# ROS Melodic 速查表

## ROS Kinetic Catkin Workspace

### Create a catkin workspace

Example:

```
    $ mkdir -p ~/catkin_ws/src
    $ cd ~/catkin_ws/src
    $ catkin_init_workspace
```

### Check out an existing ROS package

Get a local copy of an existing github ROS package, and compile it ONLY. Example:

```
    $ cd ~/catkin_ws/src
    $ git clone https://github.com/ros/ros_tutorials.git
    $ cd ~/catkin_ws && catkin_make --pkg ros_tutorials
```

### Create a new catkin ROS package

Create a new ROS package in an existing catkin workspace.

```
   $ catkin_create_pkg <pkg_name> [depend1] [depend2]
```

Example:

```
   $ roscd ~/catkin_ws/src

   $ catkin_create_pkg tutorial std_msgs roscpp rospy
```

### Build all packages in a workspace

After using catkin_make to build all packages, source the setup.bash to add the workspace to ROS_PACKAGE_PATH

Example:

```
  $ cd catkin_ws
  $ catkin_make
  $ source ~/catkin_ws/devel/setup.bash
```

## File System Management Tools

| | |
|---|---|
| rospack | A tool for inspecting packages. |
| roscd | Change directory to a package. |
| rosls | Lists package or stack information. |
| rosed | Open requested ROS  file in a text editor. |
| roscp | Copy a file from one place to another. |
| rosdep | Installs package system dependencies. |
| roswtf | Displays errors/warnings for a ROS system |

Usage:

```
  $ rospack find [package]
  $ roscd [package[/subdir]]
  $ rosls [package[/subdir]]
  $ rosed [package] [file]
  $ roscp [package] [file] [destination]
  $ rosdep install [package]
  $ roswtf or roswtf [file]
```

## Logging Tools

### rosbag

A set of tools for recording and playing back of ROS topics.

Commands:

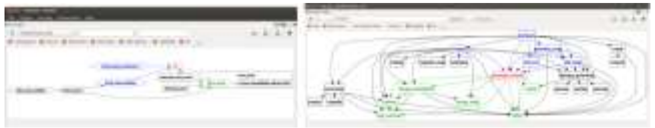
| | |
|---|---|
| rosbag record | Record a bag file with specified topics |
| rosbag play | Play content of one or more bag files |
| rosbag compress | Compress one or more bag files |
| rosbag decompress | Decompress one or more bag files |
| rosbag filter | Filter the contents of the bag |

Examples:

```
   Record selected topics:
     $ rosbag record topic1 topic2
   Replay all messages without waiting:
     $ rosbag play -a demo log.bag
   Replay several bag files at once:
     $ rosbag play demo1.bag demo2.bag
```

### rqt_graph

Tools for displaying graphs of running ROS nodes with connecting topics and package dependancies.



Usage:

```
    $ rqt_graph
```

## Data Visualization Tools

### View frames

A tool for visualizing the tree of coordinate transforms.

Usage:

```
   $ rosrun tf view_frames
```

### rqt_plot

A tool for plotting data from ROS topic fields.

Example:

```
   To graph all data on a same plot
   $ rqt_plot /topic1/field1, /topic2/field2
```

### rqt_image_view

A tool to display image topics.

Usage:  `$ rqt_image_view`

## Running System

Run ROS using plain:

```
roscore
```

Alternatively roslaunch will run roscore automatically if it can’t find one:

```
roslaunch my_package package_launchfile.launch
```

### Nodes, Topics, Messages

```
rosnode list
rostopic list
rostopic echo cmd_vel
rostopic hz cmd_vel
rostopic info cmd_vel
rosmsg show geometry_msgs/Twist
```

### Remote Connection

Master’s ROS environment:

- ROS_IP or ROS_HOSTNAME set to this machine’s network address.
- ROS_MASTER_URI set to URI containing that IP or hostname.

Your environment:

- ROS_IP or ROS_HOSTNAME set to your machine’s network address.
- ROS_MASTER_URI set to the URI from the master.

To debug, check ping from each side to the other, run roswtf on each side.

### ROS Console

Adjust using rqt_logger_level and monitor via rqt_console. To enable debug output across sessions, edit the `$HOME/.ros/config/rosconsole.config` and add a line for your package:

```
log4j.logger.ros.package_name=DEBUG
```

And then add the following to your session:

```
export ROSCONSOLE_CONFIG_FILE=$HOME/.ros/config/rosconsole.config
```

Use the roslaunch --screen flag to force all node output to the screen, as if each declared `<node>` had the `output="screen"` attribute.

### Notes

Source: The ROS system (usually /opt/ros/kinetic/share) or your catkin workspace (e.g., /home/user/catkin_ws/devel)

Include all ROS packages in environment variable, i.e., ROS_PACKAGE_PATH