Camera Grabber

2024.Summer.URP

손건화 ghson@rcv.sejong.ac.kr

RCV

Understanding Robot

Robot Operating System (ROS)

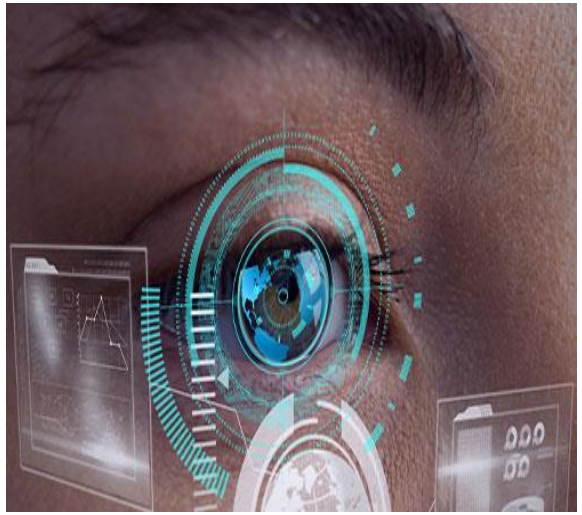
RCV

Understanding robot

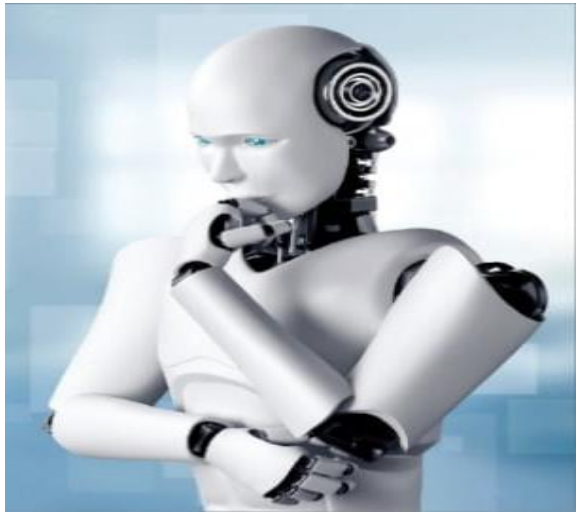
What is recent robot ?

Understanding robot

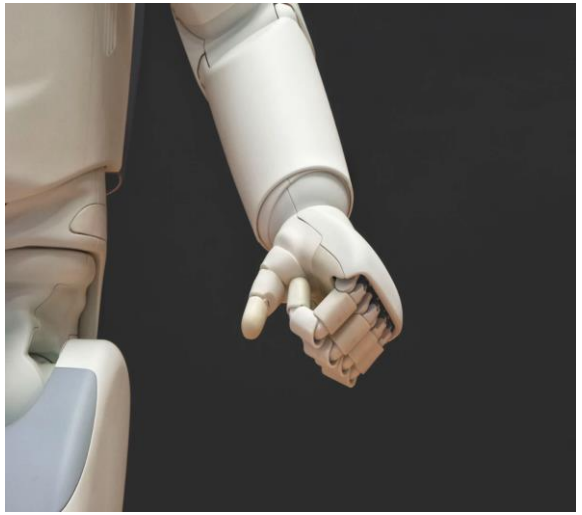
robot ?



인지-센서

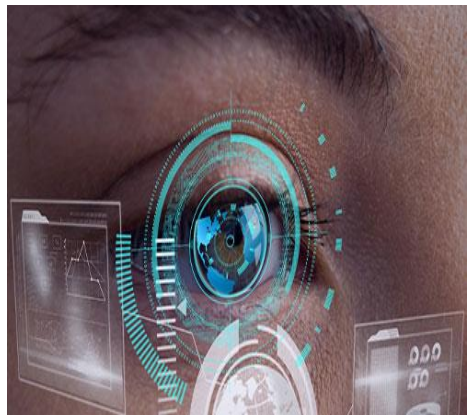


판단-소프트웨어



행동-액츄에이터

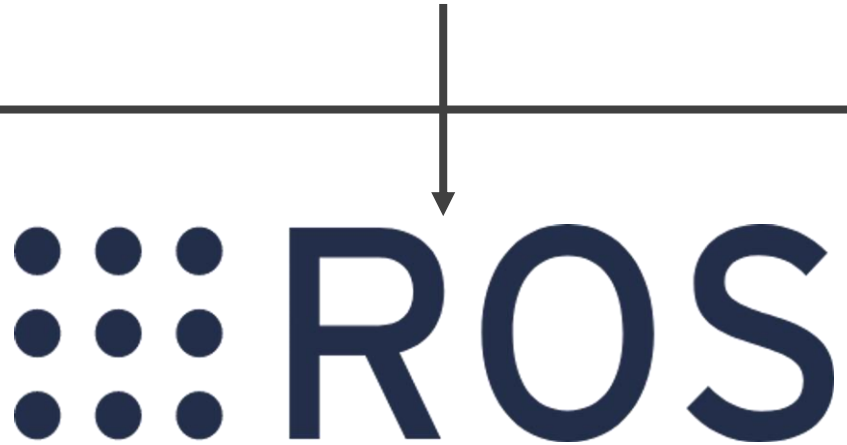
robot ?



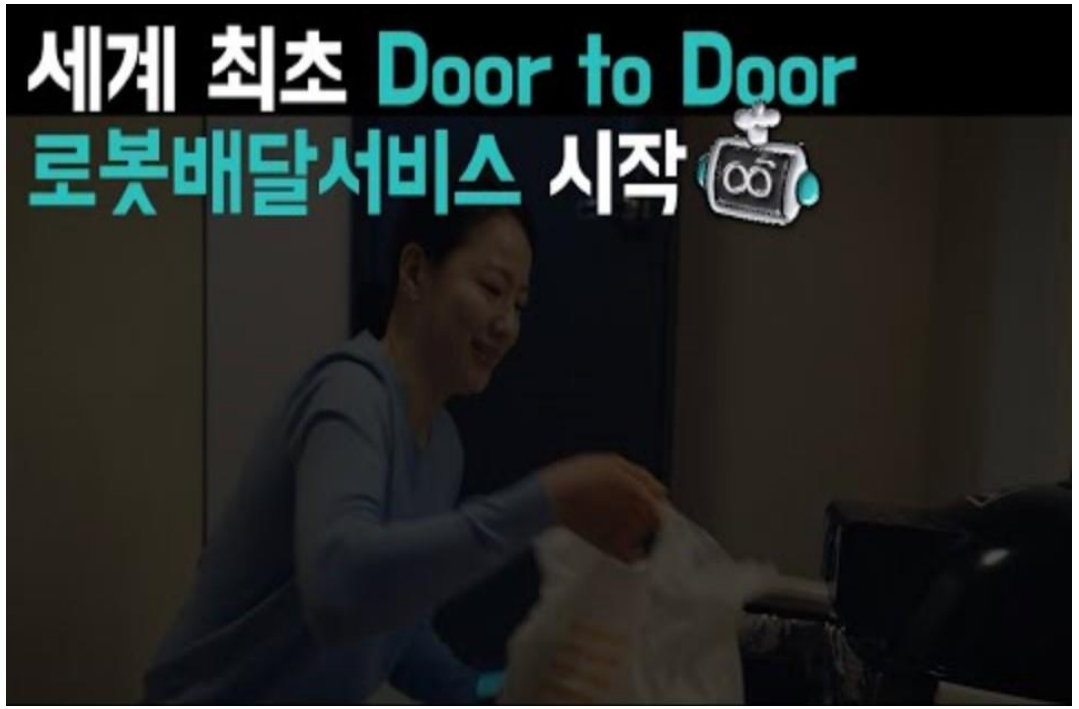
인지-센서

판단-소프트웨어

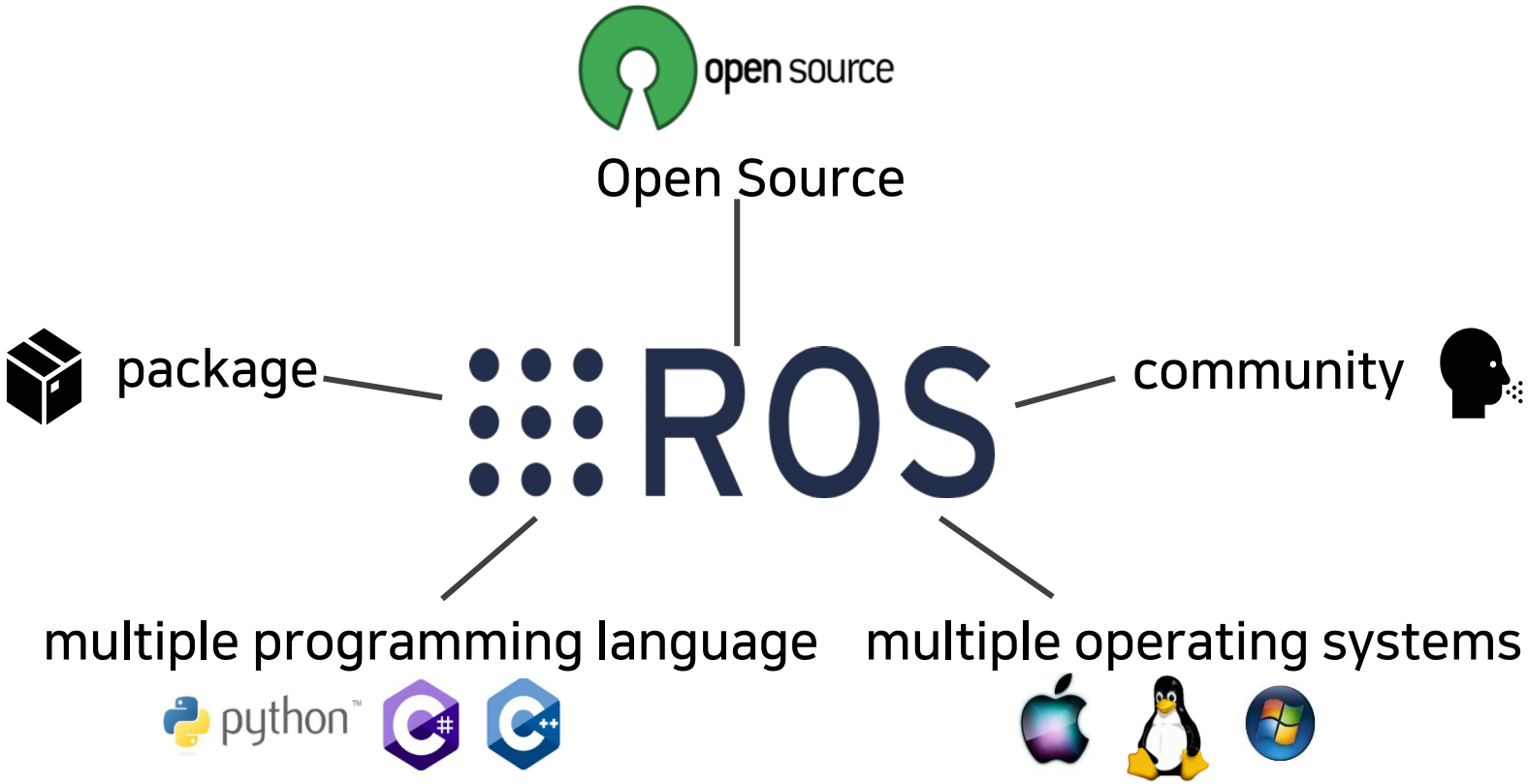
행동-액츄에이터



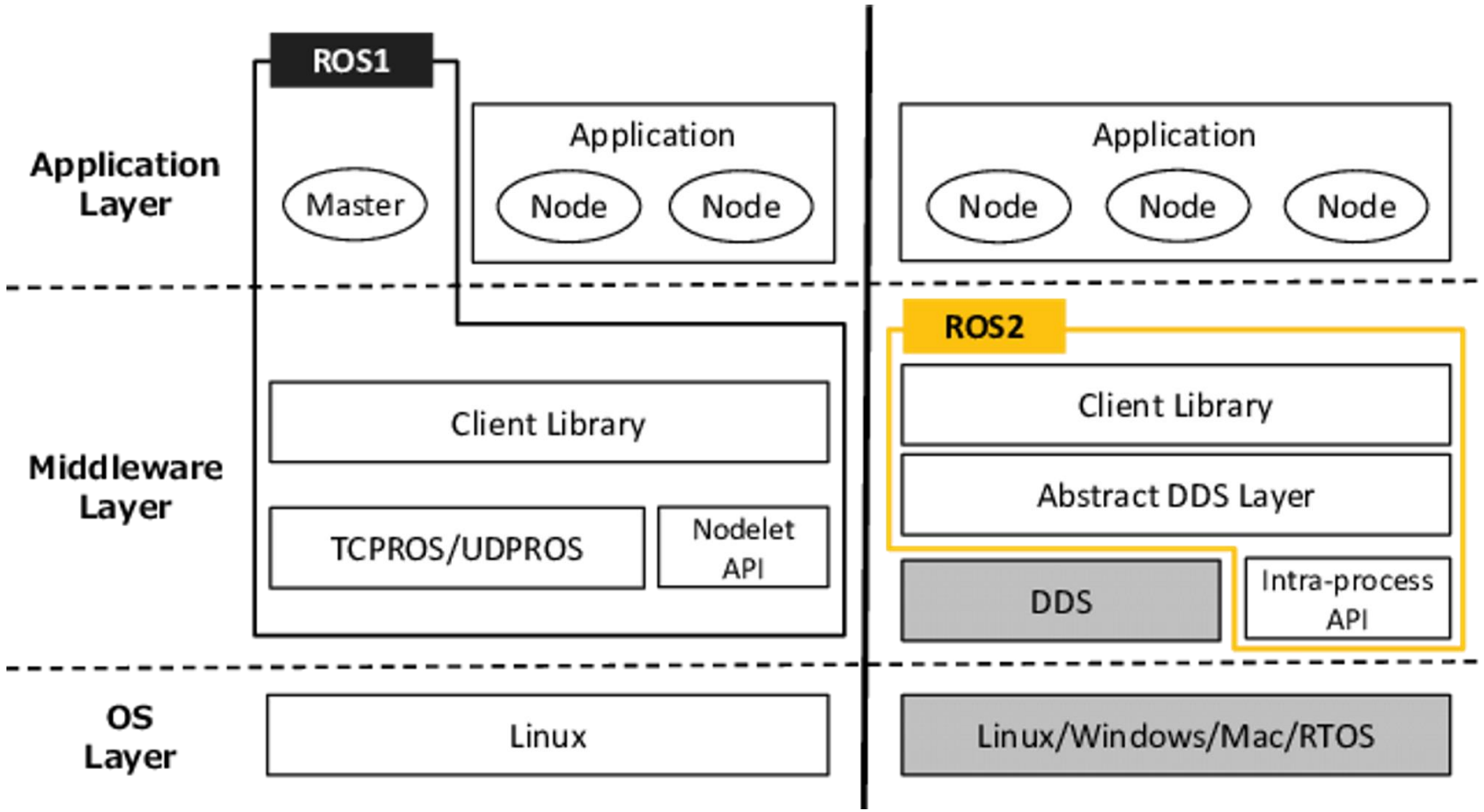
Leveraging ROS



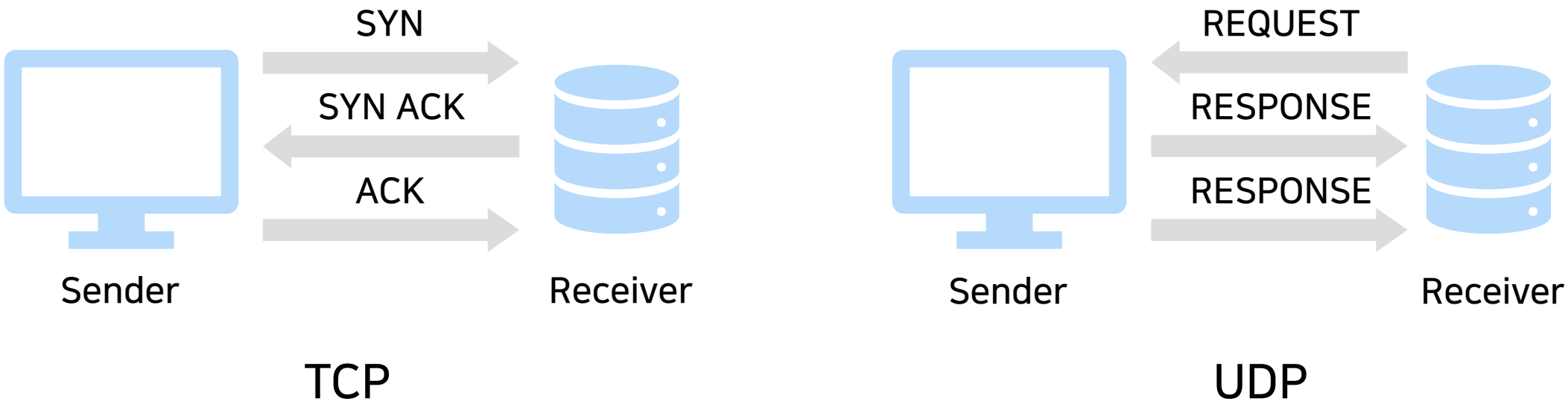
ROS


open source
Open Source
package
ROS
community
multiple programming language
python
multiple operating systems

ROS vs ROS2

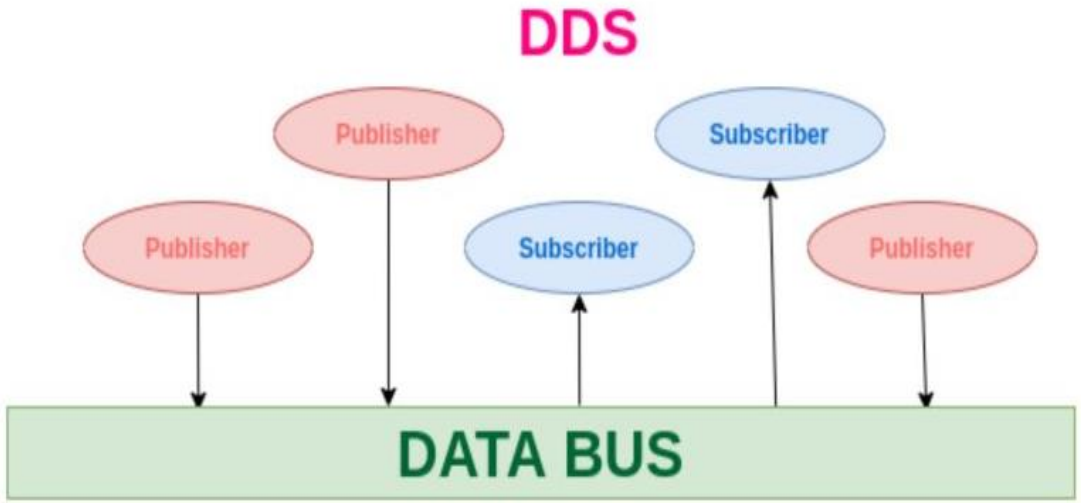

ROS1
Application Layer
Application
Master
Node
Node
Application
Node
Node
Node
ROS2
Middleware Layer
Client Library
TCPROS/UDPROS
Nodelet API
Client Library
Abstract DDS Layer
DDS
Intra-process API
OS Layer
Linux
Linux/Windows/Mac/RTOS

TCP vs UDP


SYN
SYN ACK
ACK
Sender
Receiver
TCP
REQUEST
RESPONSE
RESPONSE
Sender
Receiver
UDP

DDS (Data Distribution Service)

- DDS (Data Distribution Service)

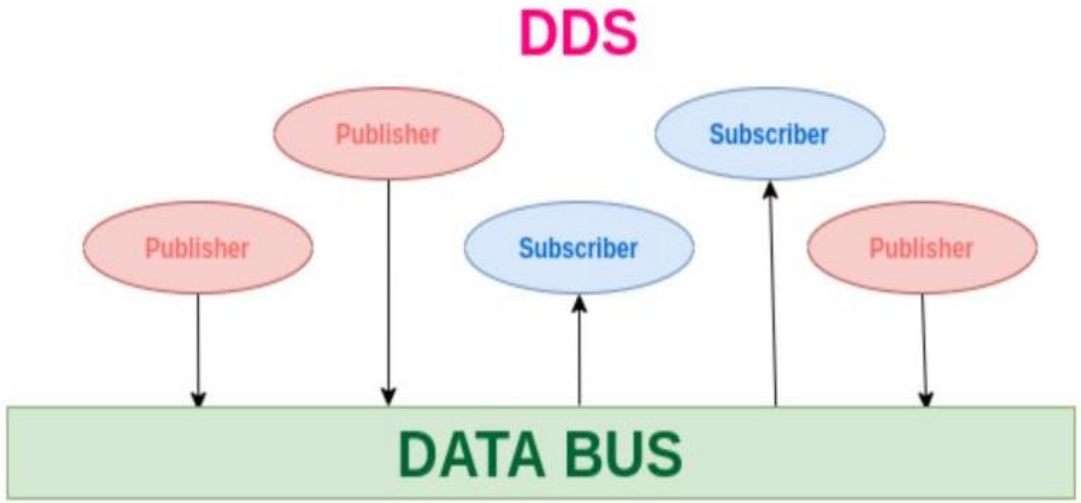


- Industrial Standards
- Operating system independence
- Language independence
- UDP based Transport Protocol
- Data centric
- Dynamic Discovery

기술 표준을 위해 비영리 단체인 OMG가 만든 미들웨어 산업 표준 방식

DDS (Data Distribution Service)

- DDS (Data Distribution Service)

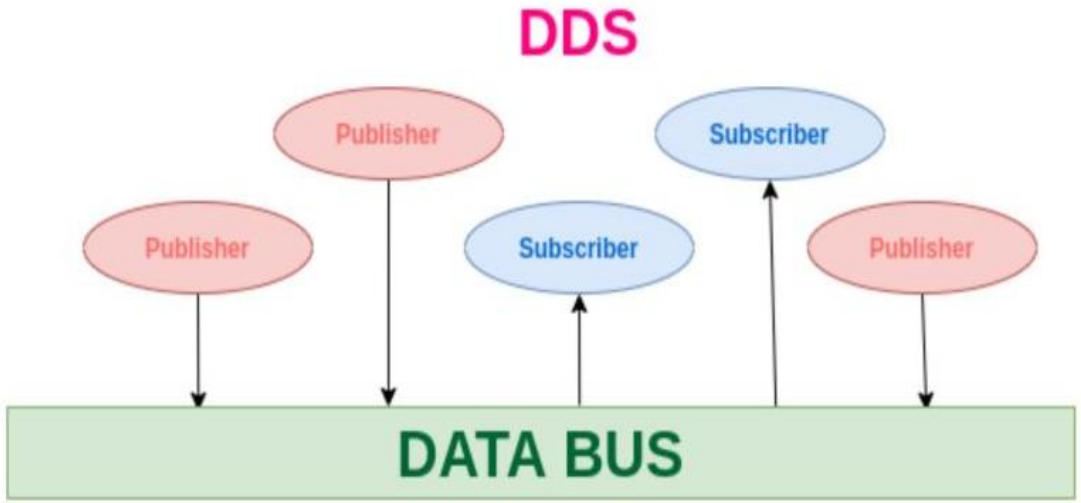


- Industrial Standards
- Operating system independence
- Language independence
- UDP based Transport Protocol
- Data centric
- Dynamic Discovery

리눅스, 윈도우와 같이 다양한 운영 체제 지원

DDS (Data Distribution Service)

- DDS (Data Distribution Service)

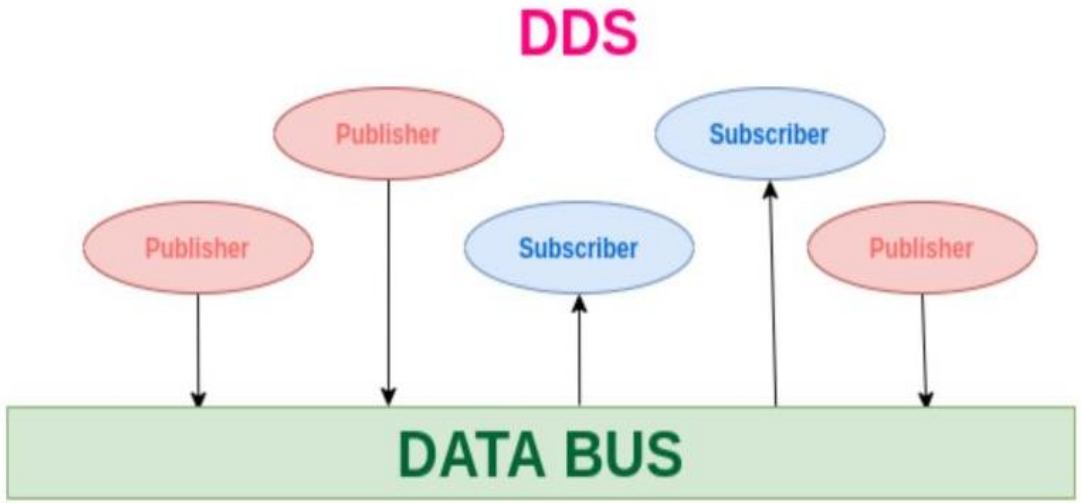


- Industrial Standards
- Operating system independence
- Language independence
- UDP based Transport Protocol
- Data centric
- Dynamic Discovery

사용자 어플리케이션에서의 코드 레벨로 변경하지 않아도 되는 다양한 프로그래밍 언어 지원

DDS (Data Distribution Service)

- DDS (Data Distribution Service)

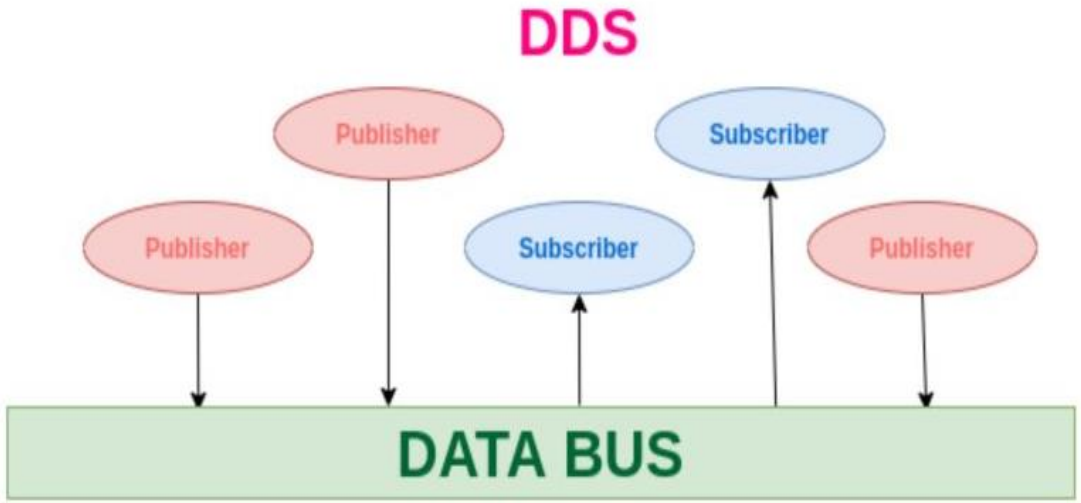


- Industrial Standards
- Operating system independence
- Language independence
- UDP based Transport Protocol
- Data centric
- Dynamic Discovery

QoS를 통해 신뢰성을 갖춘 UDP 기반 전송 방식

DDS (Data Distribution Service)

- DDS (Data Distribution Service)

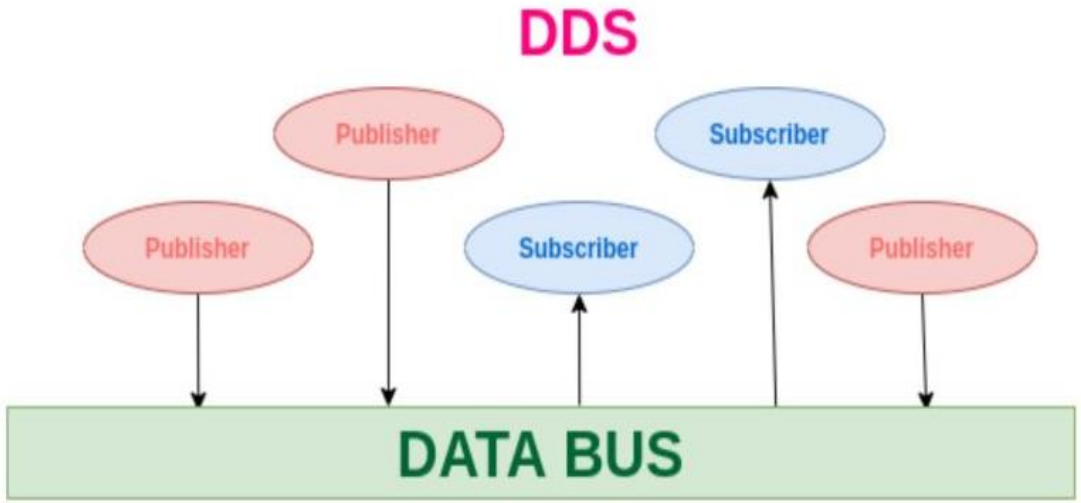


- Industrial Standards
- Operating system independence
- Language independence
- UDP based Transport Protocol
- Data centric
- Dynamic Discovery

수신자에게 적절한 정보를 전달할 수 있는 데이터 중심의 통신

DDS (Data Distribution Service)

- DDS (Data Distribution Service)



- Industrial Standards
- Operating system independence
- Language independence
- UDP based Transport Protocol
- Data centric
- Dynamic Discovery

ROS 마스터를 사용하지 않고 연결할 수 있는 동적 검색

QoS (Quality of Service)

1. History

Value	Description
KEEP_LAST	Store data for a set message queue size
KEEP_ALL	Keep all data

2. Reliability

Value	Description
BEST_EFFORT	Focuses on sending data and speed
RELIABLE	Focuses on getting data and reliability

3. Lifespan

Value	Description
Lifespan_duration	How often to check the Lifespan

4. Durability

Value	Description
TRANSIENT_LOCAL	Keep data from before the subscription was created
VOLATILE	Data before the subscription is created is invalid

5. Deadline

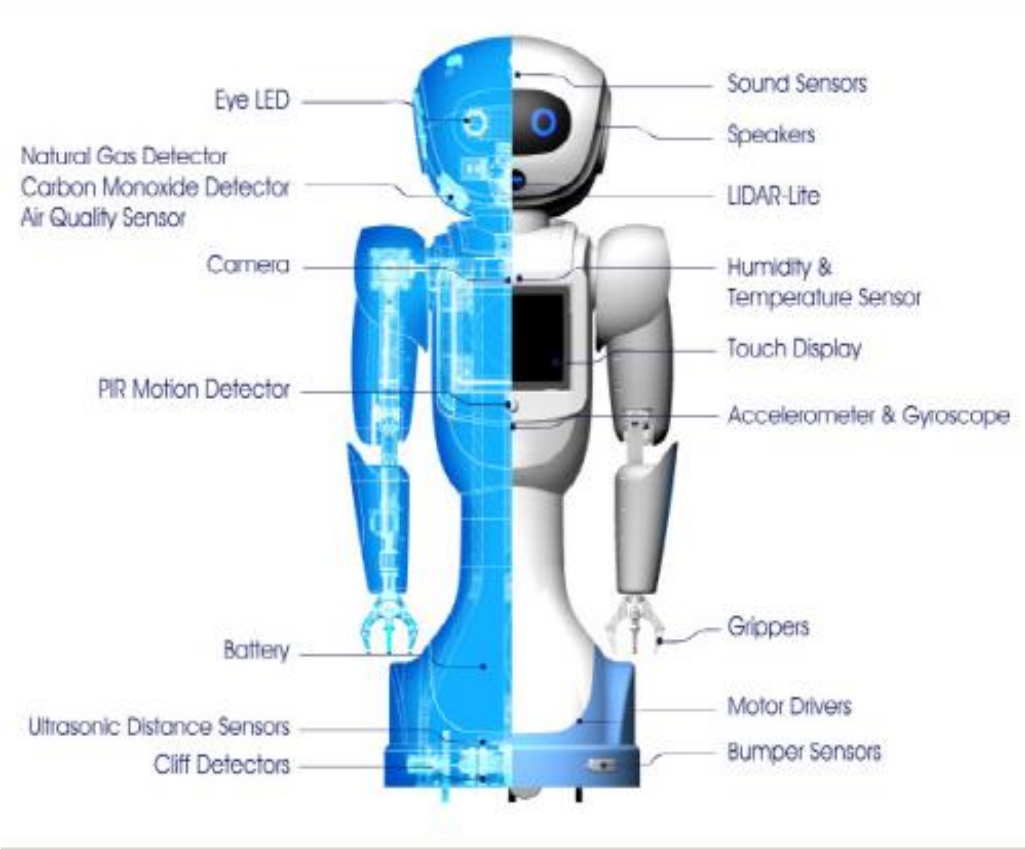
Value	Description
deadline_duration	How often to check the deadline.

6. Liveliness

Value	Description
lineliness	Option to specify whether to check automatically or manually.
lease_duration	How often to check the Liveness

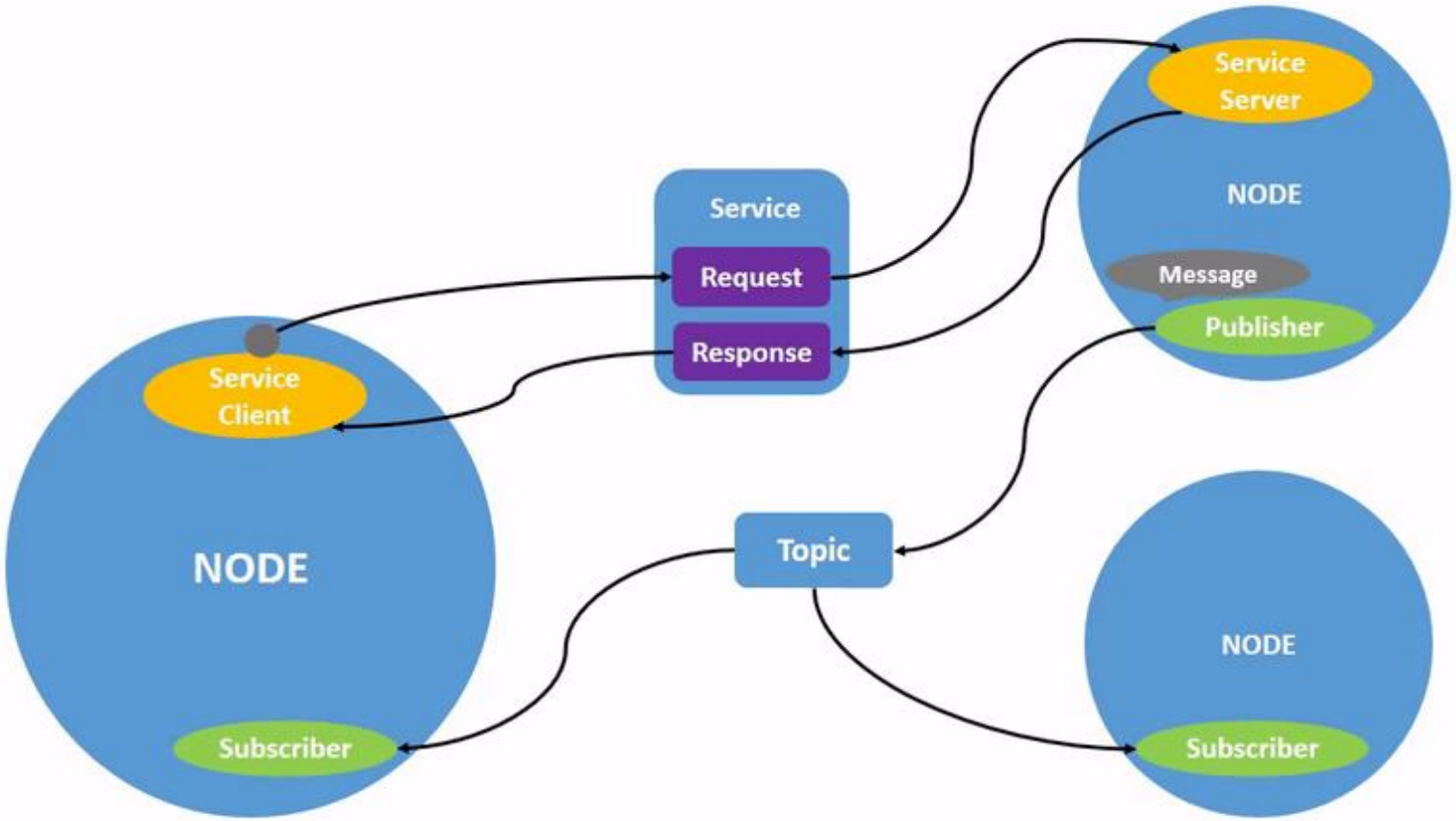
Understanding ROS

Node & Package



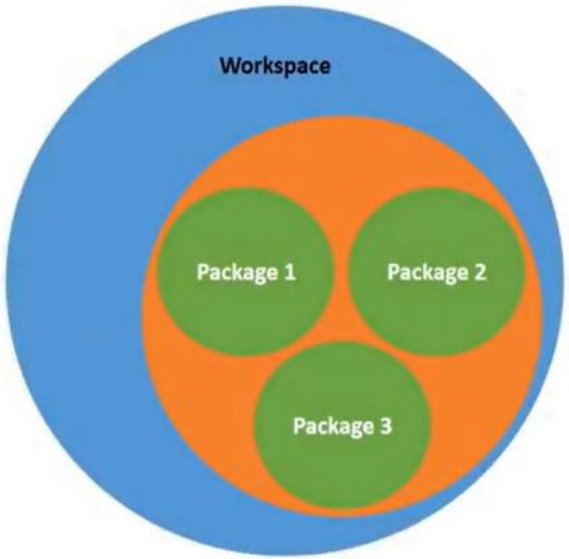
Understanding ROS

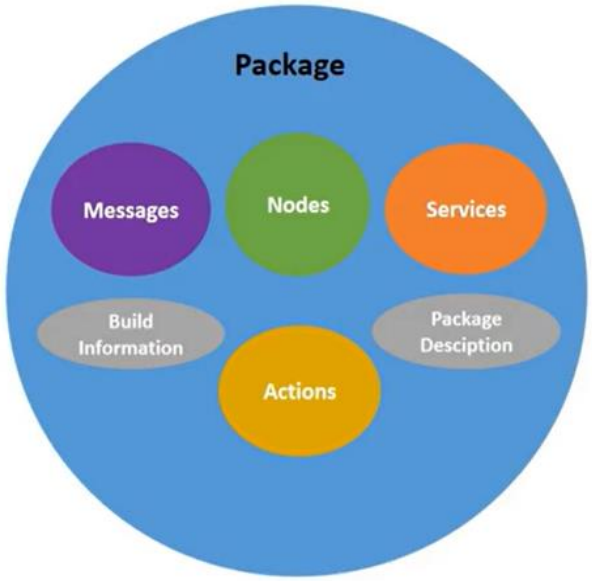
Node & Package



Understanding ROS

Node & **Package**





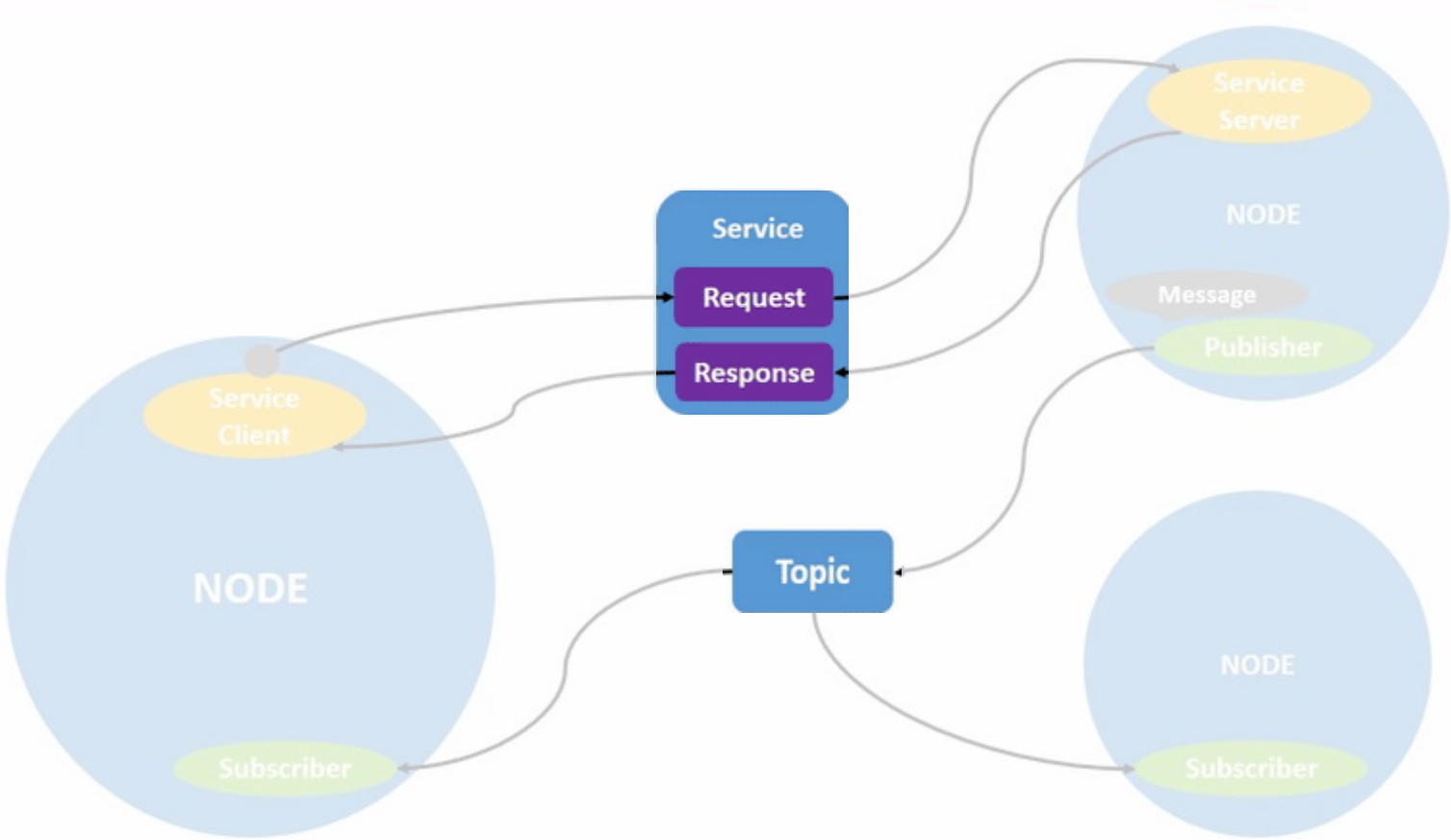

```
.
├── topic_helloworld
│   └── __init__.py
├── resource
│   └── topic_helloworld
├── test
│   ├── test_copyright.py
│   ├── test_flake8.py
│   └── test_pep257.py
│
├── package.xml # 패키지 설정 파일
├── setup.cfg # 파이썬 패키지 환경설정 파일
└── setup.py # 파이썬 패키지 설정 파일

3 directories, 8 files
```

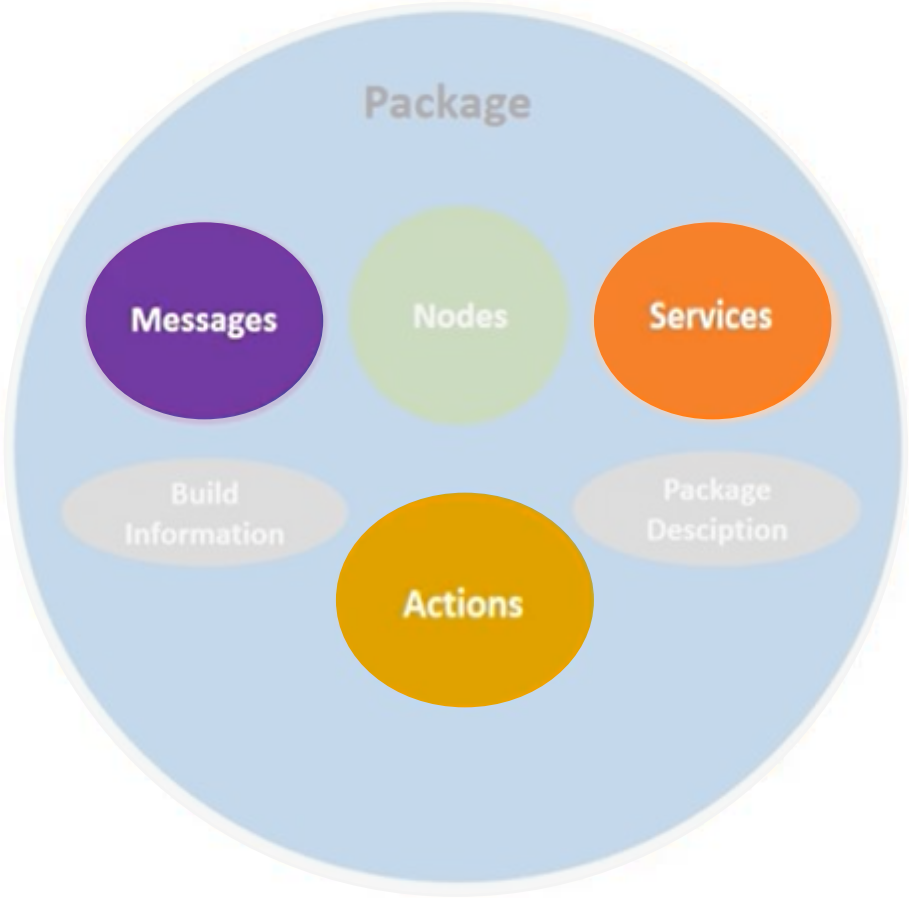
RCV

Understanding ROS

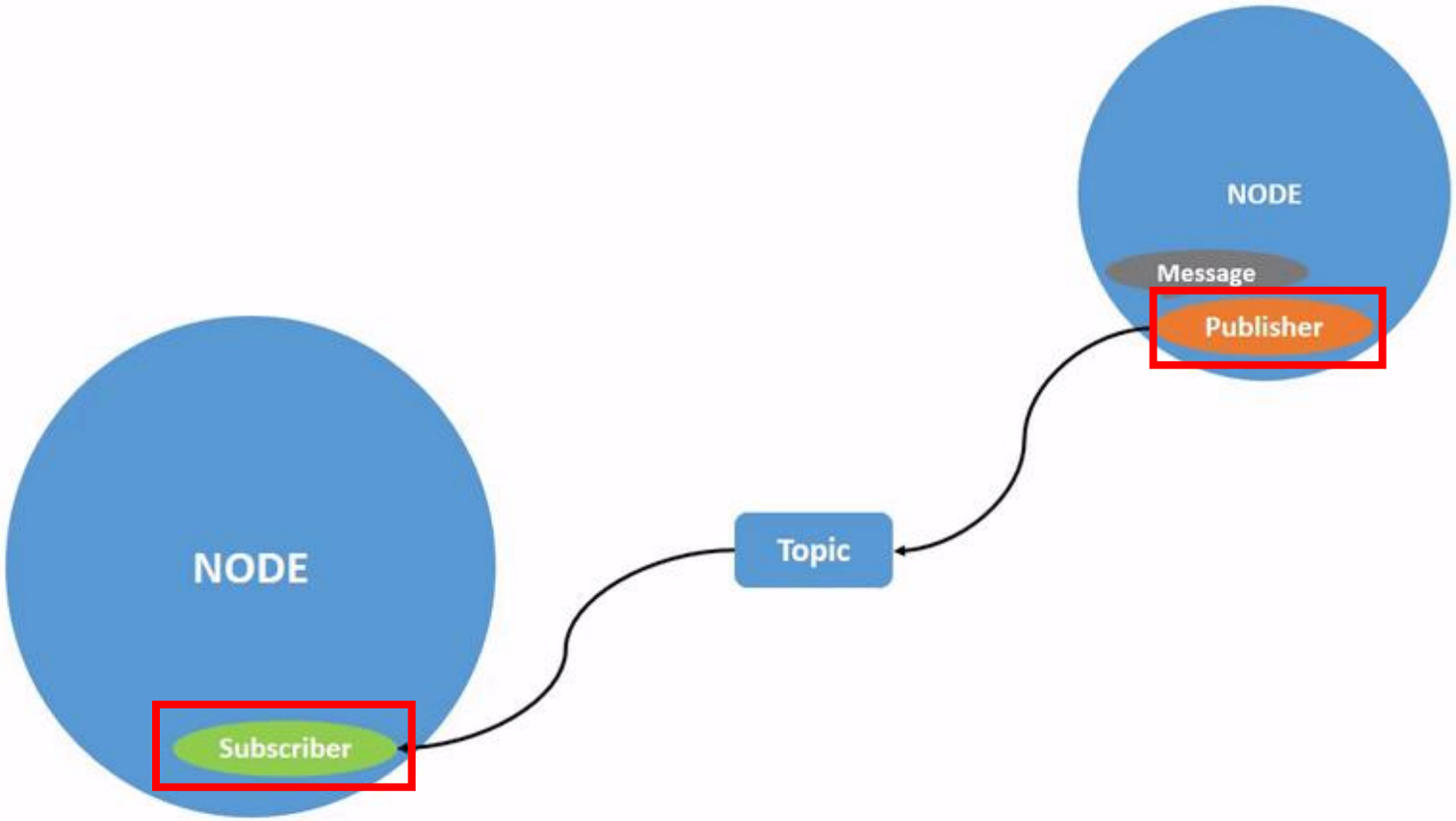
Node & Package



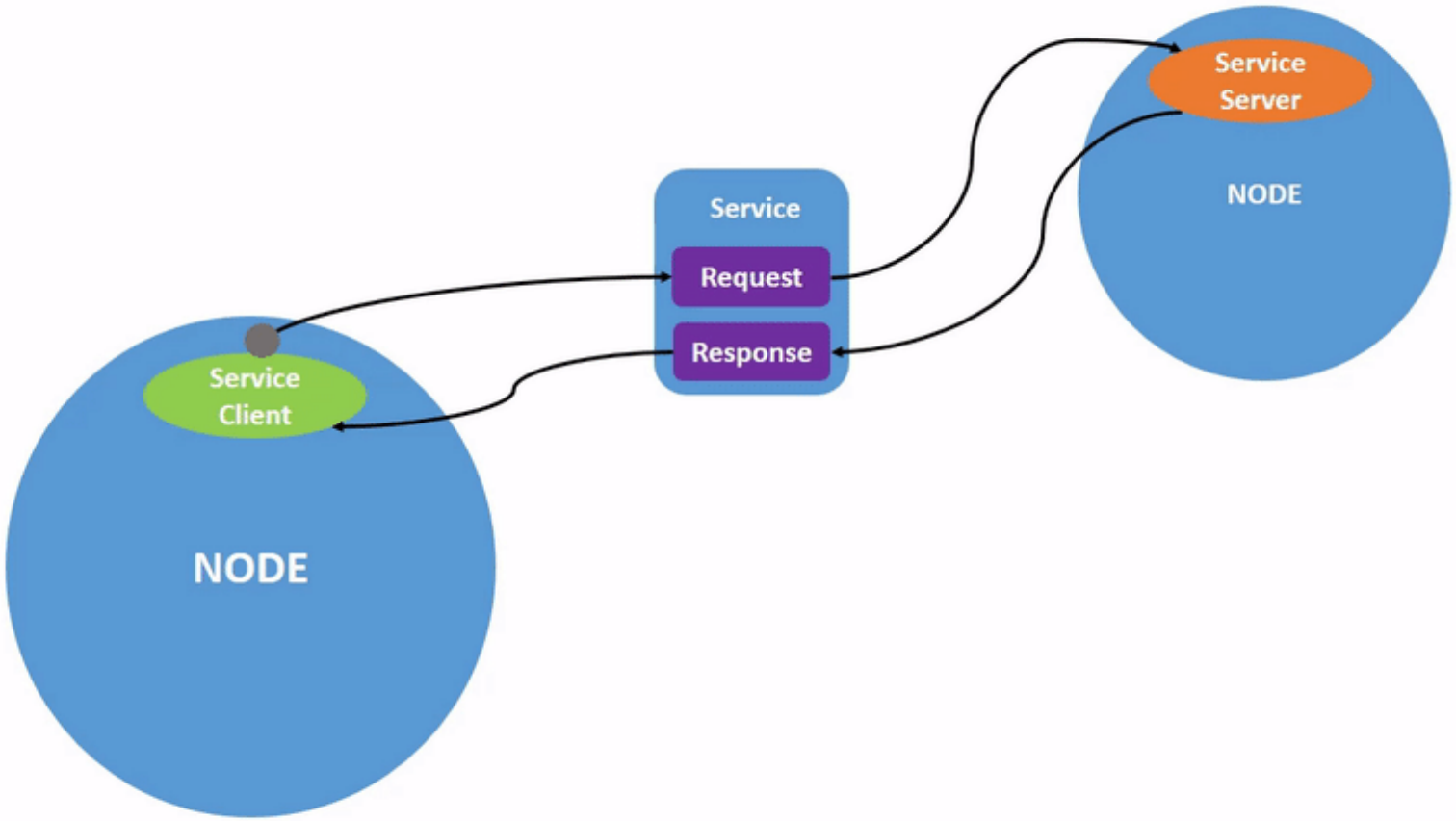
Interface


Package
Messages
Nodes
Services
Build Information
Package Description
Actions

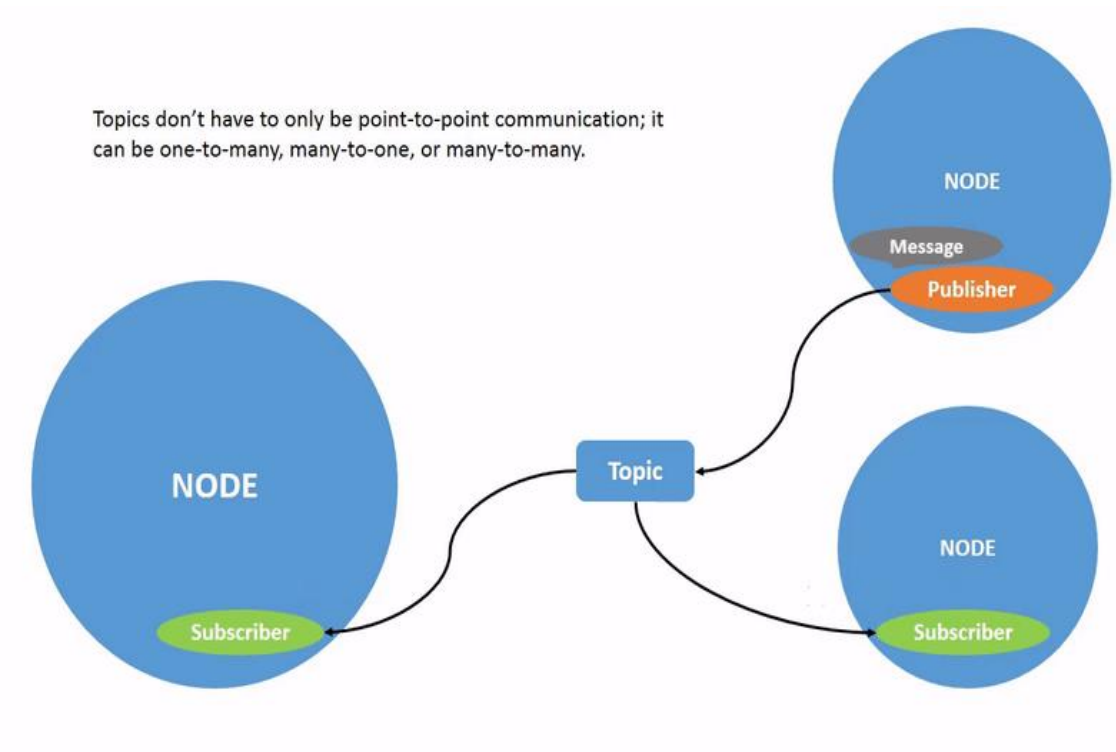
Topic

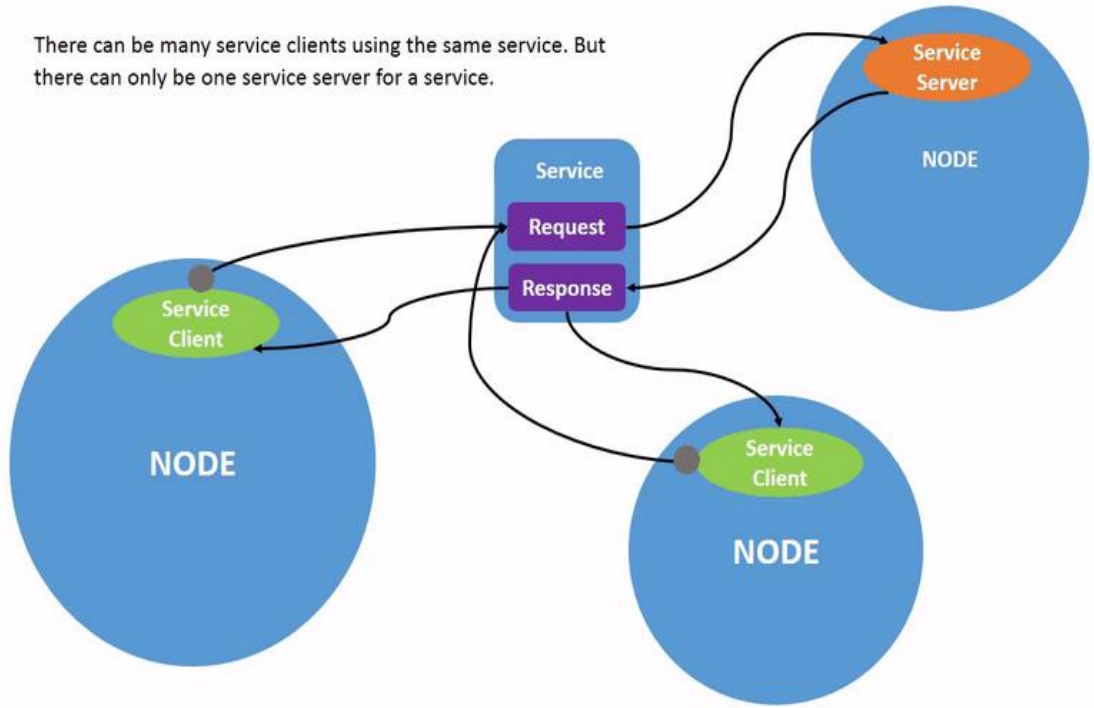


Service

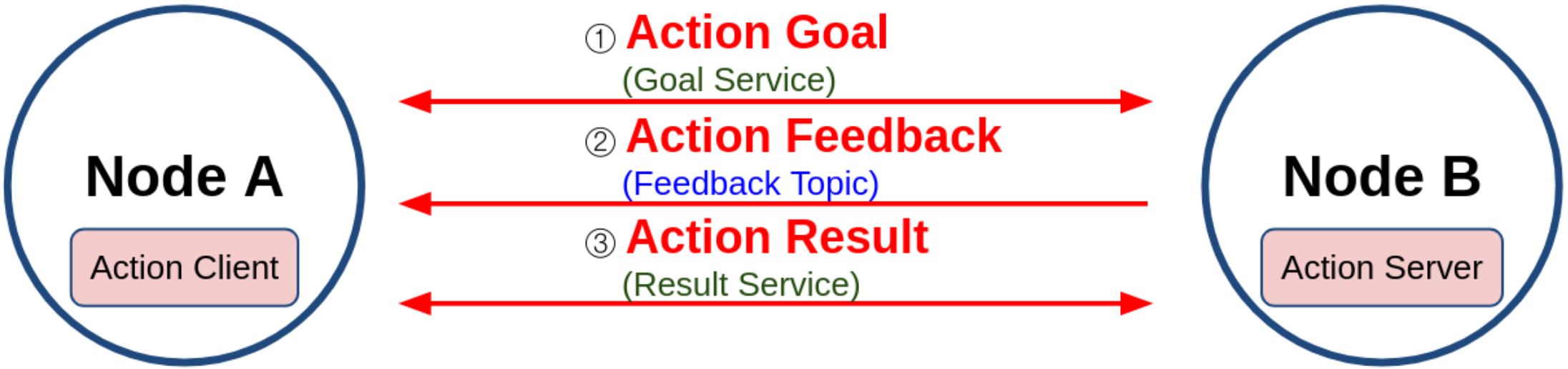

Service
Server
NODE
Service
Request
Response
Service
Client
NODE

Topic vs Service

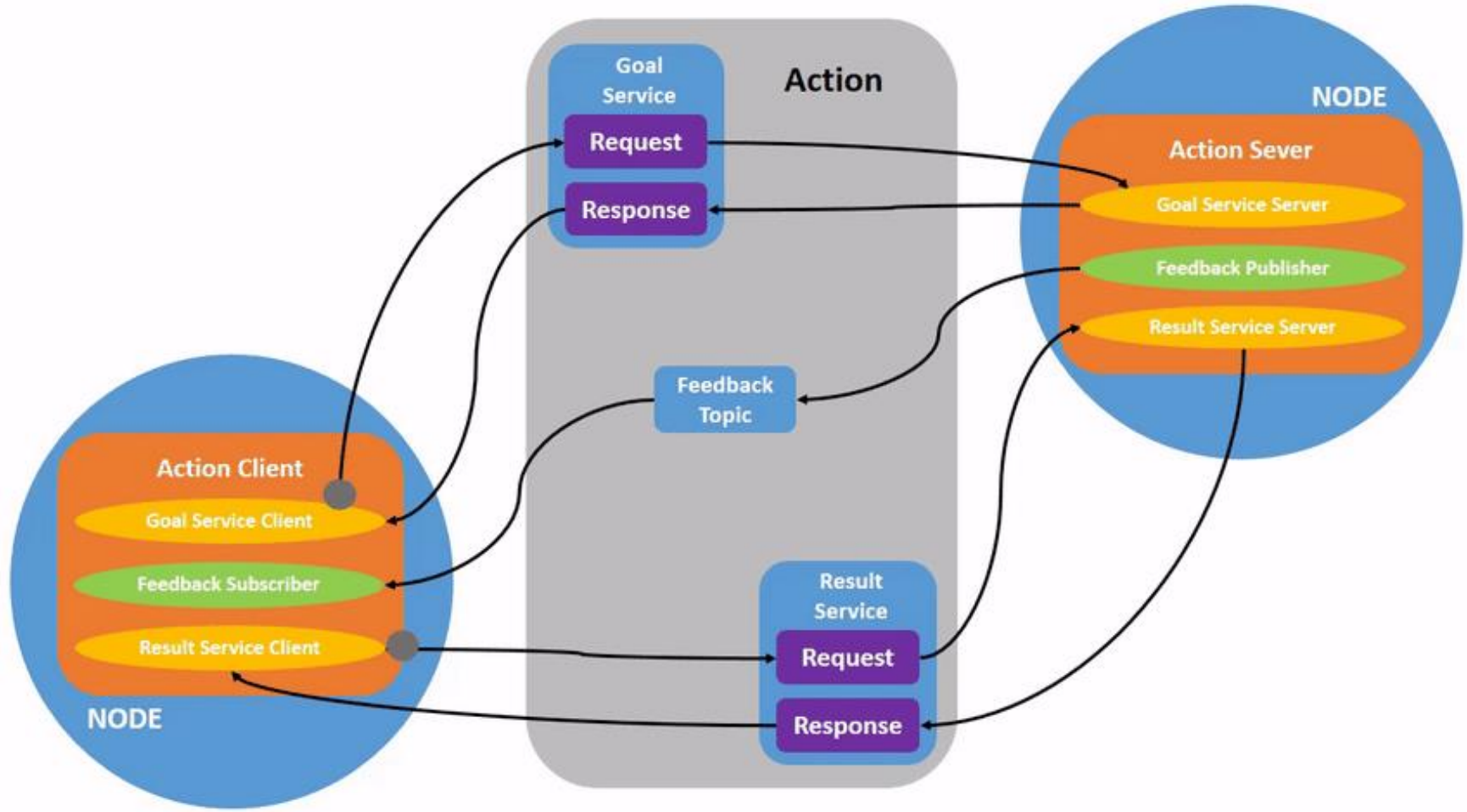




Action


Node A
Action Client
① Action Goal
(Goal Service)
② Action Feedback
(Feedback Topic)
③ Action Result
(Result Service)
Node B
Action Server

Action


Goal
Service
Request
Response
Action
NODE
Action Sever
Goal Service Server
Feedback Publisher
Result Service Server
Feedback
Topic
Action Client
Goal Service Client
Feedback Subscriber
Result Service Client
NODE
Result
Service
Request
Response

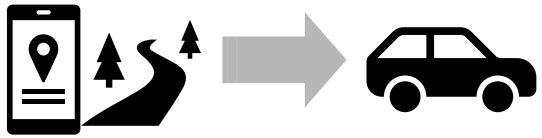
Action


(Client)

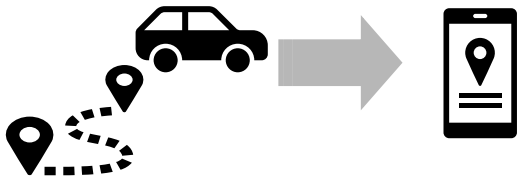

(Server)

1. Goal Request (Client → Server)

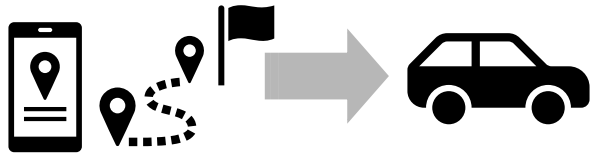
2. Goal Response (Server → Client)



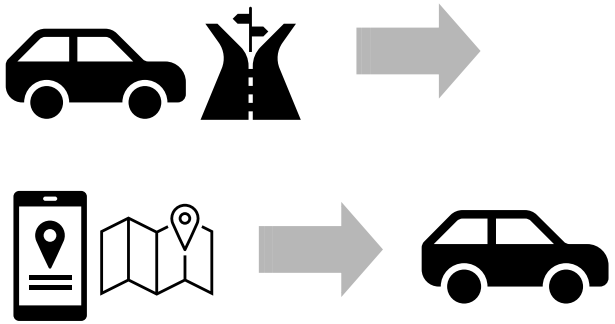
3. Result Request (Server → Client)



4. Result Response (Server → Client)



5. Feedback Topic (Server → Client)



Interface

	msg 인터페이스	srv 인터페이스	action 인터페이스
확장자	*.msg	*.srv	*.action
데이터	토픽 데이터 (data)	서비스 요청 (request) --- 서비스 응답 (response)	액션 목표 (goal) --- 액션 결과 (result) --- 액션 피드백 (feedback)
형식	fieldtype1 fieldname1 fieldtype2 fieldname2 fieldtype3 fieldname3	fieldtype1 fieldname1 fieldtype2 fieldname2 --- fieldtype3 fieldname3 fieldtype4 fieldname4	fieldtype1 fieldname1 fieldtype2 fieldname2 --- fieldtype3 fieldname3 fieldtype4 fieldname4 --- fieldtype5 fieldname5 fieldtype6 fieldname6