
Lazarus Documentation

Release 0.6.3

Lazarus Authors

December 09, 2014

1	Lazarus	3
1.1	Features	3
1.2	Examples	3
2	Installation	5
3	Usage	7
4	lazarus	9
4.1	lazarus package	9
5	Contributing	13
5.1	Types of Contributions	13
6	Credits	15
6.1	Creator and maintainer	15
6.2	Contributors	15
7	History	17
7.1	0.6.3 2014-12-10	17
7.2	0.6.2 2014-12-09	17
7.3	0.6.1 2014-12-07	17
7.4	0.5.1 2014-11-13	17
7.5	0.5 2013-10-17	17
7.6	0.4 2013-10-17	17
7.7	0.3 2013-10-16	18
7.8	0.1 2013-10-15	18
8	Indices and tables	19
	Python Module Index	21

Contents:

A Python library to restart the process when source code changes.

- Free software: Apache License, Version 2.0
- Documentation: <http://lazarus.rtfld.org>.

1.1 Features

- Automatically restart when source changes
- Customize when, why, and how restarts happen
- Runs on Linux and Mac

1.2 Examples

Restart when any Python module rooted at PYTHONPATH changes:

```
>>> import lazarus
>>> lazarus.default()
```

Same thing, but within a uWSGI container:

```
>>> import lazarus
>>> lazarus.default(restart_func=lambda: uwsgi.reload(), close_fds=False)
```

Installation

At the command line:

```
$ easy_install lazarus
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv lazarus  
$ pip install lazarus
```

Usage

To use lazarus in a project:

```
import lazarus
```

Then depending on mode:

```
lazarus.default()
```

Or:

```
lazarus.custom(...)
```


4.1 lazarus package

4.1.1 Module contents

Functions to restart a process when source changes.

Progress doesn't come from early risers - progress is made by lazy men looking for easier ways to do things.

```
lazarus.custom(srcpaths, event_cb=None, poll_interval=1, recurse=True, restart_cb=None,  
               restart_func=None, close_fds=True)
```

Sets up lazarus in custom mode.

See the `default()` function for a simpler mode of use.

The custom mode of lazarus is to watch all modules rooted at any of the source paths provided for changes and restart when they take place.

Keyword arguments:

`event_cb` – Callback invoked when a file rooted at a source path changes. Without specifying an event callback, changes to any module rooted at a source path will trigger a restart.

`poll_interval` – Rate at which changes will be detected. The default value of 1 means it may take up to one second to detect changes. Decreasing this value may lead to unnecessary overhead.

`recurse` – Whether to watch all subdirectories of every source path for changes or only the paths provided.

`restart_cb` – Callback invoked prior to restarting the process; allows for any cleanup to occur prior to restarting. Returning anything other than *None* in the callback will cancel the restart.

`restart_func` – Function invoked to restart the process. This supplants the default behavior of using *sys.executable* and *sys.argv*.

`close_fds` – Whether all file descriptors other than *stdin*, *stdout*, and *stderr* should be closed

An example of using a cleanup function prior to restarting:

```
>>> def cleanup():  
...     pass  
>>> import lazarus  
>>> lazarus.custom(os.getcwd(), restart_cb=cleanup)  
>>> lazarus.stop()
```

An example of avoiding restarts when any `__main__.py` changes:

```
>>> def skip_main(event):
...     if event.src_path == '__main__.py':
...         return False
...     return True
>>> import lazarus
>>> lazarus.custom(os.getcwd(), event_cb=skip_main)
>>> lazarus.stop()
```

`lazarus.default(restart_cb=None, restart_func=None, close_fds=True)`

Sets up lazarus in default mode.

See the `custom()` function for a more powerful mode of use.

The default mode of lazarus is to watch all modules rooted at `PYTHONPATH` for changes and restart when they take place.

Keyword arguments:

`restart_cb` – Callback invoked prior to restarting the process; allows for any cleanup to occur prior to restarting. Returning anything other than *None* in the callback will cancel the restart.

`restart_func` – Function invoked to restart the process. This supplants the default behavior of using `sys.executable` and `sys.argv`.

`close_fds` – Whether all file descriptors other than *stdin*, *stdout*, and *stderr* should be closed

A simple example:

```
>>> import lazarus
>>> lazarus.default()
>>> lazarus.stop()
```

`lazarus.is_restart_event(event)`

Default logic for whether a filesystem event is a *restart event*.

For example:

```
>>> import collections
>>> Event = collections.namedtuple('Event', 'src_path dest_path')
>>> vim_ev = Event('foo.py', None)
>>> is_restart_event(vim_ev)
True
>>> sublime_ev = Event('.subl6f0.tmp', '__main__.py')
>>> is_restart_event(sublime_ev)
True
```

If the event's source or destination path ends in `.py`, the event is considered a *restart event*. This covers most cases where a restart should take place like developers using editors, IDEs, or version control operations.

`lazarus.stop()`

Stops lazarus, regardless of which mode it was started in.

For example:

```
>>> import lazarus
>>> lazarus.default()
>>> lazarus.stop()
```

Contents

- [Contributing](#)
 - [Types of Contributions](#)
 - * [Bug Reports](#)
 - * [Bug Fixes](#)
 - * [Enhancements](#)
 - * [Documentation](#)
 - * [Feedback](#)

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

5.1 Types of Contributions

You can contribute in many ways:

5.1.1 Bug Reports

Report bugs at <https://github.com/formwork-io/lazarus/issues>.

Before reporting a bug, read these:

- <http://www.chiark.greenend.org.uk/~sgtatham/bugs.html>
- <http://www.catb.org/~esr/faqs/smart-questions.html>

5.1.2 Bug Fixes

Look through the GitHub issues for anything tagged with “bug”.

5.1.3 Enhancements

Look through the GitHub issues for anything tagged with “enhancement”.

5.1.4 Documentation

Lazarus could always use more documentation, whether as part of the official formwork-io/lazarus docs, in docstrings, or even on the web in blog posts, articles, and such.

5.1.5 Feedback

The best way to send feedback is to file an issue at <https://github.com/formwork-io/lazarus/issues>.

Credits

6.1 Creator and maintainer

- Nick Bargnesi <nick@den-4.com>

6.2 Contributors

None yet. Why not be the first?

History

7.1 0.6.3 2014-12-10

- Expanded PyPI package page.

7.2 0.6.2 2014-12-09

- Fixed Sublime Text save workflows.

7.3 0.6.1 2014-12-07

- Added Mac Support.

7.4 0.5.1 2014-11-13

- Migrated GitHub repository to formwork-io

7.5 0.5 2013-10-17

- Released to PyPI

7.6 0.4 2013-10-17

- Fixed crash when multiple changes detected and first triggers restart (#6)
- Fixed default mode always closing FDs (#5)

7.7 0.3 2013-10-16

- Added ability to specify restart command (#1)
- Added ability to cancel the restart (#2)

7.8 0.1 2013-10-15

- First functional release.

Indices and tables

- *genindex*
- *modindex*
- *search*

I

lazarus, 9

C

`custom()` (in module `lazarus`), [9](#)

D

`default()` (in module `lazarus`), [10](#)

I

`is_restart_event()` (in module `lazarus`), [10](#)

L

`lazarus` (module), [9](#)

S

`stop()` (in module `lazarus`), [10](#)