ROS Indigo Cheatsheet

Filesystem Management Tools

rospack   A tool for inspecting packages.
rospack profile   Prints path and packageable problems.
roscd   Change directory to a package.
rosdep   Enable support for ROS.
rosdep/dependencies   List package or stack information.
rosdep/manifest   Open requested ROS file in a text editor.
rosdep/manifests   Copy a file from one place to another.
rosdep/package   Install package system dependencies.
rosdep/stack   Displays a error and warnings about a running ROS system or launch file.
rosdep/stacks   Creates a new ROS stack.

caitten createAction   Manage many repos in workspace.
wetool   Builds a ROS caitten workspace.
rept   Displays package structure and dependencies.

Usage:
```bash
$ rospack find [package]
$ rospack [package/package]
$ rospack [package/package[package]]
$ rospack [package/package/...]
$ rospack [package/...]
$ rospack [package/...]
$ rospack [package/...]
```

Start-up and Process Launch Tools

rosmake   Runs a ROS package's executable with minimal typing.
Usage:
```bash
$ rosmake [package/package]
```

Example:
```bash
rosrun turtlesim turtlesim_node
```

roslaunch   Starts a roscore if needed), local nodes, remote nodes via SSH, and sets parameter server arguments.
Usage:
```bash
$ roslaunch [package/package] [filename.launch]
```
ROS Indigo Cheatsheet

Logging Tools

**rqt_console**
A tool for displaying and filtering messages published on topics.

Usage:
```
rqt console
```

**rqt_bag**
A tool for visualizing, inspecting, and replaying log files.

Usage, viewing:
```
rqt bag <bag_file>
rqt bag [play the big red record button]
```

**rqt_logger_level**
Change the logger level of ROS nodes. This will increase or decrease the information they log to the screen and replot.

Usage, viewing:
```
rqt_logger_level
```

Introspection & Command Tools

**rqt_topic**
A tool for viewing published topics in real-time.

Usage:
```
rqt
```

**rqt_msg, rqt_srv, and rqt_action**
A tool for viewing available msgs, srvs, and actions.

Usage:
```
rqt
```

**rqt_publisher, and rqt_service_caller**
Tools for publishing messages and calling services.

Usage:
```
rqt
Plugin Menu—>Topic—>Message Publisher
Plugin Menu—>Service—>Service Caller
```

**rqt_graph, and rqt_dep**
Tools for displaying graphs of running ROS nodes with connecting topics and package dependencies respectively.

Usage:
```
rqt graph
rqt dep
```

**rqt_top**
A tool for ROS specific process monitoring.

Usage:
```
rqt
Plugin Menu—>Introspection—>Process Monitor
```

**rqt_reconfigure**
A tool for dynamically reconfiguring ROS parameters.

Usage:
```
rqt
Plugin Menu—>Configuration—>Dynamic Reconfigure
```

Development Environments

**rqt.shell, and rqt.py.console**
Two tools for accessing an stern shell and python console respectively.

Usage:
```
rqt
Plugin Menu—>Miscellaneous Tools—>Shell
Plugin Menu—>Miscellaneous Tools—>Python Console
```

Data Visualization Tools

**tf_echo**
A tool that prints the information about a particular transformation between a source frame and a target frame.

Usage:
```
$ rosrun tf tf_echo <source_frame> <target_frame>
```

Examples:
To echo the transform between /map and /odom:
```
$ rosrun tf tf_echo /map /odom
```

**view_frames**
A tool for visualizing the full tree of coordinate transforms.

Usage:
```
$ rosrun tf2tools view_frames.py
$ write_frames.pdf
```

**rqt_plot**
A tool for plotting data from ROS topic fields.

Examples:
To graph the data in different plots:
```
$ rqt_plot /topic1/f1 /topic2/f2
To graph the data all on the same plot:
$ rqt_plot /topic1/f1 /topic2/f2
To graph multiple fields of a message:
$ rqt_plot /topic1/f1 /field1 /field2
```

**rqt_image_view**
A tool for displaying image topics.

Usage:
```
rqt_image_view
```

http://robosub.eecs.wsu.edu/wiki/ROS%20Indigo%20Cheatsheet
Printed on 2019/05/10 18:54
ROS Indigo Catkin Workspaces

Create a catkin workspace
Setup and use a new catkin workspace from scratch.

Example:
```
$ source /opt/ros/melodic/setup.bash
$ mkdir -p ~/catkin_ws/src
$ cd ~/catkin_ws/src
$ catkin init workspace
```

Checkout an existing ROS package
Get a local copy of the code for an existing package and keep it up to date using wstool.

Example:
```
$ cd ~/catkin_ws/src
$ wstool init
$ wstool met tutorials --git git://github.com/ros/ros_tutorials.git
$ wstool update
```

Create a new catkin ROS package
Create a new ROS catkin package in an existing workspace with catkin create package. After using this you will need to edit the CMakeLists.txt to detail how you want your package built and add information to your package.xml.

Usage:
```
catkin create.pkg <package.name> [depend1] [depend2]
```

Example:
```
$ cd ~/catkin_ws/src
$ catkin create.pkg tutorials std_msgs rospy rospypp
```

Build all packages in a workspace
Use catkin_make to build all the packages in the workspace and then source the setup.bash to add the workspace to the ROSSPACE_PATH.

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

Copyright © 2015 Open Source Robotics Foundation
Copyright © 2015 Willow Garage

From:
http://robosub.eecs.wsu.edu/wiki/ - Palouse RoboSub Technical Documentation

Permanent link:
http://robosub.eecs.wsu.edu/wiki/cs/ROS/cheatsheet/start

Last update: 2016/11/16 12:34