

1. Introduction to ROS2

1. ROS2 Overview

ROS2 is the second generation of Robot Operating System, an upgraded version of ROS1, which solves some of the problems existing in ROS1. The earliest version of ROS2 appeared Arden in 2017, and with the iteration of the version, it is constantly updated and optimized, and now there is a stable version. With ROS1 through, the Linux version and the choice of ROS2 version is also relevant, the two corresponding versions are as follows.

ROS2 version	Ubuntu version
Foxy	Ubuntu20.04
Galactic	Ubuntu20.04
Humble	Ubuntu22.04

According to your Linux version, download the corresponding ROS2 version, this product course is based on the Foxy version.

2. ROS2 Features

2.1, ROS2 fully supports three platforms

- Ubuntu
- Mac OS X
- Windows 10

2.2. Realized distributed architecture

Eliminate the Master central node to achieve distributed discovery of nodes, publish/subscribe, request/response communication.

2.3, support for real-time

2.4, using a new version of the programming language

- C++11
- Python3.5+.

2.5,using the new compilation system Ament (ROS for Catkin)

2.6, ROS1 can communicate with ROS 2 via rosbridge.

3. Differences between ROS2 and ROS1

3.1. Platform

ROS1 currently only supports the use of running in the Linux system, commonly used in Ubuntu to build the use. ROS2 can be built on Ubuntu, Windows and even embedded development boards, which is a much wider platform.

3.2 Language

- C++

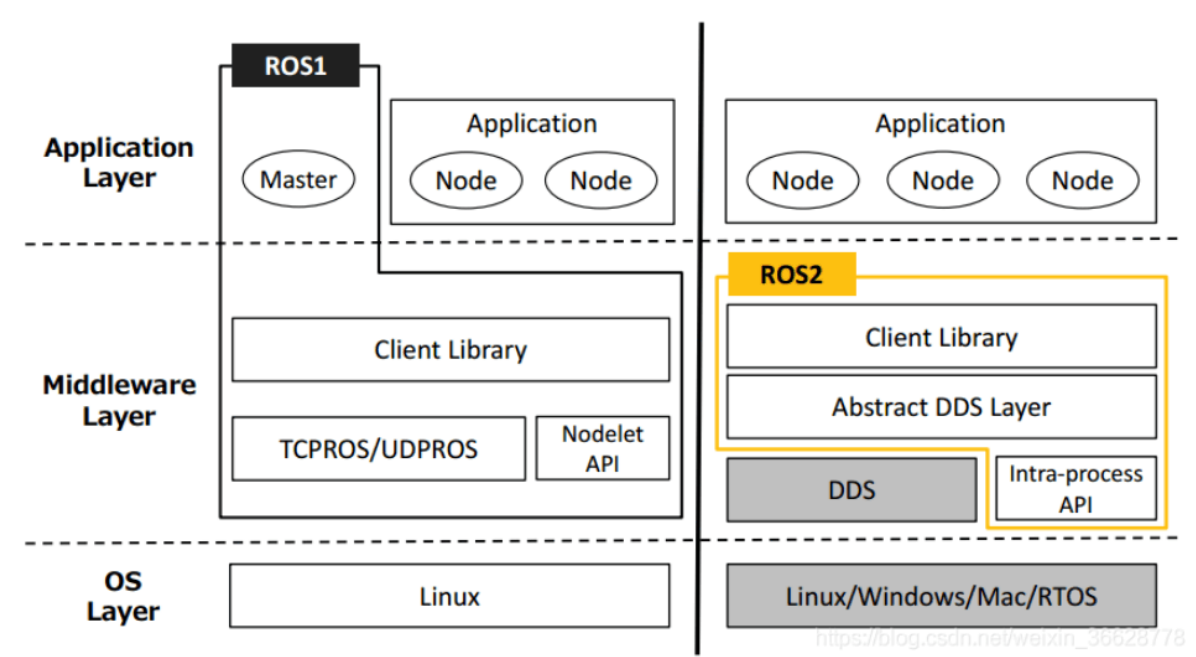
The core of ROS1 is C++03, while ROS2 uses C++11 extensively.

- Python

The version of Python used in ROS1 is Python2, while the version of Python used in ROS2 is at least 3.5, and the version of Python used in Foxy is 3.8.

3.3. Middleware

ROS1 needs to turn on roscore before starting, this master master masters all the communication between nodes, while ROS2 does not, there is only an abstract middleware interface, through which data is transferred. Currently, all implementations of this interface are based on the DDS standard. This allows ROS 2 to provide a variety of quality Qos service policies that improve communication across different networks.



3.4. Compile Commands

The compilation command for ROS1 is `catkin_make`, while the ROS2 compilation command uses the `colcon build` command.