

ROS 로봇프로그래밍

-ROS1 완독하기 챌린지-

김예나

Ch1. 로봇 소프트웨어 플랫폼

1.1 플랫폼의 구성 요소

플랫폼의 4대 생태계 구성 요소

- 1) 하드웨어 모듈(Hardware): 다양한 하드웨어들과의 결합 가능
- 2) 운영체제(Operating System): 하드웨어들을 관리
ex) 윈도우, 리눅스, 안드로이드, iOS
- 3) 응용프로그램(Application): 운영체제에서 제공하는 하드웨어 추상화 기반의
소프트웨어 개발환경에서 다양한 서비스 제공
- 4) 사용자(User): 이를 사용하는 수많은 이용자

⇒ 운영체제로 HW와 SW간의 인터페이스 통합이 되며 하드웨어 모듈화로 이어져

대량생산에 의한 낮은 가격+고성능화+**대중화**+**개인화**

1.2 로봇 소프트웨어 플랫폼

로봇 플랫폼의 종류

1)로봇 HW 플랫폼: 모바일 로봇, 드론, 휴머노이드 연구용
및 산업용(ex. 소프트뱅크사 페퍼, MIT 미디어랩 지보 등)

2)로봇 SW 플랫폼:

- 로봇 응용프로그램을 개발할 때 필요한 하드웨어 추상화
- 하위 디바이스 제어
- 로보틱스에서 많이 사용되는 기능 구현
 - : 센싱, 인식, 동시적 자기 위치 추적, 지도작성(SLAM),
내비게이션(Navigation), 매니퓰레이션(Manipulation) 등
- 패키지 관리
- 개발환경에 필요한 라이브러리와 다양한 개발, 디버깅 도구 등 포함

1.2 로봇 소프트웨어 플랫폼

로봇 SW 플랫폼의 종류

- **ROS(Robot Operating System)**
 - ⇒ 가장 많이 사용되고 있는 로봇 SW 플랫폼
- OpenRTM: 일본의 오픈 로보틱스 테크놀로지 미들웨어
- OROCOS: 유럽의 실시간 제어 중심
- OPRoS: 한국
- etc..

⇒ 로봇 SW플랫폼을 통해 로봇 구현을 위한 다양한 기능 중 각자 자신있는 부분을 **공유**함으로써, **협업**하여 어려운 문제 및 기존에 풀지 못한 **문제 해결 및 효율성**을 가져옴

1.3 소프트웨어 플랫폼의 중요성

로봇 SW 플랫폼 ROS의 특징

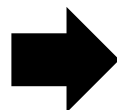
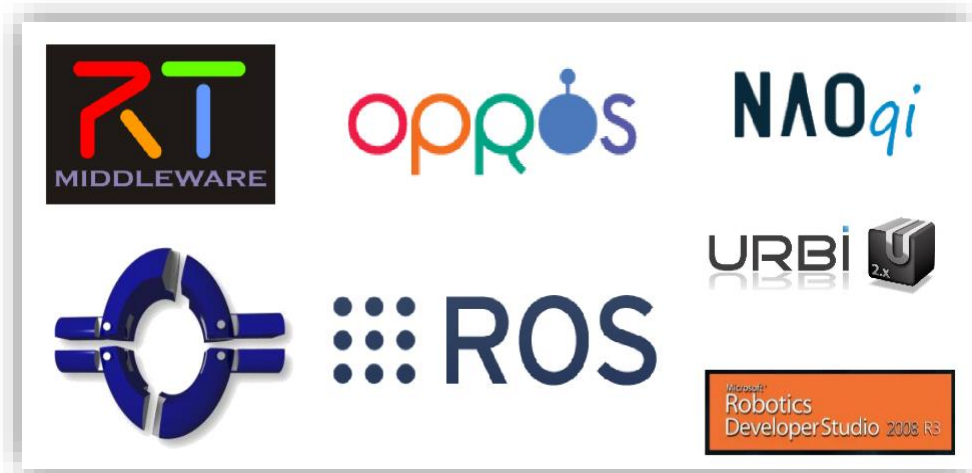
- 1) **프로그램의 재사용성**: 개발하고자 하는 부분에 집중 + 나머지는 관련 패키지 사용
- 2) **통신 기반 프로그램**: 네트워크에서 통신을 제공하여 각 **노드간 데이터 통신** 가능

노드? 하나의 서비스 제공을 위한 센서 및 액추에이터의 하드웨어 드라이버, 센싱, 인식, 동작 등을 각각의 프로세서 목적에 따라 나눈 **최소 실행 단위**

- 3) **개발 도구 지원**: 디버깅 관련 도구, 2차원 플롯(plot), 3차원 시각화(RViz) 도구 제공
- 4) **활성화된 커뮤니티**: 오픈 소스 소프트웨어 플랫폼의 커뮤니티, 협업 활성화
- 5) **생태계 조성**: 하드웨어 분야(로봇과 센서)개발자, ROS 개발 운용팀, 응용 소프트웨어 개발자, 점점 늘어나는 사용자 등 생태계의 틀을 갖추

1.4 로봇 소프트웨어가 가져올 미래

↓ 로봇 소프트웨어 플랫폼의 춘추전국시대



- 커뮤니티 활성화
- 다양한 라이브러리
- 확장성
- 개발 편의성

BUT 플랫폼 만들 것 (X)
응용프로그램 개발 능력에 집중하기!

로봇 소프트웨어 플랫폼의 영향:

하드웨어 전문 지식 없이 응용프로그램 만들기 가능

→ 소프트웨어 인력의 로봇 응용 제품 개발 참여 가능

→ 소프트웨어 플랫폼에서 제안하는 인터페이스에 맞게 하드웨어 설계

→ 로봇 기술의 급속 성장