

PyRTAI

Generated by Doxygen 1.7.5

Wed Jan 18 2012 17:30:42



# Contents

<b>1</b>	<b>Namespace Index</b>	<b>1</b>
1.1	Packages . . . . .	1
<b>2</b>	<b>Class Index</b>	<b>3</b>
2.1	Class Hierarchy . . . . .	3
<b>3</b>	<b>Class Index</b>	<b>5</b>
3.1	Class List . . . . .	5
<b>4</b>	<b>File Index</b>	<b>7</b>
4.1	File List . . . . .	7
<b>5</b>	<b>Namespace Documentation</b>	<b>9</b>
5.1	custom_exceptions Namespace Reference . . . . .	9
5.1.1	Detailed Description . . . . .	9
5.2	pyrtai Namespace Reference . . . . .	9
5.2.1	Detailed Description . . . . .	9
5.3	pyrtai::configurator Namespace Reference . . . . .	10
5.4	pyrtai::connector Namespace Reference . . . . .	10
5.5	pyrtai::custom_exceptions Namespace Reference . . . . .	10
5.6	pyrtai::data_collector Namespace Reference . . . . .	10
5.7	pyrtai::target Namespace Reference . . . . .	10
5.8	pyrtai::utility Namespace Reference . . . . .	11
5.8.1	Function Documentation . . . . .	11
5.8.1.1	getStateFromResponse . . . . .	11
<b>6</b>	<b>Class Documentation</b>	<b>13</b>

6.1	<a href="#">pyrtai::configurator::Configurator Class Reference</a>	13
6.1.1	<a href="#">Detailed Description</a>	14
6.1.2	<a href="#">Constructor &amp; Destructor Documentation</a>	14
6.1.2.1	<a href="#">__init__</a>	14
6.1.3	<a href="#">Member Function Documentation</a>	14
6.1.3.1	<a href="#">checkIfMasterOrSlave</a>	14
6.1.3.2	<a href="#">getAttr</a>	15
6.1.3.3	<a href="#">getUrl</a>	15
6.1.3.4	<a href="#">setAttr</a>	15
6.1.4	<a href="#">Member Data Documentation</a>	15
6.1.4.1	<a href="#">address</a>	15
6.1.4.2	<a href="#">config</a>	16
6.1.4.3	<a href="#">host</a>	16
6.1.4.4	<a href="#">is_slave_connection</a>	16
6.1.4.5	<a href="#">port</a>	16
6.1.4.6	<a href="#">protocol</a>	16
6.1.4.7	<a href="#">session_id</a>	16
6.1.4.8	<a href="#">slave_data_port</a>	16
6.1.4.9	<a href="#">slave_port</a>	16
6.1.4.10	<a href="#">target_name</a>	17
6.2	<a href="#">pyrtai::connector::Connector Class Reference</a>	17
6.2.1	<a href="#">Detailed Description</a>	18
6.2.2	<a href="#">Constructor &amp; Destructor Documentation</a>	18
6.2.2.1	<a href="#">__init__</a>	18
6.2.3	<a href="#">Member Function Documentation</a>	18
6.2.3.1	<a href="#">callMethod</a>	18
6.2.3.2	<a href="#">closeSession</a>	18
6.2.3.3	<a href="#">Connection_Request</a>	19
6.2.3.4	<a href="#">connectSlave</a>	19
6.2.3.5	<a href="#">disconnect</a>	19
6.2.3.6	<a href="#">getInfo</a>	20
6.2.3.7	<a href="#">stopServer</a>	20
6.2.4	<a href="#">Member Data Documentation</a>	20
6.2.4.1	<a href="#">connection</a>	20

6.3	<a href="#">pyrtai::data_collector::DataCollector Class Reference</a>	20
6.3.1	<a href="#">Constructor &amp; Destructor Documentation</a>	21
6.3.1.1	<a href="#">__init__</a>	21
6.3.2	<a href="#">Member Data Documentation</a>	21
6.3.2.1	<a href="#">input</a>	21
6.3.2.2	<a href="#">output</a>	21
6.4	<a href="#">pyrtai::custom_exceptions::RtaiConfigError Class Reference</a>	21
6.4.1	<a href="#">Detailed Description</a>	22
6.4.2	<a href="#">Constructor &amp; Destructor Documentation</a>	22
6.4.2.1	<a href="#">__init__</a>	22
6.4.3	<a href="#">Member Function Documentation</a>	22
6.4.3.1	<a href="#">__str__</a>	22
6.4.4	<a href="#">Member Data Documentation</a>	22
6.4.4.1	<a href="#">expr</a>	22
6.4.4.2	<a href="#">msg</a>	22
6.5	<a href="#">pyrtai::custom_exceptions::RtaiError Class Reference</a>	23
6.5.1	<a href="#">Detailed Description</a>	23
6.6	<a href="#">pyrtai::custom_exceptions::RtaiXmlRpcError Class Reference</a>	23
6.6.1	<a href="#">Detailed Description</a>	24
6.6.2	<a href="#">Constructor &amp; Destructor Documentation</a>	24
6.6.2.1	<a href="#">__init__</a>	24
6.6.3	<a href="#">Member Function Documentation</a>	24
6.6.3.1	<a href="#">__str__</a>	24
6.6.4	<a href="#">Member Data Documentation</a>	24
6.6.4.1	<a href="#">expr</a>	24
6.6.4.2	<a href="#">msg</a>	24
6.7	<a href="#">pyrtai::target::Target Class Reference</a>	24
6.7.1	<a href="#">Constructor &amp; Destructor Documentation</a>	25
6.7.1.1	<a href="#">__init__</a>	25
6.7.2	<a href="#">Member Function Documentation</a>	25
6.7.2.1	<a href="#">close</a>	25
6.7.2.2	<a href="#">connect</a>	26
6.7.2.3	<a href="#">disconnect</a>	26
6.7.2.4	<a href="#">getSlave</a>	26

6.7.2.5	<a href="#">stopServer</a>	26
6.7.3	<a href="#">Member Data Documentation</a>	26
6.7.3.1	<a href="#">config</a>	26
6.7.3.2	<a href="#">connection</a>	27
6.7.3.3	<a href="#">slave</a>	27
<b>7</b>	<b>File Documentation</b>	<b>29</b>
7.1	<a href="#">/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/__init__.py - File Reference</a>	29
7.2	<a href="#">/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/configurator.py File Reference</a>	29
7.3	<a href="#">/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/connector.py File Reference</a>	29
7.4	<a href="#">/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/custom_exceptions.py File Reference</a>	30
7.5	<a href="#">/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/data_collector.py File Reference</a>	30
7.6	<a href="#">/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/target.py - File Reference</a>	31
7.7	<a href="#">/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/utility.py File Reference</a>	31

# Chapter 1

## Namespace Index

### 1.1 Packages

Here are the packages with brief descriptions (if available):

<a href="#">custom_exceptions</a>	Custom exceptions for the library . . . . .	9
<a href="#">pyrtai</a>	This package handles the configurations . . . . .	9
<a href="#">pyrtai::configurator</a>	. . . . .	10
<a href="#">pyrtai::connector</a>	. . . . .	10
<a href="#">pyrtai::custom_exceptions</a>	. . . . .	10
<a href="#">pyrtai::data_collector</a>	. . . . .	10
<a href="#">pyrtai::target</a>	. . . . .	10
<a href="#">pyrtai::utility</a>	. . . . .	11





## Chapter 2

# Class Index

### 2.1 Class Hierarchy

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

pyrtai::configurator::Configurator . . . . .	13
pyrtai::connector::Connector . . . . .	17
pyrtai::data_collector::DataCollector . . . . .	20
pyrtai::custom_exceptions::RtaiError . . . . .	23
pyrtai::custom_exceptions::RtaiConfigError . . . . .	21
pyrtai::custom_exceptions::RtaiXmlRpcError . . . . .	23
pyrtai::target::Target . . . . .	24



## Chapter 3

# Class Index

### 3.1 Class List

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

<a href="#">pyrtai::configurator::Configurator</a>	
<a href="#">Configurator</a> class handles config values . . . . .	13
<a href="#">pyrtai::connector::Connector</a>	
Connect to the RTAI server and manage responses and basic operations on it . . . . .	17
<a href="#">pyrtai::data_collector::DataCollector</a> . . . . .	20
<a href="#">pyrtai::custom_exceptions::RtaiConfigError</a>	
Exception raised for errors involving connections . . . . .	21
<a href="#">pyrtai::custom_exceptions::RtaiError</a>	
Base class for exceptions in this module . . . . .	23
<a href="#">pyrtai::custom_exceptions::RtaiXmlRpcError</a>	
Exception raised for errors involving connections . . . . .	23
<a href="#">pyrtai::target::Target</a> . . . . .	24



## Chapter 4

# File Index

### 4.1 File List

Here is a list of all files with brief descriptions:

/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/	<a href="#">__init__.py</a>	. . . .	29
/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/	<a href="#">configurator.py</a>	. .	29
/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/	<a href="#">connector.py</a>	. . .	29
/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/	<a href="#">custom_exceptions.py</a>	. . . . .	30
/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/	<a href="#">data_collector.py</a>	.	30
/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/	<a href="#">target.py</a>	. . . . .	31
/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/	<a href="#">utility.py</a>	. . . . .	31



## Chapter 5

# Namespace Documentation

### 5.1 `custom_exceptions` Namespace Reference

Custom exceptions for the library.

#### 5.1.1 Detailed Description

Custom exceptions for the library.

### 5.2 `pyrtai` Namespace Reference

This package handles the configurations.

#### Packages

- namespace [configurator](#)
- namespace [connector](#)
- namespace [custom\\_exceptions](#)
- namespace [data\\_collector](#)
- namespace [target](#)
- namespace [utility](#)

#### 5.2.1 Detailed Description

This package handles the configurations. Various useful functions.

Represents the target server (master or slave)

Handles the connection to the RTAI server.

See also

[pyrtai::connector::Connector](#)  
[pyrtai::target::Target](#)

## 5.3 pyrtai::configurator Namespace Reference

### Classes

- class [Configurator](#)  
*[Configurator](#) class handles config values.*

## 5.4 pyrtai::connector Namespace Reference

### Classes

- class [Connector](#)  
*The [Connector](#) class connect to the RTAI server and manage responses and basic operations on it.*

## 5.5 pyrtai::custom\_exceptions Namespace Reference

### Classes

- class [RtaiError](#)  
*Base class for exceptions in this module.*
- class [RtaiConfigError](#)  
*Exception raised for errors involving connections.*
- class [RtaiXmlRpcError](#)  
*Exception raised for errors involving connections.*

## 5.6 pyrtai::data\_collector Namespace Reference

### Classes

- class [DataCollector](#)

## 5.7 pyrtai::target Namespace Reference



## Classes

- class [Target](#)

## 5.8 pyrtai::utility Namespace Reference

### Functions

- def [getStateFromResponse](#)

*Parses an integer state value which, converted into binary value, represents 4 states.*

### 5.8.1 Function Documentation

#### 5.8.1.1 def pyrtai::utility::getStateFromResponse ( *int\_state* )

Parses an integer state value which, converted into binary value, represents 4 states.

The 4 states are (array indexes): 0 - Target running 1 - Target connected 2 - Target exists 3 - Error occurred

#### Parameters

<i>int_state</i> ,:	The integer value of the state
---------------------	--------------------------------

#### Returns

: An hash containing explained states

Definition at line 13 of file utility.py.



## Chapter 6

# Class Documentation

### 6.1 pyrtai::configurator::Configurator Class Reference

[Configurator](#) class handles config values.

#### Public Member Functions

- def [\\_\\_init\\_\\_](#)  
*The constructor reads the config file, which can be passed as param.*
- def [setAttr](#)  
*Sets member vars.*
- def [getAttr](#)  
*Gets member vars from `__dict__` or from the config file if not previously assigned.*
- def [checkIfMasterOrSlave](#)  
*Assigns the `self.is_slave_connection` var checking if this object describes a master or slave connection.*
- def [getUrl](#)  
*Get the full Url to connect to.*

#### Public Attributes

- [config](#)  
*The ConfigParser instance.*
- [protocol](#)  
*The protocol to use to connect to the server (HTTP/HTTPS)*
- [address](#)  
*The address to connect to.*
- [port](#)  
*The port to connect to.*

- [host](#)  
*The host to connect to.*
- [session\\_id](#)  
*The session name.*
- [slave\\_port](#)  
*The port to be used to connect to slave server.*
- [slave\\_data\\_port](#)  
*The port to be used to connect to slave data stream.*
- [target\\_name](#)  
*The name of the target server.*
- [is\\_slave\\_connection](#)  
*Booblean value which indicates if it's a slave connection.*

### 6.1.1 Detailed Description

[Configurator](#) class handles config values.

The default action is to check if the value exists then return it or return the value parsed from the config file otherwise. The default file is defaults.cfg but a different one can be passed as parameter to the constructor.

### 6.1.2 Constructor & Destructor Documentation

6.1.2.1 `def pyrtai::configurator::Configurator::__init__( self, config_file = None )`

The constructor reads the config file, which can be passed as param.

Raises a `RtaiConfigError` if the file can't be found.

#### Parameters

<code>config_file</code>	The path to the config file. If not specified is used defaults.cfg in the script directory
--------------------------	--

Definition at line 48 of file configurator.py.

### 6.1.3 Member Function Documentation

6.1.3.1 `def pyrtai::configurator::Configurator::checkIfMasterOrSlave ( self )`

Assigns the `self.is_slave_connection` var checking if this object describes a master or slave connection.

The check consists in comparing the connection port assigned and the one read from the config file (which is the port of the master connection)

Definition at line 99 of file configurator.py.

### 6.1.3.2 `def pyrtai::configurator::Configurator::getAttr ( self, key )`

Gets member vars from `__dict__` or from the config file if not previously assigned.

#### Parameters

<i>key</i>	The name of the member var to retrieve
------------	--

#### Returns

The value of the var. None if not present in the `__dict__` or in the config file

Definition at line 82 of file `configurator.py`.

### 6.1.3.3 `def pyrtai::configurator::Configurator::getUrl ( self )`

Get the full Url to connect to.

#### Returns

Returns the address of the target server, e.g. <http://192.168.1.1:29500>

Definition at line 104 of file `configurator.py`.

### 6.1.3.4 `def pyrtai::configurator::Configurator::setAttr ( self, key, value )`

Sets member vars.

#### Parameters

<i>key</i>	The name of the var
<i>value</i>	The value to assign to the var

#### Returns

The value of the assigned var

Definition at line 75 of file `configurator.py`.

## 6.1.4 Member Data Documentation

### 6.1.4.1 `pyrtai::configurator::Configurator::address`

The address to connect to.

Definition at line 48 of file `configurator.py`.

#### 6.1.4.2 `pyrtai::configurator::Configurator::config`

The ConfigParser instance.

Definition at line 48 of file configurator.py.

#### 6.1.4.3 `pyrtai::configurator::Configurator::host`

The host to connect to.

It can be built using PROTOCOL://ADDRESS:PORT

Definition at line 48 of file configurator.py.

#### 6.1.4.4 `pyrtai::configurator::Configurator::is_slave_connection`

Booleen value which indicates if it's a slave connection.

It's a master collection if False

Definition at line 48 of file configurator.py.

#### 6.1.4.5 `pyrtai::configurator::Configurator::port`

The port to connect to.

Definition at line 48 of file configurator.py.

#### 6.1.4.6 `pyrtai::configurator::Configurator::protocol`

The protocol to use to connect to the server (HTTP/HTTPS)

Definition at line 48 of file configurator.py.

#### 6.1.4.7 `pyrtai::configurator::Configurator::session_id`

The session name.

Definition at line 48 of file configurator.py.

#### 6.1.4.8 `pyrtai::configurator::Configurator::slave_data_port`

The port to be used to connect to slave data stream.

Definition at line 48 of file configurator.py.

#### 6.1.4.9 `pyrtai::configurator::Configurator::slave_port`

The port to be used to connect to slave server.

Definition at line 48 of file configurator.py.

#### 6.1.4.10 pyrtai::configurator::Configurator::target\_name

The name of the target server.

Definition at line 48 of file configurator.py.

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

- </home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/configurator.py>

## 6.2 pyrtai::connector::Connector Class Reference

The [Connector](#) class connect to the RTAI server and manage responses and basic operations on it.

### Public Member Functions

- def [\\_\\_init\\_\\_](#)  
*The constructor instantiates the [configurator::Configurator](#) object.*
- def [Connection\\_Request](#)  
*Connect to the target server and handles the response to extract the slave server data params.*
- def [connectSlave](#)  
*Connect to Slave server.*
- def [disconnect](#)  
*Disconnect from the server.*
- def [closeSession](#)  
*Detach the session.*
- def [stopServer](#)  
*Stops master server.*
- def [getInfo](#)  
*Calls Info\_RT on the server.*
- def [callMethod](#)  
*Debug method to call functions directly to RPC connection.*

### Public Attributes

- [connection](#)  
*xmlrpclib::ServerProxy connection to XML-RPC server*

### 6.2.1 Detailed Description

The [Connector](#) class connect to the RTAI server and manage responses and basic operations on it.

### 6.2.2 Constructor & Destructor Documentation

6.2.2.1 `def pyrtai::connector::Connector::_init__( self, protocol = None, address = None, port = None, target_name = None )`

The constructor instantiates the [configurator::Configurator](#) object.

Definition at line 15 of file connector.py.

### 6.2.3 Member Function Documentation

6.2.3.1 `def pyrtai::connector::Connector::callMethod( self, function_name, params )`

Debug method to call functions directly to RPC connection.

#### Parameters

<i>function_name</i>	The name of the method to call
<i>params</i>	The params to pass to the method above

#### Returns

The response of the call

Definition at line 143 of file connector.py.

6.2.3.2 `def pyrtai::connector::Connector::closeSession( self, config )`

Detach the session.

#### Parameters

<i>config</i>	The <a href="#">configurator::Configurator</a> object, shared with the <a href="#">target::Target</a> instance
---------------	--

#### Returns

The boolean result of the operation

Definition at line 73 of file connector.py.



### 6.2.3.3 `def pyrtai::connector::Connector::Connection_Request ( self, config )`

Connect to the target server and handles the response to extract the slave server data params.

The data is saved in the [configurator::Configurator](#) object passed as first param.

#### Parameters

<i>config</i>	The <a href="#">configurator::Configurator</a> object, shared with the <a href="#">target::Target</a> instance
---------------	--

#### Returns

The boolean result of the operation

Definition at line 23 of file connector.py.

### 6.2.3.4 `def pyrtai::connector::Connector::connectSlave ( self, config )`

Connect to Slave server.

#### Parameters

<i>config</i>	The <a href="#">configurator::Configurator</a> object, shared with the <a href="#">target::Target</a> instance
---------------	--

#### Returns

The boolean result of the operation

Definition at line 40 of file connector.py.

### 6.2.3.5 `def pyrtai::connector::Connector::disconnect ( self, session_id )`

Disconnect from the server.

#### Parameters

<i>session_id</i>	The session ID
-------------------	----------------

#### Returns

The boolean result of the operation

Definition at line 58 of file connector.py.

### 6.2.3.6 `def pyrtai::connector::Connector::getInfo ( self )`

Calls Info\_RT on the server.

#### Returns

The boolean result of the operation

Definition at line 118 of file connector.py.

### 6.2.3.7 `def pyrtai::connector::Connector::stopServer ( self, target_name )`

Stops master server.

#### Parameters

<i>target_name</i>	The Target name
--------------------	-----------------

#### Returns

The boolean result of the operation

Definition at line 96 of file connector.py.

## 6.2.4 Member Data Documentation

### 6.2.4.1 `pyrtai::connector::Connector::connection`

xmlrpclib::ServerProxy connection to XML-RPC server

Definition at line 15 of file connector.py.

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

- </home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/connector.py>

## 6.3 `pyrtai::data_collector::DataCollector` Class Reference

### Public Member Functions

- `def __init__`

### Public Attributes

- `input`  
*The input stream.*

- [output](#)

*The output stream.*

### 6.3.1 Constructor & Destructor Documentation

#### 6.3.1.1 `def pyrtai::data_collector::DataCollector::_init__( self )`

Definition at line 13 of file `data_collector.py`.

### 6.3.2 Member Data Documentation

#### 6.3.2.1 `pyrtai::data_collector::DataCollector::input`

The input stream.

Definition at line 13 of file `data_collector.py`.

#### 6.3.2.2 `pyrtai::data_collector::DataCollector::output`

The output stream.

Definition at line 13 of file `data_collector.py`.

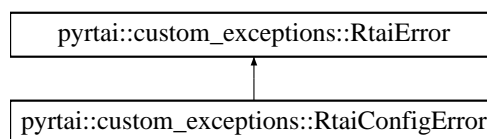
The documentation for this class was generated from the following file:

- `/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/data\_collector.py`

## 6.4 pyrtai::custom\_exceptions::RtaiConfigError Class Reference

Exception raised for errors involving connections.

Inheritance diagram for `pyrtai::custom_exceptions::RtaiConfigError`:



### Public Member Functions

- `def __init__`

*Saves expr and msg of the error.*

- `def __str__`

*Returns string representation of the error.*

## Public Attributes

- `expr`  
*Input expression in which the error occurred.*
- `msg`  
*Explanation of the error.*

### 6.4.1 Detailed Description

Exception raised for errors involving connections.

### 6.4.2 Constructor & Destructor Documentation

6.4.2.1 `def pyrtai::custom_exceptions::RtaiConfigError::__init__( self, expr, msg )`

Saves `expr` and `msg` of the error.

Definition at line 18 of file `custom_exceptions.py`.

### 6.4.3 Member Function Documentation

6.4.3.1 `def pyrtai::custom_exceptions::RtaiConfigError::__str__( self )`

Returns string representation of the error.

Definition at line 23 of file `custom_exceptions.py`.

### 6.4.4 Member Data Documentation

6.4.4.1 `pyrtai::custom_exceptions::RtaiConfigError::expr`

Input expression in which the error occurred.

Definition at line 18 of file `custom_exceptions.py`.

6.4.4.2 `pyrtai::custom_exceptions::RtaiConfigError::msg`

Explanation of the error.

Definition at line 18 of file `custom_exceptions.py`.

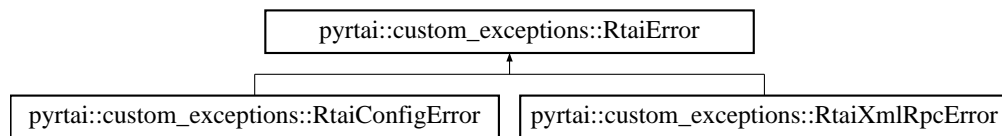
The documentation for this class was generated from the following file:

- `/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/custom_exceptions.py`

## 6.5 pyrtai::custom\_exceptions::RtaiError Class Reference

Base class for exceptions in this module.

Inheritance diagram for pyrtai::custom\_exceptions::RtaiError:



### 6.5.1 Detailed Description

Base class for exceptions in this module.

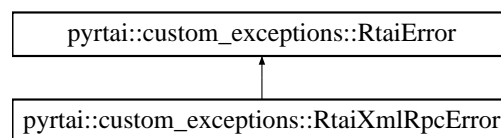
The documentation for this class was generated from the following file:

- [/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/custom\\_exceptions.py](#)

## 6.6 pyrtai::custom\_exceptions::RtaiXmlRpcError Class Reference

Exception raised for errors involving connections.

Inheritance diagram for pyrtai::custom\_exceptions::RtaiXmlRpcError:



### Public Member Functions

- `def \_\_init\_\_`  
*Saves expr and msg of the error.*
- `def \_\_str\_\_`  
*Returns string representation of the error.*

### Public Attributes

- `expr`  
*Input expression in which the error occurred.*

- [msg](#)

*Explanation of the error.*

### 6.6.1 Detailed Description

Exception raised for errors involving connections.

### 6.6.2 Constructor & Destructor Documentation

6.6.2.1 `def pyrtai::custom_exceptions::RtaiXmlRpcError::__init__( self, expr, msg )`

Saves expr and msg of the error.

Definition at line 36 of file custom\_exceptions.py.

### 6.6.3 Member Function Documentation

6.6.3.1 `def pyrtai::custom_exceptions::RtaiXmlRpcError::__str__( self )`

Returns string representation of the error.

Definition at line 41 of file custom\_exceptions.py.

### 6.6.4 Member Data Documentation

6.6.4.1 `pyrtai::custom_exceptions::RtaiXmlRpcError::expr`

Input expression in which the error occurred.

Definition at line 36 of file custom\_exceptions.py.

6.6.4.2 `pyrtai::custom_exceptions::RtaiXmlRpcError::msg`

Explanation of the error.

Definition at line 36 of file custom\_exceptions.py.

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

- /home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/[custom\\_exceptions.py](#)

## 6.7 pyrtai::target::Target Class Reference

## Public Member Functions

- def `__init__`  
*Target* initializer.
- def `connect`  
Connect to the target server.
- def `disconnect`  
Disconnects from the XML-RPC server.
- def `stopServer`  
Stops master server.
- def `close`  
Closes the session.
- def `getSlave`  
Checks if master and if affirmative, returns the slave.

## Public Attributes

- `slave`  
The slave *Target*.
- `config`  
A *configurator::Configurator* object.
- `connection`  
A *connector::Connector* instance used to send commands to XML-RPC connection.

### 6.7.1 Constructor & Destructor Documentation

6.7.1.1 `def pyrtai::target::Target::__init__( self, protocol=None, address=None, port=None, target_name=None )`

*Target* initializer.

If the params are None, they're extracted from self.configurator

Definition at line 19 of file target.py.

### 6.7.2 Member Function Documentation

6.7.2.1 `def pyrtai::target::Target::close ( self )`

Closes the session.

Returns

The boolean result of the operation

Definition at line 66 of file target.py.

#### 6.7.2.2 `def pyrtai::target::Target::connect ( self )`

Connect to the target server.

Checks if master or server and call the right method on [connector::Connector](#)

##### Returns

The boolean result of the operation

Definition at line 40 of file target.py.

#### 6.7.2.3 `def pyrtai::target::Target::disconnect ( self )`

Disconnects from the XML-RPC server.

##### Returns

The boolean result of the operation

Definition at line 48 of file target.py.

#### 6.7.2.4 `def pyrtai::target::Target::getSlave ( self )`

Checks if master and if affirmative, returns the slave.

##### Returns

The slave [Target](#) or False if not a master target

Definition at line 71 of file target.py.

#### 6.7.2.5 `def pyrtai::target::Target::stopServer ( self )`

Stops master server.

##### Returns

The boolean result of the operation

Definition at line 57 of file target.py.

### 6.7.3 Member Data Documentation

#### 6.7.3.1 `pyrtai::target::Target::config`

A [configurator::Configurator](#) object.

Definition at line 19 of file target.py.



### 6.7.3.2 pyrtai::target::Target::connection

A [connector::Connector](#) instance used to send commands to XML-RPC connection.

Definition at line 19 of file target.py.

### 6.7.3.3 pyrtai::target::Target::slave

The slave [Target](#).

Definition at line 19 of file target.py.

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

- [/home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/target.py](#)



## Chapter 7

# File Documentation

### 7.1 /home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/\_\_\_init\_\_\_.py File Reference

#### Packages

- namespace [pyrtai](#)  
*This package handles the configurations.*

### 7.2 /home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/configurator.py File Reference

#### Classes

- class [pyrtai::configurator::Configurator](#)  
*[Configurator](#) class handles config values.*

#### Packages

- namespace [pyrtai::configurator](#)
- namespace [pyrtai](#)  
*This package handles the configurations.*

### 7.3 /home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/connector.py File Reference

## Classes

- class [pyrtai::connector::Connector](#)

*The [Connector](#) class connect to the RTAI server and manage responses and basic operations on it.*

## Packages

- namespace [pyrtai::connector](#)
- namespace [pyrtai](#)

*This package handles the configurations.*

## 7.4 /home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/custom\_exceptions.py File Reference

### Classes

- class [pyrtai::custom\\_exceptions::RtaiError](#)

*Base class for exceptions in this module.*

- class [pyrtai::custom\\_exceptions::RtaiConfigError](#)

*Exception raised for errors involving connections.*

- class [pyrtai::custom\\_exceptions::RtaiXmlRpcError](#)

*Exception raised for errors involving connections.*

### Packages

- namespace [pyrtai::custom\\_exceptions](#)
- namespace [custom\\_exceptions](#)

*Custom exceptions for the library.*

## 7.5 /home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/data\_collector.py File Reference

### Classes

- class [pyrtai::data\\_collector::DataCollector](#)

### Packages

- namespace [pyrtai::data\\_collector](#)

## 7.6 /home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/target.py File Reference

### Classes

- class [pyrtai::target::Target](#)

### Packages

- namespace [pyrtai::target](#)
- namespace [pyrtai](#)

*This package handles the configurations.*

## 7.7 /home/drwolf/Devel/Remote-RTAI-lib/Python Library/pyrtai/utility.py File Reference

### Packages

- namespace [pyrtai::utility](#)
- namespace [pyrtai](#)

*This package handles the configurations.*

### Functions

- def [pyrtai::utility::getStateFromResponse](#)

*Parses an integer state value which, converted into binary value, represents 4 states.*