# ROS system commands

|  |  |  |
| --- | --- | --- |
| **Stacks: (Ansammlung von Packages)** |  | **Listing running nodes:** |
| rosstack |  | Rosnode list |
| **Packages:** |  | **Node über Package name starten:** |
| Rospack  Rospack find ar\_track\_alvar |  | rosrun [package\_name] [node\_name]  **Bsp:**  package: my\_pcl\_tutorial  Node script: **src/example.cpp**  rosrun my\_pcl\_tutorial example |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# ROS package creation (filestructure) [rosbuild]

|  |
| --- |
| • Erstellen eines Packages in /home dir |
| roscreate-pkg [package\_name] [depend1] [depend2] [depend3] |
| **Bsp:** roscreate-pkg beginner\_tutorials std\_msgs rospy roscpp |
| • **manifest.xml** |
| Enthält alle Infos zum package (Abhängigkeiten etc). Für das Programmieren in C++ z.B. roscpp oder in Python rospy.Im manifest.xml mit <depend package="roscpp"/> festgehalten |

**Creating package in different place:**

* printenv ROS\_PACKAGE\_PATH (normally: /opt/ros/hydro/share:/opt/ros/hydro/stacks)

“Add this line to your ~/.bashrc directly under the call to /opt/ros/distro/setup.bash:

export ROS\_PACKAGE\_PATH=~/ros\_workspace:${ROS\_PACKAGE\_PATH}

“

# Building the package

|  |
| --- |
| • Erstellen eines Packages in /home dir |
| rosmake [package] |
| **Bsp:** rosmake beginner\_tutorials |
| • **manifest.xml** |
| Enthält alle Infos zum package (Abhängigkeiten etc). Für das Programmieren in C++ z.B. roscpp oder in Python rospy.Im manifest.xml mit <depend package="roscpp"/> festgehalten |
|  |

# ROS data concepts:

 [Nodes](http://wiki.ros.org/Nodes): A node is **an executable** that uses ROS to communicate with other nodes.

 [Messages](http://wiki.ros.org/Messages): ROS **data type** used when subscribing or publishing to a topic.

 [Topics](http://wiki.ros.org/Topics): Nodes can publish messages to a topic as well as subscribe to a topic to receive messages.

 [Master](http://wiki.ros.org/Master): Name service for ROS (i.e. helps nodes find each other)

 [rosout](http://wiki.ros.org/rosout): ROS equivalent of stdout/stderr

|  |
| --- |
| • Grafische Darstellung der Message zu einem Topic und damit die Kommunikation zw. Nodes |
| rosrun rqt\_graph rqt\_graph |
|  |
| • ROS topics untersuchen |
| • Topic data ausgeben: (hier topic /turtle1/cmd\_vel) |
| **rostopic echo** /turtle1/cmd\_vel |

# ROS service

[srv](http://wiki.ros.org/srv): an srv file describes a service. It is composed of two parts: a request and a response.

# Building a node

|  |
| --- |
| Place script file: Soundify/scripts/talker.py |
| **Set to executable: chmod +x scripts/talker.py** |
| In /soundify: $ make |
| Rosrund soundify talker.py |

# Running a node

Rosrun soundify talker.py

Rosrun soundify listener.py

|  |
| --- |
| • Unable to register with master node |
| $ roscore |
| • Python file not listed |
| chmod +x scripts/talker.py |
| #!/usr/bin/env python (at the top) |