Step-by-step Guide for Installing Meggie

This is the simple installation guide for Meggie. For shorter, more technical instructions and requirements, see requirements.txt in the *Docs* directory of the Meggie installation package.

General Information About Requirements

Running Meggie requires you to install several external programs and add-ons. These include both commercial and open source software. Meggie works without installing commercial software, but with reduced features.

The requirements are the following:

- Enthought Python Distibution
- MNE
- MNE-Python
- PyQt4, which requires SIP
- Meggie itself

If the you need to install the packages from scratch, you need the following:

- A Linux system to install Meggie on. Meggie and its requirements have been tested on Fedora 16, but later Fedoras or other Linux distributions may or may not work.
- A network connection to download the packages.
- Some way to extract zip and tar.gz packages. The file managers of Linux desktop environments generally have a graphical application for this.

PLEASE NOTE: some or all of the required programs may already have been installed or at least downloaded by your colleagues or technical support personnel. In that case, you may skip some of the registragion/download/installation steps. Ask around to avoid unnecessary work.

1. Installing Enthought Python Distribution (EPD)

PLEASE NOTE: EPD has recently been renamed to Enthought Canony, and the installation process has been revised. Meggie has not been been tested with Canopy and thus we cannot (yet) provide installation instructions for those how wish to use it.

1.1 Registering and downloading EPD

Installing EPD is one way to get the numerous general scientific computing related add-ons required by Meggie. EPD is the also recommended by the maintainers of MNE

EPD is free for academic usage, but requires registration. Go to the page https://www.enthought.com/products/canopy/academic/ and click “Request your academic license”. After following the procedure and acquiring the license, sign in to the site and go to address https://www.enthought.com/products/epd/package-index/. The required installer can be found in the “installers” directory . The required installer is

epd-7.3-2-rh5-x86_64.sh (for 64-bit Linux) OR

epd-7.3-2-rh5-x86.sh (for 32-bit Linux)

Download the desired installer to a directory of your choice.

1.2 Installing EPD

Open a terminal program in the directory where you download the EPD installer. In the terminal, run the following command:

```
chmod +x <packagename>
```

For example, if you downloaded the installer for 64-bit Linux, the command would be the following:

```
chmod +x epd-7.3-2-rh5-x86_64.sh
```

Then run the installer with superuser permissions:

```
sudo ./epd-7.3-2-rh5-x86_64.sh
```

The installer will prompt you for an installation directory. There are no specific requirements for the location of directory, so pick whatever seems suitable or is recommended by your system administrator.

1.3 Setting up EPD Python

The following installation steps require you to use the EPD Python as the command line Python interpreter. It is likely that your system already has a different Python interpreter installed, so it is essential not to mix the two interpreters. The following instructions explain how to set up EPD as the command line Python interpreter for your system, so that all Python programs started from the command line use EPD Python instead of the system default Python.

Open a terminal program and open the .bashrc file in your home directory:

```
nano /home/<yourusername>/.bashrc
```

Add the following line to the end of the file:

```
alias python='<pathToEPDInstallationDirectory>bin/python'
```

For example, if you installed the EPD to directory /usr/local/bin/epd/, the line to add would be:

```
alias python='/usr/local/bin/epd/bin/python'
```

Log out and log back in to actually register the change.

IMPORTANT! After logging in, open the terminal and run the command “python” in it – just write “python” (without the quotation marks) in the terminal and press ENTER . The terminal should show you output that contains the words “Enthought Python Distribution”. If not, DO NOT continue with installation. Ask your sysadmin for help.

2. Installing MNE

MNE is the actual software package required for analyzing the magnetoencephalography (MEG) and electroencephalography (EEG) data. From the plain MNE package, Meggie only uses mne_browse_raw for viewing the raw data.

MNE, too, requires registration for download. Go to http://martinos.org/mne/ , click “Download MNE” and follow the instructions. Meggie has been tested with following version:

```
MNE-2.7.0-3106-Linux-x86_64
```

After downloading, follow the instructions at http://martinos.org/mne/manual/AppInstall.html to install and configure MNE. Meggie will automatically detect the MNE installation and behave accordingly.

PLEASE NOTE: Meggie doesn't do source space analysis, therefore not requiring Freesurfer for its functionality. If you don't need MNE features related to source space analysis, you don't need to install Freesurfer.

3. Installing MNE-Python

Mne-Python includes Python modules and scripts for analyzing MEG data. Meggie uses Mne-Python for nearly all of its functionality.

You can get the installation package from https://github.com/mne-tools/mne-python . Click the “ZIP” button in the top left corner of the page and save the zip file into a directory of your choice. Extract the archive (file managers usually have the option – right click the package with the mouse), go to the extracted mne-python directory and run

```
sudo <pathToEPDInstallationDirectory>bin/python setup.py install
```

4. Installing PyQt4

PyQt4 is an user interface library used by various applications, including Meggie. It requires of installation of few packages before it can itself be installed.

4.1 Getting the required packages

If you don't have development tools installed in your system, you need to start with installing them. In Red Hat and Fedora this is achieved by running the following command in the terminal:

```
sudo yum groupinstall "Development Tools" "Development Libraries"
```

You can the proceed by getting the following packages:

- Qt4 developer libraries. Can be installed via the Linux distributions' package managers. In Red Hat and Fedora 16 this includes running the following command in the terminal:

```
sudo yum install qt4-devel
```

- A stable of SIP package. See http://www.riverbankcomputing.com/software/sip/download and pick the Linux source.
- The PyQt4 package itself, from http://www.riverbankcomputing.com/software/pyqt/download (pick the Linux source)

Extract the packages to directories of your choice.

4.2 Installing the packages

Start with installing SIP, a requirement for PyQt4. Go to the directory you extracted the SIP package to and run in the terminal

```
sudo <pathToEPDInstallationDirectory>bin/python configure.py
```

After the script has finished running, run

```
sudo make
```

And after finishing, run

```
sudo make install
```

You can then finish the installation by navigating to the directory you extracted the PyQt4 to, and running

```
sudo <pathToEPDInstallationDirectory>bin/python configure.py -q
/usr/bin/qmake-qt4
```

PLEASE NOTE: the location of qmake-qt4 depends on the Linux distribution. You can try searching for it with the command

```
locate qmake-qt4
```

And then

```
sudo make
```

and then

```
sudo make install
```

5. Getting Meggie

Meggie is currently available only on the Hoksotin project CD. Browse to the application directory on the CD and copy the meggie.zip to a directory of your choice.

6. Running Meggie

Meggie itself doesn't require any installation. Just go to the root directory of Meggie directory and run in the terminal:

```
python run.py
```