

Server创建流程

1. 创建节点

```
1 | rospy.init_node(nodeName)
```

2. 创建server

```
1 | rospy.Service(serviceName, AddTwoInts, callback)
```

!!!tip

第一个参数 `serviceName` 为服务名称，为一个uri地址

```
1 | 第二个参数是服务需要的数据类型。
2 |
3 | 第三个参数为服务请求的回调
```

3.处理请求的逻辑

```
1 | def callback(request):
2 |     if not isinstance(request, AddTwoIntsRequest):
3 |         return
4 |
5 |     # 获取请求数据
6 |     a = request.a
7 |     b = request.b
8 |
9 |     # 返回响应结果
10 |    response = AddTwoIntsResponse()
11 |    response.sum = a + b
12 |    return response
```

!!!tip

回调函数的参数是请求过来的数据

```
1 | 返回值是响应的数据
```

完整示例代码

```
1 | #!/usr/bin/env python
2 | # coding:utf-8
3 |
4 | import rospy
5 | from rospy_tutorials.srv import AddTwoInts, AddTwoIntsRequest,
6 |    AddTwoIntsResponse
7 |
```

```

8  def callback(request):
9      if not isinstance(request, AddTwoIntsRequest):
10         return
11
12     # 获取请求数据
13     a = request.a
14     b = request.b
15
16     # 返回响应结果
17     response = AddTwoIntsResponse()
18     response.sum = a + b
19     return response
20
21 if __name__ == '__main__':
22     # 创建节点
23     nodeName = "my_server_node"
24     rospy.init_node(nodeName)
25
26     # 创建Service Server
27     serviceName = "my_service"
28     rospy.Service(serviceName, AddTwoInts, callback)
29
30     # 阻塞线程
31     rospy.spin()

```

调试Server

调试 server端 主要是查看 server端 是否能接收到请求，并根据请求数据处理相应的业务逻辑，然后返回处理好的结果。

在这里，我们只需要模拟 client端 发送请求就可以了。

ROS提供了命令行工具和图形化工具供我们调试开发。

1. rosservice命令行调试

通过 `rosservice list` 命令可以帮助我们查询出当前运行的所有service

```
1 | rosservice list
```

查询的结果中，我们可以得到对应的服务名称 `/demo_service/add_two_int`

通过查询的服务名称，来调用此服务

```
1 | rosservice call /my_service "a:1 b:3"
```

!!!tip

`rosservice call` 负责调用service。第一个参数是要调用的service的名称，后面的参数是调用时需要传入的参数。

2.rqt_service_caller工具调试

通过命令呼出工具

```
1 | rosrn rqt_service_caller rqt_service_caller
```

The screenshot shows the `rqt_service_caller` tool interface. The title bar is `rqt_service_caller__ServiceCaller - rqt`. The main window is divided into two sections: **Request** and **Response**.

Request Section:

Topic	Type	Expression
▼ /my_service	rospy_tutorials/AddTwoIntsRequest	
a	int64	4
b	int64	6

Response Section:

Field	Type	Value
▼ /	rospy_tutorials/AddTwoIntsResponse	
sum	int64	10

Red arrows and text labels indicate the steps:

- 步骤1:选中服务名称 (Step 1: Select service name) - points to the `/my_service` dropdown.
- 步骤2:为请求添加数据 (Step 2: Add data to request) - points to the `4` and `6` values in the `Expression` column.
- 步骤3:调用服务 (Step 3: Call service) - points to the `Call` button.
- 步骤4:查看调用结果 (Step 4: Check call result) - points to the `10` value in the `Value` column.

黑马程序员