Ubuntu install of ROS Indigo

Configure your Ubuntu repositories

Configure your Ubuntu repositories to allow "restricted," "universe," and "multiverse."

Setup your sources.list

Setup your computer to accept software from packages.ros.org. ROS Indigo **ONLY** supports Saucy (13.10) and Trusty (14.04) for debian packages.

- `sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'`

Set up your keys

- `sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net --recv-key 421C365BD9FF1F717815A3895523BAEEB01FA116`

You can try the following command by adding **:80** if you have **gpg: keyserver timed out** error due to a firewall

- `sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key 421C365BD9FF1F717815A3895523BAEEB01FA116`

Installation

First, make sure your Debian package index is up-to-date:

- `sudo apt-get update`

o If you are using Ubuntu Trusty **14.04.2** and experience dependency issues during the ROS installation, you may have to install some additional system dependencies.

- ⚠ **Do not install these packages if you are using 14.04, it will destroy your X server**:

```
sudo apt-get install xserver-xorg-dev-lts-utopic mesa-common-dev-lts-utopic libxatracker-dev-lts-utopic libopenvg1-mesa-dev-lts-utopic libgles2-mesa-dev-lts-utopic libgles1-mesa-dev-lts-utopic libgl1-mesa-dev-lts-utopic libgbm-dev-lts-utopic libegl1-mesa-dev-lts-utopic
```

⚠ **Do not install the above packages if you are using 14.04, it will destroy your X server**

o Alternatively, try installing *just* this to fix dependency issues:

- `sudo apt-get install libgl1-mesa-dev-lts-utopic`

For more information on this issue see this answers.ros.org thread or this launchpad issue

There are many different libraries and tools in ROS. We provided four default configurations to get you started. You can also install ROS packages individually.

- **Desktop-Full Install: (Recommended)** : ROS, rqt, rviz, robot-generic libraries, 2D/3D simulators and 2D/3D perception

  Indigo uses Gazebo **2** which is the default version of Gazebo on Trusty and is recommended. If you would like to instead use a newer version of Gazebo (5, 6 or 7), refer to these instructions on the Gazebo site. Note that installing a newer version of Gazebo will require you to build dependent packages (such as `turtlebot_gazebo`) to be built from source. See also Using a specific Gazebo version with ROS.

  ```
  o  sudo apt-get install ros-indigo-desktop-full
  ```

  or click here

  Desktop Install: ROS, rqt, rviz, and robot-generic libraries

  ```
  o  sudo apt-get install ros-indigo-desktop
  ```

  or click here

  ROS-Base: (Bare Bones) ROS package, build, and communication libraries. No GUI tools.

  ```
  o  sudo apt-get install ros-indigo-ros-base
  ```

  or click here

  Individual Package: You can also install a specific ROS package (replace underscores with dashes of the package name):

  ```
  o  sudo apt-get install ros-indigo-PACKAGE
  ```

  e.g.

  ```
  sudo apt-get install ros-indigo-slam-gmapping
  ```

To find available packages, use:

```
apt-cache search ros-indigo
```

Initialize rosdep

Before you can use ROS, you will need to initialize `rosdep`. `rosdep` enables you to easily install system dependencies for source you want to compile and is required to run some core components in ROS.

```
sudo rosdep init

rosdep update
```

Getting rosinstall

rosinstall is a frequently used command-line tool in ROS that is distributed separately. It enables you to easily download many source trees for ROS packages with one command.

To install this tool on Ubuntu, run:

```
sudo apt-get install python-rosinstall
sudo apt-get install ros-indigo-jackal-simulator ros-indigo-jackal-desktop
```

Environment setup

Edit the local bash environment to add a few useful aliases:

```
gedit ~/.bashrc
```

Add the following lines to the end of the file each provides a few shortcuts:

```
source /opt/ros/indigo/setup.bash
alias sws='source ./devel/setup.bash'
alias ss=’cd ~/workshop_ws/src’
```

Close your terminal and open a new one to make the changes effective

Create a work space in your home directory

```
mkdir -p ~/workshop_ws/src
cd ~/workshop_ws/src
catkin_init_workspace
cd ..
catkin_make
cd ~/workshop_ws/src
git clone https://github.com/aswinsarang/jackal.git
git clone https://github.com/aswinsarang/ROS-Workshop-Documents.git
cd ..
catkin_make
```