

ROS Crash Course

Class 4

 ROS

Agenda

- How to write a publisher and subscriber node
- How to add them to the CMakeLists

How to write a publisher and subscriber node

- Go to the ros class page and download and copy and paste the code from the document files into a cpp file within your package src folder.
- Make sure to name them publisher and subscriber respectively

How to write a publisher and subscriber node

- `ros::init(argc, argv, "talker")`

-`ros::init()` is used to initialize (start) the ros node and give it any arguments from the command line and also name it.

How to write a publisher and subscriber node

-ros::NodeHandle n;

-ros::NodeHandle is the name of the object used for communication within ros

-It is shown here just to point it towards something more managable

How to write a publisher and subscriber node

```
-ros::Publisher chatter_pub = n.advertise<std_msgs::String>("chatter",1000);
```

-This is where the publisher is defined, what topic it's going to, and what type of msg is going to go on it.

-The second variable is the buffer amount

How to write a publisher and subscriber node

-`ros::Rate loop_rate(10);`

-Setting the how many times the code is run per second

How to write a publisher and subscriber node

-ros::ok

-will return true if ros is running, false if it isn't

How to write a publisher and subscriber node

-std_msgs::String msg;

-This is the message type, we're just making it more manageable in code

How to write a publisher and subscriber node

-`chatter_pub.publish(msg);`

-telling the publisher which message it is going to publish to it's topic

How to write a publisher and subscriber node

-`ros::spinOnce();`

-This is telling the ros functions to run once, then cycle back, otherwise `ros::spin();` will cause only the ros functions to run and ignore the rest of the code

How to write a publisher and subscriber node

```
-ros::Subscriber sub = n.subscribe("chatter",1000, chatterCallback)
```

-This is calling the function chatterCallback everytime a msg is sent over the "chatter" topic

How to add them to the CMakeLists

`-add_executable(talker src/talker.cpp)`

`-target_link_libraries(talker ${catkin_LIBRARIES})`

`-add_dependencies(talker [name of package]_generate_messages_cpp)`

-Make sure that the `add_executable()` function is above the other two otherwise it won't know which executable to link it to

New HW

- Make a node that allows you to control different turtles within the turtlesim.

 - Use rosservice call spawn to create 3 turtles total

 - Then using your node be able to switch between all three turtles and controlling each of them.

 - when the node starts it asks the user which turtle you want to control and you can control it. Then you can switch which turtle you control without closing the node.