# OrbitalElementsModel

5.3

Generated by Doxygen 1.8.14

# 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

# Data Structure Index

## 3.1 Data Structures

Here are the data structures with brief descriptions:

# Chapter 4

# File Index

## 4.1 File List

Here is a list of all files with brief descriptions:

# Chapter 5

# Module Documentation

## 5.1 Models

### Modules

- Utils

#### 5.1.1 Detailed Description

## 5.2 Utils

### Modules

- OrbitalElements

#### 5.2.1 Detailed Description

## 5.3 OrbitalElements

### Files

- file orbital_elements.hh

  *Orbital elements class definition.*
- file orbital_elements_messages.hh

  *Define the class OrbitalElementsMessages, the class that specifies the message IDs used in the orbital elements model.*
- file orbital_elements.cc

  *Define methods for the OrbitalElements class.*
- file orbital_elements_messages.cc

  *Implement the class OrbitalElementsMessages.*

### Namespaces

- jeod

  *Namespace jeod.*

### 5.3.1 Detailed Description

# Chapter 6

# Namespace Documentation

## 6.1 jeod Namespace Reference

Namespace jeod.

### Data Structures

- class OrbitalElements

  *Represents state in terms of Keplerian orbital elements.*
- class OrbitalElementsMessages

  *Specifies the message IDs used in the orbital elements model.*

### 6.1.1 Detailed Description

Namespace jeod.

# Chapter 7

# Data Structure Documentation

## 7.1 jeod::OrbitalElements Class Reference

Represents state in terms of Keplerian orbital elements.

`#include <orbital_elements.hh>`

### Public Member Functions

- OrbitalElements ()=default
- virtual ~OrbitalElements ()=default
- OrbitalElements (const OrbitalElements &)=delete
- OrbitalElements & operator= (const OrbitalElements &)=delete
- const std::string & get_object_name () const

  *Return the object name.*
- const std::string & get_planet_name () const

  *Return the planet name.*
- void set_object_name (const std::string &name)

  *Set the object name.*
- void set_planet_name (const std::string &name)

  *Set the planet name.*
- int from_cartesian (double mu, const double pos[3], const double vel[3])
- int to_cartesian (double mu, double pos[3], double vel[3])
- int nu_to_anomalies ()
- int mean_anom_to_nu ()

### Data Fields

- double semi_major_axis {}

  *Semi-major-axis (a)*
- double semiparam {}

  *Semiparameter (p)*
- double e_mag {}

  *Magnitude of eccentricity (e)*
- double inclination {}

*Orbit inclination (i)*

- double arg_periapsis {}

  *Argument of periapsis (w)*
- double long_asc_node {}

  *Longitude of ascending node (Omega)*
- double r_mag {}

  *Magnitude of orbital radius.*
- double vel_mag {}

  *Magnitude of orbital velocity.*
- double true_anom {}

  *True Anomaly (v)*
- double mean_anom {}

  *Mean Anomaly (M)*
- double mean_motion {}

  *Mean motion of orbit (n)*
- double orbital_anom {}

  *Eccentric (E), Hyperbolic (H), or Parabolic (B) anomaly.*
- double sin_v {}

  *Sine of the true anomaly.*
- double cos_v {1.0}

  *Cosine of the true anomaly.*
- double orb_energy {}

  *Specific orbital energy.*
- double orb_ang_momentum {}

  *Specific orbital angular momentum.*

### Protected Member Functions

- int KepEqtnE (double M, double e, double ∗E)
- int KepEqtnH (double M, double e, double ∗H)
- int KepEqtnB (double M, double ∗B)

### Protected Attributes

- std::string object_name

  *Name of orbital object.*
- std::string planet_name

  *Name of planet about which the object orbits.*

### Friends

- class InputProcessor
- void init_attrjeod__OrbitalElements ()

#### 7.1.1 Detailed Description

Represents state in terms of Keplerian orbital elements.

Definition at line 81 of file orbital_elements.hh.

### 7.1.2 Constructor & Destructor Documentation

#### 7.1.2.1 OrbitalElements() [1/2]

```
jeod::OrbitalElements::OrbitalElements ( )  [default]
```

#### 7.1.2.2 ~OrbitalElements()

```
virtual jeod::OrbitalElements::~OrbitalElements ( )  [virtual], [default]
```

#### 7.1.2.3 OrbitalElements() [2/2]

```
jeod::OrbitalElements::OrbitalElements (
            const OrbitalElements &  )  [delete]
```

### 7.1.3 Member Function Documentation

#### 7.1.3.1 from_cartesian()

```
int jeod::OrbitalElements::from_cartesian (
            double mu,
            const double pos[3],
            const double vel[3] )
```

Definition at line 120 of file orbital_elements.cc.

References arg_periapsis, e_mag, inclination, long_asc_node, mean_motion, nu_to_anomalies(), orb_ang_momentum, orb_energy, r_mag, semi_major_axis, semiparam, true_anom, and vel_mag.

#### 7.1.3.2 get_object_name()

```
const std::string & jeod::OrbitalElements::get_object_name ( ) const
```

Return the object name.

**Returns**

Const pointer to name

Definition at line 85 of file orbital_elements.cc.

References object_name.

#### 7.1.3.3 get_planet_name()

```
const std::string & jeod::OrbitalElements::get_planet_name ( ) const
```

Return the planet name.

**Returns**

Const pointer to name

Definition at line 94 of file orbital_elements.cc.

References planet_name.

#### 7.1.3.4 KepEqtnB()

```
int jeod::OrbitalElements::KepEqtnB (
             double M,
             double * B )  [protected]
```

Definition at line 874 of file orbital_elements.cc.

Referenced by mean_anom_to_nu().

#### 7.1.3.5 KepEqtnE()

```
int jeod::OrbitalElements::KepEqtnE (
             double M,
             double e,
             double * E )  [protected]
```

Definition at line 747 of file orbital_elements.cc.

Referenced by mean_anom_to_nu().

#### 7.1.3.6 KepEqtnH()

```
int jeod::OrbitalElements::KepEqtnH (
             double M,
             double e,
             double * H )  [protected]
```

Definition at line 802 of file orbital_elements.cc.

Referenced by mean_anom_to_nu().

#### 7.1.3.7 mean_anom_to_nu()

```
int jeod::OrbitalElements::mean_anom_to_nu ( )
```

Definition at line 616 of file orbital_elements.cc.

References jeod::OrbitalElementsMessages::convergence_error, cos_v, e_mag, KepEqtnB(), KepEqtnE(), KepEqtnH(), mean_anom, orbital_anom, sin_v, and true_anom.

#### 7.1.3.8 nu_to_anomalies()

```
int jeod::OrbitalElements::nu_to_anomalies ( )
```

Definition at line 543 of file orbital_elements.cc.

References cos_v, e_mag, mean_anom, orbital_anom, sin_v, and true_anom.

Referenced by from_cartesian().

#### 7.1.3.9 operator=()

```
OrbitalElements& jeod::OrbitalElements::operator= (
            const OrbitalElements &  )  [delete]
```

#### 7.1.3.10 set_object_name()

```
void jeod::OrbitalElements::set_object_name (
            const std::string & name )
```

Set the object name.

**Parameters**

| | | |
|---|---|---|
| in | *name* | Orbital object name |

Definition at line 59 of file orbital_elements.cc.

References object_name.

#### 7.1.3.11 set_planet_name()

```
void jeod::OrbitalElements::set_planet_name (
             const std::string & name )
```

Set the planet name.

**Parameters**

| | | |
|---|---|---|
| in | *name* | Name of planet about which the object orbits |

Definition at line 72 of file orbital_elements.cc.

References planet_name.

#### 7.1.3.12 to_cartesian()

```
int jeod::OrbitalElements::to_cartesian (
             double mu,
             double pos[3],
             double vel[3] )
```

Definition at line 382 of file orbital_elements.cc.

References arg_periapsis, cos_v, jeod::OrbitalElementsMessages::domain_error, e_mag, inclination, long_asc_↩ node, semiparam, and sin_v.

### 7.1.4 Friends And Related Function Documentation

#### 7.1.4.1 init_attrjeod__OrbitalElements

```
void init_attrjeod__OrbitalElements ( )  [friend]
```

#### 7.1.4.2 InputProcessor

```
friend class InputProcessor  [friend]
```

Definition at line 83 of file orbital_elements.hh.

### 7.1.5 Field Documentation

#### 7.1.5.1 arg_periapsis

```
double jeod::OrbitalElements::arg_periapsis {}
```

Argument of periapsis (w)

trick_units(rad)

Definition at line 105 of file orbital_elements.hh.

Referenced by from_cartesian(), and to_cartesian().

#### 7.1.5.2 cos_v

```
double jeod::OrbitalElements::cos_v {1.0}
```

Cosine of the true anomaly.

trick_units(–)

Definition at line 144 of file orbital_elements.hh.

Referenced by mean_anom_to_nu(), nu_to_anomalies(), and to_cartesian().

#### 7.1.5.3 e_mag

```
double jeod::OrbitalElements::e_mag {}
```

Magnitude of eccentricity (e)

trick_units(–)

Definition at line 97 of file orbital_elements.hh.

Referenced by from_cartesian(), mean_anom_to_nu(), nu_to_anomalies(), and to_cartesian().

#### 7.1.5.4 inclination

```
double jeod::OrbitalElements::inclination {}
```

Orbit inclination (i)

trick_units(rad)

Definition at line 101 of file orbital_elements.hh.

Referenced by from_cartesian(), and to_cartesian().

#### 7.1.5.5 long_asc_node

```
double jeod::OrbitalElements::long_asc_node {}
```

Longitude of ascending node (Omega)

trick_units(rad)

Definition at line 109 of file orbital_elements.hh.

Referenced by from_cartesian(), and to_cartesian().

#### 7.1.5.6 mean_anom

```
double jeod::OrbitalElements::mean_anom {}
```

Mean Anomaly (M)

trick_units(rad)

Definition at line 127 of file orbital_elements.hh.

Referenced by mean_anom_to_nu(), and nu_to_anomalies().

#### 7.1.5.7 mean_motion

```
double jeod::OrbitalElements::mean_motion {}
```

Mean motion of orbit (n)

trick_units(rad/s)

Definition at line 131 of file orbital_elements.hh.

Referenced by from_cartesian().

#### 7.1.5.8 object_name

```
std::string jeod::OrbitalElements::object_name  [protected]
```

Name of orbital object.

trick_units(–)

Definition at line 158 of file orbital_elements.hh.

Referenced by get_object_name(), and set_object_name().

#### 7.1.5.9 orb_ang_momentum

```
double jeod::OrbitalElements::orb_ang_momentum {}
```

Specific orbital angular momentum.

trick_units(m2/s)

Definition at line 152 of file orbital_elements.hh.

Referenced by from_cartesian().

#### 7.1.5.10 orb_energy

```
double jeod::OrbitalElements::orb_energy {}
```

Specific orbital energy.

trick_units(m2/s2)

Definition at line 148 of file orbital_elements.hh.

Referenced by from_cartesian().

#### 7.1.5.11 orbital_anom

```
double jeod::OrbitalElements::orbital_anom {}
```

Eccentric (E), Hyperbolic (H), or Parabolic (B) anomaly.

trick_units(rad)

Definition at line 135 of file orbital_elements.hh.

Referenced by mean_anom_to_nu(), and nu_to_anomalies().

#### 7.1.5.12 planet_name

```
std::string jeod::OrbitalElements::planet_name  [protected]
```

Name of planet about which the object orbits.

trick_units(–)

Definition at line 162 of file orbital_elements.hh.

Referenced by get_planet_name(), and set_planet_name().

#### 7.1.5.13 r_mag

```
double jeod::OrbitalElements::r_mag {}
```

Magnitude of orbital radius.

trick_units(m)

Definition at line 115 of file orbital_elements.hh.

Referenced by from_cartesian().

#### 7.1.5.14 semi_major_axis

```
double jeod::OrbitalElements::semi_major_axis {}
```

Semi-major-axis (a)

trick_units(m)

Definition at line 89 of file orbital_elements.hh.

Referenced by from_cartesian().

#### 7.1.5.15 semiparam

```
double jeod::OrbitalElements::semiparam {}
```

Semiparameter (p)

trick_units(m)

Definition at line 93 of file orbital_elements.hh.

Referenced by from_cartesian(), and to_cartesian().

#### 7.1.5.16 sin_v

```
double jeod::OrbitalElements::sin_v {}
```

Sine of the true anomaly.

trick_units(–)

Definition at line 140 of file orbital_elements.hh.

Referenced by mean_anom_to_nu(), nu_to_anomalies(), and to_cartesian().

#### 7.1.5.17 true_anom

```
double jeod::OrbitalElements::true_anom {}
```

True Anomaly (v)

trick_units(rad)

Definition at line 123 of file orbital_elements.hh.

Referenced by from_cartesian(), mean_anom_to_nu(), and nu_to_anomalies().

#### 7.1.5.18 vel_mag

```
double jeod::OrbitalElements::vel_mag {}
```

Magnitude of orbital velocity.

trick_units(m/s)

Definition at line 119 of file orbital_elements.hh.

Referenced by from_cartesian().

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

- orbital_elements.hh
- orbital_elements.cc

## 7.2 jeod::OrbitalElementsMessages Class Reference

Specifies the message IDs used in the orbital elements model.

```
#include <orbital_elements_messages.hh>
```

### Public Member Functions

- OrbitalElementsMessages ()=delete
- OrbitalElementsMessages (const OrbitalElementsMessages &)=delete
- OrbitalElementsMessages & operator= (const OrbitalElementsMessages &)=delete

### Static Public Attributes

- static const char * domain_error = "utils/orbital_elements/" "domain_error"

  *Issued when a value / set of values is invalid.*
- static const char * convergence_error = "utils/orbital_elements/" "convergence_error"

  *Issued when a numeric search fails to converge.*

### Friends

- class InputProcessor
- void init_attrjeod__OrbitalElementsMessages ()

### 7.2.1 Detailed Description

Specifies the message IDs used in the orbital elements model.

Definition at line 81 of file orbital_elements_messages.hh.

### 7.2.2 Constructor & Destructor Documentation

#### 7.2.2.1 OrbitalElementsMessages() [1/2]

```
jeod::OrbitalElementsMessages::OrbitalElementsMessages ( )  [delete]
```

#### 7.2.2.2 OrbitalElementsMessages() [2/2]

```
jeod::OrbitalElementsMessages::OrbitalElementsMessages (
            const OrbitalElementsMessages &  )  [delete]
```

### 7.2.3 Member Function Documentation

#### 7.2.3.1 operator=()

```
OrbitalElementsMessages& jeod::OrbitalElementsMessages::operator= (
            const OrbitalElementsMessages &  )  [delete]
```

### 7.2.4 Friends And Related Function Documentation

#### 7.2.4.1 init_attrjeod__OrbitalElementsMessages

```
void init_attrjeod__OrbitalElementsMessages ( )  [friend]
```

#### 7.2.4.2 InputProcessor

```
friend class InputProcessor  [friend]
```

Definition at line 83 of file orbital_elements_messages.hh.

### 7.2.5 Field Documentation

#### 7.2.5.1 convergence_error

```
char const * jeod::OrbitalElementsMessages::convergence_error = "utils/orbital_elements/"
"convergence_error"  [static]
```

Issued when a numeric search fails to converge.

trick_units(–)

Definition at line 93 of file orbital_elements_messages.hh.

Referenced by jeod::OrbitalElements::mean_anom_to_nu().

#### 7.2.5.2 domain_error

```
char const * jeod::OrbitalElementsMessages::domain_error = "utils/orbital_elements/" "domain_↩
error"  [static]
```

Issued when a value / set of values is invalid.

trick_units(–)

Definition at line 88 of file orbital_elements_messages.hh.

Referenced by jeod::OrbitalElements::to_cartesian().

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

- orbital_elements_messages.hh
- orbital_elements_messages.cc

# Chapter 8

# File Documentation

## 8.1 orbital_elements.cc File Reference

Define methods for the OrbitalElements class.

```
#include <cmath>
#include <cstddef>
#include "utils/math/include/vector3.hh"
#include "utils/memory/include/jeod_alloc.hh"
#include "utils/message/include/message_handler.hh"
#include "../include/orbital_elements.hh"
#include "../include/orbital_elements_messages.hh"
```

### Namespaces

- jeod

  *Namespace jeod.*

### 8.1.1 Detailed Description

Define methods for the OrbitalElements class.

## 8.2 orbital_elements.hh File Reference

Orbital elements class definition.

```
#include <string>
#include "utils/sim_interface/include/jeod_class.hh"
```

### Data Structures

- class jeod::OrbitalElements

  *Represents state in terms of Keplerian orbital elements.*

### Namespaces

- jeod

  *Namespace jeod.*

#### 8.2.1 Detailed Description

Orbital elements class definition.

## 8.3 orbital_elements_messages.cc File Reference

Implement the class OrbitalElementsMessages.

```
#include "utils/message/include/make_message_code.hh"
#include "../include/orbital_elements_messages.hh"
```

### Namespaces

- jeod

  *Namespace jeod.*

### Macros

- #define MAKE_ORBITALELEMENTS_MESSAGE_CODE(id) JEOD_MAKE_MESSAGE_CODE(Orbital↩ ElementsMessages, "utils/orbital_elements/", id)

#### 8.3.1 Detailed Description

Implement the class OrbitalElementsMessages.

#### 8.3.2 Macro Definition Documentation

#### 8.3.2.1 MAKE_ORBITALELEMENTS_MESSAGE_CODE

```
#define MAKE_ORBITALELEMENTS_MESSAGE_CODE(
              id ) JEOD_MAKE_MESSAGE_CODE(OrbitalElementsMessages, "utils/orbital_elements/",
id)
```

Definition at line 43 of file orbital_elements_messages.cc.

## 8.4 orbital_elements_messages.hh File Reference

Define the class OrbitalElementsMessages, the class that specifies the message IDs used in the orbital elements model.

```
#include "utils/sim_interface/include/jeod_class.hh"
```

### Data Structures

- class jeod::OrbitalElementsMessages

  *Specifies the message IDs used in the orbital elements model.*

### Namespaces

- jeod

  *Namespace jeod.*

### 8.4.1 Detailed Description

Define the class OrbitalElementsMessages, the class that specifies the message IDs used in the orbital elements model.

# Index