Installation of BEAMPY 5.0.5

This document describes how to install BEAMPY, the SeaDAS 7/ BEAM 5 – Python Interface. This installation procedure was tested with SeaDAS 7.3 on Mac OS X Yosemite (10.10). It has not yet been tested with BEAM 5.0 or SNAP (the new successor to BEAM).

This need only be done once per computer. (Multiple installations SeaDAS can share this.) Note this has not been tested with an installation of both SeaDAS and BEAM on the same computer.

**NOTE:** Once sections I – III are complete, SeaDAS 7 users who wish to develop Python code using BEAMPY need only set the environment variables outlined in section IV.

**Overview**

* Install SeaDAS 7.x
* Setup development environment.
* Build and install the Java-Python interface jpy (The beampy build/setup for jpy on Mac OS X doesn’t work.)
* Build and install the beampy interface
* Test beampy

**I. Prerequisites (Order is important!)**

1. Mac OS X Yosemite 10.10.x
2. SeaDAS 7.x or Beam VISAT 5.0
3. Python V2.7 with numpy installed (e.g., sudo port install py27-numpy)

Check that numpy imports ok:

dhcp164-15:~ seadas7$ python

Python 2.7.11 (default, Dec 5 2015, 23:52:42)

[GCC 4.2.1 Compatible Apple LLVM 6.1.0 (clang-602.0.53)] on darwin

>>> import numpy as np

>>> quit()

d. Xcode and Xcode Command Line Tools installed

Make sure you’re using Xcode compiler, i.e., gcc invokes /usr/bin/gcc

dhcp164-15:beampy-tests seadas7$ which gcc

/usr/bin/gcc

e. Java Development Kit JDK 1.8 installed

<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

f. git source code management software installed (sudo port install git)

g. Maven3 Java Build automation software installed: (sudo port install maven3)

**II. Build and install the Java-Python Interface jpy**

1. Get latest jpy sources:

git clone https://github.com/bcdev/jpy.git

For more details, see: <http://jpy.readthedocs.org/en/latest/install.html#getting-the-sources>

2. Add environment variables to your ~/.bash\_profile and ~/.bashrc. Then log in to a new terminal window and check, e.g., ls $JDK\_HOME, etc.

export JAVA\_HOME=$(/usr/libexec/java\_home)

export JDK\_HOME=$(/usr/libexec/java\_home)

# Change BEAM\_HOME according to your installation of SeaDAS or BEAM)

export BEAM\_HOME=/Applications/seadas-7.3

3. Edit sudoers so that JDK\_HOME, JAVA\_HOME, and BEAM\_HOME environment variables are inherited from your environment when you “sudo”"

sudo visudo

(Add the following lines with the other Defaults”)

Defaults env\_keep += “JAVA\_HOME"

Defaults env\_keep += "JDK\_HOME"

Defaults env\_keep += “BEAM\_HOME"

4. Build and install jpy for all users

sudo python setup.py --maven build install

(Need to run as sudo so that the jpy module is installed in the Python site-packages, e.g., /Library/Python/2.7/site-packages/)

4. Test Python import of jpy

dhcp210-15:~ seadas7$ python

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.2.1 Compatible Apple LLVM 6.0 (clang-600.0.39)] on darwin

>>> import jpy

**III. Build and install the Beam-Python Interface beampy**

1. In SeaDAS 7, and BEAM 5 the BEAM Python Interface module is automatically istalled. Check in SeaDAS Help > Module Manager … that “BEAM Python Interface” is listed in “Installed Modules”. If not, click on “Available Modules”, select "BEAM Python Interface”, Install, OK. For additional details, consult:

https://github.com/bcdev/beam/blob/master/beam-python/src/main/resources/README.md

2. Build and install beampy for all users

cd $BEAM\_HOME/modules/beam-python-5.0.5-SNAPSHOT/beampy

sudo python setup.py install

Ignore errors at the end:

No local packages or download links found for jpy>=0.8-SNAPSHOT

error: Could not find suitable distribution for Requirement.parse('jpy>=0.8-SNAPSHOT’)

For additional details, see <http://oceancolor.gsfc.nasa.gov/forum/oceancolor/topic_show.pl?tid=5988>

(Need to run as sudo so that the beampy module is installed in the Python site-packages, e.g., /Library/Python/2.7/site-packages/)

3. Test Python import of beampy

dhcp210-15:~ seadas7$ python

Python 2.7.10 (default, Jul 14 2015, 19:46:27)

[GCC 4.2.1 Compatible Apple LLVM 6.0 (clang-600.0.39)] on darwin

>>> import beampy

>>> quit()

4. Run beampy Test Suite

a. Download MER\_RR\_\_1P.N1 from <http://www.brockmann-consult.de/cms/web/beam/meris-products> . Rename unzipped file to MER\_RR\_\_1P.N1

b. cd beampy-tests

dhcp210-15:beampy-tests ericrehm$ pwd

/Applications/seadas-7.0/modules/beam-python-5.0.5-SNAPSHOT/beampy-tests

dhcp210-15:beampy-tests ericrehm$ ls -1

MER\_RR\_\_1P.N1 .

beampy\_mem\_test.py

beampy\_perf\_test.py

beampy\_product\_test.py

c. Run tests

For example:

dhcp210-15:beampy-tests ericrehm$ python beampy\_perf\_test.py

('Band.readPixels(): w =', 1121L, ', dtype=np.float32:', 2081L, 'calls in', 601.2568473815918, 'ms, that is ', 0.2890657920103807, 'ms per call')

.('Band.readValidMask(): w =', 1121L, ', dtype=np.bool:', 2081L, 'calls in', 353.5301685333252, 'ms, that is ', 0.16996642717948326, 'ms per call')

.

-----------------------------------------------------------------

Ran 2 tests in 1.831s

OK

**IV. Steps for BEAMPY Users**

Only the following steps are required for SeaDAS / BEAM users :

1. Add environment variables to your ~/.bash\_profile and ~/.bashrc. Then log in to a new terminal window and check them, e.g., ls $JDK\_HOME, etc.

export JAVA\_HOME=$(/usr/libexec/java\_home)

export JDK\_HOME=$(/usr/libexec/java\_home)

# Change BEAM\_HOME according to your installation of SeaDAS or BEAM.)

export BEAM\_HOME=/Applications/seadas-7.3

2. Test that you can access the Python jpy and beampy modules by running the beampy-tests:

cd ~

mkdir beampy-tests

cd beampy-tests

cp $BEAM\_HOME/modules/beam-python-5.0.5-SNAPSHOT/beampy-tests/\* .

dhcp164-15:beampy-tests ericrehm$ python beampy\_perf\_test.py

('Band.readPixels(): w =', 1121L, ', dtype=np.float32:', 2081L, 'calls in', 785.9411239624023, 'ms, that is ', 0.3778563095973088, 'ms per call')

.('Band.readValidMask(): w =', 1121L, ', dtype=np.bool:', 2081L, 'calls in', 297.59907722473145, 'ms, that is ', 0.14307647943496704, 'ms per call')

.

----------------------------------------------------------------

Ran 2 tests in 1.747s