

ROS Indigo Cheatsheet

Filesystem Management Tools

rospack/rostack	A tool inspecting packages .
rospack profile	Fixes path and pluginlib problems.
roscd	Change directory to a package or stack.
rospd/rosd	Pushd equivalent for ROS .
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 a errors and warnings about a running ROS system or launch file.
roscrcat	Creates a new ROS package.
roscrcat stack	Creates a new ROS stack.
rosmake	Builds a ROS package.
rqt_dep	Displays package structure and dependencies.

Usage:

```
$ rospack find [package]
$ roscd [package[/subdir]]
$ rospd [package[/subdir] | +N | -N]
$ rosd
$ rosls [package[/subdir]]
$ rosed [package] [file]
$ roscp [package] [file] [destination]
$ rosdep install [package]
$ roswtf or roswtf [file]
$ roscrcat-pkg [package_name]
$ rosmake [package]
$ rqt_dep [options]
```

Start-up and Process Launch Tools

roscore

The basis [nodes](#) and programs for ROS-based systems. A roscore must be running for ROS nodes to communicate.

Usage:

```
$ roscore
```

roslaunch

Runs a ROS package's executable with minimal typing.

Usage:

```
$ roslaunch package_name executable_name
```

Example (runs [turtlesim](#)):

```
$ roslaunch turtlesim turtlesim_node
```

roslaunch

Starts a roscore (if needed), [local nodes](#), [remote nodes](#) via SSH, and sets parameter server [parameters](#).

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 nodes:
$ roslaunch --local package_name file_name.launch
```

Introspection and Command Tools

roscrcat

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

Commands:

roscrcat ping	Test connectivity to node.
roscrcat list	List active nodes.
roscrcat info	Print information about a node.
roscrcat machine	List nodes running on a machine.
roscrcat kill	Kill a running node.

Examples:

```
Kill all nodes:
$ roscrcat kill -a
List nodes on a machine:
$ roscrcat machine aqy.local
Ping all nodes:
$ roscrcat ping --all
```

rostopic

A tool for displaying information about ROS [topics](#), including publishers, subscribers, publishing rate, and messages.

Commands:

rostopic bw	Display bandwidth used by topic.
rostopic echo	Print messages to screen.
rostopic find	Find topics by type.
rostopic hz	Display publishing rate of topic.
rostopic info	Print information about an active topic.
rostopic list	List all published topics.
rostopic pub	Publish data to topic.
rostopic type	Print topic type.

Examples:

```
Publish hello at 10 Hz:
$ rostopic pub -r 10 /topic_name std_msgs/String hello
Clear the screen after each message is published:
$ rostopic echo -c /topic_name
Display messages that match a given Python expression:
$ rostopic echo --filter "m.data=='foo'" /topic_name
Pipe the output of rostopic to rosmmsg to view the msg type:
$ rostopic type /topic_name | rosmmsg show
```

rosservice

A tool for listing and querying ROS services.

Commands:

rosservice list	Print information about active services.
rosservice node	Print name of node providing a service.
rosservice call	Call the service with the given args.
rosservice args	List the arguments of a service.
rosservice type	Print the service type.
rosservice uri	Print the service ROSRPC uri.
rosservice find	Find services by service type.

Examples:

```
Call a service from the command-line:
$ rosservice call /add_two_ints 1 2
Pipe the output of rosservice to rossrv to view the srv type:
$ rosservice type add_two_ints | rossrv show
Display all services of a particular type:
$ rosservice find rospy_tutorials/AddTwoInts
```

roscrcat

A tool for getting and setting ROS [parameters](#) on the parameter server using YAML-encoded files.

Commands:

roscrcat set	Set a parameter.
roscrcat get	Get a parameter.
roscrcat load	Load parameters from a file.
roscrcat dump	Dump parameters to a file.
roscrcat delete	Delete a parameter.
roscrcat list	List parameter names.

Examples:

```
List all the parameters in a namespace:
$ roscrcat list /namespace
Setting a list with one as a string, integer, and float:
$ roscrcat set /foo "[1', 1, 1.0]"
Dump only the parameters in a specific namespace to file:
$ roscrcat dump dump.yaml /namespace
```

rosmmsg/rossrv

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

Commands:

rosmmsg show	Display the fields in the msg/srv.
rosmmsg list	Display names of all msg/srv.
rosmmsg md5	Display the msg/srv md5 sum.
rosmmsg package	List all the msg/srv in a package.
rosmmsg packages	List all packages containing the msg/srv.

Examples:

```
Display the Pose msg:
$ rosmmsg show Pose
List the messages in the nav_msgs package:
$ rosmmsg package nav_msgs
List the packages using sensor_msgs/CameraInfo:
$ rosmmsg packages sensor_msgs/CameraInfo
```

Logging Tools

roscrcat

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

Commands:

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

Examples:

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

tf_echo

A tool that prints the information about a particular transformation between a source_frame and a target_frame.

Usage:

```
$ roslaunch tf tf_echo <source_frame> <target_frame>
```

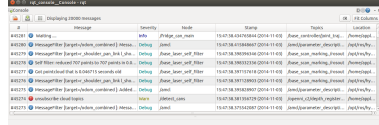
Examples:

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

Logging Tools

rqt_console

A tool to display and filtering messages published on rosout.

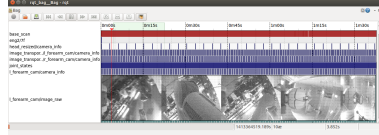


Usage:

```
$ rqt_console
```

rqt_bag

A tool for visualizing, inspecting, and replaying bag files.



Usage, viewing:

```
$ rqt_bag bag.file.bag
```

Usage, bagging:

```
$ rqt_bag *press 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 rqt_console.

Usage:

```
viewing $ rqt_logger_level
```

Introspection & Command Tools

rqt_topic

A tool for viewing published topics in real time.

Usage:

```
$ rqt
```

Plugin Menu->Topic->Topic Monitor

rqt_msg, rqt_srv, and rqt_action

A tool for viewing available msgs, srvs, and actions.

Usage:

```
$ rqt
```

Plugin Menu->Topic->Message Type Browser

Plugin Menu->Service->Service Type Browser

Plugin Menu->Action->Action Type Browser

rqt_top

A tool for ROS specific process monitoring.

Usage:

```
$ rqt
```

Plugin Menu->Introspection->Process Monitor

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_reconfigure

A tool for dynamically reconfiguring ROS parameters.

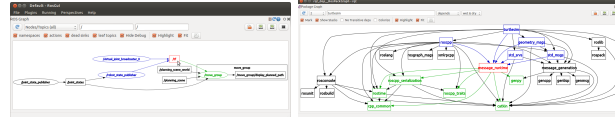
Usage:

```
$ rqt
```

Plugin Menu->Configuration->Dynamic Reconfigure

rqt_graph, and rqt_dep

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



Usage:

```
$ rqt_graph
```

```
$ rqt_dep
```

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

view_frames

A tool for visualizing the full tree of coordinate transforms.

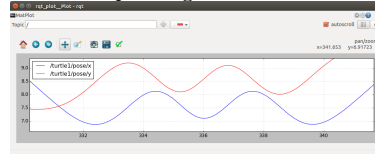
Usage:

```
$ rosrun tf2.tools view_frames.py
```

```
$ evince 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:

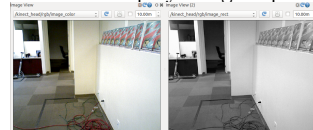
```
$ rqt_plot /topic1/field1,/topic2/field2
```

To graph multiple fields of a message:

```
$ rqt_plot /topic1/field1:field2:field3
```

rqt_image_view

A tool to display image topics.



Usage:

```
$ rqt_image_view
```