



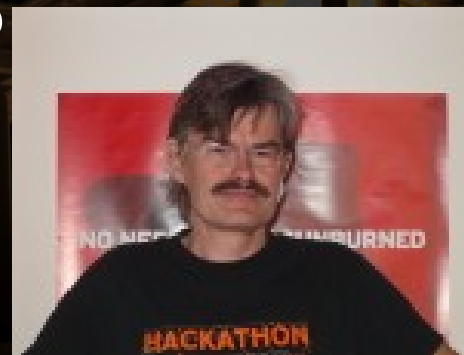
# APACHE ROADSHOW

Berlin, June 11-14, 2018

# Having fun with Raspberry(s) and Apache Projects

**Presented by:**

Jean-Frederic Clere, Manager, Software  
Engineering, Red Hat



# AGENDA

- Who I am
- How does it started
- OK now I have my demo for HTTP/2 what next.
- Get Astro Hat and have fun.
- Get another Hat and have another fun.
- More serious people using Industrino
- Questions

# Who I am I?



Jean-Frederic Clere

Red Hat

Years writing JAVA code and server software

Tomcat committer since 2001

Doing OpenSource since 1999

Cyclist/Runner etc

Lived 15 years in Spain (Barcelona)

Now in Neuchâtel (CH)



# How does it start?

## Trying to make a demo :D

- first localhost (failed)
- remote server (failed)
- try local + configurable:
  - Need a very small hardware:
  - Need real OS (no Arduino)
  - Fast
  - With WIFI



- Hardware: sd card / wifi access point
- Most distributions requires for installation:
  - Screen
  - Keyboard
  - Solution:
    - mount root
    - remove autostart (tricky SystemD)
    - add ssh keys
- Next yum install java/openssl/gcc etc...
- Done in a few hours...

- Fedora 24 (with RPI kernel and modules)
- Drivers from <https://github.com/raspberrypi/firmware>
- wifi access point from (free since September 2016)  
<https://raw.githubusercontent.com/RPi-Distro/firmware-nonfree/master/>
- dhcp (server)
- bind (name server to make captive portal)
- Oracle JDK 8 for ARM (Java Openjdk version "1.8.0" too slow)
- Tomcat apache-tomcat-8.5.6
- Apache httpd (the fedora one)
- <http://10.0.0.201/>

- HTTP/2
- Tomcat-8.5.6 (bin tar)
- Tomcat-native-1.2.10 (sources compiled on the Raspberry)
- Openssl 1.0.2j (from Fedora 24)
- <http://10.0.0.203:8080/>(normal tomcat)
- <http://10.0.0.203:8080/http2.html>
- <https://10.0.0.203:8443/http2.html> https normal
- <https://10.0.0.203:8002/http2.html> https HTTP/2
- So play with latency:
  - tc qdisc add dev eth0 delay 85ms 20ms (to get something that isn't localhost).
  - tc qdisc del dev eth0 root (remove it).
  - tc qdisc add dev eth0 root netem delay 185ms 120ms
- <https://10.0.0.202:8443/http2.html> https normal
- <https://10.0.0.202:8002/http2.html> https HTTP/2

Let's start the fun!



- Hats...
- lot experimentation boxes
- Use Astro Hat
- Sensors:
  - Magnetometer
  - Humidity sensor
  - Temperature
  - Accelerometer
  - Joystick
  - And a DISPLAY!!!



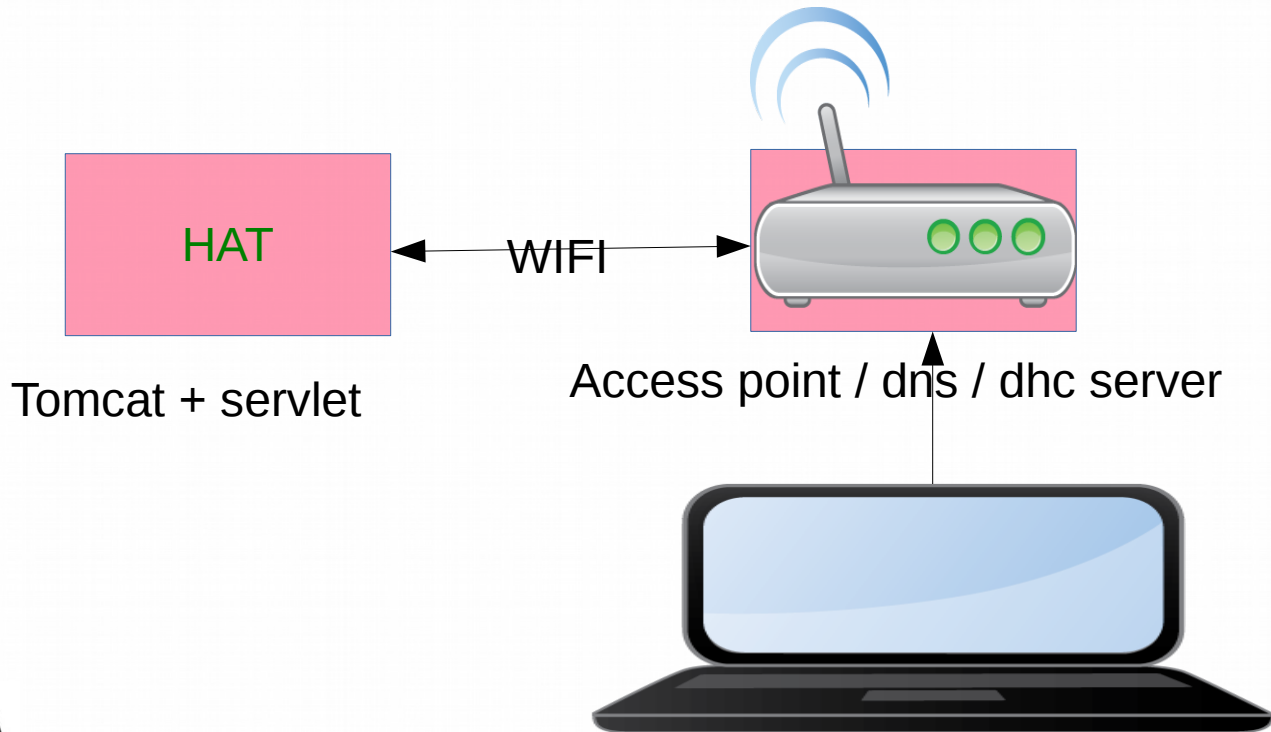


- Servlet
- Frame Buffer
- HTML5 scripts
- Read the display / write / reset etc
- Note the following:
  - Openjdk no JIT compiler (slow, so I use Oracle VM).
  - Openjdk (arm version: memory map file ~ broken)
  - Or frame buffer problem.



# First Fun demo

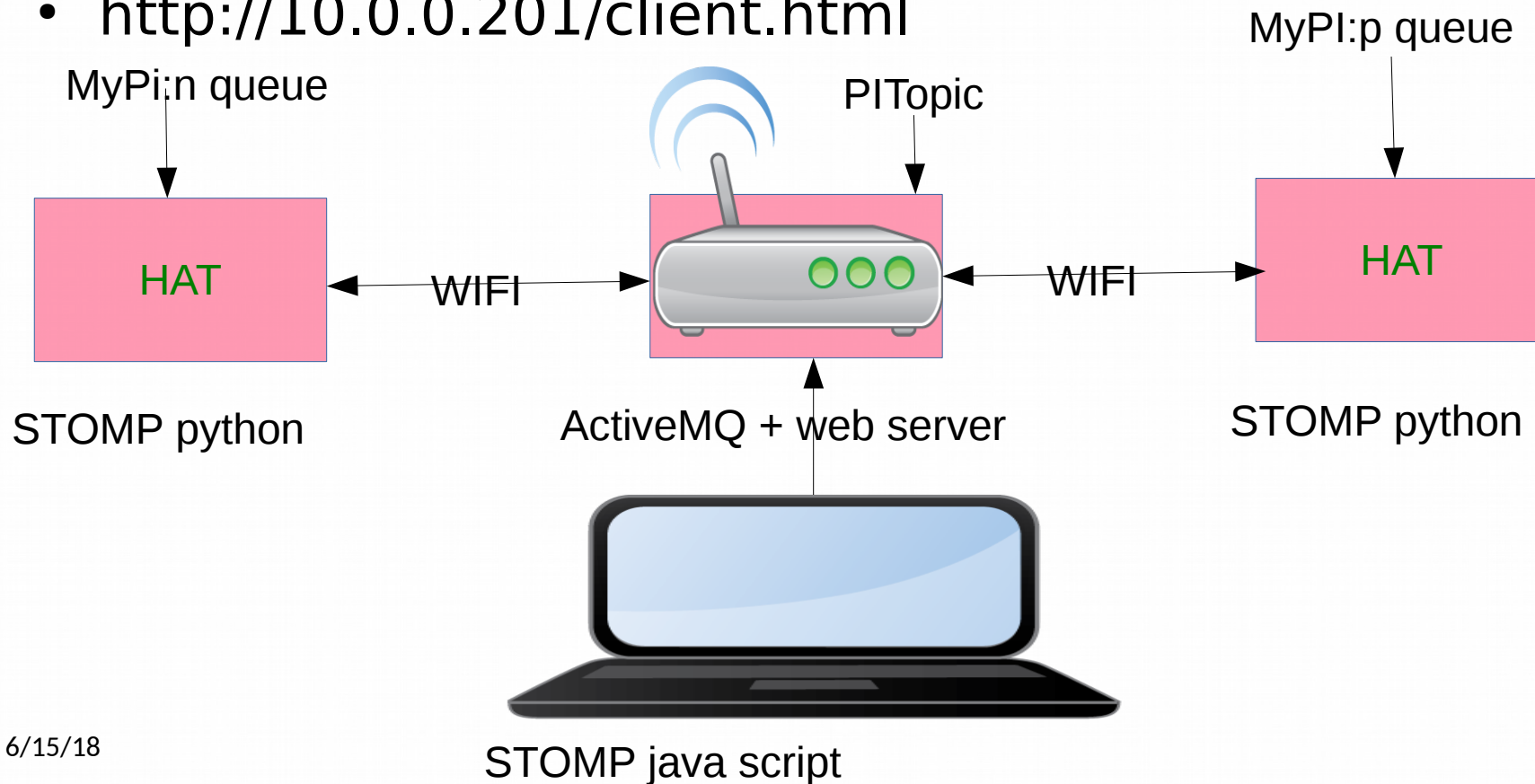
- <http://10.0.0.203:8080/demo-1.0-SNAPSHOT/FrameBuffer>



- Broker easy to collect information
- The Raspberry library are in Python
- Easy to make STOMP (on the PI)
  - Topic to send temperature in the example.
  - Queue on the PI to display a message
- Websocket STOMP on the client
  - html page with java script
  - jquery
  - stomp

# First Fun demo

- <http://10.0.0.201/client.html>





- First the client (java script):  
<http://10.0.0.201/client.html>
- bin/activemq console
- <http://10.0.0.201:8161/admin/> (the activeMQ console admin/admin)
- The object Raspberry have STOMP python application running. (autostarted):

```
root@localhost ROOT]# ps -ef | grep pytho
```

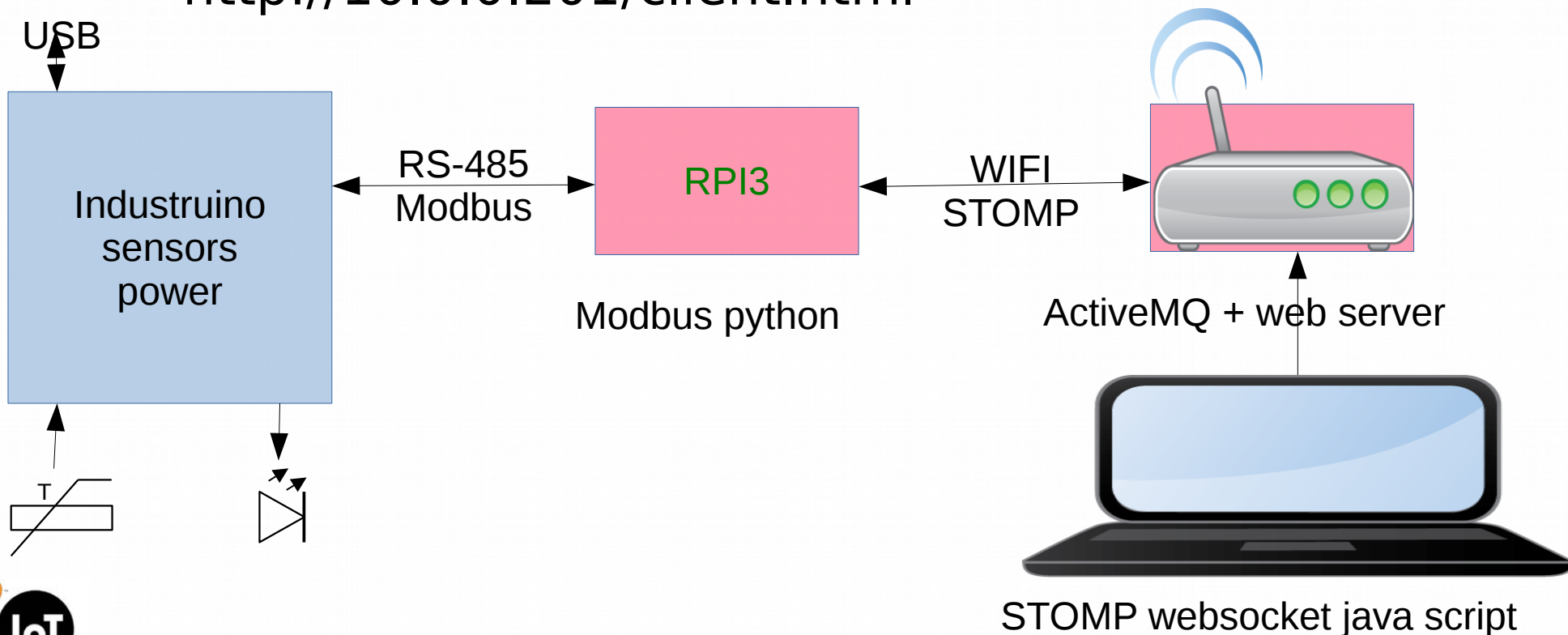
```
root    371    1  0 17:28 ?        00:00:07 /usr/bin/python3 -Es /usr/sbin/firewalld --nofork --nopid
```

```
root    2007    1  1 18:09 ?        00:00:01 /usr/bin/python  
/root/tomcatPI/python/sendtemprecvmess.py
```

```
root    2047   745  0 18:11 pts/0    00:00:00 grep --color=auto pytho
```

- Based on Arduino but for electricians.
- Powered with 24 volts
- No OS and programmed via USB
- Industrial format
- To control pumps, heaters etc
- Measures 2 temperatures
- Connected to ActiveMQ via RPI3 using Modbus
- Mostly OpenSource and OpenHardware

- <http://10.0.0.201/client.html>



- Internet of Things (IoT).
- <http://mynewt.apache.org/> Arduino
- <https://edgent.apache.org/RPI>
- Problems with hardware:
  - Partially OpenSource /OpenHardware :-(
  - Hard to explain to the players the Apache Way



# Do it your self?

- PI 3 + memory + power = 2 lunches
- <http://mynewt.apache.org/>
- <https://edgent.apache.org/>
- Blog: <http://jfclere.blogspot.com.es/>
- Github: <https://github.com/jfclere/tomcatPi>
- Fedora: [https://fedoraproject.org/wiki/Raspberry\\_Pi](https://fedoraproject.org/wiki/Raspberry_Pi)
- Industruino code (you need Industruino libs too):  
[https://github.com/jfclere/Industruino\\_HVAC\\_Functions](https://github.com/jfclere/Industruino_HVAC_Functions)
- Hardware controler: <http://econtrols.org>



• Email: [jfclere@gmail.com](mailto:jfclere@gmail.com)

# Question ?

Thanks !

