KBSW210714 ROS - set_initial_pose

set_initial_pose, rviz2D Pose Estimate

-
  -
  -
-
-
-

-
  - Ubuntu 16.04 X86
  - ROS Kinetic
-
  -
    - Slamware Slamware
    - Apollo/Ares/Athena

ROS-

1.Slamware ROS SDKROS

2.ROS sdksrccatkin_wsROS slamware_ros_samplesrccatkin

```
cd catkin_ws/src
catkin_init_workspace
```

3.

```
cd ..
catkin_make
```

4.

```
source devel/setup.bash
```

5.slamware_ros_sdk_server_node

```
roslaunch slamware_ros_sdk slamware_ros_sdk_server_node.launch ip_address:=10.6.128.141
//APip_address192.168.11.1
```

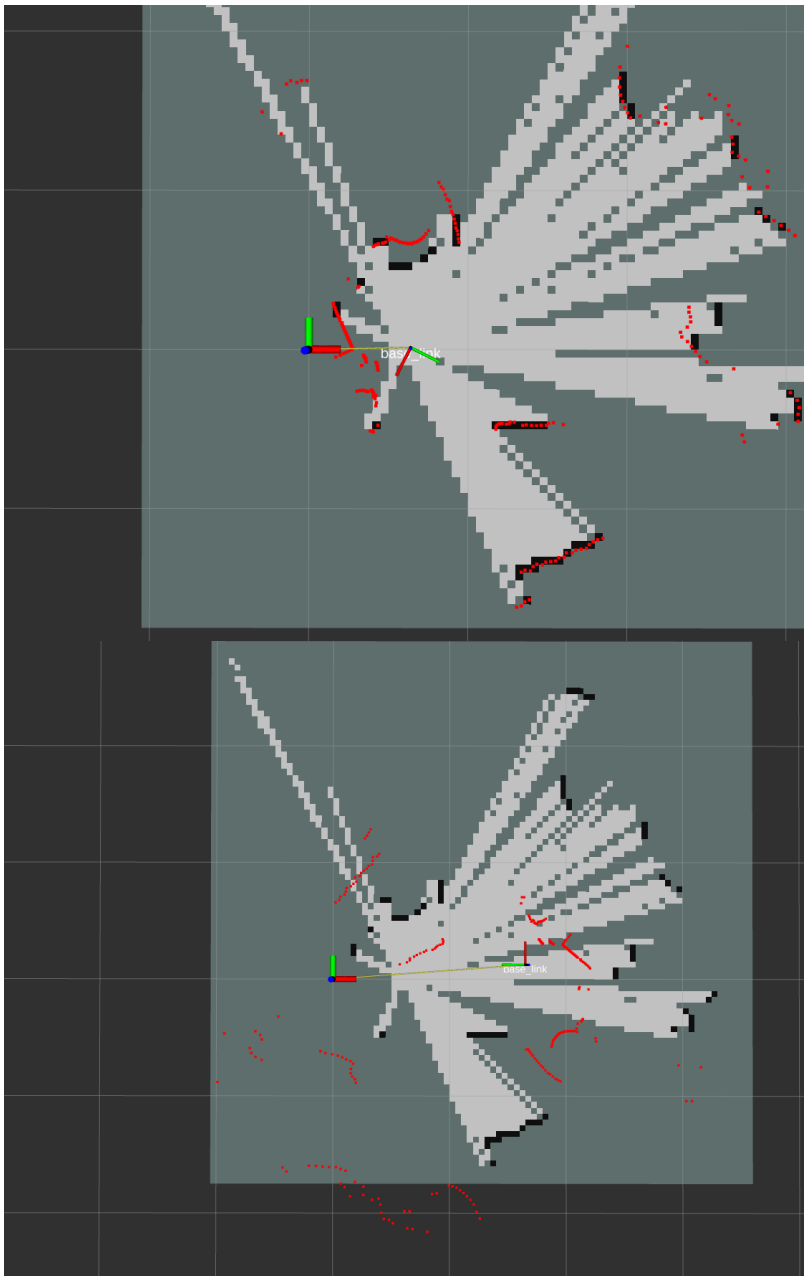
6.4view_slamware_ros_sdk_server_node

```
roslaunch slamware_ros_sdk view_slamware_ros_sdk_server_node.launch
```

7.4set_initial_pose_node

```
roslaunch slamware_ros_sample set_initial_pose.launch
```

8.Rviz2D Pose Estimate



Publisher
`ros::Publisher set_pose_pub;`

publishersubcriber

```
ros::init(argc, argv, "set_initial_pose_node");
ros::NodeHandle nh("~");
set_pose_pub = nh.advertise<geometry_msgs::Pose>("/slamware_ros_sdk_server_node/set_pose", 10);
ros::Subscriber initial_pose_sub = nh.subscribe("/initialpose", 10, initialPoseCallback);
```

initialpose

```
void initialPoseCallback(const geometry_msgs::PoseWithCovarianceStamped& msg)
{
  ROS_INFO("receive initialpose: ");
  ROS_INFO("position(xyz): %.3f %.3f %.3f", msg.pose.pose.position.x
                                          , msg.pose.pose.position.y
                                          , msg.pose.pose.position.z);
  ROS_INFO("orientation(xyzw): %.3f %.3f %.3f %.3f", msg.pose.pose.orientation.x
                                                   , msg.pose.pose.orientation.y
                                                   , msg.pose.pose.orientation.z
                                                   , msg.pose.pose.orientation.w);
  geometry_msgs::Pose _pose_msg(msg.pose.pose);
  set_pose_pub.publish(_pose_msg);
}
```