

RVIZ: introduction

- 3D visualization tool for ROS
- Subscribes to topics and visualizes the message contents
- Different camera views (orthographic, topdown, etc.)
- Interactive tools to publish user information
- Save and load setup as RViz configuration
- Extensible with plugins
- Command line help


```
roslaunch rviz rviz -help
```
- Run RViz with

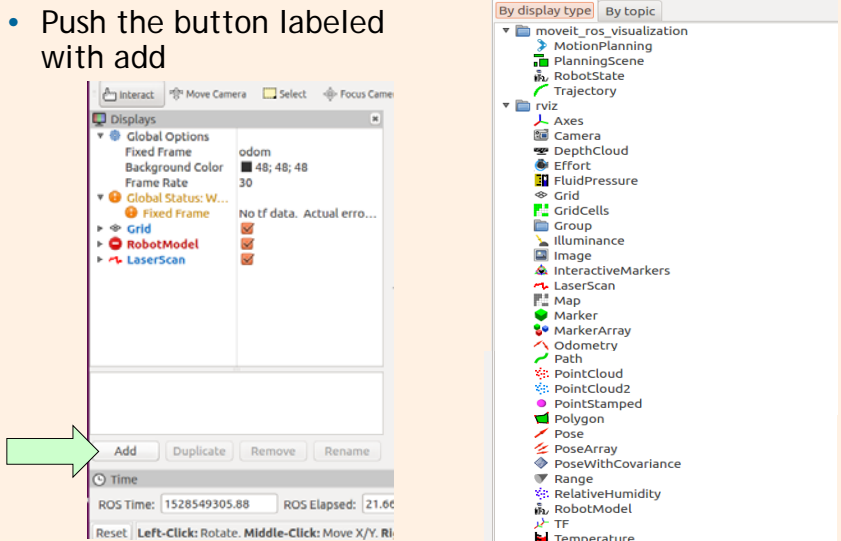

```
roslaunch rviz rviz -help
```
- More info → <http://wiki.ros.org/rviz>


 ROS

5

RVIZ: display plugins

- Push the button labeled with add



 ROS

6

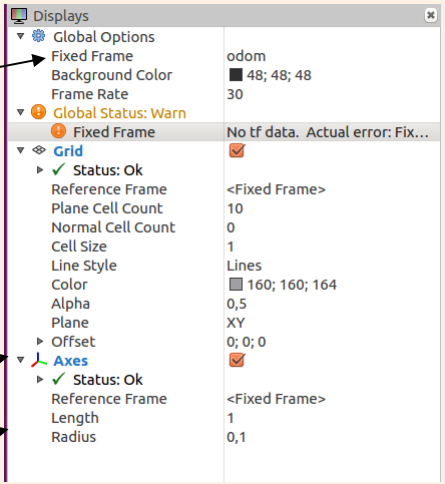
RVIZ: display plugins

- Once we insert a plugin, check its options

Reference frame

Added plugin

Plugin properties

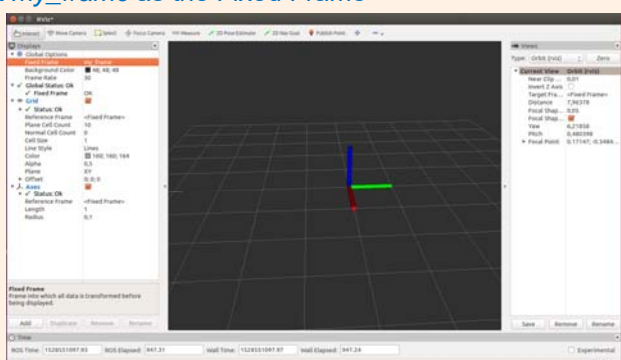


ROS

7

RVIZ: viewing fixed frame

- If you have this error
No tf data. Actual error: Fixed Frame [odom] does not exist
→ Execute in a terminal
`roslaunch static_transform_publisher 0 0 0 0 0 1 map my_frame 10`
And select *my_frame* as the Fixed Frame



ROS

8

RVIZ: the most important plugins

Name	Description	Messages Used
Axes	Displays a set of Axes	
Effort	Shows the effort being put into each revolute joint of a robot.	sensor_msgs/JointStates
Camera	Creates a new rendering window from the perspective of a camera, and overlays the image on top of it.	sensor_msgs/Image , sensor_msgs/CameraInfo
Grid	Displays a 2D or 3D grid along a plane	
Grid Cells	Draws cells from a grid, usually obstacles from a costmap from the navigation stack.	nav_msgs/GridCells
Image	Creates a new rendering window with an Image. Unlike the Camera display, this display does not use a CameraInfo. Version: Diamondback+	sensor_msgs/Image
InteractiveMarker	Displays 3D objects from one or multiple Interactive Marker servers and allows mouse interaction with them. Version: Electric+	visualization_msgs/InteractiveMarker
Laser Scan	Shows data from a laser scan, with different options for rendering modes, accumulation, etc.	sensor_msgs/LaserScan
Map	Displays a map on the ground plane.	nav_msgs/OccupancyGrid
Markers	Allows programmers to display arbitrary primitive shapes through a topic	visualization_msgs/Marker , visualization_msgs/MarkerArray
Path	Shows a path from the navigation stack.	nav_msgs/Path
Point	Draws a point as a small sphere.	geometry_msgs/PointStamped
Pose	Draws a pose as either an arrow or axes.	geometry_msgs/PoseStamped
Pose Array	Draws a "cloud" of arrows, one for each pose in a pose array	geometry_msgs/PoseArray
Point Cloud(2)	Shows data from a point cloud, with different options for rendering modes, accumulation, etc.	sensor_msgs/PointCloud , sensor_msgs/PointCloud2
Polygon	Draws the outline of a polygon as lines.	geometry_msgs/Polygon
Odometry	Accumulates odometry poses from over time.	nav_msgs/Odometry
Range	Displays cones representing range measurements from sonar or IR range sensors. Version: Electric+	sensor_msgs/Range
RobotModel	Shows a visual representation of a robot in the correct pose (as defined by the current TF transforms).	
TF	Displays the tf transform hierarchy.	
Wrench	Draws a wrench as arrow (force) and arrow + circle (torque)	geometry_msgs/WrenchStamped
Oculus	Renders the RViz scene to an Oculus headset	



RVIZ: turtle_tf

- Close everything and start from scratch
- Terminal 1


```
> roslaunch turtle_tf turtle_tf_demo.launch
```
- Try to move the turtles with the cursors
- Terminal 2


```
rosviz rviz -d 'rospack find turtle_tf'/rviz/turtle_rviz.rviz
```
- Change the options as it is showed in the following slide
an play with the turtles (push arrow keys when terminal 1 is active)

11

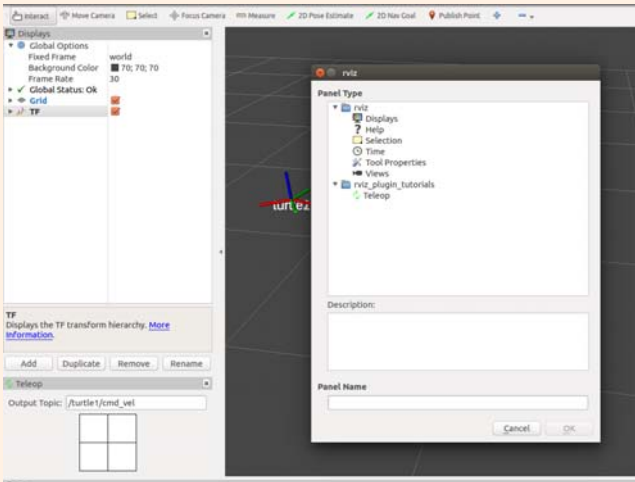
RVIZ: turtle_tf

The screenshot displays the RVIZ (Robot Visualization) interface. The main 3D view shows a 'world' frame with two turtles, 'turtle1' and 'turtle2', positioned on a grid. The left sidebar contains the 'Global Options' and 'TF' (Transform) tree. The 'TF' tree shows the 'world' frame as the parent of 'turtle1' and 'turtle2'. The right sidebar shows the 'Views' panel with 'Orbit (rviz)' selected. A terminal window at the bottom shows the command 'roslaunch turtle_tf_demo.launch' and the output of the launch process, including the start of the 'master' process and the 'turtle_tf_demo' launch file.

12

RVIZ: turtle_tf

- We can add a 'teleop' panel so we can control turtle1 directly from rviz



The screenshot shows the RVIZ interface. On the left, the 'Displays' panel is open, showing a tree view with 'Global Options', 'Fixed Frame' (set to 'world'), 'Background Color' (70, 70, 70), 'Frame Rate' (30), 'Global Status OK', 'Grid', and 'TF' (selected). Below the tree, there are buttons for 'Add', 'Duplicate', 'Remove', and 'Rename'. At the bottom of the 'Displays' panel, there is a 'Teleop' panel type selected, with an 'Output Topic' field set to '/turtle1/cmd_vel'. On the right, the 'Panel Type' dialog box is open, showing a tree view with 'rviz', 'Displays', 'Help', 'Selection', 'Time', 'Tool Properties', 'Views', 'rviz_plugins_tutorials', and 'Teleop' (selected). Below the tree, there is a 'Description' field and a 'Panel Name' field. The 'Cancel' and 'OK' buttons are at the bottom right of the dialog box.

ROS

13

ROS visualizer config files



tecrun

RVIZ: save/load config

- Rviz can save/load config
- *.rviz files in the rviz folder of the package




- To load a stored configuration
`roslaunch rviz rviz -d 'rospack find my_package'/rviz/config_file.rviz`

 ROS

15

ROS visualizer: viewing veggie





The slide features a dark red header with the word 'tecnun' in white lowercase letters on the left and the title 'RVIZ: running Veggie' in white uppercase letters on the right. The main content area has a light orange background and contains a bulleted list of steps, terminal commands, and the ROS logo. The steps include opening Rviz, configuring it with specific settings, and sending commands to the /cmd_vel topic. The terminal commands are shown in a monospaced font. The ROS logo is located at the bottom left of the content area, and the slide number '17' is at the bottom right.

RVIZ: running Veggie

- Process
 - Open Rviz

```
roslaunch rviz rviz
```
 - Configure Rviz
 - Put odom as the main frame
 - Add RobotModel
 - Add Laser topic (/scan)
 - Send commands to /cmd_vel

Move FWD

```
rostopic pub /cmd_vel geometry_msgs/Twist '{linear: {x: 1.0}}'
```

rosSPIN

```
rostopic pub /cmd_vel geometry_msgs/Twist '{angular: {z: 1.0}}'
```

 ROS

17