
Messages

Release 0.0.0

May 26, 2016

1	Wait... Why? What??	3
2	Ok... Where?	5
2.1	Changelog	5

Applications want to send messages to other endpoints. They don't want discover the other endpoint, format the payload of the message to the specified format, adjust message metadata, connect, and send the message. Sending a message is the most basic concept in distributing computing. Why do we make it so difficult?

This library makes sending a message easy again. It does this by providing a simple way to connect application code that produces or consumes message payloads, library code that knows how to transform a payload into a message, and a connector to send the message according to a specific protocol. The actual protocols are implemented by other libraries (e.g, [requests](#), [pika](#), etc.).

Wait... Why? What??

We already have `requests` for sending HTTP requests, `tornado` makes it simple to consume HTTP messages using a `RequestHandler`. There is `pika` for AMQP. `ZeroMQ` is embarrassingly easy to use. Why on Earth do we need to combine them? Just pick the library for the protocol in question and use it!

That is a very good point. Python encourages us to use well-defined, small, efficient tools whenever possible. This philosophy is one that I agree with and goes back even further to the roots of why UNIX is the way it is. So *why did I feel the need to write this library?* The answer is relatively simple. We have lots of ways to send and receive messages but there is no **universal interface** for the simple act of sending and receiving a message. That is the problem that this library is solving.

Applications want to send and receive messages. Let's make that possible.

Ok... Where?

Source	https://github.com/dave-shawley/messages
Status	https://travis-ci.org/dave-shawley/messages
Download	https://pypi.python.org/pypi/messages
Documentation	http://messages.readthedocs.org/en/latest/
Issues	https://github.com/dave-shawley/messages

2.1 Changelog

- Next Release
 - Some functionality....