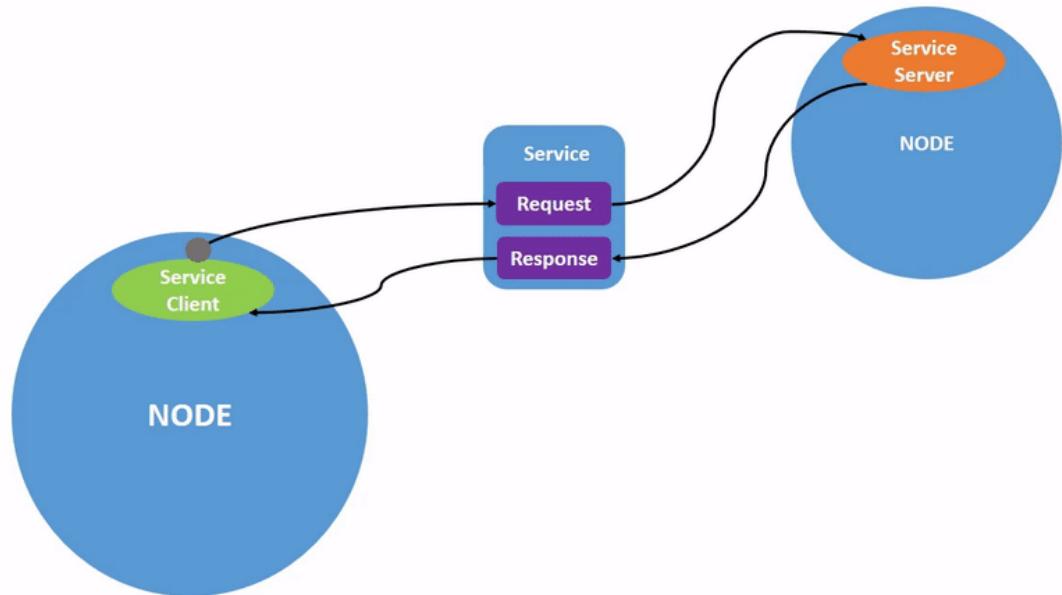


# 8、ROS2服务通讯

## 1、服务通讯简介

服务通讯是一种基于请求响应的通信模型，在通信双方中，客户端发送请求数据到服务端，服务端响应结果给客户端。

客户端/服务器模型如下：



从服务的实现机制上来看，这种你问我答的形式叫做客户端/服务器模型，简称为CS模型，客户端在需要某些数据的时候，针对某个具体的服务，发送请求信息，服务器端收到请求之后，就会进行处理并反馈应答信息。

这种通信机制在生活中也很常见，比如我们经常浏览的各种网页，此时你的电脑浏览器就是客户端，通过域名或者各种操作，向网站服务器发送请求，服务器收到之后返回需要展现的页面数据。

## 2、新建功能包

- 在工作空间src目录下：

```
ros2 pkg create pkg_service --build-type ament_python --dependencies rclpy --node-name server_demo
```

执行完上述命令，会创建pkg\_service功能包，同时会创建一个server\_demo的节点，并且已经配置好相关的配置文件

The screenshot shows a terminal window with two panes. The left pane displays the file structure of a workspace named 'YAHBOOMCAR\_ROS2\_WS' under '资源管理器'. It includes files like 'laserscan\_to\_point\_publisher.py', 'pkg\_helloworld.py', 'pkg\_service/\_init\_.py', 'server\_demo.py', 'resource.py', 'test.py', 'package.xml', 'setup.cfg', and 'setup.py'. The right pane shows the code for 'server\_demo.py':

```
def main():
    print('Hi from pkg_service.')
if __name__ == '__main__':
    main()
```

## 3、服务端实现

### 3.1 创建服务端

接下来编辑【server\_demo.py】实现服务端的功能，添加如下代码：

```
#导入相关的库文件
import rclpy
from rclpy.node import Node
from example_interfaces.srv import AddTwoInts

class Service_Server(Node):
    def __init__(self, name):
        super().__init__(name)
        #创建一个服务端，使用的是create_service函数，传入的参数分别是：
        #服务数据的数据类型、服务的名称，服务回调函数（也就是服务的内容）
        self.srv = self.create_service(AddTwoInts, '/add_two_ints',
                                      self.Add2Ints_callback)
        #这里的service回调函数的内容是把两个整型数相加，然后返回相加的结果
    def Add2Ints_callback(self, request, response):
        response.sum = request.a + request.b
        print("response.sum = ", response.sum)
        return response
    def main():
        rclpy.init()
        server_demo = Service_Server("publisher_node")
        rclpy.spin(server_demo)
        server_demo.destroy_node()                      # 销毁节点对象
        rclpy.shutdown()                                # 关闭ROS2 Python接口
```

重点看下服务回调函数，`Add2Ints_callback`，这里需要传入的参数除了`self`，还有就是`request`和`response`，`request`是服务需要的参数，`response`是服务的反馈结果。`request.a`和`request.b`是`request`部分的内容，`response.sum`是`response`部分的内容，这里首先看看下`AddTwoInts`这个类型的数据是怎么样的。

- 可以使用以下命令查看

```
ros2 interface show example_interfaces/srv/AddTwoInts
```

```
root@unbutu:~/yahboomcar_ross2_ws/yahboomcar_ross2_ws# ros2 interface show example_interfaces/srv/AddTwoInts
int64 a
int64 b
---
int64 sum
```

“---”把该类型的数据划分成了两个部分，上边代表的是request，下边代表的是response。然后各自的领域中又各自的变量，比如int64 a、int64 b，所有在再传入参数的是，需要指定a、b的值是多少。同样，反馈的结果也需要指定sum的值是多少。

### 3.2 编辑配置文件

- 打开setup.py,在console\_scripts列表中添加命令

```
'server_demo = pkg_service.server_demo:main'
```

```
资源管理器 ... + setup.py U X
YAHBOOMCAR_ROS2_WS [容器 192.168.2.51:5000/ROS... yahboomcar_ws > src > pkg_service > setup.py > ...
> log
+ src
> laserscan_to_point_publisher
> pkg_helloworld_py
+ pkg_service
+ pkg_service
+ _init_.py
+ server_demo.py
> resource
> test
package.xml
setup.cfg
+ setup.py
> pkg_topic
> rf2o_laser_odometry
> robot_pose_publisher_ros2
> yahboom_app_save_map
> yahboom_web_savmap_interfaces
> yahboomcar_astra
> yahboomcar_bringup
> yahboomcar_ctrl
> yahboomcar_description
> yahboomcar_kcftracker
> yahboomcar_laser
> yahboomcar_linefollow
> yahboomcar mediapipe

yahboomcar_ws > src > pkg_service > setup.py
1  from setuptools import setup
2
3  package_name = 'pkg_service'
4
5  setup(
6      name=package_name,
7      version='0.0.0',
8      packages=[package_name],
9      data_files=[
10          ('share/ament_index/resource_index/packages',
11              ['resource/' + package_name]),
12          ('share/' + package_name, ['package.xml']),
13      ],
14      install_requires=['setuptools'],
15      zip_safe=True,
16      maintainer='root',
17      maintainer_email='1461190907@qq.com',
18      description='TODO: Package description',
19      license='TODO: License declaration',
20      tests_require=['pytest'],
21      entry_points={
22          'console_scripts': [
23              'server_demo = pkg_service.server_demo:main'
24          ],
25      },
26  )
```

### 3.3 编译功能包

- 编译功能包

```
colcon build --packages-select pkg_service
```

```
'colcon build' successful  
/home/yahboom/workspace  
  
yahboom@yahboom-virtual-machine: ~/workspace  
yahboom@yahboom-virtual-machine: ~/workspace 95x24  
yahboom@yahboom-virtual-machine:~/workspace$ colcon build --packages-select pkg_service  
Starting >>> pkg_service  
/usr/lib/python3/dist-packages/setuptools/dist.py:723: UserWarning: Usage of dash-separated 'install-scripts' will not be supported in future versions. Please use the underscore name 'install_scripts' instead  
    warnings.warn(  
Finished <<< pkg_service [0.67s]  
  
Summary: 1 package finished [0.87s]  
yahboom@yahboom-virtual-machine:~/workspace$
```

## 3.4 运行程序

- 刷新环境变量然后运行节点

```
ros2 run pkg_service server_demo
```

```
yahboom@yahboom-virtual-machine: ~/workspace  
yahboom@yahboom-virtual-machine: ~/workspace 95x24  
yahboom@yahboom-virtual-machine:~/workspace$ source ./install/setup.bash  
yahboom@yahboom-virtual-machine:~/workspace$ ros2 run pkg_service server_demo
```

运行后，由于没有调用该服务，所以没有反馈数据，可以通过命令行方式调用该服务，首先查询当前有哪些服务，另一个终端输入：

```
ros2 service list
```

```
root@unbutu:~/yahboomcar_ros2_ws/yahboomcar_ws# ros2 service list  
/add_two_ints  
/publisher_node/describe_parameters  
/publisher_node/get_parameter_types  
/publisher_node/get_parameters  
/publisher_node/list_parameters  
/publisher_node/set_parameters  
/publisher_node/set_parameters_atomically
```

/add\_two\_ints就是我们需要调用的服务，通过以下命令进行调用，终端输入：

```
ros2 service call /add_two_ints example_interfaces/srv/AddTwoInts "{a: 1,b: 4}"
```

这里我们把a的值赋值成1， b的值赋值成4，也就是调用服务计算1和4的和：

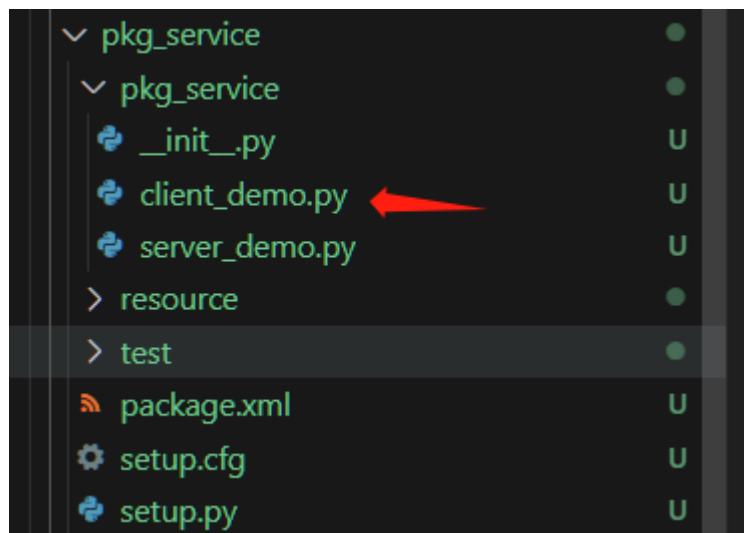
```
root@unbutu:~/yahboomcar_ros2_ws/yahboomcar_ws# ros2 service call /add_two_ints example_interfaces/srv/AddTwoInts "{a: 1,b: 4}"  
requester: making request: example_interfaces.srv.AddTwoInts_Request(a=1, b=4)  
response:  
example_interfaces.srv.AddTwoInts_Response(sum=5)
```

由上图可以看出，调用了服务后，反馈回来的结果是5，运行服务端的终端也打印了反馈的值。

## 4、客户端实现

## 4.1 创建客户端

在【server\_demo.py】同级目录下新建文件【client\_demo.py】



接下来编辑【client\_demo.py】实现客户端的功能，添加如下代码：

```
#导入相关的库
import rclpy
from rclpy.node import Node
from example_interfaces.srv import AddTwoInts

class Service_Client(Node):
    def __init__(self, name):
        super().__init__(name)
        #创建客户端，使用的是create_client函数，传入的参数是服务数据的数据类型、服务的话题名称
        self.client = self.create_client(AddTwoInts, '/add_two_ints')
        # 循环等待服务器端成功启动
        while not self.client.wait_for_service(timeout_sec=1.0):
            print("service not available, waiting again...")
        # 创建服务请求的数据对象
        self.request = AddTwoInts.Request()

    def send_request(self):
        self.request.a = 10
        self.request.b = 90
        #发送服务请求
        self.future = self.client.call_async(self.request)

def main():
    rclpy.init() #节点初始化
    service_client = Service_Client("client_node") #创建对象
    service_client.send_request() #发送服务请求
    while rclpy.ok():
        rclpy.spin_once(service_client)
        #判断数据是否处理完成
        if service_client.future.done():
            try:
                #获得服务反馈的信息并且打印
                response = service_client.future.result()
                print("service_client.request.a = ", service_client.request.a)
                print("service_client.request.b = ", service_client.request.b)
            except Exception as e:
                print(e)

if __name__ == '__main__':
    main()
```

```

        print("Result = ", response.sum)
    except Exception as e:
        service_client.get_logger().info('Service call failed %r' %
(e,))
        break
    service_client.destroy_node()
    rclpy.shutdown()

```

## 4.2 编辑配置文件

- 打开setup.py,在console\_scripts列表中添加

```
'client_demo = pkg_service.client_demo:main'
```

The screenshot shows a terminal window with two panes. The left pane is a file browser titled '资源管理器' showing the directory structure of 'YAHBOOMCAR\_ROS2\_WS'. The right pane is a code editor titled 'setup.py' showing the Python code for the package setup. The code defines a setup function with various parameters like name, version, packages, data\_files, install\_requires, zip\_safe, maintainer, and maintainer\_email. It also includes an 'entry\_points' section with a 'console\_scripts' key containing two entries: 'server\_demo = pkg\_service.server\_demo:main' and 'client\_demo = pkg\_service.client\_demo:main'. A red arrow points to the 'client\_demo' entry.

```

资源管理器
...
YAHBOOMCAR_ROS2_WS [容器 192.168.2.51:5000/ROS2]
    > log
    < src
        > laserscan_to_point_publisher
        > pkg_helloworld_py
        < pkg_service
            < pkg_service
                > __init__.py
                > client_demo.py
                > server_demo.py
            > resource
            > test
            > package.xml
            > setup.cfg
            < setup.py
        > pkg_topic
        > rf2o_laser_odometry
        > robot_pose_publisher_ros2
        > yahboom_app_save_map
        > yahboom_web_savmap_interfaces
        > yahboomcar_astra
        > yahboomcar_bringup
        > yahboomcar_ctrl
        > yahboomcar_description
        > yahboomcar_kcftracker
        > yahboomcar_laser
        > yahboomcar_linefollow
        > yahboomcar_mediapipe
    > yahboomcar_ws > src > pkg_service > setup.py > ...
yahboomcar_ws > src > pkg_service > setup.py > ...
1   from setuptools import setup
2
3   package_name = 'pkg_service'
4
5   setup(
6       name=package_name,
7       version='0.0.0',
8       packages=[package_name],
9       data_files=[
10          ('share/ament_index/resource_index/packages',
11             ['resource/' + package_name]),
12          ('share/' + package_name, ['package.xml']),
13      ],
14       install_requires=['setuptools'],
15       zip_safe=True,
16       maintainer='root',
17       maintainer_email='1461190907@qq.com',
18       description='TODO: Package description',
19       license='TODO: License declaration',
20       tests_require=['pytest'],
21       entry_points={
22           'console_scripts': [
23               'server_demo = pkg_service.server_demo:main',
24               'client_demo = pkg_service.client_demo:main'
25           ],
26       },
27   )

```

## 4.3 编译功能包

```
colcon build --packages-select pkg_service
```

```
'colcon build' successful
/home/yahboom/workspace

yahboom@yahboom-virtual-machine: ~/workspace
yahboom@yahboom-virtual-machine: ~/workspace 95x24
yahboom@yahboom-virtual-machine:~/workspace$ colcon build --packages-select pkg_service
Starting >>> pkg_service
/usr/lib/python3/dist-packages/setuptools/dist.py:723: UserWarning: Usage of dash-separated 'install-scripts' will not be supported in future versions. Please use the underscore name 'install_scripts' instead
  warnings.warn(
Finished <<< pkg_service [0.67s]

Summary: 1 package finished [0.87s]
yahboom@yahboom-virtual-machine:~/workspace$
```

## 4.4 运行程序

- 刷新环境变量然后运行节点

```
#启动服务端节点
ros2 run pkg_service server_demo
#启动客户端节点
ros2 run pkg_service client_demo
```

```
yahboom@yahboom-virtual-machine: ~/workspace 53x30
yahboom@yahboom-virtual-machine:~/workspace$ ros2 run
pkg_service server_demo
response.sum = 100
[yahboom@yahboom-virtual-machine: ~/workspace 54x30
yahboom@yahboom-virtual-machine:~/workspace$ ros2 run
pkg_service client_demo
service_client.request.a = 10
service_client.request.b = 90
Result = 100
yahboom@yahboom-virtual-machine:~/workspace$
```

先运行服务端，然后运行客户端，客户端提供 $a=10$ ,  $b=90$ ，服务端进行求和，得到结果是100，结果在两者终端打印。