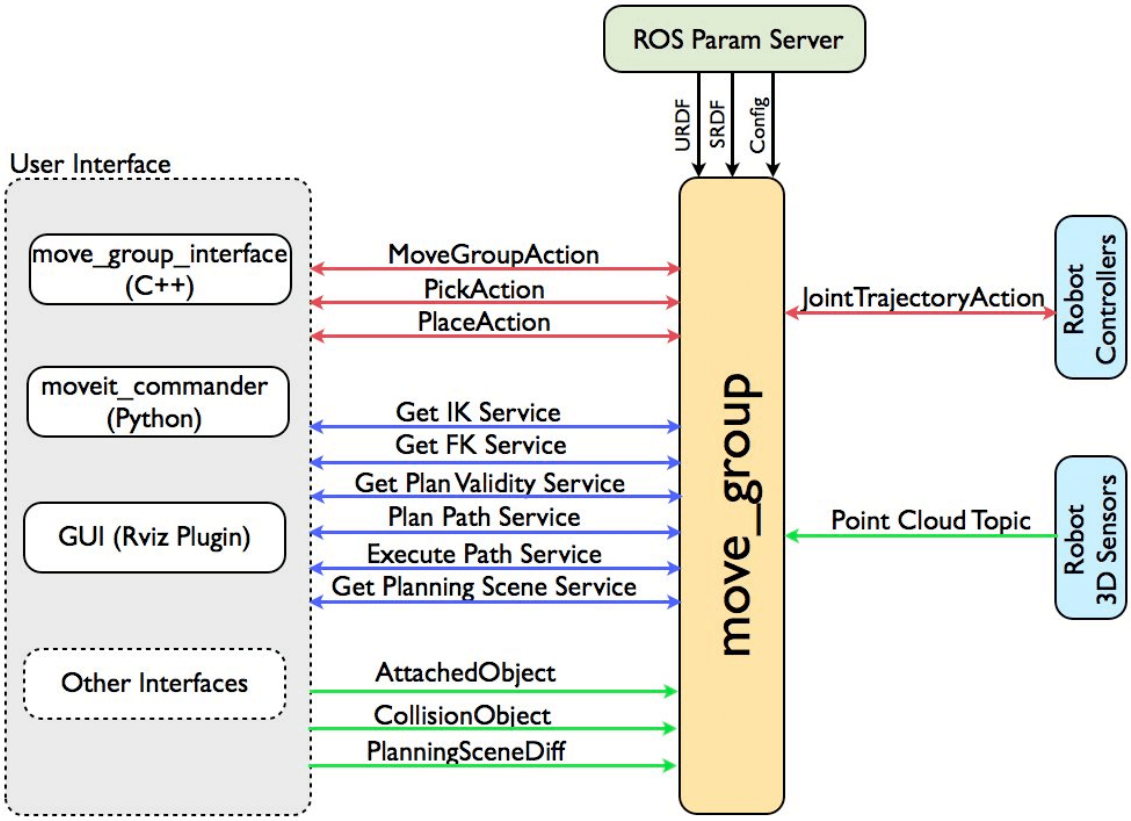
MoveIt

1 Introduction to MoveIt

MoveIt is an integrated development platform in ROS, which consists of a variety of functional packages for manipulating robot arms, including motion planning, operation, control, inverse kinematics, 3D perception, collision detection, etc.

The following figure shows the high-level structure of the main node **move_group** provided by Moveit. It is like a combiner: all the individual components are integrated together, providing a series of actions and services for users to use.



2 User interface

The user may access the operations and services provided by move_group in three ways:

- In C++, you may use move_group easily by using move_group_interface package.
- In Python, use the moveit_commander package.
- Via GUI: use Rviz (ROS visualization tool) of Motion-commander.

move_group can be configured using the ROS parameter server, from which the robot's URDF and SRDF can also be obtained.

3 Configuration

move_group is a ROS node. It uses the ROS parameter server to obtain three kinds of information:

- URDF - move_group looks for the robot_description parameter in the ROS parameter server to get the robot's URDF.

- SRDF - move_group looks for the robot_description_semantic parameter in the ROS parameter server to get the robot's SRDF. SRDF is typically created by the user using an MoveIt Setup Assistant.
- MoveIt configuration - move_group will look in the ROS parameter server for additional MoveIt-specific configurations, including joint constraint, kinematics, motion planning, perception, and other information. w The configuration files for these components are automatically generated by the MoveIt Setup Assistant and stored in the configuration directory of the robot's corresponding MoveIt configuration package.

Using MyCobot

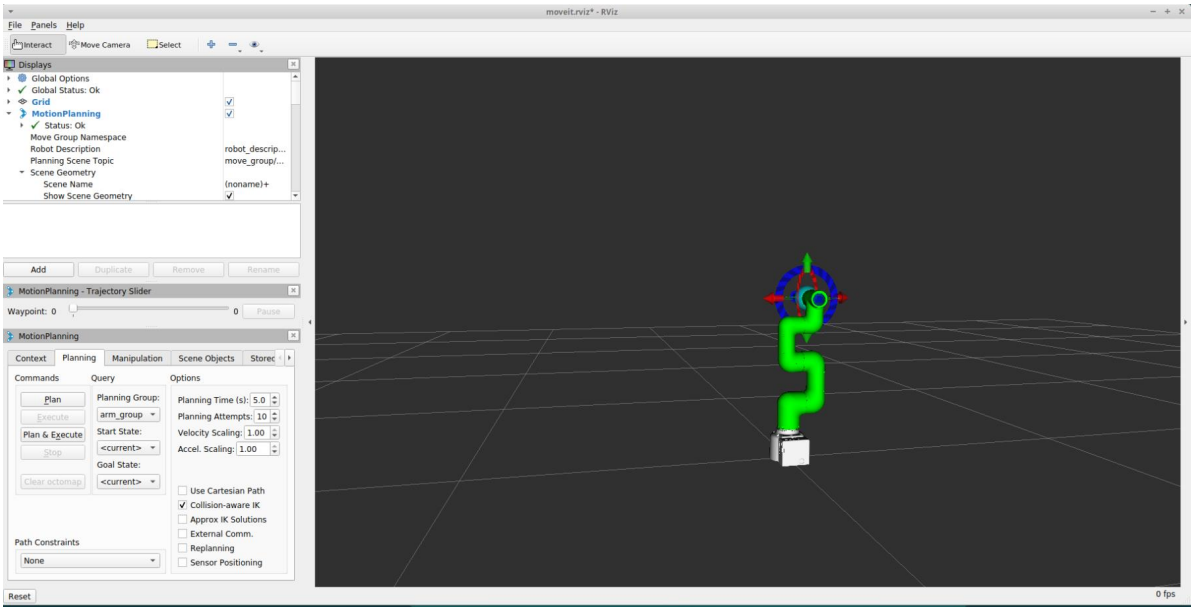
`mycobot_ros` has integrated the MoveIt section.

Open the command line and run:

- 2022 mycobot 320-M5 version:

```
roslaunch new_mycobot_320_moveit mycobot320_moveit.launch
```

The operaion effect is as follows:



If you want a real robot arm to execute a plan synchronously, you need to open another command line and run:

- 2022 mycobot 320-M5 version:

```
# The default serial port name of 2022 mycobot 320-M5 version is "/dev/ttyUSB0",
and the baud rate is 115200. The serial port name of some models is
"dev/ttyACM0". If the default serial port name is wrong, you can change the
serial port name to "/dev/ttyACM0".
rosrun new_mycobot_320_moveit sync_plan.py _port:=/dev/ttyUSB0 _baud:=115200
```