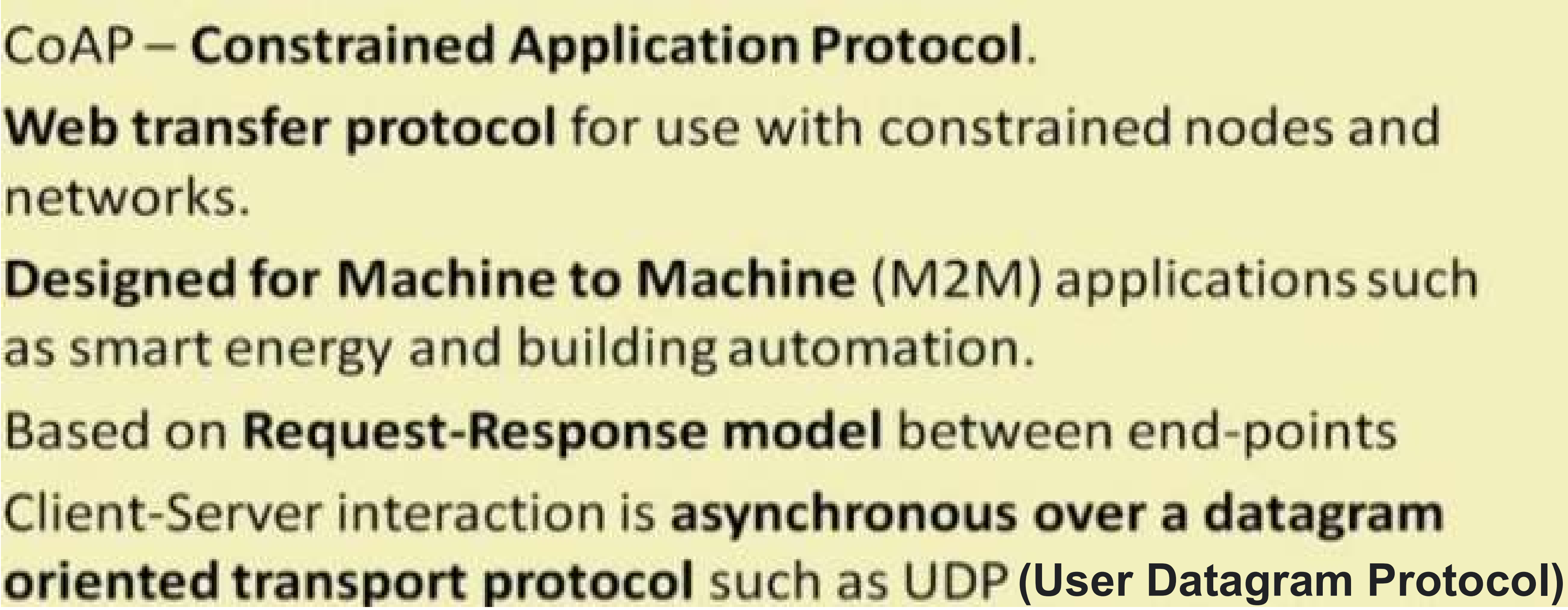
IoT Networking

Constrained Application Protocol (CoAP)



CoAP – **Constrained Application Protocol**.

Web transfer protocol for use with constrained nodes and networks.

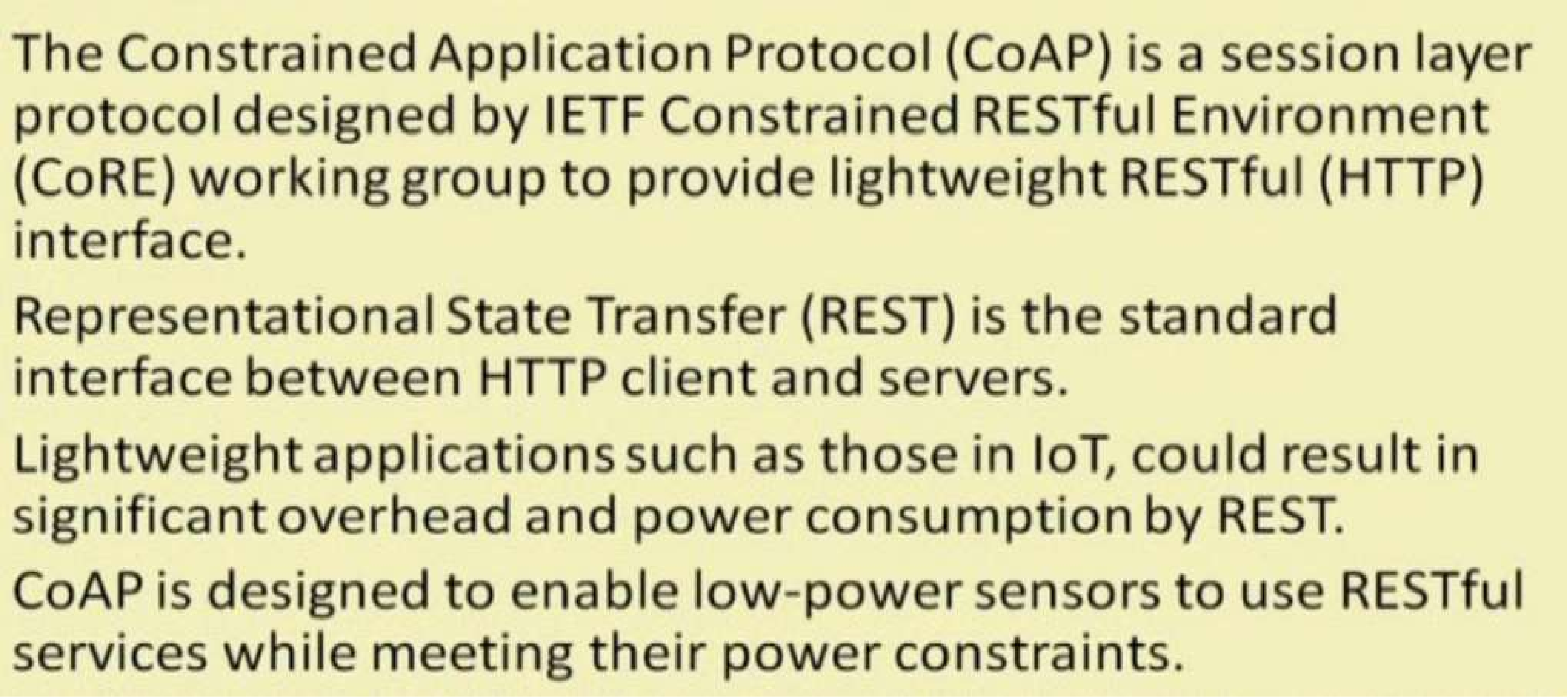
Designed for Machine to Machine (M2M) applications such as smart energy and building automation.

Based on **Request-Response model** between end-points

Client-Server interaction is **asynchronous over a datagram oriented transport protocol** such as UDP **(User Datagram Protocol)**

Ref: https://www.rfc-editor.org/rfc/rfc7252

Constrained Application Protocol (CoAP)



The Constrained Application Protocol (CoAP) is a session layer protocol designed by IETF Constrained RESTful Environment (CoRE) working group to provide lightweight RESTful (HTTP) interface.

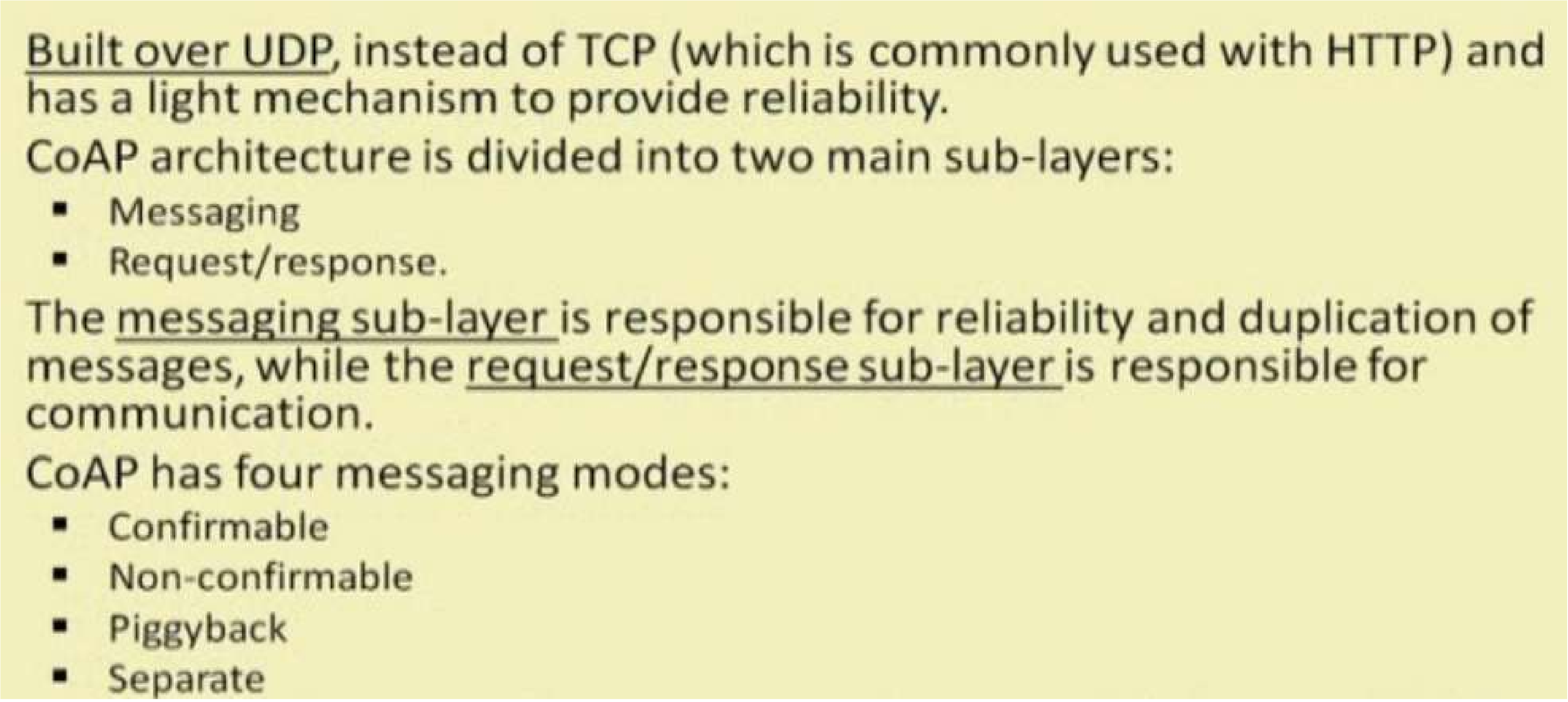
Representational State Transfer (REST) is the standard interface between HTTP client and servers.

Lightweight applications such as those in IoT, could result in significant overhead and power consumption by REST.

CoAP is designed to enable low-power sensors to use RESTful services while meeting their power constraints.

Ref: https://www.rfc-editor.org/rfc/rfc7252

Constrained Application Protocol (CoAP)



Built over UDP, instead of TCP (which is commonly used with HTTP) and has a light mechanism to provide reliability.

CoAP architecture is divided into two main sub-layers:

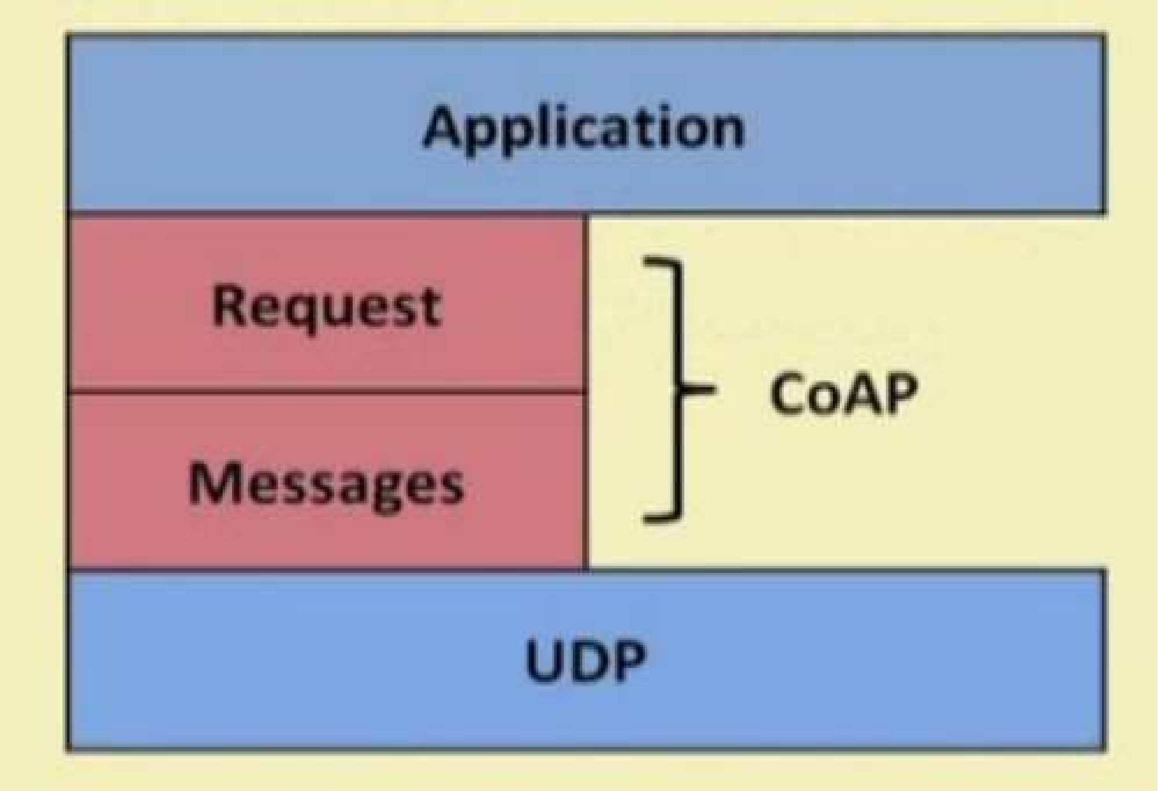
- Messaging
- Request/response.

The messaging sub-layer is responsible for reliability and duplication of messages, while the request/response sub-layer is responsible for communication.

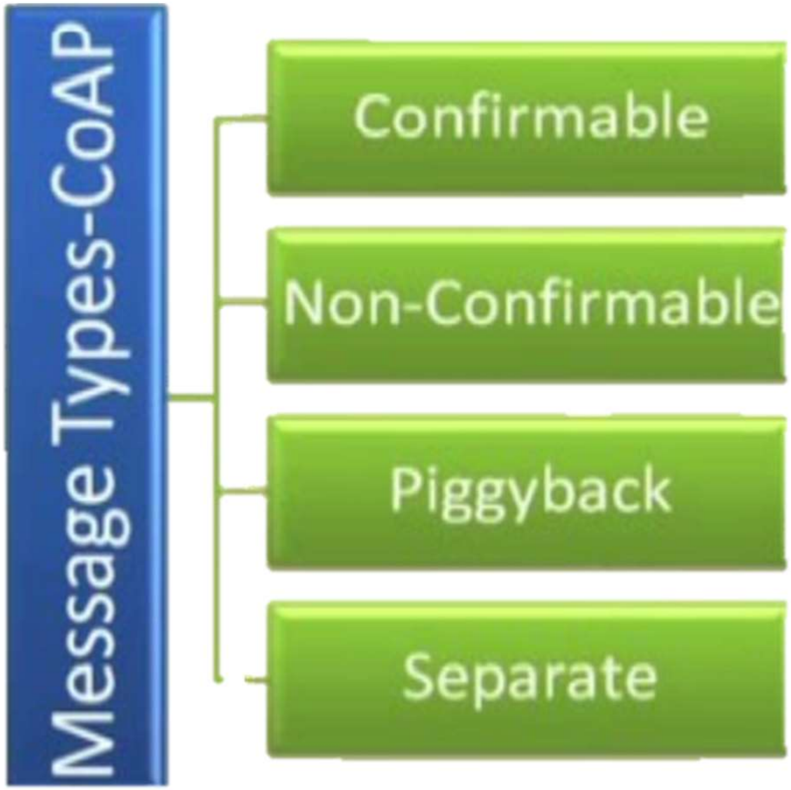
CoAP has four messaging modes:

- Confirmable
- Non-confirmable
- Piggyback
- Separate

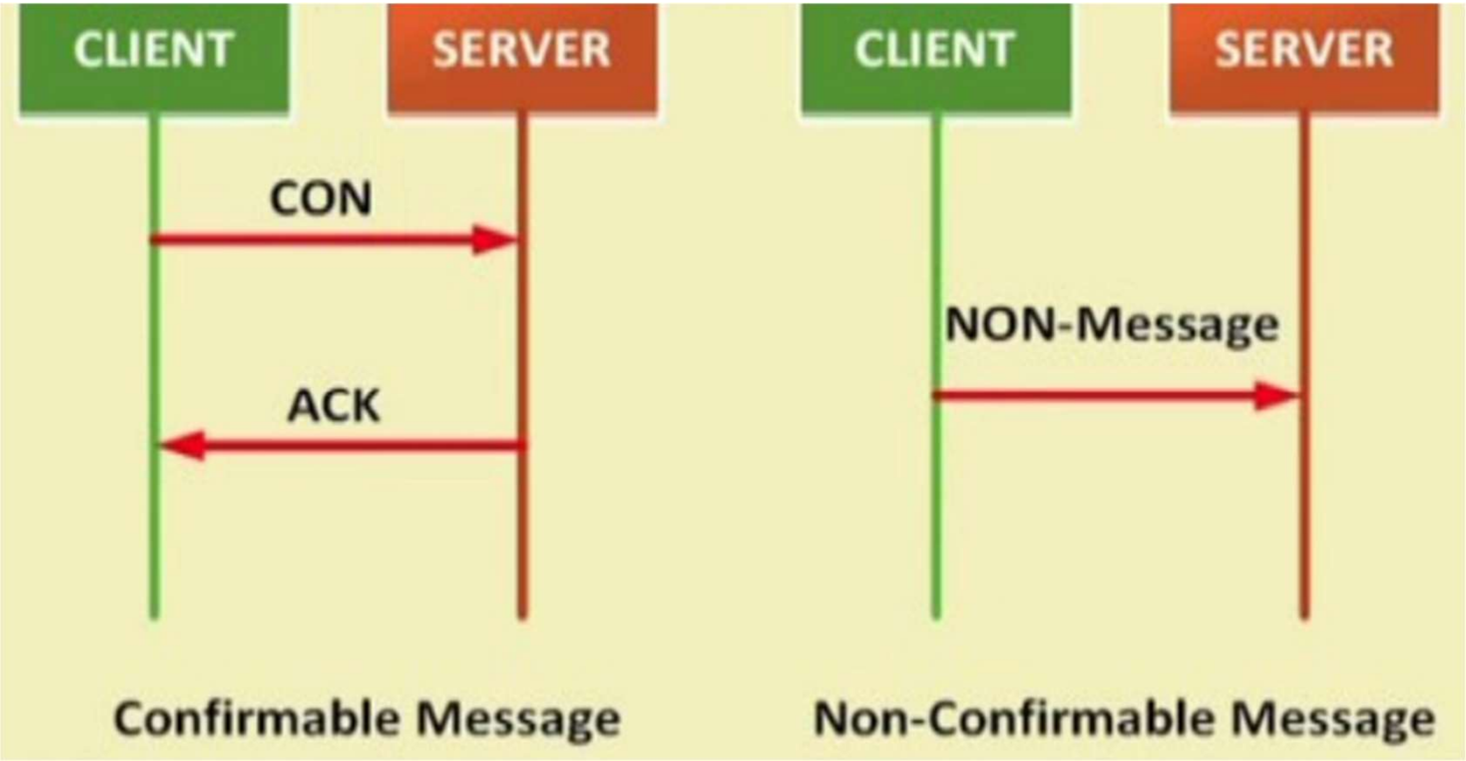
Abstract Layering of CoAP



CoAP Message types

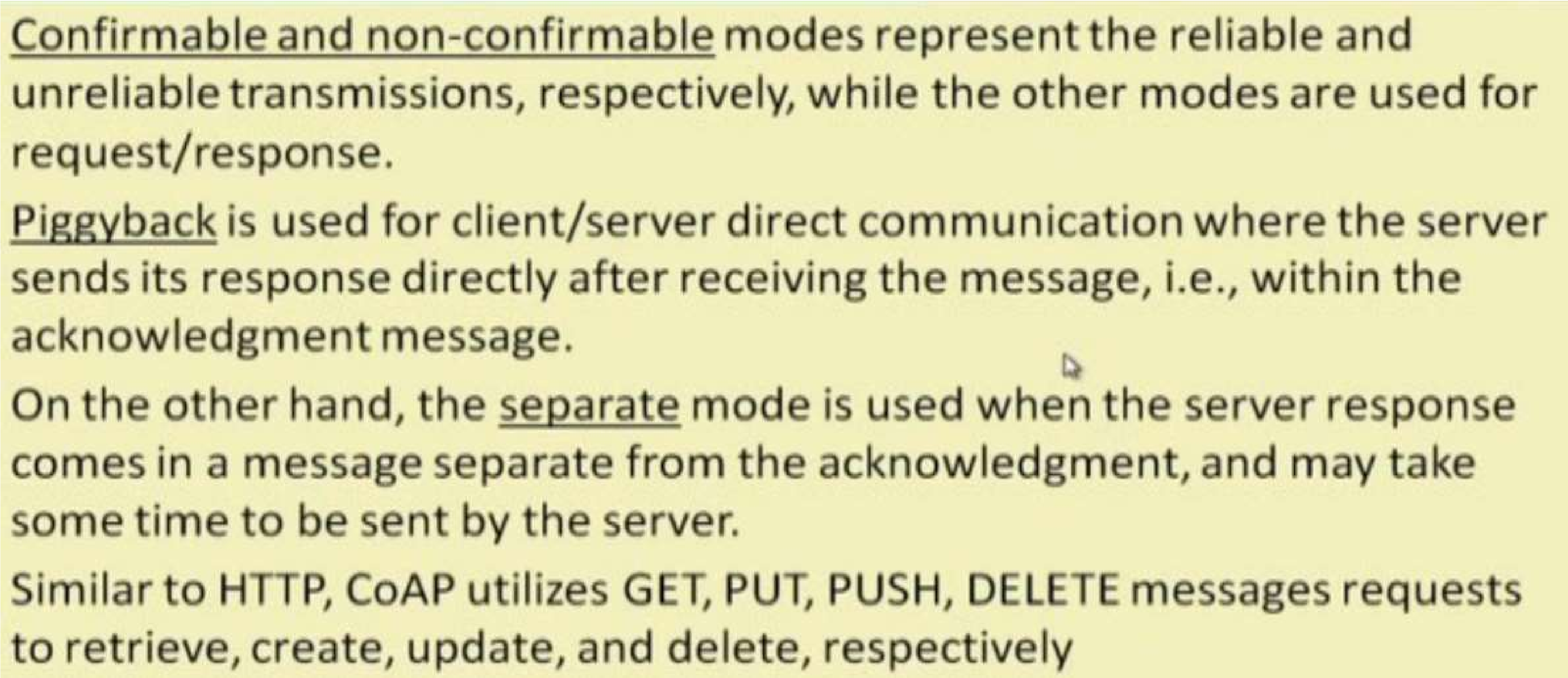


CoAP Request-Response model



Karagiannis, V., et al. (2015). A survey on application layer protocols for the internet of things. *Transaction on IoT and Cloud computing*, *3*(1), 11-17.

CoAP Request-Response model



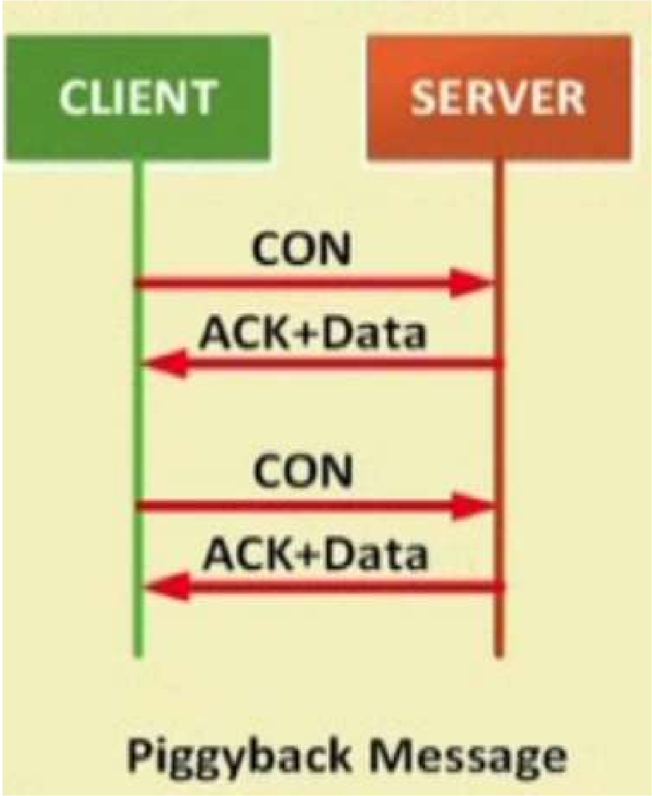
Confirmable and non-confirmable modes represent the reliable and unreliable transmissions, respectively, while the other modes are used for request/response.

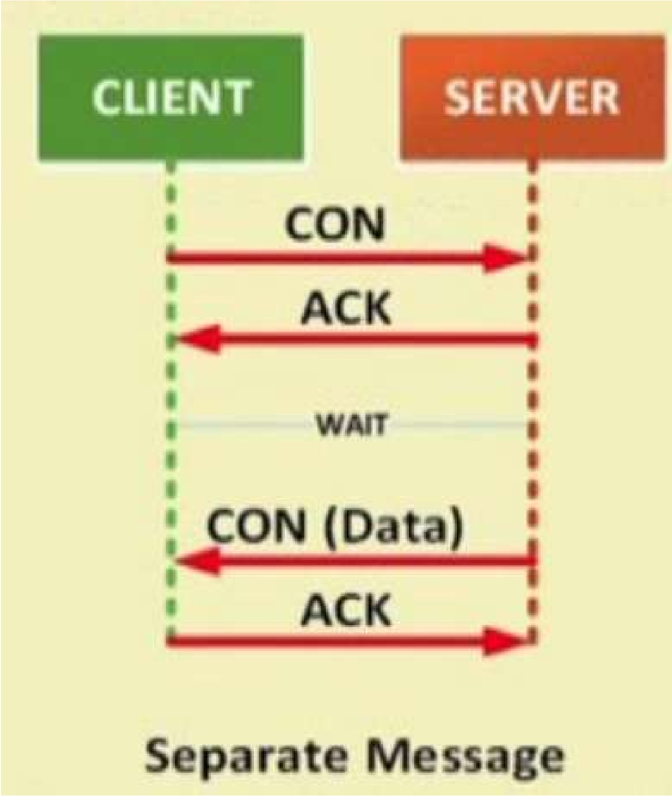
Piggyback is used for client/server direct communication where the server sends its response directly after receiving the message, i.e., within the acknowledgment message.

On the other hand, the separate mode is used when the server response comes in a message separate from the acknowledgment, and may take some time to be sent by the server.

Similar to HTTP, CoAP utilizes GET, PUT, PUSH, DELETE messages requests to retrieve, create, update, and delete, respectively

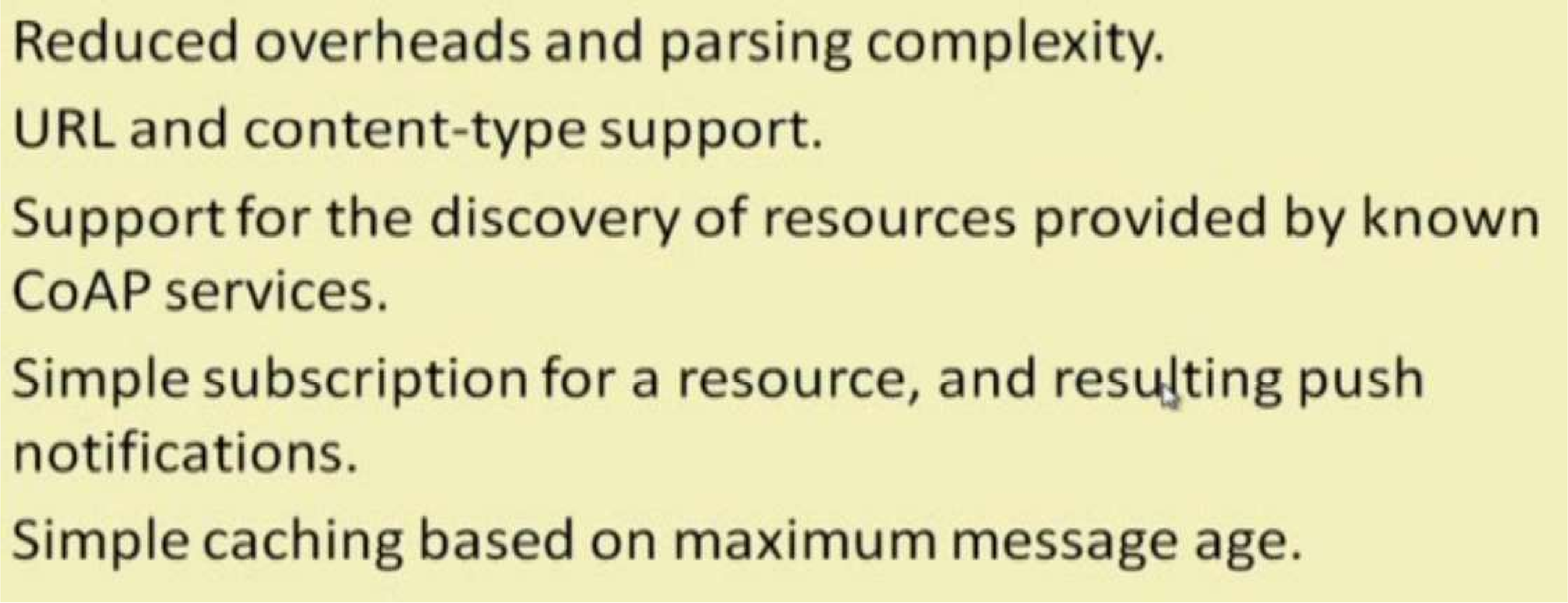
CoAP Request-Response model





Karagiannis, V., et al. (2015). A survey on application layer protocols for the internet of things. *Transaction on IoT and Cloud computing*, *3*(1), 11-17.

Features



Reduced overheads and parsing complexity.

URL and content-type support.

Support for the discovery of resources provided by known CoAP services.

Simple subscription for a resource, and resulting push notifications.

Simple caching based on maximum message age.

Thank you