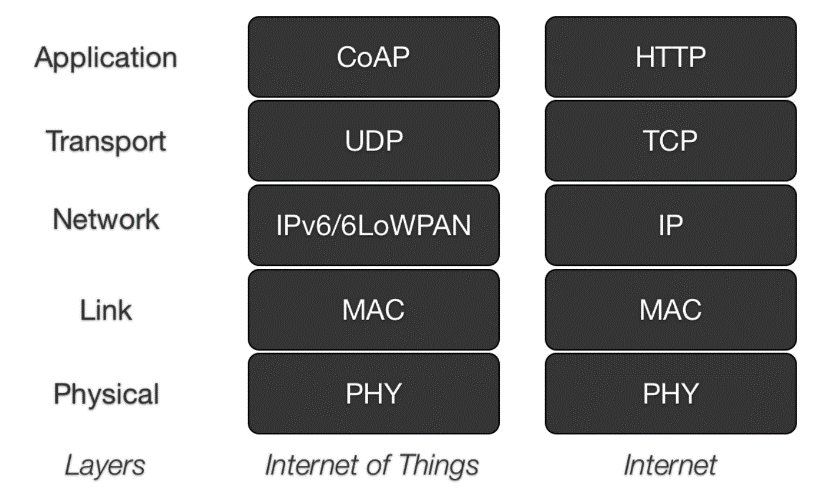
# CoAP IIoT Protocol

Born with the purpose to suit better than http and tcp as protocol.

In fact http introduce overhead while tcp lose time insetting up and maintaining connection.

CoAP stands for **Constrained Application Protocol**.

* Designed by the IETF Constrained RESTful Environments (CoRE) Working group – RFC 7252
* RESTfulProtocol
* Binary
* Maps to HTTP to guarantee integration with “www”
* Runs on lightweight **udp** (over TCP is WiP)
* **IoT** oriented features
  + asynchronous message exchange
  + multicast communication
* CoAP URIs use coap or coaps scheme
* Offers reliability/retransmission
* Message deduplication
* DTLS as secure transport
* No pub/sub use requests or response

Messages include a MsgID (16 bits) used to detect duplicates and for reliability

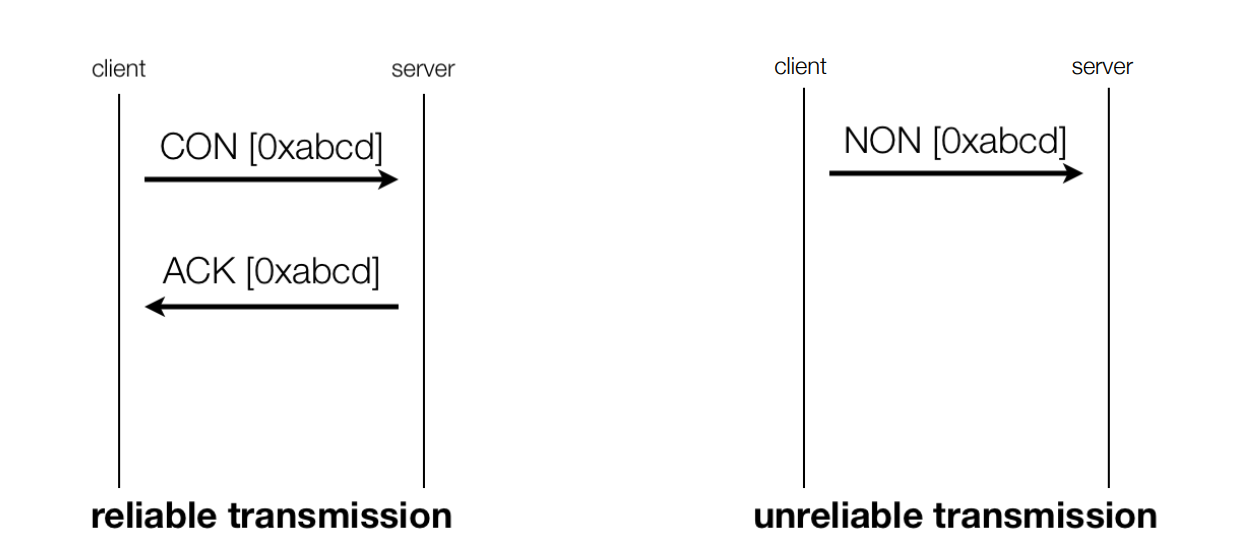
Req and Resp are matched using a Token (0 to 8 bytes)

Retransmission with exponential back-off (**timeout is doubled** at each retransmission)

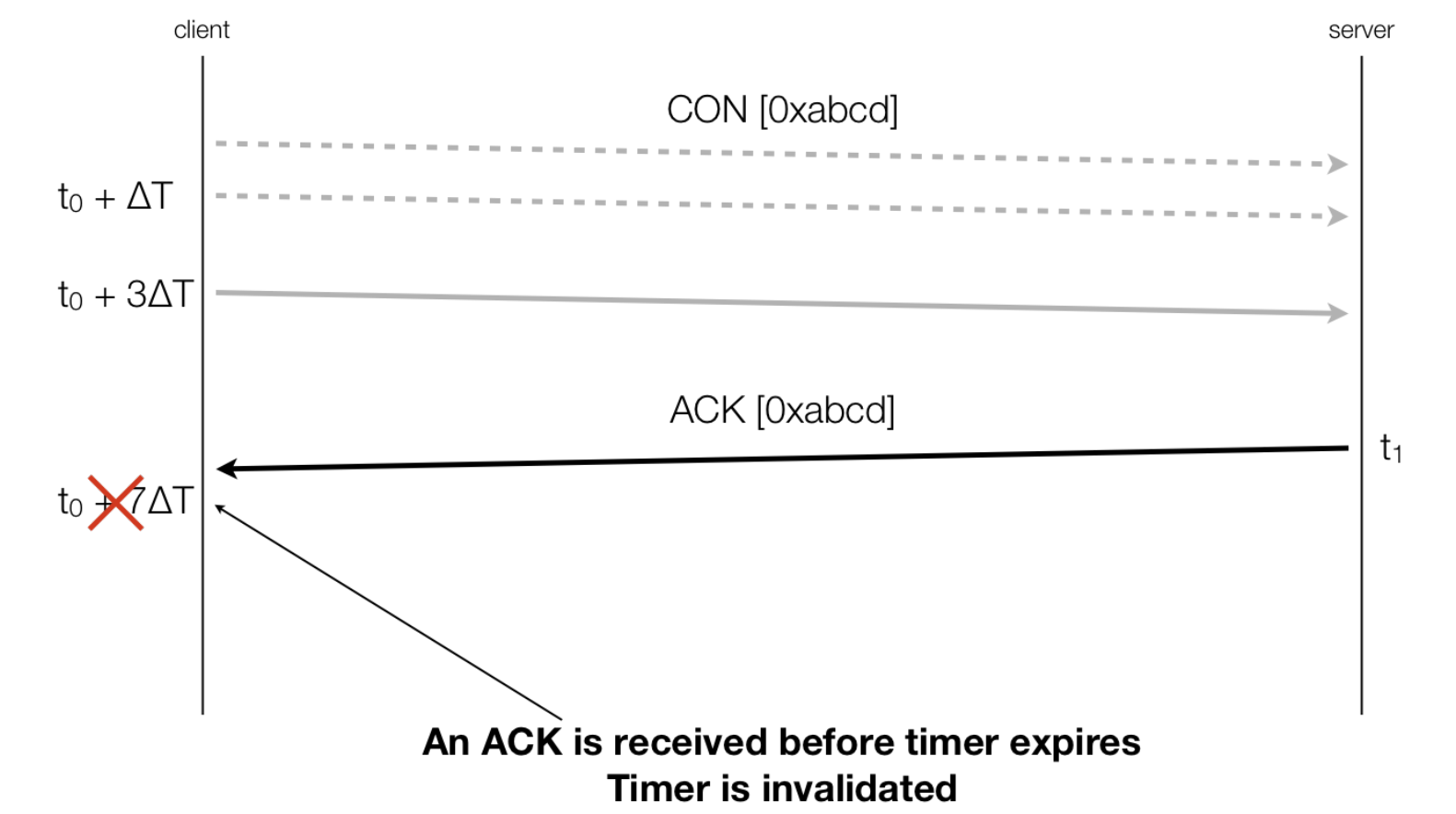
**Max 4 retransmission**

4 kinds of message

* Confirmable **CON** used for message that must be transmitted reliably
* Non Confirmable **NON** no need of reliability.
* Acknowledgment **ACK** used ti acknowledge a CON
* Reset **RST** reset a reliable transmission



**CoAP reliable transmission**

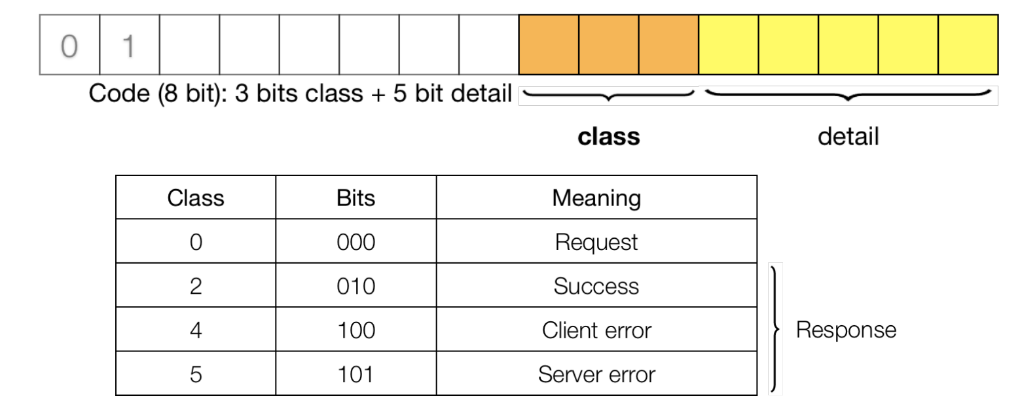


A response can be

* Piggy-backed (ACK the request) the server can respond immediately
* Separate send the ACK and then the response

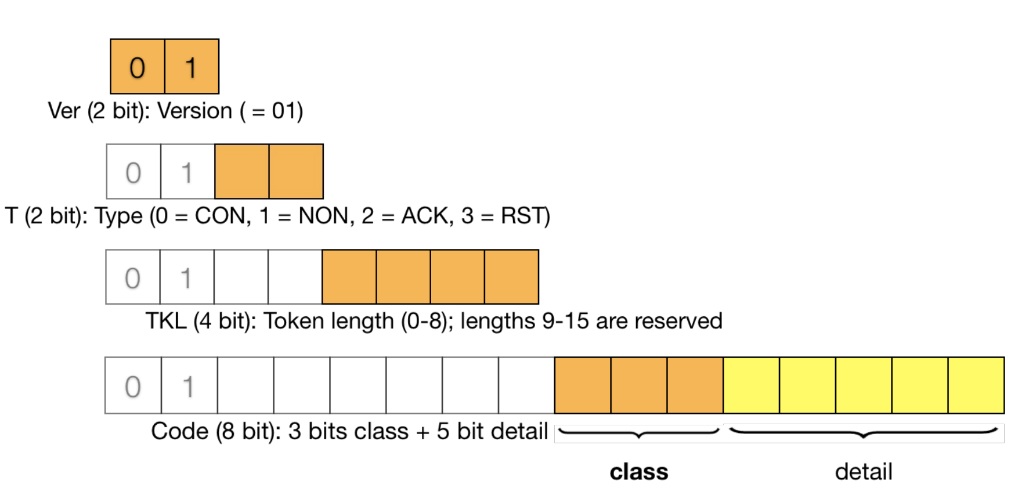
The Detail space Is used in case of request as

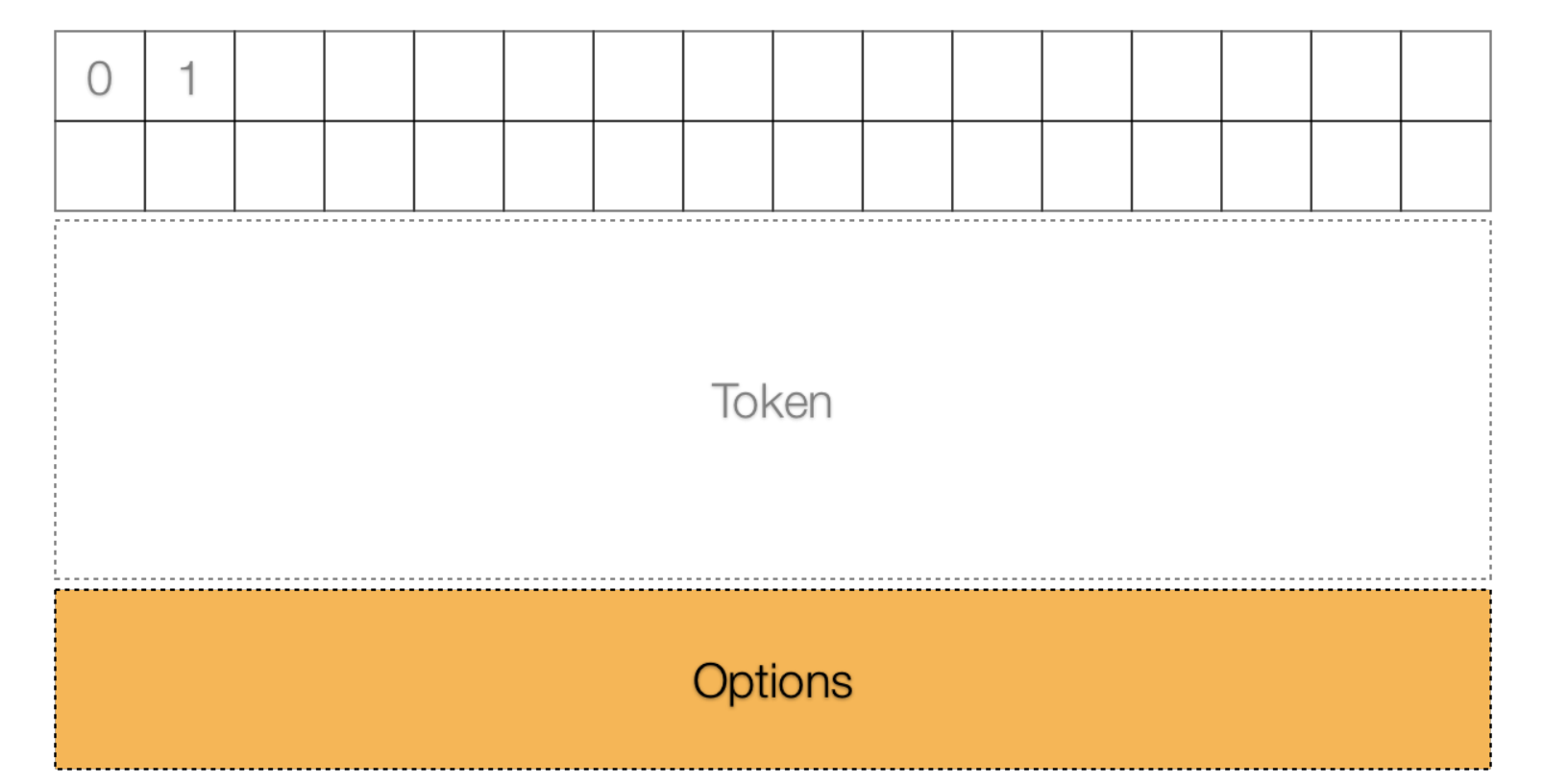
1. GET
2. POST
3. PUT
4. DELETE

****In case of reponse

As

2.01 Created, 2.02 Deleted, 2.04 Changed, 2.05 Content ○ 4.00 Bad Request, 4.02 Bad Option, 4.04 Not Found, 4.05 Method Not Allowed, ... ○ 5.00 Internal Server Error, 5.01 Not Implemented, ...



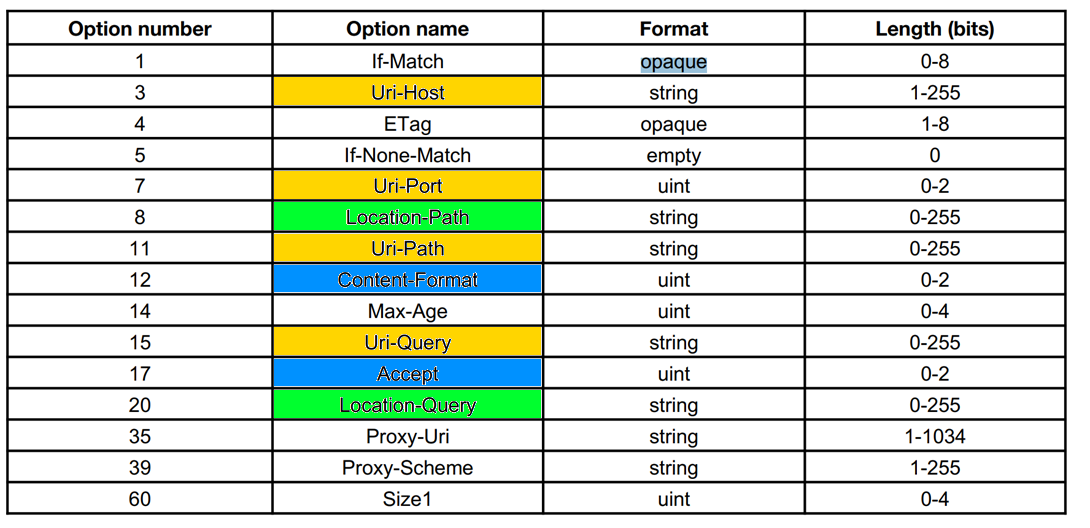


**CoAP Options**

Optional 🡪 Like URI and payload media type

Every option define his number and the length of the Option Value and the option value itself

In order to maximize compactness, options are encoded in a very efficient way, called **delta encoding**

****

The Uri-Host, Uri-Port, Uri-Path, and Uri-Query options are used to specify the target resource of a request to a CoAP origin server

● Uri-Path and Uri-Query options are repeatable

● The **Uri-Host** Option specifies the Internet host of the resource being requested

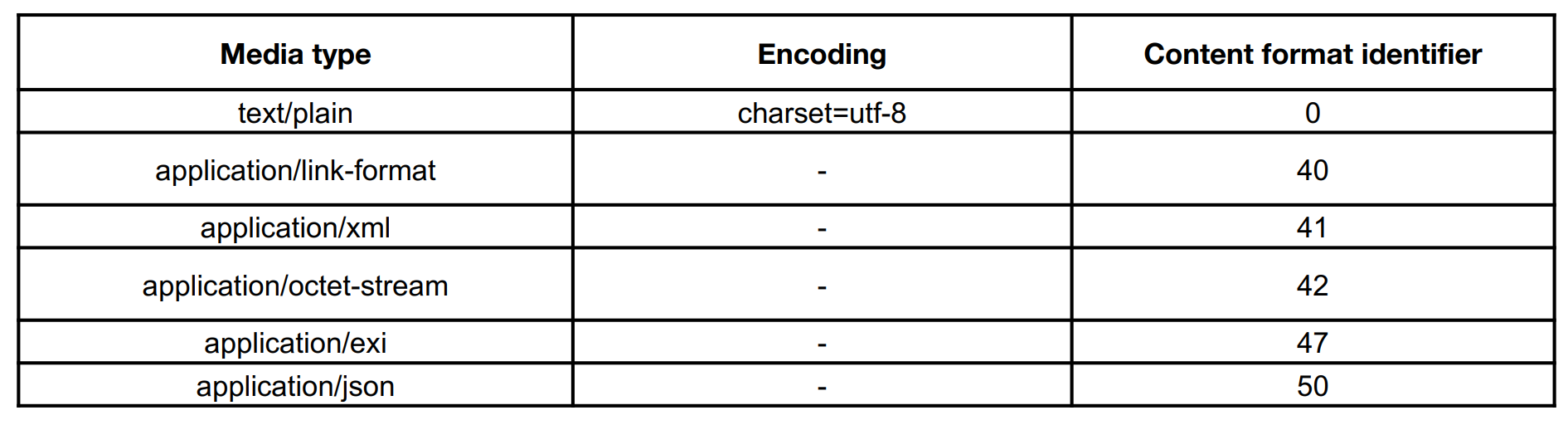
● The **Uri-Port** Option specifies the transport layer port number of the resource

● Each **Uri-Path** Option specifies one segment of the absolute path to the resource

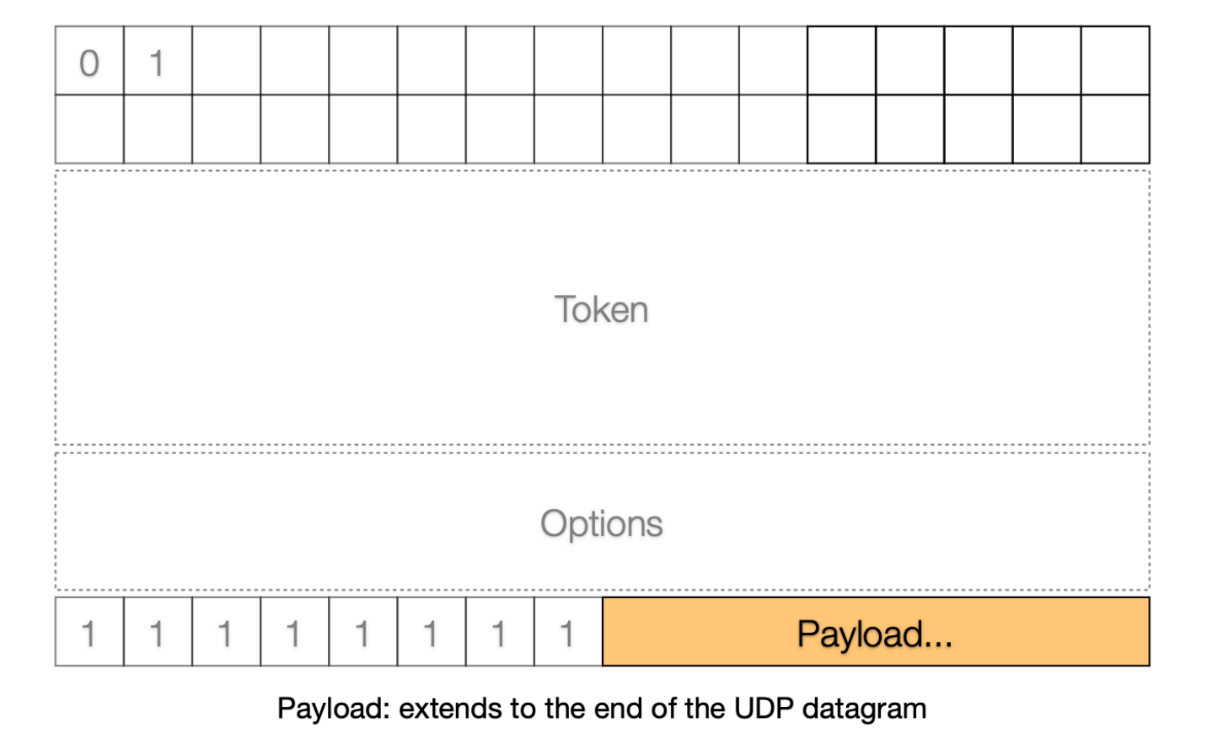
● Each **Uri-Query** Option specifies one argument parameterizing the resource

● The **Location-Path** and **Location-Query** options together indicate a relative URI that consists either of an absolute path, a query string or both

● **Accept** can be used to mark the accepted formats Option: **Content-Format**

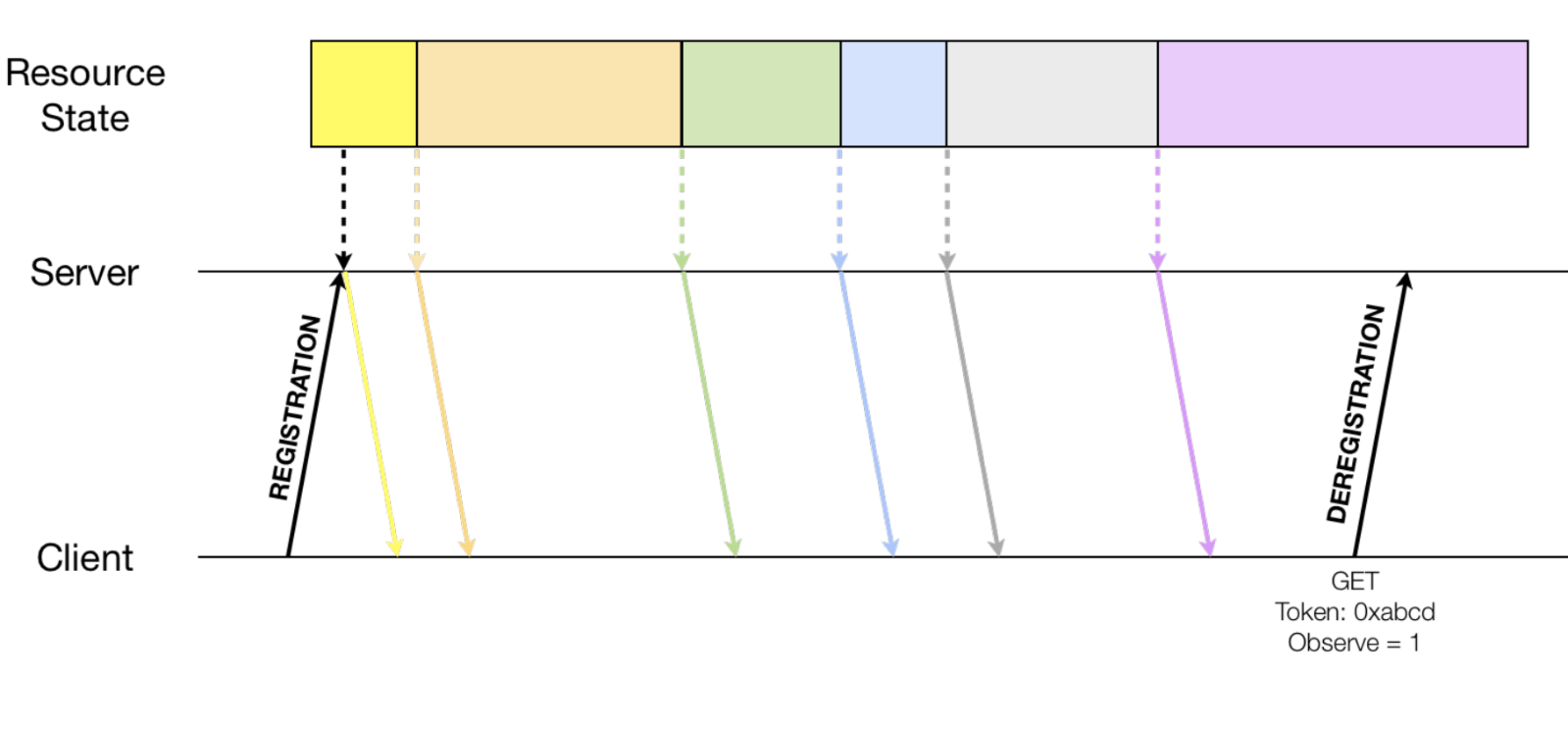


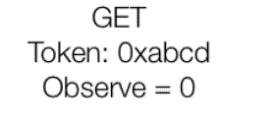
**CoAP Messsage Format**

****0xff separate the udp diagram in two half, header and payload.

The state of the resource in CoAP can change but due to polling inefficiency because can accumulate GET requests with same response, the observe option has been introduce to avoid polling.

When the option observe is in a GET of a resource tells the serve to send a notification to the client whenever the state of the resource changes.

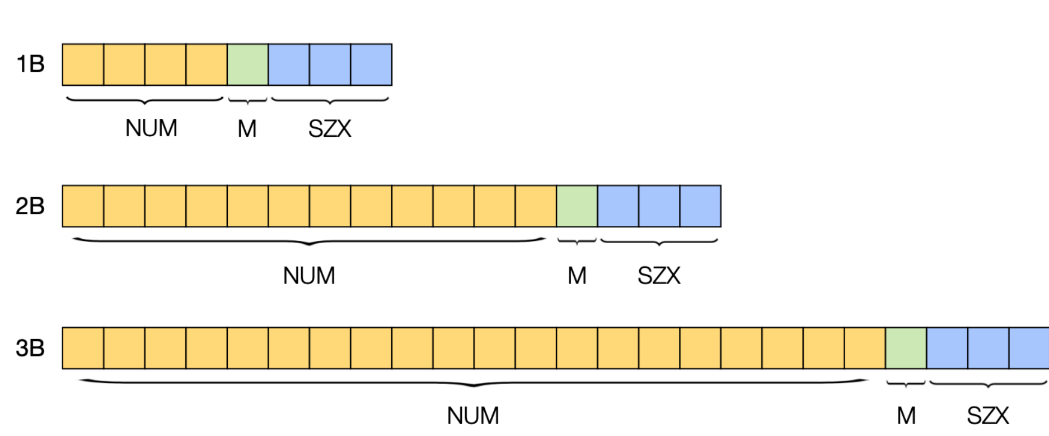




CoAP packet is limited due UDP (65535 bytes)

So exist the **Block Option** 0 to 3 bytes

Contains 3 fields:

* NUM: relative number of the block
* M (1bit) true when there will be blocks following
* SZX (3 bits) size exponent of the block 0-> to 6 (7 is reserved)

The actual size will be

Options:

0/1/128

0 NUM first of the series

1 M means that there will be a next block

128 lenght

