

 免費電子書

學習

ROS

Free unaffiliated eBook created from
Stack Overflow contributors.

#ROS

.....	1
1: ros	2
.....	2
.....	2
Examples.....	2
.....	2
Hello World Publisher.....	3
2: roslaunch	6
.....	6
Examples.....	6
Yamlros.....	6
3:	9
.....	9
Examples.....	9
rospy.....	9
4:	10
.....	10
Examples.....	10
.....	10
.....	11

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [ros](#)

It is an unofficial and free ros ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official ros.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

1: ros

ROS

2007 STAIR 2008 2013 Willow Garage 2013 ROS

ROS ROSBSD

ROS ROS ROS /

ROS Unix ROS Ubuntu Mac OS X ROS Fedora Gentoo Arch Linux Linux ROS C ++ Python

ROS 10 ROS Kinetic

ROS ROS <http://www.ros.org/>

Ros Distro	Ubuntu	
Kinetic Kame	15.10, 16.04	2016-5-23
	14.04, 14.10, 15.04	2015-5-23
Indigo Igloo	13.10, 14.04	2014-7-22
Hydro Medusa	12.04, 12.10, 13.04	2013-9-4
Groovy Galapagos	11.10, 12.04, 12.10	2012-12-31
	10.04, 11.10, 12.04	2012-04-23
Emys	10.04, 10.10, 11.04, 11.10	2011-08-30
	10.04, 10.10, 11.04	2011-03-02
C	9.04, 9.10, 10.04, 10.10	2010-08-02
	8.04	2010-03-02

Examples

ROS ROS wiki ROS ROS

ROS			
	Ubuntu 16.04 Xenial	amd64 / i386 / armhf	Xenial
	Ubuntu 15.10 Wily	amd64 / i386	

ROS				
	Debian 8	amd64 / arm64		
	OS XHomebrew	-		
	Gentoo	-		Gentoo
	OpenEmbedded/ Yocto	-		Yocto

...

Hello World Publisher

```
mkdir -p ~/catkin_ws/src
cd ~/catkin_ws/src
catkin_init_workspace
```

```
cd ~/catkin_ws/
catkin_make
```

```
source devel/setup.bash
```

hello_world

```
catkin_create_pkg hello_world std_msgs rospy roscpp
```

srctalker.cpp

```
cd hello_world/src
touch talker.cpp
```

“hello world”

```
#include "ros/ros.h"
#include "std_msgs/String.h"

#include <sstream>

int main(int argc, char **argv)
{
    ros::init(argc, argv, "talker");

    ros::NodeHandle n;

    ros::Publisher chatter_pub = n.advertise<std_msgs::String>("chatter", 1000);

    ros::Rate loop_rate(10);

    int count = 0;
    while (ros::ok())
    {
```

```
std_msgs::String msg;

std::stringstream ss;
ss << "hello world " << count;
msg.data = ss.str();

ROS_INFO("%s", msg.data.c_str());

 chatter_pub.publish(msg);

ros::spinOnce();

loop_rate.sleep();
++count;
}

return 0;
}
```

```
cd ..
```

/CMakeLists.txt

```
catkin_package(
  INCLUDE_DIRS include
  LIBRARIES hello_world
  # CATKIN_DEPENDS roscpp rospy std_msgs
  # DEPENDS system_lib
)

include_directories(include ${catkin_INCLUDE_DIRS})

add_executable(talker src/talker.cpp)
target_link_libraries(talker ${catkin_LIBRARIES})
add_dependencies(talker hello_world_generate_messages_cpp)
```

```
cd ..
```

```
catkin_make
```

```
source devel/setup.bash
```

ROS

```
roscore
```

roscore/

```
roslaunch hello_world talker
```

```
/
```

```
rostopic echo /chatter
```

ros <https://riptutorial.com/zh-TW/ros/topic/7287/ros>


```

    <param name="name" value="bumblebeeLeft" />
  </node>

  <node name="$(arg name)" pkg="stereo_camera" type="stereo_camera" output="screen">
    <param name="name" value="bumblebeeCenter" />
  </node>
</launch>

```

◦

ROS

“name”

```
<param name="name" value="bumblebeeCenter" />
```

"\$(arg parameter_name)"◦

on the terminal "" (~/.ros) “screen”◦

ROSROS

roslaunch XML◦◦◦

```
roslaunch openni_launch_marvin kinect_left.launch
roslaunch openni_launch_marvin kinect_center.launch
```

```
<include file="$(find openni_launch_marvin)/launch/kinect_left.launch" />
<include file="$(find openni_launch_marvin)/launch/kinect_center.launch" />
```

roscdROS

◦ roslaunch "\$(find package_name)" relative to the package racine◦

“kinect_center.launch” “openni_launch_marvin” / launch / “◦

YAML

ROSYAMLROS “rosparam”◦ “rosparam YAMLROS◦◦◦ “

YAML

```
<rosparam command="load" file="$(find marvin_cameras)/config/marvin_cameras.yaml" />
```

YAML “marvin_cameras.yaml” “marvin_cameras / config /”◦

“solution.launch”◦

solution.launch

```
<launch>
```

```
<rosparam command="load" file="$(find marvin_cameras)/config/marvin_cameras.yaml" />

<node name="$(arg name)" pkg="stereo_camera" type="stereo_camera" output="screen">
  <param name="name" value="bumblebeeLeft" />
</node>

<node name="$(arg name)" pkg="stereo_camera" type="stereo_camera" output="screen">
  <param name="name" value="bumblebeeCenter" />
</node>

<include file="$(find openni_launch_marvin)/launch/kinect_left.launch" />
<include file="$(find openni_launch_marvin)/launch/kinect_center.launch" />

</launch>
```

roslaunch。

roslaunch <https://riptutorial.com/zh-TW/ros/topic/7361/roslaunch>

3:

ROS^o src^o CMakeLists.txtpackage.xml^o

Examples

rospy

workspace_nameworkspace_name package_name^o

```
$ cd ~/workspace_name/src/  
$ catkin_create_pkg package_name rospy
```

<https://riptutorial.com/zh-TW/ros/topic/8314/>

4:

- ROS◦ ROS◦

Examples

```
$ mkdir -p ~/workspace_name/src
$ cd ~/workspace_name/src
$ catkin_init_workspace
$ cd ~/workspace_name/
$ catkin_make
```

workspace_name workspace_name ◦

```
$ source ~/workspace_name/devel/setup.bash
```

<https://riptutorial.com/zh-TW/ros/topic/8313/>

S. No		Contributors
1	ros	ckirksey3 , Community , Photon , Vtik
2	roslaunch	ensonic , Vtik
3		Imiguelvargasf , Michael
4		Imiguelvargasf