

---

# **pgraph Documentation**

*Release 1.0.2*

**Kouhei Maeda <mkouhei@palmtb.net>**

**May 27, 2020**



---

## Contents

---

<b>1</b>	<b>pggraph: drawing graph of Python packages dependencies</b>	<b>3</b>
1.1	Status . . . . .	3
1.2	Demo . . . . .	3
<b>2</b>	<b>How to run things locally</b>	<b>5</b>
<b>3</b>	<b>Deploying app to Heroku</b>	<b>7</b>
3.1	Runnig test locally . . . . .	8
<b>4</b>	<b>History</b>	<b>11</b>
4.1	1.0.2 (2020-05-27) . . . . .	11
4.2	1.0.1 (2020-05-27) . . . . .	11
4.3	1.0.0 (2020-05-17) . . . . .	11
4.4	0.5.1 (2016-04-09) . . . . .	11
4.5	0.5.0 (2015-10-09) . . . . .	12
4.6	0.4.8 (2015-08-19) . . . . .	12
4.7	0.4.7 (2015-08-15) . . . . .	12
4.8	0.4.6 (2015-07-27) . . . . .	12
4.9	0.4.5 (2015-07-22) . . . . .	12
4.10	0.4.4 (2015-07-19) . . . . .	12
4.11	0.4.3 (2015-07-16) . . . . .	13
4.12	0.4.2 (2015-07-12) . . . . .	13
4.13	0.4.1 (2015-07-03) . . . . .	13
4.14	0.4.0 (2015-06-29) . . . . .	13
4.15	0.3.1 (2015-06-24) . . . . .	13
4.16	0.3.0 (2015-06-24) . . . . .	13
4.17	0.2.0 (2015-06-19) . . . . .	14
4.18	0.1.0 (2015-06-08) . . . . .	14
<b>5</b>	<b>Indices and tables</b>	<b>15</b>



Contents:



---

## pgraph: drawing graph of Python packages dependencies

---

The pgraph provides the graph of Python dependencies, powered by `py-deps`, `Linkdraw`, `Celery`, `pylibmc`, and `Pyramid`.

### 1.1 Status



### 1.2 Demo

Demo site





---

## How to run things locally

---

### 1. Installing RabbitMQ.

```
$ sudo apt-get install rabbitmq-server
```

### 2. Setup RabbitMQ.

```
$ sudo rabbitmqctl add_vhost /pgraph
$ sudo rabbitmqctl add_user pgraph passwd0rd
$ sudo rabbitmqctl set_permissions -p /pgraph pgraph ".*" ".*" ".*"
$ sudo rabbitmqctl delete_user guest
```

### 3. Cloning repository.

```
$ git clone https://github.com/mkouhei/pgraph
$ cd pgraph
$ git submodule init
$ git submodule update
```

### 4. Createing virtual environment.

```
$ python -m venv /path/to/venv
$ . /path/to/venv/bin/activate
```

### 5. Installing pgraph.

```
(venv)$ cd /path/to/pgraph
(venv)$ pip install -e . -e .[development]
```

### 6. Running celery worker.

```
(venv)$ CONFIG_FILE=/path/to/pgraph/development.ini celery -A pgraph.tasks worker_
↪--loglevel=info
```

### 7. Running pserve from another venv session.

```
(venv)$ pserve /path/to/pgraph/development.ini
```

---

## Deploying app to Heroku

---

### 1. Prepare Heroku.

See also [“Getting Started with Python on Heroku”](#).

### 2. Cloning repository, and add remote repository.

```
$ git clone https://github.com/mkouhei/pgraph
$ cd pgraph
$ git remote add heroku <Heroku URL>
```

### 3. Enabling plugins

#### a. Heroku Postgres

```
$ heroku addons:create heroku-postgresql:hobby-dev
```

See also [Heroku Postgres](#)

---

**Note:** If you will use CloudAMQP instead of Heroku Postgres, execute this.

```
$ heroku addons:create cloudamqp:lemur
```

See also [CloudAMQP](#).

#### b. Memcached Cloud

```
$ heroku addons:create memcachedcloud:30
```

See also [Memcached Cloud](#)

#### c. New Relic APM

```
$ heroku addons:create newrelic:wayne
```

See also [New Relic APM](#)

3. git push to Heroku.

```
$ git push heroku master
```

4. Change the scale your app.

```
$ heroku ps:scale worker=1
```

## 3.1 Runnig test locally

### 3.1.1 Use PostgreSQL as Celery backend

1. Installing PostgreSQL, Memcached.

```
$ sudo apt-get install postgresql memcached
$ sudo su - postgres
$ psql
postgres=# CREATE USER pgraph WITH PASSWORD 'passwd';
postgres=# CREATE DATABASE pgraph OWNER pgraph;
postgres=# \q
```

---

**Note:** You can use `yrmcds` instead of `memcached`..

```
$ sudo apt-get install yrmcds
```

---

2. Cloning repository, and add remote repository.

```
$ git clone https://github.com/mkouhei/pgraph
$ cd pgraph
```

3. Generate requirements.txt.

```
$ python -m venv /path/to/heroku-venv
$ . /path/to/heroku-venv
$ pip install -e . -e .[heroku]
```

4. Running celery worker.

```
$ DATABASE_URL=postgres://pgraph:passwd@localhost:5432/pgraph CONFIG_FILE=heroku.
↪ini newrelic-admin run-program celery worker -c 1 -A pgraph.tasks --
↪loglevel=info
```

5. Execute `run` script.

```
$ DATABASE_URL=postgres://pgraph:passwd@localhost:5432/pgraph MEMCACHEDCLOUD_
↪SERVERS=127.0.0.1:11211 newrelic-admin run-program sh run
```

### 3.1.2 Use RabbigMQ as Celery backend

1. Installing RabbitMQ, Memcached.

```
$ sudo apt-get install rabbitmq-server memcached
```

**Note:** You can use `yrmcds` instead of `memcached`.

```
$ sudo apt-get install yrmcds
```

## 2. Cloning repository, and add remote repository.

```
$ git clone https://github.com/mkouhei/pgraph
$ cd pgraph
```

## 3. Generate requirements.txt.

```
$ python setup.py --version
$ pip install -e . -e .[heroku]
```

## 4. Running celery worker.

```
$ CONFIG_FILE=heroku.ini newrelic-admin run-program celery worker -c 1 -A pgraph.
↳tasks --loglevel=info
```

## 5. Execute `run` script.

```
$ MEMCACHEDCLOUD_SERVERS=127.0.0.1:11211 newrelic-admin run-program sh run
```



### 4.1 1.0.2 (2020-05-27)

- Updates JS libraries.
- Updates copyright.

### 4.2 1.0.1 (2020-05-27)

- Fixes runner

### 4.3 1.0.0 (2020-05-17)

- Changes to use py-deps 1.0.0.
- Updates dependencies.
- Support Python version 3.8 only.

### 4.4 0.5.1 (2016-04-09)

- Fixes view template layout.
- Removes backward compatibilities under IE 9.
- Updates JavaScript libraries.
- Updates locally instead of pyenv.
- Applies pytest-remove-stale-bytecode.

- Applies pytest-pylint.
- Ignores redefined-variable-type, wrong-import-order.
- Applies Python 3.5 on travis-ci.
- Fixes pep257 D211 violations.

## **4.5 0.5.0 (2015-10-09)**

- Supports Python 3.5.
- Adds showing version of running environments.
- Adjust svg\_width for PC web browser.
- Fixes unit test graph\_not\_found.

## **4.6 0.4.8 (2015-08-19)**

- Changes custom 404 not found pages.
- Closes #4, #5 error handlings 503 errors.
- Fixes INTERNALERROR occurs when run the tox.
- Changes mackerel-agent-plugin script for proxy.

## **4.7 0.4.7 (2015-08-15)**

- Changes setting SVG size from window InnerWidth, InnerHeight.
- Changes Sphinx theme to sphinx\_rtd\_theme.
- Adds mackerel metric plugin.

## **4.8 0.4.6 (2015-07-27)**

- Uses static\_path instead of static\_url.
- Changes example pathname.

## **4.9 0.4.5 (2015-07-22)**

- Adds latest\_version method, display latest version package.
- Applies override node link with link\_prefix.

## **4.10 0.4.4 (2015-07-19)**

- Adds invalid metadata handler.



## 4.11 0.4.3 (2015-07-16)

- Change the color of the nodes to give gradation by the dependency depth (py-deps>=0.5.2)

## 4.12 0.4.2 (2015-07-12)

- Changes pgraph.tasks.search (py-deps>=0.5.1).
- Fixes *None* redundant second argument of *dict.get()*.
- Fixes old-style string formatting.

## 4.13 0.4.1 (2015-07-03)

- Adds unit tests for backend\_config, config modules.
- Changes example to redirect to the graph of latest version of pgraph.
- Fixes responsive navbar, search form.
- Changes Demo site URL.

## 4.14 0.4.0 (2015-06-29)

- Revokes when tasks.gen\_dependency failed.
- Fixes extras\_requires for Heroku.
- Changes Celery backend for Heroku to PostgreSQL in default.
- Changes install requires pyramid\_celery to celery.

## 4.15 0.3.1 (2015-06-24)

- Applies GPLv3 and adds LICENSE file.

## 4.16 0.3.0 (2015-06-24)

- Supports cache backend using memcached.
- Adds side bar into layout.
- Changes for Heroku.
  - Updates setup configuration.
  - Applies New Relic.
  - Fixes Procfile and run.

## **4.17 0.2.0 (2015-06-19)**

- Integrates configurations locally and Heroku.
- Changes not use job queue if Package data exists in the cache.
- Enable to change Celery configuration using CELERY\_CONFIG variable and `.ini` file.
- Specify package version.
- Adds linkdraw configuration link.
- Adds search form at navigation bar.
- Changes searched view.
- Coverage  $\geq 90\%$ .
- Supports running on Heroku.

## **4.18 0.1.0 (2015-06-08)**

- Initial release.

## CHAPTER 5

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`