

---

# **rodario Documentation**

***Release 1.0.0a6***

**haliphax**

January 18, 2016



<b>1</b>	<b>The Registry</b>	<b>3</b>
<b>2</b>	<b>Actors and Proxies</b>	<b>5</b>
<b>3</b>	<b>Decorators</b>	<b>7</b>
<b>4</b>	<b>Exceptions</b>	<b>9</b>
<b>5</b>	<b>Indices and tables</b>	<b>11</b>



A simple, redis-backed Python actor framework

- [Homepage](#)



---

## The Registry

---

```
class rodario.registry.Registry
    Actor registry class (singleton wrapper)

    static __new__ ()
        Retrieve the singleton instance for Registry.

        Return type rodario.registry._RegistrySingleton

class rodario.registry._RegistrySingleton
    Singleton for actor registry

    __init__ ()
        Initialize the registry.

    actors
        Retrieve a list of registered actors.

        Return type set

    exists (uuid)
        Test whether an actor exists in the registry.

        Parameters uuid (str) – UUID of the actor to check for

        Return type bool

    get_proxy (uuid)
        Return an ActorProxy for the given UUID.

        Parameters uuid (str) – The UUID to return a proxy object for

        Return type rodario.actors.ActorProxy

    register (uuid)
        Register a new actor.

        Parameters uuid (str) – The UUID of the actor to register

    unregister (uuid)
        Unregister an existing actor.

        Parameters uuid (str) – The UUID of the actor to unregister
```





---

## Actors and Proxies

---

```
class rodario.actors.Actor (uuid=None)
    Base Actor class

    __init__ (uuid=None)
        Initialize the Actor object.

        Parameters uuid (str) – Optionally-provided UUID

    is_alive
        Return True if this Actor is still alive.

        Return type bool

    join (channel, func)
        Join this Actor to a pubsub cluster channel.

        Parameters

        • channel (str) – The channel to join

        • func (callable) – The message handler function

    part (channel)
        Remove this Actor from a pubsub cluster channel.

        Parameters channel (str) – The channel to part

    proxy ()
        Wrap this Actor in an ActorProxy object.

        Return type rodario.actors.ActorProxy

    start ()
        Fire up the message handler thread.

    stop ()
        Kill the message handler thread.

class rodario.actors.ActorProxy (actor=None, uuid=None)
    Proxy object that fires calls to an actor over redis pubsub

    __init__ (actor=None, uuid=None)
        Initialize instance of ActorProxy.

        Accepts either an Actor object to clone or a UUID, but not both.

        Parameters

        • actor (rodario.actors.Actor) – Actor to clone
```

- **uuid** (*str*) – UUID of Actor to clone

**\_\_proxy** (*method\_name*, \**args*, \*\**kwargs*)

Proxy a method call to redis pubsub.

This method is not meant to be called directly. Instead, it is used by the proxy's self-generated methods to provide the proxy with the same public API as the actor it represents.

**Parameters**

- **method\_name** (*str*) – The method to proxy
- **args** (*tuple*) – The arguments to pass
- **kwargs** (*dict*) – The keyword arguments to pass

**Return type** `multiprocessing.Queue`

**proxyid** = `None`

This proxy object's UUID for creating unique channels

**class** `rodario.future.Future` (*queue*)

Custom response type for proxied method calls

**\_\_init\_\_** (*queue*)

Initialize the Future by saving a reference to the Queue

**Parameters** **queue** (`multiprocessing.Queue`) – The response queue to wrap

**get** (\**args*, \*\**kwargs*)

Resolve and return the proxied method call's value.

**Return type** `mixed`

**ready**

Return True if the response value is available.

**Return type** `bool`

---

## Decorators

---

`rodario.decorators.blocking` (*func*)

Block the thread and return the proxied method call's result.

**Parameters** `func` (*instancemethod*) – The function to wrap

**Return type** `rodario.decorators.BlockingMethod`



---

## Exceptions

---

**class** `rodario.exceptions.InvalidActorException`  
Raised when a referenced actor does not exist

**class** `rodario.exceptions.InvalidProxyException`  
Raised when a proxy is not given a valid object to wrap

**class** `rodario.exceptions.UUIDInUseException`  
Raised during UUID registration if the UUID is already taken

**class** `rodario.exceptions.RegistrationException`  
Raised when actor registration fails



---

## Indices and tables

---

- `genindex`
- `search`





## Symbols

`_RegistrySingleton` (class in `rodario.registry`), 3  
`__init__()` (`rodario.actors.Actor` method), 5  
`__init__()` (`rodario.actors.ActorProxy` method), 5  
`__init__()` (`rodario.future.Future` method), 6  
`__init__()` (`rodario.registry._RegistrySingleton` method), 3  
`__new__()` (`rodario.registry.Registry` static method), 3  
`_proxy()` (`rodario.actors.ActorProxy` method), 6

## A

`Actor` (class in `rodario.actors`), 5  
`ActorProxy` (class in `rodario.actors`), 5  
`actors` (`rodario.registry._RegistrySingleton` attribute), 3

## B

`blocking()` (in module `rodario.decorators`), 7

## E

`exists()` (`rodario.registry._RegistrySingleton` method), 3

## F

`Future` (class in `rodario.future`), 6

## G

`get()` (`rodario.future.Future` method), 6  
`get_proxy()` (`rodario.registry._RegistrySingleton` method), 3

## I

`InvalidActorException` (class in `rodario.exceptions`), 9  
`InvalidProxyException` (class in `rodario.exceptions`), 9  
`is_alive` (`rodario.actors.Actor` attribute), 5

## J

`join()` (`rodario.actors.Actor` method), 5

## P

`part()` (`rodario.actors.Actor` method), 5

`proxy()` (`rodario.actors.Actor` method), 5  
`proxyid` (`rodario.actors.ActorProxy` attribute), 6

## R

`ready` (`rodario.future.Future` attribute), 6  
`register()` (`rodario.registry._RegistrySingleton` method), 3  
`RegistrationException` (class in `rodario.exceptions`), 9  
`Registry` (class in `rodario.registry`), 3

## S

`start()` (`rodario.actors.Actor` method), 5  
`stop()` (`rodario.actors.Actor` method), 5

## U

`unregister()` (`rodario.registry._RegistrySingleton` method), 3  
`UUIDInUseException` (class in `rodario.exceptions`), 9