17.ROS2 DDS

1. Introduction to DDS

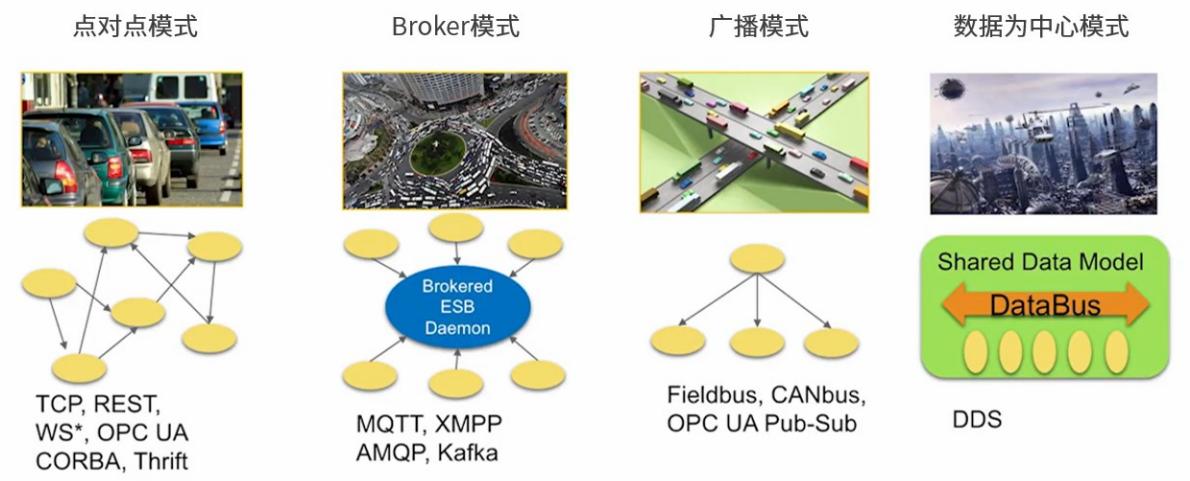
The full name of DDS is Data Distribution Service, which is Data Distribution Service. It was released and maintained by the Object Management Organization OMG in 2004. It is a set of data distribution/subscription standards specially designed for real-time systems.It was first used in the U.S. Navy to solve the compatibility issues of a large number of software upgrades in the complex network environment of ships. It has now become a mandatory standard.

DDS emphasizes being data-centric and can provide a wealth of service quality strategies to ensure real-time, efficient and flexible distribution of data, and can meet the needs of various distributed real-time communication applications.

2. Communication model

The topics, services, actions we learned in the previous courses, and the specific implementation process of their underlying communication are all completed by DDS, which is equivalent to the neural network in the ROS robot system.

The core of DDS is communication. There are many models and software frameworks that can realize communication. Here we list four commonly used models.



- The first one, **point-to-point model**, many clients connect to a server. Each time they communicate, both communicating parties must establish a connection. When communication nodes increase, the number of connections will also increase. Moreover, each client needs to know the specific address of the server and the services provided. Once the server address changes, all clients will be affected.
- The second type, **Broker model**, is optimized for the point-to-point model. The Broker centrally handles everyone's requests and further finds the role that can truly respond to the service. In this way, the client does not need to care about the specific address of the server. However, the problem is also obvious. As the core, Broker's processing speed will affect the efficiency of all nodes. When the system scale increases to a certain extent, Broker will become the performance bottleneck of the entire system. What's more troublesome is that if

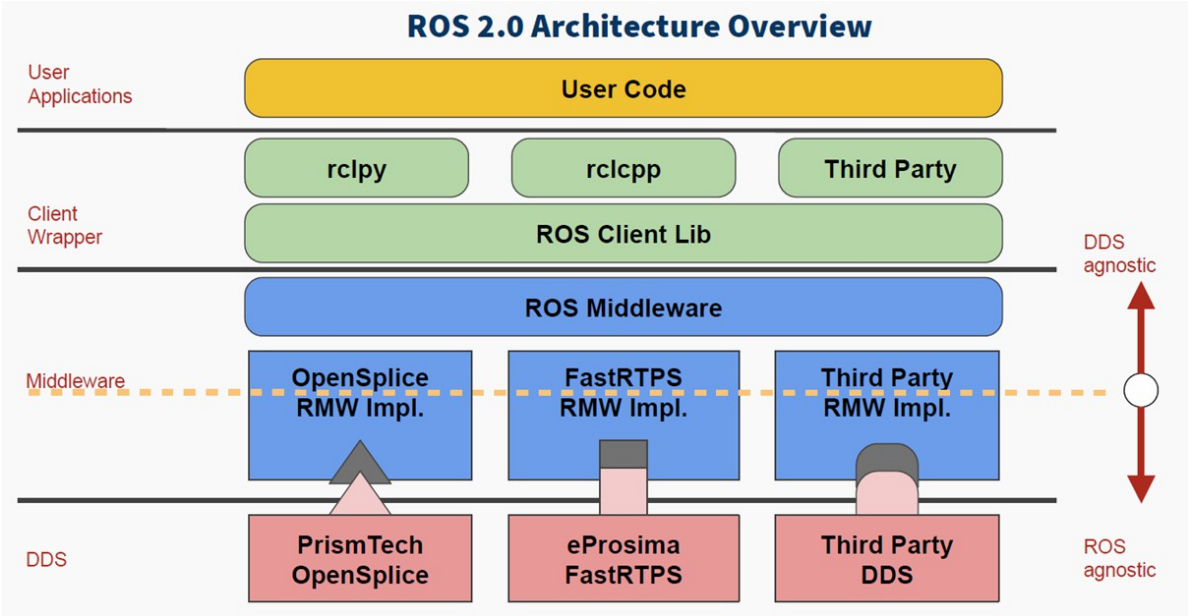
the Broker is abnormal, it may cause the entire system to fail to operate normally. The previous ROS1 system used a similar architecture.

- The third type, **broadcast model**, all nodes can broadcast messages on the channel, and all nodes can receive the messages. This model solves the problem of server addresses, and the communicating parties do not need to establish separate connections. However, there are too many messages on the broadcast channel, and all nodes must care about each message. In fact, many of them have nothing to do with themselves.
- The fourth type is the **data-centric DDS model**. This model is somewhat similar to the broadcast model. All nodes can publish and subscribe to messages on the DataBus. But its advancement is that communication contains many parallel paths. Each node can only care about the messages it is interested in and ignore the messages it is not interested in.It's a bit like a rotating hot pot. All kinds of delicious food are transmitted on this DataBus. We only need to take what we want to eat, and the rest has nothing to do with us.

It can be seen that among these communication models, the advantages of DDS are more obvious.

3. DDS application in ROS2

The position of DDS in the ROS2 system is crucial, and all upper-layer construction is built on DDS. In the architecture diagram of ROS2, the blue and red parts are DDS.



Among the four major components of ROS, the addition of DDS has greatly improved the comprehensive capabilities of the distributed communication system.In this way, when we develop robots, we don't need to worry about communication issues, and can spend more time on other parts of application development.

4. Quality service strategy QoS

The basic structure in DDS is Domain, which binds various applications together for communication.Recall that when we configured the Raspberry Pi to communicate with the computer, the DOMAIN ID configured was the group definition of the global data space. Only nodes in the same DOMAIN group can communicate with each other. This can avoid resources occupied by useless data.

Another important feature in DDS is the quality service policy: QoS.

QoS is a network transmission strategy. The application specifies the required network transmission quality behavior. The QoS service implements this behavior requirement and meets the customer's communication quality needs as much as possible.
It can be understood as a contract between the data provider and the receiver.

The strategy is as follows:

- **DEADLINE** policy means that communication data must be communicated within each deadline;
- **HISTORY** strategy, indicating a cache size for historical data;
- **RELIABILITY** policy, indicating the mode of data communication, is configured as BEST_EFFORT, which is the best-effort transmission mode.When the network condition is not good, you must also ensure data smoothness. This may cause data loss. Configure it to RELIABLE, which is the trustworthy mode.We can try to ensure the integrity of the image during communication, and we can choose the appropriate communication mode according to the application function scenario;
- **DURABILITY** strategy can be configured for nodes that join late, and it also ensures that a certain amount of historical data is sent to the system, allowing new nodes to quickly adapt to the system.