



## **ROS** IN 5 PAGES

## **Common** Command-line Tools



#### roscore

A collection of nodes and programs that are pre-requisites of a ROS-based system. You must have a roscore running in order for ROS nodes to communicate.

• Usage:

\$ roscore

#### roslaunch

Starts ROS nodes locally and remotely via SSH, as well as setting parameters on the parameter server.

- Examples:
- Execute launch file keyboard\_teleop.launch from package turtlebot\_teleop:

\$ roslaunch turtlebot\_teleop keyboard\_teleop.launch

#### rosrun

rosrun allows you to run an executable in an arbitrary package without having to cd (or roscd) there first.

• Usage:

\$ rosrun package executable

- Examples: Run Python file tel
  - Run Python file teleop\_twist\_keyboard.py from package teleop\_twist\_keyboard:
  - \$ rosrun teleop\_twist\_keyboard teleop\_twist\_keyboard.py

#### rosnode

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

• Commands: rosnode list List active nodes.

rosnode info Print information about a node.

rosnode kill Kills a running node.

• Examples: - Get list of running nodes: rosnode list

- Kill node /cmd\_el\_mux: rosnode kill /cmd\_vel\_mux- Get info from node /gazebo: rosnode info /gazebo

#### rosmsq/rossrv

rosmsg/rossrv displays Message/Service (msg/srv) data structure definitions.

• Commands: rosmsq show Display the fields in the msq.

rossrv show Display the fields in a service msg

- Examples: Show structure of message Twist: \$ rosmsg show geometry\_msgs/Twist
  - Show structure of service message Trigger: \$ rossrv show std\_srvs/Trigger



## **ROS** IN 5 PAGES

# Common Command-line Tools





## rostopic

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

• Commands: rostopic echo Print messages to screen.

rostopic info Print information about a topics.
rostopic list Print information about active topics.

rostopic pub Publish data to topic.

• Examples:

- Get list of running topics: rostopic list

- Get info from topic /cmd\_vel: rostopic info /cmd\_vel- Listen to topic /cmd\_vel: rostopic echo /cmd\_vel

- Publish a Twist message into topic /cmd\_vel: rostopic pub /cmd\_vel geometry\_msgs/ Twist "linear:

x: 0.0 y: 0.0 z: 0.0 angular: x: 0.0 y: 0.0 z: 0.0"

#### rosparam

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

• Commands: rosparam set Set a parameter.

rosparamgetGet a parameter.rosparamdeleteDelete a parameter.rosparamlistList parameter names.

• Examples: - List parameters loaded on the ROS Param Server: rosparam list

- Get value of parameter /camera/imager\_rate: rosparam get /camera/imager\_rate rosparam set /camera/imager\_rate 3.0

#### rosservice

A tool for listing and guerying ROS services.

• Commands: rosservice list Print information about active services.

rosservice type Print the service type.

rosservice info
rosservice call
Get information from a service
Call the service with the given args.

• Examples:

- Get list of running services: rosservice list

- Get service message used by the service /qazebo/delete model: rosservice type /qazebo/delete\_model

- Get information from the service /gazebo/delete model: rosservice info /gazebo/delete\_model

- Call service /gazebo/delete model and provide argument model name:

rosservice call /qazebo/delete\_model "model\_name:'bowl'"

# **ROS** IN 5 PAGES

# **Filesystem** Command-line Tools





## Filesystem Command-line Tools

• Commands: rospack A tool inspecting packages.

roscd Changes directories to a package or stack.

rosswtf Displays a errors and warnings about a running ROS system or launch file.

catkin\_create\_pkg
A tool for creating a new packages.

• Usage:

\$ rospack find [package]

\$ roscd [package [/subdir]]

\$ roswtf or roswtf [file]

\$ catkin\_create\_pkg [name\_of\_pkg][dependencies]

Examples:

- Find pack of package turtlebot\_gazebo: \$ rospack find turtlebot\_gazebo

- Go to package turtlebot\_gazebo: \$ roscd turtlebot\_gazebo

- Check for errors in keyboard\_teleop.launch file: \$ roswtf keyboard\_teleop.launch

# ROS IN 5 PAGES Logging Command-line Tools





### rosbag

This is a set of tools for recording from and playing back to ROS topics. It is intended to be high performance and avoids deserialization and reserialization of the messages.

**rosbag record** will generate a ".bag" file (so named for historical reasons) with the contents of all topics that you pass to it.

• Examples:

- Record all topics: \$ rosbag record -a

- Record select topics: \$ rosbag record topic1 topic2

rosbag play will take the contents of one or more bag file, and play them back in a time-synchronized fashion.

Examples:

Replay test.bag file: \$ rosbag play test.bagReplay several bag files at once: \$ rosbag info test.bag

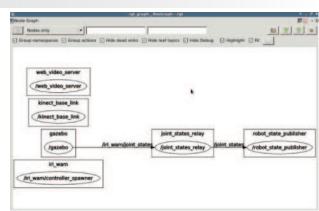
# **ROS** IN 5 PAGES **Graphical** Tools



# rx\_graph

Displays a graph of the ROS nodes that are currently running, as well as the ROS topics that connect them.

- Usage:
  - \$ rqt\_graph



## view\_frames

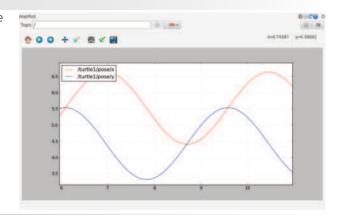
A tool for visualizing the full tree of coordinate transforms.

- Usage:
  - \$ rosrun tf view\_frames
  - \$ evince frames.pdf

# rx\_plot

A tool for plotting data from one or more ROS topic fields.

- Usage:
  - \$ rqt\_plot



**GRAPHICAL** TOOLS