

ModelTemplateModel

5.1

Generated by Doxygen 1.8.14

Contents

1	Module Index	1
1.1	Modules	1
2	Namespace Index	3
2.1	Namespace List	3
3	Data Structure Index	5
3.1	Data Structures	5
4	File Index	7
4.1	File List	7
5	Module Documentation	9
5.1	Models	9
5.1.1	Detailed Description	9
5.2	Utils	10
5.2.1	Detailed Description	10
5.3	ModelTemplate	11
5.3.1	Detailed Description	11
6	Namespace Documentation	13
6.1	jeod Namespace Reference	13
6.1.1	Detailed Description	13

7	Data Structure Documentation	15
7.1	jeod::Foo Class Reference	15
7.1.1	Detailed Description	15
7.1.2	Constructor & Destructor Documentation	16
7.1.2.1	Foo()	16
7.1.2.2	~Foo()	16
7.1.3	Member Function Documentation	16
7.1.3.1	get_num()	16
7.1.3.2	set_num()	16
7.1.4	Field Documentation	17
7.1.4.1	num	17
7.2	jeod::TemplateMessages Class Reference	17
7.2.1	Detailed Description	18
7.2.2	Constructor & Destructor Documentation	18
7.2.2.1	TemplateMessages() [1/2]	18
7.2.2.2	TemplateMessages() [2/2]	18
7.2.3	Member Function Documentation	18
7.2.3.1	operator=()	18
7.2.4	Friends And Related Function Documentation	18
7.2.4.1	init_attrjeod__TemplateMessages	18
7.2.4.2	InputProcessor	18
7.2.5	Field Documentation	19
7.2.5.1	big_negnum	19
7.2.5.2	big_posnum	19
7.2.5.3	zero	19
8	File Documentation	21
8.1	foo.cc File Reference	21
8.1.1	Detailed Description	21
8.2	foo.hh File Reference	21
8.2.1	Detailed Description	22
8.3	template_messages.cc File Reference	22
8.3.1	Detailed Description	22
8.3.2	Macro Definition Documentation	23
8.3.2.1	MAKE_TEMPLATE_MESSAGE_CODE	23
8.4	template_messages.hh File Reference	23
8.4.1	Detailed Description	24
	Index	25

Chapter 1

Module Index

1.1 Modules

Here is a list of all modules:

Models	9
Utils	10
ModelTemplate	11

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

jeod	Namespace jeod	13
----------------------	--------------------------	--------------------

Chapter 3

Data Structure Index

3.1 Data Structures

Here are the data structures with brief descriptions:

jeod::Foo	All JEOD classes should have a doxygen-style header that describes the class	15
jeod::TemplateMessages	Declares messages associated with the model template model	17

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

foo.cc	Implement class Foo	21
foo.hh	The file header purpose should describe why the file exists, briefly summarizing the contents of the file	21
template_messages.cc	Implement the class TemplateMessages	22
template_messages.hh	Define the class TemplateMessages, the class that specifies the message IDs used in the model template model	23

Chapter 5

Module Documentation

5.1 Models

Modules

- [Utils](#)

5.1.1 Detailed Description

5.2 Utils

Modules

- [ModelTemplate](#)

5.2.1 Detailed Description

5.3 ModelTemplate

Files

- file [foo.hh](#)
The file header purpose should describe why the file exists, briefly summarizing the contents of the file.
- file [template_messages.hh](#)
Define the class `TemplateMessages`, the class that specifies the message IDs used in the model template model.
- file [foo.cc](#)
Implement class `Foo`.
- file [template_messages.cc](#)
Implement the class `TemplateMessages`.

Namespaces

- [jeod](#)
Namespace `jeod`.

5.3.1 Detailed Description

Chapter 6

Namespace Documentation

6.1 jeod Namespace Reference

Namespace jeod.

Data Structures

- class [Foo](#)
All JEOD classes should have a doxygen-style header that describes the class.
- class [TemplateMessages](#)
Declares messages associated with the model template model.

6.1.1 Detailed Description

Namespace jeod.

Chapter 7

Data Structure Documentation

7.1 jeod::Foo Class Reference

All JEOD classes should have a doxygen-style header that describes the class.

```
#include <foo.hh>
```

Public Member Functions

- `Foo ()`
Foo default constructor.
- `~Foo ()`
Foo destructor.
- `void set_num (int value)`
Setter for `Foo::num`.
- `int get_num ()`
Getter for `Foo::num`.

Protected Attributes

- `int num {42}`
A non-zero number between -42 and 42 (inclusive).

7.1.1 Detailed Description

All JEOD classes should have a doxygen-style header that describes the class.

Explain why the class exists, what it does. In this case, this class exists to demonstrate some base JEOD documentation concepts.

Definition at line 79 of file foo.hh.

7.1.2 Constructor & Destructor Documentation

7.1.2.1 Foo()

```
jeod::Foo::Foo ( )
```

[Foo](#) default constructor.

Definition at line 38 of file foo.cc.

7.1.2.2 ~Foo()

```
jeod::Foo::~~Foo ( )
```

[Foo](#) destructor.

Definition at line 46 of file foo.cc.

7.1.3 Member Function Documentation

7.1.3.1 get_num()

```
int jeod::Foo::get_num ( )
```

Getter for [Foo::num](#).

Definition at line 85 of file foo.cc.

References [num](#).

7.1.3.2 set_num()

```
void jeod::Foo::set_num (
    int value )
```

Setter for [Foo::num](#).

Definition at line 54 of file foo.cc.

References [jeod::TemplateMessages::big_negnum](#), [num](#), and [jeod::TemplateMessages::zero](#).

7.1.4 Field Documentation

7.1.4.1 num

```
int jeod::Foo::num {42} [protected]
```

A non-zero number between -42 and 42 (inclusive).

trick_units(—)

Definition at line 94 of file foo.hh.

Referenced by get_num(), and set_num().

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

- [foo.hh](#)
- [foo.cc](#)

7.2 jeod::TemplateMessages Class Reference

Declares messages associated with the model template model.

```
#include <template_messages.hh>
```

Public Member Functions

- [TemplateMessages](#) ()=delete
- [TemplateMessages](#) (const [TemplateMessages](#) &)=delete
- [TemplateMessages](#) & operator= (const [TemplateMessages](#) &)=delete

Static Public Attributes

- static const char * [big_negnum](#) = "utils/model_template/" "big_negnum"
Number is less than -42.
- static const char * [zero](#) = "utils/model_template/" "zero"
Number is zero.
- static const char * [big_posnum](#) = "utils/model_template/" "big_posnum"
Number is greater than 42.

Friends

- class [InputProcessor](#)
- void [init_attrjeod__TemplateMessages](#) ()

7.2.1 Detailed Description

Declares messages associated with the model template model.

Definition at line 81 of file template_messages.hh.

7.2.2 Constructor & Destructor Documentation

7.2.2.1 TemplateMessages() [1/2]

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

7.2.2.2 TemplateMessages() [2/2]

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

7.2.3 Member Function Documentation

7.2.3.1 operator=()

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

7.2.4 Friends And Related Function Documentation

7.2.4.1 init_attrjeod__TemplateMessages

```
void init_attrjeod__TemplateMessages ( ) [friend]
```

7.2.4.2 InputProcessor

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

Definition at line 83 of file template_messages.hh.

7.2.5 Field Documentation

7.2.5.1 big_negnum

```
char const * jeod::TemplateMessages::big_negnum = "utils/model_template/" "big_negnum" [static]
```

Number is less than -42.

trick_units(-)

Definition at line 88 of file template_messages.hh.

Referenced by jeod::Foo::set_num().

7.2.5.2 big_posnum

```
char const * jeod::TemplateMessages::big_posnum = "utils/model_template/" "big_posnum" [static]
```

Number is greater than 42.

trick_units(-)

Definition at line 98 of file template_messages.hh.

7.2.5.3 zero

```
char const * jeod::TemplateMessages::zero = "utils/model_template/" "zero" [static]
```

Number is zero.

trick_units(-)

Definition at line 93 of file template_messages.hh.

Referenced by jeod::Foo::set_num().

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

- [template_messages.hh](#)
- [template_messages.cc](#)

Chapter 8

File Documentation

8.1 foo.cc File Reference

Implement class Foo.

```
#include <iostream>
#include "utils/message/include/message_handler.hh"
#include "../include/foo.hh"
#include "../include/template_messages.hh"
```

Namespaces

- `jeod`

Namespace jeod.

8.1.1 Detailed Description

Implement class Foo.

8.2 foo.hh File Reference

The file header purpose should describe why the file exists, briefly summarizing the contents of the file.

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

Data Structures

- class `jeod::Foo`

All JEOD classes should have a doxygen-style header that describes the class.

Namespaces

- [jeod](#)

Namespace jeod.

8.2.1 Detailed Description

The file header purpose should describe why the file exists, briefly summarizing the contents of the file.

In this case, the purpose is to define the class Foo.

8.3 template_messages.cc File Reference

Implement the class TemplateMessages.

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

Namespaces

- [jeod](#)

Namespace jeod.

Macros

- `#define MAKE_TEMPLATE_MESSAGE_CODE(id) JEOD_MAKE_MESSAGE_CODE(TemplateMessages, "utils/model_template/", id)`

The goal in a model message code implementation file is to create storage for and initialize each of static data members declared in the model message class.

8.3.1 Detailed Description

Implement the class TemplateMessages.

And in this case, demonstrate how to implement message code classes.

The MessageHandler defines several methods for informing the user of special conditions that can arise, ranging in severity from fatal errors to detailed debug information. Each of these methods takes a message_code argument. The intent of the message code is to identify the model that invoked the message handler and to characterize the nature of the message.

JEOD models implement that intent by making each of the message codes be a static data member of a model-specific message class. Each such data member points to a literal character string of the form "model/path/message_id", where "model/path" is the path to the model, and "message_id" is the stringified data member name.

8.3.2 Macro Definition Documentation

8.3.2.1 MAKE_TEMPLATE_MESSAGE_CODE

```
#define MAKE_TEMPLATE_MESSAGE_CODE(  
    id ) JEOD_MAKE_MESSAGE_CODE(TemplateMessages, "utils/model_template/", id)
```

The goal in a model message code implementation file is to create storage for and initialize each of static data members declared in the model message class.

There are several ways to do this. The examples below target the fictitious message class FooMessages defined in the model utils/foo. The class defines two message codes, FooMessages::fatal and FooMessages::warning.

- Most primitive:

```
const char * FooMessages::fatal = "utils/foo/fatal";  
const char * FooMessages::warning = "utils/foo/warning";
```

- Recognizing that "utils/foo/" is common:

```
#define PATH "utils/foo/"  
const char * FooMessages::fatal = PATH "fatal";  
const char * FooMessages::warning = PATH "warning";
```

- Using the JEOD_MAKE_MESSAGE_CODE macro:

```
#define PATH "utils/foo/"  
JEOD_MAKE_MESSAGE_CODE (FooMessages, PATH, fatal);  
JEOD_MAKE_MESSAGE_CODE (FooMessages, PATH, warning);
```

- Using the model-specific macro MAKE_FOO_MESSAGE_CODE:

```
#define MAKE_FOO_MESSAGE_CODE(id) \  
    JEOD_MAKE_MESSAGE_CODE(FooMessages, "utils/foo/", id)  
MAKE_FOO_MESSAGE_CODE (fatal);  
MAKE_FOO_MESSAGE_CODE (warning);
```

Define TemplateMessages data member *id*.

Parameters

<i>id</i>	The name of the data member.
-----------	------------------------------

Definition at line 84 of file template_messages.cc.

8.4 template_messages.hh File Reference

Define the class TemplateMessages, the class that specifies the message IDs used in the model template model.

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

Data Structures

- class [jeod::TemplateMessages](#)
Declares messages associated with the model template model.

Namespaces

- [jeod](#)
Namespace jeod.

8.4.1 Detailed Description

Define the class `TemplateMessages`, the class that specifies the message IDs used in the model template model.

Index

- ~Foo
 - jeod::Foo, [16](#)
- big_negnum
 - jeod::TemplateMessages, [19](#)
- big_posnum
 - jeod::TemplateMessages, [19](#)
- Foo
 - jeod::Foo, [16](#)
- foo.cc, [21](#)
- foo.hh, [21](#)
- get_num
 - jeod::Foo, [16](#)
- init_attrjeod__TemplateMessages
 - jeod::TemplateMessages, [18](#)
- InputProcessor
 - jeod::TemplateMessages, [18](#)
- jeod, [13](#)
- jeod::Foo, [15](#)
 - ~Foo, [16](#)
 - Foo, [16](#)
 - get_num, [16](#)
 - num, [17](#)
 - set_num, [16](#)
- jeod::TemplateMessages, [17](#)
 - big_negnum, [19](#)
 - big_posnum, [19](#)
 - init_attrjeod__TemplateMessages, [18](#)
 - InputProcessor, [18](#)
 - operator=, [18](#)
 - TemplateMessages, [18](#)
 - zero, [19](#)
- MAKE_TEMPLATE_MESSAGE_CODE
 - template_messages.cc, [23](#)
- ModelTemplate, [11](#)
- Models, [9](#)
- num
 - jeod::Foo, [17](#)
- operator=
 - jeod::TemplateMessages, [18](#)
- set_num
 - jeod::Foo, [16](#)
- template_messages.cc, [22](#)
- MAKE_TEMPLATE_MESSAGE_CODE, [23](#)
- template_messages.hh, [23](#)
- TemplateMessages
 - jeod::TemplateMessages, [18](#)
- Utils, [10](#)
- zero
 - jeod::TemplateMessages, [19](#)