

---

# ROS TA Session

- Ubuntu basics
  - ROS install
  - ROS execution
  - Create ROS node
  - Concepts of compiler, g++, Makefile & cmake
- 

2018 . 03. 19  
Juhwan Seo

# Before Start

---

- The reason why you feel difficulties about Ubuntu is that you have no information.

**The core purpose of this session!**

- You will see lots of errors. To solve them, you should understand why the errors are appeared. To understand, you should know. To know, you should hear at least once

**Best**

**Better**

**Good enough!**

- Don't believe me 100%. I could be wrong. Wrongly-delivered contents should be studied and corrected by yourself.

- If you installed Ubuntu in Korea, I recommend you to re-install in English for your future, not mine.

- This material is not kind. Sorry for that. Please concentrate on my explanation.

# Contents

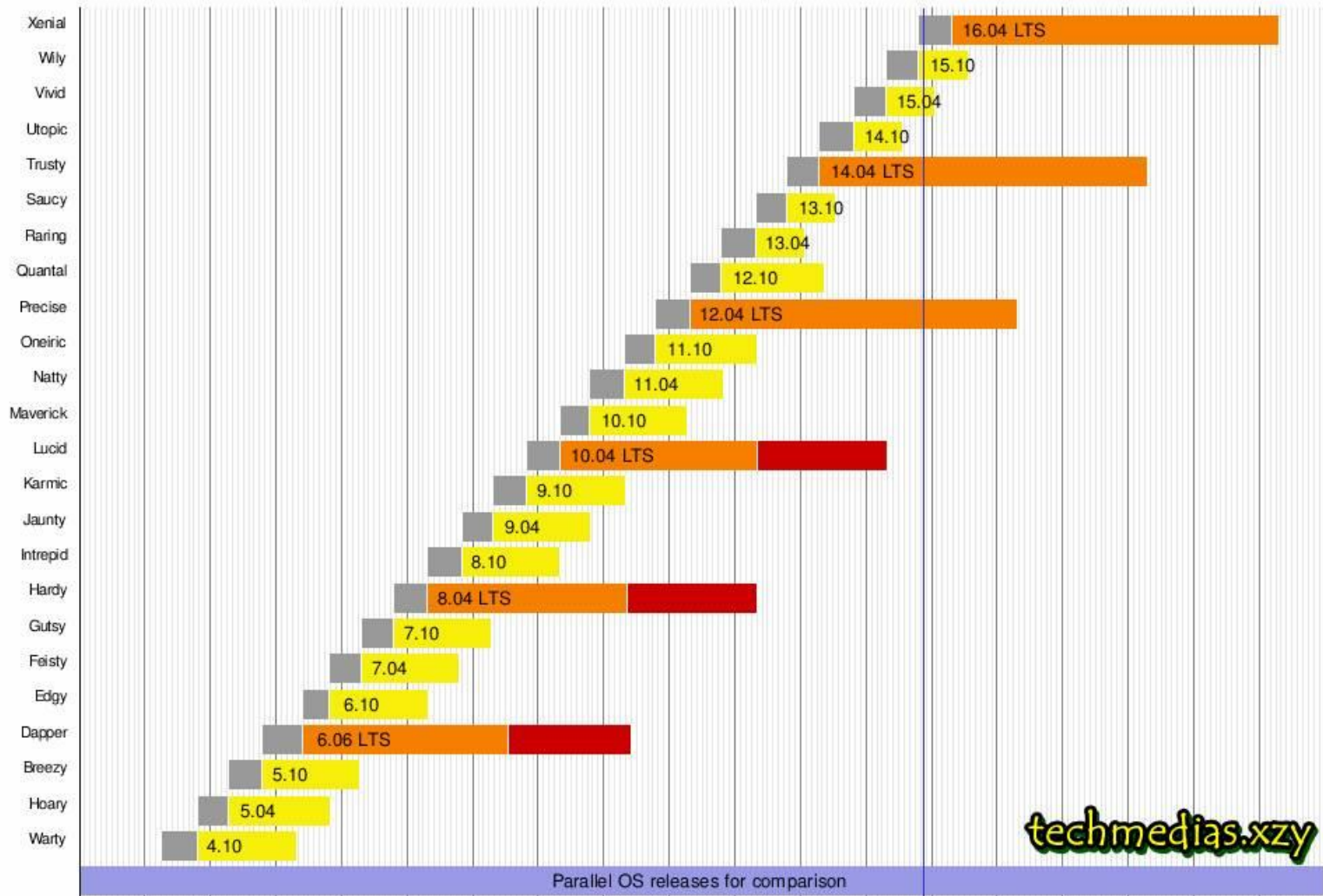
---

- Ubuntu
  - install 16.04 LTS
  - TTY(Tele-TYpewriter); brief history of computer
  - commands
    - file&folder management
    - compress/decompress
    - installation of applications; concept of mirror site
  - shell script
  - bashrc
- Re-visit the questions on KakaoTalk group chat.
- ROS install
  - what is ROS? → will be skipped. See Dr.Pyo's lecture notes.
  - install ROS kinetic
- Create ROS node (\*similar to bonus assignment)
  - step by step programming
  - concepts of compiler, g++, Makefile&cmake

# 0. Ubuntu – About version

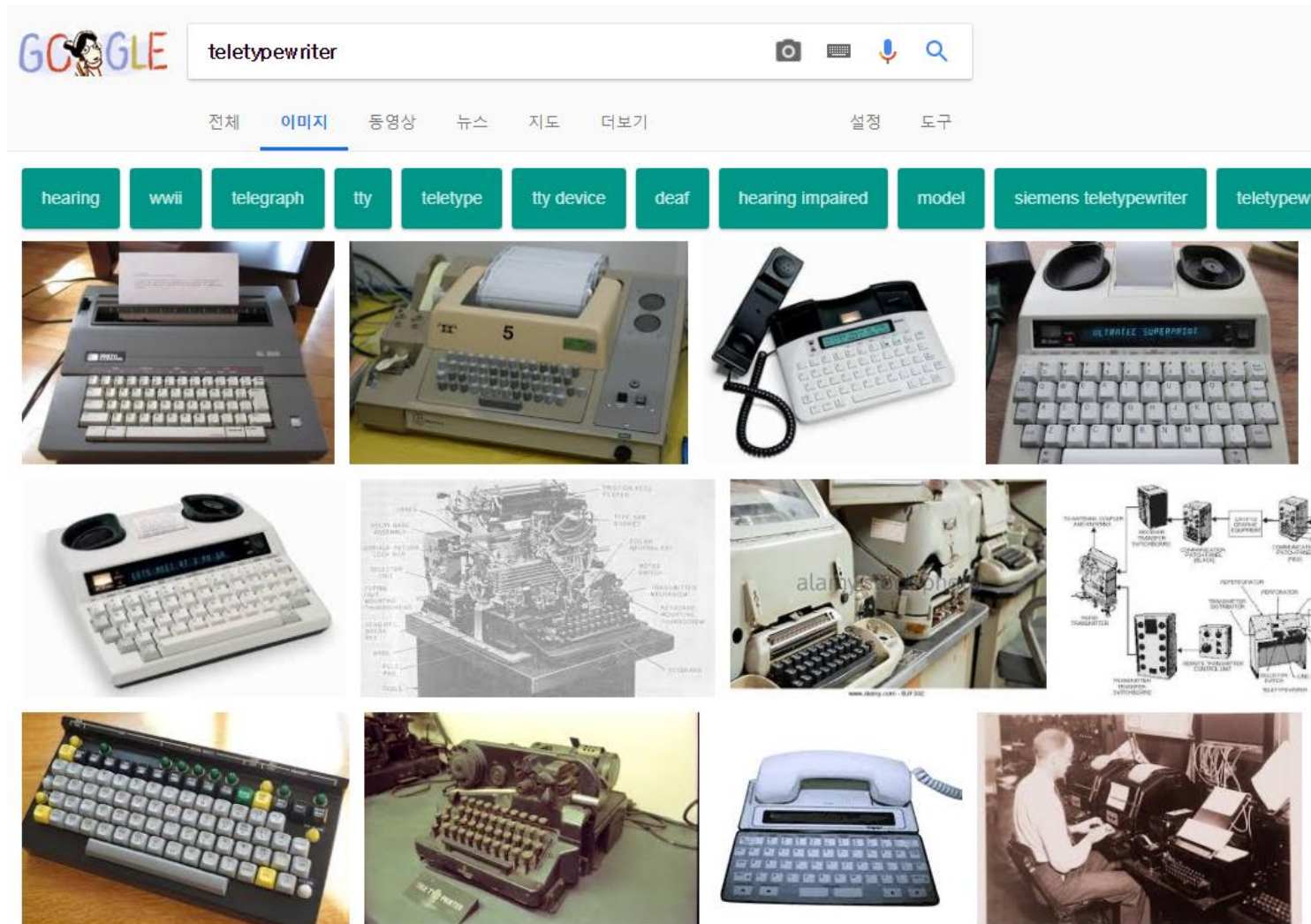
- There are many versions ...
- Ubuntu yy.mm (LTS or nothing)    yy: year, mm: month, LTS:Long-term support  
→ Install LTS version!!

Ubuntu release timeline



# 0. Ubuntu – About TTY

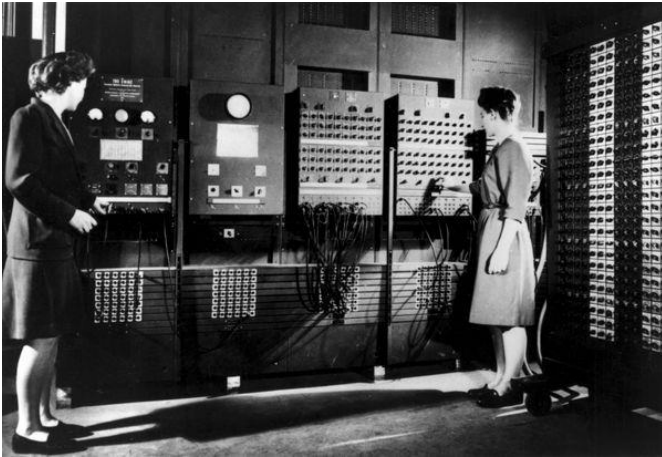
- Let's press ctrl+alt+F1~F7
- TTY: Tele-TYpewriter
- Concept of computer at early stage...



# 0. Ubuntu – About TTY

- Let's press ctrl+alt+F1~F7
- TTY: Tele-TYpewriter
- Concept of computer at early stage...

In the early stage....



\*too big, cannot have it personally...



user1



user2



user3

"There is no reason for any individual to have a computer in his home."



Present



user1



"A computer on every desk, and in every home, running Microsoft software."

# 0. Ubuntu – About TTY

---

- Let's press ctrl+alt+F1~F7
  - TTY: Tele-TYpewriter
  - Concept of computer at early stage...
- 
- Basically, there is no GUI. Everything is done in text-based console.
  - Almost the same with terminal(next slide)
  - There is no need to do something with this.

However, sometimes you are needed to do something with TTY.

Ex) installation of Graphic Driver

(also, CUDA which will be required when installing Tensorflow in fall semester), ...

```
Ubuntu 12.10 upubuntu tty3
upubuntu login: upubuntu
Password:
Last login: Thu Jan 17 12:11:32 PST 2013 on tty2
Welcome to Ubuntu 12.10 (GNU/Linux 3.5.0-17-generic i686)

 * Documentation:  https://help.ubuntu.com/

upubuntu@upubuntu:~$ sudo fbcats > image.ppm
[sudo] password for upubuntu:
upubuntu@upubuntu:~$ _
```

# 0. Ubuntu – About Linux & Ubuntu

---

- OS(Operating System) consists of mainly three parts(kernels, interface and applications)
- In case of Windows,



- All three parts are provided by Microsoft.

- In case of Linux,

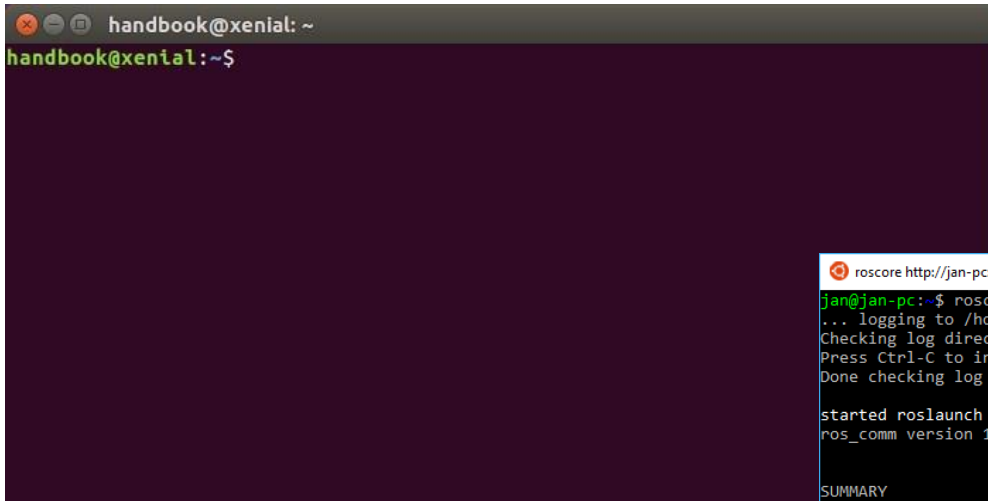


- Linux Kernel is provided by Linus Torvalds(Programmer)
- Interfaces are provided by many providers. (Ubuntu, CentOS, Linux Mint, Raspbian, ....)
- In conclusion, Ubuntu is one of Linux-based OS.
- For details, <http://moordev.tistory.com/222>

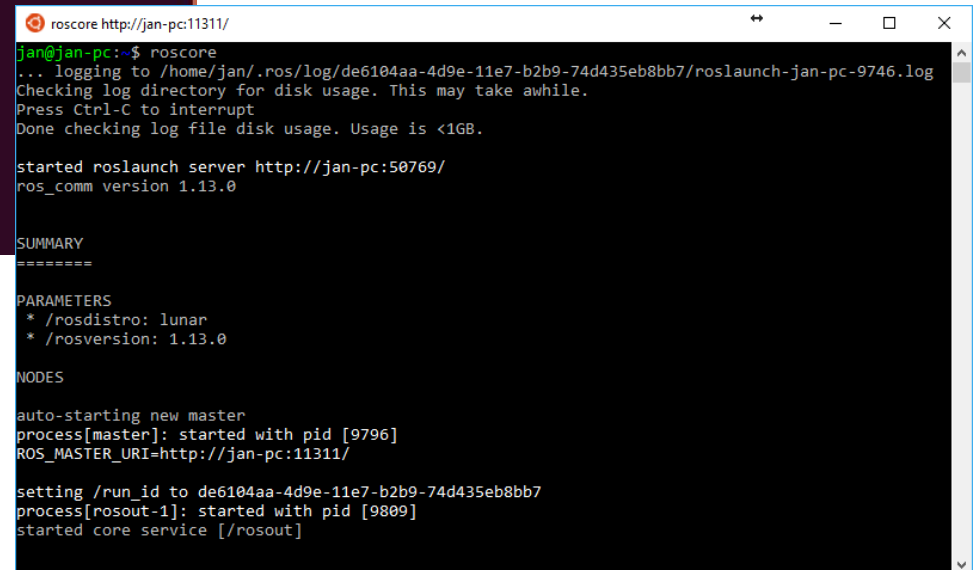


# 0. Ubuntu – About Terminal

- The thing you should master as soon as possible ...
- Shortcuts ) `ctrl+alt+t` : new terminal on new window  
`ctrl+shift+t` : new terminal on new tab  
`ctrl+page up/down` : tab change
- If a terminal is idle, you can do something with it.  
If a terminal is not idle, that means it is occupied by some process.

A screenshot of a terminal window with a dark purple background. The title bar shows 'handbook@xenial: ~'. The prompt is 'handbook@xenial:~\$'.

Idle terminal

A screenshot of a terminal window with a black background, titled 'roscore http://jan-pc:11311/'. The prompt is 'jan@jan-pc:~\$'. The terminal shows the output of the 'roscore' command, including logging information, disk usage checks, and the start of a ROS master and roslaunch server.

```
roscore http://jan-pc:11311/
jan@jan-pc:~$ roscore
... logging to /home/jan/.ros/log/de6104aa-4d9e-11e7-b2b9-74d435eb8bb7/roslaunch-jan-pc-9746.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://jan-pc:50769/
ros_comm version 1.13.0

SUMMARY
=====

PARAMETERS
* /rostdistro: lunar
* /rosversion: 1.13.0

NODES

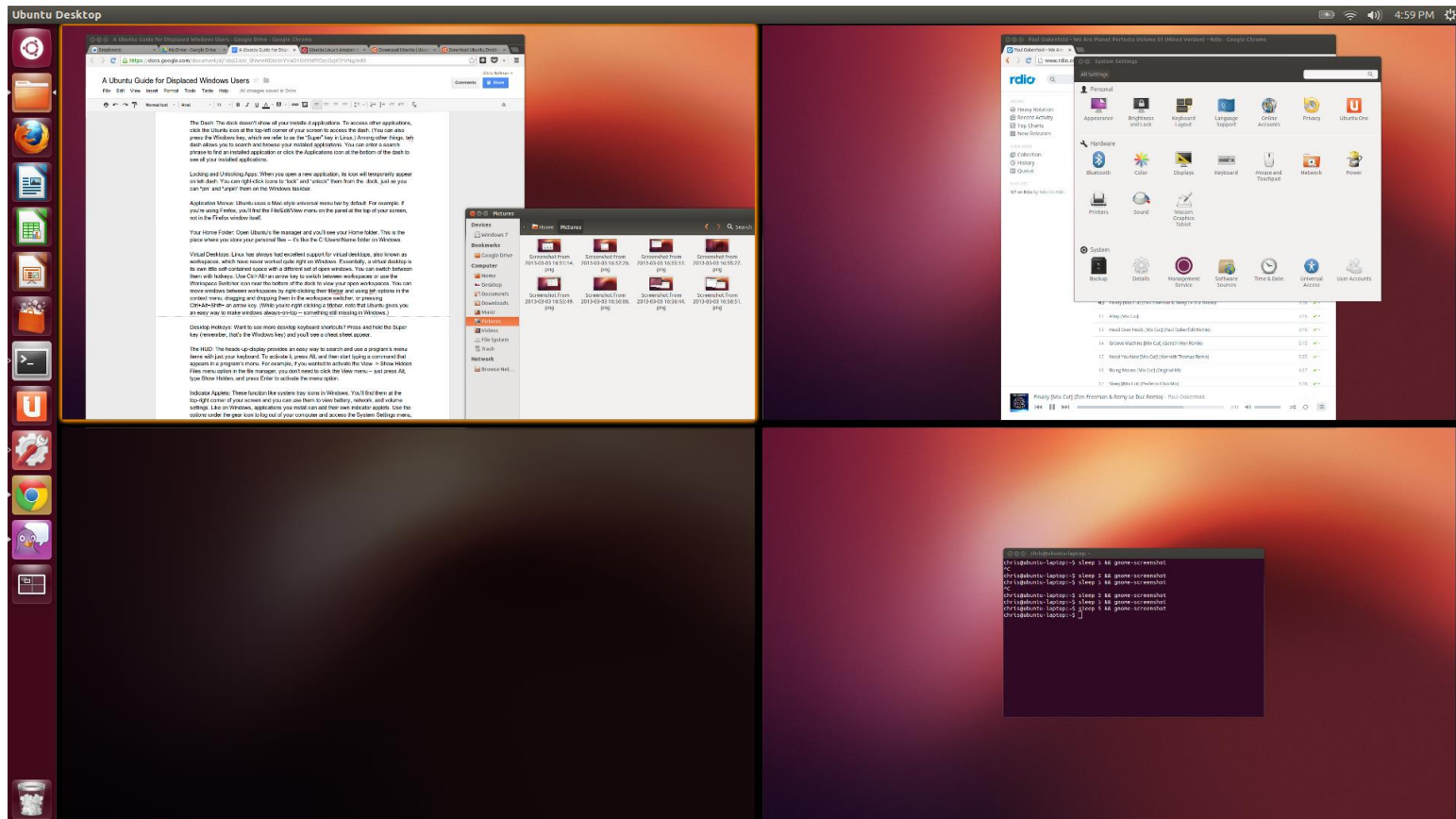
auto-starting new master
process[master]: started with pid [9796]
ROS_MASTER_URI=http://jan-pc:11311/

setting /run_id to de6104aa-4d9e-11e7-b2b9-74d435eb8bb7
process[rosout-1]: started with pid [9809]
started core service [/rosout]
```

Occupied terminal

# 0. Ubuntu – About Workspace

- For a year with capstone design, you will utilize lots of terminals.
- Screen space might not be enough.
- Enable workspace from ‘System Settings’ → ‘Appearance’ → ‘Behavior’ → check ‘Enable workspaces’
- Shortcuts) ctrl+alt+direction : change workspace  
ctrl+alt+shift+direction : change workspace & move current window



# 0. Ubuntu – About File&Folder management

- GUI is easy... but you still need to know this
- Shortcuts) `cd` : change directory, `ls`: list directory contents, `mv`: file move  
`rm` : remove, `mkdir`: make directory, `rmdir`: remove directory, `cp`: copy  
there are more and each command has options. Study by your self!
- `~`: home folder, `/`: the most basic part of Ubuntu system
- `sudo apt-get install tree & tree -d`

```
jake@ubuntu:~$ ls
jake@ubuntu:~$ ls -a
.  .bash_history  .bashrc  .profile
.. .bash_logout  .cache   .sudo_as_admin_successful
jake@ubuntu:~$ cd /
jake@ubuntu:/$ ls
bin  dev  home  lost+found  mnt  proc  run  snap  sys  usr
boot etc  lib  media      opt  root  sbin srv  tmp  var
jake@ubuntu:/$ cd home
jake@ubuntu:/home$ ls
jake  ubuntu
jake@ubuntu:/home$ cd jake
jake@ubuntu:~$ ls
jake@ubuntu:~$ cd ..
jake@ubuntu:/home$ cd ubuntu
jake@ubuntu:/home/ubuntu$ ls
jake@ubuntu:/home/ubuntu$ cd ~
jake@ubuntu:~$
```

↑  
account

↑  
Name of computer

↑  
Current directory

Your name:  ✓

Your computer's name:  ✓  
The name it uses when it talks to other computers.

Pick a username:  ✓

Choose a password:  Good password

Confirm your password:  ✓

☐ Log in automatically

☒ Require my password to log in

☐ Encrypt my home folder

# 0. Ubuntu – About File&Folder management

- GUI is easy... but you still need to know this
- Shortcuts) `cd` : change directory, `ls`: list directory contents, `mv`: file move  
`rm` : remove, `mkdir`: make directory, `rmdir`: remove directory, `cp`: copy  
there are more and each command has options. Study by your self!
- `~`: home folder, `/`: the most basic part of Ubuntu system
- `sudo apt-get install tree & tree -d`

```
hri@hri-laptop:/$ tree -L 1
```

```
.
├── bin
├── boot
├── cdrom
├── core
├── dev
├── etc
├── home
├── initrd.img -> boot/initrd.img-4.4.0-116-generic
├── initrd.img.old -> boot/initrd.img-4.4.0-112-generic
├── lib
├── lib32
├── lib64
├── libx32
├── lost+found
├── media
├── mnt
├── opt
├── proc
├── root
├── run
├── sbin
├── snap
├── srv
├── sys
├── tmp
├── usr
├── var
├── vmlinuz -> boot/vmlinuz-4.4.0-116-generic
└── vmlinuz.old -> boot/vmlinuz-4.4.0-112-generic
```

- bin(binary)
- boot(boot loader related)
- etc(Literally, etc...)
- dev(device)
- home(home folder, every users' folder will be here)
- lib, lib32, lib64, libx32(libraries for 32bit &64bit system)
- media(sdd, hdd, usb, ...)
- opt: softwares and add-on packages that are not part of the default installation
- ...

Study more by yourself!

## 0. Ubuntu – compress & decompress

---

- For ‘zip files’

zip (name of zip file) (path of the files you want to compress)

ex) zip capstonedesign.zip ~/capstone\_design

If you want to compress a folder with following folders, use ‘-r’ option

ex) zip -r capstonedesign.zip ~/capstone\_design

- For ‘tar.~~ files’

tar -xvf (path of the compressed files)

ex) tar -xvf opencv.tar.gz

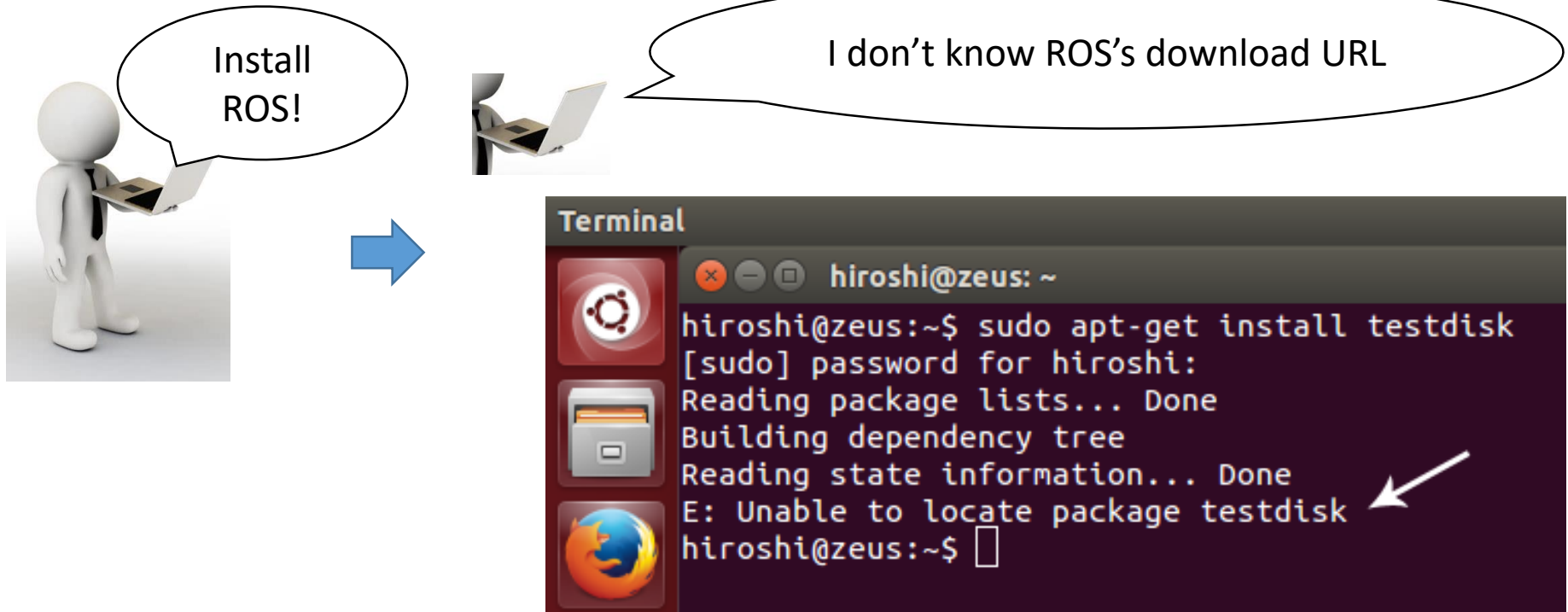
# 0. Ubuntu – About Package Installation

- Open-source based OS → Applications are free!
- Almost every programs and libraries are installed via internet.
- The process of installation in Ubuntu is like below.



# 0. Ubuntu – About Package Installation

- Open-source based OS → Applications are free!
- Almost every programs and libraries are installed via internet.
- The process of installation in Ubuntu is like below.



**Then, how can we make a computer know packages' download URL?**



# 0. Ubuntu – About Package Installation

---

- Open-source based OS → Applications are free!
  - Almost every programs and libraries are installed via internet.
  - The process of installation in Ubuntu is like below.
- `apt-get update` downloads the package lists from the repositories and "updates" them to get information on the newest versions of packages and their dependencies. It will do this for all repositories and PPAs. From <http://linux.die.net/man/8/apt-get>:  

Used to re-synchronize the package index files from their sources. The indexes of available packages are fetched from the location(s) specified in `/etc/apt/sources.list` (5). An update should always be performed before an upgrade or dist-upgrade.
  - `apt-get upgrade` will fetch new versions of packages existing on the machine if APT knows about these new versions by way of `apt-get update`.

From <http://linux.die.net/man/8/apt-get>:

Used to install the newest versions of all packages currently installed on the system from the sources enumerated in `/etc/apt/sources.list` (5). Packages currently installed with new versions available are retrieved and upgraded; under no circumstances are currently installed packages removed, nor are packages that are not already installed retrieved and installed. **New versions of currently installed packages that cannot be upgraded without changing the install status of another package will be left at their current version.** [Emphasis mine] An update must be performed first so that apt-get knows that new versions of packages are available.



# 0. Ubuntu – About Package Installation

---

- What if `/etc/apt/sources.list` is broken?
  - ➔ `sudo apt-get update` cannot output the packages' list.
  - ➔ you can't install packages not listed in your computer.
- I edited `/etc/apt/sources.list`. However, still it is not working
  - ➔ just editing `/etc/apt/sources.list` doesn't update the package list. When you modify `/etc/apt/sources.list` file, you should 'sudo apt-get update' to apply it
- If you want to check the list is refreshed, utilized 'tab'  
`sudo apt-get install ros-kinetic-(push tab here!)`  
then the list available will be appeared.
- I modified `/etc/apt/sources.list` & I can find the package I want to install when I input 'tab'
  - ➔ There are two possibility
    1. Your internet status(if you are not connected, you cannot download)
    2. Repository server might be broken(it is completely not your fault....)

# 0. Ubuntu – About Package Installation

---

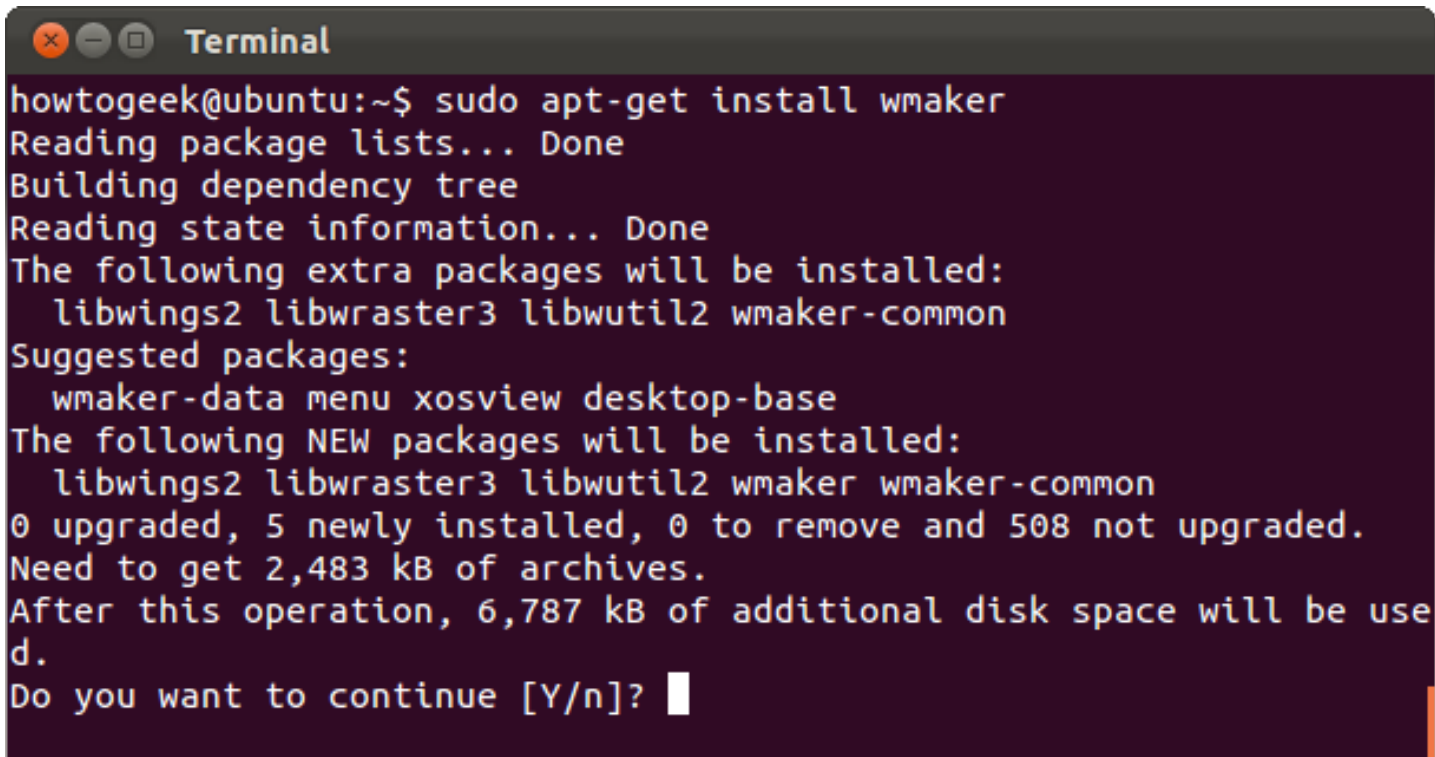
- There are many repositories(they are also called mirror site)
  - Why there are many repositories??
  - Ubuntu distributors update their own repository(it is the starting point of update because they are provider!)
    - ➔ If you are geographically close to the Ubuntu distributors server, then you can update and download packages rapidly.
    - ➔ However, in most of users are not close the their server.
    - ➔ To resolve it, many mirror site(copy&pasted server) are managed.
- For example,
- kr.archive.ubuntu.com by KAIST SPARCS
- If this mirror site is in trouble and your ubuntu is set to be connected to this mirror site, then apt-get install will not work.
    - ➔ change repository(or mirror site)

You already saw this in KakaoTalk Group Chat....

# 0. Ubuntu – About Package Installation

---

- Open-source based OS → Applications are free!
- Free applications are in repository.
- Command ‘apt-get’ access the repository and do something related with applications
  - apt-get install (package name)
  - apt-get remove (package name)
  - apt-get update : repository update
  - apt-get upgrade : upgrade packages
  - ... (there are more)

A terminal window titled "Terminal" with standard Ubuntu window controls (close, minimize, maximize). The terminal shows the command `sudo apt-get install wmaker` being executed. The output displays the process of reading package lists, building a dependency tree, and listing the packages to be installed. It shows that several extra packages will be installed along with the requested one, and provides information about disk space requirements. The prompt "Do you want to continue [Y/n]?" is visible at the bottom with a cursor.

```
howtogeek@ubuntu:~$ sudo apt-get install wmaker
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libwings2 libwraster3 libwutil2 wmaker-common
Suggested packages:
  wmaker-data menu xosview desktop-base
The following NEW packages will be installed:
  libwings2 libwraster3 libwutil2 wmaker wmaker-common
0 upgraded, 5 newly installed, 0 to remove and 508 not upgraded.
Need to get 2,483 kB of archives.
After this operation, 6,787 kB of additional disk space will be used.
Do you want to continue [Y/n]? █
```

# 0. Ubuntu – About Package Installation

---

- Be careful when execute `sudo apt-get upgrade`
- `cd /usr/lib & ls`
- You can see libraries with version number.
- `sudo apt-get upgrade` those libraries to the latest version
  - Q: The latest, isn't it good?**
  - A: not actually...**
- When you make a program with a library, your code & library files will be linked. In addition, executable file will find the libraries used in linking process.

Let's say there is a program with `opencv2.4`. The program has no problem in execution.

→ someday, you entered '`sudo apt-get upgrade`'

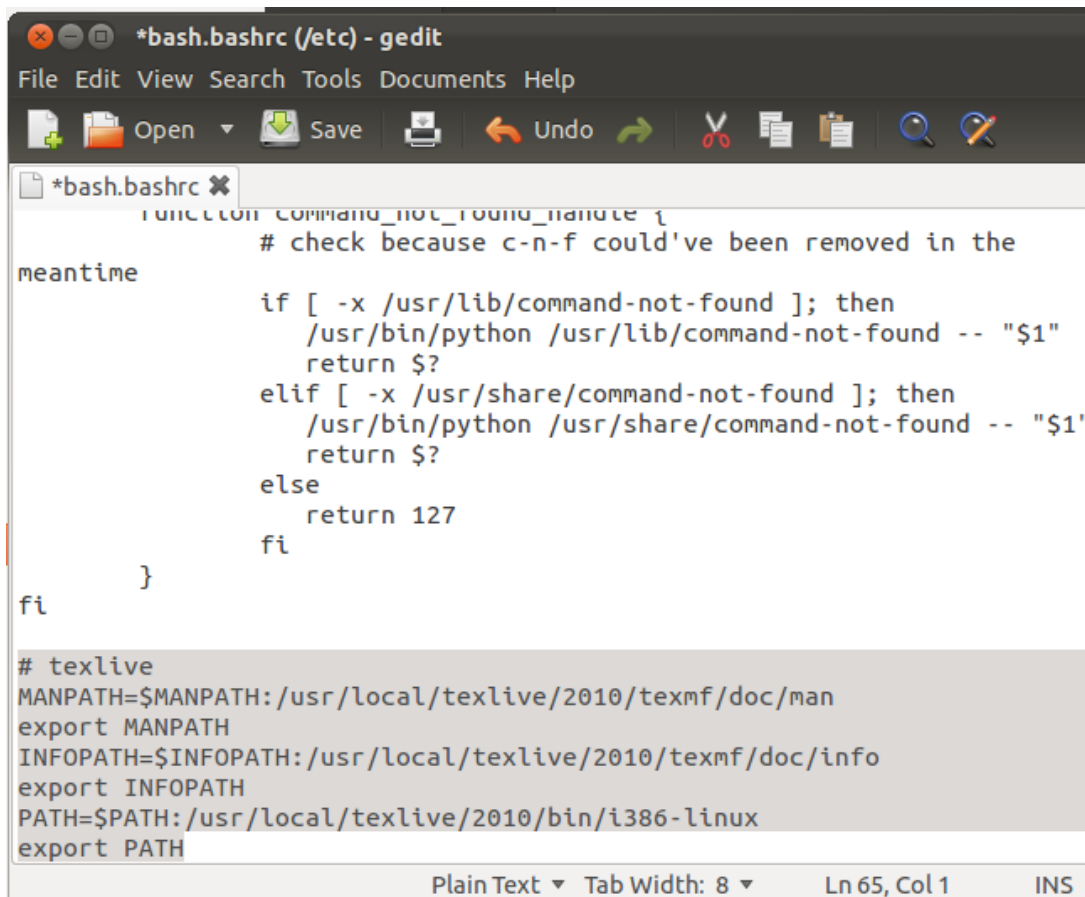
→ `opencv` is upgraded to version 2.5

→ you run the program again, but it will say like that,

**Error: No such files or libraries “libopencv2.4”**

# 0. Ubuntu – About bashrc

- a script that is executed whenever a new terminal session is started
- The file is in your home folder
- You can't find the '.bashrc' file by inputting the command 'ls' because it is hidden
- Do it yourself) gedit .bashrc or nano .bashrc
- .(filename) means the file is in hidden mode



```
function command_not_found_handle {
    # check because c-n-f could've been removed in the
    meantime
    if [ -x /usr/lib/command-not-found ]; then
        /usr/bin/python /usr/lib/command-not-found -- "$1"
        return $?
    elif [ -x /usr/share/command-not-found ]; then
        /usr/bin/python /usr/share/command-not-found -- "$1"
        return $?
    else
        return 127
    fi
}

# texlive
MANPATH=$MANPATH:/usr/local/texlive/2010/texmf/doc/man
export MANPATH
INFOPATH=$INFOPATH:/usr/local/texlive/2010/texmf/doc/info
export INFOPATH
PATH=$PATH:/usr/local/texlive/2010/bin/i386-linux
export PATH
```

bashrc is frequently used  
to set environment variables.  
(very important!!!!!!)

# 0. Ubuntu – About ssh(Secure shell)

---

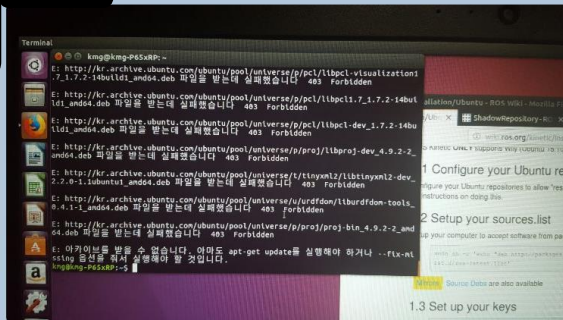
- Tele-communication
- It must be convenient for accessing NUC PC if you can be familiar with this.
- `ssh username@hostname`

ex) `ssh seojh@hostname` → no gui supported.

`ssh -X seojh@hostname` → gui supported. You can use qtcreator, gedit, and so on.

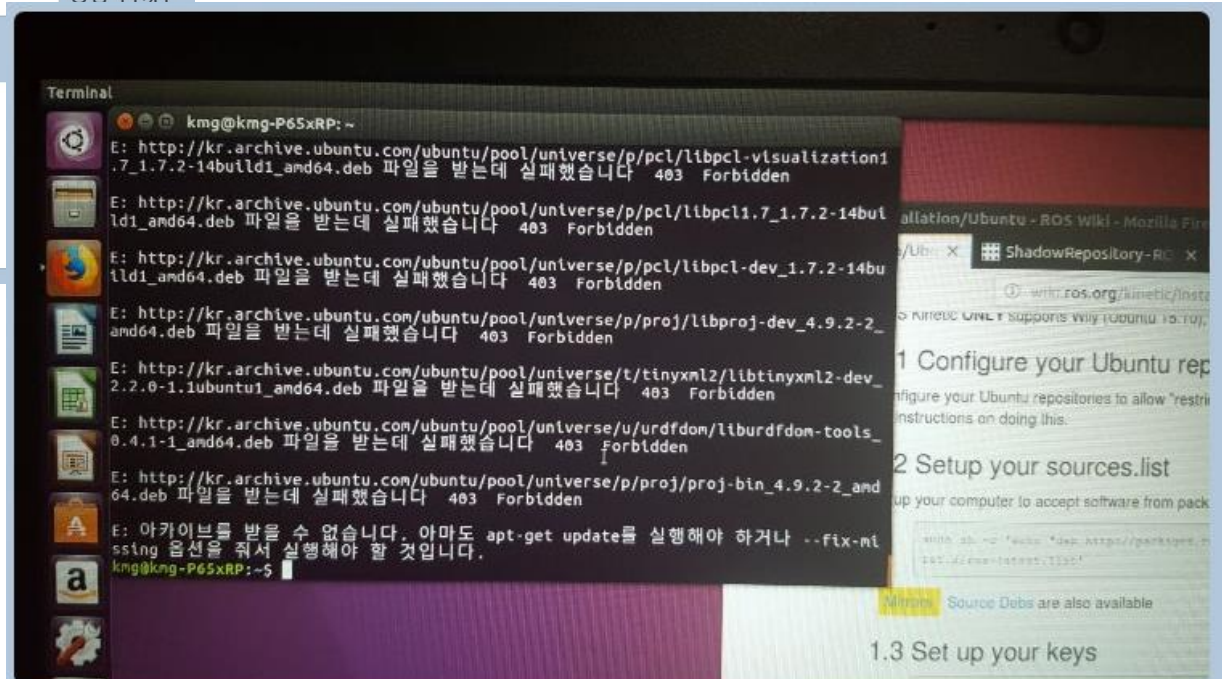
# 0. Revisit Questions...

- Guess what is the problem & solution.



ros 설치 과정에서 이런 오류가 뜨는데 어떻게 해결해야 하나요??

sudo apt-get install ros-kinetic-desktop-full 명령어 입력하고 나서 설치 진행하고 있었는데 76% 이후 과정이 진행되지 않습니다

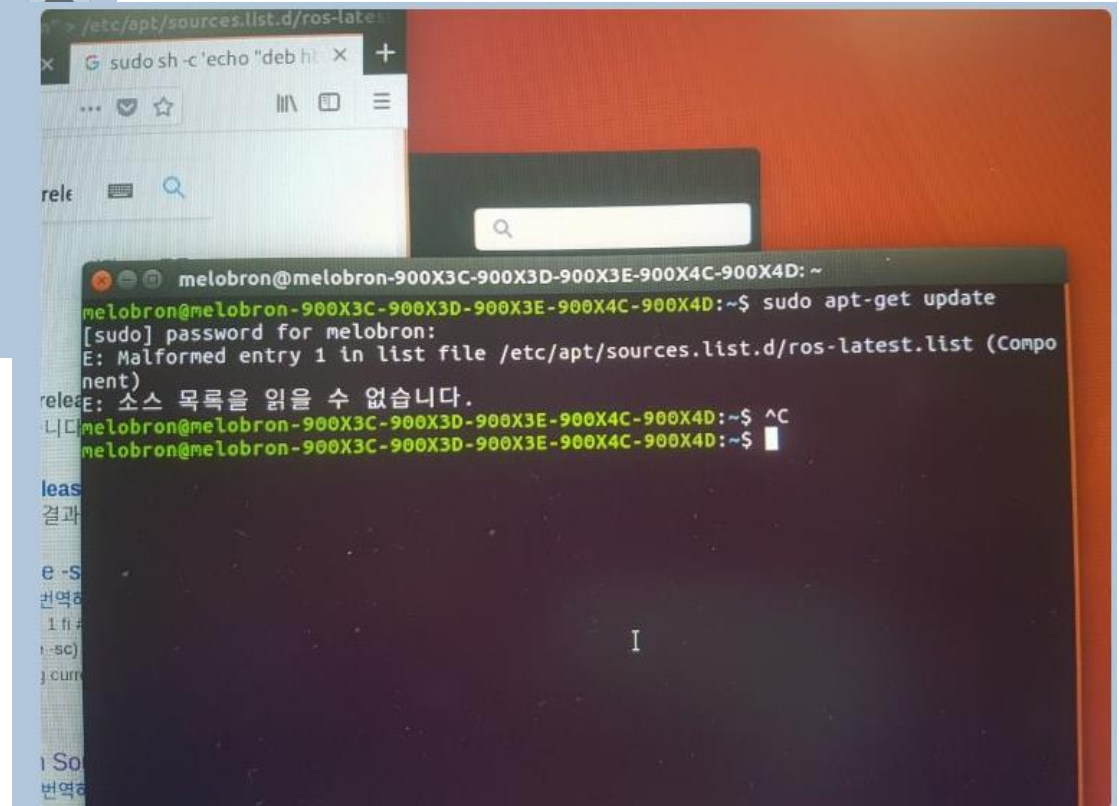


# 0. Revisit Questions...

- Guess what is the problem & solution.

Ros 설치중에 이렇게 나오네요  
무슨 오류인가요?

오후 2:48



A terminal window on a Linux system showing the output of the command `sudo apt-get update`. The prompt is `melobron@melobron-900X3C-900X3D-900X3E-900X4C-900X4D: ~`. The output shows the password prompt, followed by the error message: `E: Malformed entry 1 in list file /etc/apt/sources.list.d/ros-latest.list (Component)`. Below this, there is a Korean message: `E: 소스 목록을 읽을 수 없습니다.` (Error: cannot read the source list). The user then presses `^C` to abort the command.

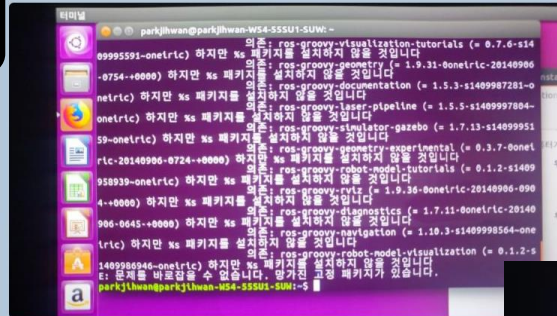
```
melobron@melobron-900X3C-900X3D-900X3E-900X4C-900X4D: ~  
melobron@melobron-900X3C-900X3D-900X3E-900X4C-900X4D:~$ sudo apt-get update  
[sudo] password for melobron:  
E: Malformed entry 1 in list file /etc/apt/sources.list.d/ros-latest.list (Component)  
E: 소스 목록을 읽을 수 없습니다.  
melobron@melobron-900X3C-900X3D-900X3E-900X4C-900X4D:~$ ^C  
melobron@melobron-900X3C-900X3D-900X3E-900X4C-900X4D:~$
```

- Hint. Read a error message carefully...



# 0. Revisit Questions...

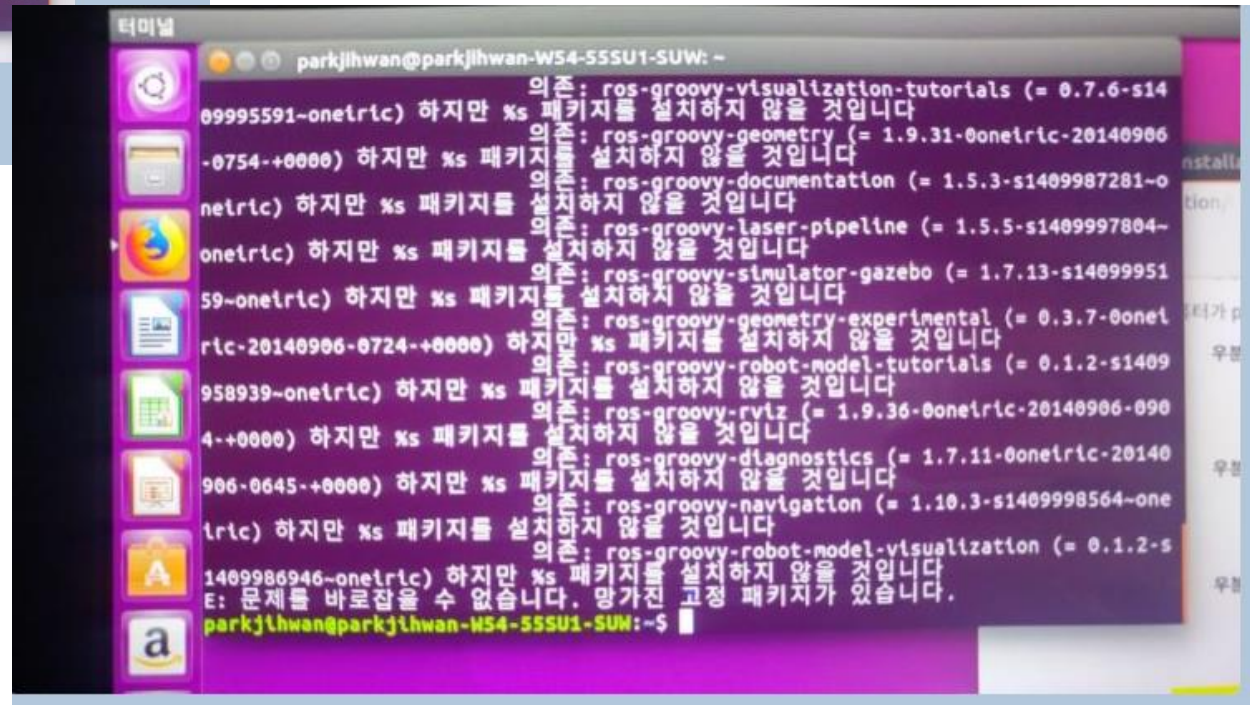
- Guess what is the problem & solution.



```
parkjihwan@parkjihwan-W54-SSSU1-SUW: -
09995591-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
-0754-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
netric) 하지만 %s 패키지를 설치하지 않을 것입니다
oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
59-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
ric-20140906-0724-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
958939-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
4-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
906-0645-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
lric) 하지만 %s 패키지를 설치하지 않을 것입니다
1409986946-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
E: 문제를 바로잡을 수 없습니다. 망가진 고정 패키지가 있습니다.
parkjihwan@parkjihwan-W54-SSSU1-SUW: -
```

이렇게 나오네요

오전 2:06

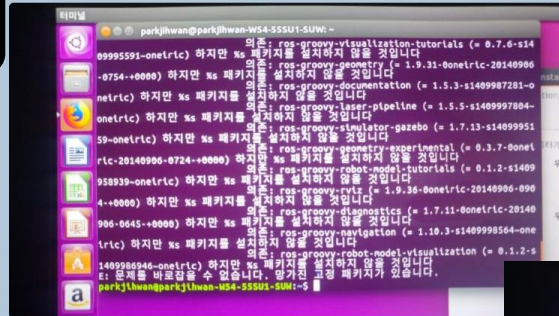


```
parkjihwan@parkjihwan-W54-SSSU1-SUW: -
의존: ros-groovy-visualization-tutorials (= 0.7.6-s14
09995591-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
의존: ros-groovy-geometry (= 1.9.31-0oneiric-20140906
-0754-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
의존: ros-groovy-documentation (= 1.5.3-s1409987281-0
netric) 하지만 %s 패키지를 설치하지 않을 것입니다
의존: ros-groovy-laser-pipeline (= 1.5.5-s1409997804-
oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
의존: ros-groovy-simulator-gazebo (= 1.7.13-s14099951
59-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
의존: ros-groovy-geometry-experimental (= 0.3.7-0onei
ric-20140906-0724-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
의존: ros-groovy-robot-model-tutorials (= 0.1.2-s1409
958939-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
의존: ros-groovy-rviz (= 1.9.36-0oneiric-20140906-090
4-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
의존: ros-groovy-diagnostics (= 1.7.11-0oneiric-20140
906-0645-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
의존: ros-groovy-navigation (= 1.10.3-s1409998564-one
lric) 하지만 %s 패키지를 설치하지 않을 것입니다
의존: ros-groovy-robot-model-visualization (= 0.1.2-s
1409986946-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
E: 문제를 바로잡을 수 없습니다. 망가진 고정 패키지가 있습니다.
parkjihwan@parkjihwan-W54-SSSU1-SUW: -
```

- Hint. Why this command handles old-version ros?  
(maybe, the link of ROS packages is indicating old URL)

# 0. Revisit Questions...

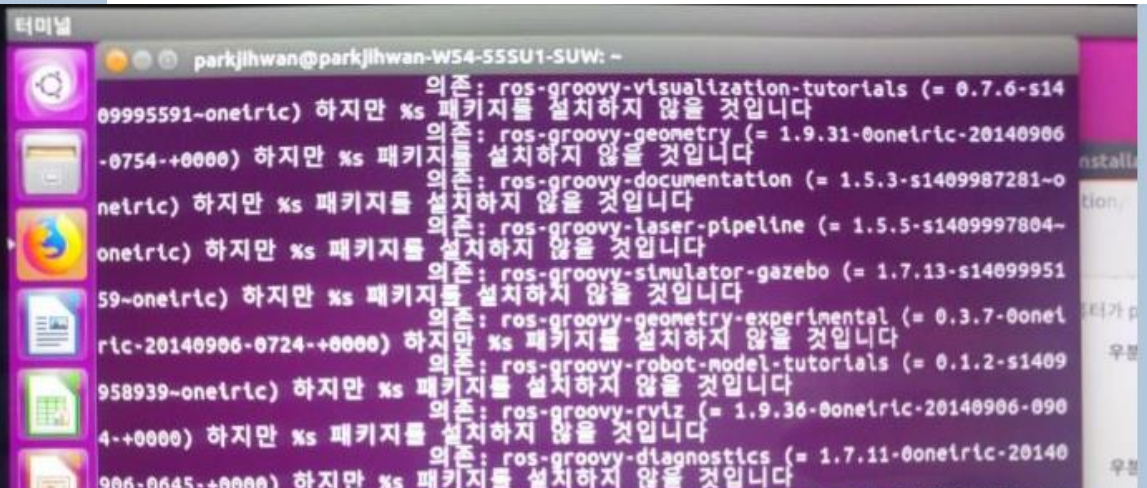
- Guess what is the problem & solution.



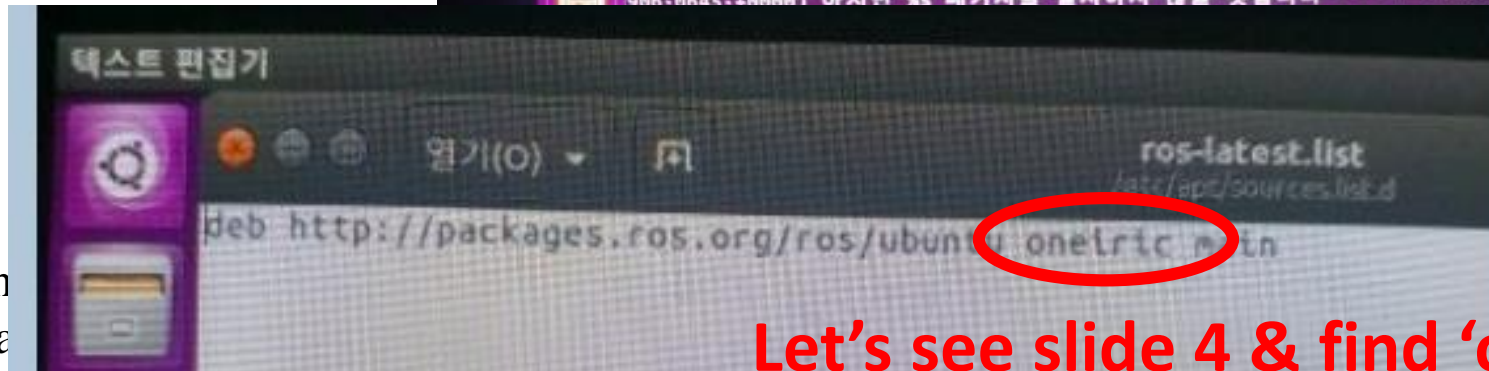
```
parkjihwan@parkjihwan-W54-SSSU1-SUW: ~$ sudo apt-get install ros-groovy-visualization-tutorials
09995591-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
-0754-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
netric) 하지만 %s 패키지를 설치하지 않을 것입니다
oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
59-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
ric-20140906-0724-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
958939-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
4-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
906-0645-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
rtic) 하지만 %s 패키지를 설치하지 않을 것입니다
140928046-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
c: 토치를 바로잡을 수 없습니다. 알려진 응용 패키지가 있습니다.
parkjihwan@parkjihwan-W54-SSSU1-SUW: ~$
```

이렇게 나오네요

오전 2:06



```
parkjihwan@parkjihwan-W54-SSSU1-SUW: ~$ sudo apt-get install ros-groovy-visualization-tutorials
09995591-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
-0754-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
netric) 하지만 %s 패키지를 설치하지 않을 것입니다
oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
59-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
ric-20140906-0724-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
958939-oneiric) 하지만 %s 패키지를 설치하지 않을 것입니다
4-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
906-0645-+0000) 하지만 %s 패키지를 설치하지 않을 것입니다
```



```
deb http://packages.ros.org/ros/ubuntu/oneiric main
```

- Hint  
(ma

Let's see slide 4 & find 'oneiric'

## 0. Revisit Questions...

---

- You will do these things again and again and again and again and again.....
- Think about the reason why we share questions and answer...  
(Let's follow a question rule from now on: Q.~~~)
- I hope that students' questions are answered by students at the end of semester(or at the end of this year...)
- You can help each others.  
Also, you can be a TA next year...

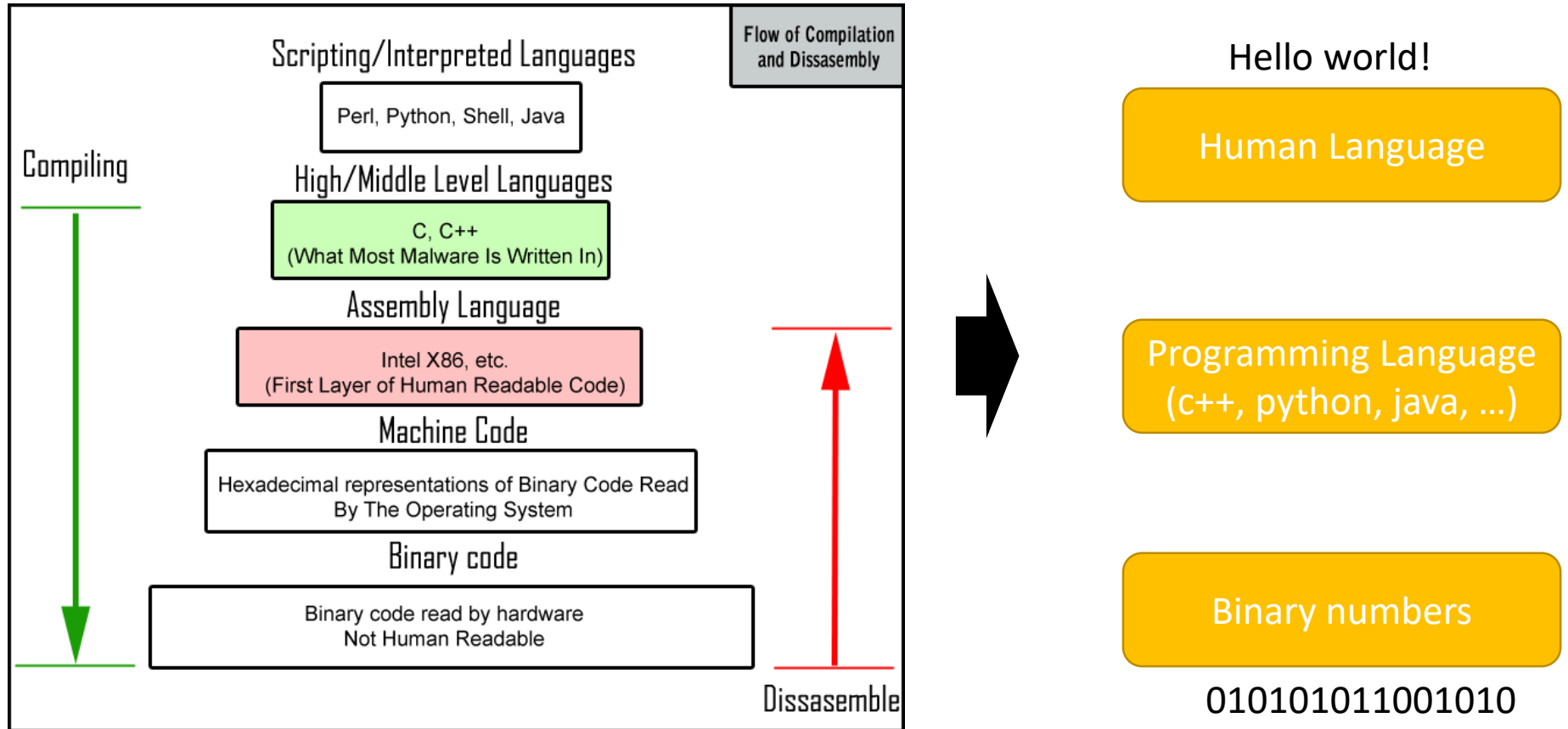
## 0. Revisit Questions...

---

- The more you try and fail, the deeper you will understand.
- Just try something. The worst case that can happen to you is just re-installation of Ubuntu(not the end of the world or F grade for the course)
- Regardless of how much you messed up ubuntu in your laptop, TAs will help you and resolve it clear.

# Additional Topic. Concepts of compiler, g++, makefile & cmake

\*\*This material is for students who are not familiar with c++



Compiling: Process of converting programming language to machine-understandable language(binary numbers)

# Additional Topic. Concepts of compiler, g++, makefile & cmake

## 컴파일러(Compiler)

컴파일러는 원시 언어로 된 프로그램을 읽어들어서 목적 언어로 된 동일한 프로그램을

출력하여 주는 언어처리기입니다. 기계어로 번역이 쉽게 이루어질 수 있으면서

실행 시간을 증시하는 경우에 사용합니다.

## 인터프리터(Interpreter)

인터프리터는 원시 언어로 작성된 명령문들을 한 번에 한 줄씩 번역하고 실행하는 프로그램입니다.

프로그램이 짧고 복잡하지 않고, 대화형 프로그램에서 많이 사용됩니다.

컴파일러는 번역, 인터프리터는 통역이라고 이해하면 쉽습니다.

	컴파일러	인터프리터
실행	프로그램 전체를 번역	프로그램 한 줄씩 번역
메모리 할당	목적 프로그램 생성으로 메모리 사용	메모리 사용하지 않음
번역 속도	번역 속도 느림	번역 속도 빠름
실행 속도	실행 속도 빠름	실행 속도 느림
프로그램 저장 위치	디스크에 저장됨	ROM에 저장됨

**Which one is better language for your mobile platform?**



# Additional Topic. Concepts of compiler, g++, makefile & cmake

---

In the early stage of compiling,

```
#include <stdio.h>

int main(int argc, char** argv){
    printf("abcd");
    return 0;
}
```

Execution

However, programs become huge & write a code in a single file becomes inefficient. Programmers started to divided them into pieces. (You leant how to create object in python!)

main.cpp  
mainwindow.cpp  
webcam.cpp  
rplidar.cpp

main.h  
mainwindow.h  
webcam.h  
rplidar.h



**compiling**

main.o  
mainwindow.o  
webcam.o  
rplidar.o

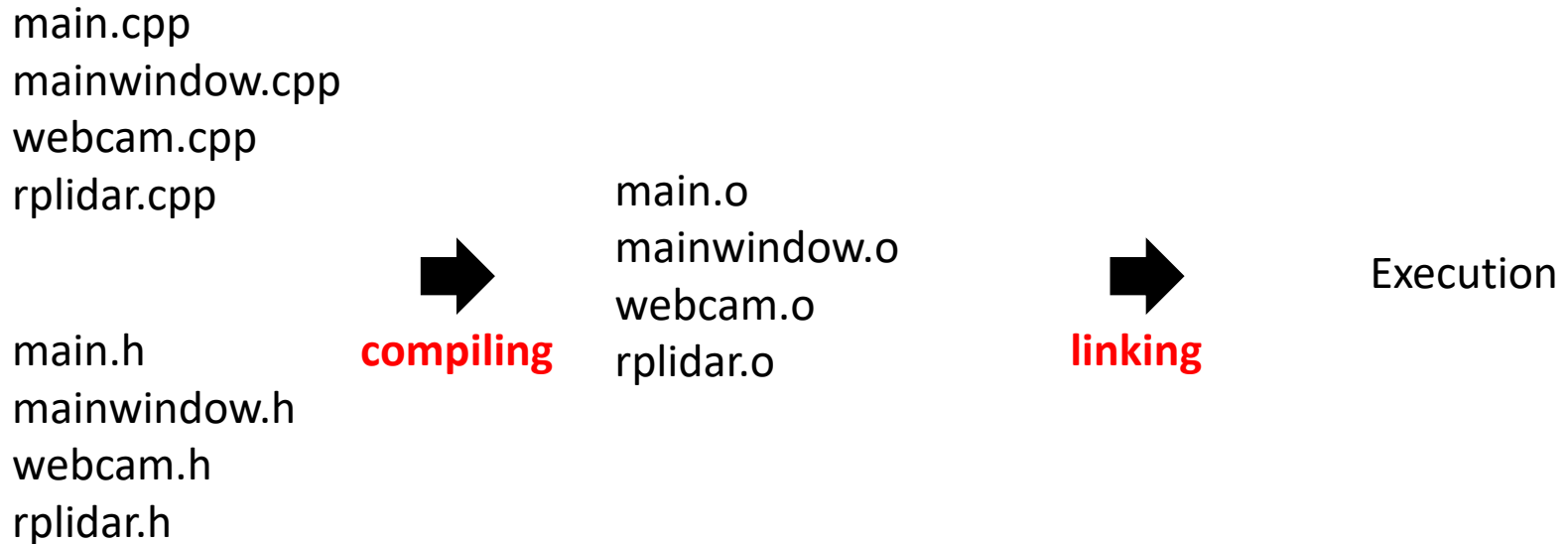


**linking**

Execution

## Additional Topic. Concepts of compiler, g++, makefile & cmake

---



**If the errors are appeared in compiling process  
→ you should check your source file(.cpp&.h files)**

**If the errors are appeared in linking process  
→ you should check the options of your packages  
(mostly, checking CMakeLists.txt file will be enough)**

**Let's see some examples...**



# Additional Topic. Concepts of compiler, g++, makefile & cmake

---

-gcc: GNU Compiler Collection

\*[https://en.wikipedia.org/wiki/GNU\\_Compiler\\_Collection](https://en.wikipedia.org/wiki/GNU_Compiler_Collection)

-g++: gcc for c++

-just make a simple code & g++ main.cpp

-g++ gets lots of options

-if the program includes many sources files and libraries, then g++ command will be complex and long.

-to resolve it, Makefile is introduced...

-The problem of Makefile is that it is not compatible in other platform(other OS like windows)

-cmake(Cross-platform Make) supports compatible build process between various platforms.

\*get the example source codes by typing,  
git clone <https://github.com/kaistmecl/roscpp>  
read 'README' file and follow it

# Additional Topic. Concepts of compiler, g++, makefile & cmake

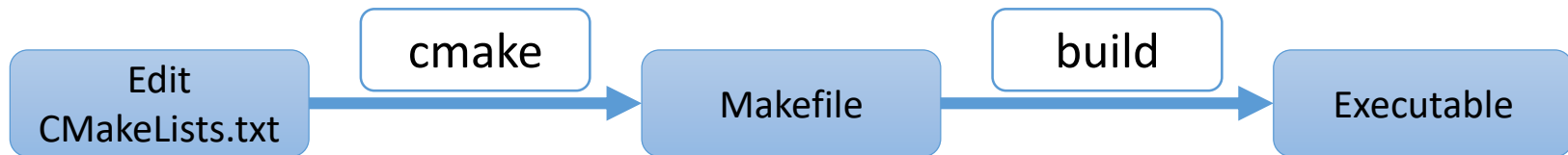
---

For ROS students,

when you create a packages, you will use a command 'catkin\_create\_pkg'. Then, your packages will be created with CMakeList.txt file.

For OpenCV students,

you have an option not to use 'cmake-based project'. However, your code should be combined with ROS based programs for the project. Therefore, I recommend you to utilize 'cmake-based project'. (you can do it easily by using qtcreator)



## Additional Topic. OpenCV installation

---

- revisit OpenCV installation process
- see the OpenCV lecture note by TA. Jisung Park

Break Time.

Any Questions?

# 1. ROS – Overview

---



- Robot Operation System
- Introduced by WillowGarage, USA (but they are vanished.....)
- Still maintaining

# 1. ROS – Install

---

- Google ‘ros install’ or visit [ros.org](http://ros.org)
- <http://wiki.ros.org/kinetic/Installation/Ubuntu>
- Installation can be done with terminal and commands only.

# 1. ROS – Install(Explanations of commands)

---

- `sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'`

우분투가 설치된 직후의 repository는 ros 설치파일들이 있는 경로를 가지고 있지 않습니다. (cd /etc/apt/에 들어가서 있는 파일들을 살펴보면 ros관련 경로들은 없는 걸 확인할 수 있음). ROS의 설치파일들이 있는 주소를 repository list에 넣어주는 명령입니다.

- `sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key 421C365BD9FF1F717815A3895523BAEEB01FA116`

ROS설치파일이 있는 repository link에 접근하기 위한 키를 설정하는 명령(어지간 하면 다시 쓸일없는 명령어)

- `sudo apt-get update`  
ROS의 설치파일이 있는 주소를 repository list를 담당하는 파일에는 포함시켰지만, 아직 그 주소를 반영하지는 않은 상태이므로, 이를 반영하기 위해 update진행.(명령어 실행 후, 나오는 메시지들을 잘 살펴보면 <http://packages.ros.org/ros/ubuntu>~~가 추가된걸 확인할 수 있음)

# 1. ROS – Install(Explanations of commands)

---

- `sudo apt-get install ros-kinetic-desktop-full`  
(이제 ROS의 위치를 알고있으니, 다른 프로그램들 설치하듯이 ros 또한 apt-get install 명령으로 설치)
- `sudo rosdep init && rosdep update`  
(ros를 설치하는 과정에서 설치된 여러 패키지들의 dependency를 자동으로 설정해주는 명령어. 이것도 한 번 실행하고 나면 다시 쓸 일이 거의 없음. Ros설치 과정에서 설치된 패키지들은 이 시점에서 실행하는걸로 dependency가 설정되고, 추후에 작성되는 패키지들은 CMakeLists.txt에서 dependency 처리를 담당함)
- `echo "source /opt/ros/kinetic/setup.bash" >> ~/.bashrc`  
(ROS관련 명령어를 실행하기 위해서는 ROS관련 패키지들의 위치를 터미널상에 설정해주어야 사용가능. 매번 터미널 열때마다 이를 반복할 수 없으니, bashrc에 추가해두고 편하게 쓰는 것)
- `source ~/.bashrc`  
(.bashrc를 수정하는것 자체는 터미널에 영향을 주지 않음. Bashrc를 수정한 후에 터미널을 새로 열거나, 아니면 위의 명령어를 이용하면 현재 터미널에서 .bashrc의 내용을 다시 불러올 수 있음)



# 1. ROS – Initialize workspace

---

- `mkdir ~/catkin_ws`  
(`catkin_ws` 라는 폴더 생성. 이 곳을 workspace로 활용함.)
- `cd catkin_ws` (`catkin_ws` 폴더로 이동)
- `mkdir src` (`src` 폴더 생성)
- `cd src` (or `cd ~/catkin_ws/src`)
- `catkin_init_workspace`  
(현재 경로인 `~/catkin_ws/src` 를 workspace로 설정)
- `cd ..` (or `~/catkin_ws`)
- `catkin_make`  
(일반적으로는 `ros package`를 작성후 빌드하는 명령어이나,  
지금은 `catkin_init_workspace`라는 명령어를 통해 생성된 `CMakeLists.txt`를 빌드하  
여 `ros`관련해서 필요한 `dependency`, `environment variables` 등을 설정하도록 해줌)
- `echo "source ~/catkin_ws/devel/setup.bash" >> ~/.bashrc`  
(`catkin_make` 후에 생성된 `setup.bash` 파일을 `.bashrc`파일에 추가함으로써 이후에 생  
성되는 터미널들이 workspace관련 설정을 활용할 수 있도록 함)
- `source ~/.bashrc`  
(이후에 생성될 터미널들은 설정이 되겠지만, 여태까지 사용해오던 터미널은 적용  
되어 있지 않으므로, 다시 `.bashrc`를 불러와주는 명령어)

# 1. ROS – Initialize workspace

---

- `mkdir ~/catkin_ws`  
(`catkin_ws` 라는 폴더 생성. 이 곳을 workspace로 활용함.)
- `cd catkin_ws` (`catkin_ws` 폴더로 이동)
- `mkdir src` (`src` 폴더 생성)
- `cd src` (or `cd ~/catkin_ws/src`)
- `catkin_init_workspace`  
(현재 경로인 `~/catkin_ws/src` 를 workspace로 설정)
- `cd ..` (or `~/catkin_ws`)
- `catkin_make`  
(일반적으로는 `ros package`를 작성후 빌드하는 명령어이나,  
지금은 `catkin_init_workspace`라는 명령어를 통해 생성된 `CMakeLists.txt`를 빌드하  
여 `ros`관련해서 필요한 `dependency`, `environment variables` 등을 설정하도록 해줌)
- `echo "source ~/catkin_ws/devel/setup.bash" >> ~/.bashrc`  
(`catkin_make` 후에 생성된 `setup.bash` 파일을 `.bashrc`파일에 추가함으로써 이후에 생  
성되는 터미널들이 workspace관련 설정을 활용할 수 있도록 함)
- `source ~/.bashrc`  
(이후에 생성될 터미널들은 설정이 되겠지만, 여태까지 사용해오던 터미널은 적용  
되어 있지 않으므로, 다시 `.bashrc`를 불러와주는 명령어)

# 1. ROS – revisit shell script

---

- [https://github.com/ROBOTIS-GIT/robotis\\_tools/blob/master/install\\_ros\\_kinetic.sh](https://github.com/ROBOTIS-GIT/robotis_tools/blob/master/install_ros_kinetic.sh)

# 1. ROS –Tutorial

---

- Run 'roscore'
- Run 'roslaunch rospy\_tutorials listener'
- Run 'roslaunch rospy\_tutorials talker'
- Run 'roslaunch rqt\_graph rqt\_graph'
- rostopic echo /talker

# 1. ROS –Tutorial(explanation)

---



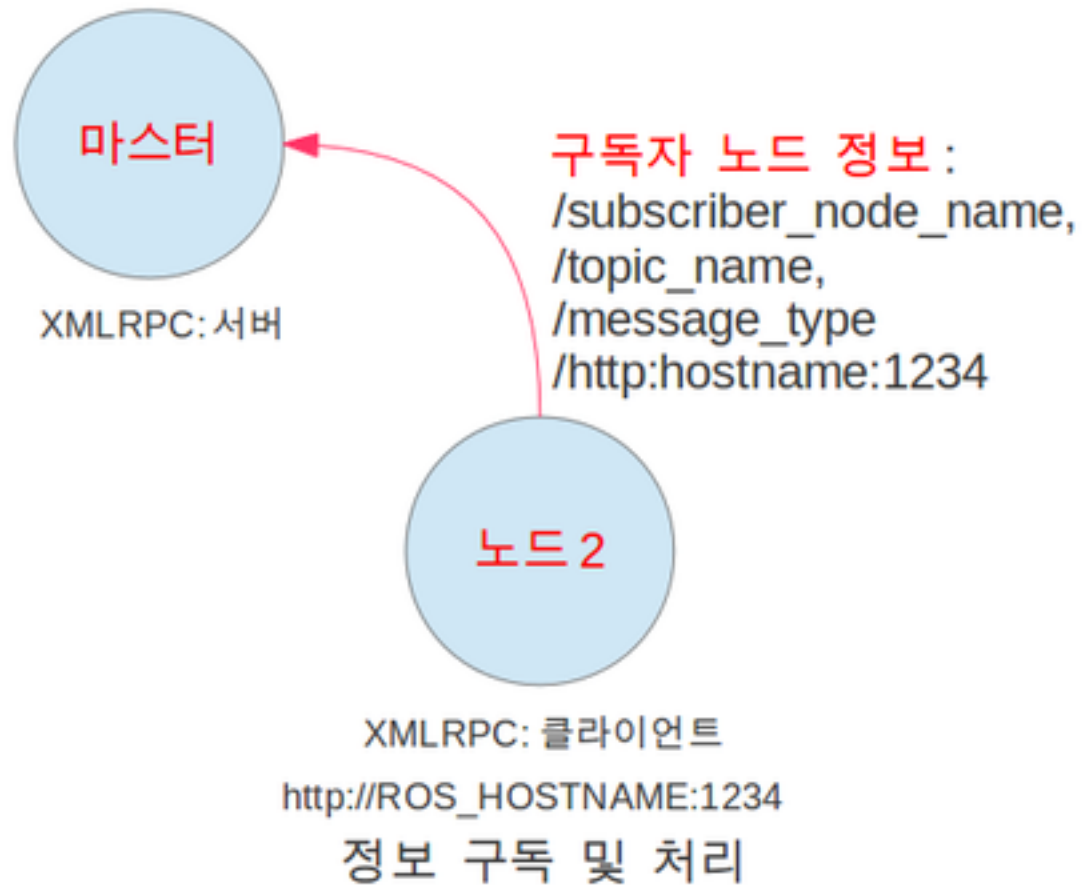
XMLRPC: 서버

`http://ROS_MASTER_URI:11311`

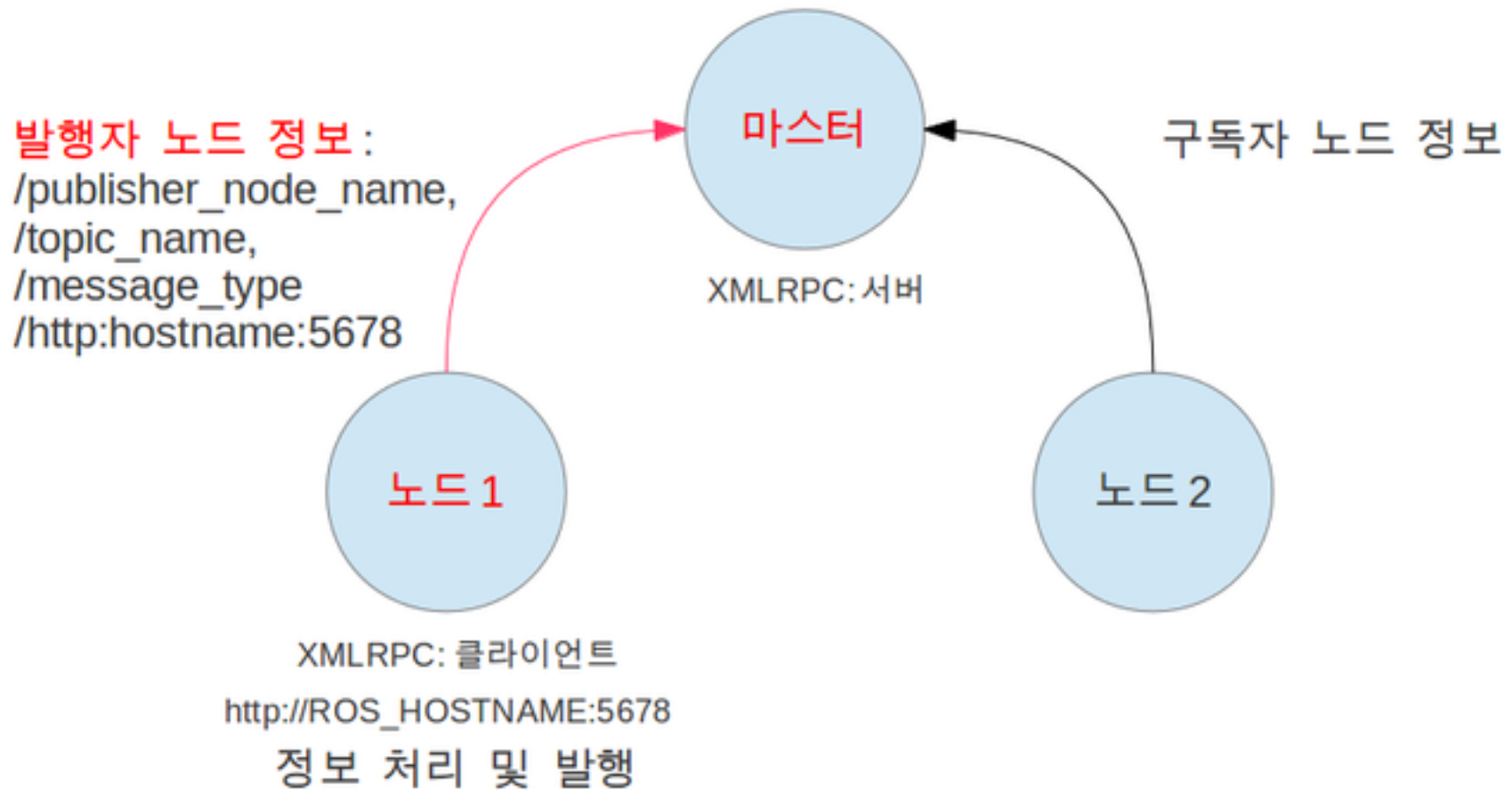
노드 정보 관리

# 1. ROS –Tutorial(explanation)

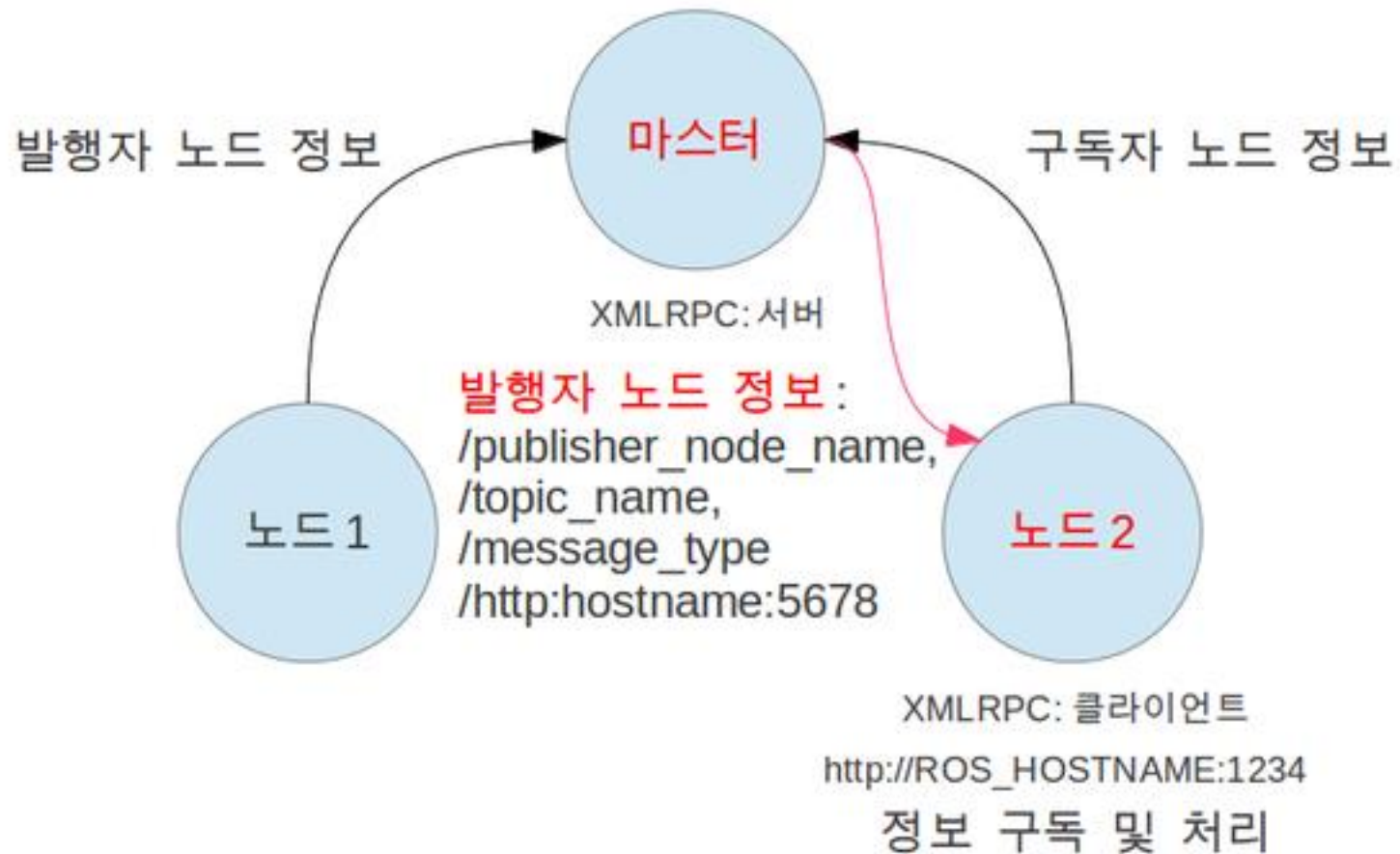
---



# 1. ROS –Tutorial(explanation)



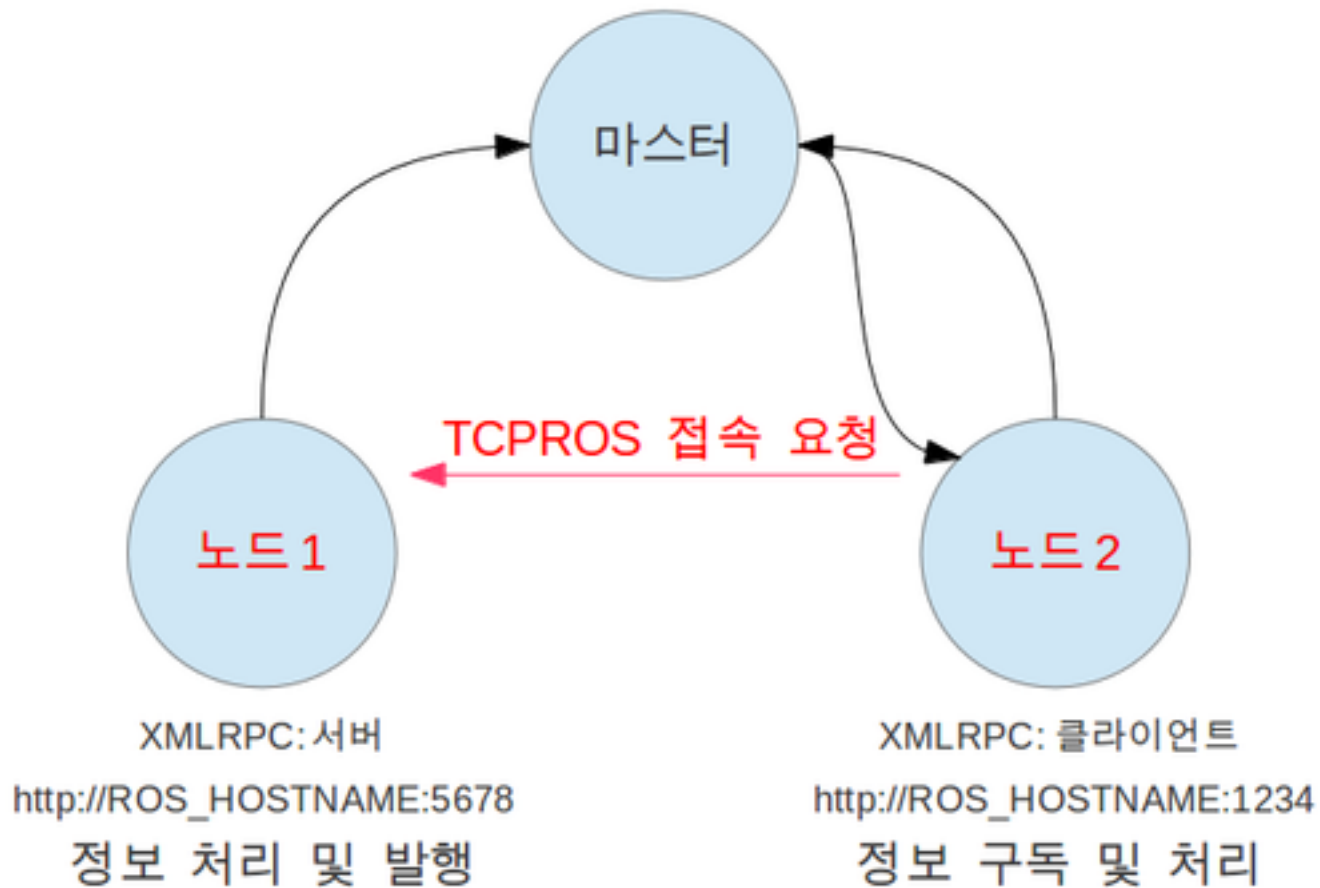
# 1. ROS –Tutorial(explanation)





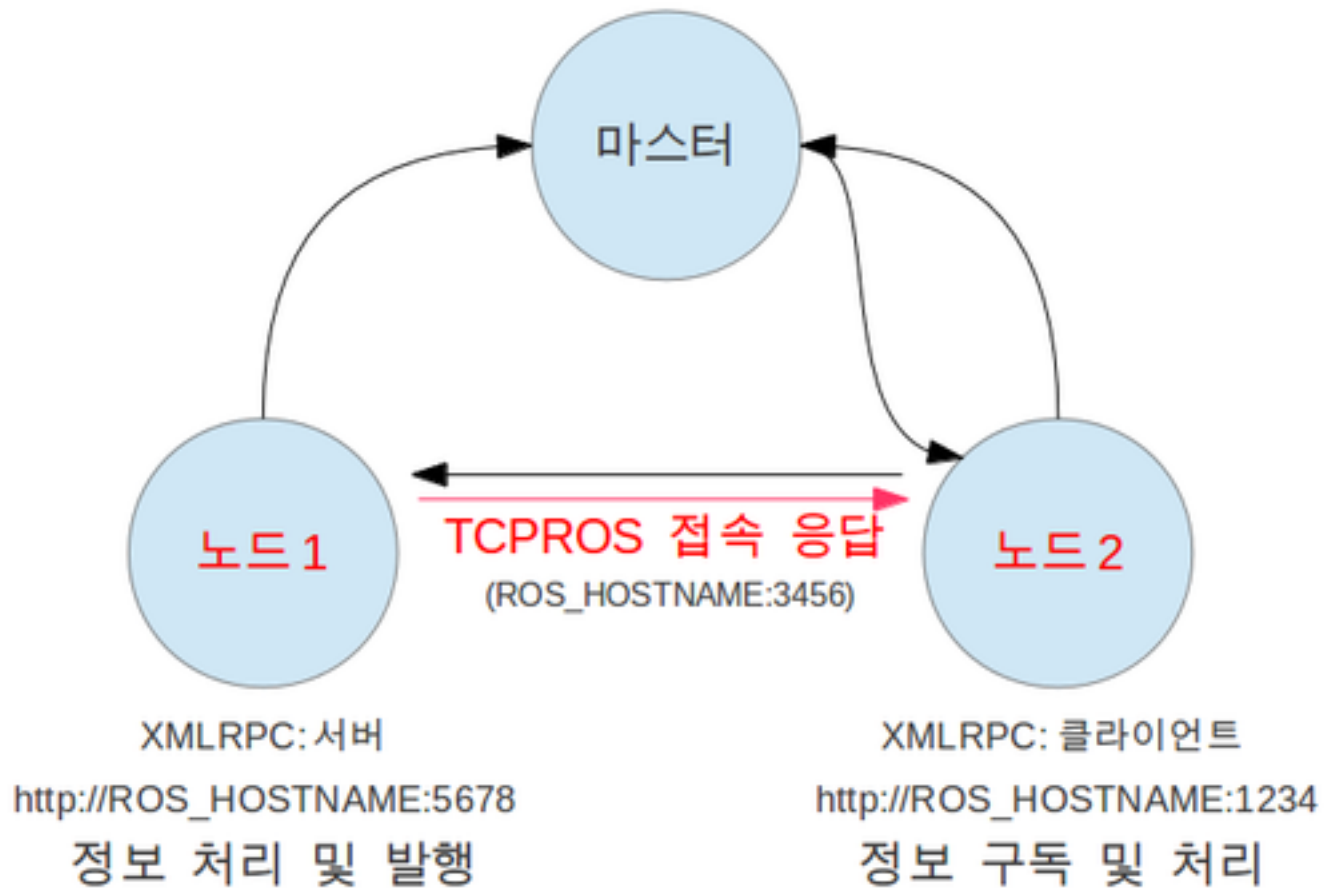
# 1. ROS –Tutorial(explanation)

---

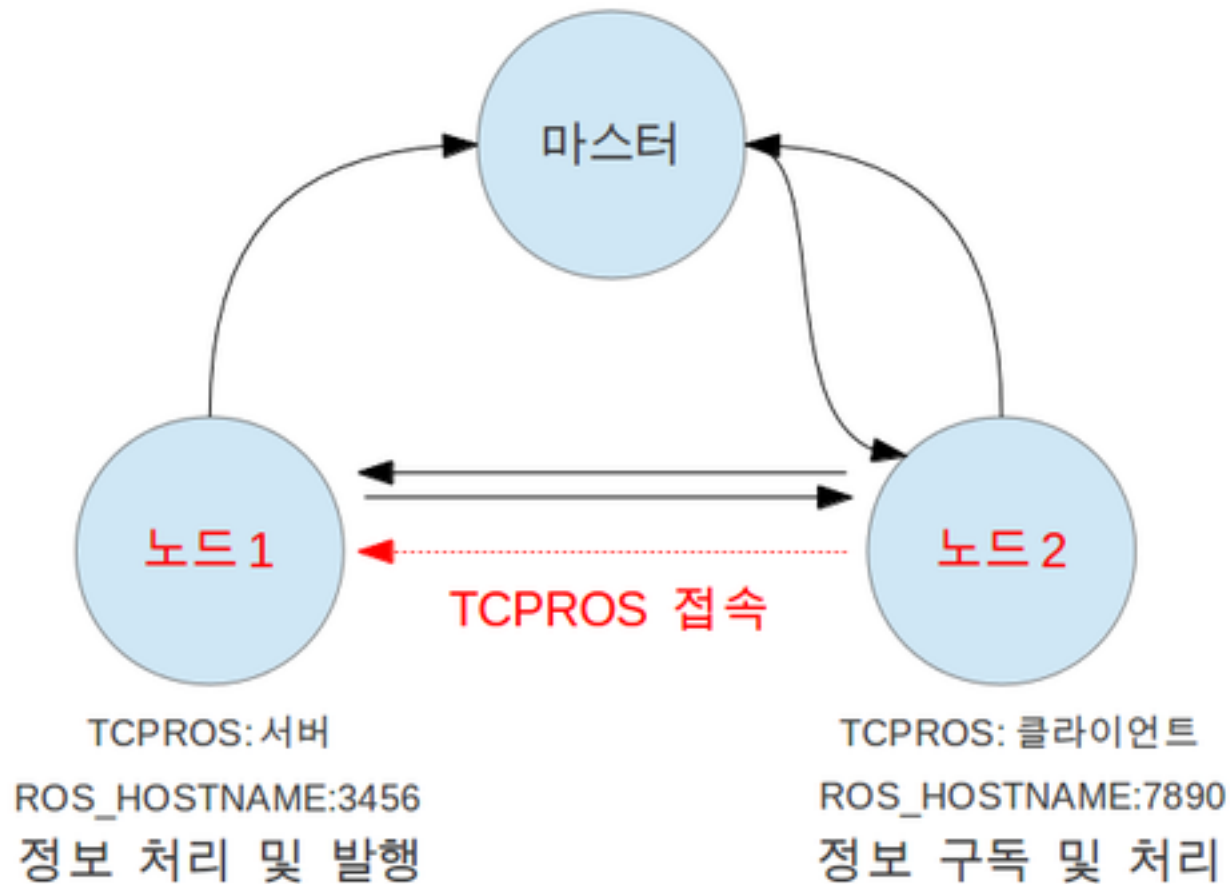


# 1. ROS –Tutorial(explanation)

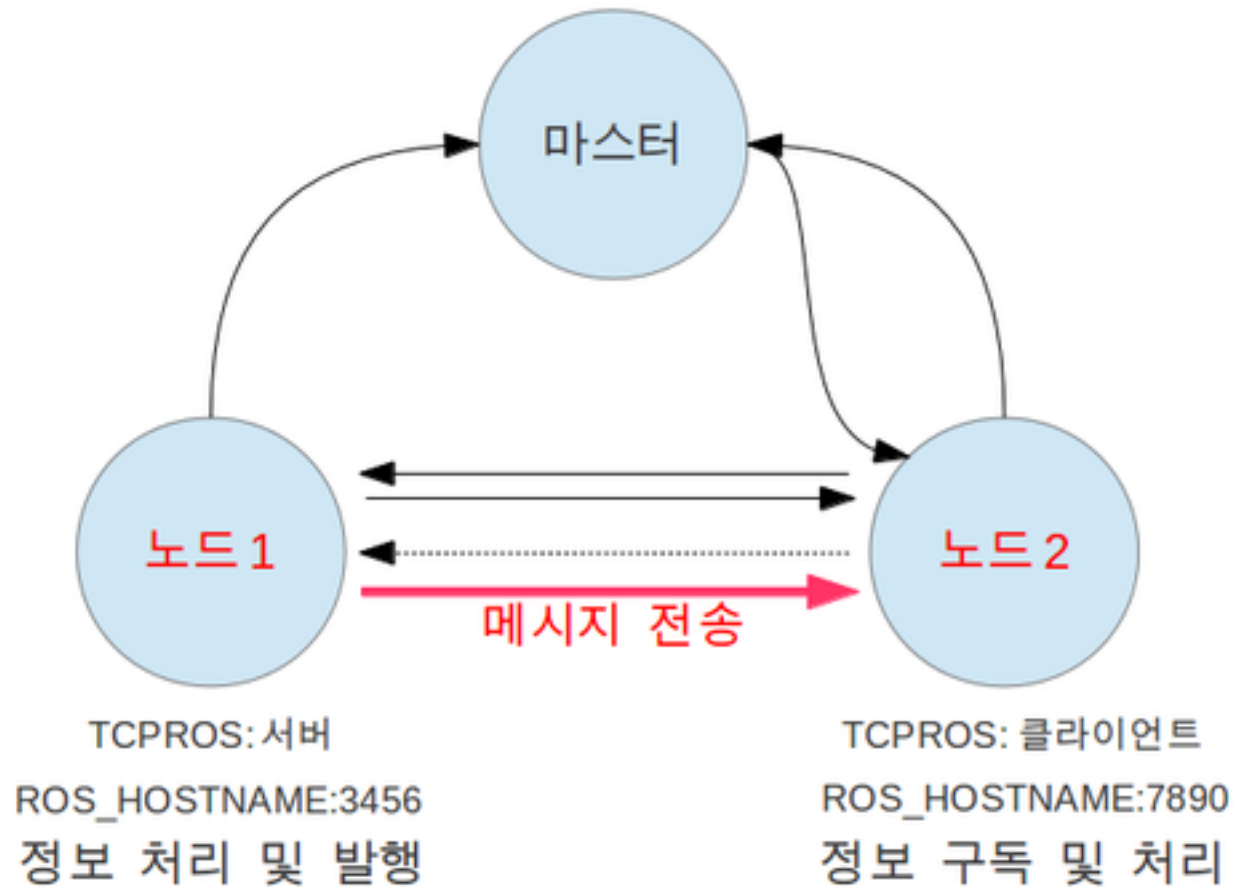
---



# 1. ROS –Tutorial(explanation)

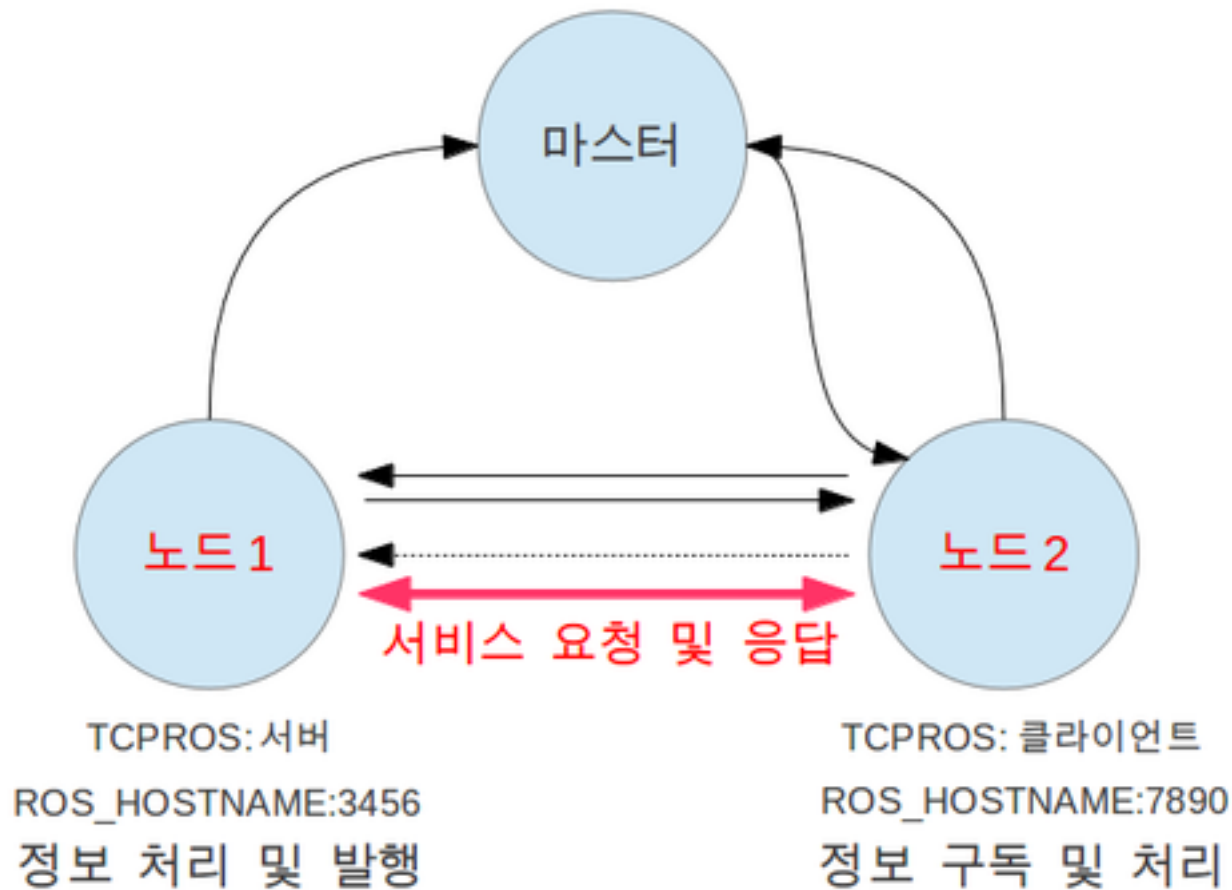


# 1. ROS –Tutorial(explanation)



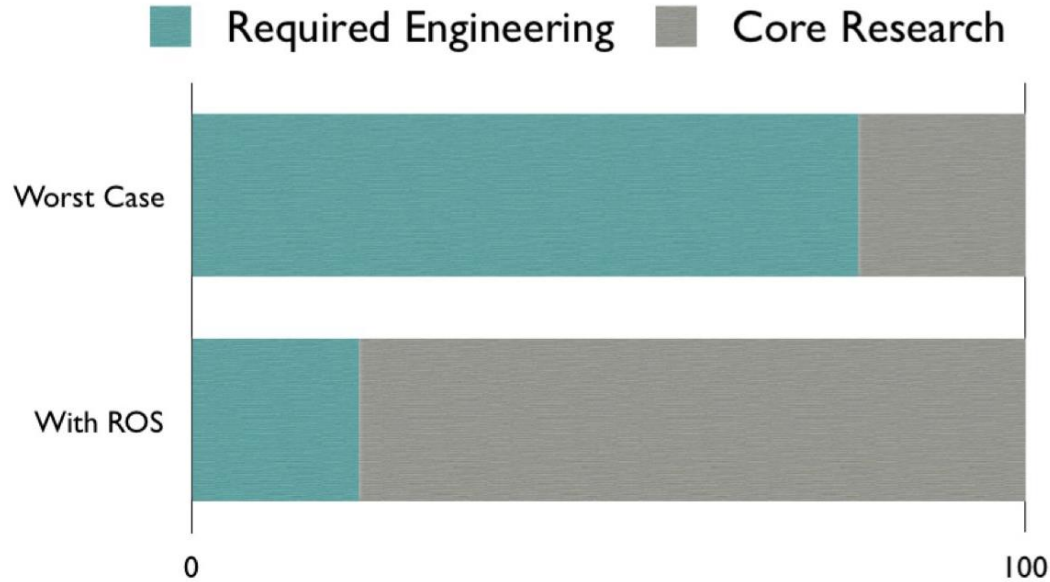
Message: asynchronous transfer mode(비동기 전송)

# 1. ROS –Tutorial(explanation)



Service : synchronous transfer mode(동기 전송)

# 1. ROS –Why ROS?

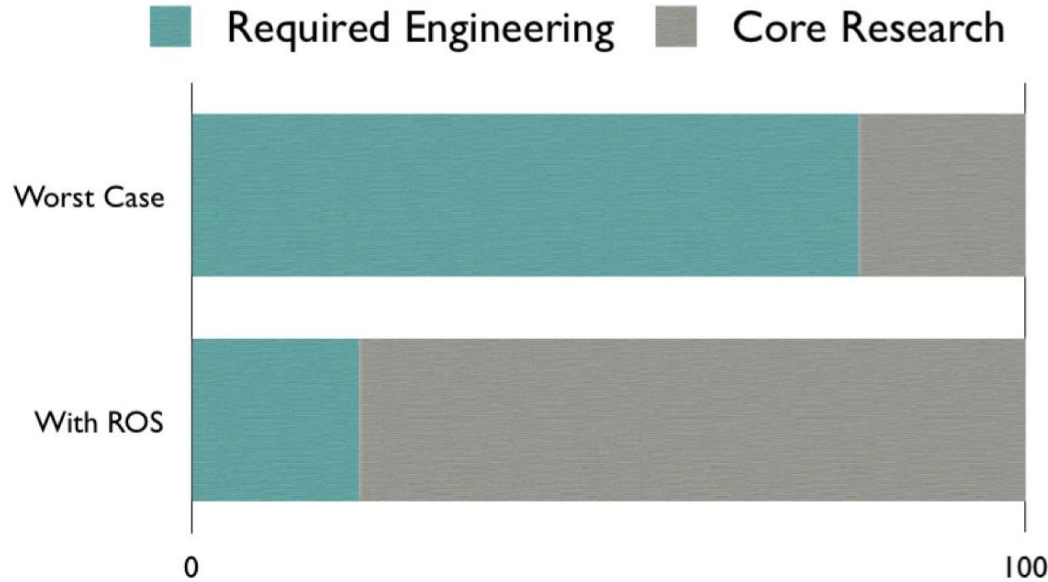


- Now, you might understand what this figure means...
- Single function for single nodes → Modularization
- Ex) Once program a node for acquiring Camera data, then this node can be used for various applications

Without ROS

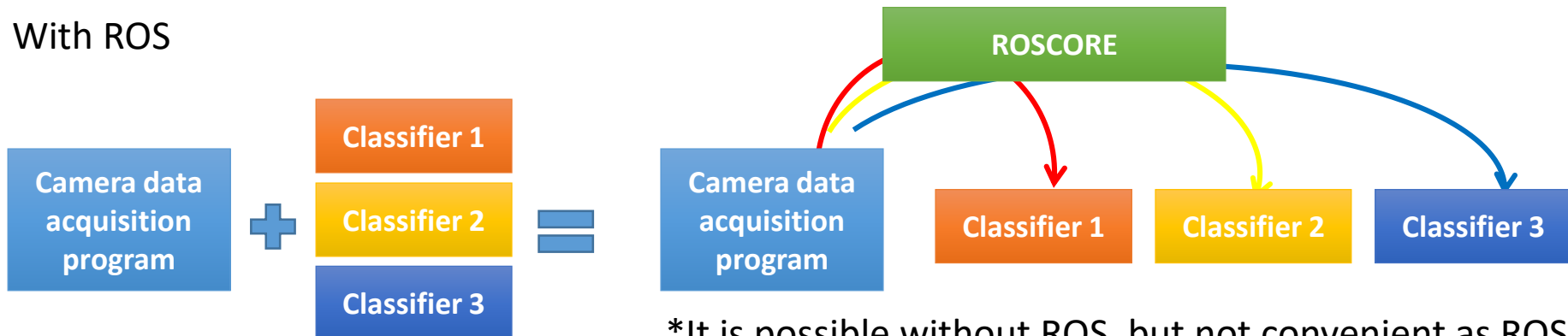


# 1. ROS –Why ROS?



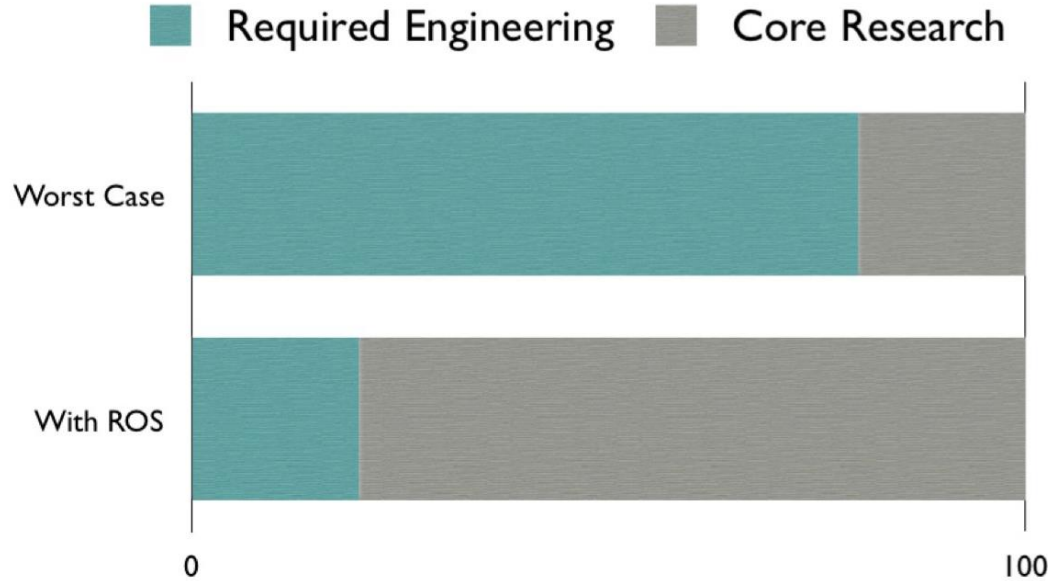
- Now, you might understand what this figure means...
- Single function for single nodes → Modularization
- Ex) Program a node for acquiring Camera data, then this node can be used for various applications

With ROS

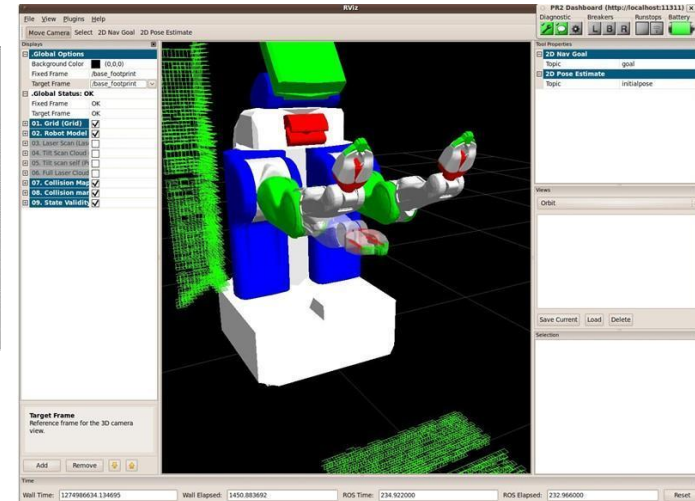
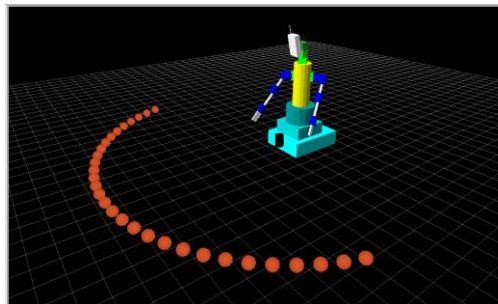
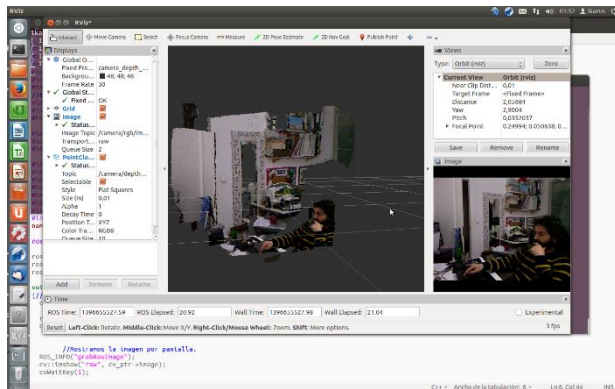


\*It is possible without ROS, but not convenient as ROS

# 1. ROS –Why ROS?



- Also, various libraries are provided.
- Ex) pointcloud library, opencv, etc ...
- In addition, data visualization, data management, simulation tools are provided





# 1. ROS –Useful commands & tools

---

- Useful ROS commands
  - roscore, rosrunc, roslaunch
  - roscd, rosls
  - rostopic, rosnode, rosservice, rosparam, rosmmsg, rossrv, rosbag
- Useful ROS tools
  - rqt\_plot
  - rqt\_graph
  - rqt\_bag
- Study by yourself

## 2. ROS – Create Nodes

---

1. Initialize your workspace (after initialization, you don't have to do this again)

```
-mkdir catkin_ws  
-cd catkin_ws  
-mkdir src  
-cd src  
-catkin_init_workspace  
-cd .. (or cd ~/catkin_ws)  
-catkin_make
```

2. Create a ros package

```
-cd ~/catkin_ws/src  
-catkin_create_pkg (pkg_name) (dependencies)  
for example, 'catkin_create_pkg ros_tutorial roscpp std_msgs'
```

3. Check the contents in the folder of the ros package created

## 2. ROS – Create Nodes

---

4. Make a source file in /src folder
  - gedit main.cpp
  - (follow step by step)
5. Modify CMakeLists.txt
6. Go to catkin\_ws and build it!
  - cd ~/catkin\_ws
  - catkin\_make
7. Update the list of ros packages in your computer
  - rospack profile
8. Execute
  - roslaunch (pkgname) (nodename)

ROS Session Finished.  
Any Questions?

Finished