ROS Tips and Tricks

ROS + PR2 Training Workshop

Overview

1. ROS network assumptions
2. Common ROS environment variables
3. rosbash commands
4. Ros paths and overlays
5. topic_tools
6. rostest

ROS Network Assumptions

http://www.ros.org/wiki/ROS/NetworkSetup

1. There must be a master running
2. All machines in network must see master
   - `ROS_MASTER_URI=http://hostname:port`
   - 11311 is the default port. Use a different port if sharing a machine
3. Any subscriber can find a publisher by hostname and communicate on all ports
   - If no control of DNS, you can use your hosts file
   - Alternatively, use `ROS_HOSTNAME` or `ROS_IP`

Common Environment Variables

http://www.ros.org/wiki/ROS/EnvironmentVariables

1. `ROS_ROOT` – path to your ROS installation
2. `ROS_PACKAGE_PATH` – path to packages
3. `ROS_MASTER_URI` – location of master
4. `PYTHONPATH` – must include:
   - `$ROS_ROOT/core/roslib/src`

rosbash

http://www.ros.org/wiki/rosbash

1. Rosbash is a helper tool for bash
   - Defines useful aliases
   - Sets up tab-completion
2. `roscd <packagename>[/path]`
3. `rosed <packageame> <filename>`
4. NOTE: tab-completion may sometimes be slow due to network or disk access

Rospack

http://www.ros.org/wiki/rospack

1. rospack is the tool that finds packages according to your package path
   - Used internally by rosrun, roslaunch, rospy import, etc.
   - If something behaving strangely, verify: `rospack find <packagename>` behaves as expected
2. packages at front of path take priority

Overlays

1. Because packages at front of path take priority, you can create an “overlay”
   - Start by sourcing existing setup.(ba)sh
   - export ROS_PACKAGE_PATH=/path/to/overlay:${ROS_PACKAGE_PATH}
   - Your overlay should include all packages that depend on your package (rospack deps)

Overlays

1. Overlays can be created with rosinstall: http://www.ros.org/wiki/rosinstall

```
wget --no-check-certificate http://ros.org/rosinstall -O ~/rosinstall

chmod 755 ~/rosinstall
```

Topic tools

http://www.ros.org/wiki/topic_tools

1. Useful utilities for manipulating topics without knowing types
2. Can be very useful for tying systems together

rostest

http://www.ros.org/wiki/rostest

1. Similar to launch file, but includes a <test> tag.
2. Brings up all nodes, and then brings up each test node.
3. Roslaunch ignores test tags, so can be used to add tests to an arbitrary launch file – good practice.