

로봇 운영체제 ROS

ROBOTIS

Open Source Team

Yoonseok Pyo



[온라인강좌](#)

You Tube

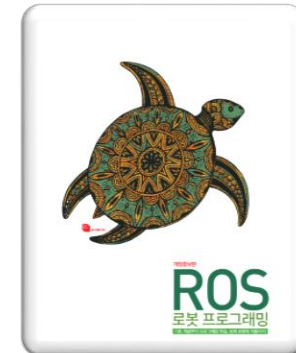
Subscribe

교재

P. 10~23

Contents

- I. ROS 소개
- II. 메타 운영체제
- III. ROS의 목적
- IV. ROS의 구성
- V. ROS의 생태계
- VI. ROS의 역사
- VII. ROS의 버전



[온라인강좌](#)

You Tube

 **Subscribe**

교재

P. 10~23

로봇 운영체제 ROS 소개

ROS 란?

ROS is an open-source, meta-operating system for your robot. It provides the services you would expect from an operating system, including hardware abstraction, low-level device control, implementation of commonly-used functionality, message-passing between processes, and package management. It also provides tools and libraries for obtaining, building, writing, and running code across multiple computers.

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



ROS의 진정한 목적

로보틱스 소프트웨어 개발을 전세계 레벨에서
공동 작업 가능하도록 생태계를 구축하는 것!



ROS는 새로운 운영 체제(OS)인가?

- 운영 체제 (Operating System)
 - 범용 컴퓨터
 - Windows(Windows XP, 7, 8 ...)
 - Linux(Ubuntu, Redhat, Fedora, Mint, Gentoo ...)
 - MAC(OS X ...) 등
 - 스마트폰
 - Android, iOS, Windows Phone, Symbian, RiMO, Tizen등
- ROS = Robot Operating System
- ROS는 메타운영체제(Meta-Operating System)이다.

메타운영체제(Meta-Operating System)

- 메타운영체제(Meta-Operating System) 딱히 정확히 정의된 용어는 아니지만, 어플리케이션과 분산 컴퓨팅 자원간의 가상화 레이어로 분산 컴퓨팅 자원을 활용하여, 스케줄링 및 로드, 감시, 에러 처리 등을 실행하는 시스템이라고 볼 수 있다.
- 즉, 윈도우, 리눅스, 안드로이드와 같은 전통적인 운영체제는 아니다. 오히려, ROS는 기존의 전통적인 운영체제(리눅스, 윈도우즈, OS-X, 안드로이드)를 이용하고 있다.
- 기존 운영체제의 프로세스 관리 시스템, 파일 시스템, 유저 인터페이스, 프로그램 유틸(컴파일러, 스레드 모델 등)등을 사용하고 있다. 이에 추가적으로 다수의 이기종 하드웨어간의 데이터 송수신, 스케줄링, 에러 처리 등 로봇 응용 소프트웨어 개발을 위한 필수 기능들을 라이브러리 형태로 제공하고 있다.
- 또한, 이러한 로봇 소프트웨어 프레임워크를 기반으로 다양한 목적의 응용 프로그램을 개발, 관리, 제공하고 있으며 유저들이 개발한 패키지 또한 유통하는 생태계(ecosystem)를 갖추고 있다.

메타운영체제(Meta-Operating System)

- **메타운영체제(Meta-Operating System)** 딱히 정확히 정의된 용어는 아니지만, 어플리케이션과 분산 컴퓨팅 자원간의 가상화 레이어로 분산 컴퓨팅 자원을 활용하여, 스케줄링 및 로드, 감시, 에러 처리 등을 실행하는 시스템이라고 볼 수 있다.
- 즉, 윈도우, 리눅스, 안드로이드와 같은 **전통적인 운영체제는 아니다**. 오히려, ROS는 기존의 **전통적인 운영체제(리눅스, 윈도우즈, OS-X, 안드로이드)를 이용**하고 있다.
- 기존 운영체제의 프로세스 관리 시스템, 파일 시스템, 유저 인터페이스, 프로그램 유틸(컴파일러, 스레드 모델 등)등을 사용하고 있다. 이에 추가적으로 다수의 이기종 하드웨어간의 데이터 송수신, 스케줄링, 에러 처리 등 **로봇 응용 소프트웨어 개발을 위한** 필수 기능들을 라이브러리 형태로 제공하고 있다.
- 또한, 이러한 **로봇 소프트웨어 프레임워크**를 기반으로 다양한 목적의 응용 프로그램을 개발, 관리, 제공하고 있으며 유저들이 개발한 패키지 또한 유통하는 **생태계(ecosystem)**를 갖추고 있다.

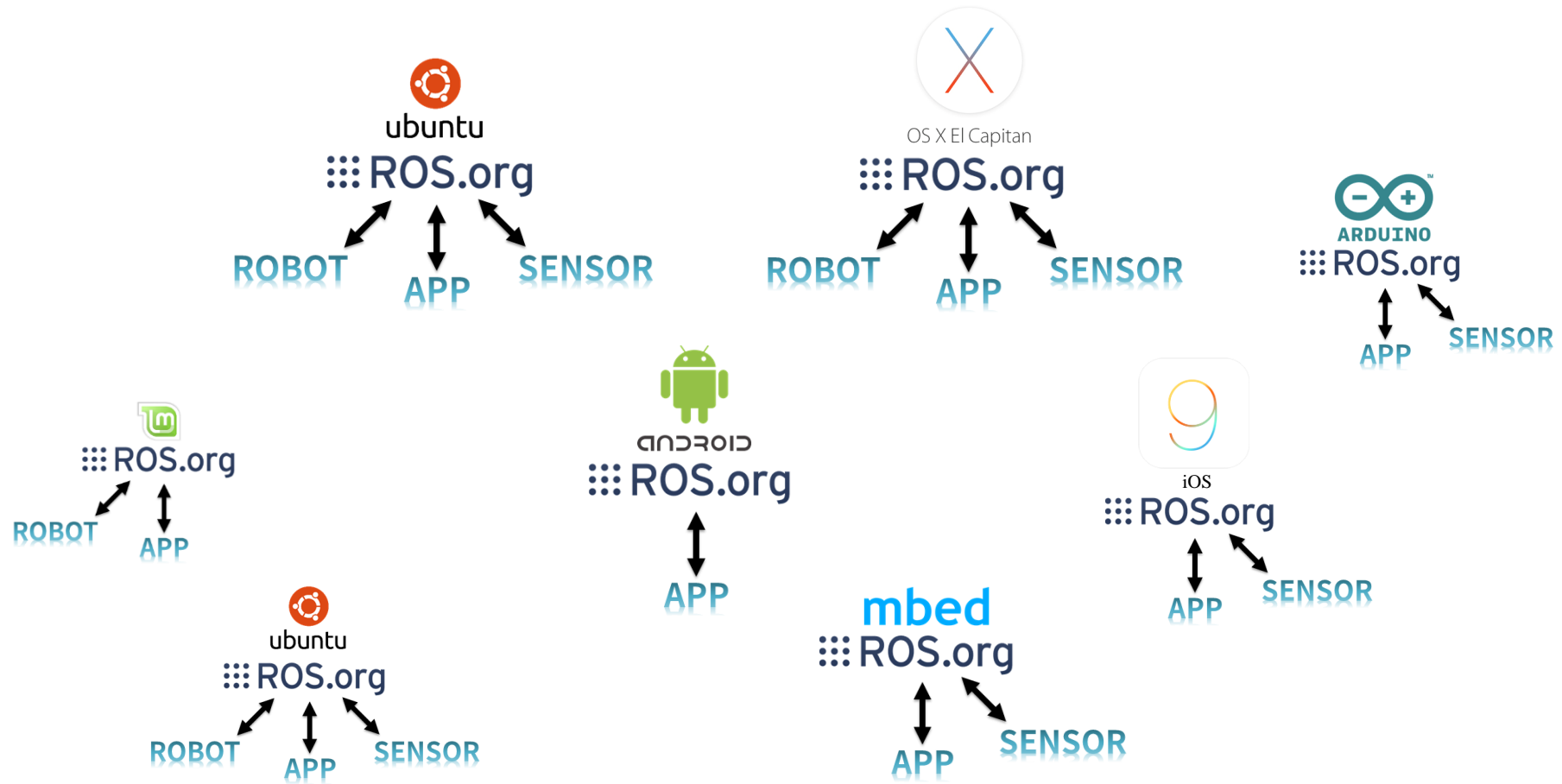
메타운영체제(Meta-Operating System)



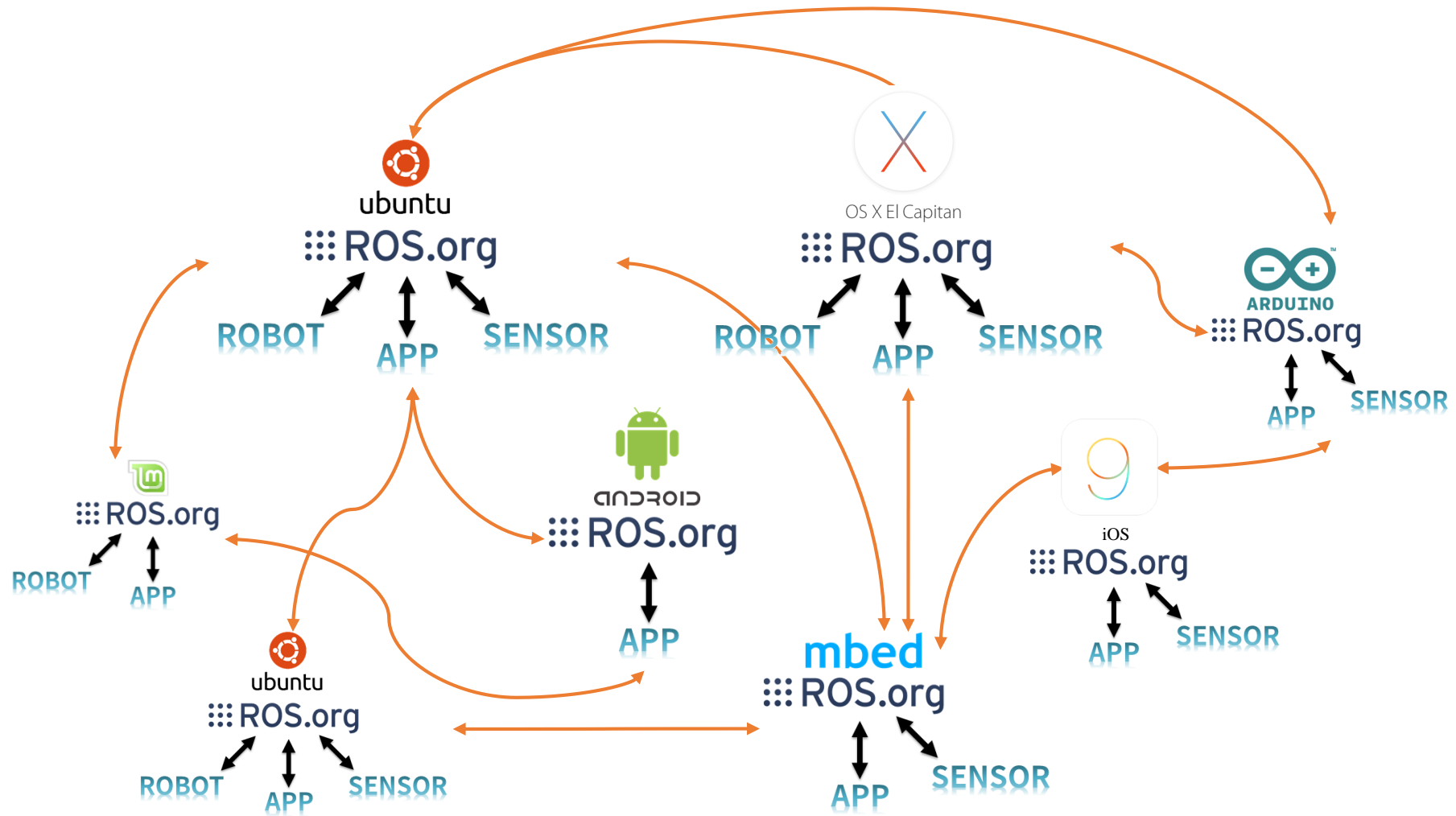
디바이스 드라이버, 라이브러리, 디버그 도구, 메시지 통신
구동 도구, 컴파일 도구, 인스톨러, 패키지 생성 및 릴리즈



이기종 디바이스 간의 통신 지원



이기종 디바이스 간의 통신 지원



ROS를 사용 가능한 운영체제

■ 기존 전통적인 운영체제






























































































- ROS를 사용 가능한 운영체제(OS)로는 Ubuntu, OS X, Windows, Fedora, Gentoo, OpenSUSE, Debian, Raspbian, Arch, QNX Realtime OS 등이 있으나 기능 제한사항이 있을 수 있다.
- 스마트폰 운영체제인 Android, iOS 의 경우, 부분적 사용 가능
- OS를 탑재할 수 없는 마이크로 컨트롤러 유닛(MCU)의 경우, 시리얼 통신, 블루투스, LAN 경유로 통신할 수 있는 라이브러리 제공
- 기본적으로 **Ubuntu**, OS X 에서 구동하는 것을 추천!

ROS 2.0 은
3대 OS 모두 지원

ROS의 구성

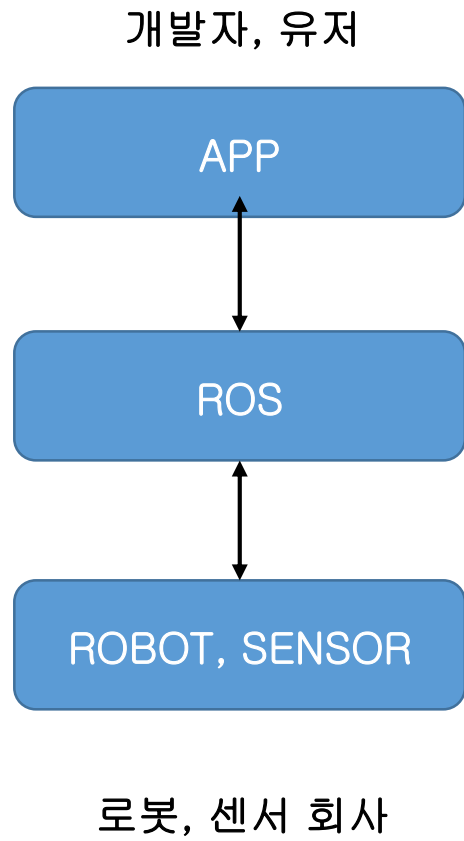
| | | | | | | | |
|---------------------------------------|---------------------|-----------------------------|----------------------|----------------------|-------------------|-----------------|--------|
| Client Layer | roscpp | rospy | roslisp | rosjava | roslibjs | | |
| Robotics Application | MoveIt! | navigatioin | executive smach | descartes | rospieex | | |
| | teleop pkgs | rocon | mapviz | people | ar track | | |
| Robotics Application Framework | dynamic reconfigure | robot localization | robot pose ekf | Industrial core | robot web tools | ros realtime | mavros |
| | tf | robot state publisher | robot model | ros control | calibration | octomap mapping | |
| | vision opencv | image pipeline | laser pipeline | perception pcl | laser filters | ecto | |
| Communication Layer | common msgs | rosbag | actionlib | pluginlib | rostopic | rosservice | |
| | roscnode | roslaunch | roscparam | roscmaster | rosout | ros console | |
| Hardware Interface Layer | camera drivers | GPS/IMU drivers | joystick drivers | range finder drivers | 3d sensor drivers | diagnostics | |
| | audio common | force/torque sensor drivers | power supply drivers | roscserial | ethernet drivers | ros canopen | |
| Software Development Tools | RViz | rqt | wstool | roscpack | catkin | rosdep | |
| Simulation | gazebo ros pkgs | stage ros | | | | | |

ROS의 생태계

| | | | | | | | | | | | |
|--|--|---|---|--|--|--|---|--|---|--|--|
|  210 Stanley Innovation V3 Segway |  220 Stanley Innovation V3 Segway |  223 Innok Heros |  224 Innok Heros |  Erle-Rover |  evarobot |  Fanuc Robotics (ROS-Industrial) |  Festo Didactic Robotino |  Clearpath Robotics Grizzly |  Clearpath Robotics Husky |  Clearpath Robotics Jackal |  Clearpath Robotics Kingfisher |
|  420 Omni Stanley Innovation V3 Segway |  440LE Stanley Innovation V3 Segway |  440SE Stanley Innovation V3 Segway |  444 Innok Heros |  Fetch robotics: Fetch |  Fetch robotics: Freight |  Fraunhofer IPA Care-O-bot 3 |  Fraunhofer IPA Care-O-bot 4 |  Clearpath Robotics Ridgeback |  Cogniteam Hamster |  CoroWare Corobot |  Cyton-Gamma |
|  ABB Robotics (ROS-Industrial) |  Adept MobileRobots Pioneer family (P3DX, P3AT, ...) |  Blue Robotics BlueX |  Blue Robotics BlueSeeker Jr. |  GoTherel Robot |  i-Cart mini |  Ingenierie teRC |  Lindero ADAS Development Vehicle |  Dataspeed Mobility Base |  Denso VS060 |  Dr. Robot Jaguar | |
|  Aldebaran Nao |  Allegro Hand SimLab |  ARMAR-III |  ARMAR-III |  Intel Edison |  iRobot Roomba |  Kawada |  Kobuki |  Enova Robotics MiniLab |  Erle-Brain |  Erle-Brain 2 | |
|  Barrett Hand |  BipedRobin |  Bitcraze Crazyflie |  Blue Robotics BlueROV |  Kinova JACO |  Kinova MICO |  Komodo |  Erle-Copter |  Erle-Copter Ubuntu Core special edition |  Erle-HexaCopter |  Erle-Plane | |
|  Lego NXT |  Lizi |  Maggie |  MecanumBot |  PAL Robotics REEM-C |  PAL Robotics TIAGo |  RazBot |  REEM |  ROS-Industrial |  Ros2Bot |  Shadow Hand |  Softbank Pepper |
|  Merlin miabotPro |  Milvus Robotics ATR |  Milvus Robotics MRP2 |  Milvus Robotics Robin |  Robonaut 2 |  RoboSavvy Self-balance platform |  Robotnik AGVS |  Robotnik GUARDIAN |  Tulip |  TurtleBot |  Universal Robots (ROS-Industrial) |  Videre Erratic |
|  Motoman, Yaskawa (ROS-Industrial) |  Nav2 |  Neobotix mp-500 |  Neobotix mpo-500 |  Robotnik RB-1 |  Robotnik RBCAR |  Robotnik SUMMIT XL |  Robotnik SUMMIT-X |  WheeledRobin |  Willow Garage PR2 |  Xaxxon Oculus Prime |  iRobot Roomba |

ROS
Enabled

ROS의 생태계



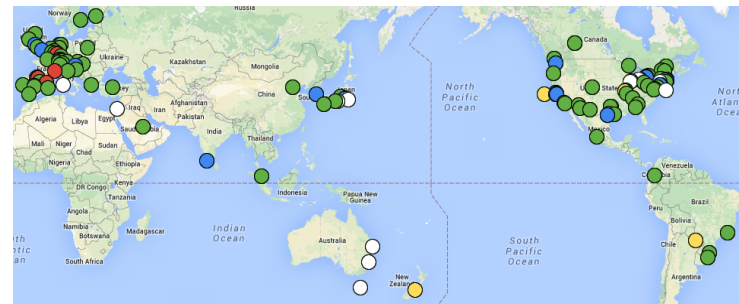
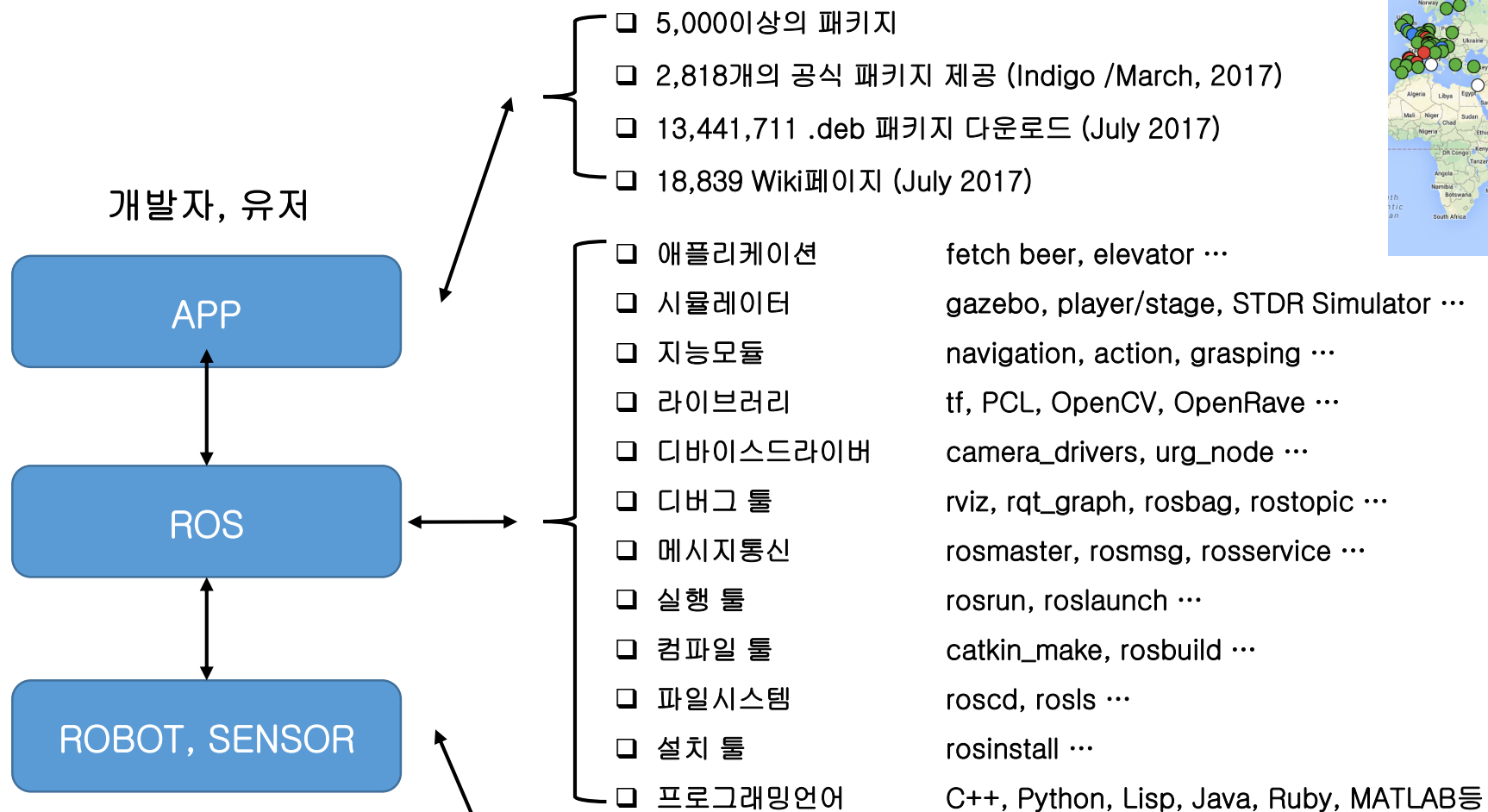
현재의 ROS 생태계



현재의 ROS 생태계



현재의 ROS 생태계



로봇 운영체제 ROS 특징

특징 1) 통신 기능

- 노드 간 데이터 통신을 제공
- 통상적 미들웨어로 지칭되는 메시지 전달 인터페이스 지원
- 메시지 파싱 기능
 - 로봇 개발 시에 빈번히 사용되는 통신 시스템 제공
 - 캡슐화 및 코드 재사용을 촉진하는 노드들 간의 메시지 전달 인터페이스
- 메시지의 기록 및 재생
 - 노드 간 송/수신되는 데이터인 메시지를 저장하고 필요시에 재사용 가능
 - 저장된 메시지를 기반으로 반복적인 실험 가능, 알고리즘 개발에 용이함
- 메시지 사용으로 인한 다양한 프로그래밍 언어 사용 가능
 - 노드 간의 데이터 교환이 메시지를 사용하기 때문에 각 노드는 서로 다른 언어로 작성 가능
 - 클라이언트 라이브러리: roscpp, rospy, roslisp, rosjava, roslua, roscs, roseus, PhaROS, rosR
- 분산 매개 변수 시스템
 - 시스템에서 사용되는 변수를 글로벌 키값으로 작성하여 공유 및 수정하여 실시간으로 반영

특징 2) 로봇 관련 다양한 기능

■ 로봇에 대한 표준 메시지 정의

- 카메라, IMU, 레이저 등의 센서 / 오도메트리, 경로 및 지도 등의 내비게이션 데이터 등의 표준 메시지를 정의하여 모듈화, 협업 작업을 유도, 효율성 향상

■ 로봇 기하학 라이브러리

- 로봇, 센서 등의 상대적 좌표를 트리화 시키는 TF 제공

■ 로봇 기술 언어

- 로봇의 물리적 특성을 설명하는 XML 문서 기술

■ 진단 시스템

- 로봇의 상태를 한눈에 파악할 수 있는 진단 시스템 제공

■ 센싱/인식

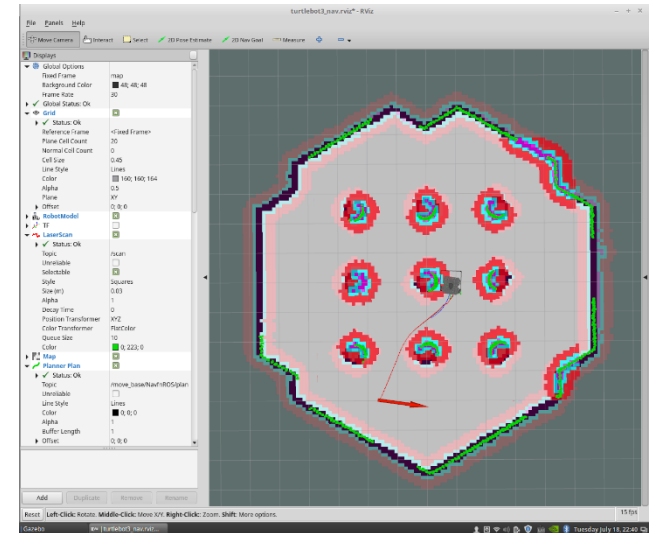
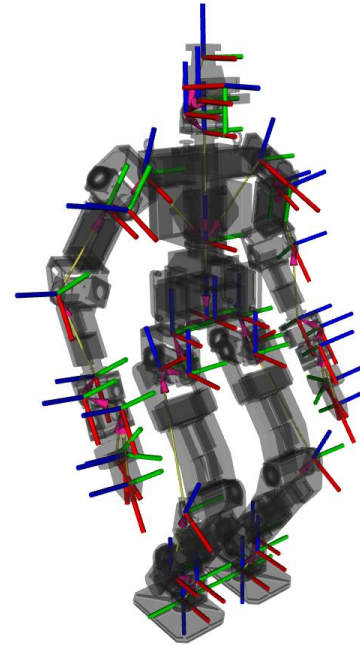
- 센서 드라이버, 센싱/인식 레벨의 라이브러리 제공

■ 내비게이션

- 로봇에서 많이 사용되는 로봇의 포즈(위치/자세) 추정, 지도내의 자기 위치 추정 제공
- 지도 작성에 필요한 SLAM, 작성된 지도 내에서 목적지를 찾아가는 Navigation 라이브러리를 제공

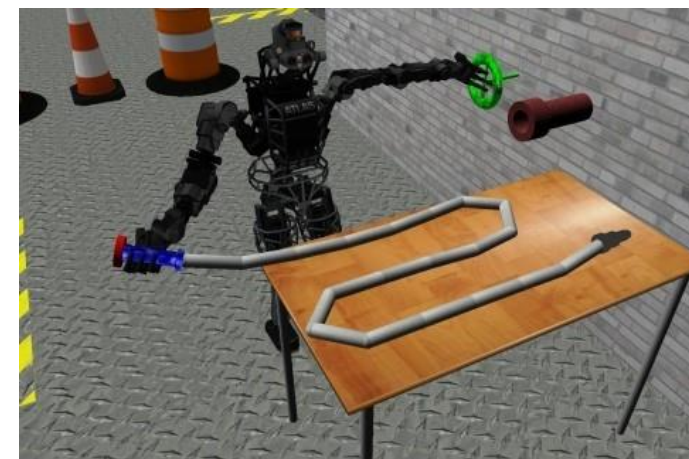
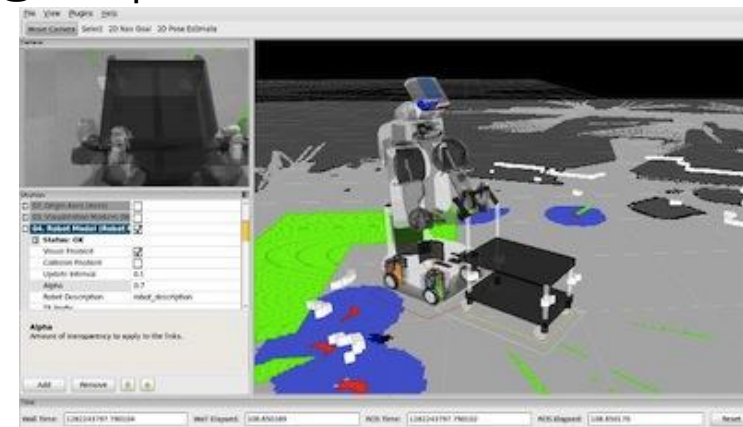
■ 매니퓰레이션

- 로봇 암에 사용되는 IK, FK 는 물론 응용단의 Pick and Place 를 지원하는 다양한 Manipulation 라이브러리 제공
- GUI 형태의 매니퓰레이션 Tools 제공(MoveIt!)



특징 3) 다양한 개발 도구

- 로봇 개발에 필요한 다양한 개발 도구를 제공
- 로봇 개발의 효율성 향상
- **Command-Line Tools**
 - GUI 없이 ROS에서 제공되는 명령어로만 로봇 액세스 및 거의 모든 ROS 기능 소화
- **RViz**
 - 강력한 3D 시각화툴 제공
 - 레이저, 카메라 등의 센서 데이터를 시각화
 - 로봇 외형과 계획된 동작을 표현
- **RQT**
 - 그래픽 인터페이스 개발을 위한 Qt 기반 프레임 워크 제공
 - 노드와 그들 사이의 연결 정보 표시(rqt_graph)
 - 인코더, 전압, 또는 시간이 지남에 따라 변화하는 숫자를 플로팅(rqt_plot)
 - 데이터를 메시지 형태로 기록하고 재생(rqt_bag)
- **Gazebo**
 - 물리 엔진을 탑재, 로봇, 센서, 환경 모델 등을 지원, 3차원 시뮬레이터
 - ROS와의 높은 호환성



ROS 버전 선택

ROS 릴리즈 스케줄과 버전 선택!

- Hydro부터 1년에 1번 정식 버전 릴리즈하기로 결정!

- 2020.05.23 - Noetic Ninjemys (LTS)
- 2018.05.23 - Melodic Morenia (LTS)
- 2017.05.23 - Lunar Loggerhead
- **2016.05.23 - Kinetic Kame (LTS) 추천**
- 2015.05.23 - Jade Turtle
- 2014.07.22 - Indigo Igloo (LTS)
- 2013.09.04 - Hydro Medusa
- 2012.12.31 - Groovy Galapagos
- 2012.04.23 - Fuerte Turtle
- 2011.08.30 - Electric Emys
- 2011.03.02 - Diamondback
- 2010.08.02 - C Turtle
- 2010.03.02 - Box Turtle
- 2010.01.22 - ROS 1.0
- Noetic Ninjemys (EOL=May 2025)
- Melodic Morenia (EOL=May 2023)
- Lunar Loggerhead (EOL=May, 2019)
- Kinetic Kame (EOL=April, 2021)
- Jade Turtle (EOL=May, 2017)
- Indigo Igloo (EOL=April, 2019)

버전선택!

1) 리눅스

- 5년간 기술 지원되는 최신 LTS버전의 우분투 선택
- 2년마다 매년 4월 LTS 버전 릴리즈, 릴리즈 3개월 이후

2) ROS

- 2년마다 매년 5월 LTS 버전 릴리즈, 릴리즈 3개월 이후

3) Gazebo

- "gazebosim.org" 에서 ROS 호환성 정보 검토 후 사용



2019년 상반기까지 추천

- **Ubuntu 16.04.x Xenial Xerus LTS**
- **ROS Kinetic Kame**
- **Gazebo 7.x**