

Documentation - ROS Wiki x +

wiki.ros.org

ROS.org

About | Support | Discussion Forum | Service Status | Q&A answers.ros.org

Search:

Documentation Browse Software News Download

### Documentation

ROS (Robot Operating System) provides libraries and tools to help software developers create robot applications. It provides hardware abstraction, device drivers, libraries, visualizers, message-passing, package management, and more. ROS is licensed under an open source, BSD license.

Available Translations: [German](#) | [Spanish](#) | [French](#) | [Italian](#) | [Japanese](#) | [Korean](#) | [Brazilian Portuguese](#) | [Portuguese](#) | [Русский \(Russian\)](#) | [Thai](#) | [Turkish](#) | [简体中文](#) | [Ukrainian](#) | [Vietnamese](#)

ROS:

- [Install](#)  
Install ROS on your machine.
- [Getting Started](#)  
Learn about various concepts, client libraries, and technical overview of ROS.
- [Tutorials](#)  
Step-by-step instructions for learning ROS hands-on
- [Contribute](#)  
How to get involved with the ROS community, such as submitting your own repository.
- [Support](#)  
What to do if something doesn't work as expected.
- [Quality Assurance](#)  
How to ensure that your ROS-based systems and your contributions to ROS are of high quality

Wiki

- [Distributions](#)
- [ROS/Installation](#)
- [ROS/Tutorials](#)
- [RecentChanges](#)
- [Documentation](#)

Page

- [Immutable Page](#)
- [Info](#)
- [Attachments](#)
- [More Actions:](#)

User

- [Login](#)

# BASIC ROS DEVELOPMENT

# Console Commands & Example

## Example

### Console Tab Nr. 1 – Starting a *roscore*

Start a roscore with

```
> roscore
```

```
student@ubuntu:~/catkin_ws$ roscore
... logging to /home/student/.ros/log/6c1852aa-e961-11e6-8543-000c297bd368/ros
launch-ubuntu-6696.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://ubuntu:34089/
ros_comm version 1.11.20

SUMMARY
=====

PARAMETERS
 * /rostdistro: indigo
 * /rosversion: 1.11.20

NODES

auto-starting new master
process[master]: started with pid [6708]
ROS_MASTER_URI=http://ubuntu:11311/

setting /run_id to 6c1852aa-e961-11e6-8543-000c297bd368
process[roscout-1]: started with pid [6721]
started core service [/roscout]
```

# Console Commands & Example

## Example

Console Tab Nr. 2 – Starting a *talker* node

Run a talker demo node with

```
> rosrunc roscpp_tutorials talker
```

```
student@ubuntu:~/catkin_ws$ rosrunc roscpp_tutorials talker
[ INFO] [1486051708.424661519]: hello world 0
[ INFO] [1486051708.525227845]: hello world 1
[ INFO] [1486051708.624747612]: hello world 2
[ INFO] [1486051708.724826782]: hello world 3
[ INFO] [1486051708.825928577]: hello world 4
[ INFO] [1486051708.925379775]: hello world 5
[ INFO] [1486051709.024971132]: hello world 6
[ INFO] [1486051709.125450960]: hello world 7
[ INFO] [1486051709.225272747]: hello world 8
[ INFO] [1486051709.325389210]: hello world 9
```

# Console Commands & Example

## Example

### Console Tab Nr. 3 – Analyze *talker* node

See the list of active nodes

```
> rosnodet list
```

```
student@ubuntu:~/catkin_ws$ rosnodet list
/rosout
/talker
```

Show information about the *talker* node

```
> rosnodet info /talker
```

```
student@ubuntu:~/catkin_ws$ rosnodet info /talker
-----
..
Node [/talker]
Publications:
  * /chatter [std_msgs/String]
  * /rosout [rosgRaph_msgs/Log]
Subscriptions: None
Services:
  * /talker/get_loggers
  * /talker/set_logger_level
```

See information about the *chatter* topic

```
> rostopic info /chatter
```

```
student@ubuntu:~/catkin_ws$ rostopic info /chatter
Type: std_msgs/String

Publishers:
  * /talker (http://ubuntu:39173/)
Subscribers: None
```

# Console Commands & Example

## Example

### Console Tab Nr. 3 – Analyze *chatter* topic

Check the type of the *chatter* topic

```
> rostopic type /chatter
```

```
student@ubuntu:~/catkin_ws$ rostopic type /chatter  
std_msgs/String
```

Show the message contents of the topic

```
> rostopic echo /chatter
```

```
student@ubuntu:~/catkin_ws$ rostopic echo /chatter  
data: hello world 11874  
...  
data: hello world 11875  
...  
data: hello world 11876
```

Analyze the frequency

```
> rostopic hz /chatter
```

```
student@ubuntu:~/catkin_ws$ rostopic hz /chatter  
subscribed to [/chatter]  
average rate: 9.991  
  min: 0.099s max: 0.101s std dev: 0.00076s window: 10  
average rate: 9.996  
  min: 0.099s max: 0.101s std dev: 0.00069s window: 20
```

# Console Commands & Example

## Example

Console Tab Nr. 4 – Starting a *listener* node

Run a listener demo node with

```
> rosrunc roscpp_tutorials listener
```

```
student@ubuntu:~/catkin_ws$ rosrunc roscpp_tutorials listener
[ INFO] [1486053802.204104598]: I heard: [hello world 19548]
[ INFO] [1486053802.304538827]: I heard: [hello world 19549]
[ INFO] [1486053802.403853395]: I heard: [hello world 19550]
[ INFO] [1486053802.504438133]: I heard: [hello world 19551]
[ INFO] [1486053802.604297608]: I heard: [hello world 19552]
```

# Console Commands & Example


## Example

### Console Tab Nr. 3 – Analyze

See the new *listener* node with

```
> rosnodet list
```

```
student@ubuntu:~/catkin_ws$ rosnodet list
/listener
/rosout
/talker
```





Show the connection of the nodes over the chatter topic with

```
> rostopic info /chatter
```

```
student@ubuntu:~/catkin_ws$ rostopic info /chatter
Type: std_msgs/String

Publishers:
 * /talker (http://ubuntu:39173/)

Subscribers:
 * /listener (http://ubuntu:34664/)
```





# Console Commands & Example

## Example

### Console Tab Nr. 3 – Publish Message from Console

Close the *talker* node in console nr. 2 with Ctrl + C

Publish your own message with

```
> rostopic pub /chatter std_msgs/String  
"data: 'Insert Own Text'"
```

```
student@ubuntu:~/catkin_ws$ rostopic pub /chatter std_msgs/String "data: 'ETH  
Zurich ROS Course'"  
publishing and latching message. Press ctrl-C to terminate
```

Check the output of the *listener* in console nr. 4

```
[INFO] [1486054667.684322265]: I heard: [hello world 28202]  
[INFO] [1486054667.704264199]: I heard: [hello world 28203]  
[INFO] [1486054667.804389858]: I heard: [hello world 28204]  
[INFO] [1486054707.646404558]: I heard:
```

Display Own Text



# Exercise: Publisher and Subscriber

## 1. Creating a ROS Package

- <http://wiki.ros.org/ROS/Tutorials/CreatingPackage>

## 2. Writing a Simple Publisher and Subscriber (Python)

- <http://wiki.ros.org/ROS/Tutorials/WritingPublisherSubscriber%28python%29>