

---

# **aiohttp-transmute Documentation**

***Release 0.1***

**Yusuke Tsutsumi**

December 29, 2016



<b>1</b>	<b>Example</b>	<b>3</b>
1.1	Installing . . . . .	3
1.2	Routes . . . . .	4
1.3	Serialization . . . . .	4
1.4	Autodocumentation . . . . .	5
	<b>Python Module Index</b>	<b>7</b>



A [transmute](#) framework for [aiohttp](#). This framework provides:

- declarative generation of http handler interfaces by parsing function annotations
- validation and serialization to and from a variety of content types (e.g. json or yaml).
- validation and serialization to and from native python objects, using [schematics](#).
- autodocumentation of all handlers generated this way, via [swagger](#).



---

## Example

---

```
from aiohttp import web
import aiohttp_transmute

# define a GET endpoint, taking a query parameter integers left and right,
# which must be integers.
@aiohttp_transmute.describe(paths="/customers/{name}")
async def multiply(request, name: str, left: int, right: int) -> int:
    return left + right

# define a POST endpoint, taking a query parameter integers left and right,
# which must be integers.
@aiohttp_transmute.describe(methods="POST", paths="/customers/{name}")
async def multiply_post(request, name: str, left: int, right: int) -> int:
    return left + right

app = web.Application()
# use add_route to add the function to the app.
aiohttp_transmute.route(app, multiply)
aiohttp_transmute.route(app, multiply_post)
# this should be at the end, to ensure all routes are considered when
# constructing the handler.
# this will add:
#   - a swagger.json definition
#   - a static page that renders the swagger.json
aiohttp_transmute.add_swagger(app, "/swagger.json", "/swagger")
web.run_app(app)
```

A lot of relevant documentation for aiohttp-transmute can be found at [transmute-core docs](#)

Contents:

## 1.1 Installing

aiohttp-transmute can be installed via [Pip](#), from [PyPI](#):

```
$ pip install aiohttp-transmute
```

Pip allows installation from source as well:

```
$ pip install git+https://github.com/toumorokoshi/aiohttp-transmute.git#egg=aiohttp-transmute
```

## 1.2 Routes

### 1.2.1 Example

Adding routes follows the standard transmute pattern, with a decorator converting a function to an aiohttp route:

```
import aiohttp_transmute

# define a GET endpoint, taking a query parameter integers left and right,
# which must be integers.
@aiohttp_transmute.describe(paths="/{name}")
async def multiply(request, name: str, left: int, right: int) -> int:
    return left + right

# append to your route later
aiohttp_transmute.route(app, multiply)
```

the aiohttp request argument is supported: it will be passed into any function that has ‘request’ in it’s function signature. see [transmute-core:function](#) for more information on customizing transmute routes.

### 1.2.2 API Documentation

`aiohttp_transmute.describe(**kwargs)`

describe is a decorator to customize the rest API that transmute generates, such as choosing certain arguments to be query parameters or body parameters, or a different method.

#### Parameters

- **paths** (*list(str)*) – the path(s) for the handler to represent (using swagger’s syntax for a path)
- **methods** (*list(str)*) – the methods this function should respond to. if non is set, transmute defaults to a GET.
- **query\_parameters** (*list(str)*) – the names of arguments that should be query parameters. By default, all arguments are query\_or path parameters for a GET request.
- **body\_parameters** (*list(str)*) – the names of arguments that should be body parameters. By default, all arguments are either body or path parameters for a non-GET request.
- **header\_parameters** (*list(str)*) – the arguments that should be passed into the header.
- **path\_parameters** (*list(str)*) – the arguments that are specified by the path. By default, arguments that are found in the path are used first before the query\_parameters and body\_parameters.

## 1.3 Serialization

See [serialization](#) in transmute-core.



## 1.4 Autodocumentation

You can use `add_swagger(app, json_path, html_path)` to add swagger documentation for all transmute routes.

```
aiohttp_transmute.add_swagger(app, "/swagger.json", "/swagger")
```

### 1.4.1 API Reference

`aiohttp_transmute.swagger.add_swagger` (*app*, *json\_route*, *html\_route*)

a convenience method for both adding a swagger.json route, as well as adding a page showing the html documentation

`aiohttp_transmute.swagger.add_swagger_api_route` (*app*, *target\_route*, *swagger\_json\_route*)

mount a swagger statics page.

*app*: the aiohttp app object *target\_route*: the path to mount the statics page. *swagger\_json\_route*: the path where the swagger json definitions is

expected to be.

`aiohttp_transmute.swagger.create_swagger_json_handler` (*app*, *\*\*kwargs*)

Create a handler that returns the swagger definition for an application.

This method assumes the application is using the TransmuteUrlDispatcher as the router.



## **a**

`aiohttp_transmute.swagger`, [5](#)



## A

`add_swagger()` (in module `aiohttp_transmute.swagger`), [5](#)  
`add_swagger_api_route()` (in module `aiohttp_transmute.swagger`), [5](#)  
`aiohttp_transmute.swagger` (module), [5](#)

## C

`create_swagger_json_handler()` (in module `aiohttp_transmute.swagger`), [5](#)

## D

`describe()` (in module `aiohttp_transmute`), [4](#)