xxFileIO::metaData, jeod::EphemerisDataItemMeta::nterms, jeod::EphemerisDataSegmentMeta::num_recs, jeod::EphemerisDataSetMeta::number_file_items, jeod::EphemerisDataSetMeta::number_segments, jeod::EphemerisDataItemMeta::offset, open(), jeod::De4xxFileSpec::pathname, jeod::De4xxFileIO::recno, jeod::De4xxFileIO::segment_index, jeod::De4xxFileIO::segment_recno, jeod::De4xxFileIO::segmentData, jeod::EphemerisDataSegmentMeta::start_epoch, jeod::EphemerisDataSegmentMeta::stop_epoch, and jeod::De4xxFileIO::total_num_recs.

Referenced by initialize(), and reopen().

8.4.3.8 void jeod::De4xxFile::reopen (void) [private]

Open the JPL ephemeris file on restart.

Assumptions and Limitations

- File spec has been reloaded.
- Data has been allocated
- Errors are fatal

Definition at line 435 of file de4xx_file.cc.

References jeod::De4xxFileIO::file, io, and pre_initialize().

Referenced by jeod::De4xxFileRestart::simple_restore().

8.4.3.9 void jeod::De4xxFile::shutdown (void)

Shutdown the JPL ephemeris file.

Definition at line 319 of file de4xx_file.cc.

References close().

Referenced by jeod::De4xxEphemeris::shutdown().

8.4.3.10 bool jeod::De4xxFile::time_is_in_range (double *time*) const

Check whether the specified time is represented in the JPL ephemeris file.

Assumptions and Limitations

- Ephemeris file is open for input
- Ephemeris file is blocked per value set in the ephem_file structure

Returns

True if time is in file

Parameters

in	*time*	Time since reference Units: s

Definition at line 500 of file de4xx_file.cc.

References jeod::De4xxFileRefTime::block_no, jeod::EphemerisDataSetMeta::delta_epoch, jeod::De4xxFileRefTime::init_time, io, jeod::De4xxFileIO::metaData, ref_time, and jeod::De4xxFileIO::total_num_recs.

Referenced by jeod::De4xxEphemeris::time_is_in_range().

8.4.3.11 void jeod::De4xxFile::update (double *time*)

Calcuate the position and velocity states of selected planetary bodies at some point in time.

The EphemeridesState structure embedded in the Ephemerides structure contains an update indicator and state for each body. The state for a body is updated if the body's indicator indicates that a state update is needed.

Body selection – The 'active_bodies' array in the EphemeridesState structure indicates which planets' states are to be updated. The function updates the position and velocity for the selected bodies.

Time specification – Four input variables are available for specifying the time.

- tt_offset is the offset between the remaining input times and Terrestrial Time (aka Terrestrial Dynamic Time, or TDT). Set this to zero if the other input times are already expressed in Terrestrial Time.
- For highest precision, set epoch_time to the Julian date at midnight of the time point of interest and set either del_day or del_time to the difference between the time point of interest and the epoch_time.
- An alternative that retains full precession is to pass the time at the start of the simulation in epoch_time and del_day and the time into the simulation in del_time. In this approach, epoch_time represents the Julian date at the midnight preceding the start of the simulation and del_day represents the time between the epoch time and simulation start.
- For ease of use, set epoch_time to the Julian date representing the the time point of interest and set both del_day and del_time to 0.0. Note that this approach has a machine granularity of about 0.2 msec.
- An intermediate alternative is to set the epoch_time to the start time of the simulation, del_day to zero and del_time to the simulation time in seconds. This approach will result in a small temporal bias due to the precision loss in the epoch time.

Outputs – Positions and velocities are expressed in the ICRF coordinate system. The states of the Sun and planets, including Earth-Moon barycenter, are expressed with respect to the solar system barycenter. Lunar states are expressed with respect to the center of the Earth.

NOTA BENE – The states of unselected bodies may or may not be changed.

Assumptions and Limitations

- Assumption 1. Ephemeris file is open.
- Assumption 2. Offset times in del_day and del_time are small. See description above.
- Assumption 3. The caller will not reference the states of bodies not requested in the active_bodies array. The unselected body states are fair game and may or may not be modified by this function.
- Limitation 1. No light speed time-of-travel corrections. Such corrections must be made by the caller if needed.
- Limitation 2. No relativistic time corrections for the difference between Terrestrial and Barycenter Dynamic Time.
- Limitation 3. The states of the sun and the major planets are expressed in ICRF coordinates relative to the Solar System barycenter. The state of the Moon is expressed in ICRF coordinates relative to the center of the Earth.

Parameters

in	*time*	Time since reference Units: s

Definition at line 121 of file de4xx_file_update.cc.

References jeod::De4xxFileItem::active, jeod::De4xxFileItem::avail, jeod::De4xxFileRefTime::block_no, jeod::-De4xxFileCoef::coef, coef, jeod::De4xxFileIO::coeffs_segment_starting_addr, jeod::De4xxFileIO::current_record-_starting_addr, jeod::EphemerisDataSetMeta::delta_epoch, jeod::De4xxFileIO::file, jeod::EphemeridesMessages-::file_error, file_spec, jeod::De4xxFileRefTime::init_time, jeod::EphemeridesMessages::internal_error, interpolate(), io, item, jeod::EphemeridesMessages::item_not_in_file, jeod::De4xxFileIO::metaData, jeod::EphemerisDataSet-Meta::ncoeff, jeod::EphemerisDataSegmentMeta::num_recs, jeod::EphemerisDataSetMeta::number_file_items, jeod::EphemerisDataSetMeta::number_segments, jeod::De4xxFileSpec::pathname, jeod::De4xxFileIO::recno, ref-_time, jeod::De4xxFileIO::segment_index, jeod::De4xxFileIO::segment_recno, jeod::De4xxFileIO::segmentData, jeod::EphemeridesMessages::time_not_in_range, jeod::De4xxFileIO::total_num_recs, and update_time.

Referenced by jeod::De4xxEphemeris::ephem_update().

8.4.4 Friends And Related Function Documentation

8.4.4.1 friend class De4xxFileRestart [friend]

Definition at line 596 of file de4xx_file.hh.

8.4.4.2 void init_attrjeod__De4xxFile ( ) [friend]

8.4.4.3 friend class InputProcessor [friend]

Definition at line 594 of file de4xx_file.hh.

8.4.5 Field Documentation

8.4.5.1 De4xxFileCoef jeod::De4xxFile::coef

Chebychev coefs.

trick_units(–)

Definition at line 656 of file de4xx_file.hh.

Referenced by close(), initialize(), interpolate(), pre_initialize(), and update().

8.4.5.2 De4xxFileSpec jeod::De4xxFile::file_spec

File specification.

trick_units(–)

Definition at line 631 of file de4xx_file.hh.

Referenced by jeod::De4xxEphemeris::activate_nodes(), jeod::De4xxEphemeris::determine_root_node(), jeod::De4xxEphemeris::ephem_activate(), jeod::De4xxEphemeris::ephem_build_tree(), jeod::De4xxEphemeris::ephem_initialize(), jeod::De4xxEphemeris::get_model_number(), initialize(), jeod::De4xxEphemeris::initialize_items(), open(), pre_initialize(), jeod::De4xxEphemeris::set_model_number(), and update().

8.4.5.3 De4xxFileHeader jeod::De4xxFile::header

File header.

trick_units(–)

Definition at line 636 of file de4xx_file.hh.

Referenced by jeod::De4xxEphemeris::ephem_update(), jeod::De4xxEphemeris::get_header_data(), and initialize().

8.4.5.4 De4xxFileIO jeod::De4xxFile::io

File descriptor.

trick_units(–)

Definition at line 646 of file de4xx_file.hh.

Referenced by close(), initialize(), jeod::De4xxEphemeris::initialize_file(), interpolate(), open(), pre_initialize(), reopen(), time_is_in_range(), and update().

8.4.5.5 De4xxFileItem* jeod::De4xxFile::item

Item data.

Sized to fit number of entries in most recent DE4xx releasetrick_units(–)

Definition at line 641 of file de4xx_file.hh.

Referenced by De4xxFile(), jeod::De4xxEphemeris::ephem_activate(), jeod::De4xxEphemeris::ephem_update(), interpolate(), pre_initialize(), update(), and ~De4xxFile().

8.4.5.6 De4xxFileRefTime jeod::De4xxFile::ref_time

Reference time.

trick_units(–)

Definition at line 651 of file de4xx_file.hh.

Referenced by initialize(), time_is_in_range(), and update().

8.4.5.7 De4xxFileRestart jeod::De4xxFile::restart

Restart handler.

trick_io(**)

Definition at line 661 of file de4xx_file.hh.

Referenced by De4xxFile(), and ~De4xxFile().

8.4.5.8 double jeod::De4xxFile::update_time

Time of last update.

trick_units(s)

Definition at line 666 of file de4xx_file.hh.

Referenced by initialize(), and update().

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

- de4xx_file.hh
- de4xx_file.cc
- de4xx_file_init.cc
- de4xx_file_update.cc

8.5 jeod::De4xxFileCoef Class Reference

Contains Chebychev polynomial coefficients and terms.

`#include <de4xx_file.hh>`

Public Member Functions

- De4xxFileCoef (void)
  Construct a De4xxFileFileCoef object.

Protected Attributes

- size_t chebyterms
  No.
- double chebyx
  Chebychev x value.
- double * chebypoly
  Chebychev polynomial coeffs.
- double * chebyderiv
  Derivative of chebypoly.
- double * coef
  Current block contents.

Private Member Functions

- De4xxFileCoef (const De4xxFileCoef &)
- De4xxFileCoef & operator= (const De4xxFileCoef &)

Friends

- class InputProcessor
- class De4xxFile
- void init_attrjeod__De4xxFileCoef ()

8.5.1 Detailed Description

Contains Chebychev polynomial coefficients and terms.

Definition at line 516 of file de4xx_file.hh.

8.5.2 Constructor & Destructor Documentation

8.5.2.1 jeod::De4xxFileCoef::De4xxFileCoef (const De4xxFileCoef &) [private]

8.5.2.2 jeod::De4xxFileCoef::De4xxFileCoef (void)

Construct a De4xxFileFileCoef object.

Definition at line 214 of file de4xx_file.cc.

8.5.3 Member Function Documentation

8.5.3.1 De4xxFileCoef& jeod::De4xxFileCoef::operator= (const De4xxFileCoef &) [private]

8.5.4 Friends And Related Function Documentation

8.5.4.1 friend class De4xxFile [friend]

Definition at line 520 of file de4xx_file.hh.

8.5.4.2 void init_attrjeod__De4xxFileCoef ( ) [friend]

8.5.4.3 friend class InputProcessor [friend]

Definition at line 518 of file de4xx_file.hh.

8.5.5 Field Documentation

8.5.5.1 double* jeod::De4xxFileCoef::chebyderiv [protected]

Derivative of chebypoly.

trick_units(–)

Definition at line 544 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::close(), jeod::De4xxFile::initialize(), and jeod::De4xxFile::interpolate().

8.5.5.2 double* jeod::De4xxFileCoef::chebypoly [protected]

Chebychev polynomial coeffs.

trick_units(–)

Definition at line 539 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::close(), jeod::De4xxFile::initialize(), and jeod::De4xxFile::interpolate().

8.5.5.3 size_t jeod::De4xxFileCoef::chebyterms [protected]

No.

Chebychev polynomials termstrick_units(–)

Definition at line 529 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::interpolate().

8.5.5.4 double jeod::De4xxFileCoef::chebyx [protected]

Chebychev x value.

trick_units(–)

Definition at line 534 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::interpolate().

8.5.5.5 double* jeod::De4xxFileCoef::coef [protected]

Current block contents.

trick_units(–) trick_io(**)

Definition at line 549 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::interpolate(), jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

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

- de4xx_file.hh
- de4xx_file.cc

8.6 jeod::De4xxFileHeader Class Reference

Contains data extracted from the ephemeris file header.

`#include <de4xx_file.hh>`

Public Member Functions

- De4xxFileHeader (void)
  Construct a De4xxFileHeader object.
- ~De4xxFileHeader (void)
  Destruct a De4xxFileHeader object.

Data Fields

- double au
  Astronomical unit in meters.
- double vlight
  Speed of light.

- double em_mass_ratio

  Earth:Moon mass ratio.
- double be_em_dist_ratio

  Ratio of Earth-to-barycenter and Earth-to-Moon distances Note: Also equal to the ratio of Moon and Earth+Moon masses.
- double bm_em_dist_ratio

  Ratio of Barycenter-to-Moon and Earth-to-Moon distances Note: Also equal to the ratio of Earth and Earth+Moon masses.
- double e1_em_dist_ratio

  Ratio of Earth to Earth-moon L1 point and Earth-to-Moon distances.
- double b1_em_dist_ratio

  Ratio of Earth-Moon barycenter to L1 point and Earth-to-Moon distances.
- double * gmbody

  Body gravitational constants.

Private Member Functions

- De4xxFileHeader (const De4xxFileHeader &)
- De4xxFileHeader & operator= (const De4xxFileHeader &)

Friends

- class InputProcessor
- class De4xxFile
- void init_attrjeod__De4xxFileHeader ()

8.6.1 Detailed Description

Contains data extracted from the ephemeris file header.

Definition at line 334 of file de4xx_file.hh.

8.6.2 Constructor & Destructor Documentation

8.6.2.1 jeod::De4xxFileHeader::De4xxFileHeader (const De4xxFileHeader &) [private]

8.6.2.2 jeod::De4xxFileHeader::De4xxFileHeader (void)

Construct a De4xxFileHeader object.

Definition at line 143 of file de4xx_file.cc.

References gmbody, and jeod::De4xxBase::number_grav_models().

8.6.2.3 jeod::De4xxFileHeader::~De4xxFileHeader (void)

Destruct a De4xxFileHeader object.

Definition at line 164 of file de4xx_file.cc.

References gmbody.

8.6.3 Member Function Documentation

8.6.3.1 De4xxFileHeader& jeod::De4xxFileHeader::operator= (const De4xxFileHeader &) [private]

8.6.4 Friends And Related Function Documentation

8.6.4.1 friend class De4xxFile [friend]

Definition at line 338 of file de4xx_file.hh.

8.6.4.2 void init_attrjeod__De4xxFileHeader ( ) [friend]

8.6.4.3 friend class InputProcessor [friend]

Definition at line 336 of file de4xx_file.hh.

8.6.5 Field Documentation

8.6.5.1 double jeod::De4xxFileHeader::au

Astronomical unit in meters.

trick_units(m)

Definition at line 346 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize().

8.6.5.2 double jeod::De4xxFileHeader::b1_em_dist_ratio

Ratio of Earth-Moon barycenter to L1 point and Earth-to-Moon distances.

trick_units(–)

Definition at line 378 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize().

8.6.5.3 double jeod::De4xxFileHeader::be_em_dist_ratio

Ratio of Earth-to-barycenter and Earth-to-Moon distances Note: Also equal to the ratio of Moon and Earth+Moon masses.

trick_units(–)

Definition at line 362 of file de4xx_file.hh.

Referenced by jeod::De4xxEphemeris::ephem_update(), and jeod::De4xxFile::initialize().

8.6.5.4 double jeod::De4xxFileHeader::bm_em_dist_ratio

Ratio of Barycenter-to-Moon and Earth-to-Moon distances Note: Also equal to the ratio of Earth and Earth+Moon masses.

trick_units(–)

Definition at line 368 of file de4xx_file.hh.

Referenced by jeod::De4xxEphemeris::ephem_update(), and jeod::De4xxFile::initialize().

8.6.5.5 double jeod::De4xxFileHeader::e1_em_dist_ratio

Ratio of Earth to Earth-moon L1 point and Earth-to-Moon distances.

trick_units(–)

Definition at line 373 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize().

8.6.5.6 double jeod::De4xxFileHeader::em_mass_ratio

Earth:Moon mass ratio.

trick_units(–)

Definition at line 356 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize().

8.6.5.7 double* jeod::De4xxFileHeader::gmbody

Body gravitational constants.

trick_units(m3/s2)

Definition at line 383 of file de4xx_file.hh.

Referenced by De4xxFileHeader(), jeod::De4xxFile::initialize(), and ~De4xxFileHeader().

8.6.5.8 double jeod::De4xxFileHeader::vlight

Speed of light.

trick_units(m/s)

Definition at line 351 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize().

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

- de4xx_file.hh
- de4xx_file.cc

8.7 jeod::De4xxFileIO Class Reference

Contains data used directly for reading the ephemeris file.

`#include <de4xx_file.hh>`

Public Member Functions

- De4xxFileIO (void)

  Construct a De4xxFileIO object.

Data Fields

- EphemerisDataSetMeta * metaData

Metadata (e.g., sizing) regarding the selected DE ephemeris data set.

- EphemerisDataItemMeta * itemData

  Metadata (e.g., number of terms) regarding each ephemeris item (e.g., Mercury) contained in the JPL data.
- EphemerisDataSegmentMeta * segmentData

  Metadata (e.g., number of records) regarding each polynomial segment of the JPL data.
- double * coeffs_segment_starting_addr

  Pointer to first value in the segment.
- double * current_record_starting_addr

  Pointer to first value in the record.
- uint32_t recno

  The current record number.
- uint32_t segment_index

  The current segment number.
- uint32_t segment_recno

  The current segment record number.
- uint32_t total_num_recs

  The number of records in the dataset.
- uint32_t max_terms

  The maximum number of Chebychev terms in the file.

Protected Attributes

- void * file

  The dl handle for the ephemeris shared object.

Private Member Functions

- De4xxFileIO (const De4xxFileIO &)
- De4xxFileIO & operator= (const De4xxFileIO &)

Friends

- class InputProcessor
- class De4xxFile
- void init_attrjeod__De4xxFileIO ()

8.7.1 Detailed Description

Contains data used directly for reading the ephemeris file.

Definition at line 247 of file de4xx_file.hh.

8.7.2 Constructor & Destructor Documentation

8.7.2.1 jeod::De4xxFileIO::De4xxFileIO (const De4xxFileIO &) [private]

8.7.2.2 jeod::De4xxFileIO::De4xxFileIO (void)

Construct a De4xxFileIO object.

Definition at line 121 of file de4xx_file.cc.

8.7.3 Member Function Documentation

8.7.3.1 De4xxFileIO& jeod::De4xxFileIO::operator= (const De4xxFileIO &) [private]

8.7.4 Friends And Related Function Documentation

8.7.4.1 friend class De4xxFile [friend]

Definition at line 251 of file de4xx_file.hh.

8.7.4.2 void init_attrjeod__De4xxFileIO ( ) [friend]

8.7.4.3 friend class InputProcessor [friend]

Definition at line 249 of file de4xx_file.hh.

8.7.5 Field Documentation

8.7.5.1 double* jeod::De4xxFileIO::coeffs_segment_starting_addr

Pointer to first value in the segment.

trick_units(–) trick_io(**)

Definition at line 276 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

8.7.5.2 double* jeod::De4xxFileIO::current_record_starting_addr

Pointer to first value in the record.

trick_units(–) trick_io(**)

Definition at line 281 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

8.7.5.3 void* jeod::De4xxFileIO::file [protected]

The dl handle for the ephemeris shared object.

trick_units(–) trick_io(**)

Definition at line 313 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::close(), jeod::De4xxFile::open(), jeod::De4xxFile::pre_initialize(), jeod::De4xxFile::reopen(), and jeod::De4xxFile::update().

8.7.5.4 EphemerisDataItemMeta* jeod::De4xxFileIO::itemData

Metadata (e.g., number of terms) regarding each ephemeris item (e.g., Mercury) contained in the JPL data.

trick_units(–) trick_io(**)

Definition at line 265 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::interpolate(), jeod::De4xxFile::open(), and jeod::De4xxFile::pre_initialize().

8.7.5.5 uint32_t jeod::De4xxFileIO::max_terms

The maximum number of Chebychev terms in the file.

trick_units(–)

Definition at line 307 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize(), and jeod::De4xxFile::pre_initialize().

8.7.5.6 EphemerisDataSetMeta* jeod::De4xxFileIO::metaData

Metadata (e.g., sizing) regarding the selected DE ephemeris data set.

trick_units(–) trick_io(**)

Definition at line 259 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::close(), jeod::De4xxFile::initialize(), jeod::De4xxEphemeris::initialize_file(), jeod::De4xxFile::interpolate(), jeod::De4xxFile::open(), jeod::De4xxFile::pre_initialize(), jeod::De4xxFile::time_is_in_range(), and jeod::De4xxFile::update().

8.7.5.7 uint32_t jeod::De4xxFileIO::recno

The current record number.

trick_units(–)

Definition at line 287 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

8.7.5.8 uint32_t jeod::De4xxFileIO::segment_index

The current segment number.

trick_units(–)

Definition at line 292 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

8.7.5.9 uint32_t jeod::De4xxFileIO::segment_recno

The current segment record number.

trick_units(–)

Definition at line 297 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

8.7.5.10 EphemerisDataSegmentMeta* jeod::De4xxFileIO::segmentData

Metadata (e.g., number of records) regarding each polynomial segment of the JPL data.

trick_units(–) trick_io(**)

Definition at line 271 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize(), jeod::De4xxFile::open(), jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

8.7.5.11 uint32_t jeod::De4xxFileIO::total_num_recs

The number of records in the dataset.

trick_units(–)

Definition at line 302 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize(), jeod::De4xxFile::pre_initialize(), jeod::De4xxFile::time_is_in_range(), and jeod::De4xxFile::update().

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

- de4xx_file.hh
- de4xx_file.cc

8.8 jeod::De4xxFileItem Class Reference

Contains data regarding one of the items in a DE ephemeris file.

`#include <de4xx_file.hh>`

Public Member Functions

- De4xxFileItem (void)

  Construct a De4xxFileItem object.

Data Fields

- bool active

  Is this item's state to be computed? (external input)
- bool avail

  Is this item represented in the ephemeris file?
- uint32_t item_idx

  trick_units(–)
- int32_t nitems

  Vector size.
- double pscale

  Zeroth derivative scale factor.
- double update_time

  Update time (simulation time)
- double state [2][3]

  State data (zeroth, first derivative)

Private Member Functions

- De4xxFileItem (const De4xxFileItem &)
- De4xxFileItem & operator= (const De4xxFileItem &)

Friends

- class InputProcessor
- class De4xxFile
- void init_attrjeod__De4xxFileItem ()

8.8.1 Detailed Description

Contains data regarding one of the items in a DE ephemeris file.

Definition at line 404 of file de4xx_file.hh.

8.8.2 Constructor & Destructor Documentation

8.8.2.1 jeod::De4xxFileItem::De4xxFileItem (const De4xxFileItem &) [private]

8.8.2.2 jeod::De4xxFileItem::De4xxFileItem (void)

Construct a De4xxFileItem object.

As most ephemeris file items are position vectors in kilometers, this constructor sets the scale to 1000 and the number of items to three.

Definition at line 177 of file de4xx_file.cc.

References state.

8.8.3 Member Function Documentation

8.8.3.1 De4xxFileItem& jeod::De4xxFileItem::operator= (const De4xxFileItem &) [private]

8.8.4 Friends And Related Function Documentation

8.8.4.1 friend class De4xxFile [friend]

Definition at line 408 of file de4xx_file.hh.

8.8.4.2 void init_attrjeod__De4xxFileItem ( ) [friend]

8.8.4.3 friend class InputProcessor [friend]

Definition at line 406 of file de4xx_file.hh.

8.8.5 Field Documentation

8.8.5.1 bool jeod::De4xxFileItem::active

Is this item's state to be computed? (external input)

trick_units(–)

Definition at line 416 of file de4xx_file.hh.

Referenced by jeod::De4xxEphemeris::ephem_activate(), jeod::De4xxFile::interpolate(), and jeod::De4xxFile::update().

8.8.5.2 bool jeod::De4xxFileItem::avail

Is this item represented in the ephemeris file?

trick_units(–)

Definition at line 421 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

8.8.5.3 uint32_t jeod::De4xxFileItem::item_idx

trick_units(–)

Definition at line 424 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::interpolate(), and jeod::De4xxFile::pre_initialize().

8.8.5.4 int32_t jeod::De4xxFileItem::nitems

Vector size.

trick_units(–)

Definition at line 429 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::De4xxFile(), and jeod::De4xxFile::interpolate().

8.8.5.5 double jeod::De4xxFileItem::pscale

Zeroth derivative scale factor.

trick_units(–)

Definition at line 434 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::De4xxFile(), and jeod::De4xxFile::interpolate().

8.8.5.6 double jeod::De4xxFileItem::state[2][3]

State data (zeroth, first derivative)

trick_units(–)

Definition at line 444 of file de4xx_file.hh.

Referenced by De4xxFileItem(), jeod::De4xxEphemeris::ephem_update(), and jeod::De4xxFile::interpolate().

8.8.5.7 double jeod::De4xxFileItem::update_time

Update time (simulation time)

trick_units(s)

Definition at line 439 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::interpolate().

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

- de4xx_file.hh
- de4xx_file.cc

8.9 jeod::De4xxFileRefTime Class Reference

Contains timing reference data.

`#include <de4xx_file.hh>`

Public Member Functions

- De4xxFileRefTime (void)

  Construct a De4xxFileRefTime object.

Data Fields

- double epoch_date

  Julian date of midnight preceding reference time point.
- double fdate

  Fractional days past epoch date of reference time point.
- double time_offset

  Time offset, Typically, Terrestrial Time offset.
- double init_time

  Initialization time (seconds from reference, typically zero).
- double block_no

  File block number corresponding to reference time.

Private Member Functions

- De4xxFileRefTime (const De4xxFileRefTime &)
- De4xxFileRefTime & operator= (const De4xxFileRefTime &)

Friends

- class InputProcessor
- class De4xxFile
- void init_attrjeod__De4xxFileRefTime ()

8.9.1 Detailed Description

Contains timing reference data.

Definition at line 464 of file de4xx_file.hh.

8.9.2 Constructor & Destructor Documentation

8.9.2.1 jeod::De4xxFileRefTime::De4xxFileRefTime (const De4xxFileRefTime &) [private]

8.9.2.2 jeod::De4xxFileRefTime::De4xxFileRefTime (void)

Construct a De4xxFileRefTime object.

Definition at line 198 of file de4xx_file.cc.

8.9.3 Member Function Documentation

8.9.3.1 De4xxFileRefTime& jeod::De4xxFileRefTime::operator= (const De4xxFileRefTime &) [private]

8.9.4 Friends And Related Function Documentation

8.9.4.1 friend class De4xxFile [friend]

Definition at line 468 of file de4xx_file.hh.

8.9.4.2 void init_attrjeod__De4xxFileRefTime ( ) [friend]

8.9.4.3 friend class InputProcessor [friend]

Definition at line 466 of file de4xx_file.hh.

8.9.5 Field Documentation

8.9.5.1 double jeod::De4xxFileRefTime::block_no

File block number corresponding to reference time.

trick_units(–)

Definition at line 496 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize(), jeod::De4xxFile::time_is_in_range(), and jeod::De4xxFile::update().

8.9.5.2 double jeod::De4xxFileRefTime::epoch_date

Julian date of midnight preceding reference time point.

trick_units(day)

Definition at line 476 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize().

8.9.5.3 double jeod::De4xxFileRefTime::fdate

Fractional days past epoch date of reference time point.

trick_units(day)

Definition at line 481 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize().

8.9.5.4 double jeod::De4xxFileRefTime::init_time

Initialization time (seconds from reference, typically zero).

trick_units(s)

Definition at line 489 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize(), jeod::De4xxFile::time_is_in_range(), and jeod::De4xxFile::update().

8.9.5.5 double jeod::De4xxFileRefTime::time_offset

Time offset, Typically, Terrestrial Time offset.

trick_units(s)

Definition at line 485 of file de4xx_file.hh.

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

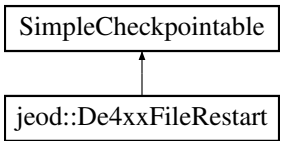
- de4xx_file.hh
- de4xx_file.cc

8.10 jeod::De4xxFileRestart Class Reference

The FILE pointer in a De4xxFileIO cannot be restored by Trick.

`#include <de4xx_file.hh>`

Inheritance diagram for jeod::De4xxFileRestart:



Public Member Functions

- De4xxFileRestart (De4xxFile &in)

  Construct a De4xxFileRestart object.
- virtual ~De4xxFileRestart (void)

  Destroy a De4xxFileRestart object.
- virtual void simple_restore (void)

  Reopen the De4xx file for a restart.

Protected Attributes

- De4xxFile & de4xx_file

  The De4xxFile object to be restored.

Private Member Functions

- De4xxFileRestart (const De4xxFileRestart &)
- De4xxFileRestart & operator= (const De4xxFileRestart &)

8.10.1 Detailed Description

The FILE pointer in a De4xxFileIO cannot be restored by Trick.

This class provides that essential restart mechanism.

Definition at line 570 of file de4xx_file.hh.

8.10.2 Constructor & Destructor Documentation

8.10.2.1 jeod::De4xxFileRestart::De4xxFileRestart (De4xxFile & *in*) [explicit]

Construct a De4xxFileRestart object.

Parameters

in,out	*in*	The De4xxFile object

Definition at line 232 of file de4xx_file.cc.

8.10.2.2 jeod::De4xxFileRestart::~De4xxFileRestart (void) [virtual]

Destroy a De4xxFileRestart object.

Definition at line 244 of file de4xx_file.cc.

8.10.2.3 jeod::De4xxFileRestart::De4xxFileRestart (const De4xxFileRestart &) [private]

8.10.3 Member Function Documentation

8.10.3.1 De4xxFileRestart& jeod::De4xxFileRestart::operator= (const De4xxFileRestart &) [private]

8.10.3.2 void jeod::De4xxFileRestart::simple_restore (void) [virtual]

Reopen the De4xx file for a restart.

Definition at line 255 of file de4xx_file.cc.

References de4xx_file, and jeod::De4xxFile::reopen().

8.10.4 Field Documentation

8.10.4.1 De4xxFile& jeod::De4xxFileRestart::de4xx_file [protected]

The De4xxFile object to be restored.

trick_io(**)

Definition at line 582 of file de4xx_file.hh.

Referenced by simple_restore().

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

- de4xx_file.hh
- de4xx_file.cc

8.11 jeod::De4xxFileSpec Class Reference

Specifies which file to use (user input initialization-time data).

`#include <de4xx_file.hh>`

Public Member Functions

- De4xxFileSpec (void)
  Construct a De4xxFileSpec object.
- void set_model_number (int denum_in)
  Set ephemeris model number.
- uint32_t get_model_number ()
  Get Ephemeris model number.

Protected Attributes

- uint32_t denum

  Ephemeris model number.
- char ephem_file_dir [256]

  Ephemeris file directory.
- char ephem_file_name [256]

  Ephemeris file name.
- char pathname [256]

  Ephemeris file path name.

Private Member Functions

- De4xxFileSpec (const De4xxFileSpec &)

  Not implemented.
- De4xxFileSpec & operator= (const De4xxFileSpec &)

  Not implemented.

Friends

- class InputProcessor
- class De4xxFile
- void init_attrjeod__De4xxFileSpec ()

8.11.1 Detailed Description

Specifies which file to use (user input initialization-time data).

Definition at line 169 of file de4xx_file.hh.

8.11.2 Constructor & Destructor Documentation

8.11.2.1 jeod::De4xxFileSpec::De4xxFileSpec (void)

Construct a De4xxFileSpec object.

Definition at line 96 of file de4xx_file.cc.

References ephem_file_dir, and set_model_number().

8.11.2.2 jeod::De4xxFileSpec::De4xxFileSpec (const De4xxFileSpec &) [private]

Not implemented.

8.11.3 Member Function Documentation

8.11.3.1 uint32_t jeod::De4xxFileSpec::get_model_number ( ) [inline]

Get Ephemeris model number.

This number is used to specify the de file to use the pathname is of the form PWD/build/de4xx_lib/libde<denum-In>.so

Definition at line 194 of file de4xx_file.hh.

References denum.

Referenced by jeod::De4xxEphemeris::activate_nodes(), jeod::De4xxEphemeris::determine_root_node(), jeod::De4xxEphemeris::ephem_activate(), jeod::De4xxEphemeris::ephem_build_tree(), jeod::De4xxEphemeris::ephem_initialize(), jeod::De4xxEphemeris::get_model_number(), and jeod::De4xxEphemeris::initialize_items().

8.11.3.2 De4xxFileSpec& jeod::De4xxFileSpec::operator= (const De4xxFileSpec &) [private]

Not implemented.

8.11.3.3 void jeod::De4xxFileSpec::set_model_number (int *denum_in*)

Set ephemeris model number.

This number is used to specify the de file to use the pathname is of the form PWD/build/de4xx_lib/libde<denum-In>.so

Definition at line 108 of file de4xx_file.cc.

References denum, ephem_file_dir, ephem_file_name, and pathname.

Referenced by De4xxFileSpec(), and jeod::De4xxEphemeris::set_model_number().

8.11.4 Friends And Related Function Documentation

8.11.4.1 friend class De4xxFile [friend]

Definition at line 172 of file de4xx_file.hh.

8.11.4.2 void init_attrjeod__De4xxFileSpec ( ) [friend]

8.11.4.3 friend class InputProcessor [friend]

Definition at line 170 of file de4xx_file.hh.

8.11.5 Field Documentation

8.11.5.1 uint32_t jeod::De4xxFileSpec::denum [protected]

Ephemeris model number.

This must match the DE number in the data file; a sanity checktrick_units(–)

Definition at line 207 of file de4xx_file.hh.

Referenced by get_model_number(), jeod::De4xxFile::initialize(), jeod::De4xxFile::pre_initialize(), and set_model_number().

8.11.5.2 char jeod::De4xxFileSpec::ephem_file_dir[256] [protected]

Ephemeris file directory.

trick_units(–)

Definition at line 212 of file de4xx_file.hh.

Referenced by De4xxFileSpec(), jeod::De4xxFile::open(), and set_model_number().

8.11.5.3 char jeod::De4xxFileSpec::ephem_file_name[256] [protected]

Ephemeris file name.

trick_units(–)

Definition at line 217 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::open(), and set_model_number().

8.11.5.4 char jeod::De4xxFileSpec::pathname[256] [protected]

Ephemeris file path name.

trick_io(∗o) trick_units(–)

Definition at line 223 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::open(), jeod::De4xxFile::pre_initialize(), set_model_number(), and jeod::De4xxFile::update().

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

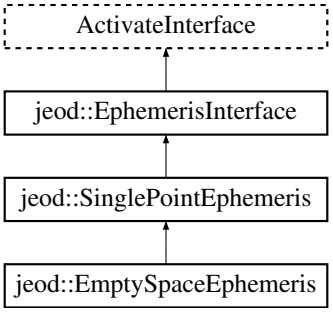
- de4xx_file.hh
- de4xx_file.cc

8.12 jeod::EmptySpaceEphemeris Class Reference

Empty space has one ephemeris point.

`#include <simple_ephemerides.hh>`

Inheritance diagram for jeod::EmptySpaceEphemeris:



Public Member Functions

- EmptySpaceEphemeris (void)

  Construct an EmptySpaceEphemeris object.
- virtual ∼EmptySpaceEphemeris (void)

  Destruct an EmptySpaceEphemeris object.
- virtual void set_name (const char ∗frame_name)

  Set the name of an EmptySpaceEphemeris object.
- virtual void initialize_model (EphemeridesManager &ephem_manager)

  Initialize an EmptySpaceEphemeris object.
- virtual void ephem_initialize (EphemeridesManager &ephem_manager)

  Initialize an EmptySpaceEphemeris object.

- virtual void ephem_activate (EphemeridesManager &ephem_manager)

  Activate an EmptySpaceEphemeris object.
- virtual void ephem_build_tree (EphemeridesManager &ephem_manager)

  Build the reference frame tree with the central frame as the root.

Protected Attributes

- EphemerisPoint central_point

  The EphemerisPoint that represents the center of an empty universe.
- EphemerisRefFrame central_frame

  The sole ephemeris frame for this model.

Private Member Functions

- EmptySpaceEphemeris (const EmptySpaceEphemeris &)

  Not implemented.
- EmptySpaceEphemeris & operator= (const EmptySpaceEphemeris &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__EmptySpaceEphemeris ()

8.12.1 Detailed Description

Empty space has one ephemeris point.

Definition at line 169 of file simple_ephemerides.hh.

8.12.2 Constructor & Destructor Documentation

8.12.2.1 jeod::EmptySpaceEphemeris::EmptySpaceEphemeris (void)

Construct an EmptySpaceEphemeris object.

Definition at line 156 of file simple_ephemerides.cc.

References central_point, jeod::EphemerisItem::enable(), and jeod::EphemerisItem::set_owner().

8.12.2.2 jeod::EmptySpaceEphemeris::~EmptySpaceEphemeris (void) [virtual]

Destruct an EmptySpaceEphemeris object.

Definition at line 167 of file simple_ephemerides.cc.

8.12.2.3 jeod::EmptySpaceEphemeris::EmptySpaceEphemeris (const EmptySpaceEphemeris &) [private]

Not implemented.

8.12.3 Member Function Documentation

8.12.3.1 void jeod::EmptySpaceEphemeris::ephem_activate (EphemeridesManager & *ephem_manager*) [virtual]

Activate an EmptySpaceEphemeris object.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::SinglePointEphemeris.

Definition at line 249 of file simple_ephemerides.cc.

8.12.3.2 void jeod::EmptySpaceEphemeris::ephem_build_tree (EphemeridesManager & *ephem_manager*) [virtual]

Build the reference frame tree with the central frame as the root.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::SinglePointEphemeris.

Definition at line 261 of file simple_ephemerides.cc.

References jeod::SinglePointEphemeris::active, and central_frame.

8.12.3.3 void jeod::EmptySpaceEphemeris::ephem_initialize (EphemeridesManager & *ephem_manager*) [virtual]

Initialize an EmptySpaceEphemeris object.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::SinglePointEphemeris.

Definition at line 226 of file simple_ephemerides.cc.

References central_point, jeod::SinglePointEphemeris::deactivate(), jeod::EphemerisItem::disable(), jeod::EphemerisItem::get_target_frame(), jeod::SinglePointEphemeris::identifier, and jeod::EphemeridesMessages::inconsistent_setup.

8.12.3.4 void jeod::EmptySpaceEphemeris::initialize_model (EphemeridesManager & *ephem_manager*) [virtual]

Initialize an EmptySpaceEphemeris object.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::SinglePointEphemeris.

Definition at line 206 of file simple_ephemerides.cc.

References jeod::SinglePointEphemeris::active, jeod::EphemeridesManager::add_ephem_item(), jeod::EphemeridesManager::add_ephemeris(), jeod::EphemeridesManager::add_integ_frame(), central_frame, and central_point.

8.12.3.5 EmptySpaceEphemeris& jeod::EmptySpaceEphemeris::operator= (const EmptySpaceEphemeris &) [private]

Not implemented.

8.12.3.6 void jeod::EmptySpaceEphemeris::set_name (const char * *new_name*) [virtual]

Set the name of an EmptySpaceEphemeris object.

Parameters

in,out	*new_name*	Ephemeris name

Reimplemented from jeod::SinglePointEphemeris.

Definition at line 179 of file simple_ephemerides.cc.

References central_frame, central_point, jeod::SinglePointEphemeris::set_name(), and jeod::EphemerisItem::set_name().

8.12.4 Friends And Related Function Documentation

8.12.4.1 void init_attrjeod__EmptySpaceEphemeris ( ) [friend]

8.12.4.2 friend class InputProcessor [friend]

Definition at line 171 of file simple_ephemerides.hh.

8.12.5 Field Documentation

8.12.5.1 EphemerisRefFrame jeod::EmptySpaceEphemeris::central_frame [protected]

The sole ephemeris frame for this model.

trick_units(–)

Definition at line 205 of file simple_ephemerides.hh.

Referenced by ephem_build_tree(), initialize_model(), and set_name().

8.12.5.2 EphemerisPoint jeod::EmptySpaceEphemeris::central_point [protected]

The EphemerisPoint that represents the center of an empty universe.

trick_units(–)

Definition at line 200 of file simple_ephemerides.hh.

Referenced by EmptySpaceEphemeris(), ephem_initialize(), initialize_model(), and set_name().

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

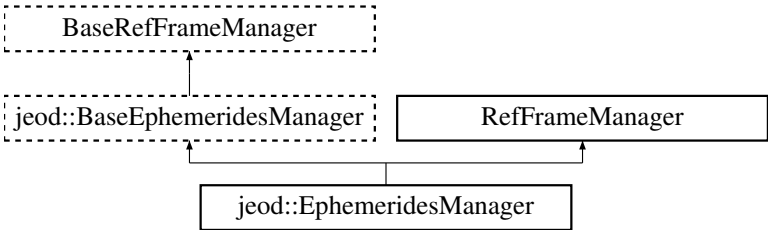
- simple_ephemerides.hh
- simple_ephemerides.cc

8.13 jeod::EphemeridesManager Class Reference

The EphemeridesManager class manages the ephemeris models in a simulation.

`#include <ephem_manager.hh>`

Inheritance diagram for jeod::EphemeridesManager:



Public Member Functions

- EphemeridesManager ()

  EphemeridesManager default constructor.
- ~EphemeridesManager ()

  EphemeridesManager destructor.
- bool ref_frame_tree_needs_rebuild () const

  Query if the reference frame tree needs to be rebuilt.
- virtual void ephem_note_tree_status_change ()

  Denote that the reference frame tree needs to be rebuilt.
- virtual void add_planet (BasePlanet &planet)

  Add a planet to the planets registry.
- virtual void add_planet (Planet &planet)

  Add a planet to the registry.
- virtual BasePlanet * find_base_planet (const char *name) const

  Find the planet with the given name.
- virtual Planet * find_planet (const char *name) const

  Find the planet with the given name.
- virtual unsigned int get_num_planets (void) const

  Return number of registered planets.
- virtual void add_ephemeris (EphemerisInterface &ephem_if)

  Add an ephemeris model to the list of managed models.
- virtual void clear_added_ephemerides (void)

  Deactivate any registered ephemeris items and remove them from the list.
- virtual void disable_add_ephemeris (void)

  Make subsequent calls to add_ephemeris deactivate the specified ephemeris model instead of adding it to the list of managed models.
- virtual void add_ephem_item (EphemerisItem &ephem_item)

  Add an ephemeris item to the set of ephemeris items known to the ephemerides manager.
- virtual EphemerisItem * find_ephem_item (const char *name) const

  Find the first registered EphemerisItem with the given name.
- virtual EphemerisOrientation * find_ephem_angle (const char *name) const

  Find the EphemerisOrientation with the given name.
- virtual EphemerisPoint * find_ephem_point (const char *name) const

  Find the EphemerisPoint with the given name.
- virtual void add_integ_frame (EphemerisRefFrame &ref_frame)

  Add a frame to the reference frame and integration frame lists.
- virtual EphemerisRefFrame * find_integ_frame (const char *name) const

  Find the integration frame with the given name.
- virtual bool is_integ_frame (const RefFrame &ref_frame) const

  Determine if supplied frame is an integration frame.
- virtual unsigned int find_integ_frame_index (const EphemerisRefFrame &ref_frame) const

Find the index of provided frame in the integration frames vector.

- virtual const std::vector < EphemerisRefFrame * > & get_integ_frames (void) const

  Get a copy of the vector of integration frames.

- virtual void add_ref_frame (RefFrame &ref_frame)

  Add a reference frame to the reference frame registry.

- void set_target_frame (RefFrame &ref_frame)

  Set the target-frame reference for the ref-frame and all known EphemItems with similar target-frame names.

- void initialize_ephemerides (void)

  Initialize the ephemeris models.

- void activate_ephemerides (void)

  Activate ephemeris items based on frame subscription status, activate ephemeris models, and build the reference frame tree.

- void update_ephemerides (void)

  Update each ephemeris model.

Protected Attributes

- bool single_ephem_mode

  Set via a call to disable_add_ephemeris, typically to allow a simple ephemeris model to be active with all other models made inactive.

- bool regenerate_ref_frame_tree

  Set when the reference frame tree needs to be regenerated.

- double update_time

  Time of last update.

- JeodPointerVector< BasePlanet > ::type planets

  The planets in a simulation, typically defined at the S_define level.

- JeodPointerVector < EphemerisInterface >::type ephemerides

  The ephemerides models managed by this EphemeridesManager.

- JeodPointerVector < EphemerisItem >::type ephem_items

  The heads of the ephemeris item lists.

- JeodPointerVector < EphemerisRefFrame >::type integ_frames

  List of reference frames that are not rotating with respect to the root node of the reference frame tree.

Private Member Functions

- EphemeridesManager (const EphemeridesManager &)

  Not implemented.

- EphemeridesManager & operator= (const EphemeridesManager &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__EphemeridesManager ()

8.13.1 Detailed Description

The EphemeridesManager class manages the ephemeris models in a simulation.

The primary functions of a EphemeridesManager are to:

- Maintain lists of planets, ephemeris models, ephemeris items, and integration frames, and to provide lookup methods for these lists.
- Dynamically determine which ephemerides are needed in a simulation.
- Initialize ephemeris models and keep them in sync with the rest of the simulation.

Definition at line 62 of file ephem_manager.hh.

8.13.2 Constructor & Destructor Documentation

8.13.2.1 jeod::EphemeridesManager::EphemeridesManager (void)

EphemeridesManager default constructor.

Definition at line 67 of file ephem_manager.cc.

References ephem_items, ephemerides, integ_frames, and planets.

8.13.2.2 jeod::EphemeridesManager::~EphemeridesManager (void)

EphemeridesManager destructor.

Definition at line 91 of file ephem_manager.cc.

References ephem_items, ephemerides, integ_frames, and planets.

8.13.2.3 jeod::EphemeridesManager::EphemeridesManager (const EphemeridesManager &) [private]

Not implemented.

8.13.3 Member Function Documentation

8.13.3.1 void jeod::EphemeridesManager::activate_ephemerides (void)

Activate ephemeris items based on frame subscription status, activate ephemeris models, and build the reference frame tree.

Definition at line 715 of file ephem_manager.cc.

References jeod::EphemerisItem::disconnect_from_tree(), jeod::EphemerisInterface::ephem_activate(), jeod::EphemerisInterface::ephem_build_tree(), ephem_items, ephemerides, and regenerate_ref_frame_tree.

Referenced by update_ephemerides().

8.13.3.2 void jeod::EphemeridesManager::add_ephem_item (EphemerisItem & *ephem_item*) [virtual]

Add an ephemeris item to the set of ephemeris items known to the ephemerides manager.

Each ephemeris model calls this method for each ephemeris item represented by that ephemeris model. The provided item is added to

- The ephem_items list if the provided item's name is unique or

- The tail of the list of items with the same name if an item with the provided item's name is already in the ephem_items list.

Multiple ephemerides models might represent the same conceptual item. The ephem_items list contains only items with unique names. These head items link to other items with the same name via the items' next data members, which form a singly-linked list of commonly-named items.

Assumptions and limitations:

- The item must have a valid name.
- Only one item with a given name can be enabled.

Parameters

ephem_item	Ephemeris item to be added to the registry.

Implements jeod::BaseEphemeridesManager.

Definition at line 304 of file ephem_manager.cc.

References jeod::EphemerisItem::disable(), jeod::EphemeridesMessages::duplicate_entry, ephem_items, find_ephem_item(), jeod::EphemerisItem::get_enabled_item(), jeod::EphemerisInterface::get_name(), jeod::EphemerisItem::get_name(), jeod::EphemerisItem::get_next(), jeod::EphemerisItem::get_owner(), jeod::EphemeridesMessages::inconsistent_setup, jeod::EphemeridesMessages::internal_error, jeod::EphemerisItem::is_enabled(), jeod::EphemerisItem::set_head(), jeod::EphemerisItem::set_manager(), jeod::EphemerisItem::set_next(), jeod::EphemerisItem::set_target_frame(), jeod::EphemeridesMessages::single_ephem_mode, single_ephem_mode, jeod::EphemerisItem::Translation, and jeod::EphemerisItem::updates_what().

Referenced by jeod::De4xxEphemeris::initialize_items(), jeod::EmptySpaceEphemeris::initialize_model(), and jeod::SinglePlanetEphemeris::initialize_model().

8.13.3.3 void jeod::EphemeridesManager::add_ephemeris (EphemerisInterface & *ephem_if*) [virtual]

Add an ephemeris model to the list of managed models.

Assumptions and limitations:

- Ephemeris models must be registered with the ephemerides manager in dependency order: Models with no dependencies are registered first, followed by models that depend on these base models, and so on.

Parameters

ephem_if	Ephemeris model to be added to the registry.

Implements jeod::BaseEphemeridesManager.

Definition at line 240 of file ephem_manager.cc.

References ephemerides.

Referenced by jeod::EmptySpaceEphemeris::initialize_model(), jeod::De4xxEphemeris::initialize_model(), and jeod::SinglePlanetEphemeris::initialize_model().

8.13.3.4 void jeod::EphemeridesManager::add_integ_frame (EphemerisRefFrame & *ref_frame*) [virtual]

Add a frame to the reference frame and integration frame lists.

Parameters

ref_frame	Integration frame to be added to the registries

Implements jeod::BaseEphemeridesManager.

Definition at line 475 of file ephem_manager.cc.

References add_ref_frame(), and integ_frames.

Referenced by jeod::De4xxEphemeris::initialize_items(), and jeod::EmptySpaceEphemeris::initialize_model().

8.13.3.5 void jeod::EphemeridesManager::add_planet (BasePlanet & *planet*) [virtual]

Add a planet to the planets registry.

Parameters

planet	Planet to be added to the registry.

Implements jeod::BaseEphemeridesManager.

Definition at line 128 of file ephem_manager.cc.

References jeod::EphemeridesMessages::duplicate_entry, find_base_planet(), and planets.

Referenced by add_planet().

8.13.3.6 void jeod::EphemeridesManager::add_planet (Planet & *planet*) [virtual]

Add a planet to the registry.

Parameters

planet	Planet to be added to the registry.

Implements jeod::BaseEphemeridesManager.

Definition at line 90 of file find_planet.cc.

References add_planet().

8.13.3.7 void jeod::EphemeridesManager::add_ref_frame (RefFrame & *ref_frame*) [virtual]

Add a reference frame to the reference frame registry.

Parameters

ref_frame	Reference frame to be added to the registry

Definition at line 602 of file ephem_manager.cc.

References set_target_frame().

Referenced by add_integ_frame().

8.13.3.8 void jeod::EphemeridesManager::clear_added_ephemerides (void) [virtual]

Deactivate any registered ephemeris items and remove them from the list.

Implements jeod::BaseEphemeridesManager.

Definition at line 253 of file ephem_manager.cc.

References jeod::EphemerisItem::disable(), ephem_items, ephemerides, and jeod::EphemeridesMessages::single_ephem_mode.

8.13.3.9 void jeod::EphemeridesManager::disable_add_ephemeris (void) [virtual]

Make subsequent calls to add_ephemeris deactivate the specified ephemeris model instead of adding it to the list of managed models.

Note

This is an irrevocable act.

Assumptions and limitations:

- This method is typically used after clearing the ephemerides model list and then adding the one allowed ephermides model:

```
EphemeridesManager::clear_added_ephemerides();
EphemeridesManager::add_ephemeris (model);
EphemeridesManager::add_ephem_item (item);
EphemeridesManager::disable_add_ephemeris ();
```

Implements jeod::BaseEphemeridesManager.

Definition at line 222 of file ephem_manager.cc.

References single_ephem_mode.

8.13.3.10 void jeod::EphemeridesManager::ephem_note_tree_status_change (void) [virtual]

Denote that the reference frame tree needs to be rebuilt.

Implements jeod::BaseEphemeridesManager.

Definition at line 110 of file ephem_manager.cc.

References regenerate_ref_frame_tree.

8.13.3.11 BasePlanet * jeod::EphemeridesManager::find_base_planet (const char * *name*) const [virtual]

Find the planet with the given name.

Parameters

name	Planet name.

Returns

Found planet; NULL if not found.

Implements jeod::BaseEphemeridesManager.

Definition at line 167 of file ephem_manager.cc.

References planets.

Referenced by add_planet(), jeod::SinglePlanetEphemeris::ephem_initialize(), jeod::PropagatedPlanet::ephem_initialize(), and find_planet().

8.13.3.12 EphemerisOrientation * jeod::EphemeridesManager::find_ephem_angle (const char * *name*) const [virtual]

Find the EphemerisOrientation with the given name.

Parameters

name	Ephemeris angle name

Returns

Found ephemeris angle

Implements jeod::BaseEphemeridesManager.

Definition at line 421 of file ephem_manager.cc.

References find_ephem_item(), and jeod::EphemeridesMessages::invalid_item.

8.13.3.13 EphemerisItem * jeod::EphemeridesManager::find_ephem_item (const char * *name*) const [virtual]

Find the first registered EphemerisItem with the given name.

Parameters

name	Ephemeris item name

Returns

Found ephemeris item

Implements jeod::BaseEphemeridesManager.

Definition at line 396 of file ephem_manager.cc.

References ephem_items, and jeod::EphemerisItem::get_name().

Referenced by add_ephem_item(), find_ephem_angle(), find_ephem_point(), and set_target_frame().

8.13.3.14 EphemerisPoint * jeod::EphemeridesManager::find_ephem_point (const char * *name*) const [virtual]

Find the EphemerisPoint with the given name.

Parameters

name	Ephemeris point name

Returns

Found ephemeris point

Implements jeod::BaseEphemeridesManager.

Definition at line 446 of file ephem_manager.cc.

References find_ephem_item(), and jeod::EphemeridesMessages::invalid_item.

8.13.3.15 EphemerisRefFrame * jeod::EphemeridesManager::find_integ_frame (const char * *name*) const [virtual]

Find the integration frame with the given name.

Parameters

name	Integration frame name

Returns

Found integration frame

Implements jeod::BaseEphemeridesManager.

Definition at line 494 of file ephem_manager.cc.

References integ_frames.

8.13.3.16 unsigned int jeod::EphemeridesManager::find_integ_frame_index (const EphemerisRefFrame & *ref_frame*) const [virtual]

Find the index of provided frame in the integration frames vector.

Parameters

ref_frame	Reference to be found

Returns

Index of found frame

Implements jeod::BaseEphemeridesManager.

Definition at line 559 of file ephem_manager.cc.

References integ_frames, and jeod::EphemeridesMessages::invalid_item.

8.13.3.17 Planet * jeod::EphemeridesManager::find_planet (const char * *name*) const [virtual]

Find the planet with the given name.

Parameters

name	Planet name

Returns

Found planet, as a Planet rather than a BasePlanet

Implements jeod::BaseEphemeridesManager.

Definition at line 54 of file find_planet.cc.

References find_base_planet(), and jeod::EphemeridesMessages::invalid_item.

8.13.3.18 const std::vector< EphemerisRefFrame * > & jeod::EphemeridesManager::get_integ_frames (void) const [virtual]

Get a copy of the vector of integration frames.

Returns

Copy of integration frames vector

Implements jeod::BaseEphemeridesManager.

Definition at line 519 of file ephem_manager.cc.

References integ_frames.

8.13.3.19 unsigned int jeod::EphemeridesManager::get_num_planets (void) const [virtual]

Return number of registered planets.

Returns

: Number of registered planets.

Implements jeod::BaseEphemeridesManager.

Definition at line 194 of file ephem_manager.cc.

References planets.

Referenced by jeod::SinglePlanetEphemeris::ephem_initialize().

8.13.3.20 void jeod::EphemeridesManager::initialize_ephemerides (void)

Initialize the ephemeris models.

Definition at line 668 of file ephem_manager.cc.

References jeod::EphemerisInterface::ephem_initialize(), ephemerides, and regenerate_ref_frame_tree.

8.13.3.21 bool jeod::EphemeridesManager::is_integ_frame (const RefFrame & *ref_frame*) const [virtual]

Determine if supplied frame is an integration frame.

Parameters

ref_frame	Reference frame to test

Returns

True if the frame is a registered integration frame, false otherwise

Implements jeod::BaseEphemeridesManager.

Definition at line 533 of file ephem_manager.cc.

References integ_frames.

8.13.3.22 EphemeridesManager& jeod::EphemeridesManager::operator= (const EphemeridesManager &) [private]

Not implemented.

8.13.3.23 bool jeod::EphemeridesManager::ref_frame_tree_needs_rebuild ( ) const [inline]

Query if the reference frame tree needs to be rebuilt.

Returns

regenerate_ref_frame_tree data member.

Definition at line 87 of file ephem_manager.hh.

References regenerate_ref_frame_tree.

8.13.3.24 void jeod::EphemeridesManager::set_target_frame (RefFrame & *ref_frame*)

Set the target-frame reference for the ref-frame and all known EphemItems with similar target-frame names.

Parameters

ref_frame	Reference frame to be used as the target-frame.

Definition at line 617 of file ephem_manager.cc.

References find_ephem_item(), jeod::EphemeridesMessages::inconsistent_setup, jeod::EphemerisRefFrame::set_ephem_manager(), and jeod::EphemerisItem::set_target_frame().

Referenced by add_ref_frame(), and jeod::PropagatedPlanet::ephem_initialize().

8.13.3.25 void jeod::EphemeridesManager::update_ephemerides (void)

Update each ephemeris model.

Definition at line 689 of file ephem_manager.cc.

References activate_ephemerides(), jeod::EphemerisInterface::ephem_update(), ephemerides, and regenerate_ref_frame_tree.

8.13.4 Friends And Related Function Documentation

8.13.4.1 void init_attrjeod__EphemeridesManager ( ) [friend]

8.13.4.2 friend class InputProcessor [friend]

Definition at line 66 of file ephem_manager.hh.

8.13.5 Field Documentation

8.13.5.1 JeodPointerVector<EphemerisItem>::type jeod::EphemeridesManager::ephem_items [protected]

The heads of the ephemeris item lists.

All items in one of these sublists share the same name. The members of the ephem_items list have distinct names.-trick_io(**)

Definition at line 217 of file ephem_manager.hh.

Referenced by activate_ephemerides(), add_ephem_item(), clear_added_ephemerides(), EphemeridesManager(), find_ephem_item(), and ~EphemeridesManager().

8.13.5.2 JeodPointerVector<EphemerisInterface>::type jeod::EphemeridesManager::ephemerides [protected]

The ephemerides models managed by this EphemeridesManager.

trick_io(**)

Definition at line 210 of file ephem_manager.hh.

Referenced by activate_ephemerides(), add_ephemeris(), clear_added_ephemerides(), EphemeridesManager(), initialize_ephemerides(), update_ephemerides(), and ~EphemeridesManager().

8.13.5.3 JeodPointerVector<EphemerisRefFrame>::type jeod::EphemeridesManager::integ_frames [protected]

List of reference frames that are not rotating with respect to the root node of the reference frame tree.

trick_io(**)

Definition at line 223 of file ephem_manager.hh.

Referenced by add_integ_frame(), EphemeridesManager(), find_integ_frame(), find_integ_frame_index(), get_integ_frames(), is_integ_frame(), and ~EphemeridesManager().

8.13.5.4 JeodPointerVector<BasePlanet>::type jeod::EphemeridesManager::planets [protected]

The planets in a simulation, typically defined at the S_define level.

trick_io(**)

Definition at line 205 of file ephem_manager.hh.

Referenced by add_planet(), EphemeridesManager(), find_base_planet(), get_num_planets(), and ~EphemeridesManager().

8.13.5.5 bool jeod::EphemeridesManager::regenerate_ref_frame_tree [protected]

Set when the reference frame tree needs to be regenerated.

trick_units(–)

Definition at line 195 of file ephem_manager.hh.

Referenced by activate_ephemerides(), ephem_note_tree_status_change(), initialize_ephemerides(), ref_frame_tree_needs_rebuild(), and update_ephemerides().

8.13.5.6 bool jeod::EphemeridesManager::single_ephem_mode [protected]

Set via a call to disable_add_ephemeris, typically to allow a simple ephemeris model to be active with all other models made inactive.

trick_units(–)

Definition at line 190 of file ephem_manager.hh.

Referenced by add_ephem_item(), and disable_add_ephemeris().

8.13.5.7 double jeod::EphemeridesManager::update_time [protected]

Time of last update.

trick_units(s)

Definition at line 200 of file ephem_manager.hh.

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

- ephem_manager.hh
- ephem_manager.cc
- find_planet.cc

8.14 jeod::EphemeridesMessages Class Reference

Specifies the message IDs used in the Ephemerides model.

`#include <ephem_messages.hh>`

Static Public Attributes

- static char const * inconsistent_setup = "environment/ephemerides/" "inconsistent_setup"

Error issued when the ephemeris model configuration is inconsistent.

- static char const * file_error = "environment/ephemerides/" "file_error"

  Error issued when the ephemeris file cannot be opened for input.
- static char const * unsupported_architecture = "environment/ephemerides/" "unsupported_architecture"

  Error issued for machine architectures that do not conform to the architecture assumptions:
- static char const * garbage_file = "environment/ephemerides/" "garbage_file"

  Error issued when the ephemeris file appears to be garbage.
- static char const * time_not_in_range = "environment/ephemerides/" "time_not_in_range"

  Error issued when the ephemeris file does not contain data for the requested time.
- static char const * item_not_in_file = "environment/ephemerides/" "item_not_in_file"

  Error issued when the ephemeris file does not contain data for the requested item.
- static char const * null_pointer = "environment/ephemerides/" "null_pointer"

  Issued when a pointer should be non-NULL but isn't.
- static char const * duplicate_entry = "environment/ephemerides/" "duplicate_entry"

  Issued on request to add a pointer to a list a second time.
- static char const * invalid_name = "environment/ephemerides/" "invalid_name"

  Issued when a name is invalid – empty, a duplicate, ...
- static char const * invalid_item = "environment/ephemerides/" "invalid_item"

  Issued when something other than a name is invalid.
- static char const * single_ephem_mode = "environment/ephemerides/" "single_ephem_mode"

  Issued when the ephemeris manager is rejecting add_ephemeris calls.
- static char const * internal_error = "environment/ephemerides/" "internal_error"

  Issued when some internal error occurred.
- static char const * debug = "environment/ephemerides/" "debug"

  Used to send a message about a non-error condition.

Private Member Functions

- EphemeridesMessages (void)

  Not implemented.
- EphemeridesMessages (const EphemeridesMessages &)

  Not implemented.
- EphemeridesMessages & operator= (const EphemeridesMessages &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__EphemeridesMessages ()

8.14.1 Detailed Description

Specifies the message IDs used in the Ephemerides model.

Assumptions and Limitations

- This is a complete catalog of the messages sent by the ephemerides model.
- This is not an exhaustive list of all the things that can go awry.

Definition at line 51 of file ephem_messages.hh.

8.14.2 Constructor & Destructor Documentation

8.14.2.1 jeod::EphemeridesMessages::EphemeridesMessages (void) [private]

Not implemented.

8.14.2.2 jeod::EphemeridesMessages::EphemeridesMessages (const EphemeridesMessages &) [private]

Not implemented.

8.14.3 Member Function Documentation

8.14.3.1 EphemeridesMessages& jeod::EphemeridesMessages::operator= (const EphemeridesMessages &) [private]

Not implemented.

8.14.4 Friends And Related Function Documentation

8.14.4.1 void init_attrjeod__EphemeridesMessages ( ) [friend]

8.14.4.2 friend class InputProcessor [friend]

Definition at line 52 of file ephem_messages.hh.

8.14.5 Field Documentation

8.14.5.1 char const * jeod::EphemeridesMessages::debug = "environment/ephemerides/" "debug" [static]

Used to send a message about a non-error condition.

trick_units(–)

Definition at line 129 of file ephem_messages.hh.

Referenced by jeod::De4xxFile::open().

8.14.5.2 char const * jeod::EphemeridesMessages::duplicate_entry = "environment/ephemerides/" "duplicate_entry" [static]

Issued on request to add a pointer to a list a second time.

trick_units(–)

Definition at line 102 of file ephem_messages.hh.

Referenced by jeod::EphemeridesManager::add_ephem_item(), and jeod::EphemeridesManager::add_planet().

8.14.5.3 char const * jeod::EphemeridesMessages::file_error = "environment/ephemerides/" "file_error" [static]

Error issued when the ephemeris file cannot be opened for input.

trick_units(–)

Definition at line 66 of file ephem_messages.hh.

Referenced by jeod::De4xxFile::close(), jeod::De4xxFile::open(), jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

8.14.5.4 char const * jeod::EphemeridesMessages::garbage_file = "environment/ephemerides/" "garbage_file" [static]

Error issued when the ephemeris file appears to be garbage.

trick_units(–)

Definition at line 80 of file ephem_messages.hh.

Referenced by jeod::De4xxFile::pre_initialize().

8.14.5.5 char const * jeod::EphemeridesMessages::inconsistent_setup = "environment/ephemerides/" "inconsistent_setup" [static]

Error issued when the ephemeris model configuration is inconsistent.

trick_units(–)

Definition at line 61 of file ephem_messages.hh.

Referenced by jeod::De4xxEphemeris::activate_em_nodes(), jeod::EphemeridesManager::add_ephem_item(), jeod::De4xxEphemeris::ephem_build_tree(), jeod::EmptySpaceEphemeris::ephem_initialize(), jeod::SinglePlanetEphemeris::ephem_initialize(), jeod::PropagatedPlanet::ephem_initialize(), jeod::De4xxEphemeris::initialize_items(), jeod::PropagatedPlanet::initialize_model(), jeod::De4xxEphemeris::initialize_time(), jeod::EphemerisRefFrame::set_active_status(), jeod::PropagatedPlanet::set_mode(), jeod::SinglePointEphemeris::set_name(), and jeod::EphemeridesManager::set_target_frame().

8.14.5.6 char const * jeod::EphemeridesMessages::internal_error = "environment/ephemerides/" "internal_error" [static]

Issued when some internal error occurred.

These errors should never happen.trick_units(–)

Definition at line 123 of file ephem_messages.hh.

Referenced by jeod::SinglePointEphemeris::activate(), jeod::De4xxEphemeris::activate(), jeod::PropagatedPlanet::activate(), jeod::EphemeridesManager::add_ephem_item(), jeod::EphemerisOrientation::note_frame_status_change(), jeod::EphemerisPoint::note_frame_status_change(), jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

8.14.5.7 char const * jeod::EphemeridesMessages::invalid_item = "environment/ephemerides/" "invalid_item" [static]

Issued when something other than a name is invalid.

trick_units(–)

Definition at line 112 of file ephem_messages.hh.

Referenced by jeod::EphemerisItem::activate(), jeod::EphemeridesManager::find_ephem_angle(), jeod::EphemeridesManager::find_ephem_point(), jeod::EphemeridesManager::find_integ_frame_index(), jeod::EphemeridesManager::find_planet(), and jeod::EphemerisItem::set_target_frame().

8.14.5.8 char const * jeod::EphemeridesMessages::invalid_name = "environment/ephemerides/" "invalid_name" [static]

Issued when a name is invalid – empty, a duplicate, ...

trick_units(–)

Definition at line 107 of file ephem_messages.hh.

Referenced by jeod::EphemerisItem::set_name(), and jeod::EphemerisItem::validate_name().

8.14.5.9 char const * jeod::EphemeridesMessages::item_not_in_file = "environment/ephemerides/" "item_not_in_file" [static]

Error issued when the ephemeris file does not contain data for the requested item.

trick_units(–)

Definition at line 92 of file ephem_messages.hh.

Referenced by jeod::De4xxFile::update().

8.14.5.10 char const * jeod::EphemeridesMessages::null_pointer = "environment/ephemerides/" "null_pointer" [static]

Issued when a pointer should be non-NULL but isn't.

trick_units(–)

Definition at line 97 of file ephem_messages.hh.

8.14.5.11 char const * jeod::EphemeridesMessages::single_ephem_mode = "environment/ephemerides/" "single_ephem_mode" [static]

Issued when the ephemeris manager is rejecting add_ephemeris calls.

trick_units(–)

Definition at line 117 of file ephem_messages.hh.

Referenced by jeod::EphemeridesManager::add_ephem_item(), and jeod::EphemeridesManager::clear_added_ephemerides().

8.14.5.12 char const * jeod::EphemeridesMessages::time_not_in_range = "environment/ephemerides/" "time_not_in_range" [static]

Error issued when the ephemeris file does not contain data for the requested time.

trick_units(–)

Definition at line 86 of file ephem_messages.hh.

Referenced by jeod::De4xxFile::initialize(), and jeod::De4xxFile::update().

8.14.5.13 char const * jeod::EphemeridesMessages::unsupported_architecture = "environment/ephemerides/" "unsupported_architecture" [static]

Error issued for machine architectures that do not conform to the architecture assumptions:

- char = 8 bits
- int32_t = 4 bytes (32 bits)
- double = 8 bytes (64 bits)trick_units(–)

Definition at line 75 of file ephem_messages.hh.

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

- ephem_messages.hh
- ephem_messages.cc

8.15 jeod::EphemerisDataItemMeta Struct Reference

Structure containing the header metadata for sizing/locating the data entries with the data segments.

`#include <de4xx_file.hh>`

Data Fields

- uint32_t offset

  Offsets into coeffs array.
- uint32_t nterms

  Chebychev polynomial terms.
- uint32_t npoly

  Number polynomials per data block.

8.15.1 Detailed Description

Structure containing the header metadata for sizing/locating the data entries with the data segments.

Definition at line 121 of file de4xx_file.hh.

8.15.2 Field Documentation

8.15.2.1 uint32_t jeod::EphemerisDataItemMeta::npoly

Number polynomials per data block.

trick_units(–)

Definition at line 136 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::interpolate().

8.15.2.2 uint32_t jeod::EphemerisDataItemMeta::nterms

Chebychev polynomial terms.

trick_units(–)

Definition at line 131 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::interpolate(), and jeod::De4xxFile::pre_initialize().

8.15.2.3 uint32_t jeod::EphemerisDataItemMeta::offset

Offsets into coeffs array.

trick_units(–)

Definition at line 126 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::interpolate(), and jeod::De4xxFile::pre_initialize().

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

- de4xx_file.hh

8.16 jeod::EphemerisDataSegmentMeta Struct Reference

Metadata implied from each data segment.

`#include <de4xx_file.hh>`

Data Fields

- uint32_t num_recs

  The number of records in the file.
- double start_epoch

  Julian date of start of file.
- double stop_epoch

  Julian date of end of file.

8.16.1 Detailed Description

Metadata implied from each data segment.

Definition at line 143 of file de4xx_file.hh.

8.16.2 Field Documentation

8.16.2.1 uint32_t jeod::EphemerisDataSegmentMeta::num_recs

The number of records in the file.

trick_units(–)

Definition at line 148 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

8.16.2.2 double jeod::EphemerisDataSegmentMeta::start_epoch

Julian date of start of file.

trick_units(day)

Definition at line 153 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize(), and jeod::De4xxFile::pre_initialize().

8.16.2.3 double jeod::EphemerisDataSegmentMeta::stop_epoch

Julian date of end of file.

trick_units(day)

Definition at line 158 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::pre_initialize().

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

- de4xx_file.hh

8.17 jeod::EphemerisDataSetMeta Struct Reference

Container for the metadata from the DE model header.

`#include <de4xx_file.hh>`

Data Fields

- uint32_t number_file_items

  Return the number of ephemeris items provided in DE data.
- double start_epoch

  Julian date of start of dataset.
- double stop_epoch

  Julian date of end of dataset.
- double delta_epoch

  Julian period length of each segment.
- uint32_t number_segments

  Number of ascp files (segments) provided by DE model.
- uint32_t ncoeff

  Size of each data record throughout the dataset.
- double de_constants [De4xxBase::De4xx_Const_MaxConsts]

  Array of supplied constants required by JEOD.

8.17.1 Detailed Description

Container for the metadata from the DE model header.

Definition at line 77 of file de4xx_file.hh.

8.17.2 Field Documentation

8.17.2.1 double jeod::EphemerisDataSetMeta::de_constants[De4xxBase::De4xx_Const_MaxConsts]

Array of supplied constants required by JEOD.

Definition at line 113 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize(), jeod::De4xxEphemeris::initialize_file(), and jeod::De4xxFile::pre_initialize().

8.17.2.2 double jeod::EphemerisDataSetMeta::delta_epoch

Julian period length of each segment.

trick_units(day)

Definition at line 98 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::initialize(), jeod::De4xxFile::interpolate(), jeod::De4xxFile::time_is_in_range(), and jeod::De4xxFile::update().

8.17.2.3 uint32_t jeod::EphemerisDataSetMeta::ncoeff

Size of each data record throughout the dataset.

trick_units(–)

Definition at line 108 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::update().

8.17.2.4 uint32_t jeod::EphemerisDataSetMeta::number_file_items

Return the number of ephemeris items provided in DE data.

(e.g., 13 for DE405/421, 15 for DE440)

Definition at line 83 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::interpolate(), jeod::De4xxFile::open(), jeod::De4xxFile::pre_initialize(), and jeod::-De4xxFile::update().

8.17.2.5 uint32_t jeod::EphemerisDataSetMeta::number_segments

Number of ascp files (segments) provided by DE model.

trick_units(–)

Definition at line 103 of file de4xx_file.hh.

Referenced by jeod::De4xxFile::pre_initialize(), and jeod::De4xxFile::update().

8.17.2.6 double jeod::EphemerisDataSetMeta::start_epoch

Julian date of start of dataset.

trick_units(day)

Definition at line 88 of file de4xx_file.hh.

8.17.2.7 double jeod::EphemerisDataSetMeta::stop_epoch

Julian date of end of dataset.

trick_units(day)

Definition at line 93 of file de4xx_file.hh.

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

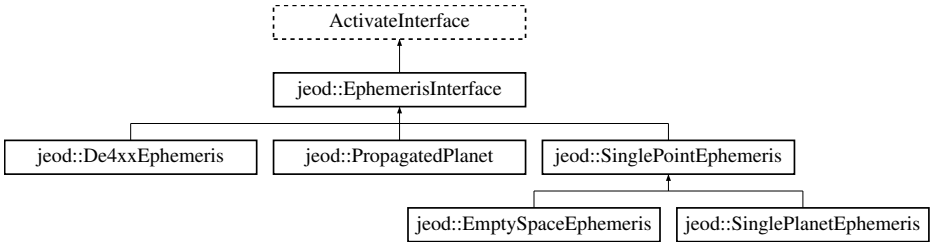
- de4xx_file.hh

8.18 jeod::EphemerisInterface Class Reference

Interface class that specifies minimal functionality of an ephemeris model.

`#include <ephem_interface.hh>`

Inheritance diagram for jeod::EphemerisInterface:



Public Member Functions

- virtual ~EphemerisInterface (void)

  Destructor; there is nothing to destroy here.
- virtual double timestamp (void) const =0

  Indicates when class was last updated.
- virtual const char * get_name (void) const =0

  Identify the model.
- virtual void ephem_initialize (EphemeridesManager &manager)=0

  Initialize the model.
- virtual void ephem_activate (EphemeridesManager &manager)=0

  Activate the model.
- virtual void ephem_build_tree (EphemeridesManager &manager)=0

  Build the model's contribution to the reference frame tree.
- virtual void ephem_update (void)=0

  Update the model.

Friends

- class InputProcessor
- void init_attrjeod__EphemerisInterface ()

8.18.1 Detailed Description

Interface class that specifies minimal functionality of an ephemeris model.

Definition at line 47 of file ephem_interface.hh.

8.18.2 Constructor & Destructor Documentation

8.18.2.1 jeod::EphemerisInterface::~EphemerisInterface (void) [inline], [virtual]

Destructor; there is nothing to destroy here.

Definition at line 129 of file ephem_interface.hh.

8.18.3 Member Function Documentation

8.18.3.1 virtual void jeod::EphemerisInterface::ephem_activate (EphemeridesManager & *manager*) [pure virtual]

Activate the model.

Parameters

in,out	*manager*	Ephemerides manager

Implemented in jeod::PropagatedPlanet, jeod::SinglePlanetEphemeris, jeod::De4xxEphemeris, jeod::EmptySpaceEphemeris, and jeod::SinglePointEphemeris.

Referenced by jeod::EphemeridesManager::activate_ephemerides().

8.18.3.2 virtual void jeod::EphemerisInterface::ephem_build_tree (EphemeridesManager & *manager*) [pure virtual]

Build the model's contribution to the reference frame tree.

Parameters

in,out	*manager*	Ephemerides manager

Implemented in jeod::PropagatedPlanet, jeod::SinglePlanetEphemeris, jeod::De4xxEphemeris, jeod::EmptySpaceEphemeris, and jeod::SinglePointEphemeris.

Referenced by jeod::EphemeridesManager::activate_ephemerides().

8.18.3.3 virtual void jeod::EphemerisInterface::ephem_initialize (EphemeridesManager & *manager*) [pure virtual]

Initialize the model.

Parameters

in,out	*manager*	Ephemerides manager

Implemented in jeod::PropagatedPlanet, jeod::SinglePlanetEphemeris, jeod::De4xxEphemeris, jeod::EmptySpaceEphemeris, and jeod::SinglePointEphemeris.

Referenced by jeod::EphemeridesManager::initialize_ephemerides().

8.18.3.4 virtual void jeod::EphemerisInterface::ephem_update (void) [pure virtual]

Update the model.

Implemented in jeod::PropagatedPlanet, jeod::De4xxEphemeris, and jeod::SinglePointEphemeris.

Referenced by jeod::EphemeridesManager::update_ephemerides().

8.18.3.5 virtual const char* jeod::EphemerisInterface::get_name (void) const [pure virtual]

Identify the model.

Returns

Model name

Implemented in jeod::PropagatedPlanet, jeod::De4xxEphemeris, and jeod::SinglePointEphemeris.

Referenced by jeod::EphemeridesManager::add_ephem_item().

8.18.3.6 virtual double jeod::EphemerisInterface::timestamp (void) const [pure virtual]

Indicates when class was last updated.

Returns

Time of last update
Units: s

Implemented in jeod::PropagatedPlanet, jeod::De4xxEphemeris, and jeod::SinglePointEphemeris.

8.18.4 Friends And Related Function Documentation

8.18.4.1 void init_attrjeod__EphemerisInterface ( ) [friend]

8.18.4.2 friend class InputProcessor [friend]

Definition at line 48 of file ephem_interface.hh.

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

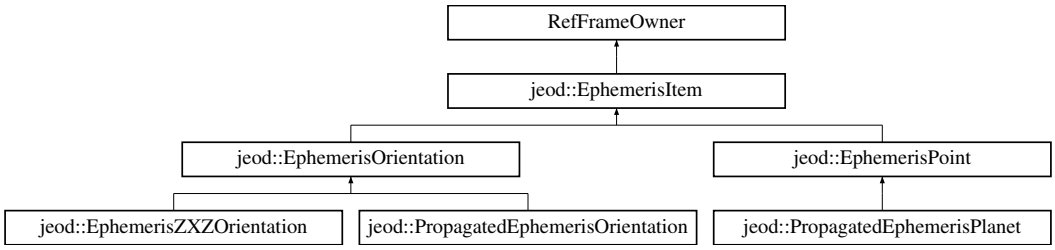
- ephem_interface.hh

8.19 jeod::EphemerisItem Class Reference

The EphemerisItem class is the base class for representing an item that is modeled in an ephemeris model.

`#include <ephem_item.hh>`

Inheritance diagram for jeod::EphemerisItem:



Public Types

- enum TargetAspect { Translation = 0, Rotation = 1 }

  Defines the aspect of the target frame that will be modified by the EphemerisItem object.

Public Member Functions

- EphemerisItem ()

  Construct an ephemeris item.
- virtual ~EphemerisItem ()

  Destroy an ephemeris item.
- virtual void set_name (const char *new_name)

  Name an ephemeris item.
- virtual void set_name (const char *pname, const char *fname)

  Name an ephemeris item.
- const char * get_name () const

  Return the name.
- virtual void set_timestamp (double time)

Set the update time of this item.

- double timestamp (void) const

  Return the update time of this item.

- virtual void set_owner (EphemerisInterface *new_owner)

  Set the owner of this item.

- EphemerisInterface * get_owner () const

  Return the owner of this item.

- virtual void set_manager (BaseEphemeridesManager *new_manager)

  Set the manager of this item.

- BaseEphemeridesManager * get_manager () const

  Return the manager of this item.

- virtual void set_next (EphemerisItem *next_item)

  Set the next item.

- EphemerisItem * get_next () const

  Get the next item.

- virtual void set_head (EphemerisItem *head_item)

  Set the head item.

- EphemerisItem * get_head () const

  Get the head item.

- virtual void set_target_frame (EphemerisRefFrame &frame)

  Set the target frame.

- EphemerisRefFrame * get_target_frame () const

  Get the target frame.

- virtual void enable ()

  Enable an EphemerisItem object.

- virtual void disable ()

  Disable an EphemerisItem object.

- bool is_enabled () const

  Return enabled status.

- EphemerisItem * get_enabled_item (void) const

  Get the item marked as enabled, if any.

- virtual void activate ()

  Activate a EphemerisItem object.

- virtual void deactivate ()

  Deactivate a EphemerisItem object.

- bool is_active () const

  Return activity status.

- bool is_activatable () const

  Is the item activatable?

- void validate_name (const char *file, unsigned int line, const char *new_value, const char *old_value, const char *variable_name)

  Name an ephemeris item.

- virtual TargetAspect updates_what (void) const =0

  Identifies which part of the target frame does the object updates.

- virtual const char * default_suffix (void) const =0

  The default suffix for the item.

- virtual void disconnect_from_tree (void)=0

  Disconnect the item from the reference frame tree.

Protected Member Functions

- virtual void set_name_internal (char *new_name)

  Name an ephemeris item.

Protected Attributes

- char * name

  The name of the item.
- EphemerisInterface * owner

  The ephemeris model that owns this object.
- BaseEphemeridesManager * manager

  The ephemeris manager that manages this object.
- EphemerisRefFrame * target_frame

  The reference frame whose non-constant state is set by this object.
- EphemerisItem * head

  The first ephemeris item with the same name as this item.
- EphemerisItem * next

  The next ephemeris item with the same name as this item.
- double update_time

  Time of last update, dynamic time seconds.
- bool enabled

  Is the item enabled?
- bool active

  Is the item active?

Private Member Functions

- EphemerisItem (const EphemerisItem &)

  Not implemented.
- EphemerisItem & operator= (const EphemerisItem &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__EphemerisItem ()

8.19.1 Detailed Description

The EphemerisItem class is the base class for representing an item that is modeled in an ephemeris model.

Ephemeris items form the bridge between the reference frame model and the ephemeris models. An EphemerisItem

- Has a name, which is not necessarily unique. Ephemeris items with the same name are linked to one another to form a linked list.
- Has a target reference frame. This is the ephemeris reference frame which has the same name as the ephemeris item.

- Can be enabled or disabled. At most one item of a set of commonly-named items can be enabled, and only if a correspondingly-named ephemeris reference frame exists. Ownership of the correspondingly-named ephemeris reference frame transfers to the enabled ephemeris item. When an ephemeris item is disabled, the ephemeris model that owns the ephemeris item should not operate on the correspondingly-named reference frame.
- Can be active or inactive. Disabled items must always be inactive. The activity level of the enabled item for a set of commonly-named items is automatically maintained to be the same as that of the target frame.

The ephemeris model that owns an active ephemeris item is responsible for ensuring that the correspondingly-named ephemeris reference frame is a part of the active reference frame tree and for updating that reference frame's state.

Definition at line 75 of file ephem_item.hh.

8.19.2 Member Enumeration Documentation

8.19.2.1 enum jeod::EphemerisItem::TargetAspect

Defines the aspect of the target frame that will be modified by the EphemerisItem object.

Enumerator

Translation

Rotation

Definition at line 86 of file ephem_item.hh.

8.19.3 Constructor & Destructor Documentation

8.19.3.1 jeod::EphemerisItem::EphemerisItem (void)

Construct an ephemeris item.

Definition at line 70 of file ephem_item.cc.

8.19.3.2 jeod::EphemerisItem::~EphemerisItem (void) [virtual]

Destroy an ephemeris item.

Definition at line 91 of file ephem_item.cc.

References name.

8.19.3.3 jeod::EphemerisItem::EphemerisItem (const EphemerisItem &) [private]

Not implemented.

8.19.4 Member Function Documentation

8.19.4.1 void jeod::EphemerisItem::activate (void) [virtual]

Activate a EphemerisItem object.

Definition at line 344 of file ephem_item.cc.

References active, jeod::EphemeridesMessages::invalid_item, is_activatable(), and name.

Referenced by jeod::De4xxEphemeris::activate_em_nodes(), jeod::EphemerisOrientation::note_frame_status_change(), jeod::EphemerisPoint::note_frame_status_change(), and set_target_frame().

8.19.4.2 void jeod::EphemerisItem::deactivate (void) [inline], [virtual]

Deactivate a EphemerisItem object.

Definition at line 97 of file ephem_item_inline.hh.

References active.

Referenced by jeod::EphemerisOrientation::note_frame_status_change(), and jeod::EphemerisPoint::note_frame_status_change().

8.19.4.3 virtual const char* jeod::EphemerisItem::default_suffix (void) const [pure virtual]

The default suffix for the item.

Implemented in jeod::EphemerisOrientation, and jeod::EphemerisPoint.

Referenced by set_name().

8.19.4.4 void jeod::EphemerisItem::disable (void) [inline], [virtual]

Disable an EphemerisItem object.

Definition at line 315 of file ephem_item.cc.

References active, enabled, jeod::BaseEphemeridesManager::ephem_note_tree_status_change(), manager, and target_frame.

Referenced by jeod::EphemeridesManager::add_ephem_item(), jeod::EphemeridesManager::clear_added_ephemerides(), enable(), jeod::EmptySpaceEphemeris::ephem_initialize(), jeod::SinglePlanetEphemeris::ephem_initialize(), jeod::De4xxEphemeris::initialize_items(), and jeod::PropagatedPlanet::set_mode().

8.19.4.5 virtual void jeod::EphemerisItem::disconnect_from_tree (void) [pure virtual]

Disconnect the item from the reference frame tree.

Implemented in jeod::EphemerisOrientation, and jeod::EphemerisPoint.

Referenced by jeod::EphemeridesManager::activate_ephemerides().

8.19.4.6 void jeod::EphemerisItem::enable (void) [virtual]

Enable an EphemerisItem object.

Reimplemented in jeod::EphemerisOrientation.

Definition at line 278 of file ephem_item.cc.

References active, disable(), enabled, jeod::BaseEphemeridesManager::ephem_note_tree_status_change(), get_enabled_item(), manager, and target_frame.

Referenced by jeod::De4xxEphemeris::activate_em_nodes(), jeod::De4xxEphemeris::De4xxEphemeris(), jeod::EmptySpaceEphemeris::EmptySpaceEphemeris(), jeod::EphemerisOrientation::enable(), jeod::PropagatedPlanet::set_mode(), and jeod::SinglePlanetEphemeris::SinglePlanetEphemeris().

8.19.4.7 EphemerisItem* jeod::EphemerisItem::get_enabled_item (void) const [inline]

Get the item marked as enabled, if any.

Returns

Enabled item

Definition at line 239 of file ephem_item_inline.hh.

References enabled, head, and next.

Referenced by jeod::De4xxEphemeris::activate_nodes(), jeod::EphemeridesManager::add_ephem_item(), jeod::EphemerisOrientation::enable(), enable(), and set_target_frame().

8.19.4.8 EphemerisItem * jeod::EphemerisItem::get_head (void) const [inline]

Get the head item.

Returns

Root item

Definition at line 187 of file ephem_item_inline.hh.

References head.

8.19.4.9 BaseEphemeridesManager * jeod::EphemerisItem::get_manager (void) const [inline]

Return the manager of this item.

Returns

Object manager

Definition at line 161 of file ephem_item_inline.hh.

References manager.

8.19.4.10 const char * jeod::EphemerisItem::get_name (void) const [inline]

Return the name.

Returns

Void

Definition at line 46 of file ephem_item_inline.hh.

References name.

Referenced by jeod::EphemeridesManager::add_ephem_item(), jeod::De4xxEphemeris::De4xxEphemeris(), and jeod::EphemeridesManager::find_ephem_item().

8.19.4.11 EphemerisItem * jeod::EphemerisItem::get_next (void) const [inline]

Get the next item.

Returns

Next item

Definition at line 213 of file ephem_item_inline.hh.

References next.

Referenced by jeod::EphemeridesManager::add_ephem_item().

8.19.4.12 EphemerisInterface * jeod::EphemerisItem::get_owner (void) const [inline]

Return the owner of this item.

Returns

Frame owner

Definition at line 135 of file ephem_item_inline.hh.

References owner.

Referenced by jeod::EphemeridesManager::add_ephem_item().

8.19.4.13 EphemerisRefFrame * jeod::EphemerisItem::get_target_frame (void) const [inline]

Get the target frame.

Returns

Target frame

Definition at line 226 of file ephem_item_inline.hh.

References target_frame.

Referenced by jeod::SinglePlanetEphemeris::ephem_build_tree(), jeod::EmptySpaceEphemeris::ephem_initialize(), jeod::De4xxEphemeris::ephem_initialize(), jeod::SinglePlanetEphemeris::ephem_initialize(), and jeod::PropagatedPlanet::ephem_initialize().

8.19.4.14 bool jeod::EphemerisItem::is_activatable (void) const

Is the item activatable?

Returns

True if item can be activated.

Definition at line 364 of file ephem_item.cc.

References active, enabled, head, and next.

Referenced by activate().

8.19.4.15 bool jeod::EphemerisItem::is_active (void) const [inline]

Return activity status.

Returns

Is item active?

Definition at line 110 of file ephem_item_inline.hh.

References active.

Referenced by jeod::De4xxEphemeris::activate_nodes().

8.19.4.16 bool jeod::EphemerisItem::is_enabled (void) const [inline]

Return enabled status.

Returns

Is item enabled?

Definition at line 85 of file ephem_item_inline.hh.

References enabled.

Referenced by jeod::EphemeridesManager::add_ephem_item().

8.19.4.17 EphemerisItem& jeod::EphemerisItem::operator= (const EphemerisItem &) [private]

Not implemented.

8.19.4.18 void jeod::EphemerisItem::set_head (EphemerisItem * *head_item*) [inline], [virtual]

Set the head item.

Parameters

in,out	*head_item*	Root item

Definition at line 174 of file ephem_item_inline.hh.

References head.

Referenced by jeod::EphemeridesManager::add_ephem_item().

8.19.4.19 void jeod::EphemerisItem::set_manager (BaseEphemeridesManager * *new_manager*) [inline], [virtual]

Set the manager of this item.

Parameters

in	*new_manager*	New owner

Definition at line 148 of file ephem_item_inline.hh.

References manager.

Referenced by jeod::EphemeridesManager::add_ephem_item().

8.19.4.20 void jeod::EphemerisItem::set_name (const char * *new_name*) [virtual]

Name an ephemeris item.

Parameters

in	*new_name*	New name

Definition at line 159 of file ephem_item.cc.

References default_suffix(), jeod::EphemeridesMessages::invalid_name, name, set_name_internal(), and validate_name().

Referenced by jeod::De4xxEphemeris::De4xxEphemeris(), jeod::PropagatedPlanet::initialize_model(), jeod::EmptySpaceEphemeris::set_name(), and jeod::SinglePlanetEphemeris::set_name().

8.19.4.21 void jeod::EphemerisItem::set_name (const char * *pname,* const char * *fname*) [virtual]

Name an ephemeris item.

Parameters

in	*pname*	Planet name
in	*fname*	Frame name

Definition at line 140 of file ephem_item.cc.

References name, set_name_internal(), and validate_name().

8.19.4.22 void jeod::EphemerisItem::set_name_internal (char * *new_name*) [protected], [virtual]

Name an ephemeris item.

Parameters

in	*new_name*	New name

Definition at line 191 of file ephem_item.cc.

References name.

Referenced by set_name().

8.19.4.23 void jeod::EphemerisItem::set_next (**EphemerisItem** * *next_item*) [inline], [virtual]

Set the next item.

Parameters

in,out	*next_item*	Next item

Definition at line 200 of file ephem_item_inline.hh.

References next.

Referenced by jeod::EphemeridesManager::add_ephem_item().

8.19.4.24 void jeod::EphemerisItem::set_owner (**EphemerisInterface** * *new_owner*) [inline], [virtual]

Set the owner of this item.

Parameters

in	*new_owner*	New owner

Definition at line 123 of file ephem_item_inline.hh.

References owner.

Referenced by jeod::De4xxEphemeris::De4xxEphemeris(), jeod::EmptySpaceEphemeris::EmptySpaceEphemeris(), jeod::PropagatedPlanet::PropagatedPlanet(), and jeod::SinglePlanetEphemeris::SinglePlanetEphemeris().

8.19.4.25 void jeod::EphemerisItem::set_target_frame (**EphemerisRefFrame** & *frame*) [virtual]

Set the target frame.

All ephemeris items that share a common name must point to the same target frame.

Parameters

in	*frame*	Target frame

Definition at line 212 of file ephem_item.cc.

References activate(), get_enabled_item(), head, jeod::EphemeridesMessages::invalid_item, jeod::BaseEphemeridesManager::is_integ_frame(), manager, name, next, set_target_frame(), target_frame, Translation,

and updates_what().

Referenced by jeod::EphemeridesManager::add_ephem_item(), set_target_frame(), and jeod::EphemeridesManager::set_target_frame().

8.19.4.26 void jeod::EphemerisItem::set_timestamp (double *time*) [inline], [virtual]

Set the update time of this item.

Parameters

in	*time*	Time Units: s

Definition at line 59 of file ephem_item_inline.hh.

References update_time.

8.19.4.27 double jeod::EphemerisItem::timestamp (void) const [inline]

Return the update time of this item.

Returns

Time of last update
Units: s

Definition at line 72 of file ephem_item_inline.hh.

References update_time.

8.19.4.28 virtual TargetAspect jeod::EphemerisItem::updates_what (void) const [pure virtual]

Identifies which part of the target frame does the object updates.

Implemented in jeod::EphemerisOrientation, and jeod::EphemerisPoint.

Referenced by jeod::EphemeridesManager::add_ephem_item(), and set_target_frame().

8.19.4.29 void jeod::EphemerisItem::validate_name (const char * *file*, unsigned int *line*, const char * *new_value*, const char * *old_value*, const char * *variable_name*)

Name an ephemeris item.

Parameters

in	*file*	Usually **FILE**
in	*line*	Usually **LINE**
in	*new_value*	Value to check
in	*old_value*	Current value
in	*variable_name*	Variable name

Definition at line 110 of file ephem_item.cc.

References jeod::EphemeridesMessages::invalid_name, and manager.

Referenced by set_name().

8.19.5 Friends And Related Function Documentation

8.19.5.1 void init_attrjeod__EphemerisItem ( ) [friend]

8.19.5.2 friend class InputProcessor [friend]

Definition at line 76 of file ephem_item.hh.

8.19.6 Field Documentation

8.19.6.1 bool jeod::EphemerisItem::active [protected]

Is the item active?

- An item can be activated only if it is enabled. The enable and activate methods assure that this is the case.
- Activity is determined by the activity of the target frame, which is in turn determined by the reference frame subscription model.trick_units(–)

Definition at line 241 of file ephem_item.hh.

Referenced by activate(), deactivate(), disable(), jeod::EphemerisPoint::disconnect_from_tree(), enable(), is_activatable(), and is_active().

8.19.6.2 bool jeod::EphemerisItem::enabled [protected]

Is the item enabled?

- An item can be enabled only if the data associated with the item such as the translational state of a planet exist somewhere in the simulation.
- Only one of a set of ephemeris items that share the same name can be enabled. The enable method ensures that this is the case.
- Exactly one of a set of ephemeris items that share same name should be enabled if some other simulation agent depends on the data associated with an ephemeris item. Ensuring that this is the case is the responsibility of the ephemeris models and the users of those models.trick_units(–)

Definition at line 232 of file ephem_item.hh.

Referenced by disable(), jeod::EphemerisOrientation::enable(), enable(), get_enabled_item(), is_activatable(), is_enabled(), jeod::PropagatedEphemerisPlanet::update(), and jeod::PropagatedEphemerisOrientation::update().

8.19.6.3 EphemerisItem* jeod::EphemerisItem::head [protected]

The first ephemeris item with the same name as this item.

trick_units(–)

Definition at line 209 of file ephem_item.hh.

Referenced by get_enabled_item(), get_head(), is_activatable(), set_head(), and set_target_frame().

8.19.6.4 BaseEphemeridesManager* jeod::EphemerisItem::manager [protected]

The ephemeris manager that manages this object.

trick_units(–)

Definition at line 199 of file ephem_item.hh.

Referenced by disable(), enable(), get_manager(), set_manager(), set_target_frame(), and validate_name().

8.19.6.5 char* jeod::EphemerisItem::name [protected]

The name of the item.

trick_units(–)

Definition at line 189 of file ephem_item.hh.

Referenced by activate(), get_name(), set_name(), set_name_internal(), set_target_frame(), and ~EphemerisItem().

8.19.6.6 EphemerisItem* jeod::EphemerisItem::next [protected]

The next ephemeris item with the same name as this item.

trick_units(–)

Definition at line 214 of file ephem_item.hh.

Referenced by get_enabled_item(), get_next(), is_activatable(), set_next(), and set_target_frame().

8.19.6.7 EphemerisInterface* jeod::EphemerisItem::owner [protected]

The ephemeris model that owns this object.

trick_units(–)

Definition at line 194 of file ephem_item.hh.

Referenced by get_owner(), and set_owner().

8.19.6.8 EphemerisRefFrame* jeod::EphemerisItem::target_frame [protected]

The reference frame whose non-constant state is set by this object.

trick_units(–)

Definition at line 204 of file ephem_item.hh.

Referenced by disable(), jeod::EphemerisPoint::disconnect_from_tree(), enable(), get_target_frame(), jeod::EphemerisPoint::initialize_state(), jeod::EphemerisOrientation::note_frame_status_change(), jeod::EphemerisPoint::note_frame_status_change(), jeod::EphemerisZXZOrientation::propagate(), set_target_frame(), jeod::EphemerisPoint::update(), jeod::EphemerisZXZOrientation::update(), jeod::PropagatedEphemerisPlanet::update(), jeod::PropagatedEphemerisOrientation::update(), and jeod::EphemerisPoint::update_scaled().

8.19.6.9 double jeod::EphemerisItem::update_time [protected]

Time of last update, dynamic time seconds.

trick_units(s)

Definition at line 219 of file ephem_item.hh.

Referenced by jeod::EphemerisZXZOrientation::propagate(), set_timestamp(), timestamp(), jeod::EphemerisPoint::update(), jeod::EphemerisZXZOrientation::update(), jeod::PropagatedEphemerisPlanet::update(), jeod::PropagatedEphemerisOrientation::update(), and jeod::EphemerisPoint::update_scaled().

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

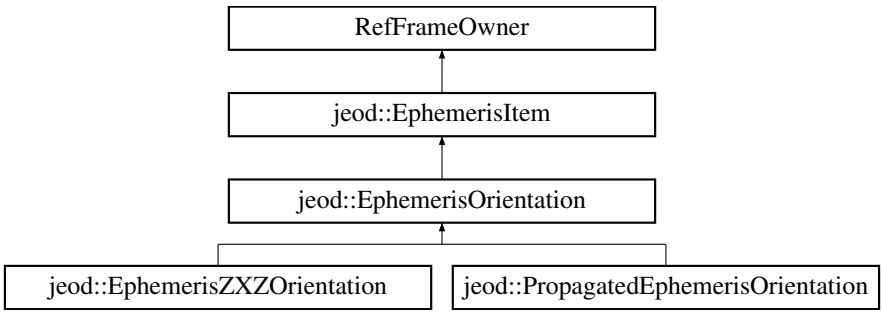
- ephem_item.hh
- ephem_item_inline.hh
- ephem_item.cc

8.20 jeod::EphemerisOrientation Class Reference

An EphemerisOrientation object updates the rotational state of an ephemeris reference frame.

`#include <ephem_orient.hh>`

Inheritance diagram for jeod::EphemerisOrientation:



Public Member Functions

- EphemerisOrientation ()

  Construct an ephemeris angle.
- virtual ~EphemerisOrientation ()

  Destroy an ephemeris angle.
- virtual TargetAspect updates_what (void) const

  Specify the aspect of the target frame updated by the object.
- virtual void enable ()

  Enable a EphemerisItem object.
- virtual void note_frame_status_change (RefFrame *frame)

  Null implementation.
- virtual const char * default_suffix () const

  Return the default suffix for this item class, i.e., "pfix".
- virtual void disconnect_from_tree ()

  Disconnect the item from the tree; this is a no-op for an EphemerisOrientation.

Protected Attributes

- bool subscribed_to_inertial

  A subscription to the planet's inertial frame is issued whenever the planet's planet-fixed frame is active to ensure that the the planet-fixed frame is a part of the ref frame tree.

Private Member Functions

- EphemerisOrientation (const EphemerisOrientation &)

  Not implemented.
- EphemerisOrientation & operator= (const EphemerisOrientation &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__EphemerisOrientation ()

Additional Inherited Members

8.20.1 Detailed Description

An EphemerisOrientation object updates the rotational state of an ephemeris reference frame.

Definition at line 55 of file ephem_orient.hh.

8.20.2 Constructor & Destructor Documentation

8.20.2.1 jeod::EphemerisOrientation::EphemerisOrientation (void)

Construct an ephemeris angle.

Definition at line 61 of file ephem_orient.cc.

8.20.2.2 jeod::EphemerisOrientation::~EphemerisOrientation (void) [virtual]

Destroy an ephemeris angle.

Definition at line 73 of file ephem_orient.cc.

8.20.2.3 jeod::EphemerisOrientation::EphemerisOrientation (const EphemerisOrientation &) [private]

Not implemented.

8.20.3 Member Function Documentation

8.20.3.1 const char * jeod::EphemerisOrientation::default_suffix (void) const [virtual]

Return the default suffix for this item class, i.e., "pfix".

Returns

Default suffix

Implements jeod::EphemerisItem.

Definition at line 168 of file ephem_orient.cc.

8.20.3.2 void jeod::EphemerisOrientation::disconnect_from_tree (void) [virtual]

Disconnect the item from the tree; this is a no-op for an EphemerisOrientation.

Implements jeod::EphemerisItem.

Definition at line 181 of file ephem_orient.cc.

8.20.3.3 void jeod::EphemerisOrientation::enable (void) [virtual]

Enable a EphemerisItem object.

Reimplemented from jeod::EphemerisItem.

Definition at line 98 of file ephem_orient.cc.

References jeod::EphemerisItem::enable(), jeod::EphemerisItem::enabled, jeod::EphemerisItem::get_enabled_item(), and subscribed_to_inertial.

Referenced by jeod::De4xxEphemeris::De4xxEphemeris(), and jeod::PropagatedPlanet::set_mode().

8.20.3.4 void jeod::EphemerisOrientation::note_frame_status_change (RefFrame * *frame*) [virtual]

Null implementation.

Parameters

in	*frame*	Frame whose status has changed

Definition at line 124 of file ephem_orient.cc.

References jeod::EphemerisItem::activate(), jeod::EphemerisItem::deactivate(), jeod::EphemeridesMessages::internal_error, subscribed_to_inertial, and jeod::EphemerisItem::target_frame.

8.20.3.5 EphemerisOrientation& jeod::EphemerisOrientation::operator= (const EphemerisOrientation &) [private]

Not implemented.

8.20.3.6 EphemerisItem::TargetAspect jeod::EphemerisOrientation::updates_what (void) const [virtual]

Specify the aspect of the target frame updated by the object.

EphemerisOrientation objects update the rotational state.

Returns

Target of object

Implements jeod::EphemerisItem.

Definition at line 86 of file ephem_orient.cc.

References jeod::EphemerisItem::Rotation.

8.20.4 Friends And Related Function Documentation

8.20.4.1 void init_attrjeod__EphemerisOrientation ( ) [friend]

8.20.4.2 friend class InputProcessor [friend]

Definition at line 56 of file ephem_orient.hh.

8.20.5 Field Documentation

8.20.5.1 bool jeod::EphemerisOrientation::subscribed_to_inertial [protected]

A subscription to the planet's inertial frame is issued whenever the planet's planet-fixed frame is active to ensure that the the planet-fixed frame is a part of the ref frame tree.

This flag is set when such a subscription is made.trick_units(–)

Definition at line 93 of file ephem_orient.hh.

Referenced by enable(), and note_frame_status_change().

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

- ephem_orient.hh

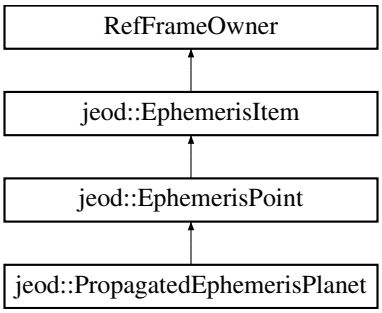
- ephem_orient.cc

8.21 jeod::EphemerisPoint Class Reference

An EphemerisPoint object updates the translational state of an ephemeris reference frame.

`#include <ephem_point.hh>`

Inheritance diagram for jeod::EphemerisPoint:



Public Member Functions

- EphemerisPoint ()
  Construct an ephemeris point.
- virtual ~EphemerisPoint ()
  Destroy an ephemeris point.
- virtual TargetAspect updates_what (void) const
  Specify the aspect of the target frame updated by the object.
- virtual const char * default_suffix () const
  Return the default suffix for this item class, i.e., "inertial".
- virtual void disconnect_from_tree ()
  Disconnect the associated inertial frame from the tree.
- virtual void note_frame_status_change (RefFrame *frame)
  Set active status to correspond with that of the inertial frame.
- virtual void initialize_state ()
  Zero-out the inertial frame's translational state.
- virtual void update (const double *pos, const double *vel, double time)
  Update the inertial frame's translational state.
- virtual void update_scaled (const double *pos, const double *vel, double scale, double time)
  Update the inertial frame's translational state.

Private Member Functions

- EphemerisPoint (const EphemerisPoint &)
  Not implemented.
- EphemerisPoint & operator= (const EphemerisPoint &)
  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__EphemerisPoint ()

Additional Inherited Members

8.21.1 Detailed Description

An EphemerisPoint object updates the translational state of an ephemeris reference frame.

Definition at line 52 of file ephem_point.hh.

8.21.2 Constructor & Destructor Documentation

8.21.2.1 jeod::EphemerisPoint::EphemerisPoint (void)

Construct an ephemeris point.

Definition at line 61 of file ephem_point.cc.

8.21.2.2 jeod::EphemerisPoint::~EphemerisPoint (void) [virtual]

Destroy an ephemeris point.

Definition at line 73 of file ephem_point.cc.

8.21.2.3 jeod::EphemerisPoint::EphemerisPoint (const EphemerisPoint &) [private]

Not implemented.

8.21.3 Member Function Documentation

8.21.3.1 const char * jeod::EphemerisPoint::default_suffix (void) const [virtual]

Return the default suffix for this item class, i.e., "inertial".

Returns

Default suffix

Implements jeod::EphemerisItem.

Definition at line 112 of file ephem_point.cc.

8.21.3.2 void jeod::EphemerisPoint::disconnect_from_tree (void) [virtual]

Disconnect the associated inertial frame from the tree.

Implements jeod::EphemerisItem.

Definition at line 124 of file ephem_point.cc.

References jeod::EphemerisItem::active, and jeod::EphemerisItem::target_frame.

8.21.3.3 void jeod::EphemerisPoint::initialize_state (void) [inline], [virtual]

Zero-out the inertial frame's translational state.

Definition at line 139 of file ephem_point.cc.

References jeod::EphemerisItem::target_frame.

8.21.3.4 void jeod::EphemerisPoint::note_frame_status_change (RefFrame * *frame*) [virtual]

Set active status to correspond with that of the inertial frame.

Parameters

in	*frame*	Frame whose status has changed

Definition at line 85 of file ephem_point.cc.

References jeod::EphemerisItem::activate(), jeod::EphemerisItem::deactivate(), jeod::EphemeridesMessages::internal_error, and jeod::EphemerisItem::target_frame.

8.21.3.5 EphemerisPoint& jeod::EphemerisPoint::operator= (const EphemerisPoint &) [private]

Not implemented.

8.21.3.6 void jeod::EphemerisPoint::update (const double * *position*, const double * *velocity*, double *time*) [virtual]

Update the inertial frame's translational state.

Parameters

in	*position*	Position wrt parent Units: M
in	*velocity*	Velocity wrt parent Units: M/s
in	*time*	Timestamp Units: s

Definition at line 156 of file ephem_point.cc.

References jeod::EphemerisItem::target_frame, and jeod::EphemerisItem::update_time.

Referenced by jeod::De4xxEphemeris::ephem_update().

8.21.3.7 void jeod::EphemerisPoint::update_scaled (const double * *position*, const double * *velocity*, double *scale*, double *time*) [virtual]

Update the inertial frame's translational state.

Parameters

in	*position*	Position wrt parent Units: M
in	*velocity*	Velocity wrt parent Units: M/s
in	*scale*	Scale factor
in	*time*	Timestamp Units: s

Definition at line 178 of file ephem_point.cc.

References jeod::EphemerisItem::target_frame, and jeod::EphemerisItem::update_time.

Referenced by jeod::De4xxEphemeris::ephem_update().

8.21.3.8 EphemerisItem::TargetAspect jeod::EphemerisPoint::updates_what (void) const [virtual]

Specify the aspect of the target frame updated by the object.

EphemerisPoint objects update the translational state.

Returns

Target of object

Implements jeod::EphemerisItem.

Definition at line 199 of file ephem_point.cc.

References jeod::EphemerisItem::Translation.

8.21.4 Friends And Related Function Documentation

8.21.4.1 void init_attrjeod__EphemerisPoint ( ) [friend]

8.21.4.2 friend class InputProcessor [friend]

Definition at line 53 of file ephem_point.hh.

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

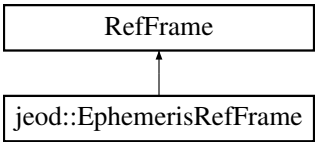
- ephem_point.hh
- ephem_point.cc

8.22 jeod::EphemerisRefFrame Class Reference

An EphemerisRefFrame is a RefFrame whose state is set by an ephemeris model.

`#include <ephem_ref_frame.hh>`

Inheritance diagram for jeod::EphemerisRefFrame:



Public Member Functions

- EphemerisRefFrame ()
  Construct an EphemerisRefFrame.
- virtual ~EphemerisRefFrame ()
  Destruct an EphemerisRefFrame.
- virtual void set_ephem_manager (BaseEphemeridesManager *manager)
  Set the EphemerisRefFrame's owner.

Protected Member Functions

- virtual void set_active_status (bool new_status)
  Augment RefFrame::set_active_status by notifying the ephemerides manager that the tree might need to be rebuilt.

Protected Attributes

- BaseEphemeridesManager * ephem_manager
  The ephemerides manager to which notifications of changes in ephemeris reference frame activity status are sent.

Private Member Functions

- EphemerisRefFrame (const EphemerisRefFrame &)

  Not implemented.
- EphemerisRefFrame & operator= (const EphemerisRefFrame &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__EphemerisRefFrame ()

8.22.1 Detailed Description

An EphemerisRefFrame is a RefFrame whose state is set by an ephemeris model.

Ephemeris reference frames come in three basic flavors:

- Planet-centered inertial frames. These are non-rotating frames whose origin is the center of mass of some planet (the Sun is a planet) and whose translational motion is (for now) assumed to be due to gravitation only. The orientation with respect to inertial is the identity transformation.
- Barycenter inertial frames. These are non-rotating frames whose origin is the center of mass of two or more planets.
- Planet-fixed frames. These are rotating frames whose origin is the center of mass of some planet (see ISSUE) and that rotate with the planet in question. The parent is always a planet-centered inertial frame with a zero translation offset between the planet-centered inertial and planet-fixed frames.

Only planet-centered inertial and barycenter inertial frames can serve as integration frames or as the root of the reference frame tree.

Definition at line 66 of file ephem_ref_frame.hh.

8.22.2 Constructor & Destructor Documentation

8.22.2.1 jeod::EphemerisRefFrame::EphemerisRefFrame (void)

Construct an EphemerisRefFrame.

Definition at line 55 of file ephem_ref_frame.cc.

8.22.2.2 jeod::EphemerisRefFrame::~EphemerisRefFrame (void) [virtual]

Destruct an EphemerisRefFrame.

Definition at line 65 of file ephem_ref_frame.cc.

8.22.2.3 jeod::EphemerisRefFrame::EphemerisRefFrame (const EphemerisRefFrame &) [private]

Not implemented.

8.22.3 Member Function Documentation

8.22.3.1 EphemerisRefFrame& jeod::EphemerisRefFrame::operator= (const EphemerisRefFrame &) [private]

Not implemented.

8.22.3.2 void jeod::EphemerisRefFrame::set_active_status (bool *new_status*) [protected], [virtual]

Augment RefFrame::set_active_status by notifying the ephemerides manager that the tree might need to be rebuilt.

Parameters

in	*new_status*	Active status

Definition at line 88 of file ephem_ref_frame.cc.

References ephem_manager, jeod::BaseEphemeridesManager::ephem_note_tree_status_change(), and jeod::EphemeridesMessages::inconsistent_setup.

8.22.3.3 void jeod::EphemerisRefFrame::set_ephem_manager (BaseEphemeridesManager * *manager*) [virtual]

Set the EphemerisRefFrame's owner.

Parameters

in,out	*manager*	Ephemeris manager

Definition at line 75 of file ephem_ref_frame.cc.

References ephem_manager.

Referenced by jeod::EphemeridesManager::set_target_frame().

8.22.4 Friends And Related Function Documentation

8.22.4.1 void init_attrjeod__EphemerisRefFrame ( ) [friend]

8.22.4.2 friend class InputProcessor [friend]

Definition at line 67 of file ephem_ref_frame.hh.

8.22.5 Field Documentation

8.22.5.1 BaseEphemeridesManager* jeod::EphemerisRefFrame::ephem_manager [protected]

The ephemerides manager to which notifications of changes in ephemeris reference frame activity status are sent.

trick_units(–)

Definition at line 95 of file ephem_ref_frame.hh.

Referenced by set_active_status(), and set_ephem_manager().

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

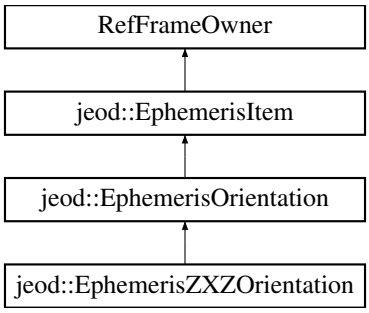
- ephem_ref_frame.hh
- ephem_ref_frame.cc

8.23 jeod::EphemerisZXZOrientation Class Reference

The EphemerisZXZOrientation is an EphemerisOrientation subclass that updates orientation based on an Z-X-Z Euler sequence and the time derivatives of this sequence.

`#include <ephem_orient_zxz.hh>`

Inheritance diagram for jeod::EphemerisZXZOrientation:



Public Member Functions

- EphemerisZXZOrientation ()

  Construct an ephemeris angle.
- virtual ~EphemerisZXZOrientation ()

  Destroy an ephemeris angle.
- const double * get_euler_angles () const

  Return the Euler angles.
- void get_euler_angles (double *angles) const

  Return the Euler angles.
- const double * get_euler_rates () const

  Return the Euler rates.
- void get_euler_rates (double *rates) const

  Return the Euler angles.
- virtual void update (const double *angles, const double *derivs, double time)

  Compute a JEOD rotational state given a 3-1-3 inertial-to-planet-fixed Euler sequence and the time derivatives of the Euler angles.
- virtual void propagate (double to_time)

  Propagate the orientation to the current time.

Protected Attributes

- double euler_angle_313 [3]

  Astronomical (zxz) Euler angles.
- double euler_rate_313 [3]

  Time derivatives of the zyz Euler angles.

Private Member Functions

- EphemerisZXZOrientation (const EphemerisZXZOrientation &)

  Not implemented.
- EphemerisZXZOrientation & operator= (const EphemerisZXZOrientation &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__EphemerisZXZOrientation ()

Additional Inherited Members

8.23.1 Detailed Description

The EphemerisZXZOrientation is an EphemerisOrientation subclass that updates orientation based on an Z-X-Z Euler sequence and the time derivatives of this sequence.

Definition at line 56 of file ephem_orient_zxz.hh.

8.23.2 Constructor & Destructor Documentation

8.23.2.1 jeod::EphemerisZXZOrientation::EphemerisZXZOrientation (void)

Construct an ephemeris angle.

Definition at line 82 of file ephem_orient_zxz.cc.

References euler_angle_313, and euler_rate_313.

8.23.2.2 jeod::EphemerisZXZOrientation::~EphemerisZXZOrientation (void) [virtual]

Destroy an ephemeris angle.

Definition at line 95 of file ephem_orient_zxz.cc.

8.23.2.3 jeod::EphemerisZXZOrientation::EphemerisZXZOrientation (const EphemerisZXZOrientation &) [private]

Not implemented.

8.23.3 Member Function Documentation

8.23.3.1 const double * jeod::EphemerisZXZOrientation::get_euler_angles (void) const

Return the Euler angles.

Returns

Euler angles

Definition at line 107 of file ephem_orient_zxz.cc.

References euler_angle_313.

8.23.3.2 void jeod::EphemerisZXZOrientation::get_euler_angles (double * *angles*) const

Return the Euler angles.

Parameters

out	*angles*	Euler angles Units: r

Definition at line 120 of file ephem_orient_zxz.cc.

References euler_angle_313.

8.23.3.3 const double * jeod::EphemerisZXZOrientation::get_euler_rates (void) const

Return the Euler rates.

Returns

Euler rates

Definition at line 137 of file ephem_orient_zxz.cc.

References euler_rate_313.

8.23.3.4 void jeod::EphemerisZXZOrientation::get_euler_rates (double * *rates*) const

Return the Euler angles.

Parameters

out	*rates*	Euler rates Units: r/s

Definition at line 150 of file ephem_orient_zxz.cc.

References euler_rate_313.

8.23.3.5 EphemerisZXZOrientation& jeod::EphemerisZXZOrientation::operator= (const EphemerisZXZOrientation &) [private]

Not implemented.

8.23.3.6 void jeod::EphemerisZXZOrientation::propagate (double *to_time*) [virtual]

Propagate the orientation to the current time.

Assumptions and Limitations

- The orientation has been computed.
- Rotation is constant during the update interval.

Parameters

in	*to_time*	Target dynamic time Units: s

Definition at line 169 of file ephem_orient_zxz.cc.

References EPSILON_TIME, jeod::EphemerisItem::target_frame, TAYLOR_CUTOFF, and jeod::EphemerisItem::update_time.

Referenced by jeod::De4xxEphemeris::propagate_lunar_rnp().

8.23.3.7 void jeod::EphemerisZXZOrientation::update (const double * *angles*, const double * *derivs*, double *time*) [virtual]

Compute a JEOD rotational state given a 3-1-3 inertial-to-planet-fixed Euler sequence and the time derivatives of the Euler angles.

Parameters

in	*angles*	zxz Euler angles Units: r
in	*derivs*	zxz Euler angle time derivatives Units: r/s
in	*time*	Update time Units: s

Definition at line 232 of file ephem_orient_zxz.cc.

References euler_angle_313, euler_rate_313, jeod::EphemerisItem::target_frame, and jeod::EphemerisItem::update_time.

Referenced by jeod::De4xxEphemeris::ephem_update().

8.23.4 Friends And Related Function Documentation

8.23.4.1 void init_attrjeod__EphemerisZXZOrientation ( ) [friend]

8.23.4.2 friend class InputProcessor [friend]

Definition at line 57 of file ephem_orient_zxz.hh.

8.23.5 Field Documentation

8.23.5.1 double jeod::EphemerisZXZOrientation::euler_angle_313[3] [protected]

Astronomical (zxz) Euler angles.

trick_units(radian)

Definition at line 96 of file ephem_orient_zxz.hh.

Referenced by EphemerisZXZOrientation(), get_euler_angles(), and update().

8.23.5.2 double jeod::EphemerisZXZOrientation::euler_rate_313[3] [protected]

Time derivatives of the zyz Euler angles.

trick_units(radian/s)

Definition at line 101 of file ephem_orient_zxz.hh.

Referenced by EphemerisZXZOrientation(), get_euler_rates(), and update().

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

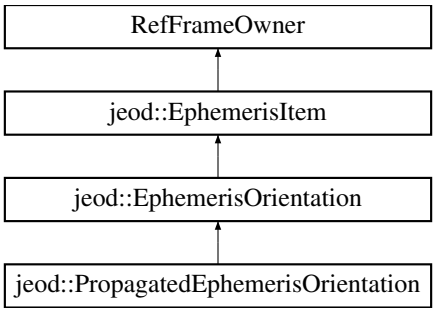
- ephem_orient_zxz.hh
- ephem_orient_zxz.cc

8.24 jeod::PropagatedEphemerisOrientation Class Reference

A PropagatedEphemerisOrientation is an EphemerisOrientation whose state is coupled with the rotational state of a DynBody reference frame.

`#include <propagated_planet.hh>`

Inheritance diagram for jeod::PropagatedEphemerisOrientation:



Public Member Functions

- PropagatedEphemerisOrientation (DynBody &dyn_body, BodyRefFrame &frame)

  PropagatedEphemerisOrientation non-default constructor.
- ~PropagatedEphemerisOrientation (void)

  PropagatedEphemerisOrientation destructor.
- virtual void update (double time)

  Copy rotational state from/to the body reference frame.

Protected Attributes

- DynBody & body

  The dynamic body whose state is tied to that of the planet.
- BodyRefFrame & body_ref_frame

  The body reference frame whose rotational state is coupled with that of the planet's planet-fixed frame.

Private Member Functions

- PropagatedEphemerisOrientation (const PropagatedEphemerisOrientation &)

  Not implemented.
- PropagatedEphemerisOrientation & operator= (const PropagatedEphemerisOrientation &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__PropagatedEphemerisOrientation ()

Additional Inherited Members

8.24.1 Detailed Description

A PropagatedEphemerisOrientation is an EphemerisOrientation whose state is coupled with the rotational state of a DynBody reference frame.

This class is intended for use by the PropagatedPlanet class. Use outside of the PropagatedPlanet is not sanctioned.

The class acts analogously to the class PropagatedEphemerisPlanet, but for rotation rather than translation. See PropagatedEphemerisPlanet for a description of the behavior of the class.

Definition at line 146 of file propagated_planet.hh.

8.24.2 Constructor & Destructor Documentation

8.24.2.1 jeod::PropagatedEphemerisOrientation::PropagatedEphemerisOrientation (DynBody & *dyn_body*, BodyRefFrame & *frame*)

PropagatedEphemerisOrientation non-default constructor.

Parameters

in,out	*dyn_body*	The DynBody that represents the planet
in,out	*frame*	The body reference frame

Definition at line 129 of file propagated_planet.cc.

8.24.2.2 jeod::PropagatedEphemerisOrientation::~PropagatedEphemerisOrientation (void)

PropagatedEphemerisOrientation destructor.

Definition at line 144 of file propagated_planet.cc.

8.24.2.3 jeod::PropagatedEphemerisOrientation::PropagatedEphemerisOrientation (const PropagatedEphemerisOrientation &) [private]

Not implemented.

8.24.3 Member Function Documentation

8.24.3.1 PropagatedEphemerisOrientation& jeod::PropagatedEphemerisOrientation::operator= (const PropagatedEphemerisOrientation &) [private]

Not implemented.

8.24.3.2 void jeod::PropagatedEphemerisOrientation::update (double *dyn_time*) [virtual]

Copy rotational state from/to the body reference frame.

Parameters

in	*dyn_time*	Dynamic time seconds Units: s

Definition at line 156 of file propagated_planet.cc.

References body, body_ref_frame, jeod::EphemerisItem::enabled, jeod::EphemerisItem::target_frame, and jeod::EphemerisItem::update_time.

Referenced by jeod::PropagatedPlanet::ephem_update().

8.24.4 Friends And Related Function Documentation

8.24.4.1 void init_attrjeod__PropagatedEphemerisOrientation ( ) [friend]

8.24.4.2 friend class InputProcessor [friend]

Definition at line 147 of file propagated_planet.hh.

8.24.5 Field Documentation

8.24.5.1 DynBody& jeod::PropagatedEphemerisOrientation::body [protected]

The dynamic body whose state is tied to that of the planet.

trick_units(–)

Definition at line 175 of file propagated_planet.hh.

Referenced by update().

8.24.5.2 BodyRefFrame& jeod::PropagatedEphemerisOrientation::body_ref_frame [protected]

The body reference frame whose rotational state is coupled with that of the planet's planet-fixed frame.

trick_units(–)

Definition at line 181 of file propagated_planet.hh.

Referenced by update().

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

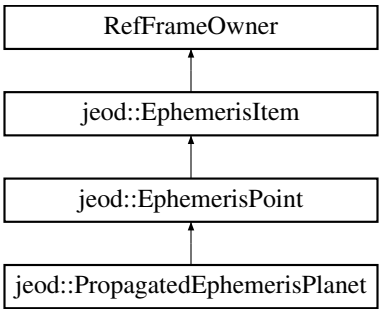
- propagated_planet.hh
- propagated_planet.cc

8.25 jeod::PropagatedEphemerisPlanet Class Reference

A PropagatedEphemerisPlanet is an EphemerisPoint whose state is coupled with the translational state of a DynBody reference frame.

`#include <propagated_planet.hh>`

Inheritance diagram for jeod::PropagatedEphemerisPlanet:



Public Member Functions

- PropagatedEphemerisPlanet (DynBody &dyn_body, BodyRefFrame &frame)

  PropagatedEphemerisPlanet non-default constructor.
- ~PropagatedEphemerisPlanet (void)

  PropagatedEphemerisPlanet destructor.
- virtual void update (double time)

  Copy rotational state from/to the body reference frame.

Protected Attributes

- DynBody & body

  The dynamic body whose state is tied to that of the planet.
- BodyRefFrame & body_ref_frame

  The body reference frame whose translational state is coupled with that of the planet's inertial frame.

Private Member Functions

- PropagatedEphemerisPlanet (const PropagatedEphemerisPlanet &)

  Not implemented.
- PropagatedEphemerisPlanet & operator= (const PropagatedEphemerisPlanet &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__PropagatedEphemerisPlanet ()

Additional Inherited Members

8.25.1 Detailed Description

A PropagatedEphemerisPlanet is an EphemerisPoint whose state is coupled with the translational state of a DynBody reference frame.

This class is intended for use by the PropagatedPlanet class. Use outside of the PropagatedPlanet is not sanctioned.

The inherited enabled flag takes on an additional meaning in this derived class. The base class meaning of this flag is that the ephemeris item, or the ephemeris model that owns the item, is responsible for maintaining the translational state of the target frame when the item is enabled but not when the item is disabled.

For this class, when the enabled flag is set still means that the item is responsible for maintaining the translational state of the target ephemeris reference frame. The dynamic body serves as the source of this state. When the enabled flag is clear, some other model is assumed to be responsible for maintaining the target frame state. The target frame state serves as the source of the dynamic body's state when the enabled flag is clear. A PropagatedEphemerisPlanet is an EphemerisPoint whose state is coupled with the translational state of a DynBody reference frame.

This class is intended for use by the PropagatedPlanet class. Use outside of the PropagatedPlanet is not sanctioned.

The class acts analogously to the class PropagatedEphemerisOrientation, but for translation rather than rotation. See PropagatedEphemerisOrientation for a description of the behavior of the class.

Definition at line 86 of file propagated_planet.hh.

8.25.2 Constructor & Destructor Documentation

8.25.2.1 jeod::PropagatedEphemerisPlanet::PropagatedEphemerisPlanet (DynBody & *dyn_body,* BodyRefFrame & *frame*)

PropagatedEphemerisPlanet non-default constructor.

Parameters

in,out	*dyn_body*	The DynBody that represents the planet
in,out	*frame*	The body reference frame

Definition at line 190 of file propagated_planet.cc.

8.25.2.2 jeod::PropagatedEphemerisPlanet::~PropagatedEphemerisPlanet (void)

PropagatedEphemerisPlanet destructor.

Definition at line 83 of file propagated_planet.cc.

8.25.2.3 jeod::PropagatedEphemerisPlanet::PropagatedEphemerisPlanet (const PropagatedEphemerisPlanet &) [private]

Not implemented.

8.25.3 Member Function Documentation

8.25.3.1 PropagatedEphemerisPlanet& jeod::PropagatedEphemerisPlanet::operator= (const PropagatedEphemerisPlanet &) [private]

Not implemented.

8.25.3.2 void jeod::PropagatedEphemerisPlanet::update (double *dyn_time*) [virtual]

Copy rotational state from/to the body reference frame.

Parameters

in	*dyn_time*	Dynamic time seconds Units: s

Definition at line 95 of file propagated_planet.cc.

References body, body_ref_frame, jeod::EphemerisItem::enabled, jeod::EphemerisItem::target_frame, and jeod::EphemerisItem::update_time.

Referenced by jeod::PropagatedPlanet::ephem_update().

8.25.4 Friends And Related Function Documentation

8.25.4.1 void init_attrjeod__PropagatedEphemerisPlanet ( ) [friend]

8.25.4.2 friend class InputProcessor [friend]

Definition at line 87 of file propagated_planet.hh.

8.25.5 Field Documentation

8.25.5.1 DynBody& jeod::PropagatedEphemerisPlanet::body [protected]

The dynamic body whose state is tied to that of the planet.

trick_units(–)

Definition at line 124 of file propagated_planet.hh.

Referenced by update().

8.25.5.2 BodyRefFrame& jeod::PropagatedEphemerisPlanet::body_ref_frame [protected]

The body reference frame whose translational state is coupled with that of the planet's inertial frame.

trick_units(–)

Definition at line 130 of file propagated_planet.hh.

Referenced by update().

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

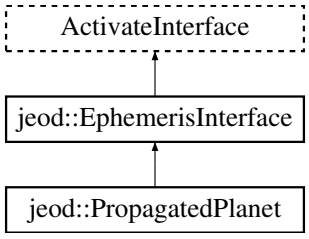
- propagated_planet.hh
- propagated_planet.cc

8.26 jeod::PropagatedPlanet Class Reference

The PropagatedPlanet ephemeris model provides planetary state via a DynBody object whose state is propagated using the JEOD state integration techniques.

`#include <propagated_planet.hh>`

Inheritance diagram for jeod::PropagatedPlanet:



Public Types

- enum Mode { TransFromPlanet_RotFromPlanet = 0, TransFromPlanet_RotFromBody = 1, TransFromBody_RotFromPlanet = 2, TransFromBody_RotFromBody = 3 }

  Defines the modes in which an active PropagatedPlanet object operates.

Public Member Functions

- PropagatedPlanet (void)

  PropagatedPlanet default constructor.
- ~PropagatedPlanet (void)

  PropagatedPlanet destructor.
- void initialize_model (const TimeManager &time_manager, DynManager &dyn_manager)

  Initialize a PropagatedPlanet model.
- void shutdown (void)

  Free resources allocated by the PropagatedPlanet model.
- void activate (void)

  Nominally, activate the object.
- void deactivate (void)

Deactivate the PropagatedPlanet object.

- double timestamp (void) const

  Return time of last update.
- const char * get_name (void) const

  Return model name.
- void ephem_initialize (EphemeridesManager &ephem_manager)

  Mark appropriate items in the model as active.
- void ephem_activate (EphemeridesManager &ephem_manager)

  Activate ephemerides.
- void ephem_build_tree (EphemeridesManager &ephem_manager)

  Construct the ephemeris model portions of the reference frame tree.
- void ephem_update (void)

  Update ephemerides for subscribed items.
- void set_commanded_mode (Mode new_mode)

  Setter for the commanded mode.

Data Fields

- char * planet_name

  The name of the planet.
- char * parent_name

  The name of the parent frame.
- DynBody body

  The dynamic body whose state is tied to that of the planet.
- Mode commanded_mode

  The mode in which the model should operate.

Protected Member Functions

- void set_mode (void)

  Change the behavior of a PropagatedPlanet.

Protected Attributes

- bool initialized

  Has the model been initialized?
- Mode mode

  The mode in which the model is operating.
- char * ident

  Model name; used for reporting errors.
- bool active

  Is the planet present and marked as active?
- double update_time

  Time of last update, dynamic time seconds.
- BasePlanet * planet

  The planet tied to the body.
- EphemerisRefFrame * parent_frame

  The parent of the planet.
- DynManager * dyn_manager

  The dynamics manager.

- const TimeDyn * time_dyn

  The source of dynamic time information.
- PropagatedEphemerisPlanet ephem_planet

  The ephemeris item that couples the translational states of the body's composite body frame and the planet's inertial frame.
- PropagatedEphemerisOrientation ephem_orient

  The ephemeris item that couples the rotational states of the body's composite body frame and the planet's planet-fixed frame.

Private Member Functions

- PropagatedPlanet (const PropagatedPlanet &)

  Not implemented.
- PropagatedPlanet & operator= (const PropagatedPlanet &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__PropagatedPlanet ()

8.26.1 Detailed Description

The PropagatedPlanet ephemeris model provides planetary state via a DynBody object whose state is propagated using the JEOD state integration techniques.

Scenarios in which a simulation will use a PropagatedPlanet object include:

- An object such as an asteroid for which an ephemeris model is not readily available.
- An object such as a planet that is represented in some other ephemeris model but the simulation developer wants the planet to be propagated to ensure that the planet and the vehicles operating in the vicinity of the planet obey the same laws of physics.

The PropagatedPlanet model provides mechanisms that accommodate these scenarios. The class defines multiple modes in which a propagated planet planet object operates. In all modes, the model ensures consistency between the translational states of the dynamic body's composite frame and the planet's planet-centered frame and between the rotational states of the dynamic body's composite frame and the planet's planet-fixed frame.

Definition at line 213 of file propagated_planet.hh.

8.26.2 Member Enumeration Documentation

8.26.2.1 enum jeod::PropagatedPlanet::Mode

Defines the modes in which an active PropagatedPlanet object operates.

A PropagatedPlanet contains a BasePlanet pointer PropagatedPlanet::planet and a DynBody PropagatedPlanet::body. The translational states of the planet-centered inertial frame and the body's composite_body frame are tied to one another, as are the rotational states of the planet-fixed frame and the the body's composite_body frame. This enum identifies which of the planet or the body is the source of translational and the the rotational parts of the state.

Enumerator

TransFromPlanet_RotFromPlanet

TransFromPlanet_RotFromBody

TransFromBody_RotFromPlanet

TransFromBody_RotFromBody

Definition at line 230 of file propagated_planet.hh.

8.26.3 Constructor & Destructor Documentation

8.26.3.1 jeod::PropagatedPlanet::PropagatedPlanet (void)

PropagatedPlanet default constructor.

Definition at line 205 of file propagated_planet.cc.

References ephem_orient, ephem_planet, and jeod::EphemerisItem::set_owner().

8.26.3.2 jeod::PropagatedPlanet::~PropagatedPlanet (void)

PropagatedPlanet destructor.

Definition at line 232 of file propagated_planet.cc.

References shutdown().

8.26.3.3 jeod::PropagatedPlanet::PropagatedPlanet (const PropagatedPlanet &) [private]

Not implemented.

8.26.4 Member Function Documentation

8.26.4.1 void jeod::PropagatedPlanet::activate (void)

Nominally, activate the object.

In the case of a PropagatedPlanet object, an inactive object cannot be activated.

Definition at line 260 of file propagated_planet.cc.

References active, and jeod::EphemeridesMessages::internal_error.

8.26.4.2 void jeod::PropagatedPlanet::deactivate (void)

Deactivate the PropagatedPlanet object.

Definition at line 277 of file propagated_planet.cc.

References active.

8.26.4.3 void jeod::PropagatedPlanet::ephem_activate (EphemeridesManager & *ephem_manager*) [virtual]

Activate ephemerides.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::EphemerisInterface.

Definition at line 514 of file propagated_planet.cc.

References body, mode, TransFromBody_RotFromBody, TransFromBody_RotFromPlanet, and TransFromPlanet_-RotFromBody.

8.26.4.4 void jeod::PropagatedPlanet::ephem_build_tree (EphemeridesManager & *ephem_manager*) [virtual]

Construct the ephemeris model portions of the reference frame tree.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::EphemerisInterface.

Definition at line 536 of file propagated_planet.cc.

References active, parent_frame, and planet.

8.26.4.5 void jeod::PropagatedPlanet::ephem_initialize (EphemeridesManager & *ephem_manager*) [virtual]

Mark appropriate items in the model as active.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::EphemerisInterface.

Definition at line 398 of file propagated_planet.cc.

References active, ephem_orient, ephem_planet, jeod::EphemeridesManager::find_base_planet(), jeod::-EphemerisItem::get_target_frame(), jeod::EphemeridesMessages::inconsistent_setup, planet, planet_name, set_-mode(), and jeod::EphemeridesManager::set_target_frame().

8.26.4.6 void jeod::PropagatedPlanet::ephem_update (void) [virtual]

Update ephemerides for subscribed items.

Implements jeod::EphemerisInterface.

Definition at line 557 of file propagated_planet.cc.

References active, body, commanded_mode, dyn_manager, ephem_orient, ephem_planet, initialized, mode, set_-mode(), time_dyn, TransFromBody_RotFromBody, jeod::PropagatedEphemerisPlanet::update(), jeod::Propagated-EphemerisOrientation::update(), and update_time.

8.26.4.7 const char * jeod::PropagatedPlanet::get_name (void) const [virtual]

Return model name.

Returns

Name

Implements jeod::EphemerisInterface.

Definition at line 304 of file propagated_planet.cc.

References ident.

8.26.4.8 void jeod::PropagatedPlanet::initialize_model (const TimeManager & *time_manager*, DynManager & *dyn_manager_ref*)

Initialize a PropagatedPlanet model.

Assumptions and Limitations

- This method must be called after the ephemeris model that defines the parent frame has been initialized.

Parameters

in	*time_manager*	Time manager
in,out	*dyn_manager_-ref*	Dynamics manager

Definition at line 322 of file propagated_planet.cc.

References active, body, dyn_manager, ephem_orient, ephem_planet, ident, jeod::EphemeridesMessages-::inconsistent_setup, parent_frame, parent_name, planet_name, jeod::EphemerisItem::set_name(), and time_dyn.

8.26.4.9 PropagatedPlanet& jeod::PropagatedPlanet::operator= (const PropagatedPlanet &) [private]

Not implemented.

8.26.4.10 void jeod::PropagatedPlanet::set_commanded_mode (PropagatedPlanet::Mode *new_mode*)

Setter for the commanded mode.

Parameters

in	*new_mode*	New commanded mode

Definition at line 455 of file propagated_planet.cc.

References commanded_mode.

8.26.4.11 void jeod::PropagatedPlanet::set_mode (void) [protected]

Change the behavior of a PropagatedPlanet.

Definition at line 466 of file propagated_planet.cc.

References commanded_mode, jeod::EphemerisItem::disable(), dyn_manager, jeod::EphemerisOrientation-::enable(), jeod::EphemerisItem::enable(), ephem_orient, ephem_planet, jeod::EphemeridesMessages::inconsistent-_setup, mode, TransFromBody_RotFromBody, TransFromBody_RotFromPlanet, TransFromPlanet_RotFromBody, and TransFromPlanet_RotFromPlanet.

Referenced by ephem_initialize(), and ephem_update().

8.26.4.12 void jeod::PropagatedPlanet::shutdown (void)

Free resources allocated by the PropagatedPlanet model.

Definition at line 243 of file propagated_planet.cc.

References ident.

Referenced by ~PropagatedPlanet().

8.26.4.13 double jeod::PropagatedPlanet::timestamp (void) const [virtual]

Return time of last update.

Returns

Timestamp
Units: day

Implements jeod::EphemerisInterface.

Definition at line 291 of file propagated_planet.cc.

References update_time.

8.26.5 Friends And Related Function Documentation

8.26.5.1 void init_attrjeod__PropagatedPlanet ( ) [friend]

8.26.5.2 friend class InputProcessor [friend]

Definition at line 214 of file propagated_planet.hh.

8.26.6 Field Documentation

8.26.6.1 bool jeod::PropagatedPlanet::active [protected]

Is the planet present and marked as active?

trick_units(–)

Definition at line 345 of file propagated_planet.hh.

Referenced by activate(), deactivate(), ephem_build_tree(), ephem_initialize(), ephem_update(), and initialize_model().

8.26.6.2 DynBody jeod::PropagatedPlanet::body

The dynamic body whose state is tied to that of the planet.

trick_units(–)

Definition at line 309 of file propagated_planet.hh.

Referenced by ephem_activate(), ephem_update(), and initialize_model().

8.26.6.3 Mode jeod::PropagatedPlanet::commanded_mode

The mode in which the model should operate.

trick_units(–)

Definition at line 314 of file propagated_planet.hh.

Referenced by ephem_update(), set_commanded_mode(), and set_mode().

8.26.6.4 DynManager* jeod::PropagatedPlanet::dyn_manager [protected]

The dynamics manager.

trick_units(–)

Definition at line 365 of file propagated_planet.hh.

Referenced by ephem_update(), initialize_model(), and set_mode().

8.26.6.5 PropagatedEphemerisOrientation jeod::PropagatedPlanet::ephem_orient [protected]

The ephemeris item that couples the rotational states of the body's composite body frame and the planet's planet-fixed frame.

trick_units(–)

Definition at line 382 of file propagated_planet.hh.

Referenced by ephem_initialize(), ephem_update(), initialize_model(), PropagatedPlanet(), and set_mode().

8.26.6.6 PropagatedEphemerisPlanet jeod::PropagatedPlanet::ephem_planet [protected]

The ephemeris item that couples the translational states of the body's composite body frame and the planet's inertial frame.

trick_units(–)

Definition at line 376 of file propagated_planet.hh.

Referenced by ephem_initialize(), ephem_update(), initialize_model(), PropagatedPlanet(), and set_mode().

8.26.6.7 char* jeod::PropagatedPlanet::ident [protected]

Model name; used for reporting errors.

trick_units(–)

Definition at line 340 of file propagated_planet.hh.

Referenced by get_name(), initialize_model(), and shutdown().

8.26.6.8 bool jeod::PropagatedPlanet::initialized [protected]

Has the model been initialized?

trick_units(–)

Definition at line 330 of file propagated_planet.hh.

Referenced by ephem_update().

8.26.6.9 Mode jeod::PropagatedPlanet::mode [protected]

The mode in which the model is operating.

trick_units(–)

Definition at line 335 of file propagated_planet.hh.

Referenced by ephem_activate(), ephem_update(), and set_mode().

8.26.6.10 EphemerisRefFrame* jeod::PropagatedPlanet::parent_frame [protected]

The parent of the planet.

trick_units(–)

Definition at line 360 of file propagated_planet.hh.

Referenced by ephem_build_tree(), and initialize_model().

8.26.6.11 char* jeod::PropagatedPlanet::parent_name

The name of the parent frame.

This is used at initialization time only.trick_units(–)

Definition at line 304 of file propagated_planet.hh.

Referenced by initialize_model().

8.26.6.12 BasePlanet* jeod::PropagatedPlanet::planet [protected]

The planet tied to the body.

trick_units(–)

Definition at line 355 of file propagated_planet.hh.

Referenced by ephem_build_tree(), and ephem_initialize().

8.26.6.13 char* jeod::PropagatedPlanet::planet_name

The name of the planet.

This is used at initialization time only.trick_units(–)

Definition at line 298 of file propagated_planet.hh.

Referenced by ephem_initialize(), and initialize_model().

8.26.6.14 const TimeDyn* jeod::PropagatedPlanet::time_dyn [protected]

The source of dynamic time information.

trick_units(–)

Definition at line 370 of file propagated_planet.hh.

Referenced by ephem_update(), and initialize_model().

8.26.6.15 double jeod::PropagatedPlanet::update_time [protected]

Time of last update, dynamic time seconds.

trick_units(s)

Definition at line 350 of file propagated_planet.hh.

Referenced by ephem_update(), and timestamp().

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

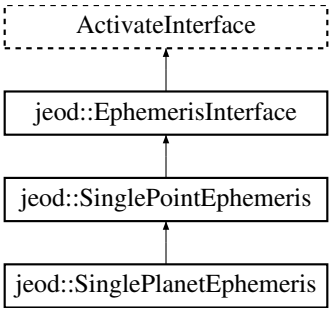
- propagated_planet.hh
- propagated_planet.cc

8.27 jeod::SinglePlanetEphemeris Class Reference

A space with one gravitation body has one ephemeris point.

`#include <simple_ephemerides.hh>`

Inheritance diagram for jeod::SinglePlanetEphemeris:



Public Member Functions

- SinglePlanetEphemeris (void)

  Construct an SinglePlanetEphemeris object.
- virtual ~SinglePlanetEphemeris (void)

  Destruct an SinglePlanetEphemeris object.
- virtual void set_name (const char *frame_name)

  Set the name of a SinglePlanetEphemeris object.
- virtual void initialize_model (EphemeridesManager &ephem_manager)

  Initialize a SinglePlanetEphemeris object.
- virtual void ephem_initialize (EphemeridesManager &ephem_manager)

  Initialize a SinglePlanetEphemeris object.
- virtual void ephem_activate (EphemeridesManager &ephem_manager)

  Activate a SinglePlanetEphemeris object.
- virtual void ephem_build_tree (EphemeridesManager &ephem_manager)

  Build the reference frame tree with the central frame as the root.

Protected Attributes

- EphemerisPoint central_point

  The EphemerisPoint that represents the center of a simple universe.

Private Member Functions

- SinglePlanetEphemeris (const SinglePlanetEphemeris &)

  Not implemented.
- SinglePlanetEphemeris & operator= (const SinglePlanetEphemeris &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__SinglePlanetEphemeris ()

8.27.1 Detailed Description

A space with one gravitation body has one ephemeris point.

Note well: A SinglePlanetEphemeris does not contain a Planet object. The planet must be specified elsewhere.

Definition at line 223 of file simple_ephemerides.hh.

8.27.2 Constructor & Destructor Documentation

8.27.2.1 jeod::SinglePlanetEphemeris::SinglePlanetEphemeris (void)

Construct an SinglePlanetEphemeris object.

Definition at line 288 of file simple_ephemerides.cc.

References central_point, jeod::EphemerisItem::enable(), and jeod::EphemerisItem::set_owner().

8.27.2.2 jeod::SinglePlanetEphemeris::~SinglePlanetEphemeris (void) [virtual]

Destruct an SinglePlanetEphemeris object.

Definition at line 299 of file simple_ephemerides.cc.

8.27.2.3 jeod::SinglePlanetEphemeris::SinglePlanetEphemeris (const SinglePlanetEphemeris &) [private]

Not implemented.

8.27.3 Member Function Documentation

8.27.3.1 void jeod::SinglePlanetEphemeris::ephem_activate (EphemeridesManager & *ephem_manager*) [virtual]

Activate a SinglePlanetEphemeris object.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::SinglePointEphemeris.

Definition at line 400 of file simple_ephemerides.cc.

8.27.3.2 void jeod::SinglePlanetEphemeris::ephem_build_tree (EphemeridesManager & *ephem_manager*) [virtual]

Build the reference frame tree with the central frame as the root.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::SinglePointEphemeris.

Definition at line 412 of file simple_ephemerides.cc.

References jeod::SinglePointEphemeris::active, central_point, and jeod::EphemerisItem::get_target_frame().

8.27.3.3 void jeod::SinglePlanetEphemeris::ephem_initialize (EphemeridesManager & *ephem_manager*) [virtual]

Initialize a SinglePlanetEphemeris object.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::SinglePointEphemeris.

Definition at line 355 of file simple_ephemerides.cc.

References jeod::SinglePointEphemeris::active, central_point, jeod::SinglePointEphemeris::deactivate(), jeod::EphemerisItem::disable(), jeod::EphemeridesManager::find_base_planet(), jeod::EphemeridesManager::get_num_planets(), jeod::EphemerisItem::get_target_frame(), jeod::SinglePointEphemeris::identifier, and jeod::EphemeridesMessages::inconsistent_setup.

8.27.3.4 void jeod::SinglePlanetEphemeris::initialize_model (EphemeridesManager & *ephem_manager*) [virtual]

Initialize a SinglePlanetEphemeris object.

Parameters

in,out	*ephem_manager*	Ephemerides manager

Implements jeod::SinglePointEphemeris.

Definition at line 336 of file simple_ephemerides.cc.

References jeod::SinglePointEphemeris::active, jeod::EphemeridesManager::add_ephem_item(), jeod::EphemeridesManager::add_ephemeris(), and central_point.

8.27.3.5 SinglePlanetEphemeris& jeod::SinglePlanetEphemeris::operator= (const SinglePlanetEphemeris &) [private]

Not implemented.

8.27.3.6 void jeod::SinglePlanetEphemeris::set_name (const char * *new_name*) [virtual]

Set the name of a SinglePlanetEphemeris object.

Parameters

in,out	*new_name*	Central point name

Reimplemented from jeod::SinglePointEphemeris.

Definition at line 311 of file simple_ephemerides.cc.

References central_point, jeod::SinglePointEphemeris::set_name(), and jeod::EphemerisItem::set_name().

8.27.4 Friends And Related Function Documentation

8.27.4.1 void init_attrjeod__SinglePlanetEphemeris ( ) [friend]

8.27.4.2 friend class InputProcessor [friend]

Definition at line 225 of file simple_ephemerides.hh.

8.27.5 Field Documentation

8.27.5.1 EphemerisPoint jeod::SinglePlanetEphemeris::central_point [protected]

The EphemerisPoint that represents the center of a simple universe.

trick_units(–)

Definition at line 254 of file simple_ephemerides.hh.

Referenced by ephem_build_tree(), ephem_initialize(), initialize_model(), set_name(), and SinglePlanetEphemeris().

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

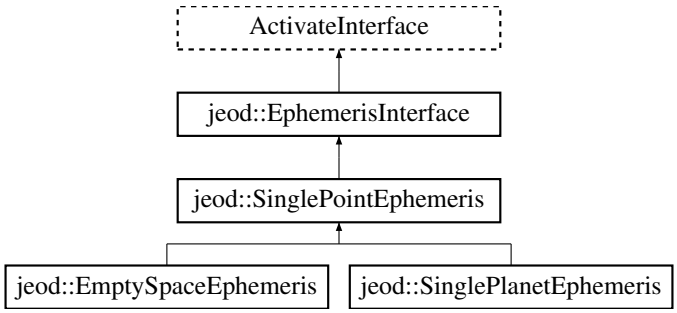
- simple_ephemerides.hh
- simple_ephemerides.cc

8.28 jeod::SinglePointEphemeris Class Reference

A SinglePointEphemeris has one ephemeris point.

`#include <simple_ephemerides.hh>`

Inheritance diagram for jeod::SinglePointEphemeris:



Public Member Functions

- SinglePointEphemeris (void)

  Construct a SinglePointEphemeris object.
- virtual ~SinglePointEphemeris (void)

  Destruct a SinglePointEphemeris object.
- virtual void set_name (const char *new_name)

  Set the name of a SinglePointEphemeris object.
- virtual void activate (void)

  Nominally, activate the model.
- virtual void deactivate (void)

  Deactivate the model.
- virtual double timestamp (void) const

  Retrieve the timestamp.
- virtual const char * get_name (void) const

  Retrieve the identifier.
- virtual void ephem_update (void)

  Update the ephemerides, which in this case is a no-op.
- virtual void initialize_model (EphemeridesManager &manager)=0

  Register the model and its ephemeris points.
- virtual void ephem_initialize (EphemeridesManager &manager)=0

  Initialize the ephemerides.
- virtual void ephem_activate (EphemeridesManager &manager)=0

  Activate the model.
- virtual void ephem_build_tree (EphemeridesManager &manager)=0

  Build the model's contribution to the reference frame tree.

Protected Attributes

- char * identifier

  Identifier for this model.
- double update_time

  Time of last update, dynamic time seconds.
- bool active

  Is the model active?

Private Member Functions

- SinglePointEphemeris (const SinglePointEphemeris &)

  Not implemented.
- SinglePointEphemeris & operator= (const SinglePointEphemeris &)

  Not implemented.

Friends

- class InputProcessor
- void init_attrjeod__SinglePointEphemeris ()

8.28.1 Detailed Description

A SinglePointEphemeris has one ephemeris point.

Definition at line 51 of file simple_ephemerides.hh.

8.28.2 Constructor & Destructor Documentation

8.28.2.1 jeod::SinglePointEphemeris::SinglePointEphemeris (void)

Construct a SinglePointEphemeris object.

Definition at line 67 of file simple_ephemerides.cc.

8.28.2.2 jeod::SinglePointEphemeris::~SinglePointEphemeris (void) [virtual]

Destruct a SinglePointEphemeris object.

Definition at line 81 of file simple_ephemerides.cc.

References identifier.

8.28.2.3 jeod::SinglePointEphemeris::SinglePointEphemeris (const SinglePointEphemeris &) [private]

Not implemented.

8.28.3 Member Function Documentation

8.28.3.1 void jeod::SinglePointEphemeris::activate (void) [virtual]

Nominally, activate the model.

Here, reject the request.

Definition at line 94 of file simple_ephemerides.cc.

References active, and jeod::EphemeridesMessages::internal_error.

8.28.3.2 void jeod::SinglePointEphemeris::deactivate (void) [virtual]

Deactivate the model.

Definition at line 109 of file simple_ephemerides.cc.

References active.

Referenced by jeod::EmptySpaceEphemeris::ephem_initialize(), and jeod::SinglePlanetEphemeris::ephem_initialize().

8.28.3.3 virtual void jeod::SinglePointEphemeris::ephem_activate (EphemeridesManager & *manager*) [pure virtual]

Activate the model.

Parameters

in,out	*manager*	Ephemerides manager

Implements jeod::EphemerisInterface.

Implemented in jeod::SinglePlanetEphemeris, and jeod::EmptySpaceEphemeris.

8.28.3.4 virtual void jeod::SinglePointEphemeris::ephem_build_tree (EphemeridesManager & *manager*) [pure virtual]

Build the model's contribution to the reference frame tree.

Parameters

in,out	*manager*	Ephemerides manager

Implements jeod::EphemerisInterface.

Implemented in jeod::SinglePlanetEphemeris, and jeod::EmptySpaceEphemeris.

8.28.3.5 virtual void jeod::SinglePointEphemeris::ephem_initialize (EphemeridesManager & *manager*) [pure virtual]

Initialize the ephemerides.

Parameters

in,out	*manager*	Ephemerides manager

Implements jeod::EphemerisInterface.

Implemented in jeod::SinglePlanetEphemeris, and jeod::EmptySpaceEphemeris.

8.28.3.6 void jeod::SinglePointEphemeris::ephem_update (void) [inline], [virtual]

Update the ephemerides, which in this case is a no-op.

Implements jeod::EphemerisInterface.

Definition at line 297 of file simple_ephemerides.hh.

8.28.3.7 const char * jeod::SinglePointEphemeris::get_name (void) const [inline], [virtual]

Retrieve the identifier.

Returns

Identifier

Implements jeod::EphemerisInterface.

Definition at line 285 of file simple_ephemerides.hh.

References identifier.

8.28.3.8 virtual void jeod::SinglePointEphemeris::initialize_model (EphemeridesManager & *manager*) [pure virtual]

Register the model and its ephemeris points.

Parameters

in, out	*manager*	Ephemerides manager

Implemented in jeod::SinglePlanetEphemeris, and jeod::EmptySpaceEphemeris.

8.28.3.9 SinglePointEphemeris& jeod::SinglePointEphemeris::operator= (const SinglePointEphemeris &) [private]

Not implemented.

8.28.3.10 void jeod::SinglePointEphemeris::set_name (const char * *new_name*) [virtual]

Set the name of a SinglePointEphemeris object.

Parameters

in, out	*new_name*	Central point name

Reimplemented in jeod::SinglePlanetEphemeris, and jeod::EmptySpaceEphemeris.

Definition at line 122 of file simple_ephemerides.cc.

References identifier, and jeod::EphemeridesMessages::inconsistent_setup.

Referenced by jeod::EmptySpaceEphemeris::set_name(), and jeod::SinglePlanetEphemeris::set_name().

8.28.3.11 double jeod::SinglePointEphemeris::timestamp (void) const [inline], [virtual]

Retrieve the timestamp.

Returns

Timestamp
Units: s

Implements jeod::EphemerisInterface.

Definition at line 272 of file simple_ephemerides.hh.

References update_time.

8.28.4 Friends And Related Function Documentation

8.28.4.1 void init_attrjeod__SinglePointEphemeris ( ) [friend]

8.28.4.2 friend class InputProcessor [friend]

Definition at line 53 of file simple_ephemerides.hh.

8.28.5 Field Documentation

8.28.5.1 bool jeod::SinglePointEphemeris::active [protected]

Is the model active?

trick_units(–)

Definition at line 153 of file simple_ephemerides.hh.

Referenced by activate(), deactivate(), jeod::EmptySpaceEphemeris::ephem_build_tree(), jeod::SinglePlanetEphemeris::ephem_build_tree(), jeod::SinglePlanetEphemeris::ephem_initialize(), jeod::EmptySpaceEphemeris::initialize_model(), and jeod::SinglePlanetEphemeris::initialize_model().

8.28.5.2 char* jeod::SinglePointEphemeris::identifier [protected]

Identifier for this model.

trick_units(–)

Definition at line 143 of file simple_ephemerides.hh.

Referenced by jeod::EmptySpaceEphemeris::ephem_initialize(), jeod::SinglePlanetEphemeris::ephem_initialize(), get_name(), set_name(), and ~SinglePointEphemeris().

8.28.5.3 double jeod::SinglePointEphemeris::update_time [protected]

Time of last update, dynamic time seconds.

trick_units(s)

Definition at line 148 of file simple_ephemerides.hh.

Referenced by timestamp().

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

- simple_ephemerides.hh
- simple_ephemerides.cc

Chapter 9

File Documentation

9.1 base_ephem_manager.hh File Reference

Define the BaseEphemManager class, which defines the interfaces to the class EphemManager.

```
#include <vector>
#include "utils/ref_frames/include/base_ref_frame_manager.hh"
#include "utils/sim_interface/include/jeod_class.hh"
```

Data Structures

- class jeod::BaseEphemeridesManager

  The EphemManager class augments the RefFrameManager with ephemeris-related items.

Namespaces

- jeod

  Namespace jeod.

9.1.1 Detailed Description

Define the BaseEphemManager class, which defines the interfaces to the class EphemManager.

Definition in file base_ephem_manager.hh.

9.2 class_declarations.hh File Reference

Forward declarations of classes defined in the DE4xx model.

Namespaces

- jeod

  Namespace jeod.

9.2.1 Detailed Description

Forward declarations of classes defined in the DE4xx model.

Definition in file de4xx_ephem/include/class_declarations.hh.

9.3 class_declarations.hh File Reference

Forward declarations of classes defined in models/environment/ephemerides/ephem_interface files.

Namespaces

- jeod

  Namespace jeod.

9.3.1 Detailed Description

Forward declarations of classes defined in models/environment/ephemerides/ephem_interface files.

Definition in file ephem_interface/include/class_declarations.hh.

9.4 class_declarations.hh File Reference

Forward declarations of classes defined in models/environment/ephemerides/ephem_item files.

Namespaces

- jeod

  Namespace jeod.

9.4.1 Detailed Description

Forward declarations of classes defined in models/environment/ephemerides/ephem_item files.

Definition in file ephem_item/include/class_declarations.hh.

9.5 de405_0.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- jeod::EphemerisDataSetMeta metaData
- jeod::EphemerisDataItemMeta itemData [13]
- jeod::EphemerisDataSegmentMeta segmentData [31]
- double segment_coeffs_0 [229][1018]

9.5.1 Variable Documentation

9.5.1.1 jeod::EphemerisDataItemMeta itemData[13]

Definition at line 44 of file de405_0.cc.

Referenced by jeod::De4xxFile::interpolate(), and jeod::De4xxFile::pre_initialize().

9.5.1.2 jeod::EphemerisDataSetMeta metaData

Initial value:

```
= {
    .number_file_items = 13,
    .start_epoch = 2305424.50,
    .stop_epoch  = 2525008.50,
    .delta_epoch = 32,
    .number_segments = 31,
    .ncoeff = 1018,
    .de_constants = {
        0.40500000000000000E+03,
        0.40500000000000000E+03,
        0.14959787070691000015E+09,
        0.81300560000000044E+02,
        0.29979245799999984E+06,
        0.49125474514508187E-10,
        0.72434524861627027OE-09,
        0.89970113467124988 2E-09,
        0.95495351057792580 6E-10,
        0.28253459095242264 3E-06,
        0.84597151856806587 4E-07,
        0.12920249167819693 9E-07,
        0.15243589007842762 8E-07,
        0.21886997654259696 8E-11,
        0.29591220828559109 5E-03
}
}
```

Definition at line 17 of file de405_0.cc.

9.5.1.3 double segment_coeffs_0[229][1018]

Definition at line 275 of file de405_0.cc.

9.5.1.4 jeod::EphemerisDataSegmentMeta segmentData[31]

Definition at line 112 of file de405_0.cc.

9.6 de405_1.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_1 [229][1018]

9.6.1 Variable Documentation

9.6.1.1 double segment_coeffs_1[229][1018]

Definition at line 17 of file de405_1.cc.

9.7 de405_10.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_10 [229][1018]

9.7.1 Variable Documentation

9.7.1.1 double segment_coeffs_10[229][1018]

Definition at line 17 of file de405_10.cc.

9.8 de405_11.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_11 [230][1018]

9.8.1 Variable Documentation

9.8.1.1 double segment_coeffs_11[230][1018]

Definition at line 17 of file de405_11.cc.

9.9 de405_12.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_12 [229][1018]

9.9.1 Variable Documentation

9.9.1.1 double segment_coeffs_12[229][1018]

Definition at line 17 of file de405_12.cc.

9.10 de405_13.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_13 [229][1018]

9.10.1 Variable Documentation

9.10.1.1 double segment_coeffs_13[229][1018]

Definition at line 17 of file de405_13.cc.

9.11 de405_14.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_14 [229][1018]

9.11.1 Variable Documentation

9.11.1.1 double segment_coeffs_14[229][1018]

Definition at line 17 of file de405_14.cc.

9.12 de405_15.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_15 [230][1018]

9.12.1 Variable Documentation

9.12.1.1 double segment_coeffs_15[230][1018]

Definition at line 17 of file de405_15.cc.

9.13 de405_16.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_16 [229][1018]

9.13.1 Variable Documentation

9.13.1.1 double segment_coeffs_16[229][1018]

Definition at line 17 of file de405_16.cc.

9.14 de405_17.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_17 [229][1018]

9.14.1 Variable Documentation

9.14.1.1 double segment_coeffs_17[229][1018]

Definition at line 17 of file de405_17.cc.

9.15 de405_18.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_18 [229][1018]

9.15.1 Variable Documentation

9.15.1.1 double segment_coeffs_18[229][1018]

Definition at line 17 of file de405_18.cc.

9.16 de405_19.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_19 [230][1018]

9.16.1 Variable Documentation

9.16.1.1 double segment_coeffs_19[230][1018]

Definition at line 17 of file de405_19.cc.

9.17 de405_2.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_2 [229][1018]

9.17.1 Variable Documentation

9.17.1.1 double segment_coeffs_2[229][1018]

Definition at line 17 of file de405_2.cc.

9.18 de405_20.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_20 [229][1018]

9.18.1 Variable Documentation

9.18.1.1 double segment_coeffs_20[229][1018]

Definition at line 17 of file de405_20.cc.

9.19 de405_21.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_21 [229][1018]

9.19.1 Variable Documentation

9.19.1.1 double segment_coeffs_21[229][1018]

Definition at line 17 of file de405_21.cc.

9.20 de405_22.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_22 [230][1018]

9.20.1 Variable Documentation

9.20.1.1 double segment_coeffs_22[230][1018]

Definition at line 17 of file de405_22.cc.

9.21 de405_23.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_23 [229][1018]

9.21.1 Variable Documentation

9.21.1.1 double segment_coeffs_23[229][1018]

Definition at line 17 of file de405_23.cc.

9.22 de405_24.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_24 [229][1018]

9.22.1 Variable Documentation

9.22.1.1 double segment_coeffs_24[229][1018]

Definition at line 17 of file de405_24.cc.

9.23 de405_25.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_25 [229][1018]

9.23.1 Variable Documentation

9.23.1.1 double segment_coeffs_25[229][1018]

Definition at line 17 of file de405_25.cc.

9.24 de405_26.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_26 [230][1018]

9.24.1 Variable Documentation

9.24.1.1 double segment_coeffs_26[230][1018]

Definition at line 17 of file de405_26.cc.

9.25 de405_27.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_27 [229][1018]

9.25.1 Variable Documentation

9.25.1.1 double segment_coeffs_27[229][1018]

Definition at line 17 of file de405_27.cc.

9.26 de405_28.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_28 [229][1018]

9.26.1 Variable Documentation

9.26.1.1 double segment_coeffs_28[229][1018]

Definition at line 17 of file de405_28.cc.

9.27 de405_29.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_29 [230][1018]

9.27.1 Variable Documentation

9.27.1.1 double segment_coeffs_29[230][1018]

Definition at line 17 of file de405_29.cc.

9.28 de405_3.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_3 [229][1018]

9.28.1 Variable Documentation

9.28.1.1 double segment_coeffs_3[229][1018]

Definition at line 17 of file de405_3.cc.

9.29 de405_30.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_30 [13][1018]

9.29.1 Variable Documentation

9.29.1.1 double segment_coeffs_30[13][1018]

Definition at line 17 of file de405_30.cc.

9.30 de405_4.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_4 [230][1018]

9.30.1 Variable Documentation

9.30.1.1 double segment_coeffs_4[230][1018]

Definition at line 17 of file de405_4.cc.

9.31 de405_5.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_5 [229][1018]

9.31.1 Variable Documentation

9.31.1.1 double segment_coeffs_5[229][1018]

Definition at line 17 of file de405_5.cc.

9.32 de405_6.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_6 [229][1018]

9.32.1 Variable Documentation

9.32.1.1 double segment_coeffs_6[229][1018]

Definition at line 17 of file de405_6.cc.

9.33 de405_7.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_7 [229][1018]

9.33.1 Variable Documentation

9.33.1.1 double segment_coeffs_7[229][1018]

Definition at line 17 of file de405_7.cc.

9.34 de405_8.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_8 [230][1018]

9.34.1 Variable Documentation

9.34.1.1 double segment_coeffs_8[230][1018]

Definition at line 17 of file de405_8.cc.

9.35 de405_9.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_9 [229][1018]

9.35.1 Variable Documentation

9.35.1.1 double segment_coeffs_9[229][1018]

Definition at line 17 of file de405_9.cc.

9.36 de421_0.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- jeod::EphemerisDataSetMeta metaData
- jeod::EphemerisDataItemMeta itemData [13]
- jeod::EphemerisDataSegmentMeta segmentData [2]
- double segment_coeffs_0 [1713][1018]

9.36.1 Variable Documentation

9.36.1.1 jeod::EphemerisDataItemMeta itemData[13]

Definition at line 44 of file de421_0.cc.

9.36.1.2 jeod::EphemerisDataSetMeta metaData

Initial value:

```
= {
    .number_file_items = 13,
    .start_epoch = 2414992.50,
    .stop_epoch  = 2524624.50,
    .delta_epoch = 32,
    .number_segments = 2,
    .ncoeff = 1018,
    .de_constants = {
        0.42100000000000000E+03,
        0.42100000000000000E+03,
        0.14959787069962620E+09,
        0.81300569069915300E+02,
        0.29979245800000000E+06,
        0.49125495718679400E-10,
        0.72434523326984410E-09,
        0.89970114082680490E-09,
        0.95495486956223900E-10,
        0.28253458408550500E-06,
        0.84597060733084780E-07,
        0.12920248257926500E-07,
        0.15243591092497400E-07,
        0.21784410519905200E-11,
        0.29591220828559110E-03
}
}
```

Definition at line 17 of file de421_0.cc.

9.36.1.3 double segment_coeffs_0[1713][1018]

Definition at line 130 of file de421_0.cc.

9.36.1.4 jeod::EphemerisDataSegmentMeta segmentData[2]

Initial value:

```
= {
        {
        .num_recs = 1713,
        .start_epoch = 2.41499250000000000E+06,
        .stop_epoch = 2.46980850000000000E+06
    },
    {
        .num_recs = 1714,
        .start_epoch = 0.24697765000000000E+07,
        .stop_epoch = 0.25246245000000000E+07
    }
}
```

Definition at line 112 of file de421_0.cc.

9.37 de421_1.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_1 [1714][1018]

9.37.1 Variable Documentation

9.37.1.1 double segment_coeffs_1[1714][1018]

Definition at line 17 of file de421_1.cc.

9.38 de440_0.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- jeod::EphemerisDataSetMeta metaData
- jeod::EphemerisDataItemMeta itemData [15]
- jeod::EphemerisDataSegmentMeta segmentData [11]
- double segment_coeffs_0 [1142][1018]

9.38.1 Variable Documentation

9.38.1.1 jeod::EphemerisDataItemMeta itemData[15]

Definition at line 44 of file de440_0.cc.

9.38.1.2 jeod::EphemerisDataSetMeta metaData

Initial value:

```
= {
    .number_file_items = 15,
    .start_epoch = 2287184.50,
    .stop_epoch  = 2688976.50,
    .delta_epoch = 32,
    .number_segments = 11,
    .ncoeff = 1018,
    .de_constants = {
        0.44000000000000000E+03,
        0.44000000000000000E+03,
        0.14959787070999999988E+09,
        0.81300568221497215 4E+02,
        0.29979245799999984E+06,
        0.49125001948893181 8E-10,
        0.72434523326441186 9E-09,
        0.89970113929473466 0E-09,
        0.95495488297258118 9E-10,
        0.28253458252257917 5E-06,
        0.84597059933762902 7E-07,
        0.12920265649682399 4E-07,
        0.15243573478851938 6E-07,
        0.21750964648933581 1E-11,
        0.29591220828411956 1E-03
}
}
```

Definition at line 17 of file de440_0.cc.

9.38.1.3 double segment_coeffs_0[1142][1018]

Definition at line 185 of file de440_0.cc.

9.38.1.4 jeod::EphemerisDataSegmentMeta segmentData[11]

Definition at line 122 of file de440_0.cc.

9.39 de440_1.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_1 [1142][1018]

9.39.1 Variable Documentation

9.39.1.1 double segment_coeffs_1[1142][1018]

Definition at line 17 of file de440_1.cc.

9.40 de440_10.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_10 [1143][1018]

9.40.1 Variable Documentation

9.40.1.1 double segment_coeffs_10[1143][1018]

Definition at line 17 of file de440_10.cc.

9.41 de440_2.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_2 [1143][1018]

9.41.1 Variable Documentation

9.41.1.1 double segment_coeffs_2[1143][1018]

Definition at line 17 of file de440_2.cc.

9.42 de440_3.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_3 [1142][1018]

9.42.1 Variable Documentation

9.42.1.1 double segment_coeffs_3[1142][1018]

Definition at line 17 of file de440_3.cc.

9.43 de440_4.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_4 [1142][1018]

9.43.1 Variable Documentation

9.43.1.1 double segment_coeffs_4[1142][1018]

Definition at line 17 of file de440_4.cc.

9.44 de440_5.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_5 [1143][1018]

9.44.1 Variable Documentation

9.44.1.1 double segment_coeffs_5[1143][1018]

Definition at line 17 of file de440_5.cc.

9.45 de440_6.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_6 [1142][1018]

9.45.1 Variable Documentation

9.45.1.1 double segment_coeffs_6[1142][1018]

Definition at line 17 of file de440_6.cc.

9.46 de440_7.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_7 [1143][1018]

9.46.1 Variable Documentation

9.46.1.1 double segment_coeffs_7[1143][1018]

Definition at line 17 of file de440_7.cc.

9.47 de440_8.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_8 [1142][1018]

9.47.1 Variable Documentation

9.47.1.1 double segment_coeffs_8[1142][1018]

Definition at line 17 of file de440_8.cc.

9.48 de440_9.cc File Reference

```
#include "environment/ephemerides/de4xx_ephem/include/de4xx_file.hh"
```

Variables

- double segment_coeffs_9 [1142][1018]

9.48.1 Variable Documentation

9.48.1.1 double segment_coeffs_9[1142][1018]

Definition at line 17 of file de440_9.cc.

9.49 de4xx_base.hh File Reference

Define data types for JPL ephemeris model.

```
#include <cstdint>
#include "utils/sim_interface/include/jeod_class.hh"
```

Namespaces

- jeod

  Namespace jeod.
- jeod::De4xxBase

  Defines enumerations used in the DE4xx ephemeris model.

Enumerations

- enum jeod::De4xxBase::De4xxFileEntries {
  jeod::De4xxBase::De4xx_File_Mercury = 0, jeod::De4xxBase::De4xx_File_Venus = 1, jeod::De4xxBase::-
  De4xx_File_EMbary = 2, jeod::De4xxBase::De4xx_File_Mars = 3,
  jeod::De4xxBase::De4xx_File_Jupiter = 4, jeod::De4xxBase::De4xx_File_Saturn = 5, jeod::De4xxBase::-
  De4xx_File_Uranus = 6, jeod::De4xxBase::De4xx_File_Neptune = 7,
  jeod::De4xxBase::De4xx_File_Pluto = 8, jeod::De4xxBase::De4xx_File_Moon = 9, jeod::De4xxBase::De4xx-
  _File_Sun = 10, jeod::De4xxBase::De4xx_File_ENutation = 11,
  jeod::De4xxBase::De4xx_File_LLibration = 12, jeod::De4xxBase::De4xx_File_LAngVel = 13, jeod::De4xx-
  Base::De4xx_File_tt_tdb = 14, jeod::De4xxBase::De4xx_File_MaxEntries }

  Defines names for planetary body descriptors in the ephemeris file.
- enum jeod::De4xxBase::De4xxEphemConsts {
  jeod::De4xxBase::De4xx_Const_DENUM = 0, jeod::De4xxBase::De4xx_Const_LENUM, jeod::De4xxBase::-
  De4xx_Const_AU, jeod::De4xxBase::De4xx_Const_EMRAT,
  jeod::De4xxBase::De4xx_Const_CLIGHT, jeod::De4xxBase::De4xx_Const_GM1, jeod::De4xxBase::De4xx-
  _Const_GM2, jeod::De4xxBase::De4xx_Const_GMB,
  jeod::De4xxBase::De4xx_Const_GM4, jeod::De4xxBase::De4xx_Const_GM5, jeod::De4xxBase::De4xx_-
  Const_GM6, jeod::De4xxBase::De4xx_Const_GM7,
  jeod::De4xxBase::De4xx_Const_GM8, jeod::De4xxBase::De4xx_Const_GM9, jeod::De4xxBase::De4xx_-
  Const_GMS, jeod::De4xxBase::De4xx_Const_MaxConsts }

  Index aliases for the constants listed in the DE header that are used by JEOD.

- enum jeod::De4xxBase::De4xxEphemBodies {
  jeod::De4xxBase::De4xx_Ephem_Sun = 0, jeod::De4xxBase::De4xx_Ephem_Mercury = 1, jeod::De4xxBase::De4xx_Ephem_Venus = 2, jeod::De4xxBase::De4xx_Ephem_Earth = 3,
  jeod::De4xxBase::De4xx_Ephem_Mars = 4, jeod::De4xxBase::De4xx_Ephem_Jupiter = 5, jeod::De4xxBase::De4xx_Ephem_Saturn = 6, jeod::De4xxBase::De4xx_Ephem_Uranus = 7,
  jeod::De4xxBase::De4xx_Ephem_Neptune = 8, jeod::De4xxBase::De4xx_Ephem_Pluto = 9, jeod::De4xxBase::De4xx_Ephem_Moon = 10, jeod::De4xxBase::De4xx_Ephem_EMbary = 11,
  jeod::De4xxBase::De4xx_Ephem_SSbary = 12, jeod::De4xxBase::De4xx_Ephem_EML1 = 13, jeod::De4xxBase::De4xx_Ephem_ENutation = 14, jeod::De4xxBase::De4xx_Ephem_LLibration = 15,
  jeod::De4xxBase::De4xx_Ephem_MaxBodies }

  Defines names for ephemeris items as represented in the JEOD DE4xx model.

Functions

- static const char *point_names[32] jeod::De4xxBase::__attribute__ ((unused))
- static uint32_t jeod::De4xxBase::number_jeod_items (int de_version_num)

  Total number of items in the JEOD ephemeris.
- static uint32_t jeod::De4xxBase::number_trans_points (int de_version_num)

  Total number of translational states in the JEOD ephemeris.
- static uint32_t jeod::De4xxBase::number_grav_models (int de_version_num)

  Number of gravity models in the JEOD ephemeris (Mercury to Sun + implied Earth) Currently only one possibility, but written for extensibility.
- static uint32_t jeod::De4xxBase::number_physical_bodies (int de_version_num)

  Number of bodies in the JEOD ephemeris (Planets + Pluto + Moon + Sun) Currently only one possibility, but written for extensibility.

9.49.1 Detailed Description

Define data types for JPL ephemeris model.

Definition in file de4xx_base.hh.

9.50 de4xx_ephem.cc File Reference

Define the methods of the classes defined in de4xx_ephem.hh.

```
#include <cstddef>
#include <cstdio>
#include <climits>
#include "environment/ephemerides/ephem_interface/include/ephem_messages.-
hh"
#include "environment/ephemerides/ephem_manager/include/ephem_manager.hh"
#include "environment/planet/include/base_planet.hh"
#include "environment/time/include/time_manager.hh"
#include "environment/time/include/time_tt.hh"
#include "environment/time/include/time_dyn.hh"
#include "utils/named_item/include/named_item.hh"
#include "utils/math/include/vector3.hh"
#include "utils/memory/include/jeod_alloc.hh"
#include "utils/message/include/message_handler.hh"
#include "../include/de4xx_ephem.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.50.1 Detailed Description

Define the methods of the classes defined in de4xx_ephem.hh.

Definition in file de4xx_ephem.cc.

9.51 de4xx_ephem.hh File Reference

Define class for the De4xx ephemeris model.

```
#include <string>
#include "environment/ephemerides/ephem_interface/include/ephem_interface.-
hh"
#include "environment/ephemerides/ephem_interface/include/ephem_ref_frame.-
hh"
#include "environment/ephemerides/ephem_item/include/ephem_item.hh"
#include "environment/ephemerides/ephem_item/include/ephem_orient_zxz.hh"
#include "environment/ephemerides/ephem_item/include/ephem_point.hh"
#include "environment/time/include/class_declarations.hh"
#include "utils/ref_frames/include/ref_frame_interface.hh"
#include "utils/sim_interface/include/jeod_class.hh"
#include "class_declarations.hh"
#include "de4xx_base.hh"
#include "de4xx_file.hh"
```

Data Structures

- class jeod::De4xxEphemItem

  Describes a point modeled in a DE4xx ephemeris file.
- class jeod::De4xxEphemeris

  The S_define-level class that provides planetary ephemerides.

Namespaces

- jeod

  Namespace jeod.

9.51.1 Detailed Description

Define class for the De4xx ephemeris model.

Definition in file de4xx_ephem.hh.

9.52 de4xx_ephem_dynmanager.cc File Reference

Wall off dependencies on the dynamics manager.

```
#include "dynamics/dyn_manager/include/dyn_manager.hh"
#include "environment/ephemerides/ephem_manager/include/ephem_manager.hh"
#include "environment/time/include/time_manager.hh"
#include "utils/message/include/message_handler.hh"
#include "../include/de4xx_ephem.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.52.1 Detailed Description

Wall off dependencies on the dynamics manager.

Definition in file de4xx_ephem_dynmanager.cc.

9.53 de4xx_file.cc File Reference

This file defines several utility functions used to read a binary JPL DE405 ephemeris file.

```
#include <cerrno>
#include <cstddef>
#include <cstdlib>
#include <cstdio>
#include <cstring>
#include <limits>
#include <unistd.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <dlfcn.h>
#include <ios>
#include <iostream>
#include <fstream>
#include <string>
#include "environment/ephemerides/ephem_interface/include/ephem_messages.-
hh"
#include "utils/memory/include/jeod_alloc.hh"
#include "utils/message/include/message_handler.hh"
#include "../include/de4xx_file.hh"
```

Namespaces

- jeod

  Namespace jeod.

Macros

- #define __STDC_LIMIT_MACROS

Functions

- void jeod::process_mem_usage (double &vm_usage, double &resident_set)

9.53.1 Detailed Description

This file defines several utility functions used to read a binary JPL DE405 ephemeris file. The functions are

open - Open an ephemeris file for input close - Close a previously open ephemeris file read_record - Read a record from the ephemeris file get_string - Get a string from the current data record get_int - Get integer array from the current data record get_double - Get double array from the current data record

NOTA BENE – The functions defined in this file are intended for use by the top-level ephemeris functions only.

Definition in file de4xx_file.cc.

9.54 de4xx_file.hh File Reference

Define the class responsible for reading the DE4xx ephemeris file.

```
#include <cstdio>
#include <cstddef>
#include <stdint.h>
#include "utils/container/include/simple_checkpointable.hh"
#include "utils/sim_interface/include/config.hh"
#include "utils/sim_interface/include/jeod_class.hh"
#include "de4xx_base.hh"
```

Data Structures

- struct jeod::EphemerisDataSetMeta

  Container for the metadata from the DE model header.
- struct jeod::EphemerisDataItemMeta

  Structure containing the header metadata for sizing/locating the data entries with the data segments.
- struct jeod::EphemerisDataSegmentMeta

  Metadata implied from each data segment.
- class jeod::De4xxFileSpec

  Specifies which file to use (user input initialization-time data).
- class jeod::De4xxFileIO

  Contains data used directly for reading the ephemeris file.
- class jeod::De4xxFileHeader

  Contains data extracted from the ephemeris file header.
- class jeod::De4xxFileItem

  Contains data regarding one of the items in a DE ephemeris file.
- class jeod::De4xxFileRefTime

  Contains timing reference data.
- class jeod::De4xxFileCoef

  Contains Chebychev polynomial coefficients and terms.
- class jeod::De4xxFileRestart

  The FILE pointer in a De4xxFileIO cannot be restored by Trick.
- class jeod::De4xxFile

  Provides the ability to read and interpret a DE4xx ephemeris file.

Namespaces

- jeod

  Namespace jeod.

9.54.1 Detailed Description

Define the class responsible for reading the DE4xx ephemeris file.

Definition in file de4xx_file.hh.

9.55 de4xx_file_init.cc File Reference

Define De4xx initialization methods.

```
#include <cerrno>
#include <climits>
#include <cmath>
#include <cstddef>
#include <cstring>
#include <sys/types.h>
#include <sys/stat.h>
#include <unistd.h>
#include <dlfcn.h>
#include "environment/ephemerides/ephem_interface/include/ephem_messages.-
hh"
#include "utils/memory/include/jeod_alloc.hh"
#include "utils/message/include/message_handler.hh"
#include "../include/de4xx_file.hh"
```

Namespaces

- jeod

  Namespace jeod.

Functions

- static double jeod::l1_point (double b1b2_mass_ratio)

  Calculate the location of the L1 point as a ratio.

9.55.1 Detailed Description

Define De4xx initialization methods.

Definition in file de4xx_file_init.cc.

9.56 de4xx_file_update.cc File Reference

Define De4xxFile::update.

```
#include <cstddef>
#include <limits>
#include <cstdint>
#include <dlfcn.h>
#include "environment/ephemerides/ephem_interface/include/ephem_messages.-
hh"
#include "utils/message/include/message_handler.hh"
#include "../include/de4xx_file.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.56.1 Detailed Description

Define De4xxFile::update.

Definition in file de4xx_file_update.cc.

9.57 ephem_interface.hh File Reference

Define base class for all ephemeris interface models.

```
#include "utils/ref_frames/include/subscription.hh"
#include "utils/sim_interface/include/jeod_class.hh"
```

Data Structures

- class jeod::EphemerisInterface

  Interface class that specifies minimal functionality of an ephemeris model.

Namespaces

- jeod

  Namespace jeod.

9.57.1 Detailed Description

Define base class for all ephemeris interface models.

Definition in file ephem_interface.hh.

9.58 ephem_item.cc File Reference

Define member functions for the EphemItem class and subclasses.

```
#include <cstddef>
#include "environment/ephemerides/ephem_interface/include/ephem_interface.-
hh"
#include "environment/ephemerides/ephem_interface/include/ephem_messages.-
hh"
#include "environment/ephemerides/ephem_manager/include/base_ephem_manager.-
hh"
#include "environment/planet/include/base_planet.hh"
#include "utils/math/include/vector3.hh"
#include "utils/memory/include/jeod_alloc.hh"
#include "utils/message/include/message_handler.hh"
#include "utils/named_item/include/named_item.hh"
#include "../include/ephem_item.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.58.1 Detailed Description

Define member functions for the EphemItem class and subclasses.

Definition in file ephem_item.cc.

9.59 ephem_item.hh File Reference

Define classes for items represented in some ephemeris model.

```
#include "environment/ephemerides/ephem_interface/include/ephem_interface.-
hh"
#include "environment/ephemerides/ephem_interface/include/ephem_ref_frame.-
hh"
#include "utils/sim_interface/include/jeod_class.hh"
#include "class_declarations.hh"
#include "ephem_item_inline.hh"
```

Data Structures

- class jeod::EphemerisItem

  The EphemerisItem class is the base class for representing an item that is modeled in an ephemeris model.

Namespaces

- jeod

  Namespace jeod.

9.59.1 Detailed Description

Define classes for items represented in some ephemeris model.

Definition in file ephem_item.hh.

9.60 ephem_item_inline.hh File Reference

Define inline methods for the EphemerisItem class.

```
#include "ephem_item.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.60.1 Detailed Description

Define inline methods for the EphemerisItem class.

Definition in file ephem_item_inline.hh.

9.61 ephem_manager.cc File Reference

Define EphemeridesManager methods.

```
#include <algorithm>
#include <cstddef>
#include "environment/ephemerides/ephem_interface/include/ephem_messages.-
hh"
#include "environment/ephemerides/ephem_interface/include/ephem_ref_frame.-
hh"
#include "environment/ephemerides/ephem_item/include/ephem_item.hh"
#include "environment/ephemerides/ephem_item/include/ephem_orient.hh"
#include "environment/ephemerides/ephem_item/include/ephem_point.hh"
#include "environment/planet/include/base_planet.hh"
#include "utils/message/include/message_handler.hh"
#include "utils/memory/include/jeod_alloc.hh"
#include "utils/ref_frames/include/ref_frame.hh"
#include "../include/ephem_manager.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.61.1 Detailed Description

Define EphemeridesManager methods.

Definition in file ephem_manager.cc.

9.62 ephem_manager.hh File Reference

Define the EphemManager class, which manages the ephemeris models in a JEOD-based simulation.

```
#include "utils/ref_frames/include/ref_frame_manager.hh"
#include "utils/container/include/pointer_vector.hh"
#include "utils/sim_interface/include/jeod_class.hh"
#include "base_ephem_manager.hh"
```

Data Structures

- class jeod::EphemeridesManager

  The EphemeridesManager class manages the ephemeris models in a simulation.

Namespaces

- jeod

  Namespace jeod.

9.62.1 Detailed Description

Define the EphemManager class, which manages the ephemeris models in a JEOD-based simulation.

Definition in file ephem_manager.hh.

9.63 ephem_messages.cc File Reference

Implement the class EphemeridesMessages.

```
#include "utils/message/include/make_message_code.hh"
#include "../include/ephem_messages.hh"
```

Namespaces

- jeod

  Namespace jeod.

Macros

- #define MAKE_EPHEMERIDES_MESSAGE_CODE(id) JEOD_MAKE_MESSAGE_CODE(EphemeridesMessages, "environment/ephemerides/", id)

9.63.1 Detailed Description

Implement the class EphemeridesMessages.

Definition in file ephem_messages.cc.

9.63.2 Macro Definition Documentation

9.63.2.1 #define MAKE_EPHEMERIDES_MESSAGE_CODE(*id*) JEOD_MAKE_MESSAGE_CODE(EphemeridesMessages, "environment/ephemerides/", id)

Definition at line 44 of file ephem_messages.cc.

9.64 ephem_messages.hh File Reference

Define the class EphemeridesMessages, the class that specifies the message IDs used in the JEOD ephemerides model.

```
#include "utils/sim_interface/include/jeod_class.hh"
```

Data Structures

- class jeod::EphemeridesMessages

  Specifies the message IDs used in the Ephemerides model.

Namespaces

- jeod

  Namespace jeod.

9.64.1 Detailed Description

Define the class EphemeridesMessages, the class that specifies the message IDs used in the JEOD ephemerides model.

Definition in file ephem_messages.hh.

9.65 ephem_orient.cc File Reference

Define member functions for the EphemItem class and subclasses.

```
#include <cstddef>
#include "environment/ephemerides/ephem_interface/include/ephem_interface.-
hh"
#include "environment/ephemerides/ephem_interface/include/ephem_messages.-
hh"
#include "environment/ephemerides/ephem_manager/include/ephem_manager.hh"
#include "environment/planet/include/base_planet.hh"
#include "utils/math/include/vector3.hh"
#include "utils/memory/include/jeod_alloc.hh"
#include "utils/message/include/message_handler.hh"
#include "utils/named_item/include/named_item.hh"
#include "../include/ephem_orient.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.65.1 Detailed Description

Define member functions for the EphemItem class and subclasses.

Definition in file ephem_orient.cc.

9.66 ephem_orient.hh File Reference

Define class EphemerisOrientation.

```
#include "environment/ephemerides/ephem_interface/include/ephem_interface.-
hh"
#include "environment/ephemerides/ephem_interface/include/ephem_ref_frame.-
hh"
#include "utils/sim_interface/include/jeod_class.hh"
#include "class_declarations.hh"
#include "ephem_item.hh"
```

Data Structures

- class jeod::EphemerisOrientation

  An EphemerisOrientation object updates the rotational state of an ephemeris reference frame.

Namespaces

- jeod

  Namespace jeod.

9.66.1 Detailed Description

Define class EphemerisOrientation.

Definition in file ephem_orient.hh.

9.67 ephem_orient_zxz.cc File Reference

Define member functions for the EphemItem class and subclasses.

```
#include <cmath>
#include <cstddef>
#include "environment/ephemerides/ephem_interface/include/ephem_interface.-
hh"
#include "environment/ephemerides/ephem_interface/include/ephem_messages.-
hh"
#include "environment/ephemerides/ephem_manager/include/ephem_manager.hh"
#include "environment/planet/include/base_planet.hh"
#include "utils/math/include/vector3.hh"
#include "utils/memory/include/jeod_alloc.hh"
#include "utils/message/include/message_handler.hh"
#include "utils/named_item/include/named_item.hh"
#include "utils/quaternion/include/quat.hh"
#include "../include/ephem_orient_zxz.hh"
```

Namespaces

- jeod

  Namespace jeod.

Macros

- #define EPSILON_TIME 1e-12
- #define TAYLOR_CUTOFF 0.00786

9.67.1 Detailed Description

Define member functions for the EphemItem class and subclasses.

Definition in file ephem_orient_zxz.cc.

9.68 ephem_orient_zxz.hh File Reference

Define classes for items represented in some ephemeris model.

```
#include "environment/ephemerides/ephem_interface/include/ephem_interface.-
hh"
#include "environment/ephemerides/ephem_interface/include/ephem_ref_frame.-
hh"
#include "utils/sim_interface/include/jeod_class.hh"
#include "class_declarations.hh"
#include "ephem_orient.hh"
```

Data Structures

- class jeod::EphemerisZXZOrientation

  The EphemerisZXZOrientation is an EphemerisOrientation subclass that updates orientation based on an Z-X-Z Euler sequence and the time derivatives of this sequence.

Namespaces

- jeod

  Namespace jeod.

9.68.1 Detailed Description

Define classes for items represented in some ephemeris model.

Definition in file ephem_orient_zxz.hh.

9.69 ephem_point.cc File Reference

Define member functions for the EphemPoint class.

```
#include <cstddef>
#include "environment/ephemerides/ephem_interface/include/ephem_interface.-
hh"
#include "environment/ephemerides/ephem_interface/include/ephem_messages.-
```

```
hh"
#include "environment/ephemerides/ephem_manager/include/ephem_manager.hh"
#include "environment/planet/include/base_planet.hh"
#include "utils/math/include/vector3.hh"
#include "utils/memory/include/jeod_alloc.hh"
#include "utils/message/include/message_handler.hh"
#include "utils/named_item/include/named_item.hh"
#include "../include/ephem_point.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.69.1 Detailed Description

Define member functions for the EphemPoint class.

Definition in file ephem_point.cc.

9.70 ephem_point.hh File Reference

Define class EphemerisPoint.

```
#include "utils/sim_interface/include/jeod_class.hh"
#include "class_declarations.hh"
#include "ephem_item.hh"
```

Data Structures

- class jeod::EphemerisPoint

  An EphemerisPoint object updates the translational state of an ephemeris reference frame.

Namespaces

- jeod

  Namespace jeod.

9.70.1 Detailed Description

Define class EphemerisPoint.

Definition in file ephem_point.hh.

9.71 ephem_ref_frame.cc File Reference

Define non-inlined member functions for the EphemRefFrame class.

```
#include <cstddef>
#include "environment/ephemerides/ephem_manager/include/base_ephem_manager.-
hh"
#include "utils/message/include/message_handler.hh"
#include "../include/ephem_interface.hh"
#include "../include/ephem_messages.hh"
#include "../include/ephem_ref_frame.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.71.1 Detailed Description

Define non-inlined member functions for the EphemRefFrame class.

Definition in file ephem_ref_frame.cc.

9.72 ephem_ref_frame.hh File Reference

Define the class EphemerisRefFrame.

```
#include <cstddef>
#include "utils/ref_frames/include/ref_frame.hh"
#include "utils/ref_frames/include/ref_frame_interface.hh"
#include "utils/sim_interface/include/jeod_class.hh"
```

Data Structures

- class jeod::EphemerisRefFrame

  An EphemerisRefFrame is a RefFrame whose state is set by an ephemeris model.

Namespaces

- jeod

  Namespace jeod.

9.72.1 Detailed Description

Define the class EphemerisRefFrame.

Definition in file ephem_ref_frame.hh.

9.73 find_planet.cc File Reference

Define EphemeridesManager::find_planet.

```
#include <cstddef>
#include "environment/ephemerides/ephem_interface/include/ephem_messages.-
hh"
#include "environment/planet/include/base_planet.hh"
#include "environment/planet/include/planet.hh"
#include "utils/message/include/message_handler.hh"
#include "../include/ephem_manager.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.73.1 Detailed Description

Define EphemeridesManager::find_planet. This method is isolated from the other EphemeridesManager methods because the object file drags in a whole lot of stuff.

Definition in file find_planet.cc.

9.74 propagated_planet.cc File Reference

Define the methods of the classes defined in propagated_planet.hh.

```
#include <cstddef>
#include <cstdio>
#include <climits>
#include "dynamics/dyn_manager/include/dyn_manager.hh"
#include "environment/ephemerides/ephem_interface/include/ephem_messages.-
hh"
#include "environment/ephemerides/ephem_manager/include/ephem_manager.hh"
#include "environment/planet/include/base_planet.hh"
#include "environment/time/include/time_manager.hh"
#include "environment/time/include/time_tt.hh"
#include "environment/time/include/time_dyn.hh"
#include "utils/named_item/include/named_item.hh"
#include "utils/math/include/vector3.hh"
#include "utils/memory/include/jeod_alloc.hh"
#include "utils/message/include/message_handler.hh"
#include "../include/propagated_planet.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.74.1 Detailed Description

Define the methods of the classes defined in propagated_planet.hh.

Definition in file propagated_planet.cc.

9.75 propagated_planet.hh File Reference

Define the classes needed to propagate a planet.

```
#include "dynamics/dyn_body/include/dyn_body.hh"
#include "environment/ephemerides/ephem_interface/include/ephem_interface.-
hh"
#include "environment/ephemerides/ephem_item/include/ephem_item.hh"
#include "environment/ephemerides/ephem_item/include/ephem_orient.hh"
#include "environment/ephemerides/ephem_item/include/ephem_point.hh"
#include "utils/sim_interface/include/jeod_class.hh"
```

Data Structures

- class jeod::PropagatedEphemerisPlanet

  A PropagatedEphemerisPlanet is an EphemerisPoint whose state is coupled with the translational state of a DynBody reference frame.
- class jeod::PropagatedEphemerisOrientation

  A PropagatedEphemerisOrientation is an EphemerisOrientation whose state is coupled with the rotational state of a DynBody reference frame.
- class jeod::PropagatedPlanet

  The PropagatedPlanet ephemeris model provides planetary state via a DynBody object whose state is propagated using the JEOD state integration techniques.

Namespaces

- jeod

  Namespace jeod.

9.75.1 Detailed Description

Define the classes needed to propagate a planet.

Definition in file propagated_planet.hh.

9.76 simple_ephemerides.cc File Reference

Define member functions for the SinglePointEphemeris class and subclasses.

```
#include <cstddef>
#include "environment/ephemerides/ephem_manager/include/ephem_manager.hh"
#include "utils/memory/include/jeod_alloc.hh"
#include "utils/message/include/message_handler.hh"
#include "../include/ephem_messages.hh"
#include "../include/simple_ephemerides.hh"
```

Namespaces

- jeod

  Namespace jeod.

9.76.1 Detailed Description

Define member functions for the SinglePointEphemeris class and subclasses.

Definition in file simple_ephemerides.cc.

9.77 simple_ephemerides.hh File Reference

Define classes that define simple ephemeris models.

```
#include "environment/ephemerides/ephem_item/include/ephem_point.hh"
#include "utils/sim_interface/include/jeod_class.hh"
#include "ephem_interface.hh"
#include "ephem_ref_frame.hh"
```

Data Structures

- class jeod::SinglePointEphemeris

  A SinglePointEphemeris has one ephemeris point.
- class jeod::EmptySpaceEphemeris

  Empty space has one ephemeris point.
- class jeod::SinglePlanetEphemeris

  A space with one gravitation body has one ephemeris point.

Namespaces

- jeod

  Namespace jeod.

9.77.1 Detailed Description

Define classes that define simple ephemeris models.

Definition in file simple_ephemerides.hh.

Index