ROSProcessingjs NXT

Evan Stites-Clayton
Goals

• Make programming NXT easier
• Allow for coding of Lego Mindstorms NXT brick via Processing language
• Create an easy environment for coding and display of an output program
• Make it as easy as possible to recreate this setup
Tools

- Jihoon’s processing – ros js platform
- Rosjs (Websockets)
- Ros
- Processingjs
- Nxt_ros
- Nxt_python
- Rospy

Draw on board how they fit together
How It Works:

ROSProcessing\/js Interface

User Processing Code

Websocket

Internet

ROSBridge

Websocket

ROSCore

Websocket

ROSpy

NXT-Python

ROSProcessing\(\text{\textregistered}\)NXT
Processing Examples

• WASD Driver
  – Simple one way communication
• Enclosure Escape
  – Autonomous, 2way communication
  – buggy
What it takes: Roscore
What it takes: Rosjs
What it takes: Special Chrome
What it takes: nxt_python

User doesn’t need to modify the python file, but needs to run it

```
File "~/home/esc/nxtros/nxt/nxt/nxt_python/src/nxt/telegram.py",
    return unpack('<B', self.pkt.read(1))[0]
struct.error: unpack requires a string argument of length 1
esc@ubuntu:~$ rosrun nxt_python touch_sensor_test.py
This is the touch sensor test. Make sure that your touch sensor is
TOUCH READING:
TOUCH RIGHT: False
TOUCH LEFT: False
TOUCH RIGHT: False
TOUCH LEFT: False
TOUCH RIGHT: False
TOUCH LEFT: False
TOUCH RIGHT: False
TOUCH LEFT: False
TOUCH RIGHT: False
TOUCH LEFT: False
```
Goals - Results

• Make programming NXT easier ✔
• Allow for coding of Lego Mindstorms NXT brick via Processing language ✔
• Create an easy environment for coding and display of an output program ✔
• Make it as easy as possible to recreate this setup ✔
Challenges

- Lots of moving parts
- Wired connection
- Battery Life
- Difficult to debug various Systems
Possible Improvements for Final

• Bluetooth connective for wireless
• Install instructions (Make Wiki)
• Roslaunch file for command line simplification
• Combined install package
• Nicer user interface html