

Installing from source

Install from source requires that you download and compile the source code on your own. ROS Kinetic supports Ubuntu Wily and Xenial. Other platforms are possible to use but are not expected to work out of the box. Target platforms are defined in [REP 3](http://www.ros.org/reps/rep-0003.html) (<http://www.ros.org/reps/rep-0003.html>)

Contents

1. Installing from source
 1. Prerequisites
 1. Installing bootstrap dependencies
 2. Initializing rosdep
 2. Installation
 1. Building the catkin Packages
 3. Maintaining a Source Checkout
 1. Update the workspace
 2. Rebuild your workspace

1. Prerequisites

1.1 Installing bootstrap dependencies

Install bootstrap dependencies (Ubuntu):

These tools are used to facilitate the download and management of ROS packages and their dependencies, among other things.

Ubuntu or Debian:

```
$ sudo apt-get install python-rosdep python-rosinstall-generator python-wstool python-rosinstall build-essential
```

If you have trouble installing the packages in the command above, make sure you have added the [packages.ros.org](http://packages.ros.org/debian) debian repository to your apt source lists as described starting here: [kinetic/Installation/Ubuntu#kinetic.2BAC8-Installation.2BAC8-Sources.Setup_your_sources.list](http://wiki.ros.org/kinetic/Installation/Ubuntu#kinetic.2BAC8-Installation.2BAC8-Sources.Setup_your_sources.list) ([/kinetic/Installation/Ubuntu#kinetic.2BAC8-Installation.2BAC8-Sources.Setup_your_sources.list](http://wiki.ros.org/kinetic/Installation/Ubuntu#kinetic.2BAC8-Installation.2BAC8-Sources.Setup_your_sources.list))

If apt complains that the python-support package is not installable, make sure that you have the Ubuntu universe repositories enabled in your `/etc/apt/sources.list`

Fedora:

```
$ sudo yum install python-rosdep python-rosinstall_generator python-ws  
tool python-rosinstall @buildsys-build
```

Some packages used in ROS are not currently available in the Fedora RPM repositories. Most of the other packages are available in RPM Fusion. See [RPM Fusion Command Line Setup \(http://rpmfusion.org/Configuration#Command_Line_Setup_using_rpm\)](http://rpmfusion.org/Configuration#Command_Line_Setup_using_rpm). Any packages not available in RPM Fusion are staged in the SDSMT ROS RPM staging repository, which is available from [csc.mcs.sdsmt.edu \(http://csc.mcs.sdsmt.edu/\)](http://csc.mcs.sdsmt.edu).

Generic (pip):

If you are using a non-Debian system you need to make sure that you have all build tools (compiler, CMake, etc.) installed. You can install all ROS Python tools via PIP:

```
$ sudo pip install -U rosdep rosinstall_generator wstool rosinstall
```

If there are errors with this or the rosdep step below, your system's version of pip may be out-of-date. Use your system's package management to update it, or use it to update itself:

```
$ sudo pip install --upgrade setuptools
```

Note that on many platforms such as OSX you may want to install the above packages and use ROS inside a virtualenv so as to make sure not to collide with system dependencies.

1.2 Initializing rosdep

```
$ sudo rosdep init  
$ rosdep update
```

2. Installation

Start by building the core ROS packages.

2.1 Building the catkin Packages

ROS is in the process of converting to the catkin (/catkin) build system, but not all of the packages have been converted and the two build systems cannot be used simultaneously. Therefore it is necessary to build the core ROS packages first (catkin packages) and then the rest.

2.1.1 Create a catkin Workspace

In order to build the core packages, you will need a catkin workspace. Create one now:

```
$ mkdir ~/ros_catkin_ws
$ cd ~/ros_catkin_ws
```

Next we will want to fetch the core packages so we can build them. We will use `wstool` for this. Select the `wstool` command for the particular variant you want to install:

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

```
$ rosinstall_generator desktop_full --rosdistro kinetic --deps --wet-only --tar > kinetic-desktop-full-wet.rosinstall
$ wstool init -j8 src kinetic-desktop-full-wet.rosinstall
```

Desktop Install (recommended): ROS, `rqt (/rqt)`, `rviz (/rviz)`, and robot-generic libraries

```
$ rosinstall_generator desktop --rosdistro kinetic --deps --wet-only --tar > kinetic-desktop-wet.rosinstall
$ wstool init -j8 src kinetic-desktop-wet.rosinstall
```

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

```
$ rosinstall_generator ros_comm --rosdistro kinetic --deps --wet-only --tar > kinetic-ros_comm-wet.rosinstall
$ wstool init -j8 src kinetic-ros_comm-wet.rosinstall
```

This will add all of the catkin or wet packages in the given variant and then fetch the sources into the `~/ros_catkin_ws/src` directory. The command will take a few minutes to download all of the core ROS packages into the `src` folder. The `-j8` option downloads 8 packages in parallel.

In addition to the 3 variants above, more are defined in [REP 131](http://wiki.ros.org/REP-131) (<http://ros.org/reps/rep-0131.html#variants>) such as `robot`, `perception`, etc. Just change the package path to the one you want, e.g., for `robot` do:

```
$ rosinstall_generator robot --rosdistro kinetic --deps --wet-only --tar > kinetic-robot-wet.rosinstall
$ wstool init -j8 src kinetic-robot-wet.rosinstall
```

If `wstool init` fails or is interrupted, you can resume the download by running:

```
wstool update -j 4 -t src
```

2.1.2 Resolving Dependencies

Before you can build your catkin workspace you need to make sure that you have all the required dependencies. We use the `rosdep` tool for this:

```
$ rosdep install --from-paths src --ignore-src --rosdistro kinetic -y
```

This will look at all of the packages in the `src` directory and find all of the dependencies they have. Then it will recursively install the dependencies.

The `--from-paths` option indicates we want to install the dependencies for an entire directory of packages, in this case `src`. The `--ignore-src` option indicates to `rosdep` that it shouldn't try to install any ROS packages in the `src` folder from the package manager, we don't need it to since we are building them ourselves. The `--rosdistro` option is required because we don't have a ROS environment setup yet, so we have to indicate to `rosdep` what version of ROS we are building for. Finally, the `-y` option indicates to `rosdep` that we don't want to be bothered by too many prompts from the package manager.

After a while (and maybe some prompts for your password) `rosdep` will finish installing system dependencies and you can continue.

2.1.3 Building the catkin Workspace

Once it has completed downloading the packages and resolving the dependencies you are ready to build the catkin packages. We will use the `catkin_make_isolated` command because there are both catkin and plain `cmake` packages in the base install, when developing on your catkin only workspaces you should use `catkin/commands/catkin_make` (`/catkin/commands/catkin_make`).

Invoke `catkin_make_isolated`:

```
$ ./src/catkin/bin/catkin_make_isolated --install -DCMAKE_BUILD_TYPE=Release
```

Note: You might want to select a different CMake build type (e.g. `RelWithDebInfo` or `Debug`, see http://cmake.org/cmake/help/v2.8.12/cmake.html#variable:CMAKE_BUILD_TYPE (http://cmake.org/cmake/help/v2.8.12/cmake.html#variable:CMAKE_BUILD_TYPE)).

Note: The default catkin installation location would be `~/ros_catkin_ws/install_isolated`, if you would like to install some where else then you can do this by adding the `--install-space /opt/ros/kinetic` argument to your `catkin_make_isolated` call.

For usage on a robot without Ubuntu, it is recommended to install compiled code into `/opt/ros/kinetic` just as the Ubuntu packages would do. Don't do this in Ubuntu, as the packages would collide with `apt-get` packages. It is also possible to install elsewhere (e.g. `/usr`), but it is not recommended unless you really know what you are doing.

Please see [REP 122: Filesystem Hierarchy Layout](http://ros.org/reps/rep-0122.html) (<http://ros.org/reps/rep-0122.html>) for more detailed documentation on how the installed files are placed.

Note: In the above command we are running the `catkin_make_isolated` command from the `catkin` source folder because it has not been installed yet, once installed it can be called directly.

Now the packages should have been installed to `~/ros_catkin_ws/install_isolated` or to wherever you specified with the `--install-space` argument. If you look in that directory you will see that a `setup.bash` file have been generated. To utilize the things installed there simply source that file. Lets do that now before building the rest of ROS:

```
$ source ~/ros_catkin_ws/install_isolated/setup.bash
```

3. Maintaining a Source Checkout

If we want to keep our source checkout up to date, we will have to periodically update our `rosinstall` file, download the latest sources, and rebuild our workspace.

3.1 Update the workspace

To update your workspace, first move your existing `rosinstall` file so that it doesn't get overwritten, and generate an updated version. For simplicity, we will cover the `*desktop-full*` variant. For other variants, update the filenames and `rosinstall_generator` arguments appropriately.

```
$ mv -i kinetic-desktop-full-wet.rosinstall kinetic-desktop-full-wet.rosinstall.old
$ rosinstall_generator desktop_full --rosdistro kinetic --deps --wet-only --tar > kinetic-desktop-full-wet.rosinstall
```

Then, compare the new `rosinstall` file to the old version to see which packages will be updated:

```
$ diff -u kinetic-desktop-full-wet.rosinstall kinetic-desktop-full-wet.rosinstall.old
```

If you're satisfied with these changes, incorporate the new `rosinstall` file into the workspace and update your workspace:

```
$ wstool merge -t src kinetic-desktop-full-wet.rosinstall
$ wstool update -t src
```

3.2 Rebuild your workspace

Now that the workspace is up to date with the latest sources, rebuild it:

```
$ ./src/catkin/bin/catkin_make_isolated --install
```

If you specified the `--install-space` option when your workspace initially, you should specify it again when rebuilding your workspace

Once your workspace has been rebuilt, you should source the setup files again:

```
$ source ~/ros_catkin_ws/install_isolated/setup.bash
```

Except where otherwise

noted, the ROS wiki is

Wiki: [kinetic/Installation/Source](#) (last edited 2016-06-17 23:30:37 by TullyFoote (/TullyFoote))

licensed under the

Creative Commons Attribution 3.0 (<http://creativecommons.org/licenses/by/3.0/>) | Find us on

Google+ (<https://plus.google.com/113789706402978299308>)

Brought to you by:  Open Source Robotics Foundation

(<http://www.osrfoundation.org>)