



redhat.

Home automation with pymodbus

Raspberry pi as a home automation controller using
pymodbus and node-red

Yaakov Zamir

PyCon Israel 2018

Links:

- <https://github.com/yaacov/pyconil-2018-pymodbus>
- <https://github.com/riptideio/pymodbus>
- <https://github.com/yaacov/node-modbus-serial>
- <https://www.arduino.cc/>
- <https://www.raspberrypi.org/>
- <https://nodered.org/>

Agenda

- Who am I and why am I here ?
- About automation.
- About technologies.
- Python ?
- PyModbus.
- Raspberry Pi and Arduino, doing the hard lifting.
- Node Red, making things fun.
- Summary - Pymodbus, Raspberry Pi and Node Red.
- Play time.



Who am I and why am I here ?

I'm a software engineer at Red Hat, part of the Openshift team. Before working for red Hat I worked for a company that manufactured industrial controllers and measurement equipment.

About four years ago the mechanical timer controlling my water heater broke, and I decided to automate and connect my home to the Internet using the tools I work with every day.

My pre-conditions where:

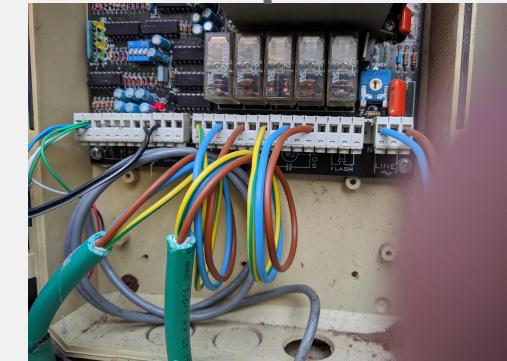
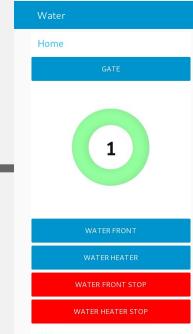
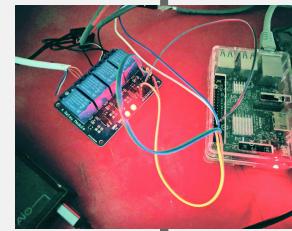
- Free protocols.
- Free software.
- Free Hardware.

This is my automation story ...



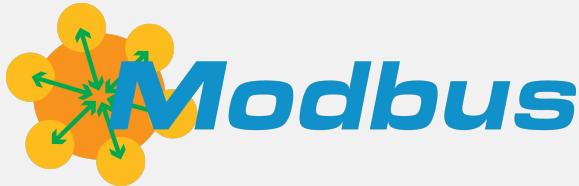
About automation

Interactive objects that can sense and control objects in the physical and digital world.

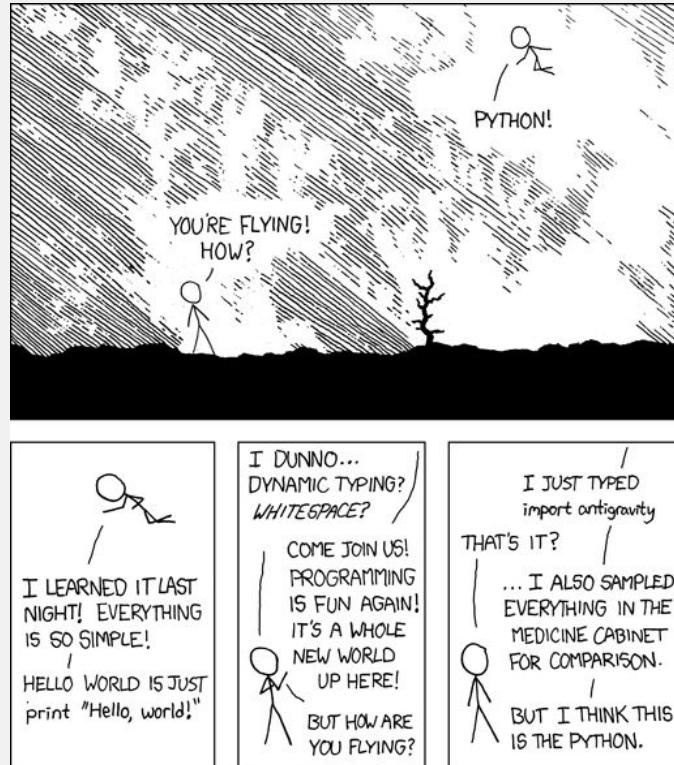


About technologies

Home automation and the Internet of Things (IoT) isn't just about new technologies: It's also about integration with older technologies, a key attribute of which is communication. The available methods of communication are diverse, however, and numerous protocols play a role in bringing the plethora of "things" to the Internet. We will explore two complementary protocols for automation: Modbus, a local protocol for short-distance device attachment, and Message Queuing Telemetry Transport (MQTT), a scalable Internet protocol that enables global communication for the IoT.



Python ?



<https://xkcd.com/353/>

PyModbus

Pymodbus is a full Modbus protocol implementation using twisted for its asynchronous communications core. It can also be used without any third party dependencies (aside from pyserial) if a more lightweight project is needed. Furthermore, it should work fine under any python version > 2.7 (including python 3+)



Raspberry Pi, doing the hard lifting.

The Raspberry Pi is a credit-card-sized computer that plugs into your TV and a keyboard. It is a capable little computer which can be used in electronics projects, and for many of the things that your desktop PC does.

Arduino

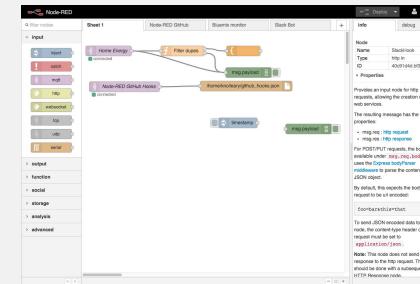
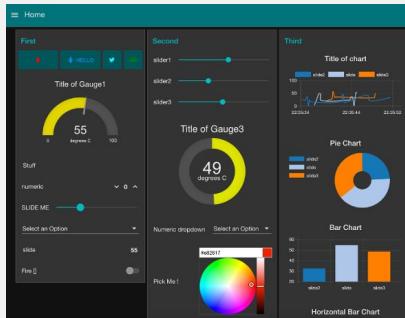
Arduino is an open source computer hardware and software company, project, and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical and digital world.



Node Red, making things fun.

Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways.

It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.



Summary

Free protocol:

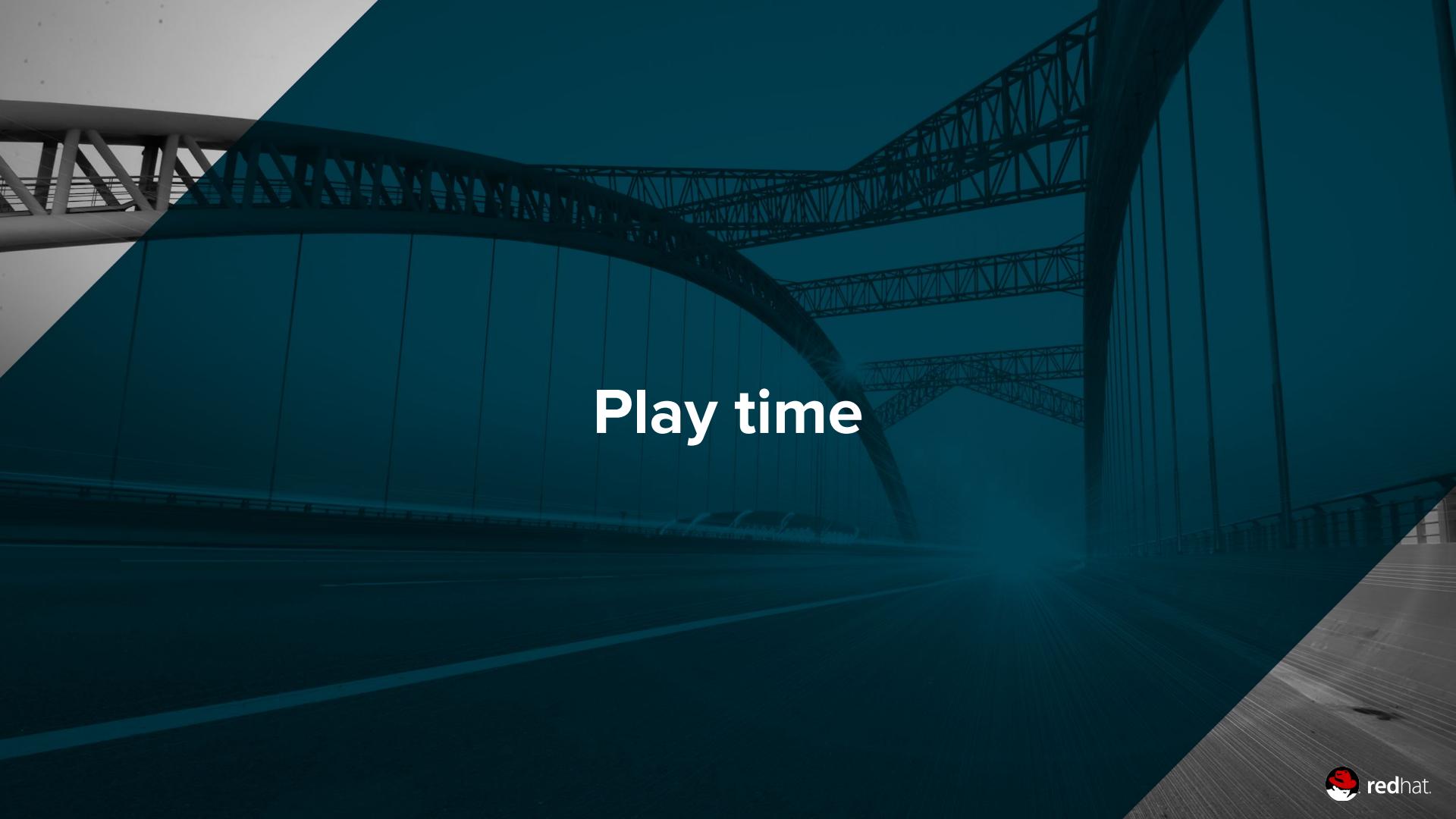
- Modbus - simple, robust and open communication protocol.

Free hardware:

- Raspberry Pi - capable, network ready controller.
- Arduino - an open source single-board microcontrollers that can sense and control objects in the physical world.

Free software:

- Node Red - a programming tool for hardware devices.



Play time

Devices

Input Devices:

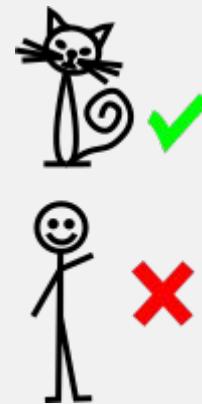
- Cat Detector (CD) - a cat detecting device.
- Thermometer - humidity and temperature sensor.
- Tweets - the internet as a data source.

Output Devices:

- Doorway lights - controlling 220v lights.
- Mood lights - colored lights.

Devices (Images) :

Cat detector (CD):



Thermometer::



UI

- Node Red - dashboard.

The background of the slide features a large, modern bridge with multiple arches and cables, set against a dark, moody background. The bridge's structure is illuminated from below, creating a dramatic effect. A white diagonal bar is positioned in the top left corner.

Q&A

The background of the slide features a photograph of a large bridge at night, with its lights reflecting on the water below. A solid red diagonal band runs from the top-left corner to the bottom-right corner, partially obscuring the bridge. The text "More slides" is centered in white within this red area.

More slides

Modbus & MQTT

Modbus is a serial communications protocol that first appeared in 1979 and is the de facto standard protocol for connecting industrial devices. MQTT appeared 20 years later, bringing these two protocols together gives deeply embedded devices the scale and connectivity of the Internet. In this talk we will focus on modbus.

Modbus - Serial Communications

Modbus is a serial communications protocol originally published by Modicon (now Schneider Electric) in 1979 for use with its programmable logic controllers (PLCs). Modbus has become a de facto standard communication protocol and is now a commonly available means of connecting industrial electronic devices.^[1] The main reasons for the use of Modbus in the industrial environment are:

- developed with industrial applications in mind.
- openly published and royalty-free.
- easy to deploy and maintain.
- moves raw bits or words without placing many restrictions on vendors.

MQTT - Message Queuing

MQTT (Message Queuing Telemetry Transport) is an ISO standard (ISO/IEC PRF 20922) publish-subscribe-based messaging protocol. It works on top of the TCP/IP protocol. It is designed for connections with remote locations where a "small code footprint" is required or the network bandwidth is limited. The publish-subscribe messaging pattern requires a message broker.

RS 485

RS-485, also known as TIA-485(-A), EIA-485, is a standard defining the electrical characteristics of drivers and receivers for use in serial communications systems. Electrical signaling is balanced, and multipoint systems are supported.

P.S - Ethernet

[ANSI/TIA-568 is a set of telecommunications standards from the Telecommunications Industry Association (TIA). The standards address commercial building cabling for telecommunications products and services.]

Why I choose Modbus.

MQTT and Modbus help automation in a complementary way. Using Modbus as a local interface to manage devices and MQTT as a global protocol to expand the reach of those devices' data.

Physical layer:

- RS/485, above 1 km, can run on many cable types, multiple devices on one line.
- Ethernet, 100m, require specific cables, device to device.

I choose modbus because:

- I did not want to rely on Ethernet or WiFi.
- All my automation was on local network.
- Most of my communication is from the master to the device.

The four essential freedoms

What makes something free software:

- The freedom to run the program as you wish, for any purpose (freedom 0).
- The freedom to study how the program works, and change it so it does your computing as you wish (freedom 1). Access to the source code is a precondition for this.
- The freedom to redistribute copies so you can help others (freedom 2).
- The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.