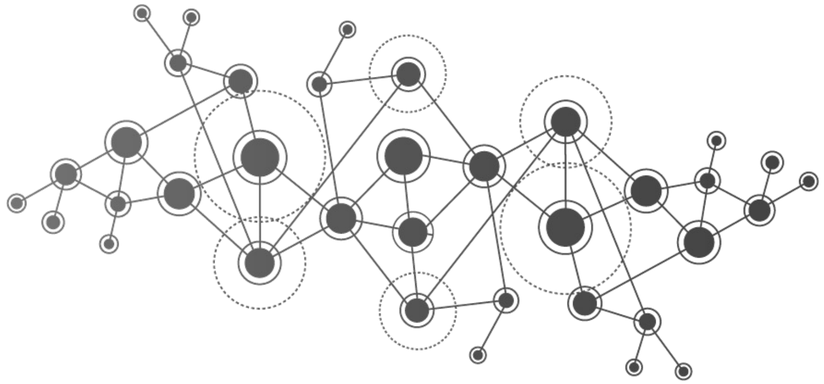
UNIMORE
UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA



Intelligent Internet of Things

Hands-On Session - CoAP

Prof. Marco Picone

A.A 2023/2024

UNIMORE
UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA

Hands-On Session - CoAP

- Simple CoAP Server with String Demo Resource
- Simple GET, POST, PUT Clients
- Asynchronous CoAP Client
- Demo Temperature Resource (with observability)
- Observing Client
- Json Content Format Negotiation

2022 - Marco Picone, Ph.D. (marco.picone@unimore.it)

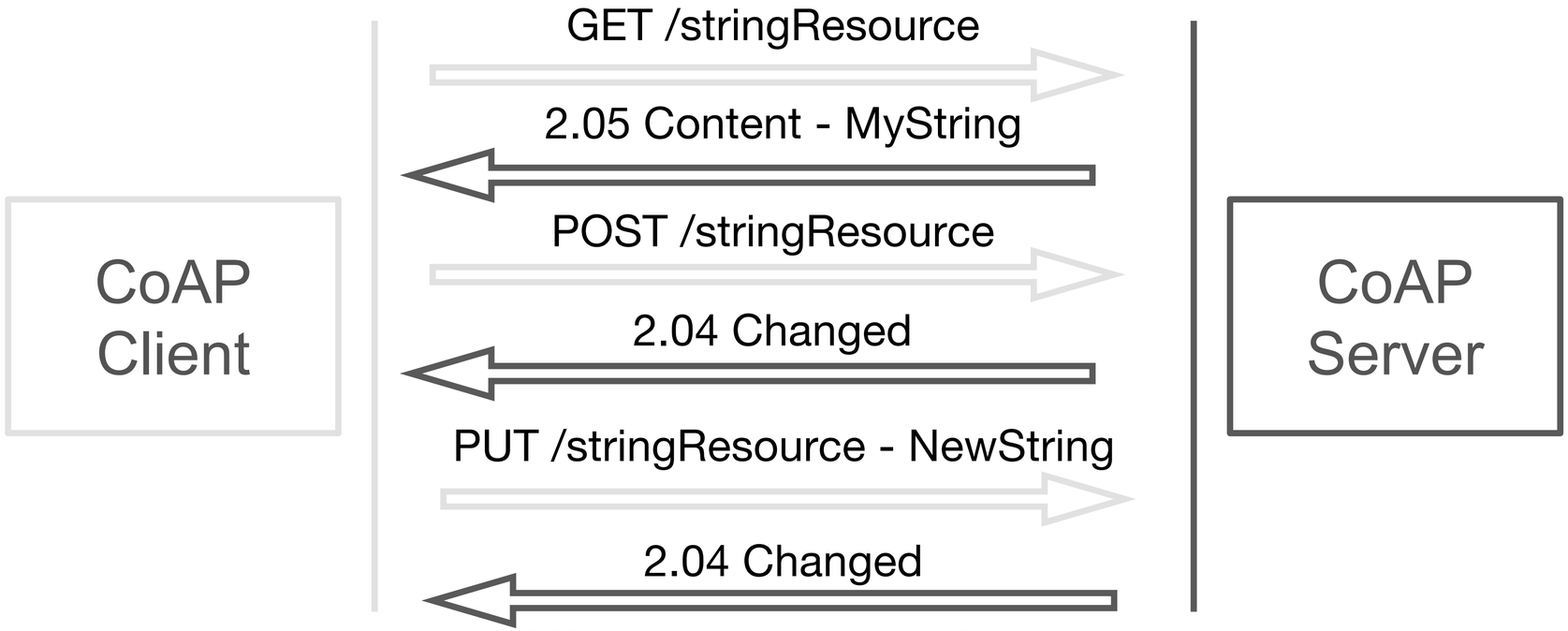
Example 1 - Simple CoAP Server with String Demo Resource

- Create a Simple CoAP Server with the following characteristics
  - Hold a String resource
  - Handle the following requests:

    GET to expose the current value of the resource (2.05 Content)

    POST to update the value with a random String (2.04 Changed)

    PUT to update the value with the received Payload (2.04 Changed)



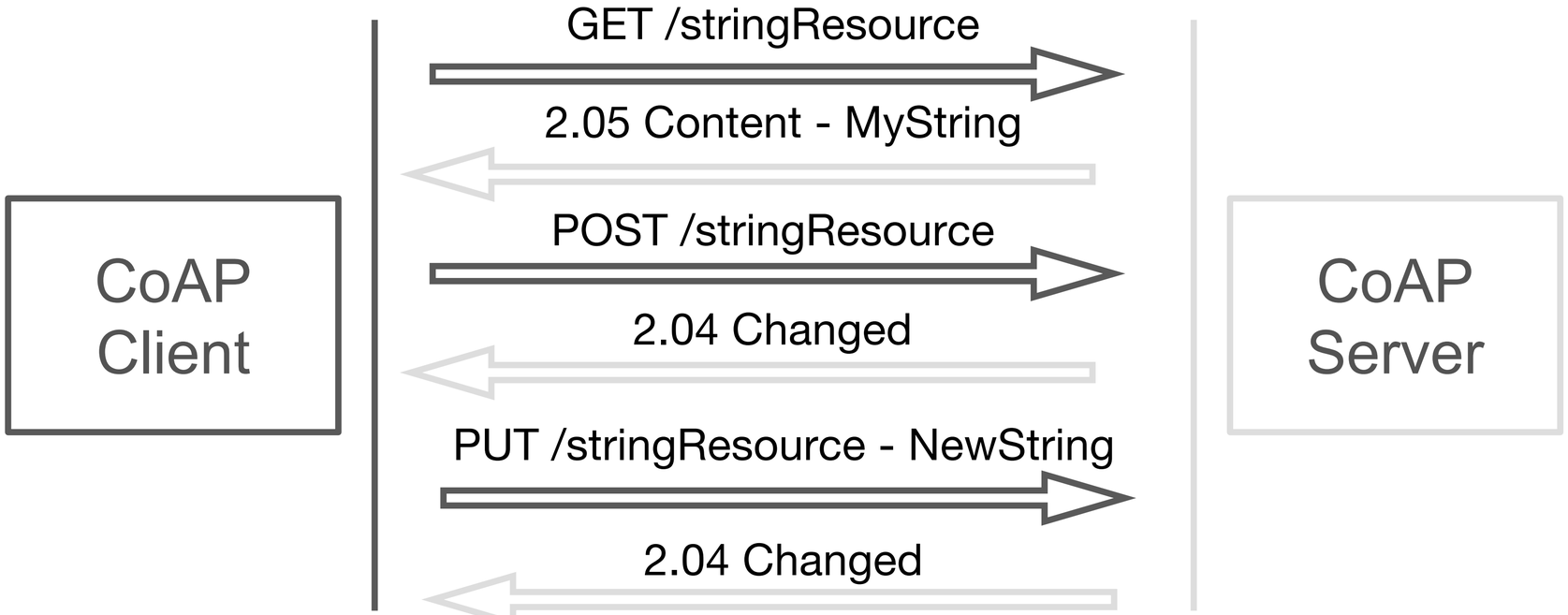
Example 2 - Simple GET, POST, PUT Clients

- Create multiple multiple independent **synchronous** CoAP Clients to interact with the created server. Client must handle:

  GET to retrieve the string value hosted by the server

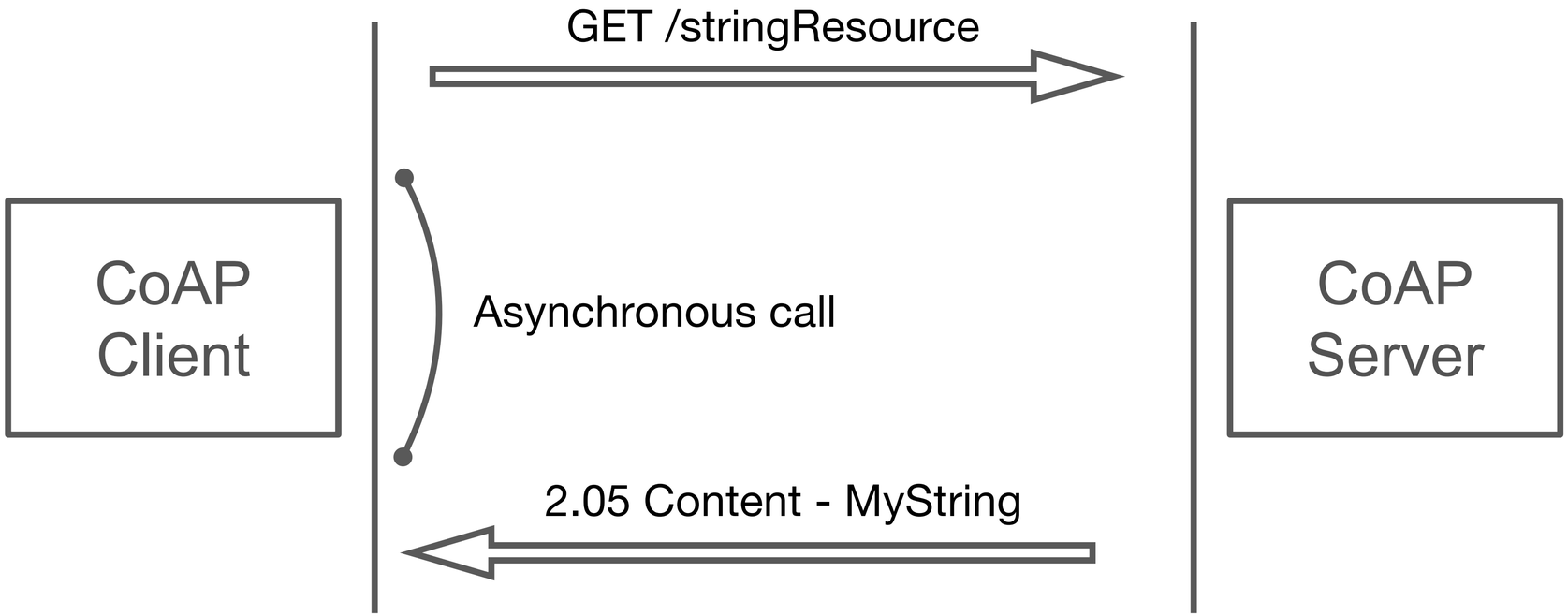
  POST to update the value with a random String generated by the Server (empty request body)

  PUT to update the value with a random String generated by the client and passed as request Body



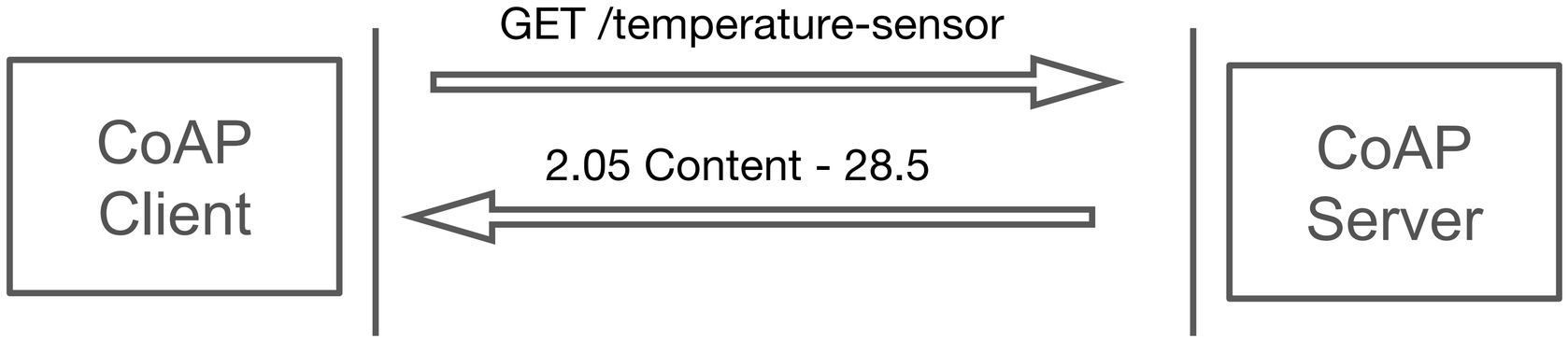
Example 3 - Asynchronous CoAP Client

- Create a single **asynchronous** CoAP Client to interact with the created server handling only the GET to retrieve the string value hosted by the server

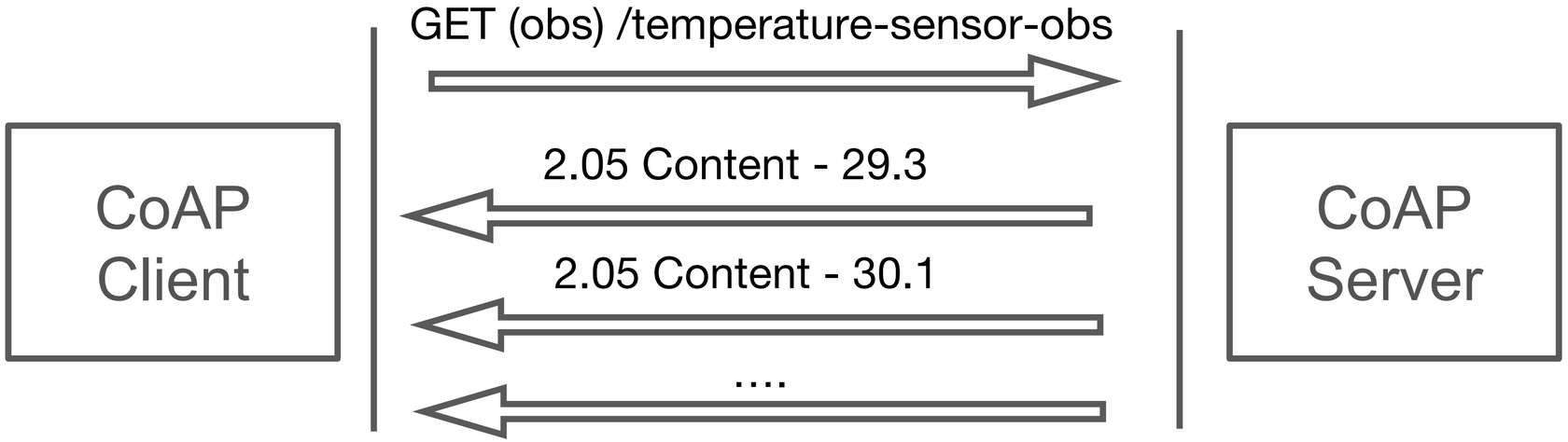


Example 4 - Demo Temperature Resource (with observability)

- Create CoAP Server with a “temperature-sensor” resource hosting a double value representing a demo temperature sensor.
- This resource supports only GET request and for each GET generates a new random temperature value

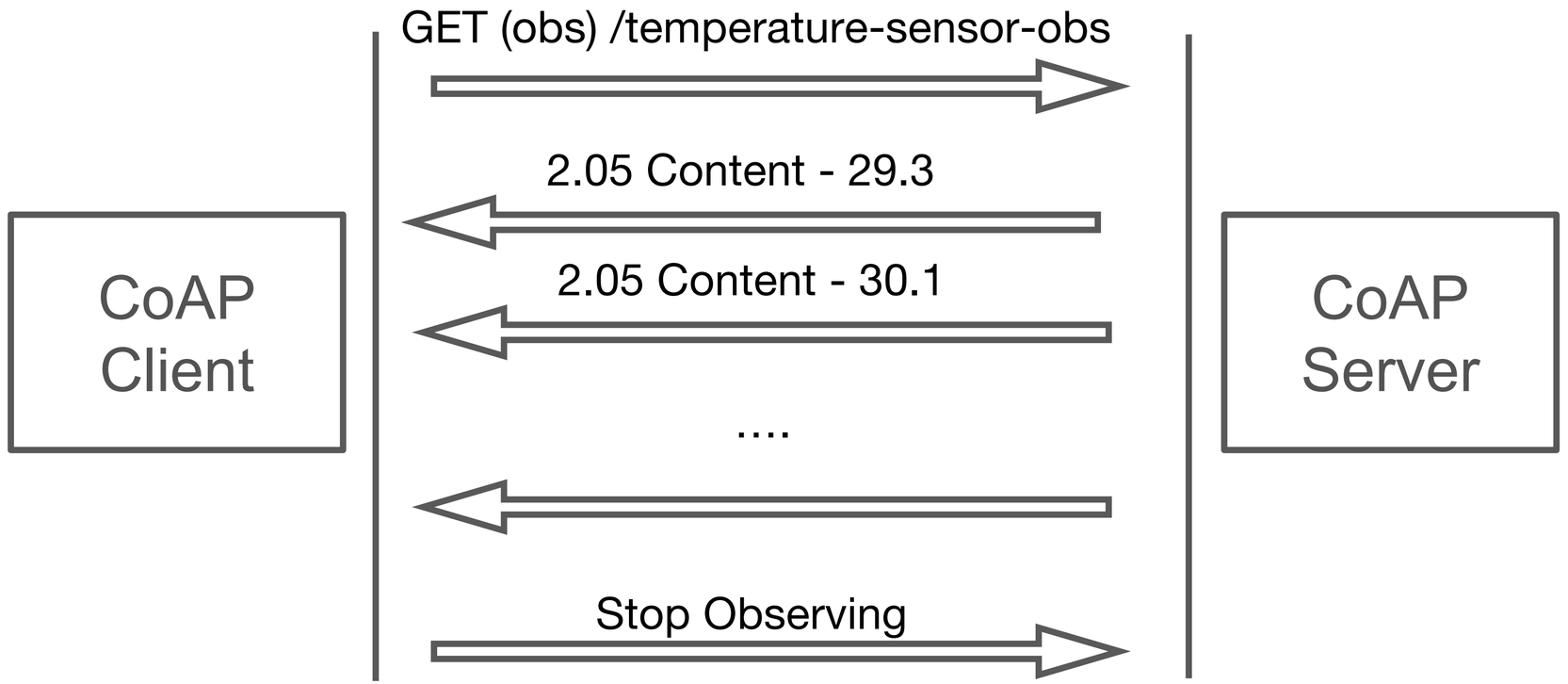


- Add a “temperature-sensor-obs” observable resource hosting a double value representing a demo temperature sensor.
- Periodically (T=1sec) the value is updated and the clients are notified. Each GET Response add the Max-Age Option = T



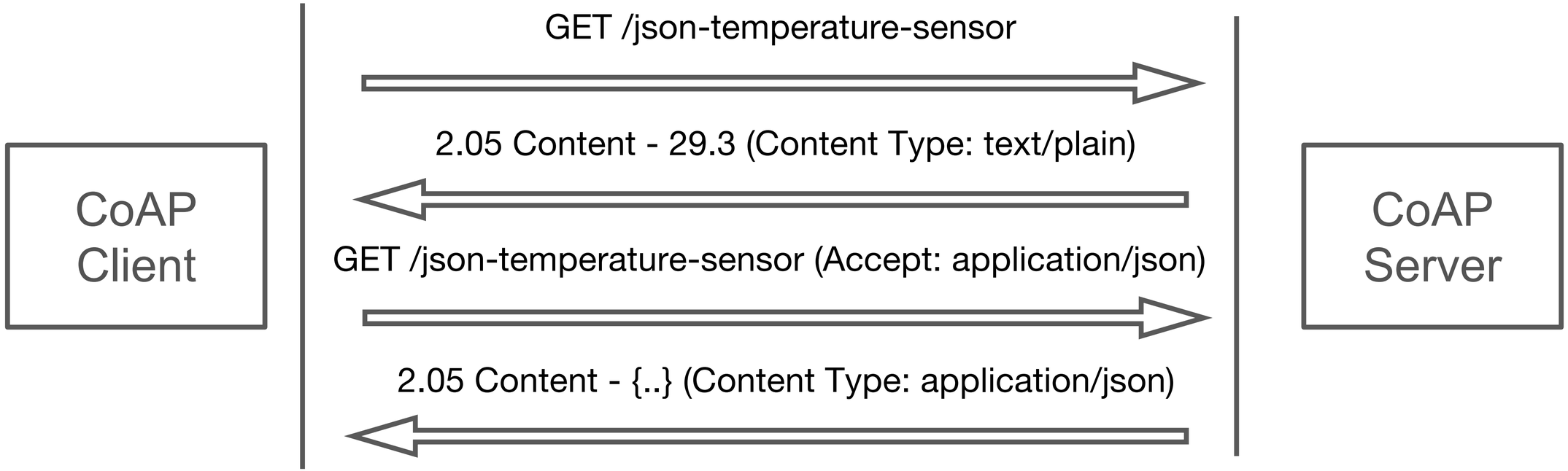
Example 5 - Observing Client

- Create a CoAP Client able to start observing the “temperature-sensor-obs” resource
- For each received notification the client print the received message
- After a target period of time (e.g. 10 sec) the client stops the observation

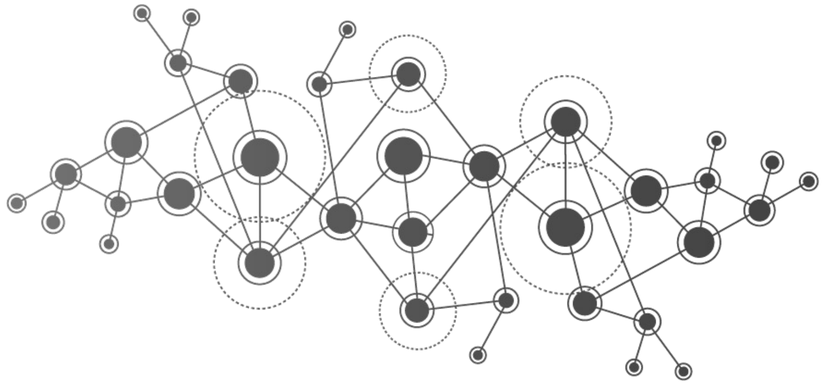


Example 6 - Json Content Format Negotiation

- Add to the created Temperature Server a new not observable Resource “json-temperature-sensor” with the following characteristics
  - Host a double temperature value
  - If receive a GET request without any Accept Option responds with a TextPlain payload containing only the double value
  - If receive a GET request with the Accept Option equals to **application/json** responds with a structured JSON payload containing the double value and the sampling timestamp
- Create a client adding the Accept Option to test the behaviour of the new resource



UNIMORE
UNIVERSITÀ DEGLI STUDI DI
MODENA E REGGIO EMILIA



Intelligent Internet of Things

Hands-On Session - CoAP

Prof. Marco Picone

A.A 2023/2024