

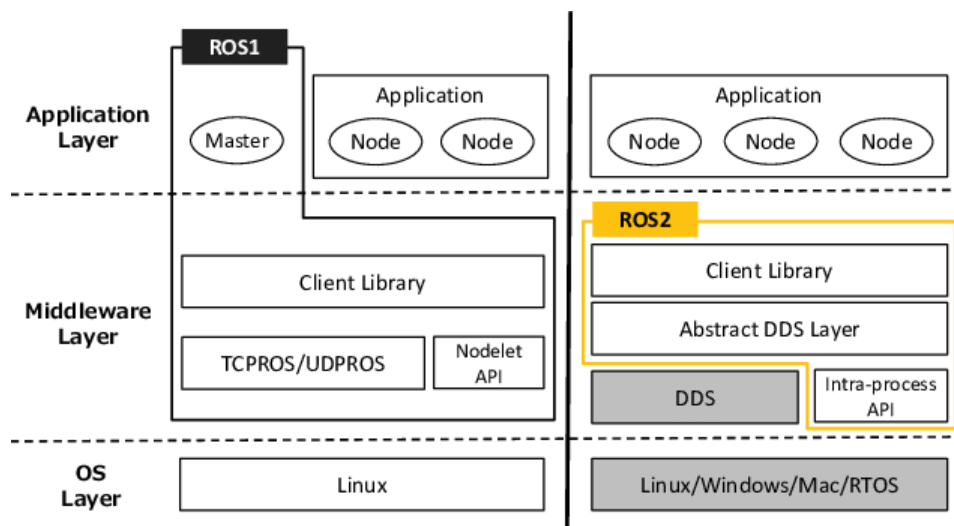
# 1부 7장 \_ROS2와 DDS

## 요약

- DDS 도입으로 인한 ROS2의 변화

## ROS2와 DDS

- ROS1에서 사용했던 메시지 통신 (토픽, 서비스, 액션, 파라미터)을 ROS2에서도 중요한 개념으로 가져왔으나 통신 라이브러리가 기존 TCPROS에서 DDS의 RTPS로 바뀌어 ROS의 레이아웃이 많이 바뀌었다.



<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7743223>

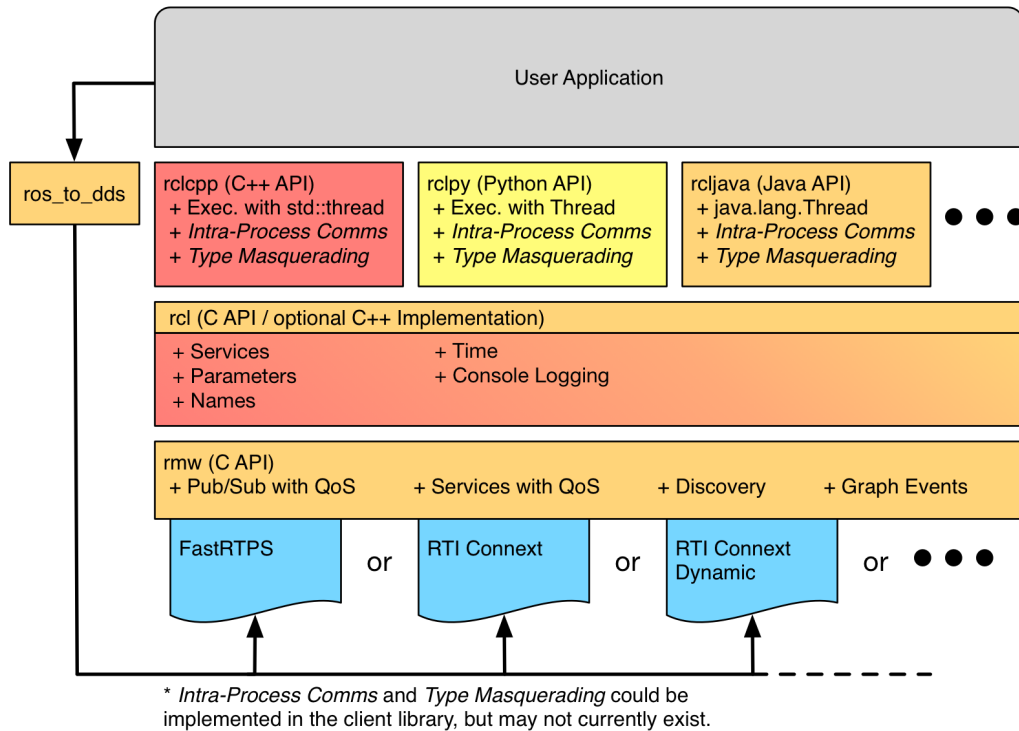
- 기존 메시지 통신 형태 이외에도 DDS 도입으로 인해 IDL을 사용하여 메시지 정의 및 직렬화를 더 쉽게, 포괄적으로 사용할 수 있게 되었다.
- DDSI-RTPS 통신 프로토콜을 채용하여 실시간 데이터 전송을 보장하고 임베디드 시스템에서도 사용할 수 있게 되었다.
- ROS Master 없이도 DDS 프로그램 간에 통신할 수 있다.
- 노드 간의 데이터 통신을 QoS로 세부적으로 조정하여 네트워크 최적화를 할 수 있다.
- DDS-Security로 인해 보안성이 증가하였다.

## DDS란?

- 데이터 분산 시스템으로 데이터 통신을 위한 미들웨어 API다.
  - 참고 사이트 (6장 study)
    - <https://www.dds-foundation.org/>
    - [https://fast-dds.docs.eprosima.com/en/latest/fastdds/getting\\_started/definitions.html](https://fast-dds.docs.eprosima.com/en/latest/fastdds/getting_started/definitions.html)
    - <https://www.omg.org/spec/DDS-RPC/1.0/PDF>
    - [https://blog.naver.com/neos\\_rtos/220346180839](https://blog.naver.com/neos_rtos/220346180839)
    - [https://blog.naver.com/PostList.naver?blogId=neos\\_rtos&from=postList&categoryNo=18&parentCategoryNo=18](https://blog.naver.com/PostList.naver?blogId=neos_rtos&from=postList&categoryNo=18&parentCategoryNo=18)

## DDS의 특징

- 산업 표준
- 운영체제 독립
- 언어 독립
  - 미들웨어이기 때문에 상위 레벨인 사용자 코드 레벨에서 사용하는 프로그래밍 언어를 사용할 수 있다.
  - ROS2에서도 DDS를 RMW (ROS middleware)로 추상화 하였고 (벤더 별로 RMW 지원) 그 위에 사용자 코드를 위한 RCL (ROS Client Library)를 지원한다.



<https://docs.ros.org/en/dashing/Concepts/About-Internal-Interfaces.html>

## • UDP 기반의 전송 방식

- UDP의 멀티캐스트는 브로드 캐스트처럼 여러 목적지로 동시에 데이터를 보낼 수 있지만, 불특정 목적지가 아닌 특정된 도메인 그룹에 대해서만 데이터를 전송한다.
- UDP로 인해 ROS2에서는 전역 공간이라 불리는 DDS Global Space에 등록된 토픽들에 대해서 퍼블리시 및 서브스크라이브를 할 수 있게 되었다.
- UDP의 단점은 QoS를 통해 보완 및 해결되었다.

## • 데이터 중심적 기능

- DCPS (Data-Centric Publish-Subscribe), 적절한 수신자에 적절한 정보를 효율적으로 전달할 수 있는 퍼블리시 및 서브스크라이브 방식을 사용할 수 있다.
  - 어떤 데이터인지
  - 데이터가 어떤 형식인지
  - 데이터를 어떻게 보낼 것인지
  - 데이터를 어떻게 안전하게 보낼 것인지

- 동적 검색 (Dynamic Discovery)
  - 동적 검색 기능을 통해 응용 프로그램은 어떤 토픽이 지정 도메인 영역에 있으며 어떤 노드가 이를 발견하고 수신하는 지 알 수 있다.
  - ROS2에서는 노드를 DDS의 Participant로 취급하고 DDS의 동적 검색 기능을 사용하여 이들을 연결할 수 있게 되었다.
- 확장 가능한 아키텍처
- 상호 운용성
  - DDS의 표준 사양을 지키고 있는 벤더를 사용하면 여러 회사의 벤더를 사용해도 상호 통신 등이 가능하다.
- 서비스 품질 (QoS)
  - 퍼블리셔 및 서브스크라이브 등을 선언할 때 QoS를 통해 노드 간의 DDS 통신 옵션을 설정할 수 있다.
    - Reliability
    - History
    - Durability
    - Deadline
    - Lifespan
    - Liveliness
- 보안
  - DDS-Security
  - SROS2 (Secure ROS2)