Pull out all of the linux commands

I need to understand each one – to fully install opencog

How many commands are there?

|  |  |  |
| --- | --- | --- |
| OpenCog Linux Command |  | Description – so I understand it |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Building OpenCog

Namespaces

* [Page](https://wiki.opencog.org/w/Building_OpenCog)
* [Discussion](https://wiki.opencog.org/w/Talk:Building_OpenCog)

Page actions

* [Read](https://wiki.opencog.org/w/Building_OpenCog)
* [View source](https://wiki.opencog.org/wikihome/index.php?title=Building_OpenCog&action=edit)
* [History](https://wiki.opencog.org/wikihome/index.php?title=Building_OpenCog&action=history)

GNU/Linux is the primary development platform for OpenCog. The reference configuration used for continuous integration is Ubuntu, starting from 14.04. Although OpenCog should be compatible with any GNU/Linux distribution, Ubuntu is the recommended platform for development.



Contents

* [1 Setting Development Environment](https://wiki.opencog.org/w/Building_OpenCog#Setting_Development_Environment)
  + [1.1 GNU/Linux](https://wiki.opencog.org/w/Building_OpenCog#GNU.2FLinux)
    - [1.1.1 Manual](https://wiki.opencog.org/w/Building_OpenCog#Manual)
    - [1.1.2 Automatic](https://wiki.opencog.org/w/Building_OpenCog#Automatic)
      * [1.1.2.1 Ubuntu](https://wiki.opencog.org/w/Building_OpenCog#Ubuntu)
      * [1.1.2.2 Raspbian](https://wiki.opencog.org/w/Building_OpenCog#Raspbian)
      * [1.1.2.3 Other GNU/Linux Distributions](https://wiki.opencog.org/w/Building_OpenCog#Other_GNU.2FLinux_Distributions)
  + [1.2 Virtual Machine](https://wiki.opencog.org/w/Building_OpenCog#Virtual_Machine)
    - [1.2.1 Windows setup](https://wiki.opencog.org/w/Building_OpenCog#Windows_setup)
    - [1.2.2 Mac setup](https://wiki.opencog.org/w/Building_OpenCog#Mac_setup)
    - [1.2.3 Linux setup](https://wiki.opencog.org/w/Building_OpenCog#Linux_setup)
    - [1.2.4 Run Vagrant](https://wiki.opencog.org/w/Building_OpenCog#Run_Vagrant)
  + [1.3 Containers](https://wiki.opencog.org/w/Building_OpenCog#Containers)
  + [1.4 Other OS](https://wiki.opencog.org/w/Building_OpenCog#Other_OS)
* [2 Building](https://wiki.opencog.org/w/Building_OpenCog#Building)
* [3 Updating Dependencies](https://wiki.opencog.org/w/Building_OpenCog#Updating_Dependencies)
* [4 Persistance](https://wiki.opencog.org/w/Building_OpenCog#Persistance)
  + [4.1 RocksDB](https://wiki.opencog.org/w/Building_OpenCog#RocksDB)
  + [4.2 Network serving](https://wiki.opencog.org/w/Building_OpenCog#Network_serving)
  + [4.3 PostgreSQL](https://wiki.opencog.org/w/Building_OpenCog#PostgreSQL)

Setting Development Environment

These are options for setting development environment for working on opencog and it's subprojects.

GNU/Linux

Manual

If you prefer to manually install opencog and its dependencies, jump directly to the [Building](https://wiki.opencog.org/w/Building_OpenCog#Building) Section, and install whatever library CMake complains about. The only mandatory library is [boost](http://www.boost.org/), the others depend on the projects and components that you wish to build.

Otherwise, for automatic installation, follow the instructions below.

Automatic

Ubuntu

For a quick start using Ubuntu version >= 16.04, use octool script.

* get it by running

sudo curl -L http://raw.github.com/opencog/ocpkg/master/ocpkg -o /usr/local/bin/octool &&\

sudo chmod +x /usr/local/bin/octool &&\

octool

* Options that are available can be displayed by running

octool -h

* To install all dependencies necessary to build OpenCog run

octool -rdcpav -l default

# Optional: Add -s for installing dependencies for haskell binding.

* To install all dependencies necessary to build AtomSpace and MOSES run

octool -rdcv

# Optional: For atomspace add -s for installing dependencies for haskell binding.

* To install all dependencies necessary to build Cogutil run

octool -rdv

Raspbian

For a quick start using Raspbian OS on RPI3 board, use octool\_rpi script.

* get it by running

wget https://raw.githubusercontent.com/opencog/ocpkg/master/octool\_rpi.sh && chmod +rx octool\_rpi.sh

The script is suppose to run both on the cross-compiling PC and the RPI3 running Raspbian. More info [here](https://github.com/opencog/opencog_rpi/blob/master/README.md)

* To Install base dependencies and install opencog with pre-compiled binaries (it may be outdated) on RPI3 running Raspbian

./octool\_rpi.sh -do

* To cross compile on a much larger computer. This option cross-compiles opencog along with its dependencies (cogutil and atomspace) and creates a debian package.

./octool\_rpi.sh -tc

**For building opencog or updating it's dependencies scroll to the sections below**

Other GNU/Linux Distributions

On systems that have other GNU/Linux OS installed, use the appropriate script for installing system dependencies. Distros which have dependency installation scripts are,

* [Debian](https://raw.githubusercontent.com/opencog/ocpkg/master/debian/install-debian-dependencies.sh)
* [ArchLinux](https://raw.githubusercontent.com/opencog/ocpkg/master/install-archlinux-dependencies.sh)
* [Fedora](https://raw.githubusercontent.com/opencog/ocpkg/master/install-fedora-dependencies.sh)
* [openSUSE](https://raw.githubusercontent.com/opencog/ocpkg/master/install-opensuse-dependencies.sh)

Virtual Machine

For using Ubuntu in virtual machine managed by vagrant for <https://github.com/opencog/opencog> repo, first install the tools for your os and then follow instruction below under **Run Vagrant**.

[Windows setup](https://wiki.opencog.org/w/Building_OpenCog_in_a_Linux_Virtual_Machine_on_Windows)

[Mac setup](https://wiki.opencog.org/w/Building_OpenCog_in_a_Linux_Virtual_Machine_on_Mac_OS_X)

Linux setup

Install VirtualBox, git and vagrant. See [here](https://www.vagrantup.com/docs/installation/) on instructions on vagrant installation.

Run Vagrant

**The virtual machine created has all the dependencies required for working on cogutils, atomspace, moses, and opencog.**

If you are new to vagrant, see <https://www.vagrantup.com/docs/why-vagrant/> or <https://www.vagrantup.com/docs/getting-started/> if you like to understand what vagrant is. Understanding the working of vagrant is not required for contributing to opencog.

The common steps after installing VirtualBox, git and Vagrant for your os and running a bash(on windows use the equivalent powershell commands) shell are:

mkdir opencog\_repos

cd opencog\_repos

git clone https://github.com/opencog/opencog

# You can clone other repos if you so choose

ln -s $PWD/opencog/lib/Vagrantfile Vagrantfile

vagrant up

vagrant ssh

# The folder from which your run 'vagrant up' is accessiable at

# '/home/vagrant/opencog' on sshing into the virtual machine.

# Do your thing

exit

vagrant halt

**For building opencog or updating it's dependencies scroll to the sections below**

Containers

For using containers managed by docker, follow instructions [here](https://github.com/opencog/docker/blob/master/opencog/README.md).

**For building opencog or updating its dependencies scroll to the sections below**

Other OS

* [MacPorts](http://wiki.opencog.org/wikihome/index.php/Building_OpenCog_on_Mac_OS_X_with_MacPorts) (not tested for a while)
* [Cygwin](http://wiki.opencog.org/wikihome/index.php/Building_OpenCog_on_Cygwin)

Building

Build OpenCog using the following commands after moving to the root of the three cloned directories ([cogutil](https://github.com/opencog/cogutil), [atomspace](https://github.com/opencog/atomspace), optionally [moses](https://github.com/opencog/moses), and [opencog](https://github.com/opencog/opencog), in that order) or the corresponding directories in your vagrant/docker setup:

mkdir build

cd build

cmake ..

make -j$(nproc)

sudo make install

Provided you have Python2 and Python3 development libraries installed, and only want to build atomspace/opencog python-bindings against Python2, instead of the default Python3, run the following commands

mkdir build

cd build

cmake -DCMAKE\_DISABLE\_FIND\_PACKAGE\_Python3Interp=TRUE ..

make -j$(nproc)

sudo make install

If you are using octool from the git clone directory, run

octool -b

Updating Dependencies

To update your opencog's sub-project installations(cogutil, link-grammar and atomsapce), as they are in a continues state of development, you will need to run these series of commands, once/multiple times a week/day depending on what type of fun you are having with opencog, from your git clone.

If you are using octool, run

octool -cal

Persistance

For saving and sharing the contents of the AtomSpace. See [StorageNode](https://wiki.opencog.org/w/StorageNode" \o "StorageNode) for a general overview.

RocksDB

RocksDB is recommended, vs. Postgres: its faster, smaller, and there's nothing to configure. (Installing and configuring Postgres is hard.)

See <https://github.com/opencog/atomspace-rocks> for install instructions.

See [RocksStorageNode](https://wiki.opencog.org/w/RocksStorageNode" \o "RocksStorageNode) for usage details.

Network serving

Multiple AtomSpaces can share Atoms with one-another, over the network. This uses exactly the same API asthe toher StorageNodes. See [CogStorageNode](https://wiki.opencog.org/w/CogStorageNode" \o "CogStorageNode) and [CogServer](https://wiki.opencog.org/w/CogServer" \o "CogServer).

PostgreSQL

See instructions [here](https://github.com/opencog/atomspace/tree/master/opencog/persist/sql#install-setup-and-usage-howto)

* If you are using octool then use -rd option for installing dependencies.
* If you are using docker full configuration are done for you.

See [PostgresStorageNode](https://wiki.opencog.org/w/PostgresStorageNode" \o "PostgresStorageNode) for usage details.