

---

# **edeposit.amqp.storage**

***Release 0.2.0***

June 23, 2015



<b>1</b>	<b>Package structure</b>	<b>3</b>
1.1	File relations . . . . .	4
1.2	API . . . . .	4
<b>2</b>	<b>Installation</b>	<b>7</b>
2.1	Installation notes . . . . .	7
2.2	Source code . . . . .	8
2.3	Unittests . . . . .	8
<b>3</b>	<b>Indices and tables</b>	<b>9</b>



Long term storage subsystem for the [E-deposit](#) project.

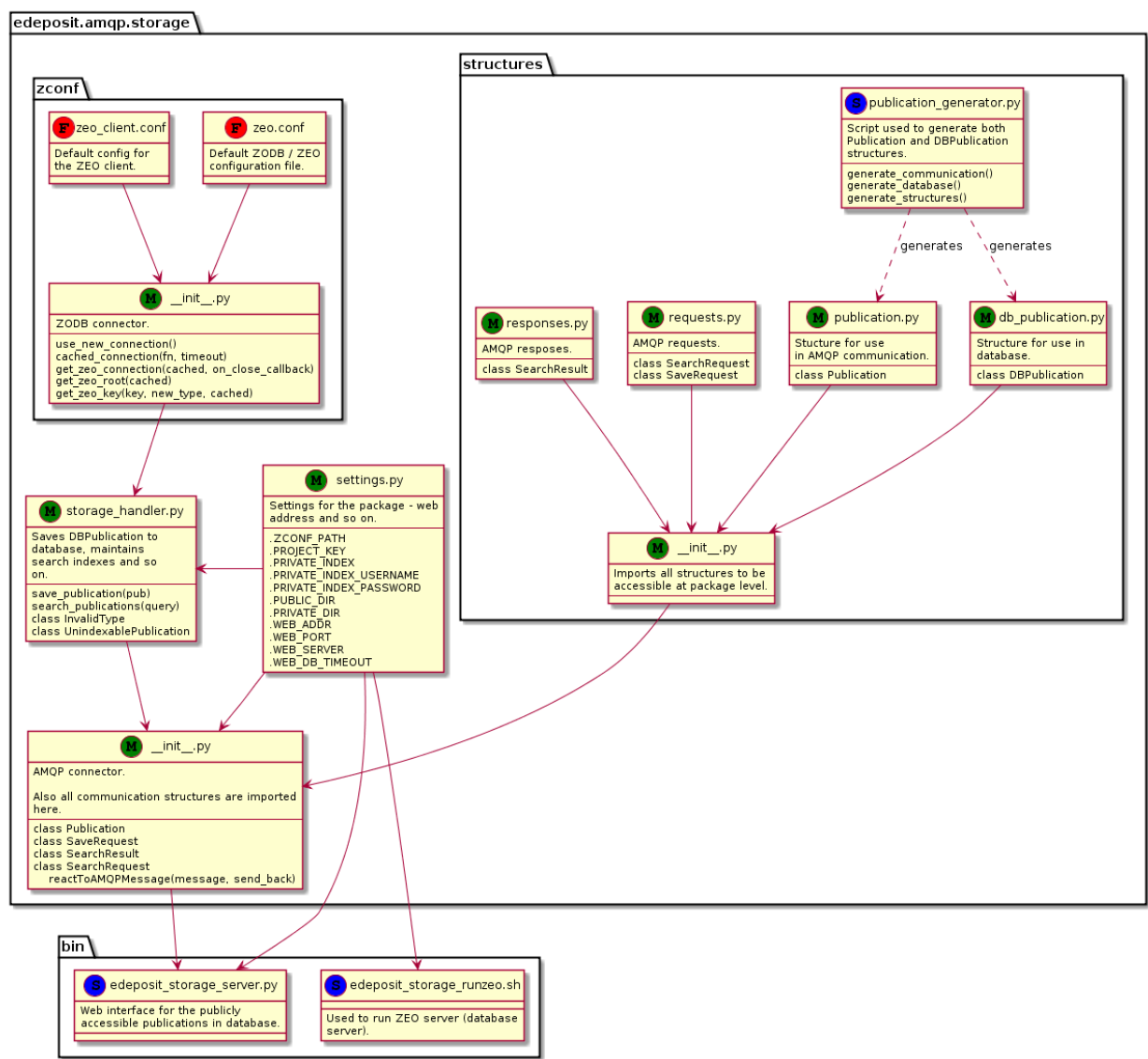
This project allows to store and retrieve publications over [AMQP](#) and also to optionally access accessible publications via HTTP using builtin webserver written in [bottle.py](#).





Package structure

1.1 File relations



1.2 API



### 1.2.1 storage\_handler module

This module handles the database, maintains indexes and provides search function over this indexes.

### 1.2.2 zconf submodule

`ZEO` and `ZODB` configuration and access functions.

### API

### 1.2.3 settings submodule

`/api/structures/structures:`

### 1.2.4 publication structure

Communication structure used by AMQP.

### 1.2.5 db\_publication structure

Structure used in ZODB (database).

### 1.2.6 requests submodule

Structures used for AMQP communication requests.

### 1.2.7 responses submodule

Structures used for AMQP responses.

### 1.2.8 publication\_generator script

This script is used to generate `Publication` and `DBPublication` structures from common data.



---

## Installation

---

Installation of this project is little bit more complicated. Please read installation notes:

### 2.1 Installation notes

Module itself can be installed using PIP:

```
sudo pip install edeposit.amqp.storage
```

#### 2.1.1 Configuration

After the installation, some configuration is required. Configuration is done using `settings.py` script, which reads data from configuration path `~/edeposit/storage.json`.

Each uppercase attribute defined in `settings` can be reconfigured using the `storage.json` configuration file.

Required configuration options is:

- `PUBLIC_DIR`
- `PRIVATE_DIR`

Highly recommended options:

- `PRIVATE_INDEX`
- `PRIVATE_INDEX_USERNAME`
- `PRIVATE_INDEX_PASSWORD`
- `WEB_ADDR`
- `WEB_PORT`
- `WEB_SERVER`

You should definitely change the `WEB_SERVER` to `paste`. By default, the `wsgiref` backend is used, but that is only single-thread server. Paste will allow multithread access of users to your server.

#### 2.1.2 Example of the configuration

```
{
  "PUBLIC_DIR": "/var/edeposit_storage/public",
  "PRIVATE_DIR": "/var/edeposit_storage/private",
  "PRIVATE_INDEX": true,
  "PRIVATE_INDEX_PASSWORD": "secret password",
  "WEB_SERVER": "paste"
}
```

## 2.2 Source code

Project is released under MIT license. Source code can be found at GitHub:

- <https://github.com/edeposit/edeposit.amqp.storage>

## 2.3 Unittests

Almost every feature of the project is tested by unittests. You can run those tests using provided `run_tests.sh` script, which can be found in the root of the project.

If you have any trouble, just add `--pdb` switch at the end of your `run_tests.sh` command like this: `./run_tests.sh --pdb`. This will drop you to **PDB** shell.

### 2.3.1 Requirements

This script expects that packages `pytest`, `fake-factory` and `sh` is installed. In case you don't have it yet, it can be easily installed using following command:

```
pip install --user pytest fake-factory sh
```

or for all users:

```
sudo pip install pytest fake-factory sh
```

### 2.3.2 Example

```
./run_tests.sh
===== test session starts =====
platform linux2 -- Python 2.7.6 -- py-1.4.26 -- pytest-2.6.4
plugins: cov
collected 15 items

tests/test_amqp_chain.py ..
tests/test_storage_handler.py .....
tests/structures/test_db_publication.py ...
tests/structures/test_publication.py .

===== 15 passed in 11.20 seconds =====
```

---

## Indices and tables

---

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