
Phoenix Documentation

Release 0.11

Birdhouse

May 11, 2023

Contents

1	Installation	3
2	Configuration	5
3	User Guide	7
3.1	Login	7
3.2	Dashboard	8
3.3	Processes	8
3.4	Monitor	11
3.5	My Account	12
3.6	Settings (admins only)	13
4	Indices and tables	17

Phoenix (the bird) *Phoenix is a long-lived bird that is cyclically regenerated or reborn.* (Wikipedia). [..]

Pyramid Phoenix is a web-application build with the Python web-framework [Pyramid](#). Phoenix has a user interface to interact with [Web Processing Services](#). The user interface gives you the possibility to *register Web Processing Services*. For these registered WPS services you can see which *Processes* are available. You are provided with a form page to enter the parameters to *execute a process (job)*. You can *monitor the jobs* and see the results.

Phoenix is installed using the [Conda](#) Python distribution and [Buildout](#).

CHAPTER 1

Installation

Start with downloading Phoenix with sources from github:

```
$ git clone https://github.com/bird-house/pyramid-phoenix.git
$ cd pyramid-phoenix
```

Create the `conda` environment and activate it:

```
$ conda env create -f environment.yml
$ conda activate pyramid-phoenix
```

Edit the configuration `custom.cfg` (see `custom.cfg.example`). For example change the admin password:

```
$ vim custom.cfg
# phoenix admin password
phoenix-password = qwerty
```

When you're finished, run `make install` to install Phoenix into the conda environment. The installation is using `buildout`:

```
$ (pyramid-phoenix) make install
```

By default phoenix will be installed into the folder `~/birdhouse`.

After successful installation you need to start the services:

```
$ (pyramid-phoenix) make start    # starts supervisor services
$ (pyramid-phoenix) make status  # shows status of supervisor services
```

Phoenix web application is available on `http://localhost:8081`.

Check the log file for errors:

```
$ tail -f ~/birdhouse/var/log/supervisor/phoenix.log
$ tail -f ~/birdhouse/var/log/supervisor/celery.log
```


CHAPTER 2

Configuration

You can configure Phoenix by editing `custom.cfg` in the Phoenix source folder:

```
$ cd pyramid-phoenix
$ vim custom.cfg
$ cat custom.cfg
```

```
[settings]
hostname = localhost
http-port = 8081
https-port = 8443
log-level = INFO
# phoenix admin password
phoenix-password = qwerty
```

By default Phoenix runs on localhost. The HTTP port 8081 is redirected to the HTTPS port 8443. If you want to use a different hostname/port then edit the default values in `custom.cfg`:

```
[settings]
hostname = localhost
http-port = 8081
https-port = 8443
```

To activate the GitHub login for external users you need to configure a GitHub application key for your Phoenix web application:

```
[settings]
# register at github: https://github.com/settings/applications/new
github-consumer-key = #####
github-consumer-secret = #####
```

See the [GitHub Settings](#) on how to generate the application key for Phoenix.

After any change to your `custom.cfg` you **need** to run `make install` again and restart the supervisor service:

```
$ make install  
$ make restart
```

The user guide explains how to use the Phoenix web application to interact with Web Processing Services.

- *Login*
- *Dashboard*
- *Processes*
- *Monitor*
- *My Account*
- *Settings (admins only)*
 - *Register a WPS service*

3.1 Login

Press the `Sign in` button in the upper right corner.



Sign In

The login page offers you several options to login to Phoenix.

Sign In

[Sign in with ESGF](#)

[Sign in with GitHub](#)

OR

Admin password *

If you have not configured your password yet then it is likely to be "qwerty"

[Sign In](#)

[Register for an account](#)

You can login using your GitHub account.

If you are Phoenix admin you can also enter the admin password here.

3.2 Dashboard

The dashboard shows some statistics about jobs and users.

PHOENIX Processes Monitor Help ▾ testuser ▾

Dashboard

Overview

Jobs

People



People

2



Jobs

5

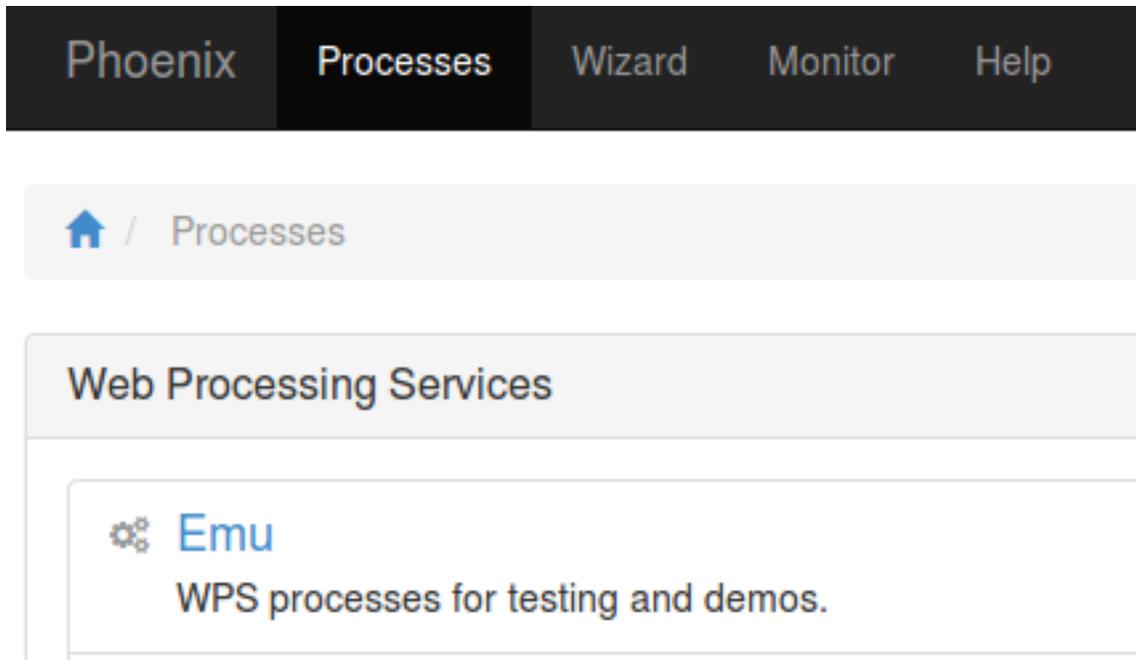


Web Processing Services

1

3.3 Processes

When you have registered WPS services you can run a process. Go to the Processes tab.



Choose one of your registered WPS services. You will get a list of available processes (WPS `GetCapabilities` request).

Phoenix Processes Wizard Monitor Help

Home / Processes / Emu

Description

WPS processes for testing and demos.

XML Provider: Birdhouse/Emu

Processes

- ⚙️ **Hello World 1.0**
Welcome user and say hello ...
- ⚙️ **Answer to Life, the Universe and Everything 2.0**
Numerical solution that is the answer to Life, Universe and Everything. T takes 7.5 milion years, but only a few seconds to give a response, with a

Choose one of these processes by using the `Execute` button.

In case of Emu you may try the `Hello World` process. You will then be prompted to enter your username:

Phoenix	Processes	Wizard	Monitor	Help
---------	-----------	--------	---------	------

[Home](#) / [Processes](#) / [Emu](#) / Hello World

Description

Welcome user and say hello ...

Inputs

Your name *

Please enter your name

Press the `Submit` button. When the process is submitted you will be shown your job list in `Monitor`.

3.4 Monitor

In `Monitor` all your running or finished jobs are listed. The list shows the status and progress of your jobs.

Job Monitor

This page shows the status of all your jobs.

My Jobs Public Private

All

Delete Make Public Make Private Set Favorite Unset Favorite

Process Status

Running 1 Finished 41 Matching 42

Sort C

<input type="checkbox"/>	Status	User	Process	Service	Caption	Finished	Duration	Labels	
<input type="checkbox"/>	20:00	Carsten Ehbrecht	cloud_taylor	copernicus	???	???	0:00:04	dev single async edit labels	Details Restart
<input type="checkbox"/>	✓	Carsten Ehbrecht	cchecker	hummingbird	???	less than 1 minute ago	0:00:02	dev single async edit labels	Details Restart

When a job has finished with success you can see the results by clicking the `Details` button.

Job Details

This page shows the job details and polls the status of a running job.

✓ cchecker

100%

ProcessSucceeded

Delete Job Restart Job

???

Ran for 0:00:02

PyWPS Process IOOS Compliance

dev single async

3 minutes ago

Checker finished

Runs the IOOS Compliance Checker tool to check datasets against compliance standards. Each compliance standard is executed by a Check Suite, which functions similar to a Python standard Unit Test. A Check Suite runs one or more checks against a dataset, returning a list of Results which are then aggregated into a summary. Development and maintenance for the compliance checker is done by the Integrated Ocean Observing System (IOOS).

Job Log

Inputs Outputs View as XML

```
1 0:00:02 0%: PyWPS Process cchecker accepted
2 0:00:02 100%: PyWPS Process IOOS Compliance Checker finished
```

If the result has a document (XML, text, NetCDF, ...) you can view or download this document with the `Download` button.

3.5 My Account

In `My Account` you can change your user settings (user name, organisation, openid, ...).

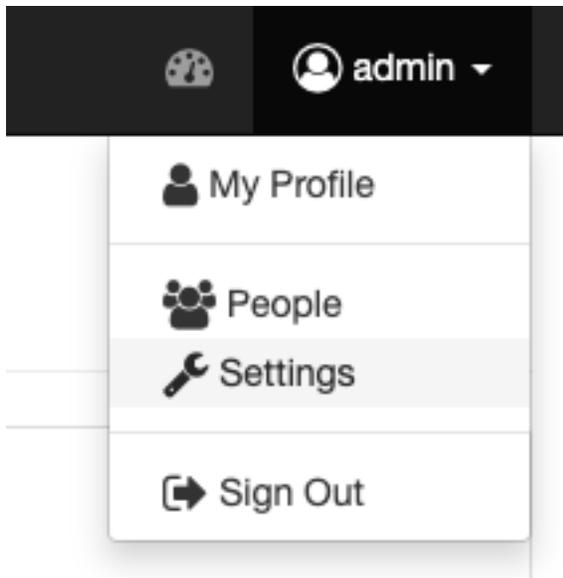
The screenshot shows the Phoenix user profile page for 'testuser'. The navigation bar at the top includes 'PHOENIX', 'Processes', 'Monitor', and 'Help'. The user's name 'testuser' is visible in the top right. The page title is 'testuser'. On the left, there are three menu items: 'Profile' (highlighted in blue), 'Personal access token', and 'Group Permission'. The main content area is titled 'Profile' and contains several input fields: 'Your Name' (with 'testuser' entered), 'EMail', 'Organisation', and 'Notes'. A green 'Update Profile' button is located at the bottom of the form.

You can also see your current access token which you can use to access a protected WPS service directly.

The screenshot shows the Phoenix user profile page for 'testuser', specifically the 'Personal access token' section. The navigation bar and user name are the same as in the previous screenshot. The left menu has 'Personal access token' highlighted in blue. The main content area is titled 'Personal access token' and features a 'Refresh Token' button. Below this, the 'Access Token' is displayed as a long alphanumeric string. The 'Expires at' date and time are also shown.

3.6 Settings (admins only)

When you are logged-in as admin user you have the `Settings` page. Here you can make administrative changes and monitor services.



3.6.1 Register a WPS service

Open the `Settings/Services` page. Here you can see which services are registered in the catalog. All these services are known and usable by Phoenix.



To add a new WPS service, press the `Register a new Service` button and enter the WPS URL in the field `Service URL`.

For example, to register Emu WPS:

`http://localhost:5000/wps`

PHOENIX Processes Monitor Help ▾   admin ▾

[Home](#) / [Settings](#) / [Services](#) / Register New Service

Register New Service

Service URL

Add URL of service (WPS). Example: http://localhost:5000/wps

Service Title

An optional service title. The title is used as a display name for the service. If a title is not provided it will be taken for the service metadata.

Public access?

Check this option if your service has no access restrictions.

CHAPTER 4

Indices and tables

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