

MessageHandlingClass

5.0

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	Hierarchical Index	5
3.1	Class Hierarchy	5
4	Data Structure Index	7
4.1	Data Structures	7
5	File Index	9
5.1	File List	9
6	Module Documentation	11
6.1	Models	11
6.1.1	Detailed Description	11
6.2	Utils	12
6.2.1	Detailed Description	12
6.3	Message	13
6.3.1	Detailed Description	13
7	Namespace Documentation	15
7.1	jeod Namespace Reference	15
7.1.1	Detailed Description	15

8 Data Structure Documentation	17
8.1 jeod::MessageHandler Class Reference	17
8.1.1 Detailed Description	20
8.1.2 Constructor & Destructor Documentation	20
8.1.2.1 MessageHandler() [1/2]	20
8.1.2.2 ~MessageHandler()	21
8.1.2.3 MessageHandler() [2/2]	21
8.1.3 Member Function Documentation	21
8.1.3.1 add_suppressed_code()	21
8.1.3.2 clear_suppressed_codes()	21
8.1.3.3 debug()	22
8.1.3.4 delete_suppressed_code()	22
8.1.3.5 deregister_contents()	23
8.1.3.6 error()	23
8.1.3.7 fail()	23
8.1.3.8 get_suppress_id()	24
8.1.3.9 get_suppress_location()	24
8.1.3.10 get_suppression_level()	25
8.1.3.11 inform()	25
8.1.3.12 no_handler_error()	25
8.1.3.13 operator=()	26
8.1.3.14 process_add_suppressed_code()	26
8.1.3.15 process_clear_suppressed_codes()	27
8.1.3.16 process_delete_suppressed_code()	27
8.1.3.17 process_message()	27
8.1.3.18 register_contents()	28
8.1.3.19 send_message()	28
8.1.3.20 set_suppress_id()	29
8.1.3.21 set_suppress_location()	29
8.1.3.22 set_suppression_level()	30

8.1.3.23	va_send_message()	30
8.1.3.24	warn()	31
8.1.4	Friends And Related Function Documentation	31
8.1.4.1	init_attrjeod__MessageHandler	31
8.1.4.2	InputProcessor	31
8.1.5	Field Documentation	31
8.1.5.1	Debug	32
8.1.5.2	Error	32
8.1.5.3	Failure	32
8.1.5.4	handler	33
8.1.5.5	Notice	33
8.1.5.6	suppress_id	33
8.1.5.7	suppress_location	34
8.1.5.8	suppression_level	34
8.1.5.9	Warning	34
8.2	jeod::MessageMessages Class Reference	35
8.2.1	Detailed Description	35
8.2.2	Constructor & Destructor Documentation	35
8.2.2.1	MessageMessages() [1/2]	35
8.2.2.2	MessageMessages() [2/2]	35
8.2.3	Member Function Documentation	36
8.2.3.1	operator=()	36
8.2.4	Field Documentation	36
8.2.4.1	singleton_error	36
8.3	jeod::SuppressedCodeMessageHandler Class Reference	36
8.3.1	Detailed Description	38
8.3.2	Constructor & Destructor Documentation	38
8.3.2.1	SuppressedCodeMessageHandler() [1/2]	38
8.3.2.2	~SuppressedCodeMessageHandler()	38
8.3.2.3	SuppressedCodeMessageHandler() [2/2]	38
8.3.3	Member Function Documentation	38
8.3.3.1	deregister_contents()	39
8.3.3.2	message_is_to_be_printed()	39
8.3.3.3	operator=()	39
8.3.3.4	process_add_suppressed_code()	39
8.3.3.5	process_clear_suppressed_code()	40
8.3.3.6	process_delete_suppressed_code()	40
8.3.3.7	register_contents()	40
8.3.4	Friends And Related Function Documentation	41
8.3.4.1	init_attrjeod__SuppressedCodeMessageHandler	41
8.3.4.2	InputProcessor	41
8.3.5	Field Documentation	41
8.3.5.1	suppressed_codes	41

9 File Documentation	43
9.1 <code>class_declarations.hh</code> File Reference	43
9.1.1 Detailed Description	43
9.2 <code>make_message_code.hh</code> File Reference	43
9.2.1 Detailed Description	44
9.2.2 Macro Definition Documentation	44
9.2.2.1 <code>JEOD_MAKE_MESSAGE_CODE</code>	44
9.3 <code>message_handler.cc</code> File Reference	44
9.3.1 Detailed Description	44
9.4 <code>message_handler.hh</code> File Reference	45
9.4.1 Detailed Description	45
9.5 <code>message_messages.cc</code> File Reference	45
9.5.1 Detailed Description	45
9.5.2 Macro Definition Documentation	46
9.5.2.1 <code>MAKE_MESSAGE_MESSAGE_CODE</code>	46
9.6 <code>message_messages.hh</code> File Reference	46
9.6.1 Detailed Description	46
9.7 <code>suppressed_code_message_handler.cc</code> File Reference	46
9.7.1 Detailed Description	47
9.8 <code>suppressed_code_message_handler.hh</code> File Reference	47
9.8.1 Detailed Description	47
Index	49

Chapter 1

Module Index

1.1 Modules

Here is a list of all modules:

Models	11
Utils	12
Message	13

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

jeod	Namespace jeod	15
----------------------	--------------------------	--------------------

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

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

jeod::MessageHandler	17
jeod::SuppressedCodeMessageHandler	36
jeod::MessageMessages	35

Chapter 4

Data Structure Index

4.1 Data Structures

Here are the data structures with brief descriptions:

jeod::MessageHandler	
The base class for generating JEOD messages	17
jeod::MessageMessages	
Specifies the message IDs used in the message handler model	35
jeod::SuppressedCodeMessageHandler	
Adds the capability to suppress messages by their message code to the base MessageHandler class	36

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

class_declarations.hh	Forward declarations of classes defined in this module	43
make_message_code.hh	Define JEOD_MAKE_MESSAGE_CODE	43
message_handler.cc	Define member functions for the class MessageHandler	44
message_handler.hh	Define the class MessageHandler, the base class for generating messages	45
message_messages.cc	Implement the class MessageMessages	45
message_messages.hh	Define the class MessageMessages	46
suppressed_code_message_handler.cc	Define member functions for the class SuppressedCodeMessageHandler	46
suppressed_code_message_handler.hh	Define the class SuppressedCodeMessageHandler, which adds the capability to suppress messages by their message code	47

Chapter 6

Module Documentation

6.1 Models

Modules

- [Utils](#)

6.1.1 Detailed Description

6.2 Utils

Modules

- [Message](#)

6.2.1 Detailed Description

6.3 Message

Files

- file [class_declarations.hh](#)
Forward declarations of classes defined in this module.
- file [make_message_code.hh](#)
Define JEOD_MAKE_MESSAGE_CODE.
- file [message_handler.hh](#)
Define the class MessageHandler, the base class for generating messages.
- file [message_messages.hh](#)
Define the class MessageMessages.
- file [suppressed_code_message_handler.hh](#)
Define the class SuppressedCodeMessageHandler, which adds the capability to suppress messages by their message code.
- file [message_handler.cc](#)
Define member functions for the class MessageHandler.
- file [message_messages.cc](#)
Implement the class MessageMessages.
- file [suppressed_code_message_handler.cc](#)
Define member functions for the class SuppressedCodeMessageHandler.

Namespaces

- [jeod](#)
Namespace jeod.

6.3.1 Detailed Description

Chapter 7

Namespace Documentation

7.1 jeod Namespace Reference

Namespace jeod.

Data Structures

- class [MessageHandler](#)
The base class for generating JEOD messages.
- class [MessageMessages](#)
Specifies the message IDs used in the message handler model.
- class [SuppressedCodeMessageHandler](#)
Adds the capability to suppress messages by their message code to the base [MessageHandler](#) class.

7.1.1 Detailed Description

Namespace jeod.

Chapter 8

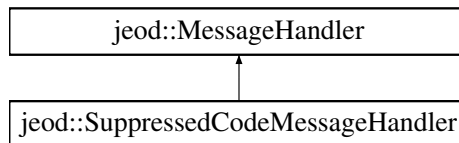
Data Structure Documentation

8.1 jeod::MessageHandler Class Reference

The base class for generating JEOD messages.

```
#include <message_handler.hh>
```

Inheritance diagram for jeod::MessageHandler:



Public Member Functions

- [MessageHandler](#) ()
Construct a [MessageHandler](#).
- virtual [~MessageHandler](#) ()
Destruct a [MessageHandler](#).
- virtual void [register_contents](#) ()
Register the checkpointable contents of the handler with the simulation interface.
- virtual void [deregister_contents](#) ()
Deregister the checkpointable contents of the handler with the simulation interface.

Static Public Member Functions

- static void [fail](#) (const char *file, unsigned int line, const char *msg_code, const char *format,...)
Generate a message with negative severity, [MessageHandler::Failure](#), and terminate the simulation.
- static void [error](#) (const char *file, unsigned int line, const char *msg_code, const char *format,...)
Generate a message with severity [MessageHandler::Error](#).
- static void [warn](#) (const char *file, unsigned int line, const char *msg_code, const char *format,...)
Generate a message with severity [MessageHandler::Warning](#).
- static void [inform](#) (const char *file, unsigned int line, const char *msg_code, const char *format,...)

Generates a message with severity [MessageHandler::Notice](#).

- static void [debug](#) (const char *file, unsigned int line, const char *msg_code, const char *format,...)

Generate a message with severity [MessageHandler::Debug](#).

- static void [send_message](#) (int severity, const char *prefix, const char *file, unsigned int line, const char *msg_code, const char *format,...)

Generic variable arguments message interface.

- static void [va_send_message](#) (int severity, const char *prefix, const char *file, unsigned int line, const char *msg_code, const char *format, va_list args)

Generic variable arguments message interface.

- static void [set_suppression_level](#) (unsigned int [suppression_level](#))

Set the suppression level in the global message handler.

- static unsigned int [get_suppression_level](#) ()

Get the suppress_id of the global message handler.

- static void [add_suppressed_code](#) (const char *msg_code)

Add a message code to the set of messages that are to be suppressed for messages with positive severity level.

- static void [delete_suppressed_code](#) (const char *msg_code)

Delete a message code from the set of suppressed message codes.

- static void [clear_suppressed_codes](#) ()

Clear the set of suppressed message codes.

- static void [set_suppress_id](#) (bool [suppress_id](#))

Set the suppress_id flag in the global message handler.

- static bool [get_suppress_id](#) ()

Get the suppress_id of the global message handler.

- static void [set_suppress_location](#) (bool [suppress_location](#))

Set the suppress_location in the global message handler.

- static bool [get_suppress_location](#) ()

Get the suppress_location of the global message handler.

Static Public Attributes

- static const int [Failure](#) = -1

The severity value passed by the static public [MessageHandler::fail](#) method to the derived class process_message method.

- static const int [Error](#) = 0

The severity value passed by the static public [MessageHandler::error](#) method to the derived class process_message method.

- static const int [Warning](#) = 9

The severity value passed by the static public [MessageHandler::warn](#) method to the derived class process_message method.

- static const int [Notice](#) = 99

The severity value passed by the static public [MessageHandler::inform](#) method to the derived class process_message method.

- static const int [Debug](#) = 999

The severity value passed by the static public [MessageHandler::debug](#) method to the derived class process_message method.

Protected Member Functions

- virtual void [process_message](#) (int severity, const char *prefix, const char *file, unsigned int line, const char *msg_code, const char *format, va_list args) const =0
Generate the message.
- virtual void [process_add_suppressed_code](#) (const char *msg_code)
Add a message code to the set of messages that are to be suppressed.
- virtual void [process_delete_suppressed_code](#) (const char *msg_code)
Delete a message code from the set of suppressed message codes.
- virtual void [process_clear_suppressed_codes](#) ()
Clear the set of suppressed message codes.

Static Protected Member Functions

- static void [no_handler_error](#) (void)
Handle the error condition where there is no global handler.

Protected Attributes

- unsigned int [suppression_level](#)
All messages have an associated severity level, with increasingly positive values indicating warnings of decreasing severity.
- bool [suppress_id](#)
This flag indicates whether the message ID is printed for unsuppressed messages.
- bool [suppress_location](#)
This flag indicates whether the message source file and line number printed for unsuppressed messages.

Static Protected Attributes

- static [MessageHandler](#) * [handler](#) = NULL
The [MessageHandler](#) instance that generates messages.

Private Member Functions

- [MessageHandler](#) (const [MessageHandler](#) &)
Not implemented.
- [MessageHandler](#) & [operator=](#) (const [MessageHandler](#) &)
Not implemented.

Friends

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

8.1.1 Detailed Description

The base class for generating JEOD messages.

This class provides:

- A suite of public static message generation and message control functions. The message generation functions provide the mechanism to generate and present messages of various levels of severity to the simulation user. The message control functions provide the user the ability to control which messages will be presented and to control their presentation.
- A set of defined constants that denote the severity levels used by JEOD.
 - Failure (-1): Any negative severity level value indicates an irrecoverable error. The simulation will terminate immediately.
 - Error (0): Errors almost certainly invalidate the simulation output. All severity zero messages are printed; they cannot be disabled. The difference between failures and errors is that there is some recovery from an error that lets the simulation limp on and find the next error.
 - Warning (9): Warnings represent conditions that a model deems to be suspect but not necessarily dangerous. Warning messages are printed by default, but they can be disabled.
 - Notice (99): Notices represent conditions that a model deems to be suspicious, but not necessarily in error. Notifications are not printed by default, but they can be enabled.
 - Debug (999): Debug messages typically demonstrate progress of some sort. Enabling them may well result in spew.
- A public default constructor and destructor. The constructor ensures that the created object is indeed a singleton.
-

Definition at line 108 of file message_handler.hh.

8.1.2 Constructor & Destructor Documentation

8.1.2.1 MessageHandler() [1/2]

```
jeod::MessageHandler::MessageHandler (
    void )
```

Construct a [MessageHandler](#).

This default constructor sets the suppression level to [MessageHandler::Warning](#), which means that messages of that severity and lower will be printed. The `suppress_id` and `suppress_location` flags are set to false; auxiliary information is not suppressed.

Definition at line 575 of file message_handler.cc.

References [error\(\)](#), [handler](#), and [jeod::MessageMessages::singleton_error](#).

8.1.2.2 ~MessageHandler()

```
jeod::MessageHandler::~~MessageHandler (
    void ) [virtual]
```

Destruct a [MessageHandler](#).

Definition at line 601 of file message_handler.cc.

References handler.

8.1.2.3 MessageHandler() [2/2]

```
jeod::MessageHandler::MessageHandler (
    const MessageHandler & ) [private]
```

Not implemented.

8.1.3 Member Function Documentation

8.1.3.1 add_suppressed_code()

```
void jeod::MessageHandler::add_suppressed_code (
    const char * msg_code ) [static]
```

Add a message code to the set of messages that are to be suppressed for messages with positive severity level.

Note: Fatal errors and serious errors cannot be suppressed.

Parameters

in	<i>msg_code</i>	Message code to be suppressed
----	-----------------	-------------------------------

Definition at line 393 of file message_handler.cc.

References handler, no_handler_error(), and process_add_suppressed_code().

8.1.3.2 clear_suppressed_codes()

```
void jeod::MessageHandler::clear_suppressed_codes (
    void ) [static]
```

Clear the set of suppressed message codes.

Definition at line 436 of file message_handler.cc.

References handler, no_handler_error(), and process_clear_suppressed_codes().

8.1.3.3 debug()

```
void jeod::MessageHandler::debug (
    const char * file,
    unsigned int line,
    const char * msg_code,
    const char * format,
    ... ) [static]
```

Generate a message with severity [MessageHandler::Debug](#).

Debug messages should never be used for erroneous conditions. They should instead be used for describing nominal behavior. Note that debug messages are nominally suppressed.

Parameters

in	<i>file</i>	Typically FILE
in	<i>line</i>	Typically LINE
in	<i>msg_code</i>	Message code
in	<i>format</i>	sprintf format
in	...	sprintf arguments

Definition at line 237 of file message_handler.cc.

References Debug, handler, no_handler_error(), and process_message().

8.1.3.4 delete_suppressed_code()

```
void jeod::MessageHandler::delete_suppressed_code (
    const char * msg_code ) [static]
```

Delete a message code from the set of suppressed message codes.

Parameters

in	<i>msg_code</i>	Message code to be unsuppressed
----	-----------------	---------------------------------

Definition at line 415 of file message_handler.cc.

References handler, no_handler_error(), and process_delete_suppressed_code().

8.1.3.5 deregister_contents()

```
virtual void jeod::MessageHandler::deregister_contents (
    void ) [inline], [virtual]
```

Deregister the checkpointable contents of the handler with the simulation interface.

The base [MessageHandler](#) has not such content.

Reimplemented in [jeod::SuppressedCodeMessageHandler](#).

Definition at line 250 of file message_handler.hh.

8.1.3.6 error()

```
void jeod::MessageHandler::error (
    const char * file,
    unsigned int line,
    const char * msg_code,
    const char * format,
    ... ) [static]
```

Generate a message with severity [MessageHandler::Error](#).

An error represents a very serious problem. The intent is to represent errors that invalidate simulation results but for which a recovery path does exist. Using [MessageHandler::error](#) rather than [MessageHandler::fail](#) enables the user to address multiple errors at a time.

A conforming implementation of a class that derives [MessageHandler](#) will always report error messages. Errors should not be suppressed.

Parameters

in	<i>file</i>	Typically FILE
in	<i>line</i>	Typically LINE
in	<i>msg_code</i>	Message code
in	<i>format</i>	sprintf format
in	...	sprintf arguments

Definition at line 128 of file message_handler.cc.

References [Error](#), [handler](#), [no_handler_error\(\)](#), and [process_message\(\)](#).

Referenced by [MessageHandler\(\)](#).

8.1.3.7 fail()

```
void jeod::MessageHandler::fail (
    const char * file,
```

```

    unsigned int line,
    const char * msg_code,
    const char * format,
    ... ) [static]

```

Generate a message with negative severity, [MessageHandler::Failure](#), and terminate the simulation.

The intent of this method is to handle erroneous situations for which no recovery path exists. If a recovery path does exist, even if very suspect, callers of this method should consider calling [MessageHandler::error](#) as an alternative.

A conforming implementation of a class that derives [MessageHandler](#) will force the simulation to terminate upon receipt of a message with negative severity.

Parameters

in	<i>file</i>	Typically FILE
in	<i>line</i>	Typically LINE
in	<i>msg_code</i>	Message code
in	<i>format</i>	sprintf format
in	...	sprintf arguments

Definition at line 85 of file `message_handler.cc`.

References `Failure`, `handler`, `no_handler_error()`, and `process_message()`.

8.1.3.8 `get_suppress_id()`

```

bool jeod::MessageHandler::get_suppress_id (
    void ) [static]

```

Get the `suppress_id` of the global message handler.

Returns

ID value
Units: Suppress

Definition at line 478 of file `message_handler.cc`.

References `handler`, `no_handler_error()`, and `suppress_id`.

8.1.3.9 `get_suppress_location()`

```

bool jeod::MessageHandler::get_suppress_location (
    void ) [static]

```

Get the `suppress_location` of the global message handler.

Returns

Suppress location value

Definition at line 523 of file `message_handler.cc`.

References `handler`, `no_handler_error()`, and `suppress_location`.

8.1.3.10 get_suppression_level()

```
unsigned int jeod::MessageHandler::get_suppression_level (
    void ) [static]
```

Get the suppress_id of the global message handler.

Returns

Suppression level value

Definition at line 366 of file message_handler.cc.

References handler, no_handler_error(), and suppression_level.

8.1.3.11 inform()

```
void jeod::MessageHandler::inform (
    const char * file,
    unsigned int line,
    const char * msg_code,
    const char * format,
    ... ) [static]
```

Generates a message with severity [MessageHandler::Notice](#).

Informational notices should not represent problems of any significance as the default behavior is to suppress such messages.

Parameters

in	<i>file</i>	Typically FILE
in	<i>line</i>	Typically LINE
in	<i>msg_code</i>	Message code
in	<i>format</i>	sprintf format
in	...	sprintf arguments

Definition at line 201 of file message_handler.cc.

References handler, no_handler_error(), Notice, and process_message().

8.1.3.12 no_handler_error()

```
void jeod::MessageHandler::no_handler_error (
    void ) [static], [protected]
```

Handle the error condition where there is no global handler.

Note

That this condition exists means the simulation is non-compliant.

Assumptions and Limitations

- All JEOD-based simulations must have a message handler and memory handler that at instantiated prior to instantiating any other JEOD-based model and destroyed after all those other models have been destroyed.
- That no message handler exists means the simulation is not a compliant with the above restrictions.
- The handling of this condition is intentionally simplistic. An error message is printed and the simulation is terminated via a system call to exit.

Definition at line 558 of file message_handler.cc.

Referenced by `add_suppressed_code()`, `clear_suppressed_codes()`, `debug()`, `delete_suppressed_code()`, `error()`, `fail()`, `get_suppress_id()`, `get_suppress_location()`, `get_suppression_level()`, `inform()`, `send_message()`, `set_suppress_id()`, `set_suppress_location()`, `set_suppression_level()`, `va_send_message()`, and `warn()`.

8.1.3.13 operator=()

```
MessageHandler& jeod::MessageHandler::operator= (
    const MessageHandler & ) [private]
```

Not implemented.

8.1.3.14 process_add_suppressed_code()

```
virtual void jeod::MessageHandler::process_add_suppressed_code (
    const char * msg_code ) [inline], [protected], [virtual]
```

Add a message code to the set of messages that are to be suppressed.

The method `add_suppressed_code` relays the call to the message handler as a call to `process_add_suppressed_code`.

The default behavior is a no-op. Suppressing messages by the message code is an optional capability.

Parameters

<code>msg_code</code>	Message code to be suppressed
-----------------------	-------------------------------

Reimplemented in `jeod::SuppressedCodeMessageHandler`.

Definition at line 350 of file message_handler.hh.

Referenced by `add_suppressed_code()`.

8.1.3.15 process_clear_suppressed_codes()

```
virtual void jeod::MessageHandler::process_clear_suppressed_codes ( ) [inline], [protected],
[virtual]
```

Clear the set of suppressed message codes.

The method `clear_suppressed_codes` relays the call to the message handler as a call to `process_clear_suppressed_codes`.

As with `process_add_suppressed_code`, the default for this function is a no-op; suppressed codes are an optional capability.

Definition at line 378 of file `message_handler.hh`.

Referenced by `clear_suppressed_codes()`.

8.1.3.16 process_delete_suppressed_code()

```
virtual void jeod::MessageHandler::process_delete_suppressed_code (
    const char * msg_code ) [inline], [protected], [virtual]
```

Delete a message code from the set of suppressed message codes.

The method `delete_suppressed_code` relays the call to the message handler as a call to `process_delete_suppressed_code`.

As with `process_add_suppressed_code`, the default for this function is a no-op; suppressed codes are an optional capability.

Parameters

<i>msg_code</i>	Message code to be suppressed
-----------------	-------------------------------

Reimplemented in [jeod::SuppressedCodeMessageHandler](#).

Definition at line 365 of file `message_handler.hh`.

Referenced by `delete_suppressed_code()`.

8.1.3.17 process_message()

```
virtual void jeod::MessageHandler::process_message (
    int severity,
    const char * prefix,
    const char * file,
    unsigned int line,
    const char * msg_code,
```

```
const char * format,
va_list args ) const [protected], [pure virtual]
```

Generate the message.

All of the [send_message\(\)](#) methods relay the message to the message handler in the form of a call to [process_message\(\)](#).

An instantiable derived [MessageHandler](#) class must supply this function.

Parameters

<i>severity</i>	Severity level
<i>prefix</i>	Message prefix (e.g., Error)
<i>file</i>	Typically FILE
<i>line</i>	Typically LINE
<i>msg_code</i>	Message code
<i>format</i>	sprintf format
<i>args</i>	Arguments

Referenced by [debug\(\)](#), [error\(\)](#), [fail\(\)](#), [inform\(\)](#), [send_message\(\)](#), [va_send_message\(\)](#), and [warn\(\)](#).

8.1.3.18 register_contents()

```
virtual void jeod::MessageHandler::register_contents (
    void ) [inline], [virtual]
```

Register the checkpointable contents of the handler with the simulation interface.

The base [MessageHandler](#) has not such content.

Reimplemented in [jeod::SuppressedCodeMessageHandler](#).

Definition at line 243 of file [message_handler.hh](#).

8.1.3.19 send_message()

```
void jeod::MessageHandler::send_message (
    int severity,
    const char * prefix,
    const char * file,
    unsigned int line,
    const char * msg_code,
    const char * format,
    ... ) [static]
```

Generic variable arguments message interface.

This method gives the caller control over the severity level and over the message prefix. These are automatically generated in the standard set of [MessageHandler](#) interface methods.

Parameters

in	<i>severity</i>	Severity level
in	<i>prefix</i>	Message prefix (e.g., Error)
in	<i>file</i>	Typically FILE
in	<i>line</i>	Typically LINE
in	<i>msg_code</i>	Message code
in	<i>format</i>	sprintf format
in	...	sprintf arguments

Definition at line 275 of file message_handler.cc.

References handler, no_handler_error(), and process_message().

8.1.3.20 set_suppress_id()

```
void jeod::MessageHandler::set_suppress_id (
    bool suppress_id ) [static]
```

Set the suppress_id flag in the global message handler.

Parameters

in	<i>suppress_id</i>	New suppress id value
----	--------------------	-----------------------

Definition at line 457 of file message_handler.cc.

References handler, no_handler_error(), and suppress_id.

8.1.3.21 set_suppress_location()

```
void jeod::MessageHandler::set_suppress_location (
    bool suppress_location ) [static]
```

Set the suppress_location in the global message handler.

Parameters

in	<i>suppress_location</i>	New suppress_loc value
----	--------------------------	------------------------

Definition at line 502 of file message_handler.cc.

References handler, no_handler_error(), and suppress_location.

8.1.3.22 `set_suppression_level()`

```
void jeod::MessageHandler::set_suppression_level (
    unsigned int suppression_level ) [static]
```

Set the suppression level in the global message handler.

Parameters

in	<i>suppression_level</i>	New suppression level
----	--------------------------	-----------------------

Definition at line 345 of file `message_handler.cc`.

References `handler`, `no_handler_error()`, and `suppression_level`.

8.1.3.23 `va_send_message()`

```
void jeod::MessageHandler::va_send_message (
    int severity,
    const char * prefix,
    const char * file,
    unsigned int line,
    const char * msg_code,
    const char * format,
    va_list args ) [static]
```

Generic variable arguments message interface.

This method behaves similarly to [MessageHandler::send_message](#) except that the caller has already captured the variable arguments in the form of a `va_list`. Note that [MessageHandler::va_send_message](#) does not call `va_end` macro.

Parameters

in	<i>severity</i>	Severity level
in	<i>prefix</i>	Message prefix (e.g., Error)
in	<i>file</i>	Typically FILE
in	<i>line</i>	Typically LINE
in	<i>msg_code</i>	Message code
in	<i>format</i>	sprintf format
in, out	<i>args</i>	Varargs stack

Definition at line 316 of file `message_handler.cc`.

References `handler`, `no_handler_error()`, and `process_message()`.

8.1.3.24 warn()

```
void jeod::MessageHandler::warn (
    const char * file,
    unsigned int line,
    const char * msg_code,
    const char * format,
    ... ) [static]
```

Generate a message with severity [MessageHandler::Warning](#).

Warnings represent situations where the model developer had to make some assumptions to recover from what would otherwise be an erroneous condition. The recovery based on those assumptions does not necessarily invalidate the simulation results.

Parameters

in	<i>file</i>	Typically FILE
in	<i>line</i>	Typically LINE
in	<i>msg_code</i>	Message code
in	<i>format</i>	sprintf format
in	...	sprintf arguments

Definition at line 166 of file message_handler.cc.

References [handler](#), [no_handler_error\(\)](#), [process_message\(\)](#), and [Warning](#).

8.1.4 Friends And Related Function Documentation

8.1.4.1 init_attrjeod_MessageHandler

```
void init_attrjeod_MessageHandler ( ) [friend]
```

8.1.4.2 InputProcessor

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

Definition at line 109 of file message_handler.hh.

8.1.5 Field Documentation

8.1.5.1 Debug

```
const int jeod::MessageHandler::Debug = 999 [static]
```

The severity value passed by the static public [MessageHandler::debug](#) method to the derived class `process_message` method.

This is set to 999 in the implementation. The intent is to summarize to the user of some event that the user requested did indeed transpire. Ideally, JEOD code, particularly initialization code, will be peppered with calls to `MessageHandler::debug.trick_io(*o) trick_units(-)`

Definition at line 302 of file `message_handler.hh`.

Referenced by `debug()`.

8.1.5.2 Error

```
const int jeod::MessageHandler::Error = 0 [static]
```

The severity value passed by the static public [MessageHandler::error](#) method to the derived class `process_message` method.

This is set to 0 in the implementation, representing the most severe non-fatal error.

Non-negative severity levels indicate non-fatal conditions for which messages might nonetheless need to be generated, depending on the value of the user-settable `suppression_level.trick_io(*o) trick_units(-)`

Definition at line 276 of file `message_handler.hh`.

Referenced by `error()`.

8.1.5.3 Failure

```
const int jeod::MessageHandler::Failure = -1 [static]
```

The severity value passed by the static public [MessageHandler::fail](#) method to the derived class `process_message` method.

This is set to -1 in the implementation, representing a fatal error.

A valid implementation of the `process_message` method must treat negative severity levels as fatal; they must not return to the calling procedure. In other words, failures eventually result in a call to `exit.trick_io(*o) trick_units(-)`

Definition at line 264 of file `message_handler.hh`.

Referenced by `fail()`.

8.1.5.4 handler

```
MessageHandler * jeod::MessageHandler::handler = NULL [static], [protected]
```

The [MessageHandler](#) instance that generates messages.

The static [MessageHandler](#) functions invoked by various models pass the message on to this instance in the form of a call to `process_message.trick_io(*o) trick_units(-)`

Definition at line 389 of file `message_handler.hh`.

Referenced by `add_suppressed_code()`, `clear_suppressed_codes()`, `debug()`, `delete_suppressed_code()`, `error()`, `fail()`, `get_suppress_id()`, `get_suppress_location()`, `get_suppression_level()`, `inform()`, `MessageHandler()`, `send_↔message()`, `set_suppress_id()`, `set_suppress_location()`, `set_suppression_level()`, `va_send_message()`, `warn()`, and `~MessageHandler()`.

8.1.5.5 Notice

```
const int jeod::MessageHandler::Notice = 99 [static]
```

The severity value passed by the static public [MessageHandler::inform](#) method to the derived class `process_↔message` method.

This is set to 99 in the implementation. The intent is to indicate a non-error condition that might be worthy of a user notification.`trick_io(*o) trick_units(-)`

Definition at line 292 of file `message_handler.hh`.

Referenced by `inform()`.

8.1.5.6 suppress_id

```
bool jeod::MessageHandler::suppress_id [protected]
```

This flag indicates whether the message ID is printed for unsuppressed messages.

The ID is not printed if this flag is set to true. The message ID is always printed for fatal errors.

Default value: `false.trick_units(-)`

Definition at line 412 of file `message_handler.hh`.

Referenced by `get_suppress_id()`, and `set_suppress_id()`.

8.1.5.7 suppress_location

```
bool jeod::MessageHandler::suppress_location [protected]
```

This flag indicates whether the message source file and line number printed for unsuppressed messages.

The location is not printed if this flag is set to true. The message location is always printed for fatal errors.

Default value: `false`.`trick_units(-)`

Definition at line 422 of file `message_handler.hh`.

Referenced by `get_suppress_location()`, and `set_suppress_location()`.

8.1.5.8 suppression_level

```
unsigned int jeod::MessageHandler::suppression_level [protected]
```

All messages have an associated severity level, with increasingly positive values indicating warnings of decreasing severity.

Fatal errors have a negative severity level. Messages whose severity exceeds the value of the global message handler's `suppression_level` are suppressed. Note that fatal errors and severe errors cannot be suppressed.

Default value: `MessageHandler::Warning` (warnings and non-fatal errors).`trick_units(-)`

Definition at line 403 of file `message_handler.hh`.

Referenced by `get_suppression_level()`, `jeod::SuppressedCodeMessageHandler::message_is_to_be_printed()`, and `set_suppression_level()`.

8.1.5.9 Warning

```
const int jeod::MessageHandler::Warning = 9 [static]
```

The severity value passed by the static public `MessageHandler::warn` method to the derived class `process_message` method.

This is set to 9 in the implementation. The intent is to indicate a condition that might indicate that results are suspect.`trick_io(*o) trick_units(-)`

Definition at line 284 of file `message_handler.hh`.

Referenced by `warn()`.

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

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

8.2 jeod::MessageMessages Class Reference

Specifies the message IDs used in the message handler model.

```
#include <message_messages.hh>
```

Static Public Attributes

- static char const * [singleton_error](#) = "utils/message/" "singleton_error"
Error issued when multiple instance of a class that should be a singleton are created or when no such instance exists (but should).

Private Member Functions

- [MessageMessages](#) (void)
Not implemented.
- [MessageMessages](#) (const [MessageMessages](#) &)
Not implemented.
- [MessageMessages](#) & [operator=](#) (const [MessageMessages](#) &)
Not implemented.

8.2.1 Detailed Description

Specifies the message IDs used in the message handler model.

Definition at line 73 of file message_messages.hh.

8.2.2 Constructor & Destructor Documentation

8.2.2.1 MessageMessages() [1/2]

```
jeod::MessageMessages::MessageMessages (
    void ) [private]
```

Not implemented.

8.2.2.2 MessageMessages() [2/2]

```
jeod::MessageMessages::MessageMessages (
    const MessageMessages & ) [private]
```

Not implemented.

8.2.3 Member Function Documentation

8.2.3.1 operator=()

```
MessageMessages& jeod::MessageMessages::operator= (
    const MessageMessages & ) [private]
```

Not implemented.

8.2.4 Field Documentation

8.2.4.1 singleton_error

```
char const * jeod::MessageMessages::singleton_error = "utils/message/" "singleton_error" [static]
```

Error issued when multiple instance of a class that should be a singleton are created or when no such instance exists (but should).

trick_units(-)

Definition at line 82 of file message_messages.hh.

Referenced by jeod::MessageHandler::MessageHandler().

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

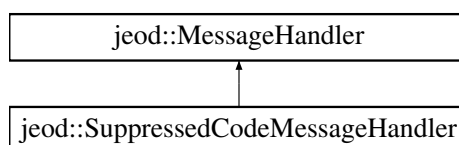
- [message_messages.hh](#)
- [message_messages.cc](#)

8.3 jeod::SuppressedCodeMessageHandler Class Reference

Adds the capability to suppress messages by their message code to the base [MessageHandler](#) class.

```
#include <suppressed_code_message_handler.hh>
```

Inheritance diagram for jeod::SuppressedCodeMessageHandler:



Public Member Functions

- [SuppressedCodeMessageHandler](#) (void)

Default constructor.

- virtual [~SuppressedCodeMessageHandler](#) (void)

Destructor.

Protected Member Functions

- virtual void [register_contents](#) (void)

Register the [MessageHandler](#)'s checkpointable contents.

- virtual void [deregister_contents](#) (void)

Deregister the [MessageHandler](#)'s checkpointable contents.

- virtual void [process_add_suppressed_code](#) (const char *msg_code)

Add a message code to the set of messages that are to be suppressed.

- virtual void [process_delete_suppressed_code](#) (const char *msg_code)

Delete a message code from the set of suppressed message codes.

- virtual void [process_clear_suppressed_code](#) (void)

Clear the set of messages that are to be suppressed.

- bool [message_is_to_be_printed](#) (int severity, const char *msg_code) const

Determine whether output for a message is to be printed.

Protected Attributes

- JeodPrimitiveSet< std::string >::type [suppressed_codes](#)

The set of message code that are to be suppressed.

Private Member Functions

- [SuppressedCodeMessageHandler](#) (const [SuppressedCodeMessageHandler](#) &)

Not implemented.

- [SuppressedCodeMessageHandler](#) & operator= (const [SuppressedCodeMessageHandler](#) &)

Not implemented.

Friends

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

Additional Inherited Members

8.3.1 Detailed Description

Adds the capability to suppress messages by their message code to the base [MessageHandler](#) class.

Definition at line 83 of file `suppressed_code_message_handler.hh`.

8.3.2 Constructor & Destructor Documentation

8.3.2.1 SuppressedCodeMessageHandler() [1/2]

```
jeod::SuppressedCodeMessageHandler::SuppressedCodeMessageHandler (  
    void ) [inline]
```

Default constructor.

Definition at line 102 of file `suppressed_code_message_handler.hh`.

8.3.2.2 ~SuppressedCodeMessageHandler()

```
virtual jeod::SuppressedCodeMessageHandler::~~SuppressedCodeMessageHandler (  
    void ) [inline], [virtual]
```

Destructor.

Definition at line 107 of file `suppressed_code_message_handler.hh`.

8.3.2.3 SuppressedCodeMessageHandler() [2/2]

```
jeod::SuppressedCodeMessageHandler::SuppressedCodeMessageHandler (  
    const SuppressedCodeMessageHandler & ) [private]
```

Not implemented.

8.3.3 Member Function Documentation

8.3.3.1 deregister_contents()

```
void jeod::SuppressedCodeMessageHandler::deregister_contents (
    void ) [protected], [virtual]
```

Deregister the [MessageHandler](#)'s checkpointable contents.

Reimplemented from [jeod::MessageHandler](#).

Definition at line 60 of file `suppressed_code_message_handler.cc`.

References `suppressed_codes`.

8.3.3.2 message_is_to_be_printed()

```
bool jeod::SuppressedCodeMessageHandler::message_is_to_be_printed (
    int severity,
    const char * msg_code ) const [inline], [protected]
```

Determine whether output for a message is to be printed.

Returns

True => print message

Parameters

in	<i>severity</i>	Severity level
in	<i>msg_code</i>	Message code

Definition at line 159 of file `suppressed_code_message_handler.hh`.

References `suppressed_codes`, and `jeod::MessageHandler::suppression_level`.

8.3.3.3 operator=()

```
SuppressedCodeMessageHandler& jeod::SuppressedCodeMessageHandler::operator= (
    const SuppressedCodeMessageHandler & ) [private]
```

Not implemented.

8.3.3.4 process_add_suppressed_code()

```
virtual void jeod::SuppressedCodeMessageHandler::process_add_suppressed_code (
    const char * msg_code ) [inline], [protected], [virtual]
```

Add a message code to the set of messages that are to be suppressed.

Parameters

in	<i>msg_code</i>	Message code to be suppressed
----	-----------------	-------------------------------

Reimplemented from [jeod::MessageHandler](#).

Definition at line 125 of file `suppressed_code_message_handler.hh`.

References `suppressed_codes`.

8.3.3.5 process_clear_suppressed_code()

```
virtual void jeod::SuppressedCodeMessageHandler::process_clear_suppressed_code (
    void ) [inline], [protected], [virtual]
```

Clear the set of messages that are to be suppressed.

Definition at line 146 of file `suppressed_code_message_handler.hh`.

References `suppressed_codes`.

8.3.3.6 process_delete_suppressed_code()

```
virtual void jeod::SuppressedCodeMessageHandler::process_delete_suppressed_code (
    const char * msg_code ) [inline], [protected], [virtual]
```

Delete a message code from the set of suppressed message codes.

Parameters

in	<i>msg_code</i>	Message code to be unsuppressed
----	-----------------	---------------------------------

Reimplemented from [jeod::MessageHandler](#).

Definition at line 136 of file `suppressed_code_message_handler.hh`.

References `suppressed_codes`.

8.3.3.7 register_contents()

```
void jeod::SuppressedCodeMessageHandler::register_contents (
    void ) [protected], [virtual]
```

Register the [MessageHandler](#)'s checkpointable contents.

Reimplemented from [jeod::MessageHandler](#).

Definition at line 48 of file `suppressed_code_message_handler.cc`.

References `suppressed_codes`.

8.3.4 Friends And Related Function Documentation

8.3.4.1 `init_attrjeod__SuppressedCodeMessageHandler`

```
void init_attrjeod__SuppressedCodeMessageHandler ( ) [friend]
```

8.3.4.2 `InputProcessor`

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

Definition at line 84 of file `suppressed_code_message_handler.hh`.

8.3.5 Field Documentation

8.3.5.1 `suppressed_codes`

```
JeodPrimitiveSet<std::string>::type jeod::SuppressedCodeMessageHandler::suppressed_codes  
[protected]
```

The set of message code that are to be suppressed.

`trick_io(**)`

Definition at line 175 of file `suppressed_code_message_handler.hh`.

Referenced by `deregister_contents()`, `message_is_to_be_printed()`, `process_add_suppressed_code()`, `process_clear_suppressed_code()`, `process_delete_suppressed_code()`, and `register_contents()`.

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

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

Chapter 9

File Documentation

9.1 `class_declarations.hh` File Reference

Forward declarations of classes defined in this module.

Namespaces

- [jeod](#)
Namespace jeod.

9.1.1 Detailed Description

Forward declarations of classes defined in this module.

9.2 `make_message_code.hh` File Reference

Define JEOD_MAKE_MESSAGE_CODE.

Namespaces

- [jeod](#)
Namespace jeod.

Macros

- `#define JEOD_MAKE_MESSAGE_CODE(cname, path, id) char const * cname::id = path #id`
Shortcut macro to define the static member `cname::id` as the catenation of the path and the stringified id.

9.2.1 Detailed Description

Define JEOD_MAKE_MESSAGE_CODE.

9.2.2 Macro Definition Documentation

9.2.2.1 JEOD_MAKE_MESSAGE_CODE

```
#define JEOD_MAKE_MESSAGE_CODE(  
    cname,  
    path,  
    id ) char const * cname::id = path #id
```

Shortcut macro to define the static member `cname::id` as the catenation of the `path` and the stringified `id`.

Parameters

in	<i>cname</i>	The name of the message class.
in	<i>path</i>	The path from \$JEOD_HOME/models to the model in question. This must be a <code>char*</code> string and should terminate in a <code>'/'</code> .
in	<i>id</i>	The static member data name to be assigned.

Definition at line 70 of file `make_message_code.hh`.

9.3 message_handler.cc File Reference

Define member functions for the class `MessageHandler`.

```
#include <cstdarg>  
#include <cstddef>  
#include <cstdio>  
#include <cstdlib>  
#include "../include/message_handler.hh"  
#include "../include/message_messages.hh"
```

Namespaces

- [jeod](#)
Namespace jeod.

9.3.1 Detailed Description

Define member functions for the class `MessageHandler`.

9.4 message_handler.hh File Reference

Define the class MessageHandler, the base class for generating messages.

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

Data Structures

- class [jeod::MessageHandler](#)
The base class for generating JEOD messages.

Namespaces

- [jeod](#)
Namespace jeod.

9.4.1 Detailed Description

Define the class MessageHandler, the base class for generating messages.

9.5 message_messages.cc File Reference

Implement the class MessageMessages.

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

Namespaces

- [jeod](#)
Namespace jeod.

Macros

- #define [MAKE_MESSAGE_MESSAGE_CODE](#)(id) [JEOD_MAKE_MESSAGE_CODE](#)(MessageMessages, "utils/message/", id)

9.5.1 Detailed Description

Implement the class MessageMessages.

9.5.2 Macro Definition Documentation

9.5.2.1 MAKE_MESSAGE_MESSAGE_CODE

```
#define MAKE_MESSAGE_MESSAGE_CODE(  
    id ) JEOD_MAKE_MESSAGE_CODE(MessageMessages, "utils/message/", id)
```

Definition at line 39 of file message_messages.cc.

9.6 message_messages.hh File Reference

Define the class MessageMessages.

Data Structures

- class [jeod::MessageMessages](#)
Specifies the message IDs used in the message handler model.

Namespaces

- [jeod](#)
Namespace jeod.

9.6.1 Detailed Description

Define the class MessageMessages.

9.7 suppressed_code_message_handler.cc File Reference

Define member functions for the class SuppressedCodeMessageHandler.

```
#include <cstdarg>  
#include <cstddef>  
#include <cstdio>  
#include <cstdlib>  
#include "utils/memory/include/jeod_alloc.hh"  
#include "../include/suppressed_code_message_handler.hh"
```

Namespaces

- [jeod](#)
Namespace jeod.

9.7.1 Detailed Description

Define member functions for the class SuppressedCodeMessageHandler.

9.8 suppressed_code_message_handler.hh File Reference

Define the class SuppressedCodeMessageHandler, which adds the capability to suppress messages by their message code.

```
#include "utils/container/include/primitive_set.hh"
#include "utils/sim_interface/include/jeod_class.hh"
#include "message_handler.hh"
```

Data Structures

- class [jeod::SuppressedCodeMessageHandler](#)

Adds the capability to suppress messages by their message code to the base [MessageHandler](#) class.

Namespaces

- [jeod](#)

Namespace jeod.

9.8.1 Detailed Description

Define the class SuppressedCodeMessageHandler, which adds the capability to suppress messages by their message code.

This capability cannot be a part of the base MessageHandler class because that base class needs to stand on its own.

Index

- ~MessageHandler
 - jeod::MessageHandler, [20](#)
- ~SuppressedCodeMessageHandler
 - jeod::SuppressedCodeMessageHandler, [38](#)
- add_suppressed_code
 - jeod::MessageHandler, [21](#)
- class_declarations.hh, [43](#)
- clear_suppressed_codes
 - jeod::MessageHandler, [21](#)
- Debug
 - jeod::MessageHandler, [31](#)
- debug
 - jeod::MessageHandler, [22](#)
- delete_suppressed_code
 - jeod::MessageHandler, [22](#)
- deregister_contents
 - jeod::MessageHandler, [22](#)
 - jeod::SuppressedCodeMessageHandler, [38](#)
- Error
 - jeod::MessageHandler, [32](#)
- error
 - jeod::MessageHandler, [23](#)
- fail
 - jeod::MessageHandler, [23](#)
- Failure
 - jeod::MessageHandler, [32](#)
- get_suppress_id
 - jeod::MessageHandler, [24](#)
- get_suppress_location
 - jeod::MessageHandler, [24](#)
- get_suppression_level
 - jeod::MessageHandler, [24](#)
- handler
 - jeod::MessageHandler, [32](#)
- inform
 - jeod::MessageHandler, [25](#)
- init_attrjeod__MessageHandler
 - jeod::MessageHandler, [31](#)
- init_attrjeod__SuppressedCodeMessageHandler
 - jeod::SuppressedCodeMessageHandler, [41](#)
- InputProcessor
 - jeod::MessageHandler, [31](#)
 - jeod::SuppressedCodeMessageHandler, [41](#)
- JEOD_MAKE_MESSAGE_CODE
 - make_message_code.hh, [44](#)
- jeod, [15](#)
- jeod::MessageHandler, [17](#)
 - ~MessageHandler, [20](#)
 - add_suppressed_code, [21](#)
 - clear_suppressed_codes, [21](#)
 - Debug, [31](#)
 - debug, [22](#)
 - delete_suppressed_code, [22](#)
 - deregister_contents, [22](#)
 - Error, [32](#)
 - error, [23](#)
 - fail, [23](#)
 - Failure, [32](#)
 - get_suppress_id, [24](#)
 - get_suppress_location, [24](#)
 - get_suppression_level, [24](#)
 - handler, [32](#)
 - inform, [25](#)
 - init_attrjeod__MessageHandler, [31](#)
 - InputProcessor, [31](#)
 - MessageHandler, [20](#), [21](#)
 - no_handler_error, [25](#)
 - Notice, [33](#)
 - operator=, [26](#)
 - process_add_suppressed_code, [26](#)
 - process_clear_suppressed_codes, [26](#)
 - process_delete_suppressed_code, [27](#)
 - process_message, [27](#)
 - register_contents, [28](#)
 - send_message, [28](#)
 - set_suppress_id, [29](#)
 - set_suppress_location, [29](#)
 - set_suppression_level, [29](#)
 - suppress_id, [33](#)
 - suppress_location, [33](#)
 - suppression_level, [34](#)
 - va_send_message, [30](#)
 - warn, [30](#)
 - Warning, [34](#)
- jeod::MessageMessages, [35](#)
 - MessageMessages, [35](#)
 - operator=, [36](#)
 - singleton_error, [36](#)
- jeod::SuppressedCodeMessageHandler, [36](#)
 - ~SuppressedCodeMessageHandler, [38](#)
 - deregister_contents, [38](#)
 - init_attrjeod__SuppressedCodeMessageHandler,

- 41
- InputProcessor, 41
- message_is_to_be_printed, 39
- operator=, 39
- process_add_suppressed_code, 39
- process_clear_suppressed_code, 40
- process_delete_suppressed_code, 40
- register_contents, 40
- suppressed_codes, 41
- SuppressedCodeMessageHandler, 38
- MAKE_MESSAGE_MESSAGE_CODE
 - message_messages.cc, 46
- make_message_code.hh, 43
- JEOD_MAKE_MESSAGE_CODE, 44
- Message, 13
- message_handler.cc, 44
- message_handler.hh, 45
- message_is_to_be_printed
 - jeod::SuppressedCodeMessageHandler, 39
- message_messages.cc, 45
 - MAKE_MESSAGE_MESSAGE_CODE, 46
- message_messages.hh, 46
- MessageHandler
 - jeod::MessageHandler, 20, 21
- MessageMessages
 - jeod::MessageMessages, 35
- Models, 11
- no_handler_error
 - jeod::MessageHandler, 25
- Notice
 - jeod::MessageHandler, 33
- operator=
 - jeod::MessageHandler, 26
 - jeod::MessageMessages, 36
 - jeod::SuppressedCodeMessageHandler, 39
- process_add_suppressed_code
 - jeod::MessageHandler, 26
 - jeod::SuppressedCodeMessageHandler, 39
- process_clear_suppressed_code
 - jeod::SuppressedCodeMessageHandler, 40
- process_clear_suppressed_codes
 - jeod::MessageHandler, 26
- process_delete_suppressed_code
 - jeod::MessageHandler, 27
 - jeod::SuppressedCodeMessageHandler, 40
- process_message
 - jeod::MessageHandler, 27
- register_contents
 - jeod::MessageHandler, 28
 - jeod::SuppressedCodeMessageHandler, 40
- send_message
 - jeod::MessageHandler, 28
- set_suppress_id
 - jeod::MessageHandler, 29
- set_suppress_location
 - jeod::MessageHandler, 29
- set_suppression_level
 - jeod::MessageHandler, 29
- singleton_error
 - jeod::MessageMessages, 36
- suppress_id
 - jeod::MessageHandler, 33
- suppress_location
 - jeod::MessageHandler, 33
- suppressed_code_message_handler.cc, 46
- suppressed_code_message_handler.hh, 47
- suppressed_codes
 - jeod::SuppressedCodeMessageHandler, 41
- SuppressedCodeMessageHandler
 - jeod::SuppressedCodeMessageHandler, 38
- suppression_level
 - jeod::MessageHandler, 34
- Utils, 12
- va_send_message
 - jeod::MessageHandler, 30
- warn
 - jeod::MessageHandler, 30
- Warning
 - jeod::MessageHandler, 34