

# INTRODUCTION TO COAP

## THE REST PROTOCOL FOR M2M

Julien Vermillard

[jvermillar@sierrawireless.com](mailto:jvermillar@sierrawireless.com)

[jvermillard@apache.org](mailto:jvermillard@apache.org)

Twitter @vrmvrm







# 50 BILLION CONNECTED THINGS BY 2020

- Cars
- Fridges
- Pacemakers
- Light bulbs
- Power outlets







# 50 BILLION CONNECTED THINGS BY 2020

- Wireless bandwidth is limited
- Very cheap connected hardware







# CONSTRAINED APPLICATION PROTOCOL

A new protocol effort at IETF for M2M





# COAP = CONSTRAINED

binary and compact

```

      0               1               2               3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|Ver| T |  TKL  |      Code      |           Message ID           |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|  Token (if any, TKL bytes) ...
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|  Options (if any) ...
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|1 1 1 1 1 1 1 1|      Payload (if any) ...
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```





# **CO**AP = CONSTRAINED

- Simple to decode
- Target 8-bit microprocessors



# COAP = APPLICATION

- REST paradigm for things
- URI: <coap://mysensor/room1/temperature/max>





# COAP = APPLICATION

HTTP-like verbs:

- GET for read
- PUT, POST, DELETE for mutation

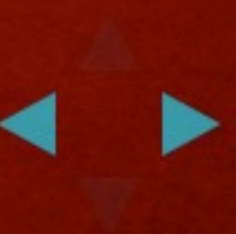




A small white robot with a backpack and a yellow cone on a red background. The robot is positioned on the left side of the frame, facing right. It has a white body, a backpack, and a yellow cone. The background is a solid red color. A semi-transparent dark grey rectangle is overlaid on the right side of the image, containing text and a list of bullet points.

# COAP = PROTOCOL

- IP/UDP only for easy implementation
- UDP to save some bandwidth





# COAP = PROTOCOL

Security with DTLS (SSL/TLS for Datagram)





# DISCOVERY AND METADATA

- Issue a GET on <coap://myhome/.well-known/core>
- Links to all the known resources!




# EXAMPLE

**GET** `coap://myhome/.well-known/core`

```
</power/cons>;if="Watt";title="Power consumption",  
</power/max>;if="Watt";title="Max power consumption in last 24h",  
</uptime>;if="Seconds";title="Seconds since last restart"  
</door/open>;if="Boolean";title="Does the main door is open"
```



A large satellite dish antenna is the central focus, mounted on a complex metal tripod structure. The dish is white with a grid-like pattern. The background is a dramatic sunset or sunrise sky, with a bright orange and yellow glow on the horizon. Silhouettes of trees and a smaller antenna are visible in the distance.

# OBSERVE

- Issue a GET with the **Observe** option
- The resource will be pushed on value change

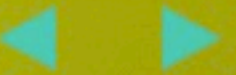






# ADOPTION

- A young protocol
- No large industrial deployment?





# ADOPTION

- Pushed in all the incoming M2M standards
- ETSI M2M: European Telecommunications Standard Institute
- OneM2M (ETSI, TTA, CCSA, ARIB, ATIS, etc..)





# OMA LIGHTWEIGHT M2M

- Standard for device management on top of CoAP
- Provides a model for managing wireless devices
- Registration, bootstrapping, firmware upgrades





# OPENSOURCE

- Contiki – embedded O/S
- Copper – Firefox client plugin
- Libcoap – C CoAP library
- Even on Arduino!





# APACHE MINA SUPPORTS COAP

- Java network application framework
- Fast and scalable: perfect for large number of devices!





# COAP AT ECLIPSE M2M?

- Some interest for CoAP and OMA LW M2M
- Both server-side and client-side
- Join us! [m2m.eclipse.org](http://m2m.eclipse.org)

[eclipse.org](http://eclipse.org)





# THANKS!

[jvermillar@sierrawireless.com](mailto:jvermillar@sierrawireless.com)

[jvermillard@apache.org](mailto:jvermillard@apache.org)

Twitter @vrmvrm

