ROS Indigo Cheatsheet

Filesystem Management Tools

rospack
rospack profile
roscd
roscfg
rosnode
rosrun
roscore
roslaunch
rostopic
rostopic create
rostopic killers
rostopic info
rostopic list
rostopic pub
rostopic type

Usage:
$ rospack find [package]
$ rospack [package/subdir] [-t | -l | -d]
$ rostopic list
$ rostopic info [topic]
$ rostopic list [-p]

Logging Tools

rosbag

A tool for recording and playing back ROS topics.

Commands:
rosbag record
rosbag replay
rosbag compress
rosbag decompress
rosbag filter

Examples:
Record record topics:
$ rosbag record topic1 topic2
Replay all messages without waiting:
$ rosbag play -a demo_bag
Filter saved bag files at once:
$ rosbag play demo1_bag demo2_bag

Introspection and Command Tools

rosmsg/rosesrv

Displays Message/Service (msg/srv) data structures definitions.

Commands:
rosmsg show
rosmsg list
rosmsg diff
rosmsg package
rosmsg packages

Examples:
Display the Pose msg:
$ rosmsg show Pose
List the messages in the msgpackage:
$ rosmsg list mymsgpackage
Display the msg/service differences:
$ rosmsg diff mymsgpackage mymsgpackage_2
List all the messages in a package:
$ rosmsg package mymsgpackage

rosnode

Displays debugging information about ROS nodes, including publications, subscriptions and connections.

Commands:
rosnode ping
rosnode list
rosnode info
rosnode machine
rosnode kill

Examples:
Kill all nodes:
$ rosnode kill
List nodes on a machine:
$ rosnode machine agp,local
Ping all nodes:
$ rosnode ping

rosaction

A tool for displaying information about ROS topics, including publishers, subscribers, publishing rate, and messages.

Commands:
rosaction
rosaction info
rosaction list
rosaction pub
rosaction type

Examples:
Display the bandwidth used by topic:
$ rosaction info /my_topic
Print messages to screen:
$ rosaction pub /my_topic
Print publishing rate of topic:
$ rosaction info /my_topic
Print information about an active topic:
$ rosaction info /my_topic
List all published topics:
$ rosaction list
Publish data to topic:
$ rosaction pub topic_name data
Print topic type:
$ rosaction type topic_name

rospy

A tool for getting and setting ROS parameters on the parameter server using YAML-encoded files.

Commands:
rospy get
rospy load
rospy dump
rospy delete
rospy list

Examples:
List all the parameters in a namespace:
$ rospy list /my_namespace
Get a parameter:
$ rospy get /my_namespace/param_name
Load parameters from a file:
$ rospy load /path/to/param_file
Dump parameters to a file:
$ rospy dump /path/to/param_file
Delete a parameter:
$ rospy delete /my_namespace/param_name

rosrun

Starts a rosnode (if needed), local nodes, remote nodes via SSH, and sets parameter server arguments.

Usage:
$ rosrun
$ rosrun [package] executable_name

Examples:
Launch a file in a package:
$ rosrun package_name file_name.launch
Launch on a different port:
$ rosrun -p 1234 package_name file_name.launch
Launch on the local node:
$ rosrun --local package_name file_name.launch

Palign:
$ rosrun package_name file_name.launch

roscore

Runs a ROS package's executable with minimal typing.

Usage:
$ roscore
$ rosrun package_name executable_name

Example (run turtlebot):
$ rosrun turtlebot turtlesim_node

roslaunch

Starts a rosnode (if needed), local nodes, remote nodes via SSH, and sets parameter server arguments.

Usage:
$ roslaunch

Examples:
Launch a file in a package:
$ roslaunch package_name file_name.launch
Launch on a different port:
$ roslaunch -p 1234 package_name file_name.launch
Launch on the local node:
$ roslaunch --local package_name file_name.launch
ROS Indigo Cheatsheet

Logging Tools

rqt_console
A tool to display and filtering messages published on a topic.

Usage:
$ rqt.console

rqt_bag
A tool for stimulating, inspecting, and replaying log files.

Usage, viewing:
$ rqt_bag bag_file_bag
Usage, replaying:
$ rqt_bag [ress 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 rtpmsgs.

Usage:
viewing $ rqt_logger_level

Introduction & Command Tools

rqt_topic
A tool for viewing published topics in real time.

Usage:
$ rqt

rqtmsg, 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_depth
Tools for displaying graphs of running ROS nodes with connecting topics and package dependencies respectively.

Usage:
$ rqt_graph
$ rqt_depth

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 xterm 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 visualising the full tree of coordinate transforms.

Usage:
$ rosrun tf2 tools view_frames.py
$ source frames.pdf

rqt_plot
A tool for plotting data from ROS topic fields.

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

rqt_image_view
A tool for displaying image topics.

Usage:
$ rqt_image_view
ROS Indigo Catkin Workspaces

Create a catkin workspace

Setup and use a new catkin workspace from scratch.

Examples:
```
$ source /opt/ros/hydro/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.

Examples:
```
$ cd ~/catkin_ws/src
$ wstool init
$ wstool set 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.

Example:
```
$ cd ~/catkin_ws/src
$ catkin_create_pkg <package_name> [depend1] [depend2]
```

Build all packages in a workspace

Use catkin_make to build all the packages in the workspace and then source the setup script to add the workspace to the ROS_PACKAGE_PATH.

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

Copyright © 2015 Open Source Robotics Foundation
Copyright © 2010 Willow Garage