Project 3 – Group E Report

Description

For this project, we used two nodes: (1) vitals_node which collects all the information (wifi, battery, bumps/wheel drops) and publishes each to its own topic , and (2) monitor_node which deciphers all this information and prints it to ROS_INFO_STREAM.

We first got everything setup with the netbook, router, and workstation which all went pretty smoothly. Our next step was to find and publish the wifi strength from the netbook. We found the 'proc/net/wireless' file and parsed it in order to determine the wifi strength. We then published this info as a std_msgs::Int32 to topic "/wifi_ss". We also found the battery charge and capacity in the 'createNode/TurtlebotSensorState' message posted on topic '/turtle_node/sensor_state'. Dividing the charge by the capacity and multiplying by 100 gave us the percentage, which we published to the "/battery_percentage" topic.

Our next step was to determine where we listen for warning and error messages. We looked into available topics and found that "/diagnostics" and "/diagnostics_agg" had the information we were looking for. We had to take the DiagnosticsArray object and iterate through the DiagnosticStatus objects it contained. From these statuses, we check the level to see if they are a warning(1) or an error(2).

After getting the diagnostics working, we had to figure out how to listen to the wheeldrops and bump events. This step took us the longest to figure out. Eventually, we found the 'bumps_wheeldrops' byte in the same message as the battery charge/capacity. We decided to echo this topic while dropping the wheels and pressing the bumper to decipher the meaning of the byte. From this we outlined the meanings of the different bumps_wheeldrops values. Then in our monitor_node we wrote a switch case statement to handle the different values and print the appropriate output as soon as its received.

Evaluation

We believe that there was an issue with some of the ROS packages on the computers in the "Robotics Lab". It seemed like each computer had a different installation of rviz and it was a coin flip to see if all the necessary ROS packages were installed. The point is that we got everything to work on various computers, but we could not get everything to work on one computer. This convinced us that our program works just fine and the issues lie in the ROS configuration.

Allocation of Effort

Our group met 4 times and collectively worked through each part of the project. The only exception to this is that Davis installed Ubuntu on the netbook before our first meeting.