

Foswiki > FLAO Web > NewHowTo (26 May 2015, LucaFini)

Installation, deployment and usage

[Installation, deployment and usage](#)

[prepare.py](#)

[deploy.py](#)

[flao.py](#)

[Implementation Notes](#)

[AdOpt servers](#)

[Runtime accounts on AdOpt servers](#)

[More details about deploy.py](#)

[More details about flao.py](#)

In the following page we resume the usage of procedures to install FLAO software, deploy on the AdOpt servers and start stop the software subsystem.

The management of FLAO software is performed by three procedures (which can be found at the root of the source code directory tree as checked out from the SVN repository).

- **prepare.py**. Environment setup for creating an installation to be used for software development and tests. The environment is suitable to run a single subsystem (`adsec / wfs, right / left`).
- **deploy.py**. To deploy a runtime build of FLAO software onto the four subsystem servers. The deployed build is named TEST and is suitable for running the AO system at the telescope. The same procedure will be used at the end of tests to release (freeze) the build after testing.
- **flao.py**. Procedure to be used by the TO to manage the FLAO software, i.e.: select an available build, start/stop processes and the like. This procedure is intended to be usable as a standalone tool, i.e.: it does not need any other FLAO software component and can be run on any computer with an SSH access to the FLAO servers.

Here follows a detailed description of the procedures and their use.

prepare.py

The purpose of the procedure is to setup a proper environment for FLAO software development (and/or maintenance) and is usually run once after cychecking out a fresh version of the FLAO source code. Here follows the help page which comes out with `python prepare.py`.

FLAO Supervisor environment setup procedure. Version 2.4

L.Fini, April 2015.

Usage:

```
python prepare.py check           Check environment
python prepare.py make           Create environment
python prepare.py set adsec|wfs left|right Set installation target
```

This procedure operated on the local environment to be used for build and tests.

The actual installation of "science ready" FLAO system must be done with:

```
python deploy.py
```

The procedure provides three subcommands:

- **make.** The `make` subcommand creates (or checks) the environment for development, more in details:
 - Creates local runtime directories: `~/aoroot`, `~/aolog`, `~/aomeasures`.
 - Creates a private/public key file pair to be used by `ssh` and stores them into `~/ .ssh` directory of the current user.
 - Creates an identity code for the specific build.
 - Creates a `bash` compatible file for setting up environment variables: `flao_environment.sh==`

The environment definition file must be explicitly executed to take effect. If desired it can be executed from the `bash` startup procedure (usually `~/ .bashrc=`). This is enough to be able to make and install the FLAO code, configuration and calibration files.

Note 1: when installing and running the development build, executable, configuration and runtime files are all stored in users' specific directories.

Note 2: In order to be able to execute the FLAO software, the specific server identity (i.e.: `adsec / wfs` and `left / right`) must be established with the `set` subcommand.

- **check.** The `check` subcommand verifies the environment to check that it is properly set for compilation and installation of the FLAO software.
- **set.** The `set` subcommand defines the specific identity of the current build so that it can operate as one of the AdOpt servers (i.e.: either as `adsec` or as `wfs` and with the required side (`right` or `left`)).

The effect of the procedure is to create the required symbolik link in the ADOPT_ROOT runtime directory tree and to generate a new version fo the environment setup file `flao_environment.sh` suitable for running the code with the proper identity.

Note: the environment setup procedure must be explicitly executed in order to have effect.

deploy.py

The deploy procedure has the purpose to deploy all required runtime files onto the four AdOpt servers. The procedure assumes that the servers have been already configured for the purpose. Each server must have an account `flao` which will be used for software deployment and to run the FLAO software.

The procedeuure is intended to be launched from the root of the FLAO software source tree after properly setting the environment as defined in the `flao_environment.sh` file generated by `prepare.py`. Moreover the full software generation sequence: `make`, `make install`, `make install_conf` and `make_install_calib` must have been completed.

Here follows the relevant items from the help page as displayed by `python deploy.py`:

```
FLAO Supervisor deployment procedure. Version 1.6
```

```
L.Fini, May 15, 2015
```

```
Usage:
```

```
python deploy.py [-v] command [args]
```

```
-v: Verbose command mode (for debug)
```

```
Commands:
```

```
key: Send SSH public key to targets (to be done once)
```

```
rel: Release the TEST installation
```

```
test: Deploy a test installation (possibly overwriting a previous one)
```

The procedure provides four subcommands:

- `key`. Transmits the ssh public key specific for FLAO management to the four AdOpt servers. Usually this will require to specify the password defined for the `flao` user on the four servers. This operation must be done once, before a build is deployed for the first time.
- `test`. Deploy current build as TEST build. All the required files will be copied onto proper directories on

the four targets and required links will be created. The deployment can be repeated, in which case the new build is written over the previous one.

Note: After the deployment the current setup of server is not modified, i.e.: the previously active build (if any) is still active. The selection of active build can be done with the [flao.py](#) procedure.

- `rel`. Releases the TEST build as an *science ready* build. The name is generated automatically and is of the form: 2105X (current year plus a single letter).
- `lnk`. Redoes the links in the specified build (for test purposes)

The `deploy.py` also provides a copy of commands provided by `flao.py`: the description can be found in the [procedure](#).

`flao.py`

The purpose of the procedure is to provide the Telescope Operator with the required commands to manage the FLAO supervisor software. Because it is not intended to be used exclusively by software developer/maintainers, it is described in a [dedicated document](#).

Implementation Notes

AdOpt servers

The procedures described above have the specification of AdOpt servers and their roles built in. If server names will change in the future, the related tables into the file `flao.py` must be changed accordingly. The relevant table name is: MTGRAHAM_HOSTS.

Runtime accounts on AdOpt servers

The procedures rely on proper configuration of a user account on the four Adopt servers. The userid must be: `flao`, with any suitable password. A specific public key will be stored in the `.ssh` directory of the account when using the `key` subcommand of `deploy.py` procedure. After that all remote operations will be done using the key, i.e. without the need to specify a password.

More details about `deploy.py`

The `deploy.py` procedure has some more options useful for developers and debuggers. Here follows the additional help output which is shown with: `python deploy.py -h`

ADDITIONAL INFO FOR DEVELOPERS

The procedure accepts an additional option:

```
-a:  select a list of target servers available at Arcetri to be used
      for tests instead of the servers used for science ready operations.
```

The procedure also accept additional commands:

```
fake:  Make a FAKE build (for debugging)
lnk:   Redo build links
```

Note: The FAKE build just contains a fake implementation of the FLAO process launching python scripts.

More details about `flao.py`

The `flao.py` procedure has some more options useful for developers and debuggers. Here follows the additional help output which is shown with: `python flao.py -h`

ADDITIONAL INFO FOR DEVELOPERS

The procedure accepts an additional option:

```
-a:  select a list of target servers available at Arcetri to be used
      for tests instead of the servers used for science ready operations.
```

[Edit](#) | [Attach](#) | [Print version](#) | [History: r4 < r3 < r2 < r1](#) | [Backlinks](#) | [View wiki text](#) | [Edit wiki text](#) | [More topic actions](#)

Topic revision: r4 - 26 May 2015, LucaFini

Copyright © by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding Foswiki? Send feedback

