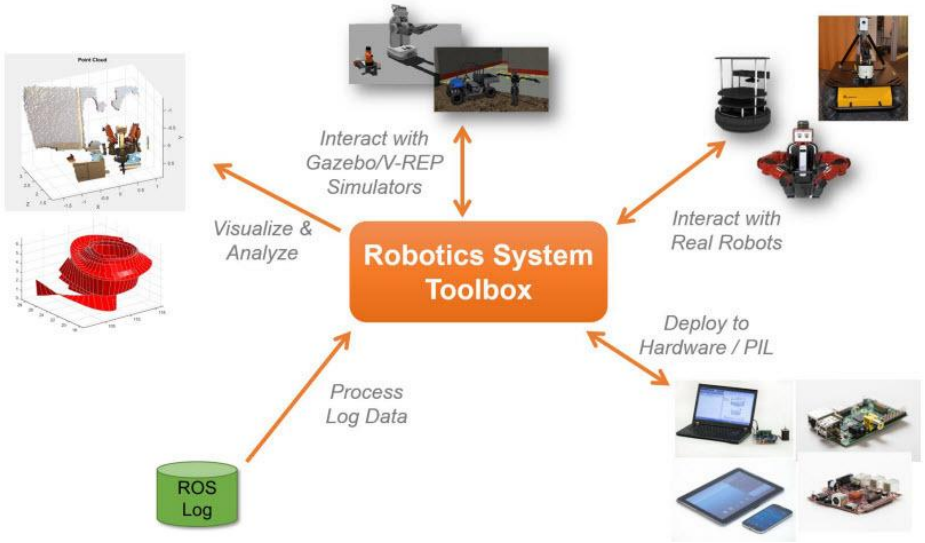
Ros課程

ROS Introduction

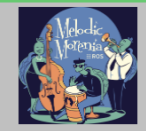
- ROS是一個開源的機器人操作系統
- 分散式架構，支援多節點運行且透過訊息通訊來實現數據交換
- 發佈（Publish）和訂閱（Subscribe）不同類型的訊息，包括感測器數據、控制指令等
- ROS提供強大的工具集，例如可視化工具Rviz或虛擬環境Gazebo，用於模擬和測試機器人行為
- ROS套件管理器（rospack）和依賴管理器（rosdep）使開發人員更容易分享和重用程式碼



ROS version

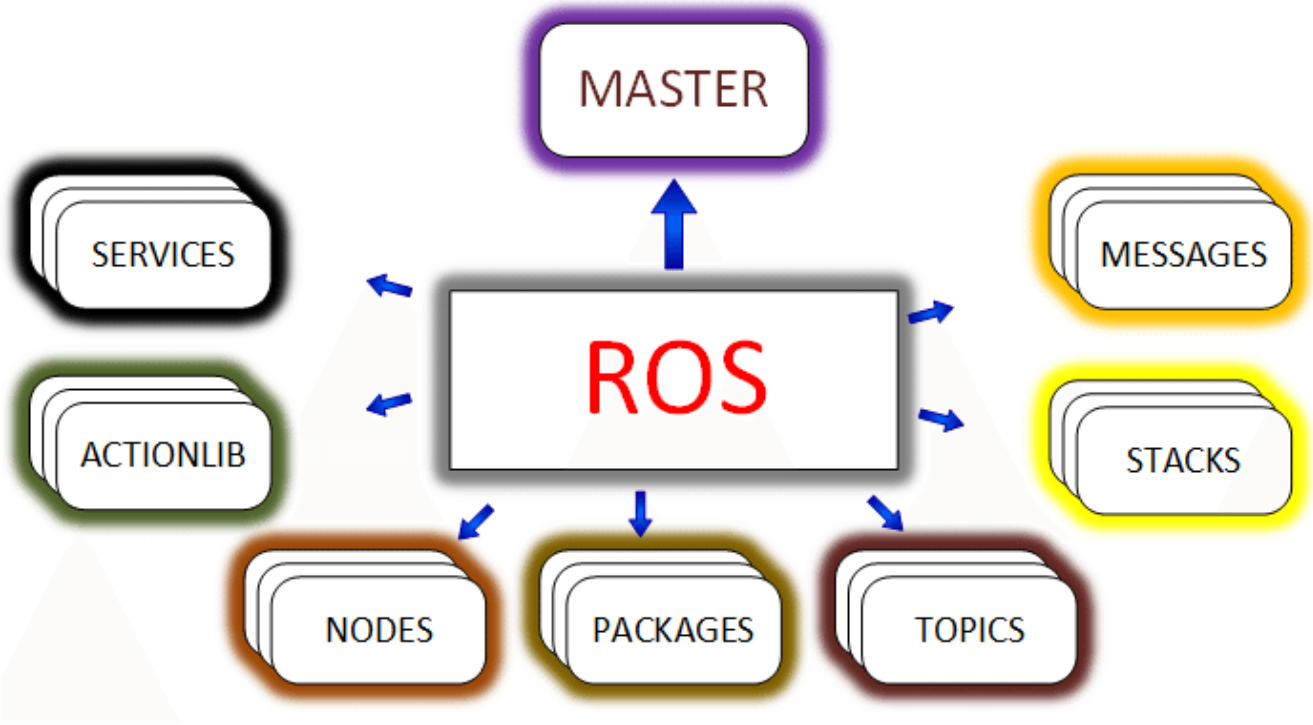
- 運行於Unix-based平台，其中包括Ubuntu和Mac OS X
- 支援C++和Python程式語言

- ROS version
  - ROS1
  - ROS2
- ROS1 version
  - **Noetic(Ubuntu 20.04)**
  - Melodic(Ubuntu 18.04)
  - Kinetic(Ubuntu 16.04)

Distro	Release date	Poster	*Tuturtle*, turtle in tutorial	EOL date
ROS Noetic Ninjemys (**Recommended**)	May 23rd, 2020			May, 2025 (Focal EOL)
ROS Melodic Morenia	May 23rd, 2018			June 27, 2023 (Bionic EOL)
ROS Lunar Loggerhead	May 23rd, 2017			May, 2019
ROS Kinetic Kame	May 23rd, 2016			April, 2021 (Xenial EOL)
ROS Jade Turtle	May 23rd, 2015			May, 2017
ROS Indigo Igloo	July 22nd, 2014			April, 2019 (Trusty EOL)
ROS Hydro Medusa	September 4th, 2013			May, 2015
ROS Groovy Galapagos	December 31, 2012			July, 2014

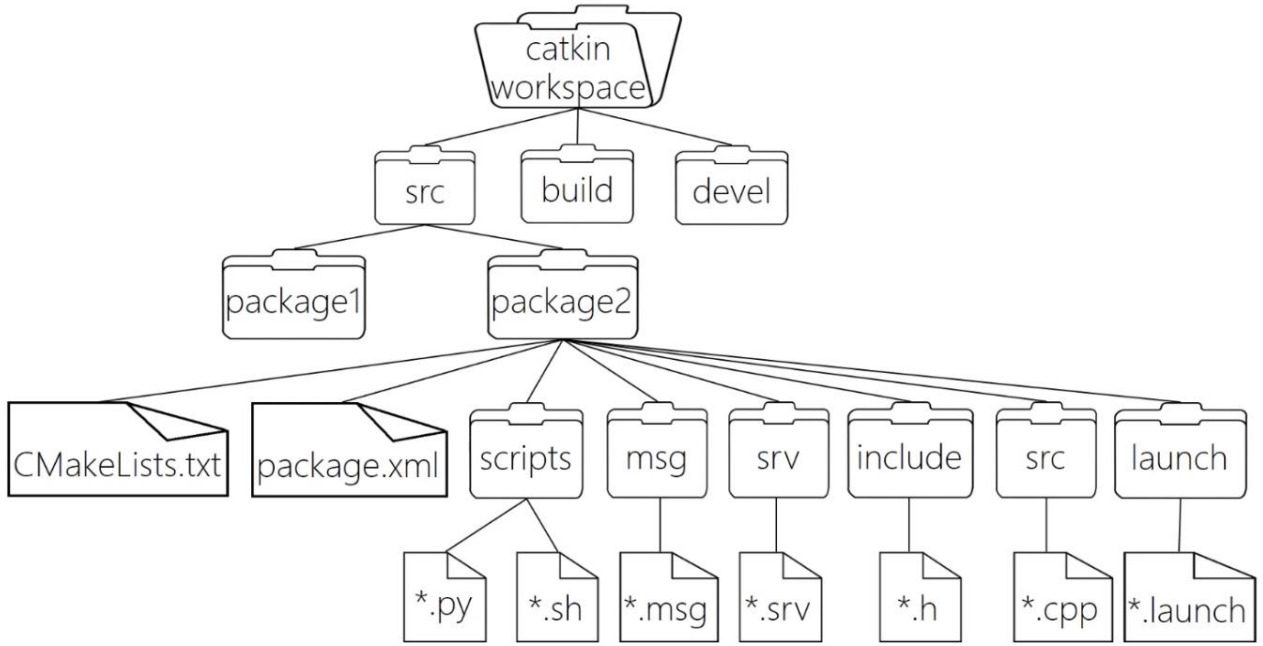
ROS Core Concepts

- 節點(Node)
- 主題(Topic)
- 服務(Service)
- 參數(Parameter)
- 套件(Package)
- ROS Master



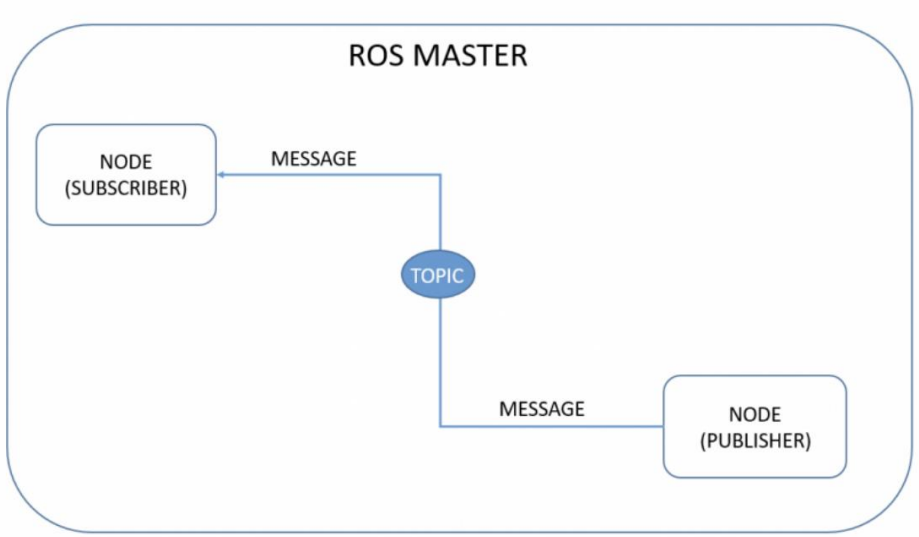
ROS catkin workspace

- src:用來放 package、程式碼的地方
- build:存放cmake 、catkin的暫存及中間文件
- devel:存放編譯後的目標文件，如標頭檔、動態鏈結庫、靜態連結庫、可執行文件
- CMakeLists.txt:用於指導ROS如何編譯和構建你的ROS package
- Package.xml:用於描述ROS package的原數據文件，如包名、作者、版本、依賴項等



Operation System

- Node
  - ROS系統中最基本的運行單位，是一個獨立的執行緒，可在系統中運行不同的程式
  - 節點之間透過消息進行通信，可以發佈（Publish）和訂閱（Subscribe）不同類型的訊息
- Topic
  - 用於在不同節點之間傳遞相同類型的消息
  - 發佈者（Publisher）節點將消息發佈到特定主題，而訂閱者（Subscriber）節點可以從該主題接收並處理消息
- ROS Master
  - 是ROS系統的中央管理器，負責管理節點之間的通信
  - 所有的節點必須向ROS Master註冊，以便進行通信和尋找其他節點

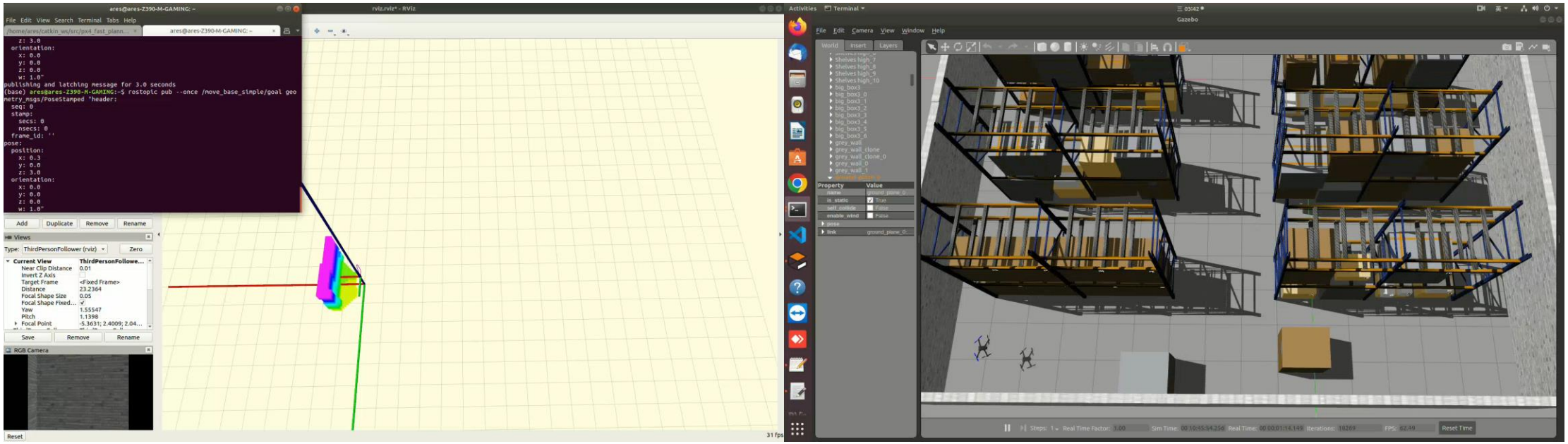


Launch file

- ROS launch file是一種XML格式的檔案，用於配置和啟動ROS系統中的節點和其他組件
- launch file可以在一次運行中同時啟動多個節點，設定節點的參數，訂閱和發佈主題，以及進行其他配置
- launch file的結構包含節點（node）、參數（param）、重映射（remap）、引數（arg）等元素
- 可以使用參數和引數，使launch file更具彈性，根據不同的需求，修改節點參數或者傳遞引數
- 可以使用include指令在一個launch file中引用其他的launch file，這使得系統配置更為模組化和可重複

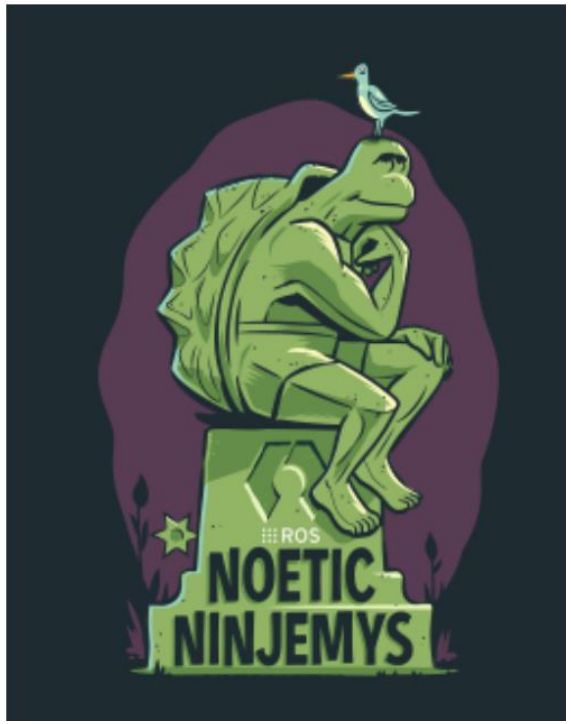
Gazebo & Rviz

- Gazebo
  - 構建機器人運動模擬模型
  - 構建現實世界各種場景的模擬模型
  - 構建感測器模擬模型
  - 為機器人模型新增現實世界物理性質
- Rviz
  - 三維參數可視化軟體,
  - 將機器人於 Gazebo 運行時所取得的參數,具體化呈現
  - 利用 gazebo 創造資料及引數,並使用 Rviz 來訂閱相關 Topic,並完成視覺化的渲染，讓開發者更容易理解資料的意義

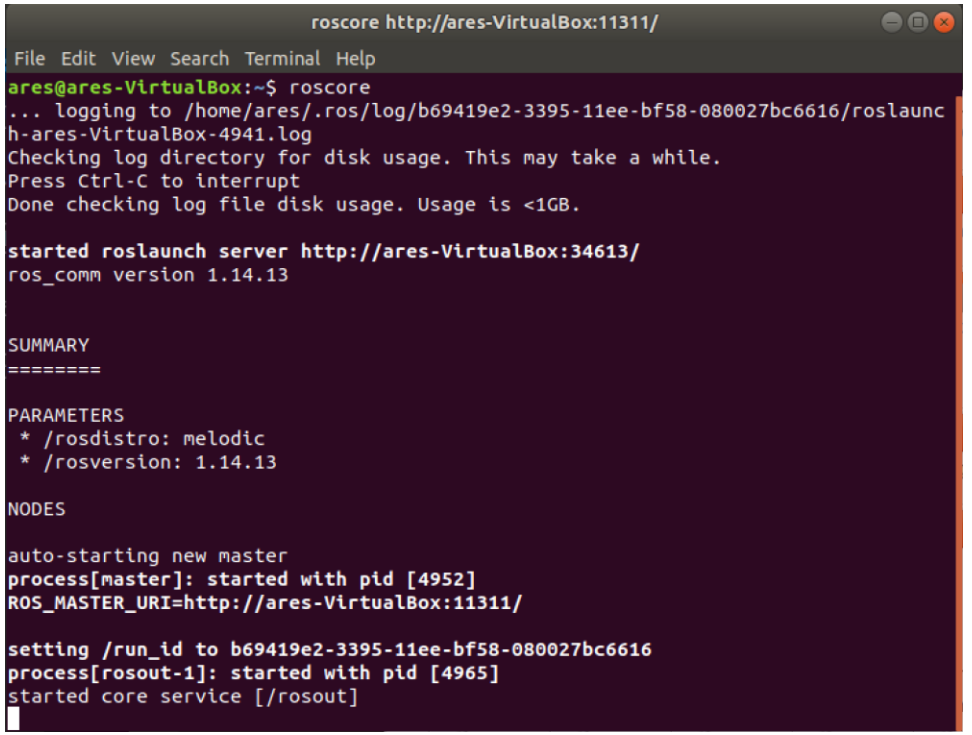

ares@ares-Z390-M-GAMING: ~
rviz.rviz* - RViz
Gazebo
Add
Duplicate
Remove
Rename
Views
RGB Camera
Reset

Install ROS

- http://wiki.ros.org/noetic/Installation/Ubuntu
- Ubuntu 20.04 -> ROS Noetic
- Key roscore

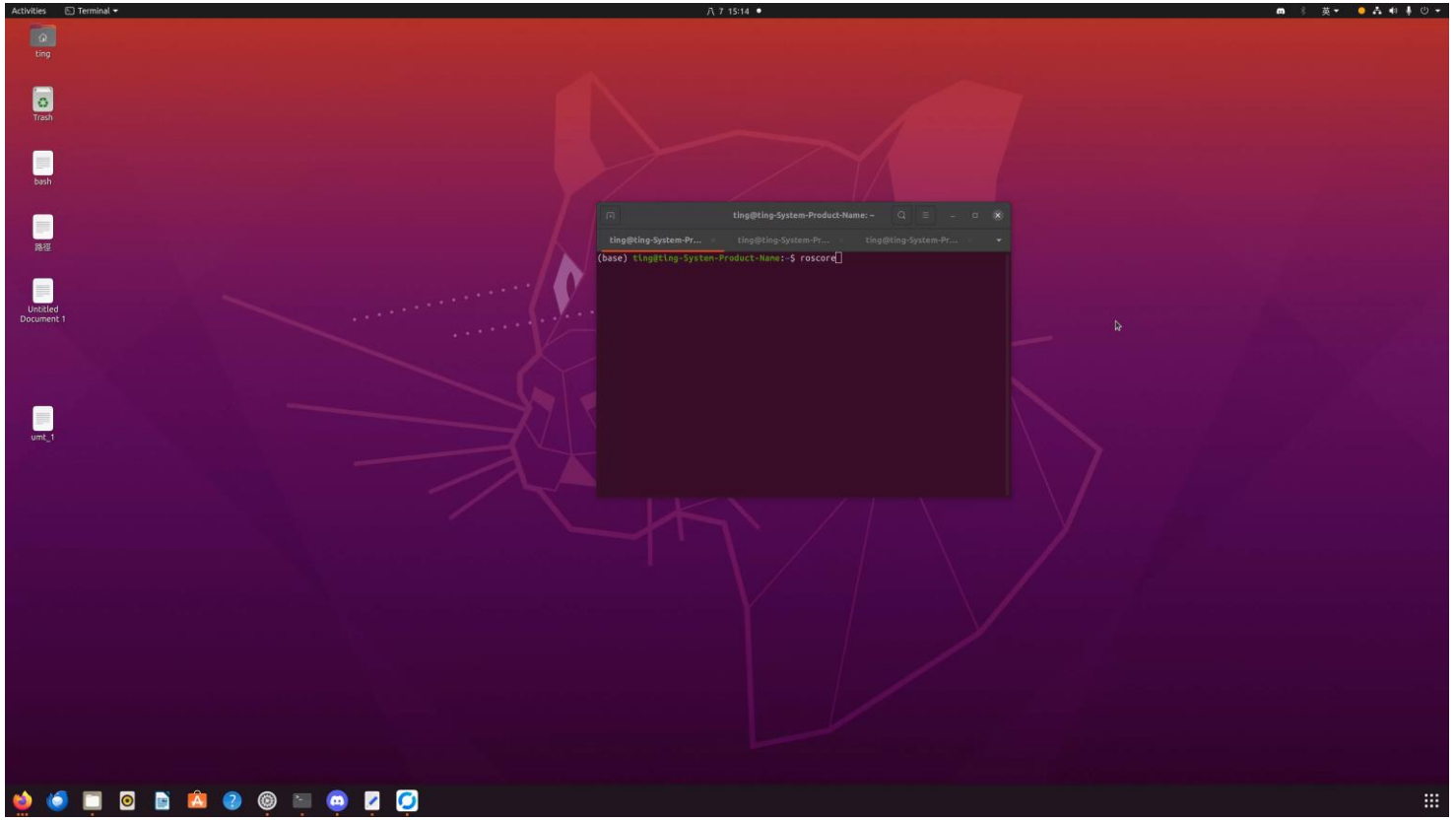


範例影片



Turtlebot simulator

- http://wiki.ros.org/turtlesim
- rosrun turtlesim turtle_teleop_key



Catkin workspace & ROS Package

1. mkdir -p ~/catkin_ws/src
2. cd catkin_ws/
3. catkin_make
4. source ~/catkin_ws/devel/setup.bash
5. cd src
6. catkin_create_pkg <package_name> std_msgs rospy roscpp
7. cd ..
8. catkin_make
9. source ~/catkin_ws/devel/setup.bash
10. rospack find <package_name>

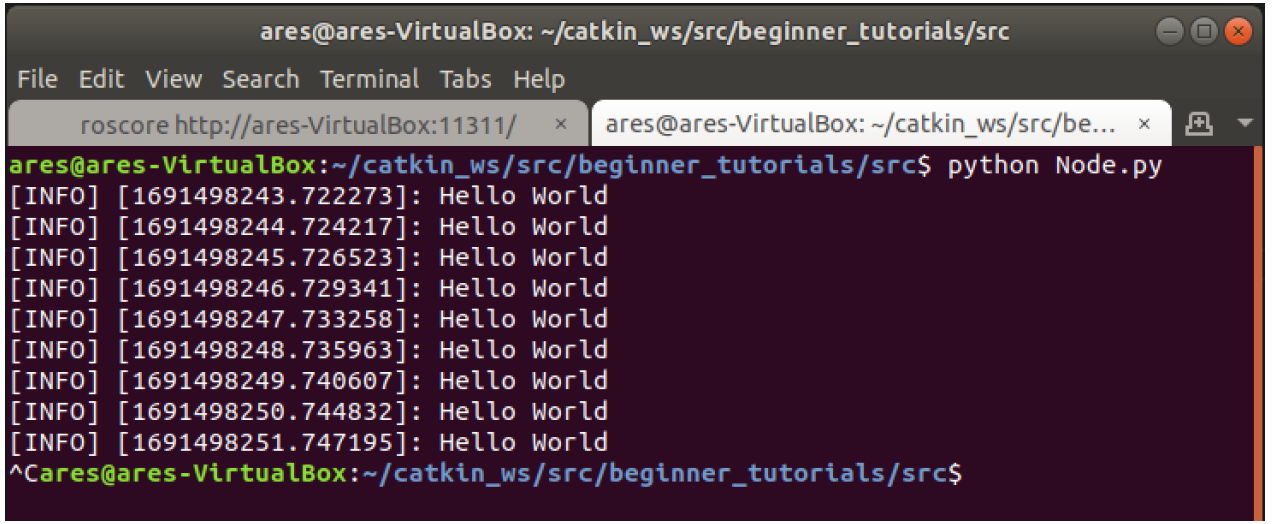
Implement ROS node

1. roscd <package_name>/src/
2. 建立.py file

talker.py | subscriber.py | Node.py

home > ares > catkin_ws > src > beginner_tutorials > src > Node.py

```python
#!/usr/bin/env python
import rospy

rospy.init_node('hello_python_node')

while not rospy.is_shutdown():
    rospy.loginfo('Hello World')
    rospy.sleep(1)

```



3. roscore(another Terminal)
4. Python file_name.py

Implement ROS Publisher && subscriber

talker.py | subscriber.py | Node.py

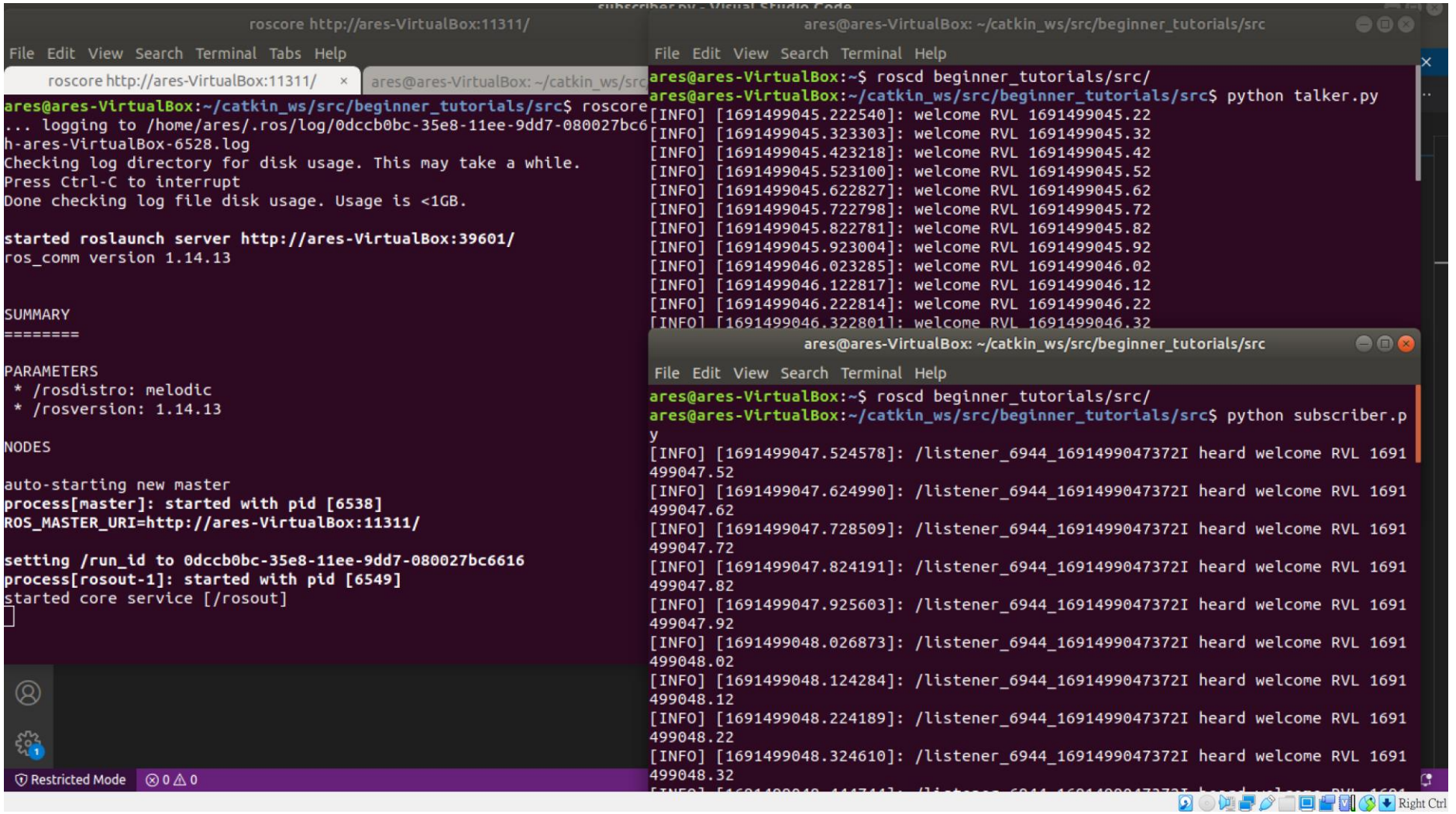
home > ares > catkin_ws > src > beginner_tutorials > src > talker.py > ...

```python
#!/usr/bin/env python
import rospy
from std_msgs.msg import String

def talker():
    pub = rospy.Publisher('chatter',String,queue_size = 10)
    rospy.init_node('talker',anonymous = True)
    rate = rospy.Rate(10)
    while not rospy.is_shutdown():
        hello_str = "welcome RVL %s" % rospy.get_time()
        rospy.loginfo(hello_str)
        pub.publish(hello_str)
        rate.sleep()

if __name__ == '__main__':
    try:
        talker()
    except rospy.ROSInterruptException:
        pass
```

talker.py | subscriber.py | Node.py

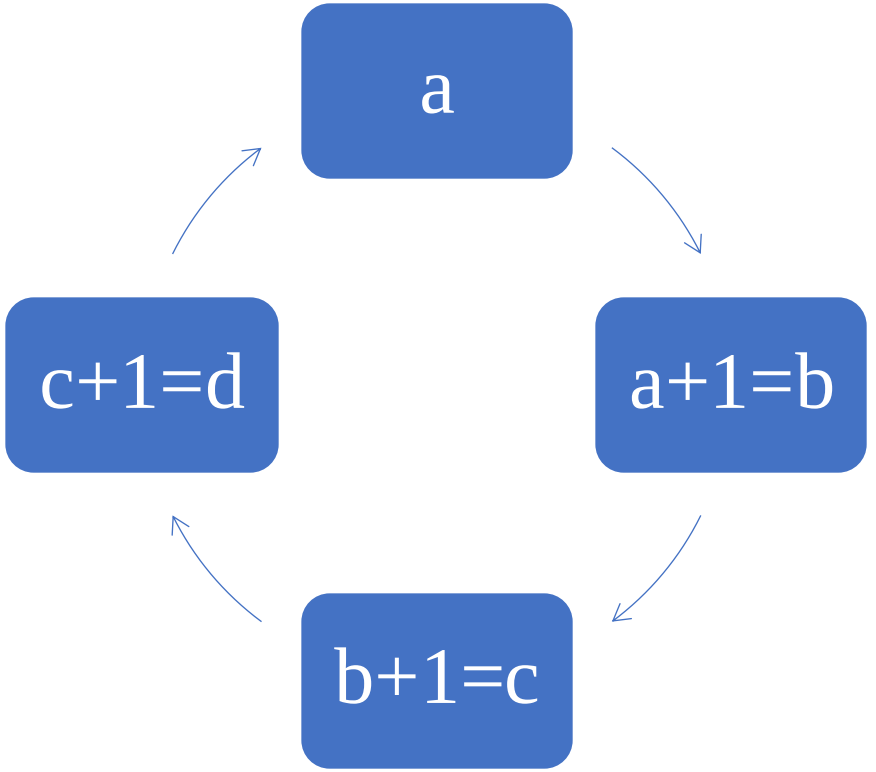
home > ares > catkin_ws > src > beginner_tutorials > src > subscriber.py > listner

```python
#!/usr/bin/env python
import rospy
from std_msgs.msg import String

def callback(data):
    rospy.loginfo(rospy.get_caller_id() + "I heard %s",data.data)

def listner():
    rospy.init_node('listener',anonymous=True)
    rospy.Subscriber("chatter",String,callback)
    rospy.spin()

if __name__ == '__main__':
    listner()
```


roscore http://ares-VirtualBox:11311/
File Edit View Search Terminal Tabs Help
roscore http://ares-VirtualBox:11311/
ares@ares-VirtualBox:~/catkin_ws/src/beginner_tutorials/src$ roscore
... logging to /home/ares/.ros/log/0dccb0bc-35e8-11ee-9dd7-080027bc6
h-ares-VirtualBox-6528.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.
started roslaunch server http://ares-VirtualBox:39601/
ros_comm version 1.14.13
SUMMARY
========
PARAMETERS
* /rosdistro: melodic
* /rosversion: 1.14.13
NODES
auto-starting new master
process[master]: started with pid [6538]
ROS_MASTER_URI=http://ares-VirtualBox:11311/
setting /run_id to 0dccb0bc-35e8-11ee-9dd7-080027bc6616
process[rosout-1]: started with pid [6549]
started core service [/rosout]
Restricted Mode
ares@ares-VirtualBox: ~/catkin_ws/src/beginner_tutorials/src
File Edit View Search Terminal Help
ares@ares-VirtualBox:~$ roscd beginner_tutorials/src/
ares@ares-VirtualBox:~/catkin_ws/src/beginner_tutorials/src$ python talker.py
[INFO] [1691499045.222540]: welcome RVL 1691499045.22
[INFO] [1691499045.323303]: welcome RVL 1691499045.32
[INFO] [1691499045.423218]: welcome RVL 1691499045.42
[INFO] [1691499045.523100]: welcome RVL 1691499045.52
[INFO] [1691499045.622827]: welcome RVL 1691499045.62
[INFO] [1691499045.722798]: welcome RVL 1691499045.72
[INFO] [1691499045.822781]: welcome RVL 1691499045.82
[INFO] [1691499045.923004]: welcome RVL 1691499045.92
[INFO] [1691499046.023285]: welcome RVL 1691499046.02
[INFO] [1691499046.122817]: welcome RVL 1691499046.12
[INFO] [1691499046.222814]: welcome RVL 1691499046.22
ares@ares-VirtualBox: ~/catkin_ws/src/beginner_tutorials/src
File Edit View Search Terminal Help
ares@ares-VirtualBox:~$ roscd beginner_tutorials/src/
ares@ares-VirtualBox:~/catkin_ws/src/beginner_tutorials/src$ python subscriber.p
y
[INFO] [1691499047.524578]: /listener_6944_16914990473721 heard welcome RVL 1691
499047.52
[INFO] [1691499047.624990]: /listener_6944_16914990473721 heard welcome RVL 1691
499047.62
[INFO] [1691499047.728509]: /listener_6944_16914990473721 heard welcome RVL 1691
499047.72
[INFO] [1691499047.824191]: /listener_6944_16914990473721 heard welcome RVL 1691
499047.82
[INFO] [1691499047.925603]: /listener_6944_16914990473721 heard welcome RVL 1691
499047.92
[INFO] [1691499048.026873]: /listener_6944_16914990473721 heard welcome RVL 1691
499048.02
[INFO] [1691499048.124284]: /listener_6944_16914990473721 heard welcome RVL 1691
499048.12
[INFO] [1691499048.224189]: /listener_6944_16914990473721 heard welcome RVL 1691
499048.22
[INFO] [1691499048.324610]: /listener_6944_16914990473721 heard welcome RVL 1691
499048.32
Right Ctrl

Homework1 訊息發佈練習

- 四節點迴圈型Pub & Sub
- C++(禁止使用python)
- 每一節點皆同時為Publisher和Subscriber
- 一開始發布數值:整數1
- 每完成一次迴圈時請讓接收的整數加1
- 上傳內容:程式碼及結果影片
- 繳交期限:8/21



Homework2 專案建立練習

- PX4無人機利用Mavros於Gazebo運行
- **https://docs.px4.io/main/en/dev_setup/dev_env_linux_ubuntu.html#ros-gazebo-classic**
- **https://docs.px4.io/main/en/ros/mavros_installation.html**
- **https://docs.px4.io/main/en/ros/mavros_offboard_python.html**
- 上傳內容:結果影片
- 繳交期限:8/21