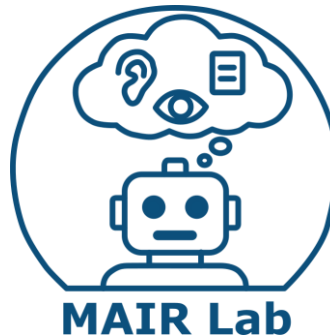


HW2: 온도 센서 + 경고 시스템

운영체제의 실제
안인규 (Inkyu An)



HW2: 온도 센서 + 경고 시스템

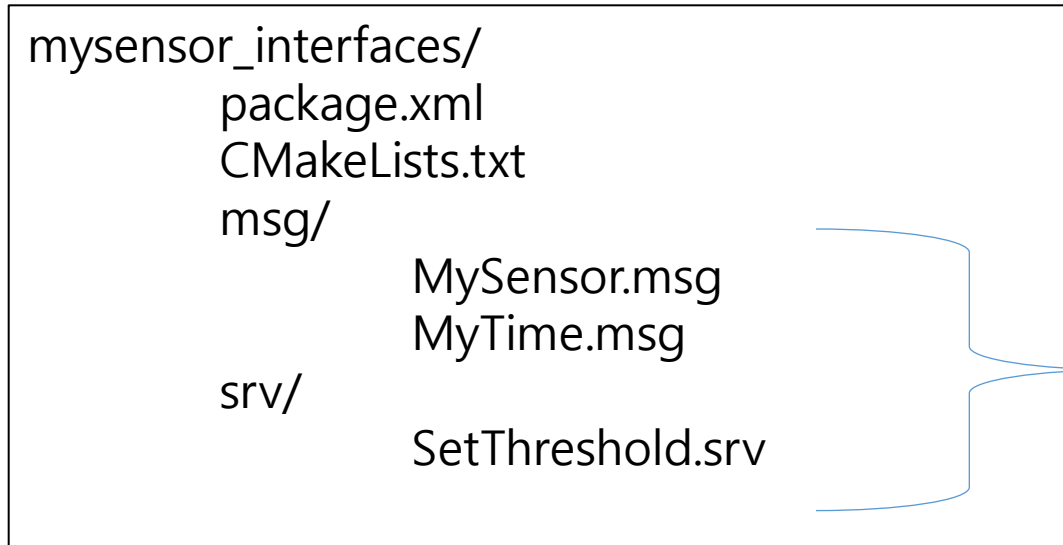
- 목표: 여러 node를 구성하여 간단한 센서 데이터 감시 시스템 만들기
- 구성

노드	역할	주기 / 호출 방식
temperature_publisher	온도 + 습도 + 타임스탬프 등을 퍼블리시	예: 1초 단위
monitor_node	퍼블리시된 데이터를 구독, 이상 감지	callback 시점
threshold_client	임계값 (threshold)을 설정하는 서비스	외부 요청

HW2: 온도 센서 + 경고 시스템

- Package 1: **mysensor_interfaces**
 - 목적: node간 센서 데이터 통신을 위한 msg와 srv 생성

"mysensor_interfaces" 디렉토리



직접 해당 msg 와 srv 파일을 만들어야 함

HW2: 온도 센서 + 경고 시스템

- Package 1: **mysensor_interfaces**

- MySensorMsg.msg

- float64 temp → -10~50 도 사이에서 uniform sample
 - float64 humidity → 10~99 % 사이에서 uniform sample
 - string sensor_id → Sensor name (어떤 것이든 괜찮음)
 - MyTime stamp → Temperature_publisher가 시작된 후 흐른 시간 출력

- MyTime.msg

- int32 sec
 - int32 nsec
 - ROS2의 builtin_interfaces/Time 으로부터 확인하여 update (Hint: node.get_clock().now().to_msg() 이용)

- SetThreshold.srv

- float64 new_threshold
 - ---
 - bool success

HW2: 온도 센서 + 경고 시스템

- Package 2: **sensor_monitor_py**
 - 구성: temperature_publisher, threshold_client, monitor_node
 - temperature_publisher: 1초마다 MySensorMsg를 publish
 - monitor_node:
 - MySensorMsg를 subscribe하여, 터미널에 INFO level로 출력
 - 만약 temperature가 threshold 이상이면, INFO WARN로 출력
 - threshold는 SetThreshold.srv를 통해 설정 가능 (Service)
 - threshold_client:
 - 입력 argument를 통해, new threshold를 SetThreshold.srv로 monitor_node로 전송 (client)
 - Update가 성공했다면 INFO level로 출력 / 실패했다면 WARN level로 출력