

# **Lecture #9**

# **CoAP - Constrained**

# **Application Protocol**

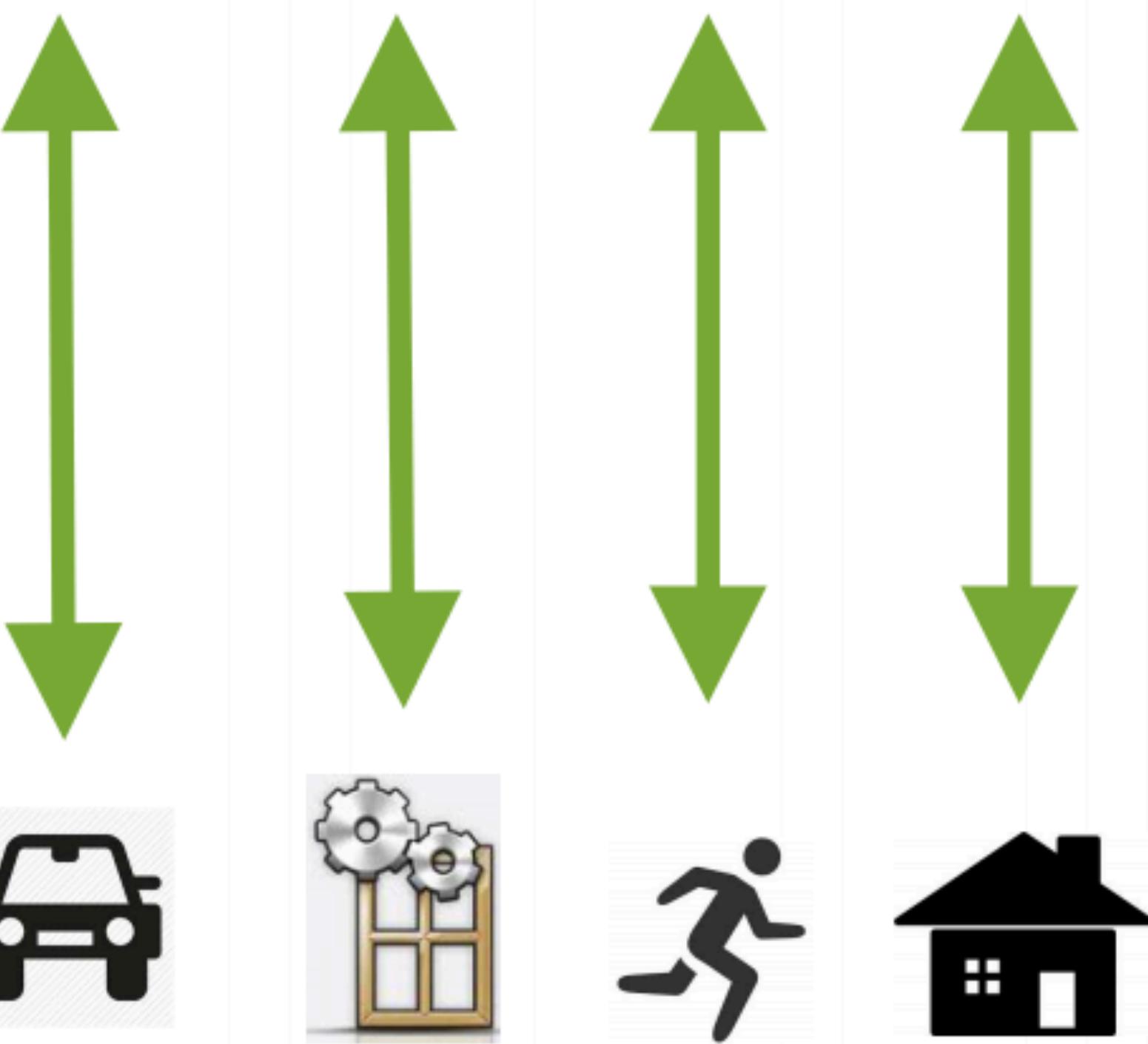
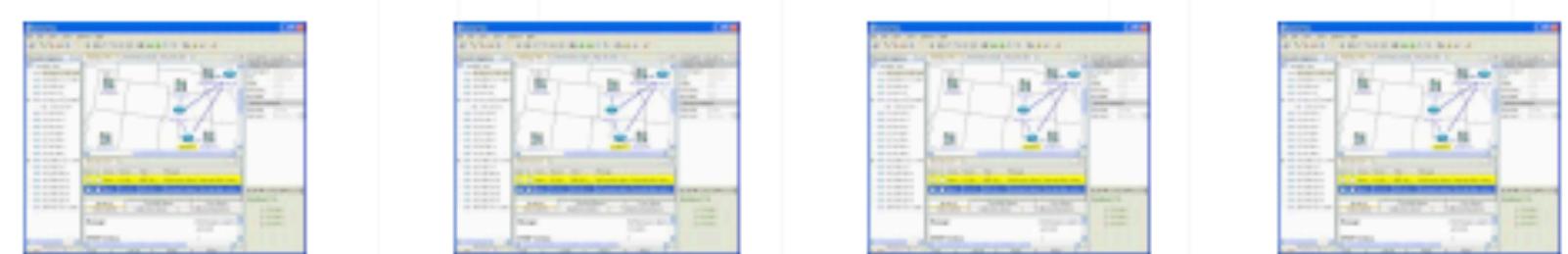
Android Things 2023

# Internet of people

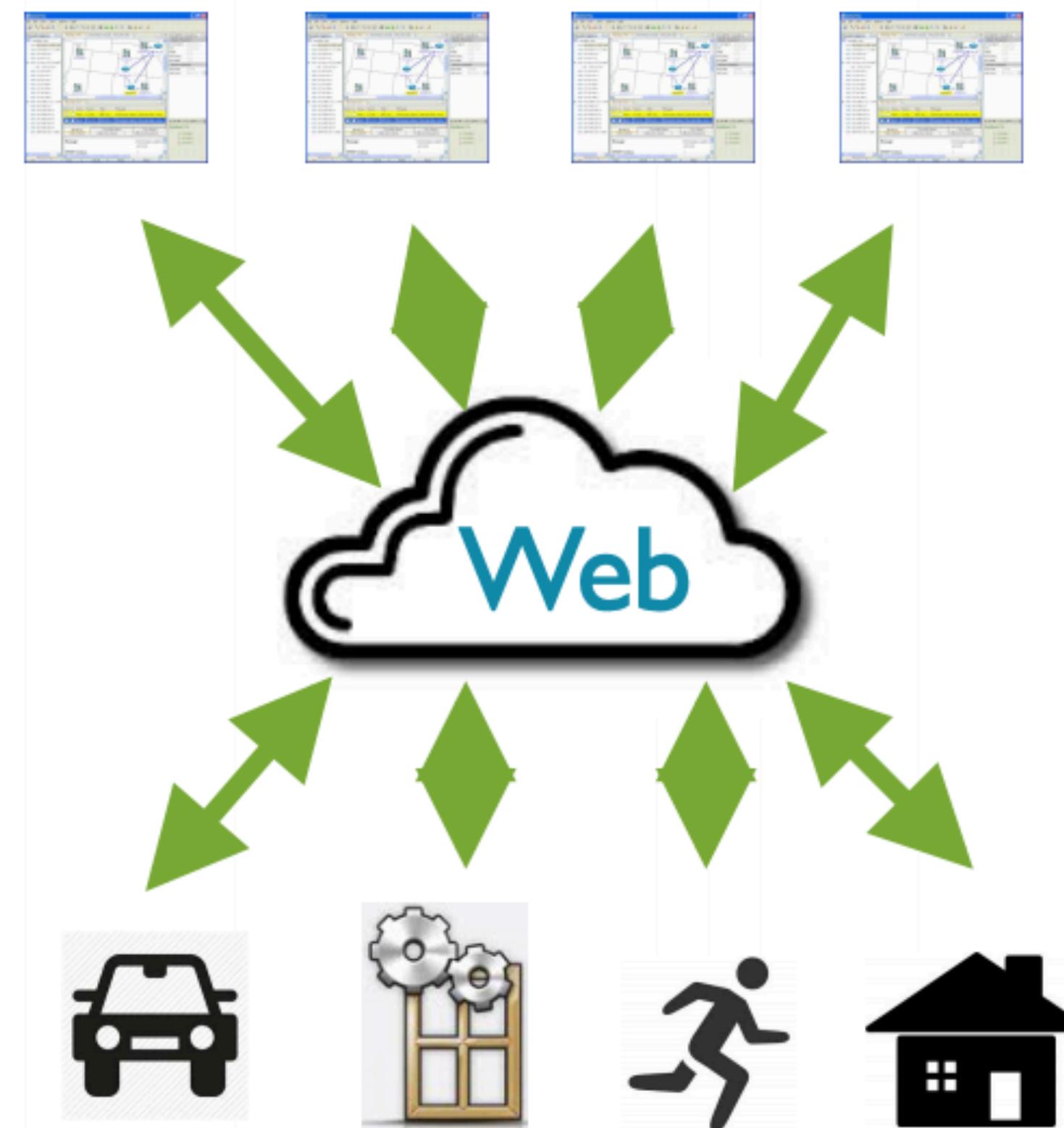




# M2M



# Big Data Internet of Things

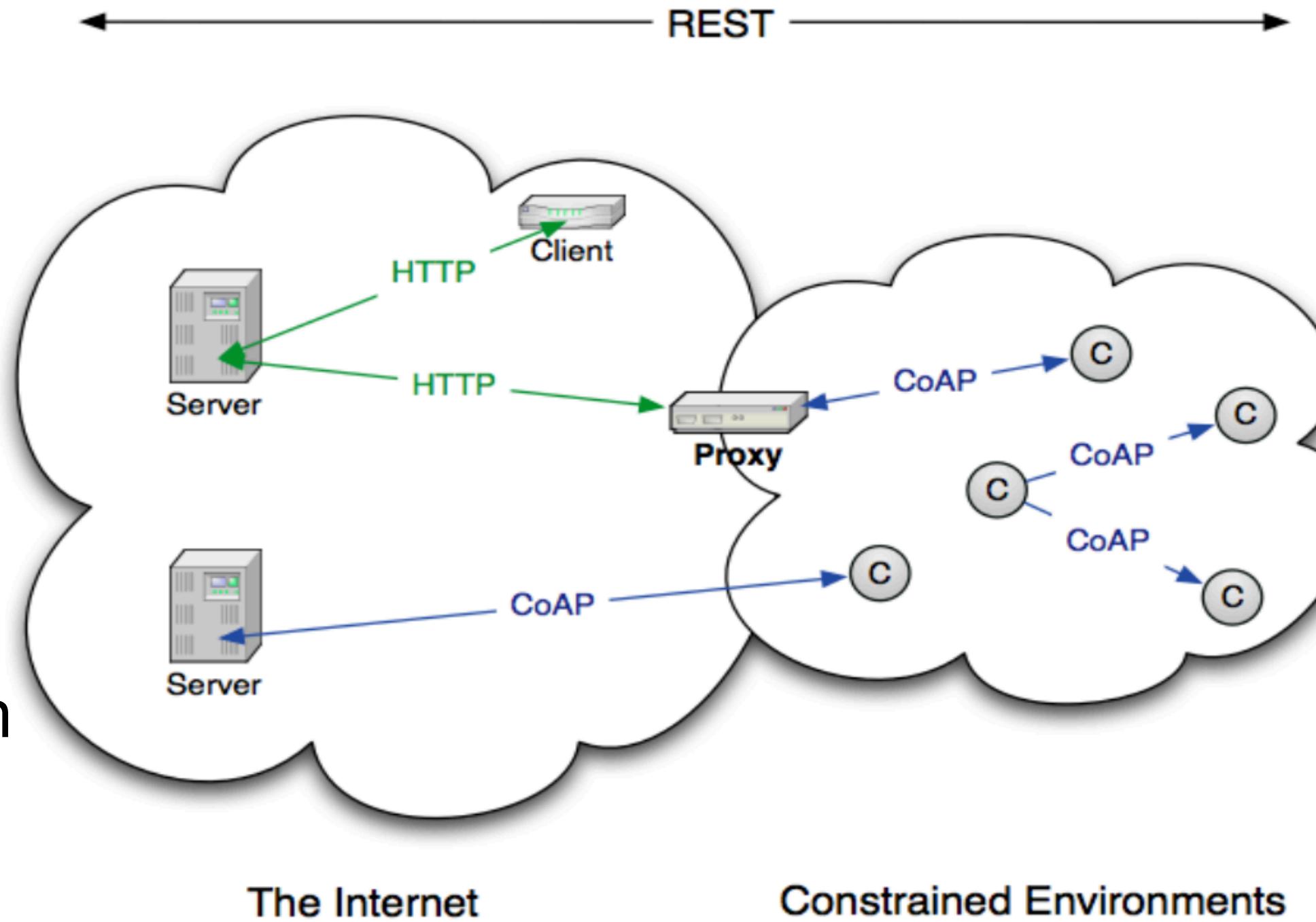


Little Data

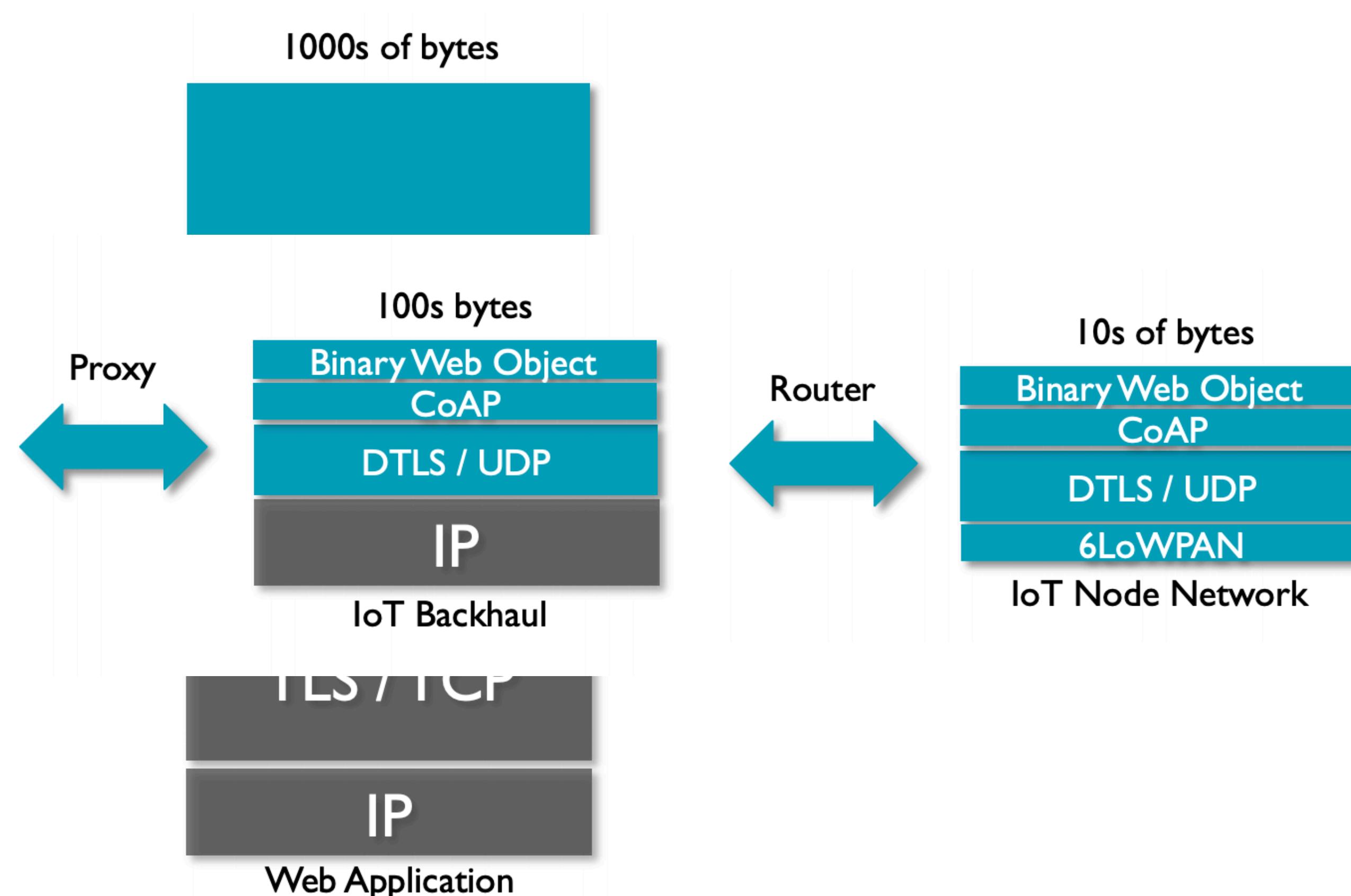
Services  
|  
The Web  
|  
Things

# CoAP

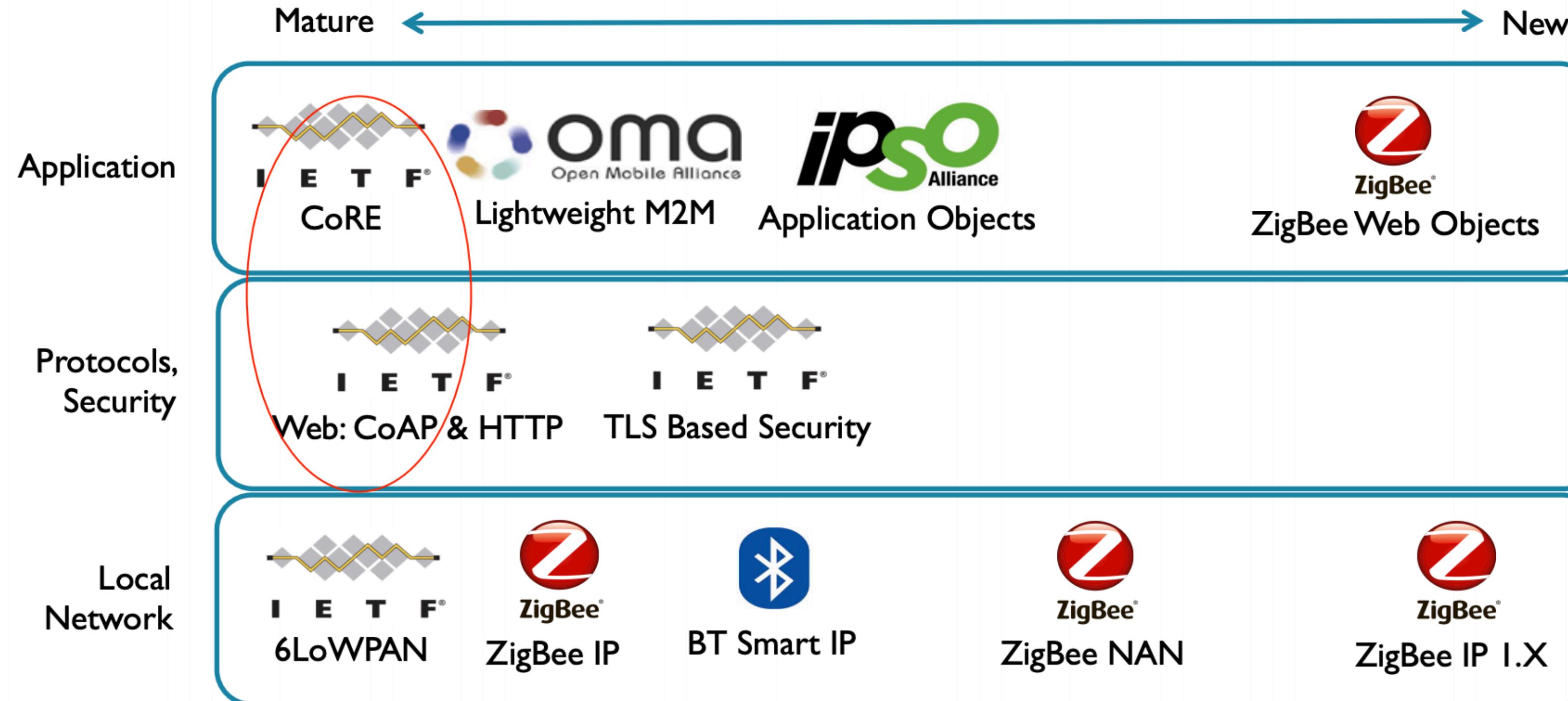
- Open IETF Standard
- Compact 4-byte Header
- UDP, SMS, (TCP) Support
- Strong DTLS Security
- Asynchronous Subscription
- Built-in Discovery



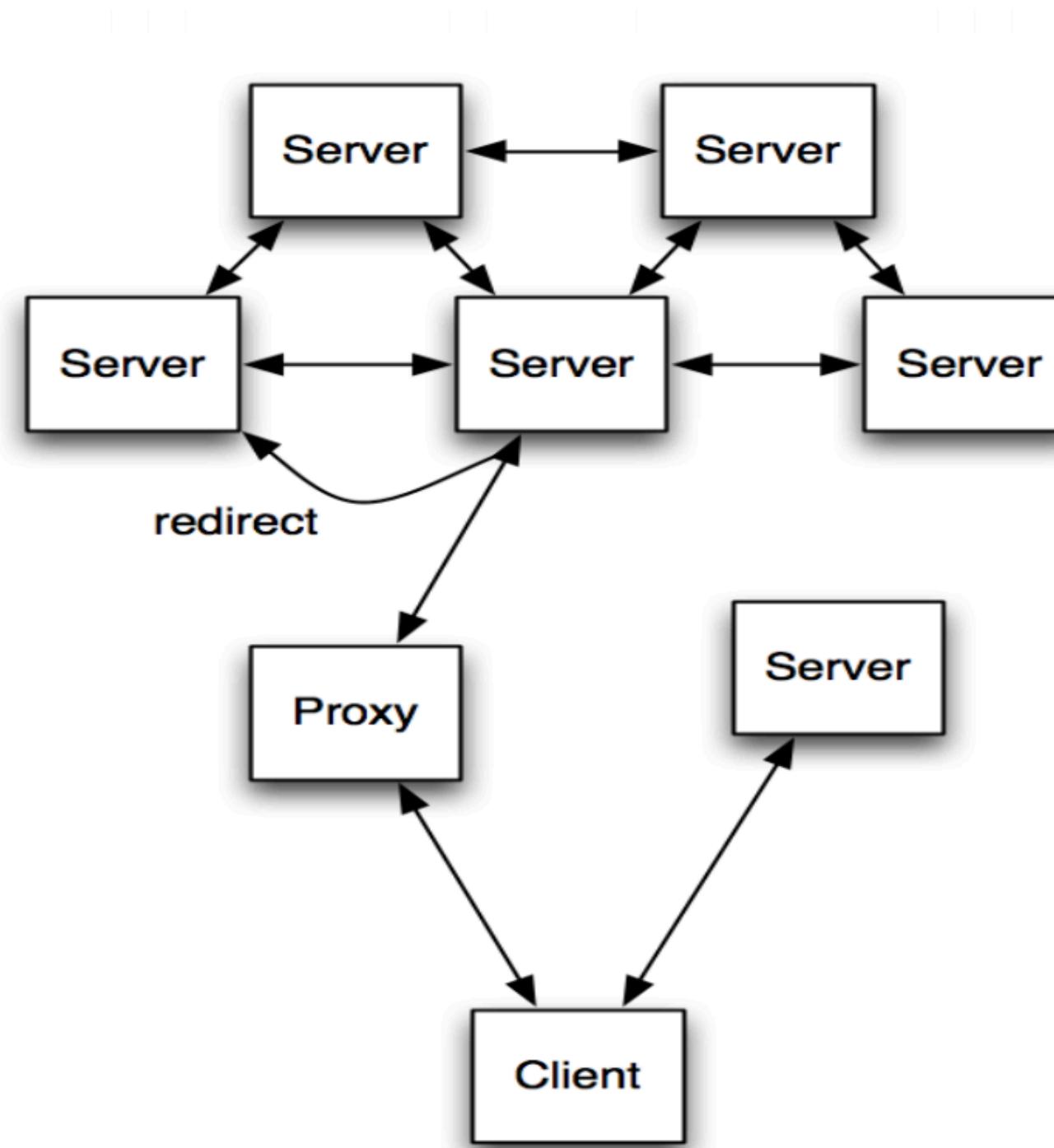
# Transitions from Web to IoT



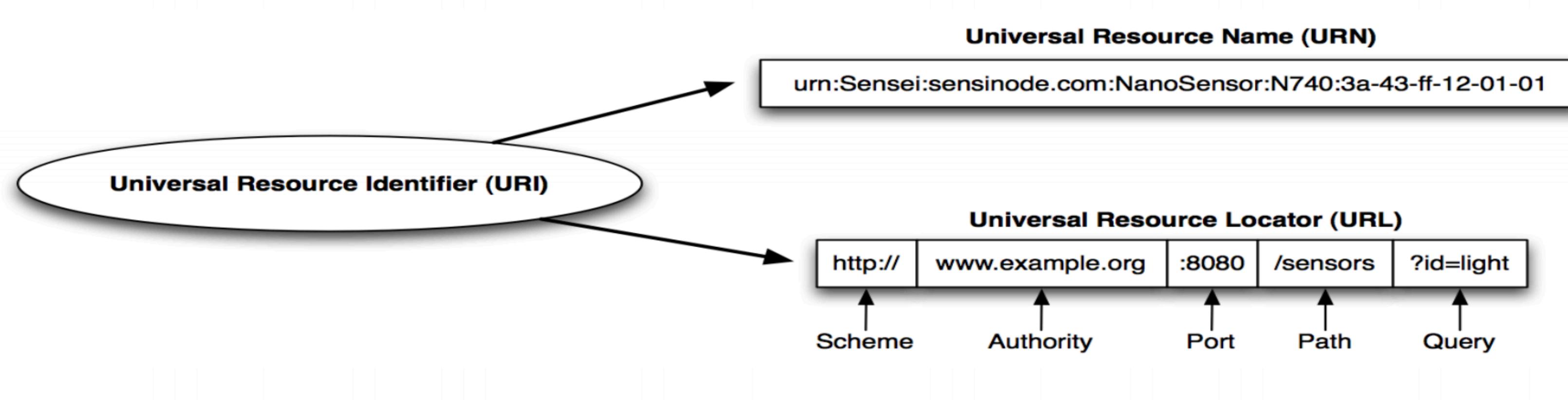
# Community



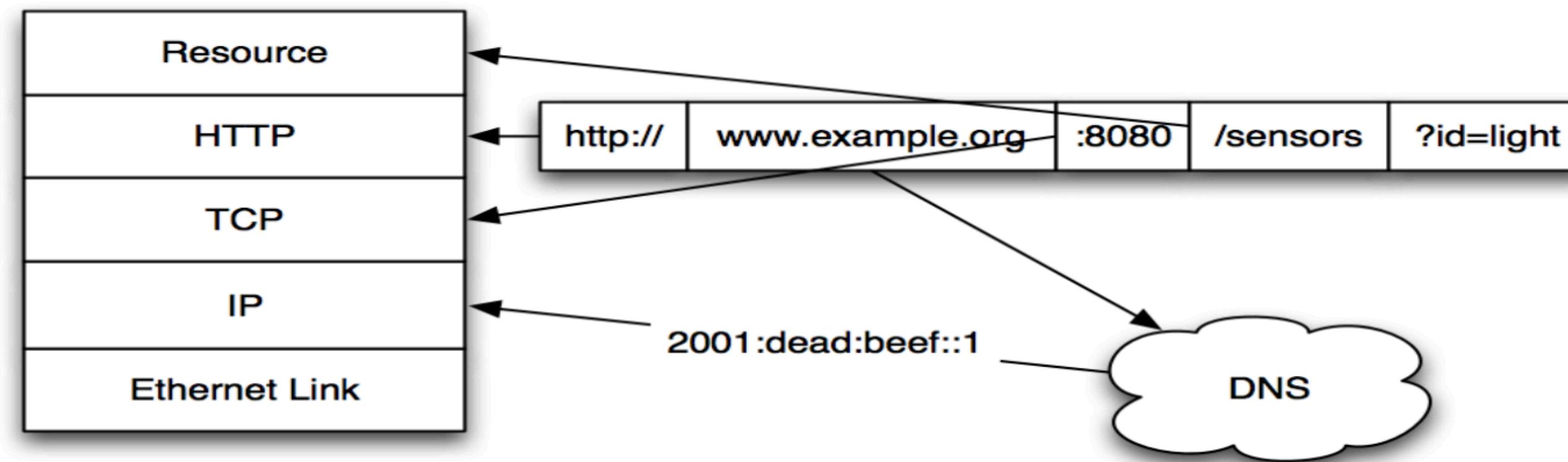
# Web Architecture



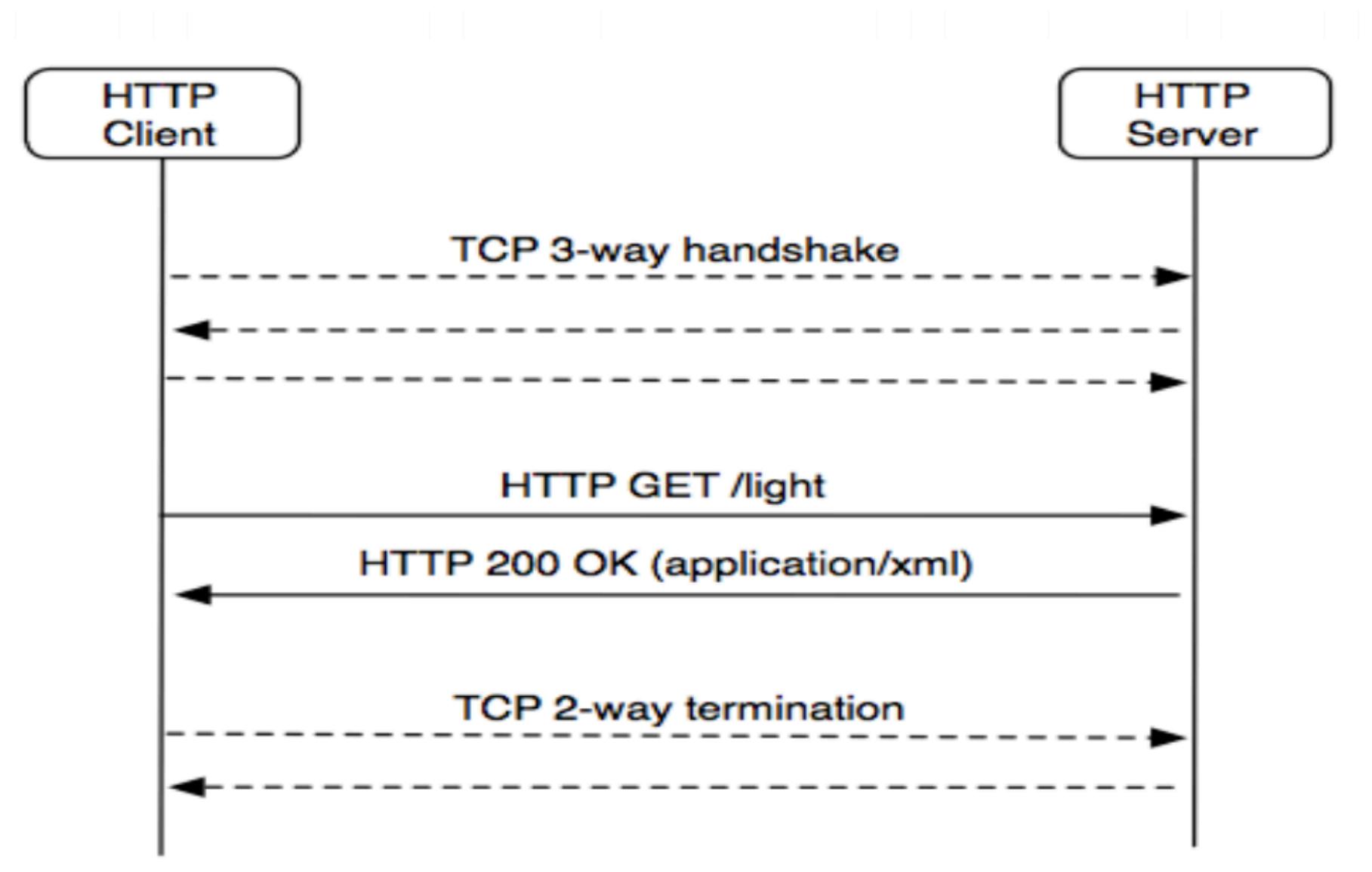
# Naming



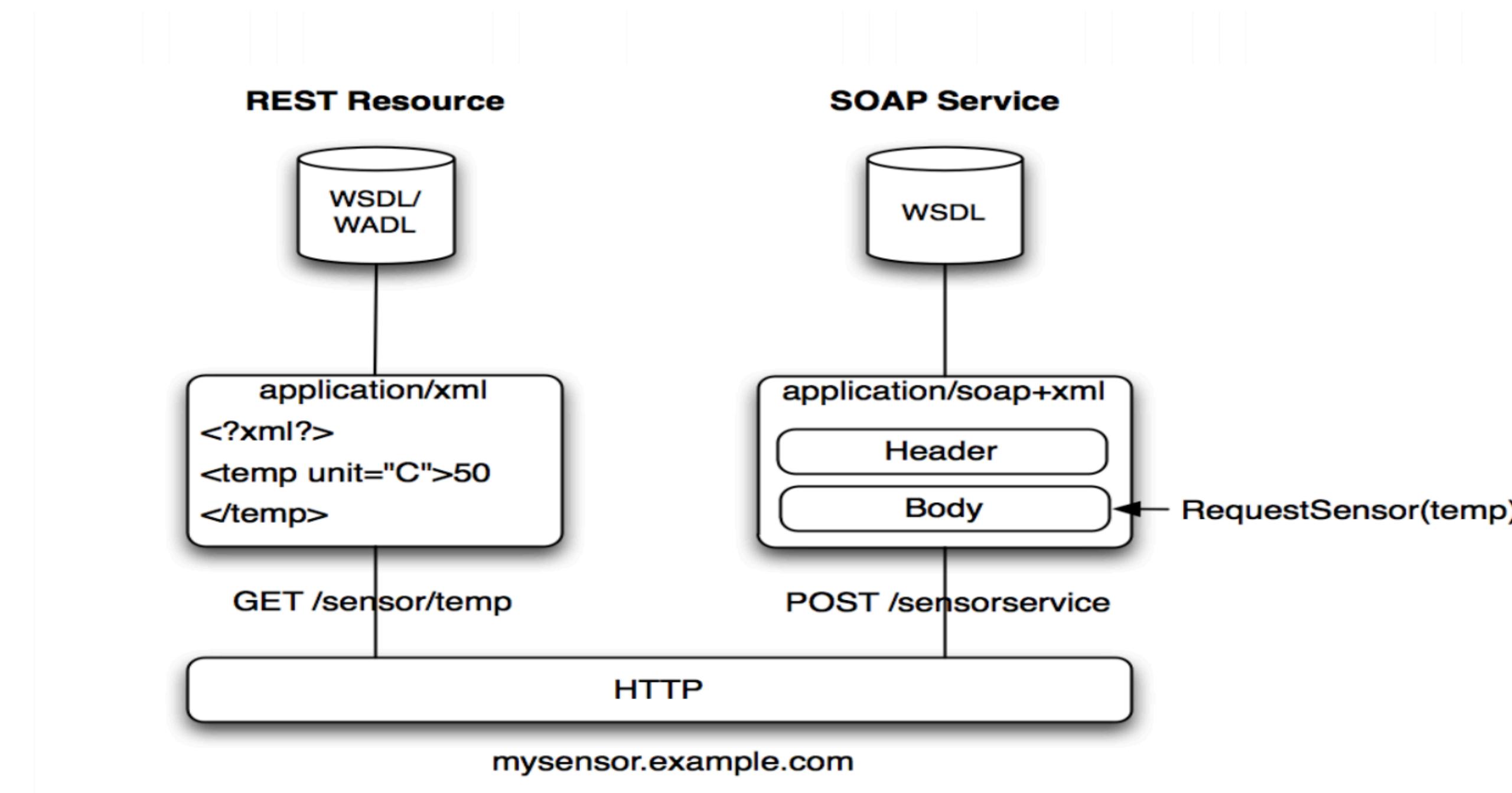
# Resolution



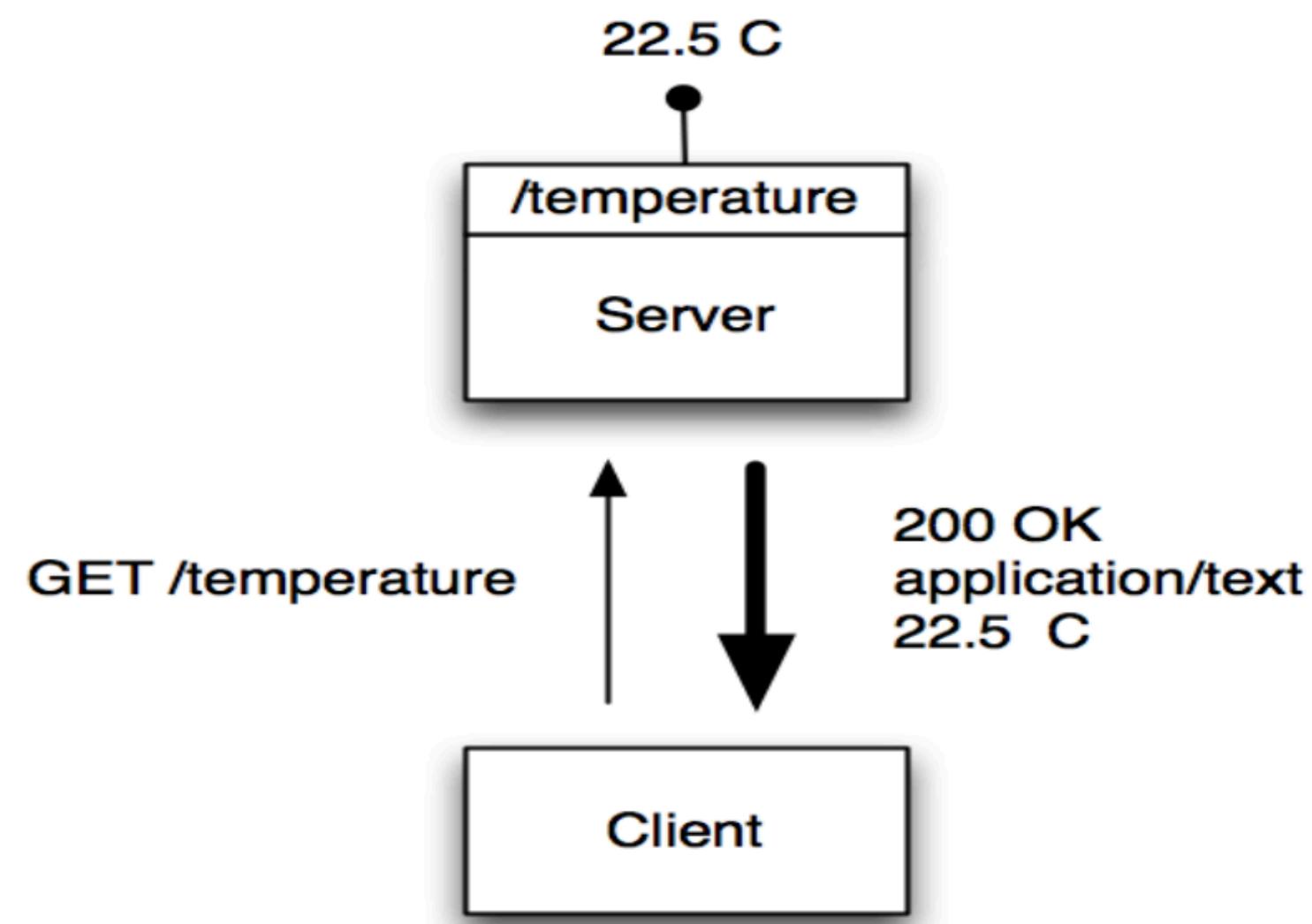
# Traditional HTTP



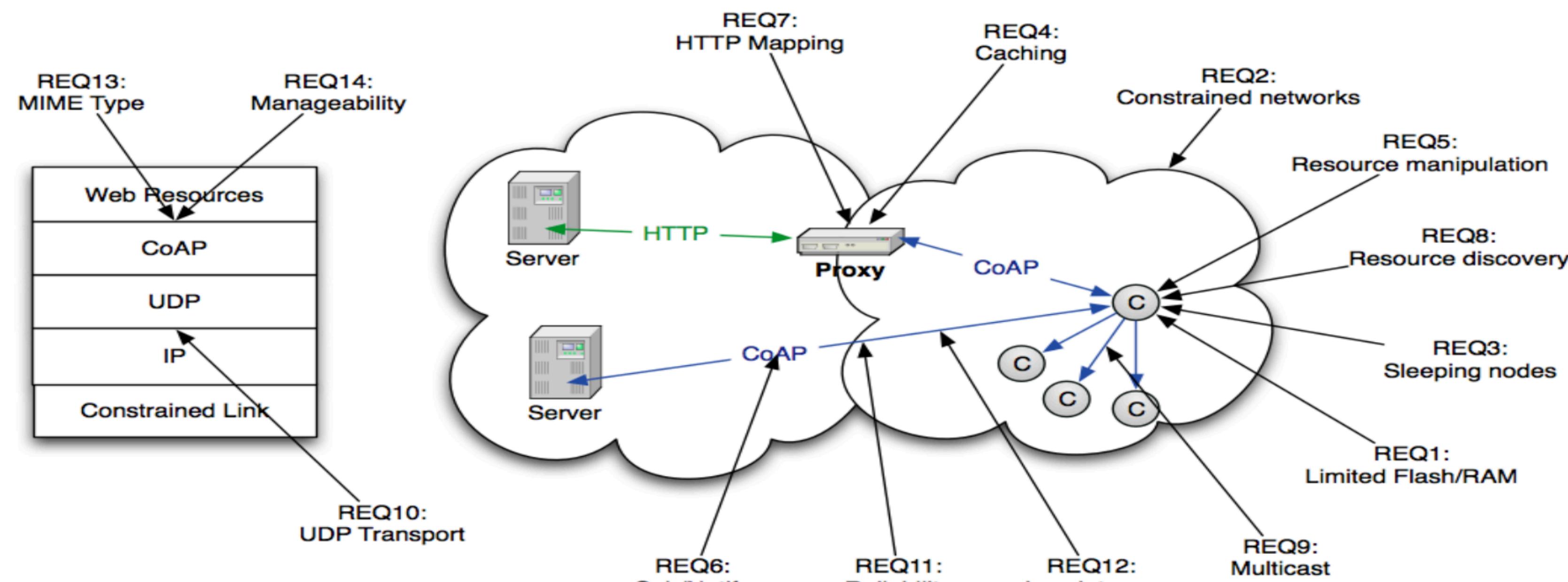
# Web Paradigms



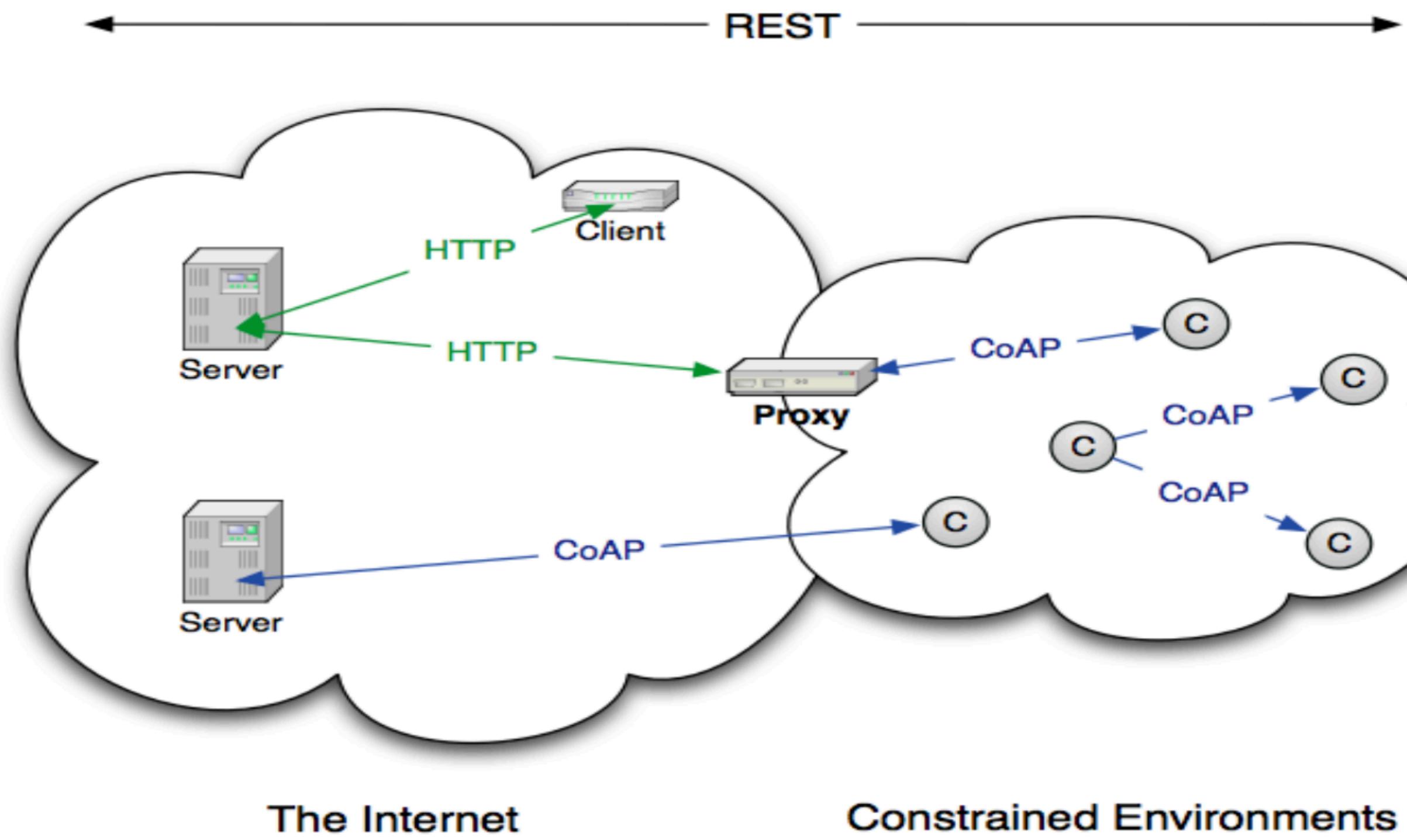
# REST Request



# CoAP



# Architecture



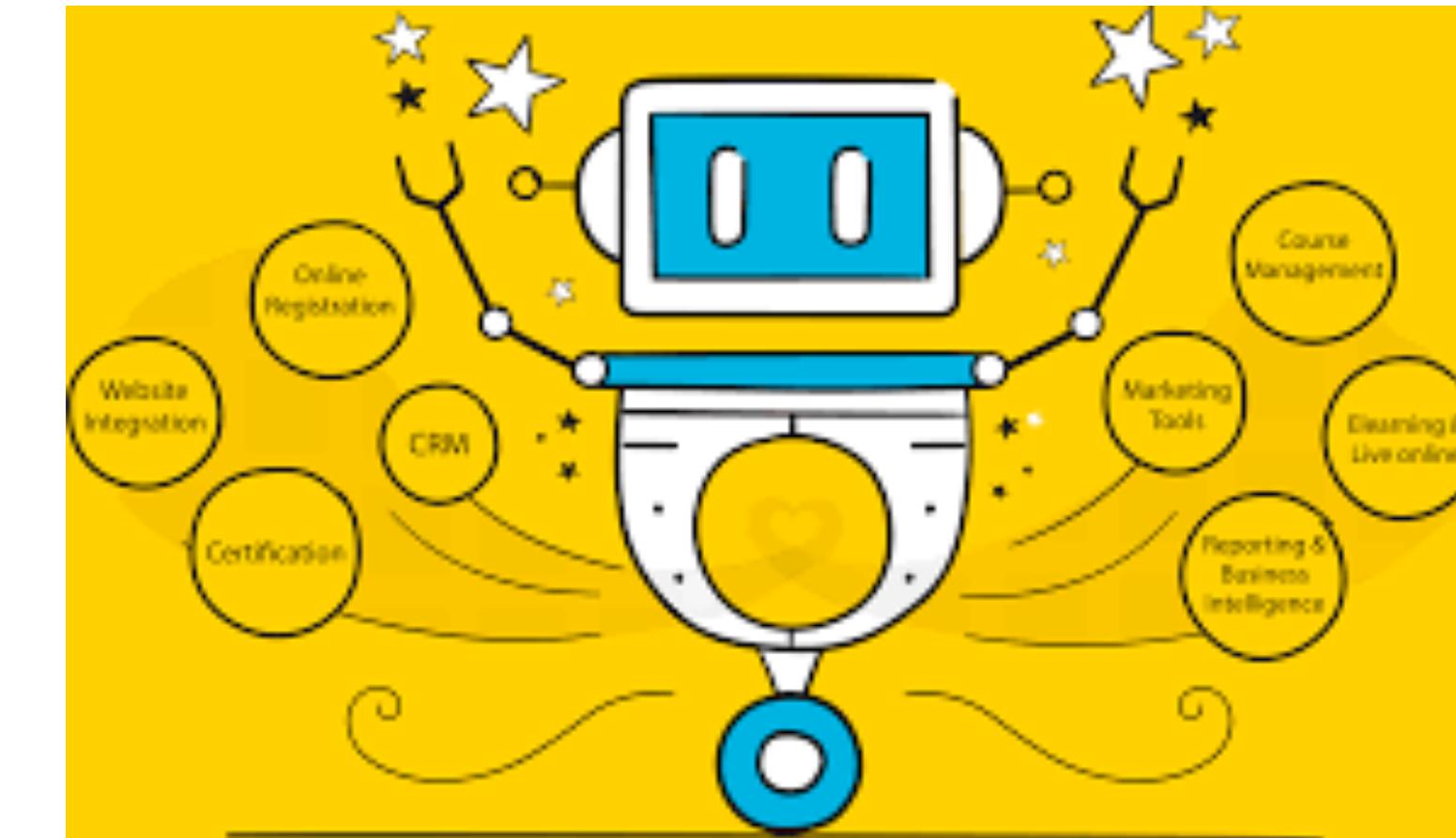
# Pro/Cons

- CoAP is:
  - A very efficient RESTful protocol
  - Ideal for constrained devices and networks
  - Specialized for M2M applications
  - Easy to proxy to/from HTTP
- CoAP is not:
  - A general replacement for HTTP
  - HTTP compression
  - Restricted to isolated “automation” networks



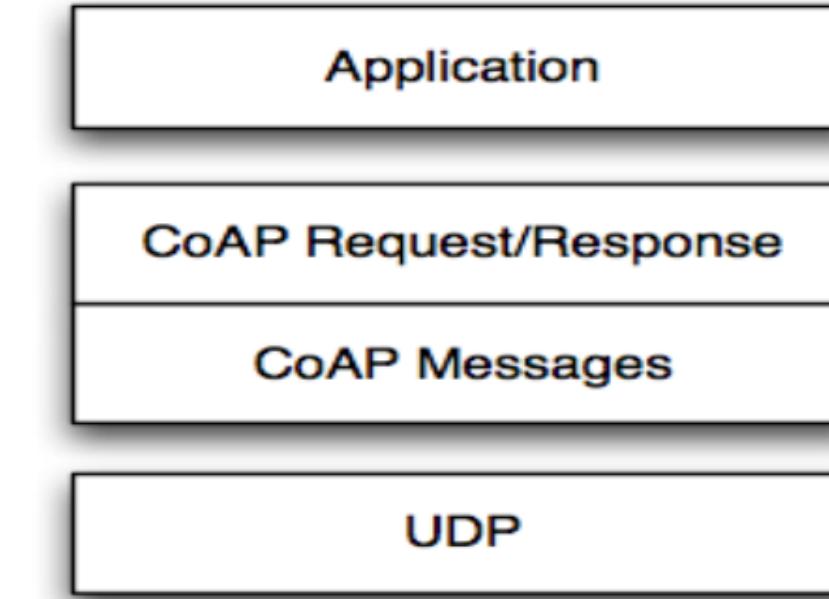
# Features

- Embedded web transfer protocol (coap://)
- Asynchronous transaction model
- UDP binding with reliability and multicast support
- GET, POST, PUT, DELETE methods
- URI support
- Small, simple 4 byte header
- DTLS based PSK, RPK and Certificate security
- Subset of MIME types and HTTP response codes
- Built-in discovery
- Optional observation and block transfer

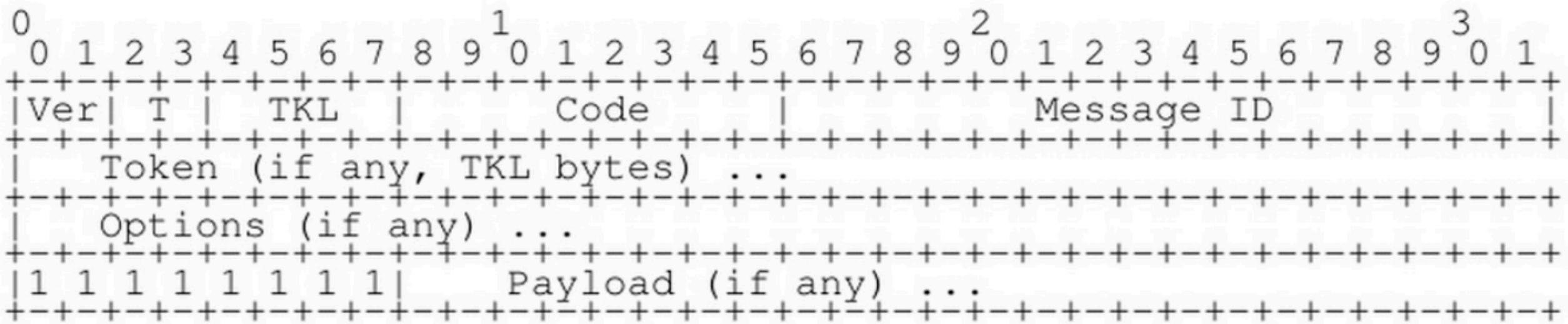


# Transactional Model

- Transport
  - CoAP currently defines:
    - UDP binding with DTLS security
    - CoAP over SMS or TCP possible
  - Base Messaging
    - Simple message exchange between endpoints
    - Confirmable or Non-Confirmable Message answered by Acknowledgement or Reset Message
  - REST Semantics
    - REST Request/Response piggybacked on CoAP Messages
    - Method, Response Code and Options (URI, content-type etc.)



# Header



**Ver** - Version (1)

**T** - Message Type (Confirmable, Non-Confirmable, Acknowledgement, Reset)

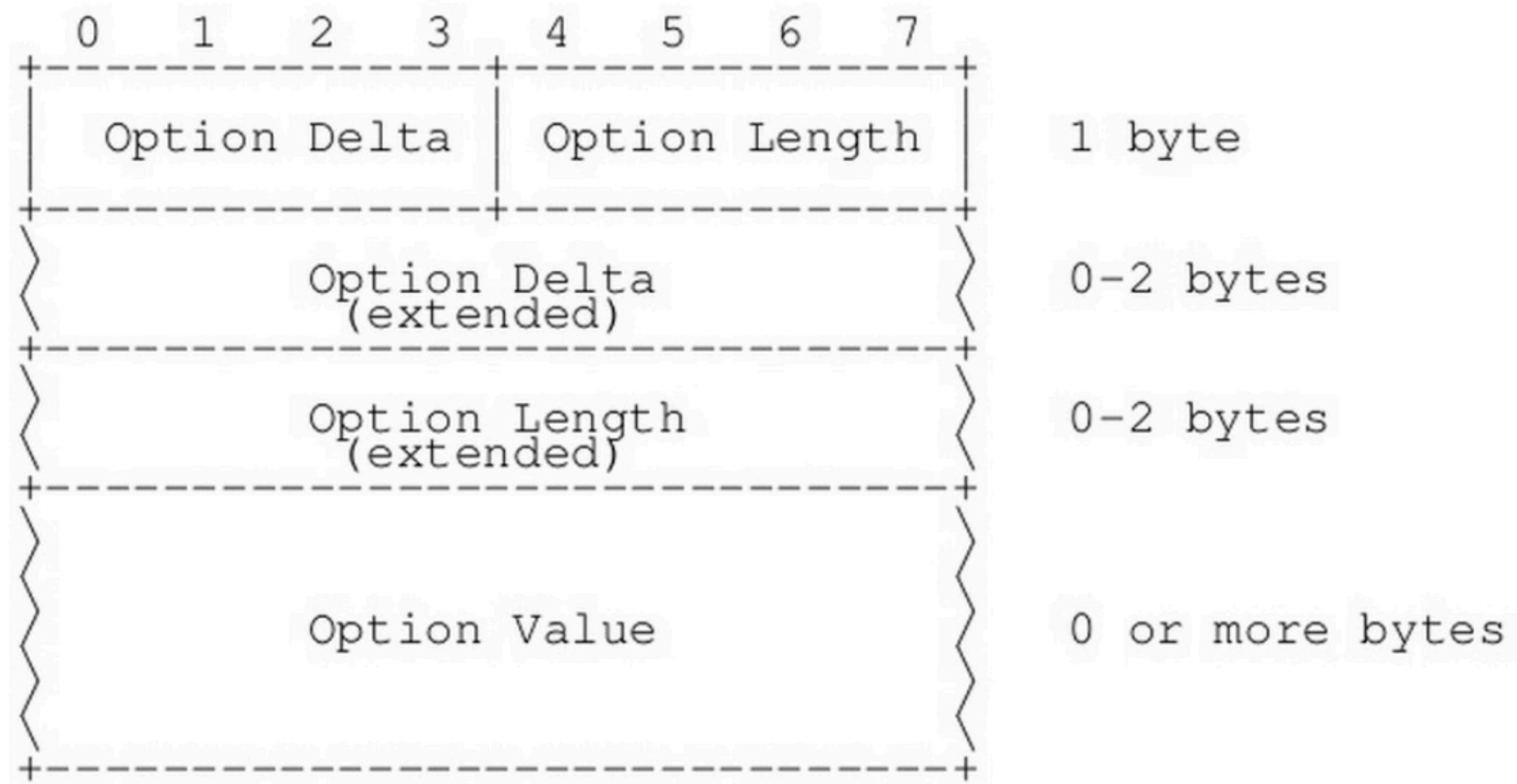
**TKL**- Token Length, if any, the number of Token bytes after this header

**Code** - Request Method (1-10) or Response Code (40-255)

**Message ID** - 16-bit identifier for matching responses

**Token** - Optional response matching token

# Options Field



**Option Delta** - Difference between this option type and the previous

**Length** - Length of the option value

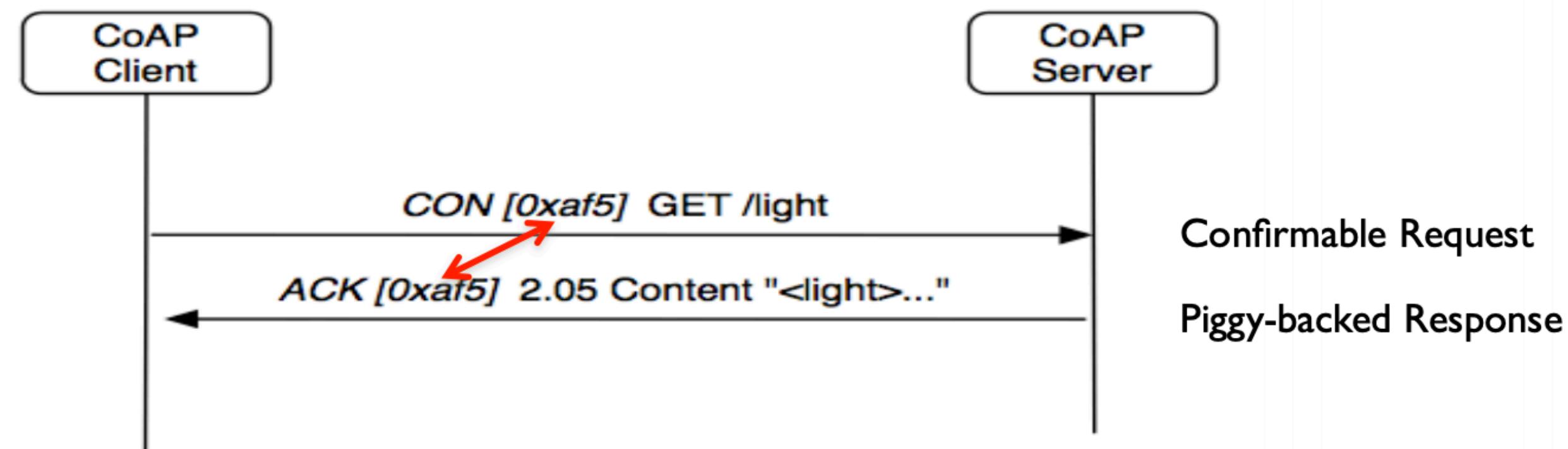
**Value** - The value of Length bytes immediately follows Length

# Base Specification

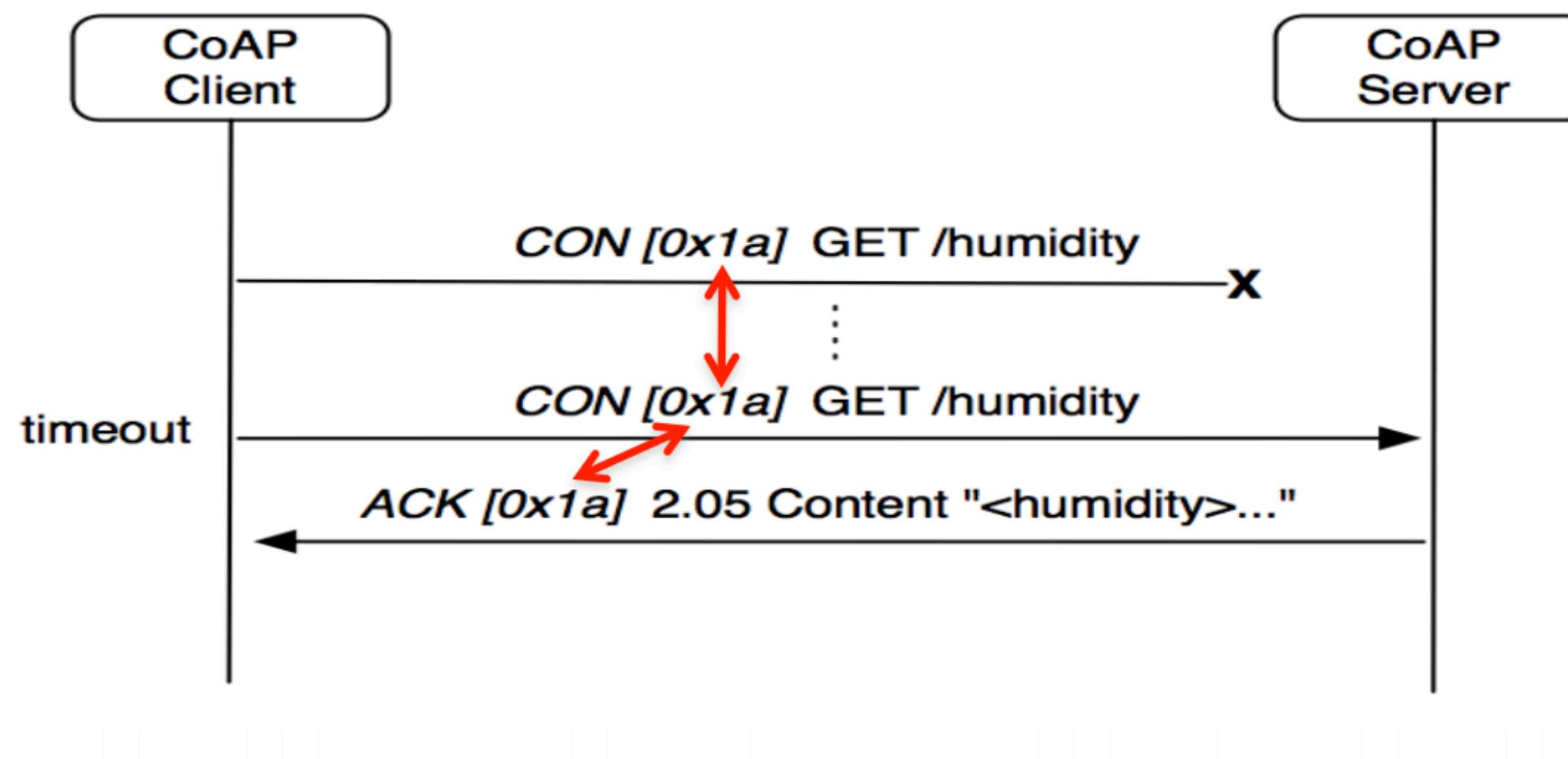
| No. | C | U | N | R | Name           | Format | Length | Default     |
|-----|---|---|---|---|----------------|--------|--------|-------------|
| 1   | x |   |   | x | If-Match       | opaque | 0-8    | (none)      |
| 3   | x | x | - |   | Uri-Host       | string | 1-255  | (see below) |
| 4   |   |   |   | x | ETag           | opaque | 1-8    | (none)      |
| 5   | x |   |   |   | If-None-Match  | empty  | 0      | (none)      |
| 7   | x | x | - |   | Uri-Port       | uint   | 0-2    | (see below) |
| 8   |   |   |   | x | Location-Path  | string | 0-255  | (none)      |
| 11  | x | x | - | x | Uri-Path       | string | 0-255  | (none)      |
| 12  |   |   |   |   | Content-Format | uint   | 0-2    | (none)      |
| 14  |   | x | - |   | Max-Age        | uint   | 0-4    | 60          |
| 15  | x | x | - | x | Uri-Query      | string | 0-255  | (none)      |
| 16  |   |   |   |   | Accept         | uint   | 0-2    | (none)      |
| 20  |   |   |   | x | Location-Query | string | 0-255  | (none)      |
| 35  | x | x | - |   | Proxy-Uri      | string | 1-1034 | (none)      |
| 39  | x | x | - |   | Proxy-Scheme   | string | 1-255  | (none)      |

C=Critical, U=Unsafe, N=NoCacheKey, R=Repeatable

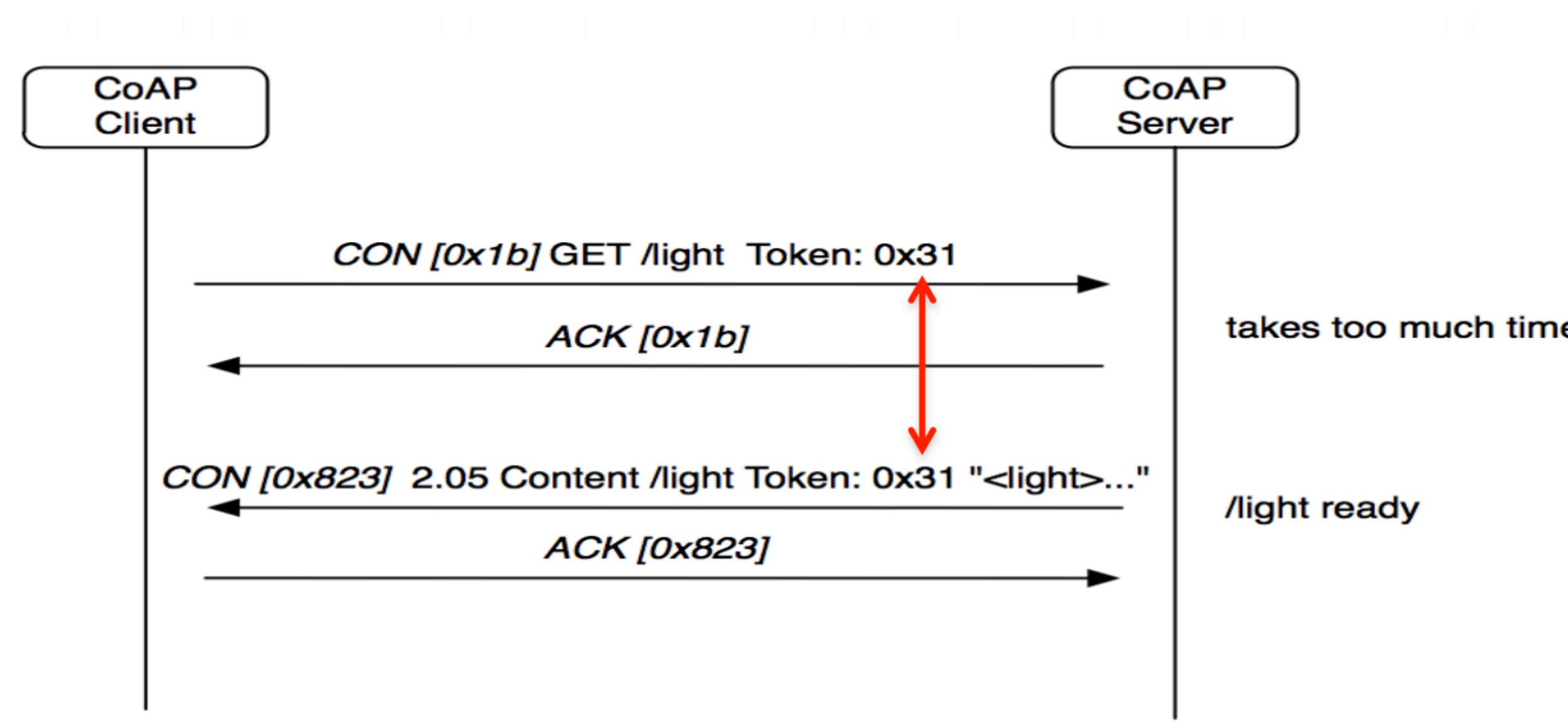
# Simple Request



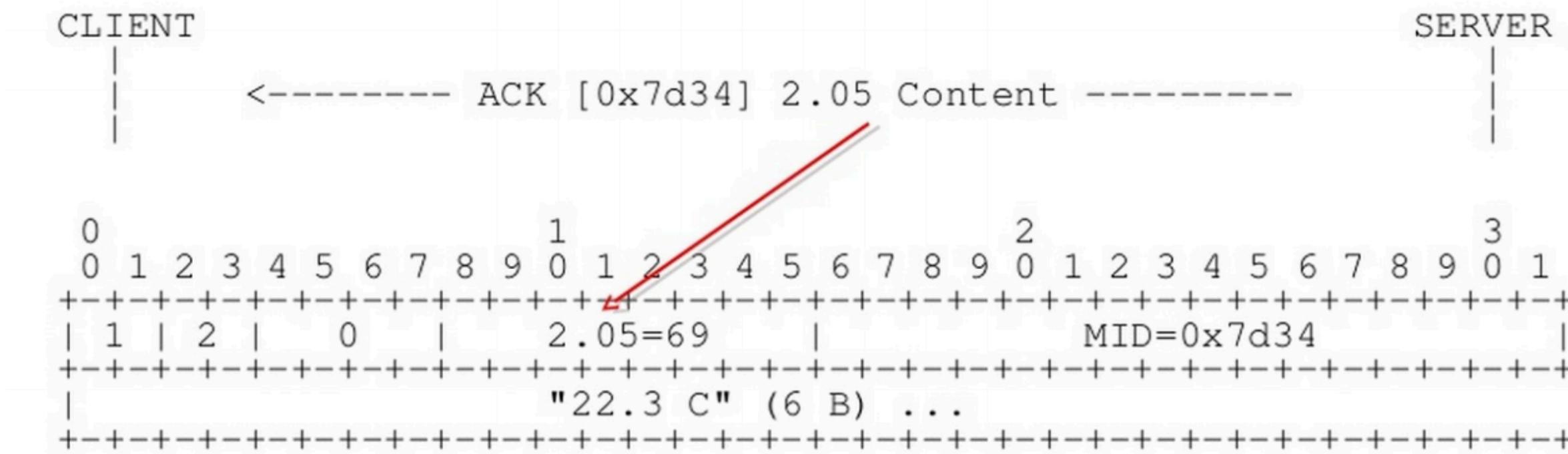
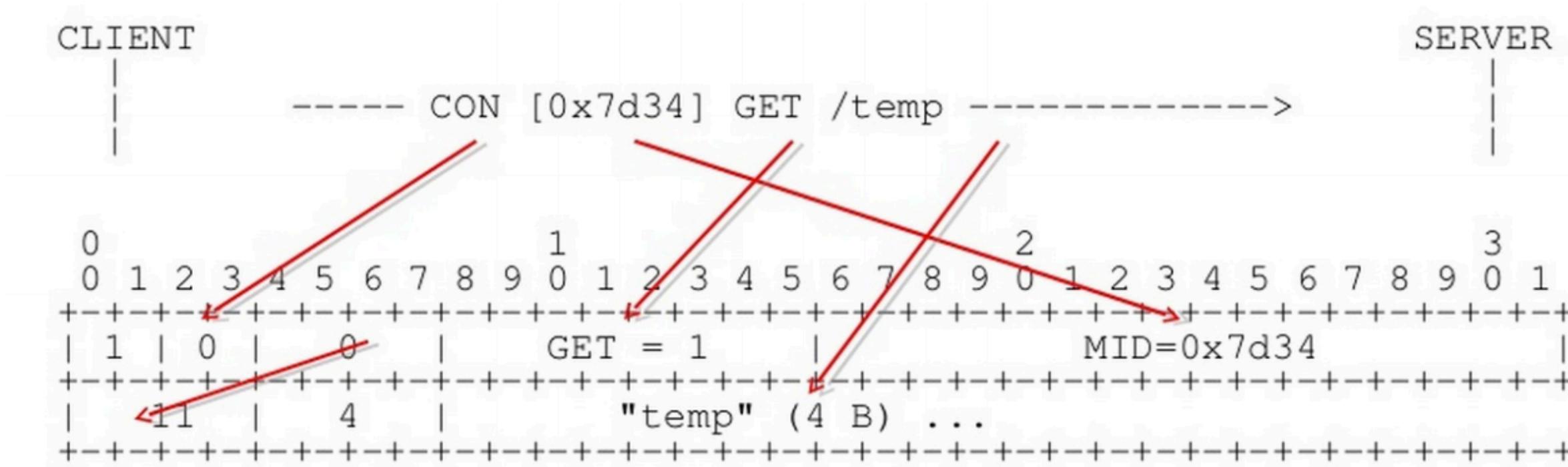
# Data Loss



# Separate Response



# Bits & Bytes

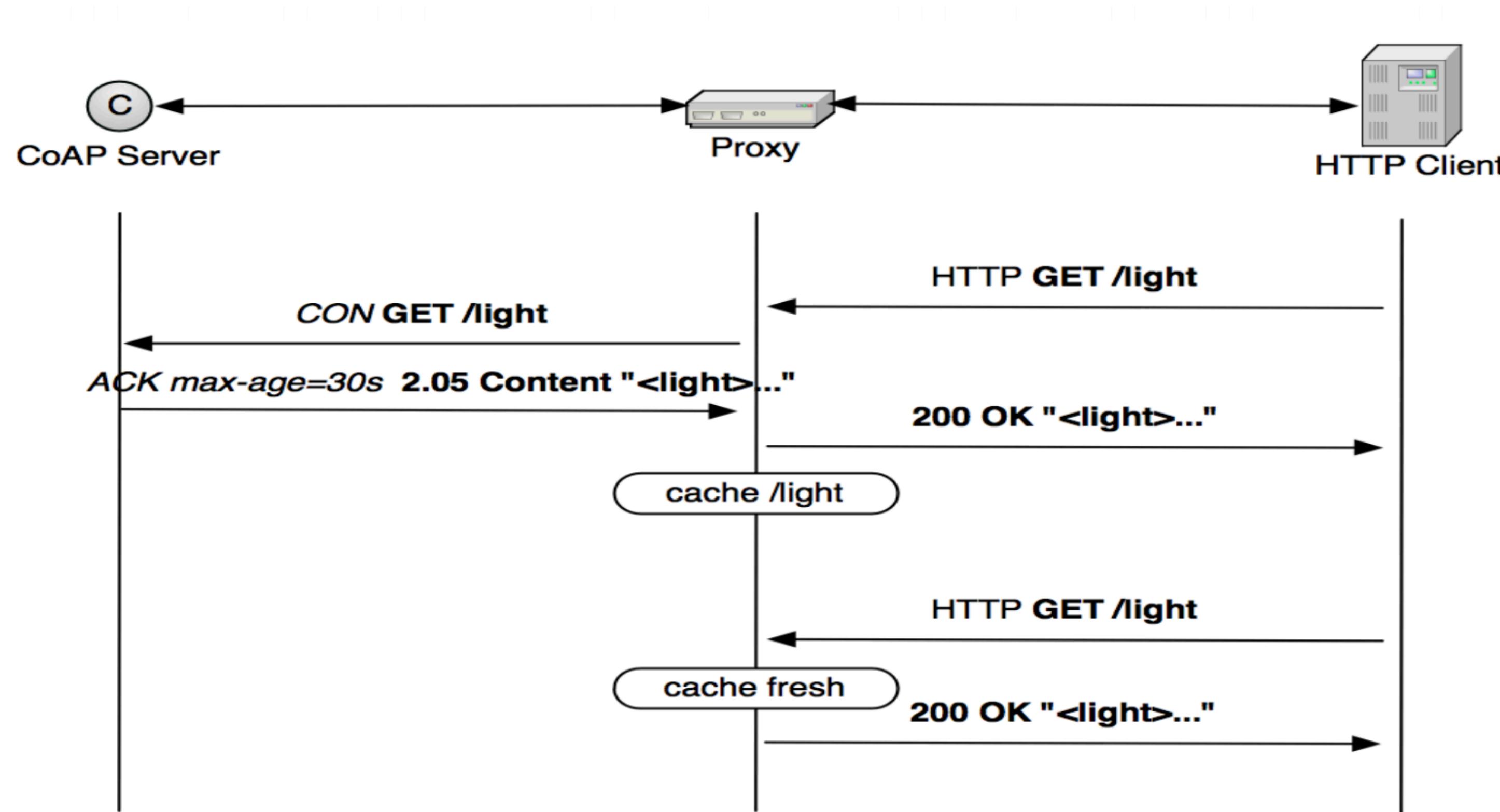


# Caching

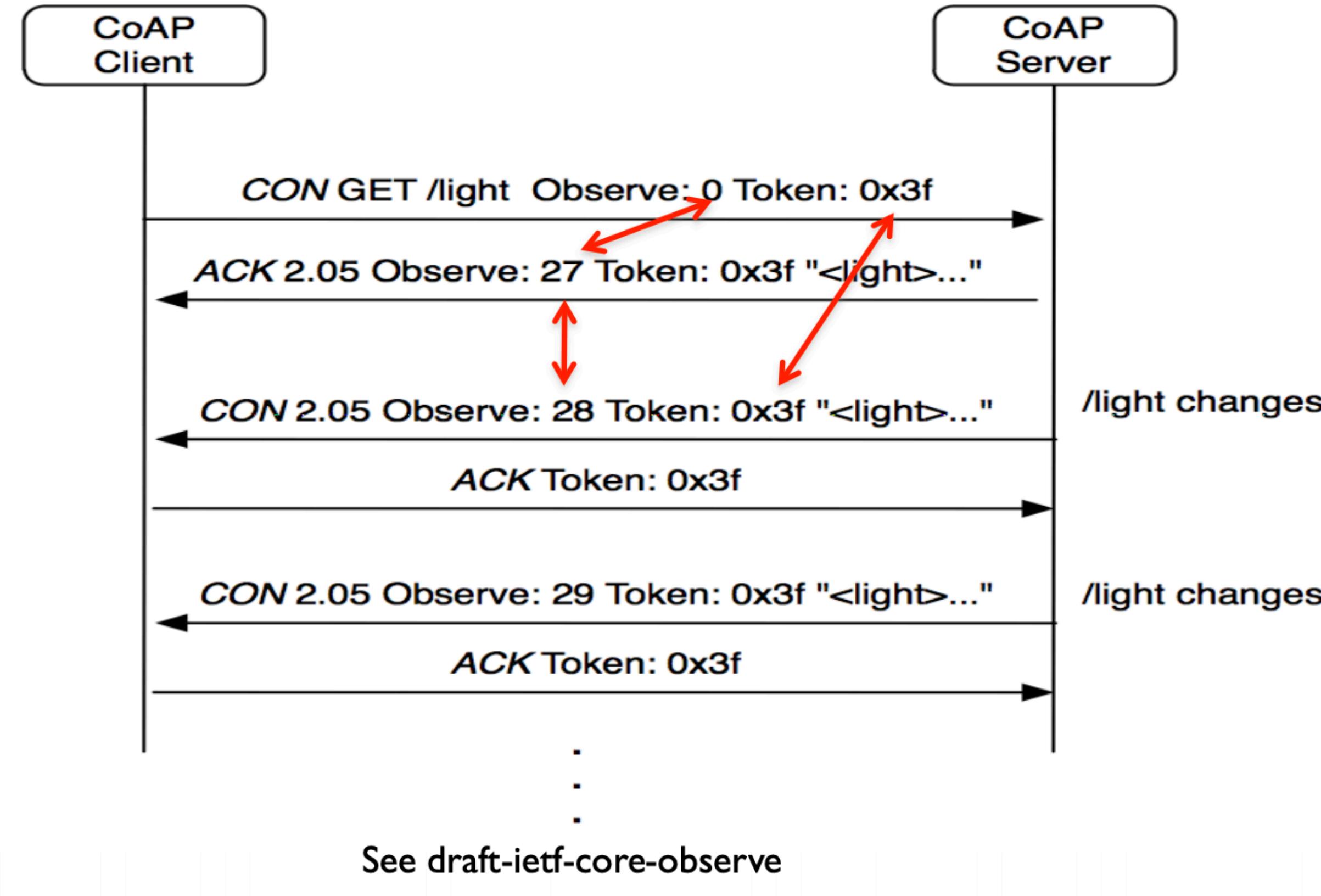
- CoAP includes a simple caching model
  - Determined by response code
  - An option number mask determines if it is a cache key
- Freshness model
  - Max-Age option indicates cache lifetime
- Validation model
  - Validity checked using the Etag Option
- A proxy often supports caching
  - Usually on behalf of a constrained node,
    - a sleeping node,
    - or to reduce network load.



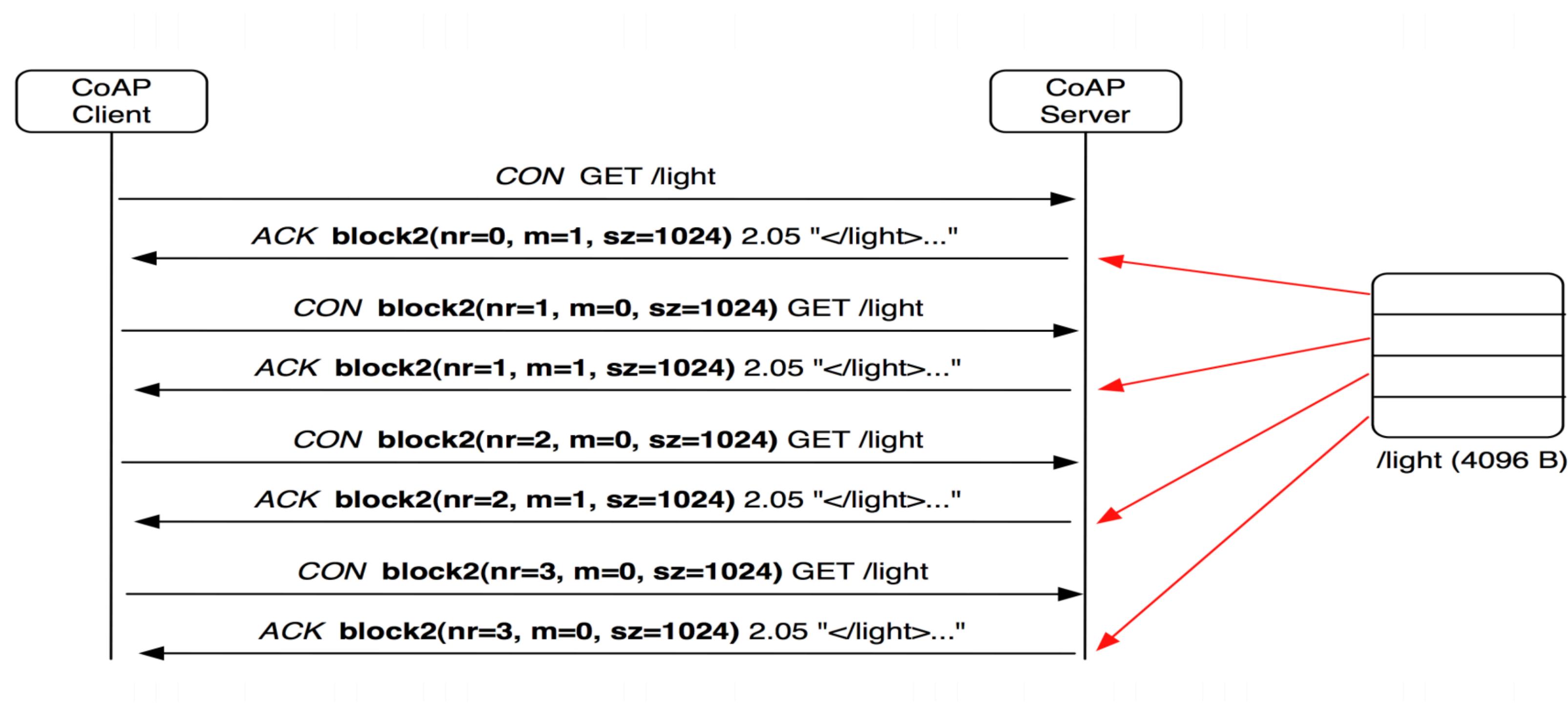
# Proxy



# Subscription



# Block Transfer

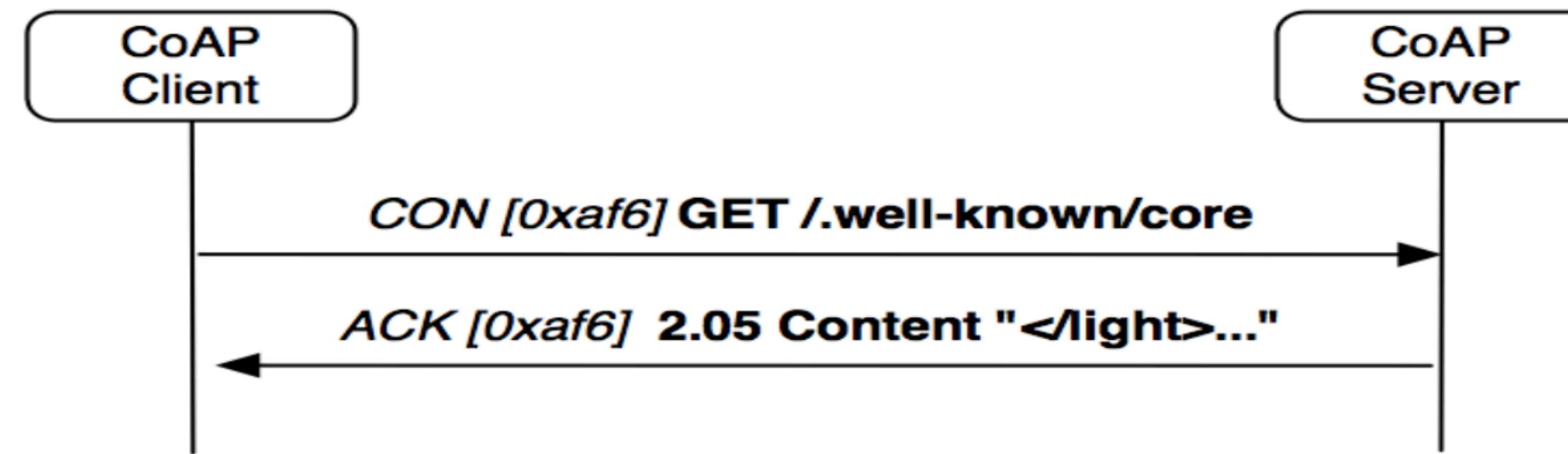


# Community & Open Source

- There are many open source implementations available
  - mbed includes CoAP support
  - Java CoAP Library Californium
  - C CoAP Library Erbium
  - libCoAP C Library
  - jCoAP Java Library
  - OpenCoAP C Library
  - TinyOS and Contiki include CoAP support
- CoAP is already part of many commercial products/systems
  - ARM Sensinode NanoService
  - RTX 4100 WiFi Module
- Firefox has a CoAP plugin called Copper
- Wireshark has CoAP dissector support
- Implement CoAP yourself, it is not that hard!



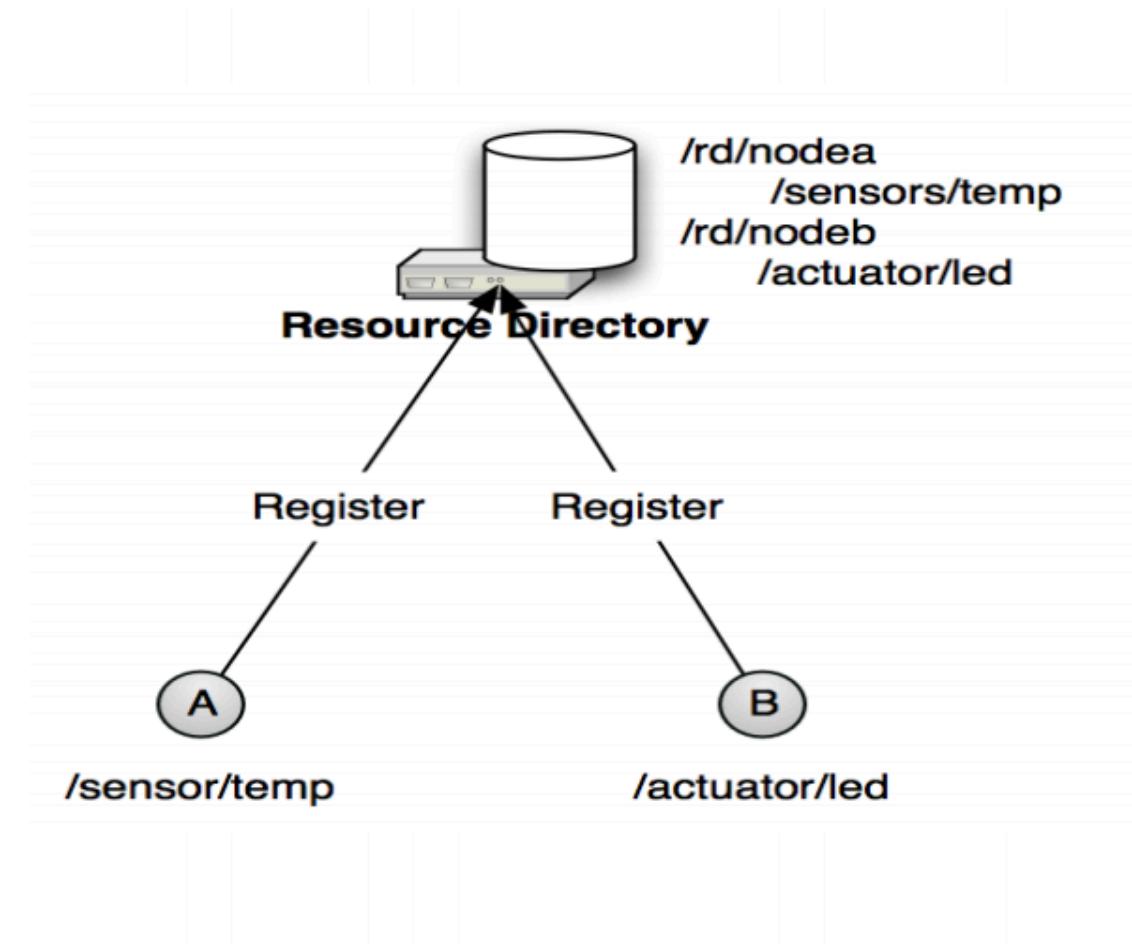
# Resource Discovery



```
</dev/bat>;obs;rt="ipso:dev-bat";ct="0",
</dev.mdl>;rt="ipso:dev-mdl";ct="0",
</dev/mfg>;rt="ipso:dev-mfg";ct="0",
</pwr/0/rel>;obs;rt="ipso:pwr-rel";ct="0",
</pwr/0/w>;obs;rt="ipso:pwr-w";ct="0",
</sen/temp>;obs;rt="ucum:Cel";ct="0"
```

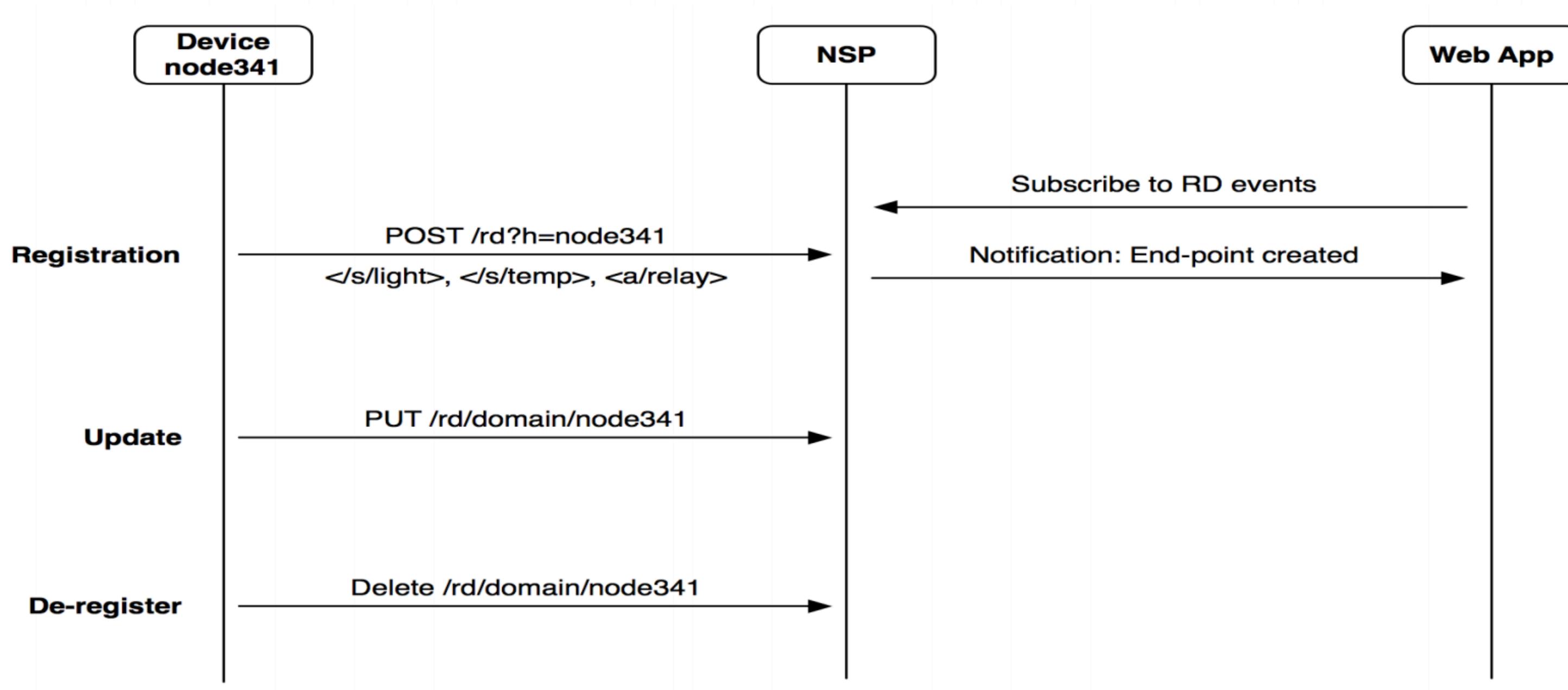
# Resource Directory

- Link Format only defines
  - The link format
  - Peer-to-peer discovery
- A directory approach is also useful
  - Supports sleeping nodes
  - No multicast traffic, longer battery life
  - Remote lookup, hierarchical and federated distribution
- The CoRE Link Format can be used to build Resource Directories
  - Nodes POST (register) their link-format to an RD
  - Nodes PUT (refresh) to the RD periodically
  - Nodes may DELETE (remove) their RD entry
  - Nodes may GET (lookup) the RD or resource of other nodes



DEMO

# Resource Directory



# Lecture outcomes

- CoAP Protocol.
- Practice using a sample.

