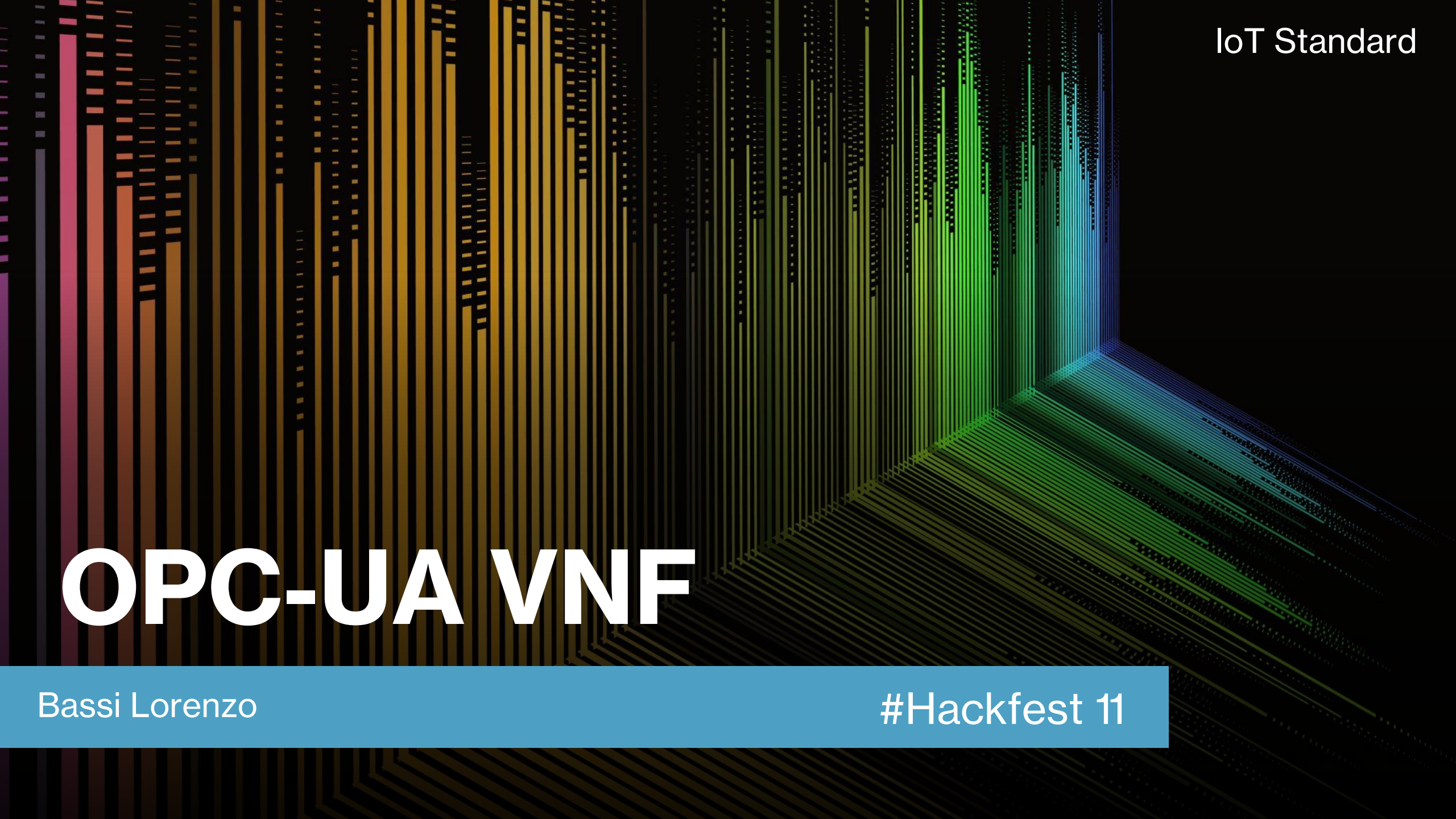

IoT Standard
OPC-UA VNF
Bassi Lorenzo
#Hackfest 11

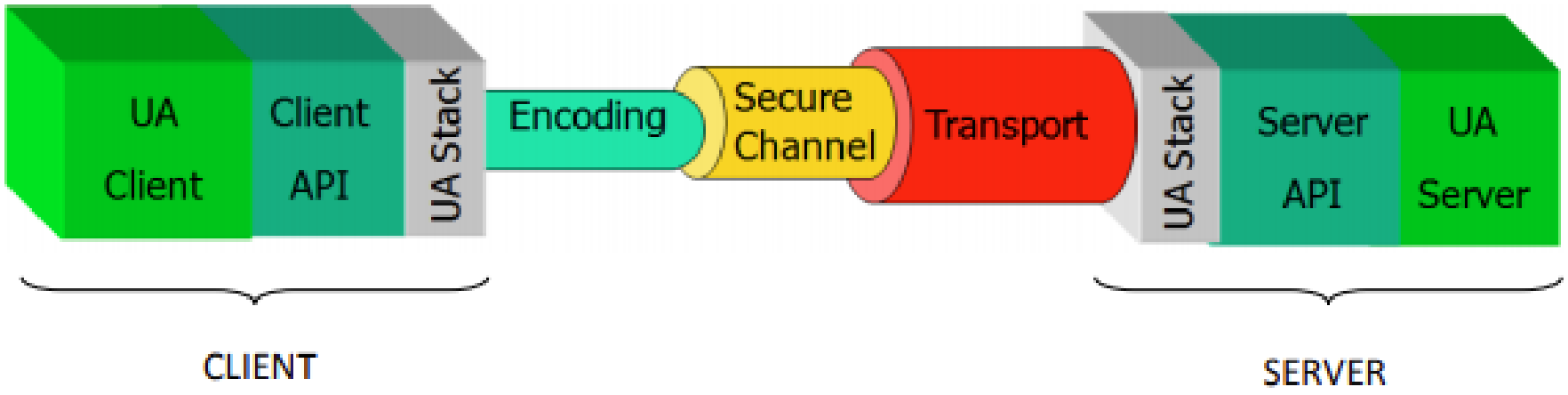
What is OPC UA ?

OPC = Open Platform Communication

UA = Unified Architecure

OPC UA Features

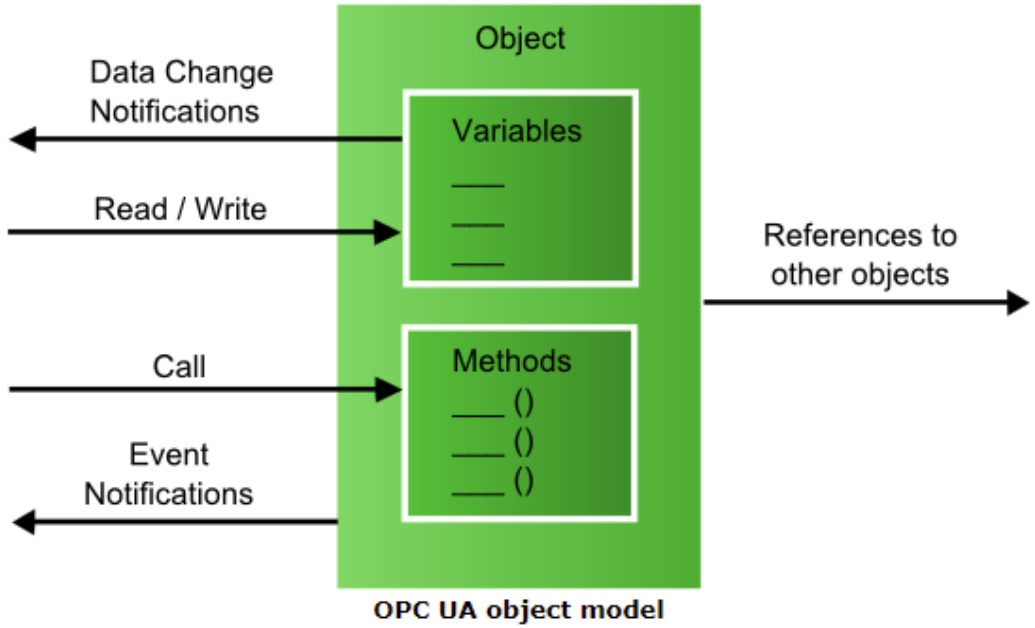
Based on the Client-Server paradigm



OPC UA Features

Based on the Client-Server paradigm

Vendor-independent Platform

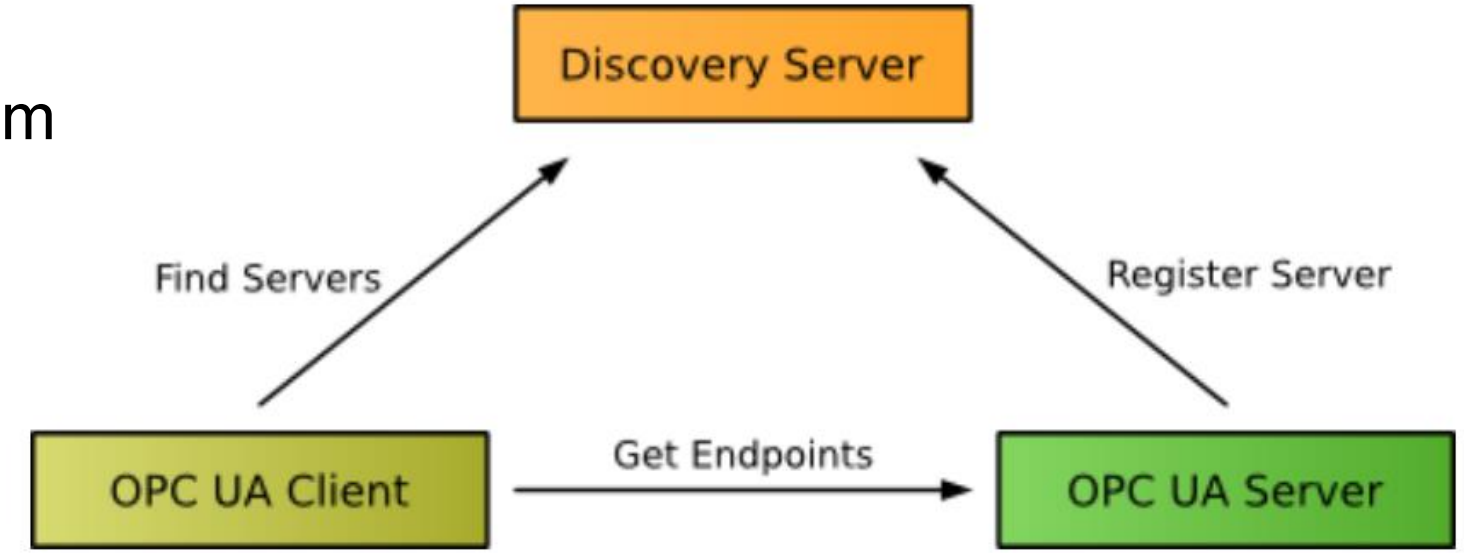


OPC UA Features

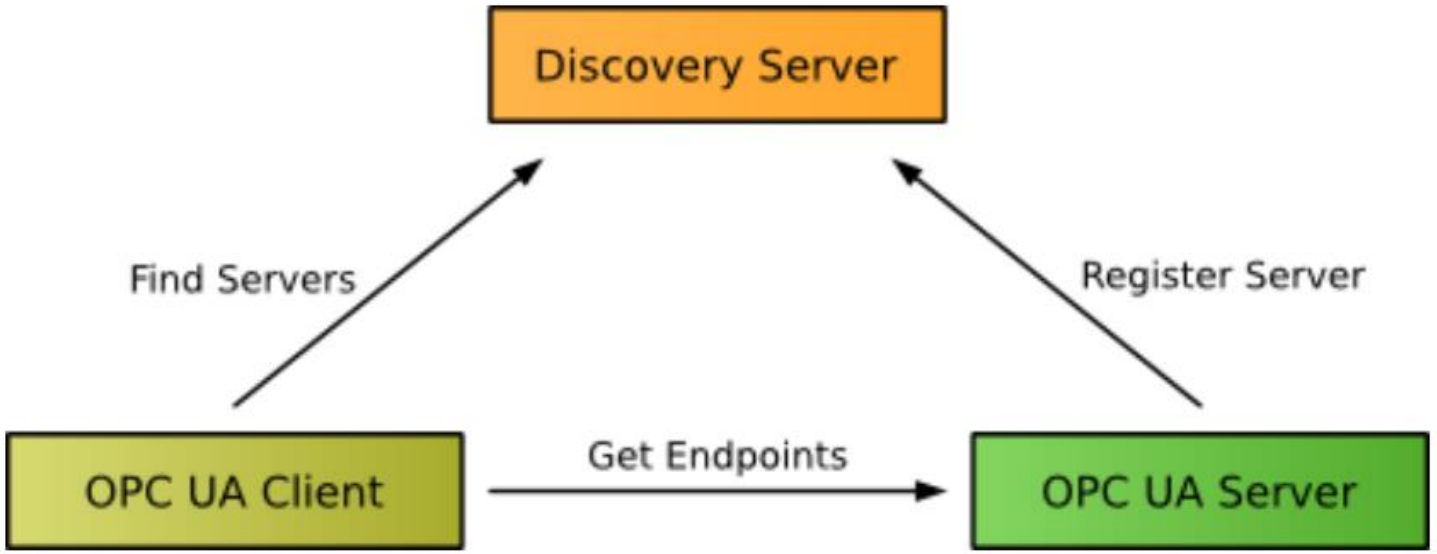
Based on the Client-Server paradigm

Vendor-independent Platform

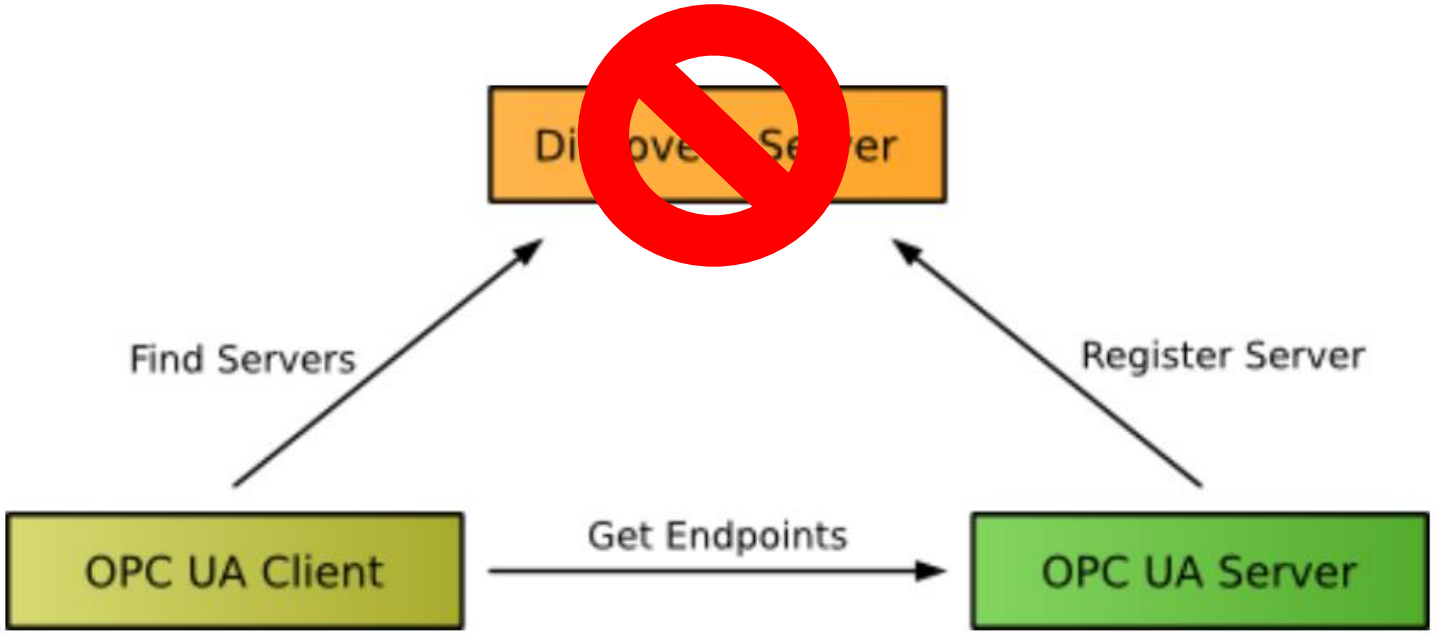
Node Discovery server



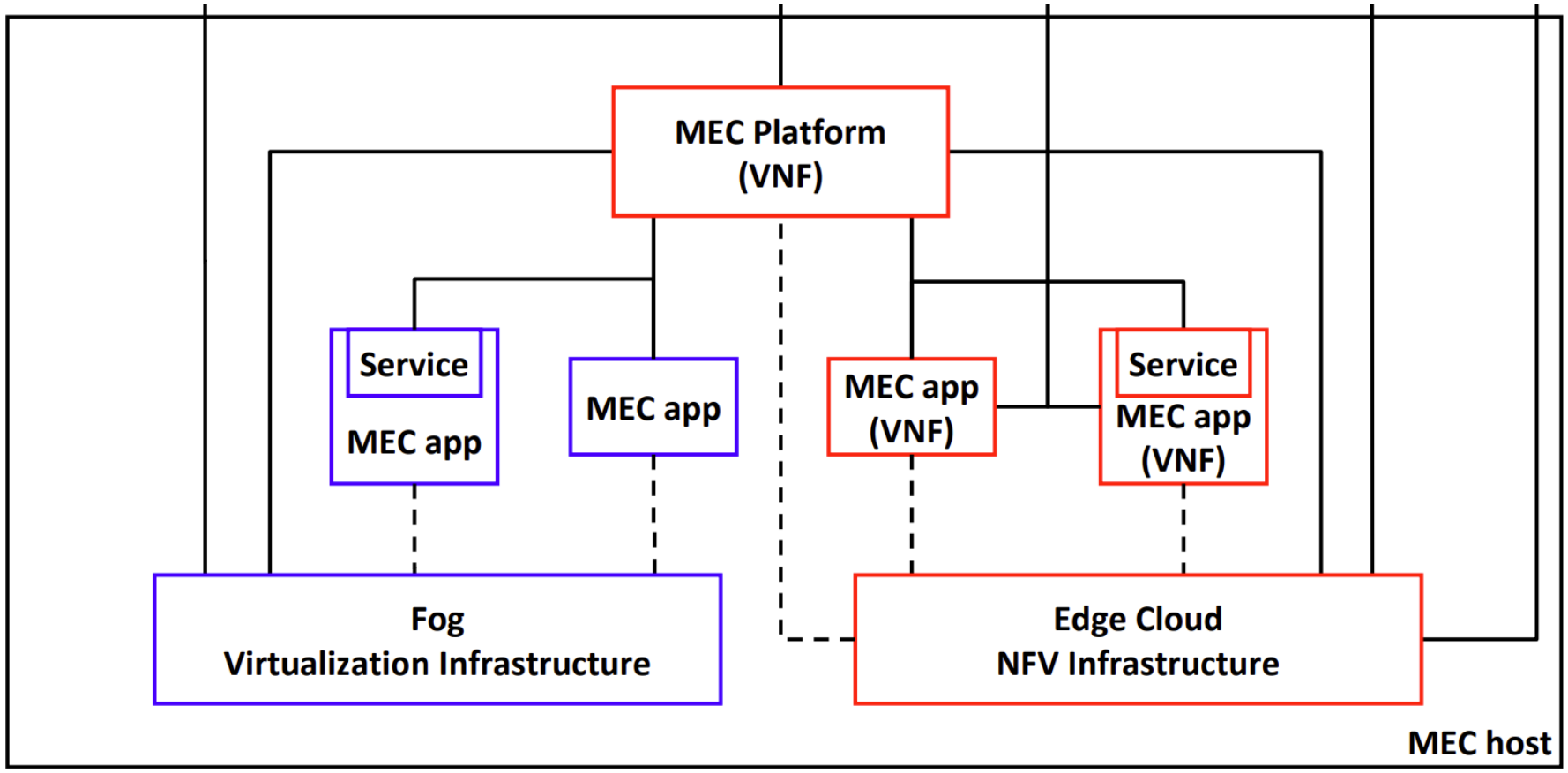
My objective...



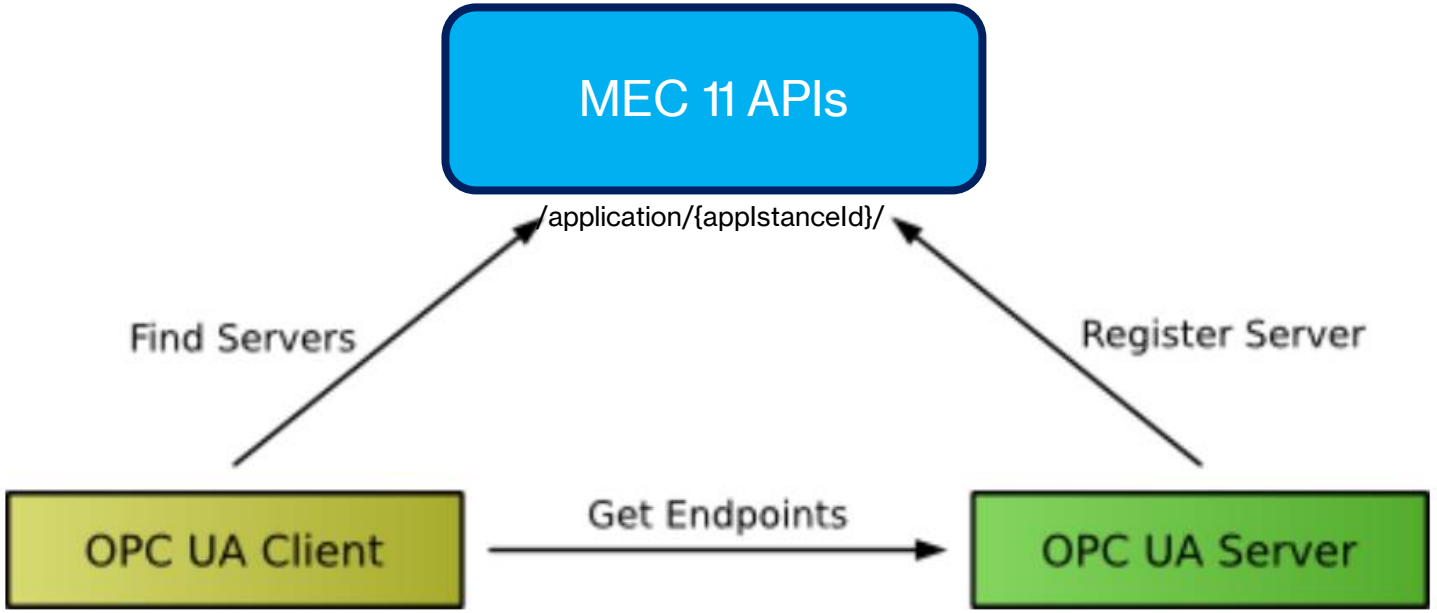
My objective...



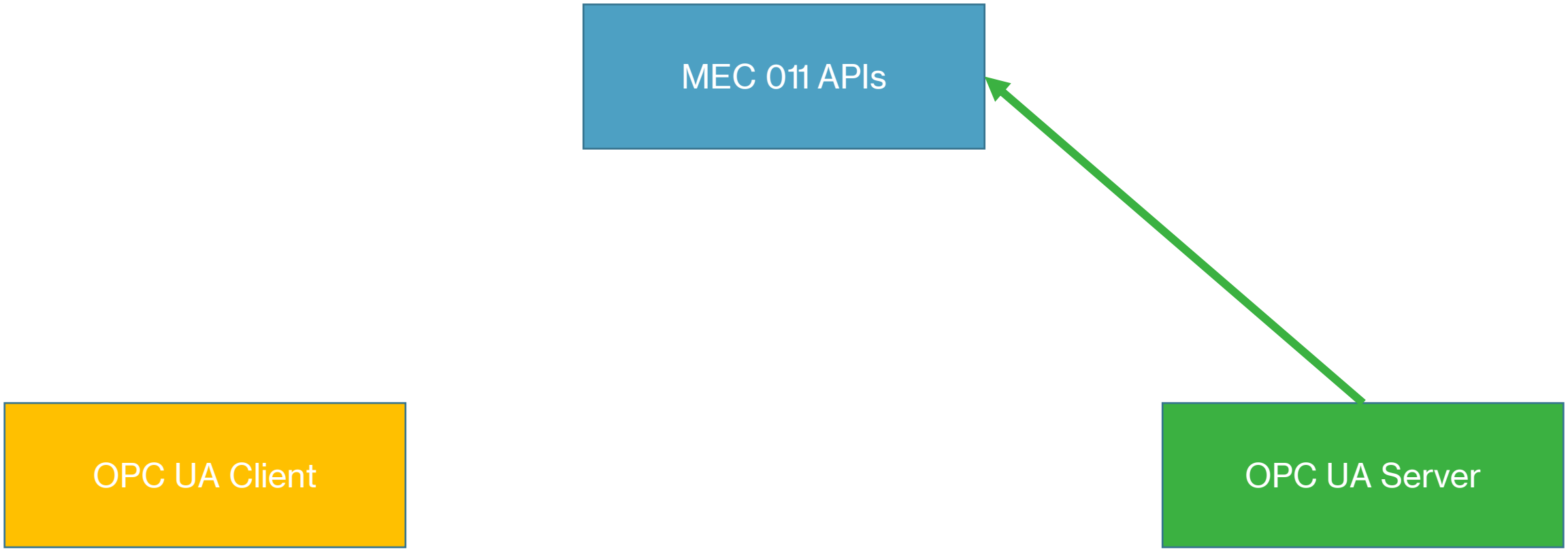
MEC 011 APIs


MEC Platform
(VNF)
Service
MEC app
MEC app
MEC app
(VNF)
Service
MEC app
(VNF)
Fog
Virtualization Infrastructure
Edge Cloud
NFV Infrastructure
MEC host

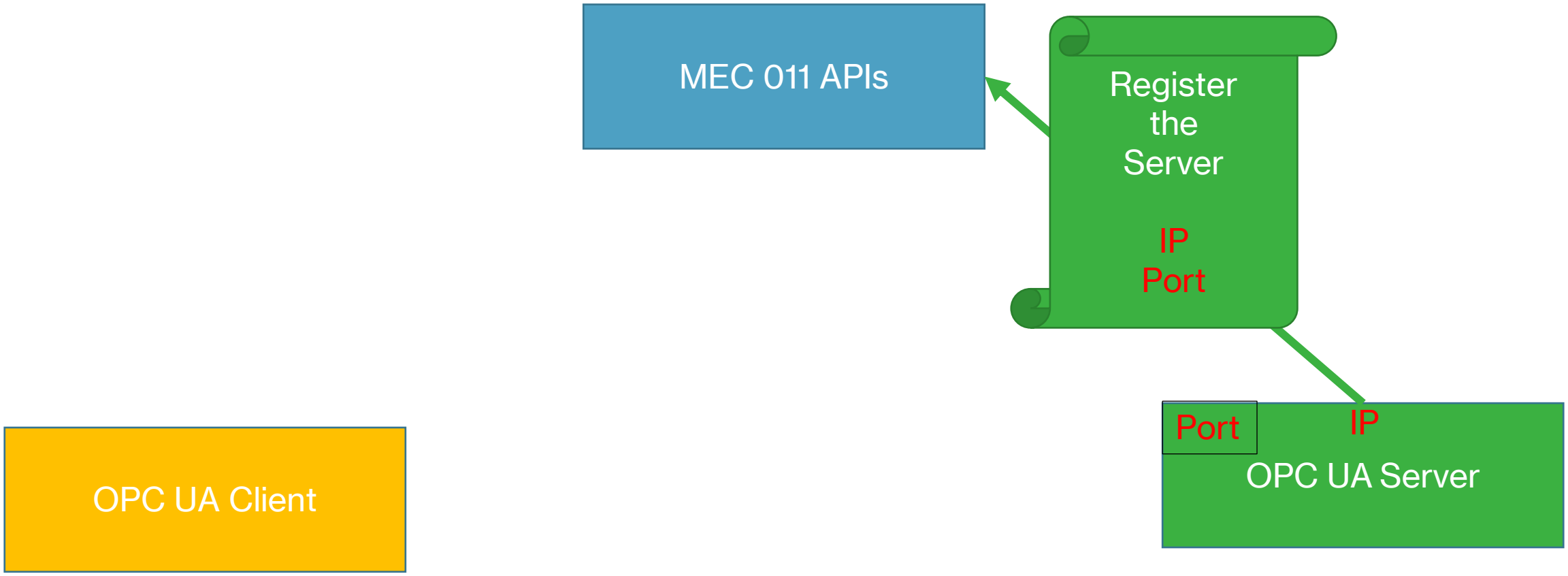
My objective...



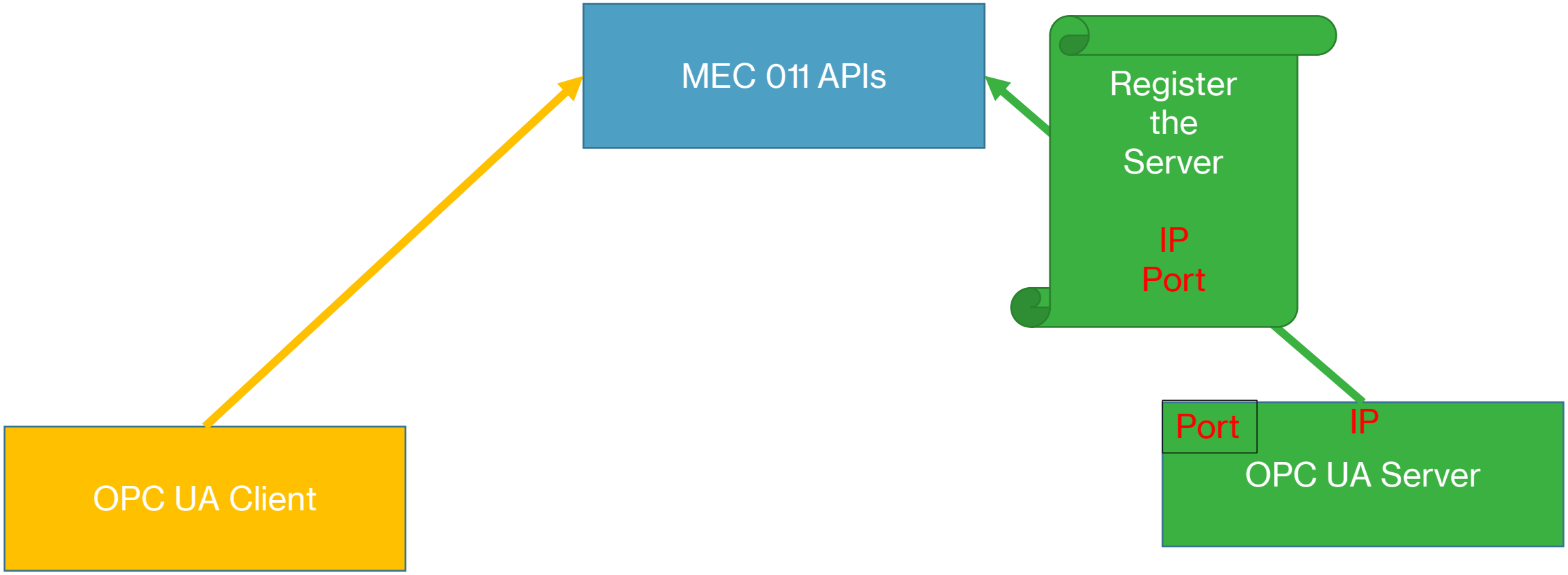
MEC 011 APIs



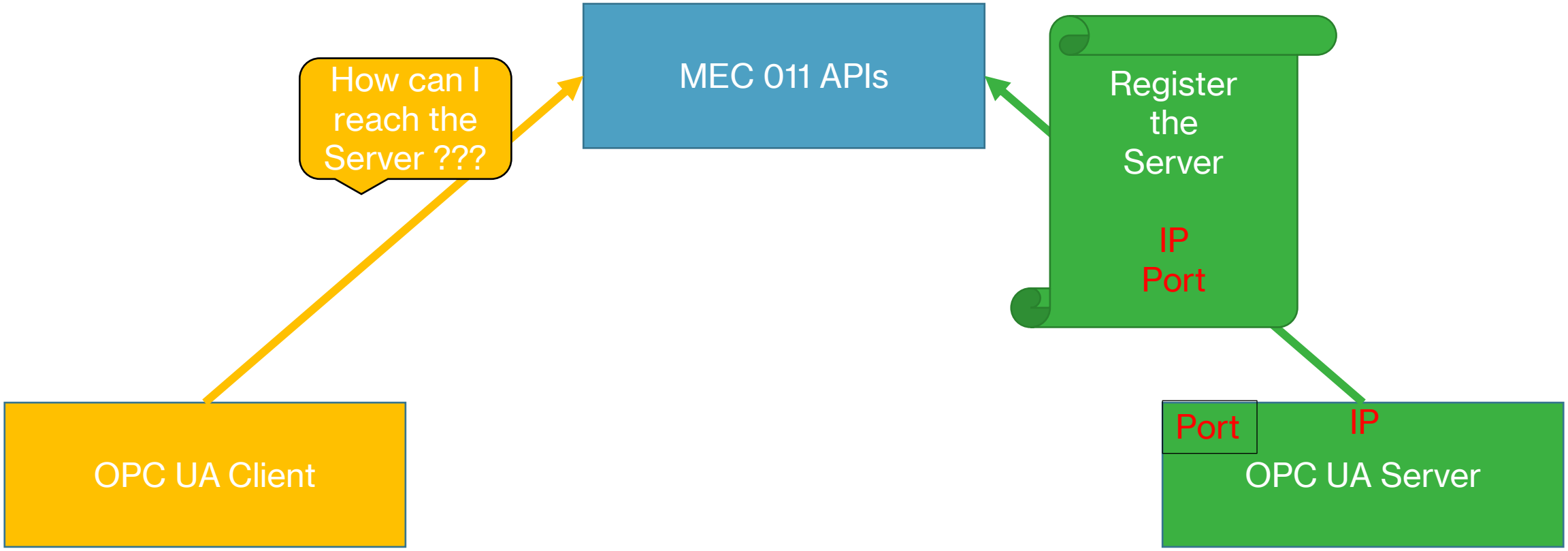
MEC 011 APIs


MEC 011 APIs
Register the Server
IP
Port
Port
IP
OPC UA Server
OPC UA Client

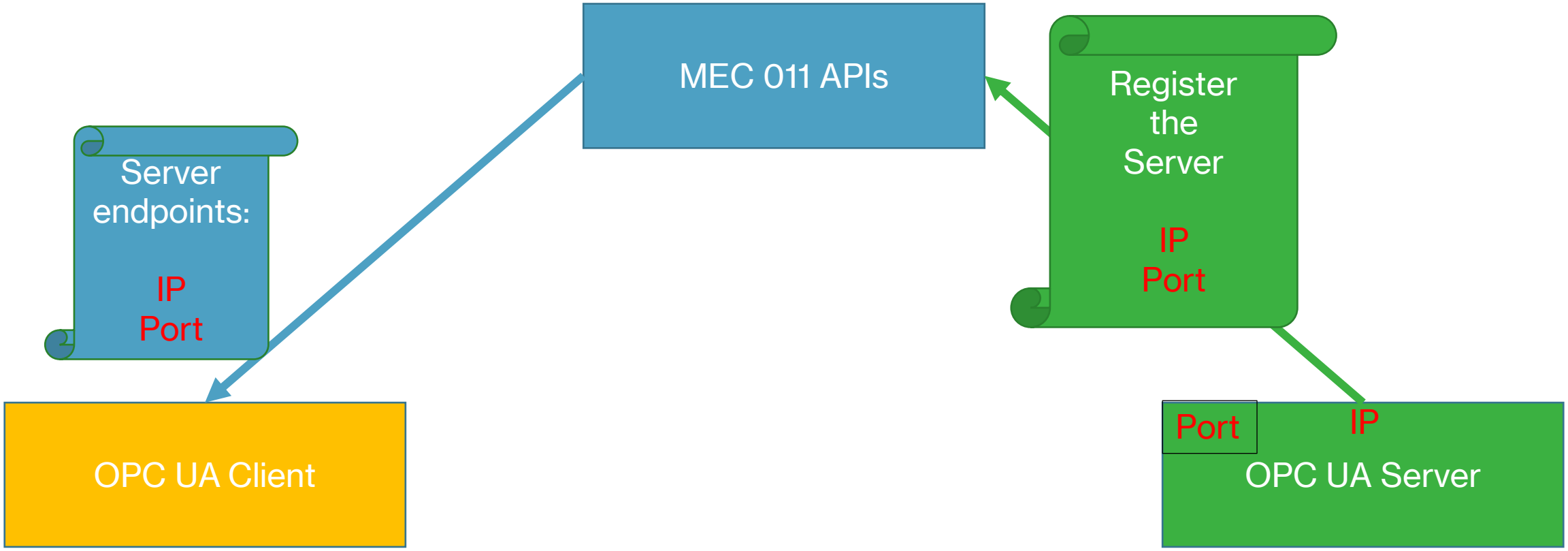
MEC 011 APIs


MEC 011 APIs
Register the Server
IP
Port
OPC UA Client
Port
IP
OPC UA Server

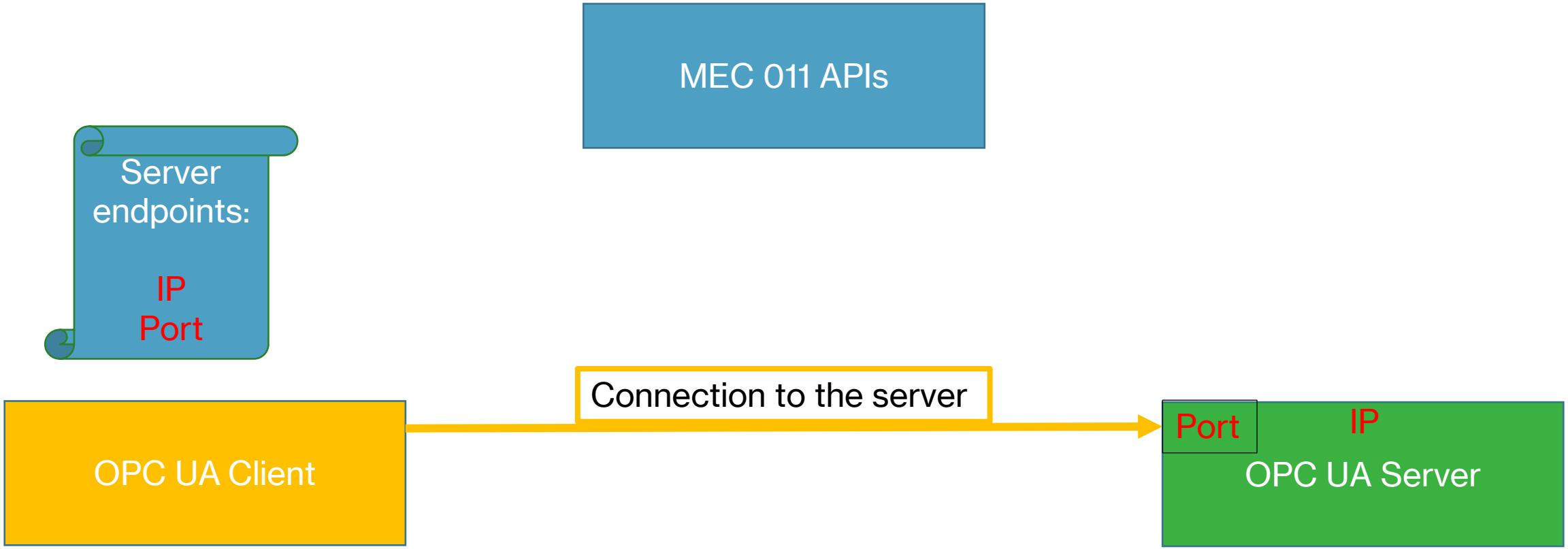
MEC 011 APIs


How can I reach the Server ???
MEC 011 APIs
Register the Server
IP
Port
OPC UA Client
Port
IP
OPC UA Server

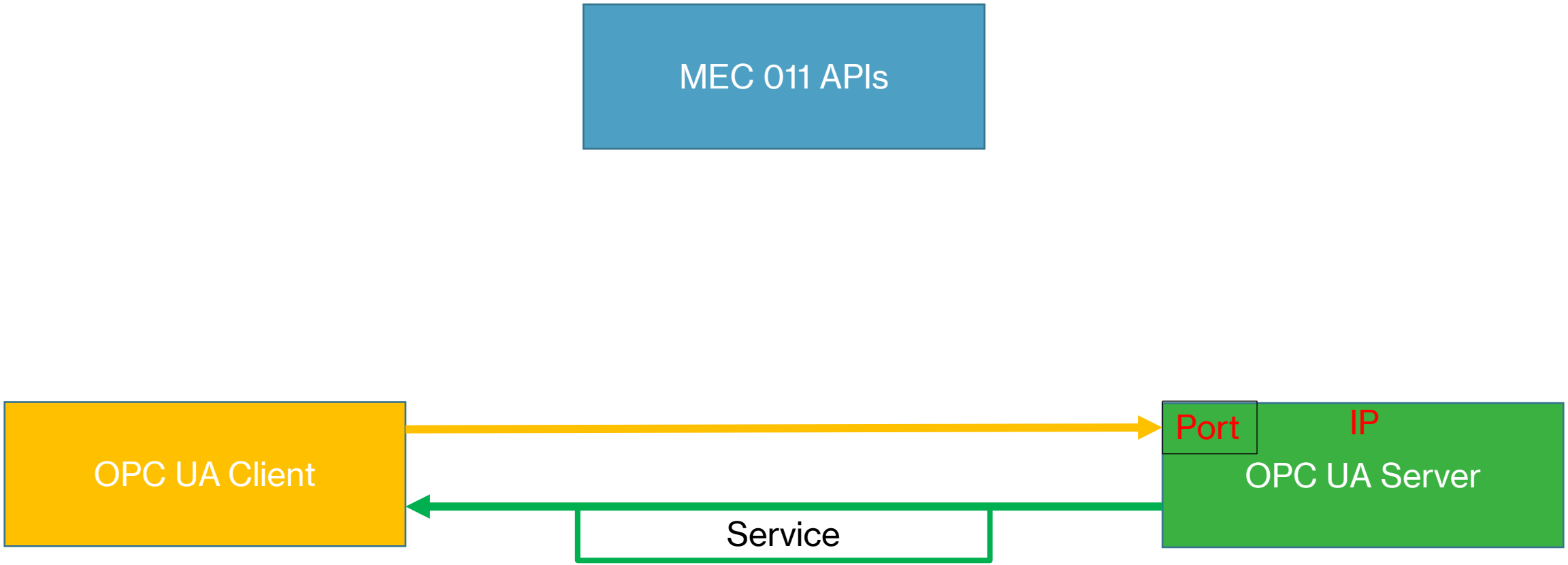
MEC 011 APIs


MEC 011 APIs
Server endpoints:
IP
Port
Register the Server
IP
Port
OPC UA Client
Port
IP
OPC UA Server

MEC 011 APIs


MEC 011 APIs
Server endpoints:
IP
Port
Connection to the server
OPC UA Client
Port
IP
OPC UA Server

MEC 011 APIs



MEC 011 APIs

Server


```
service_data = {
    "serInstanceId": "OPC_UA_SERVER",
    "serName": "OPC_UA_SERVER-service",
    "serCategory": {
    "href": "/example/catalogue1",
    "id": "id12345",
    "name": "RNI",
    "version": "version1"
    },
    "version": "ServiceVersion1",
    "state": "ACTIVE",
    "transportInfo": {
    "id": "OPC_UA_SERVER---1",
    "name": "OPC_UA_SERVER",
    "description": "OPC_UA_SERVER",
    "type": "MB_TOPIC_BASED",
    "protocol": "OPCUA",
    "version": "2.0",
    "endpoint": {
        "addresses":[{
            "host" : "opc_ua_server",
            "port" : "4840"
        }]
    },
    . . .
```

MEC 011 APIs

Client

OPC UA Client

```
def start_sensing():
    endpoint = {}

    try:
        srvs = get_all_services()
        for s in srvs:
            if s['transportInfo']['type'] == 'MB_TOPIC_BASED' and s['transportInfo']['protocol'] == 'OPCUA'
:                   endpoint=s['transportInfo']['endpoint']['addresses']
                                opc_ua_server_service_id = s['serInstanceId']
          . . .
    address = endpoint[0]['host']
    port = endpoint[0]['port']
          . . .



t = pubThread("OPCUA-Client", rate, address, port , qos)
    t.start()
```

Day-2 actions

get-node-hierachy

```
lcm-operations-configuration:
      operate-vnf-op-config:
        day1-2:
            - name: get-node-hierachy
              execution-environment-ref: simple-ee
              parameter:
              - data-type: STRING
                name: hostname
              - data-type: STRING
                name: port
```

Day-2 actions

get-namespace-variable

```
lcm-operations-configuration:
      operate-vnf-op-config:
        day1-2
            - name: get-namespace-variable
              execution-environment-ref: simple-ee
              parameter:
              - data-type: STRING
                name: hostname
              - data-type: STRING
                name: port
              - data-type: STRING
                name: index
```

Day-2 actions

change-server-status

```
lcm-operations-configuration:
      operate-vnf-op-config:
        day1-2:
            - name: change-server-status
              execution-environment-ref: simple-ee
              parameter:
              - data-type: STRING
                name: hostname
              - data-type: STRING
                name: port
              - data-type: STRING
                name: index
              - data-type: STRING
                name: status
```

Day-1 and Day-2 operations

- Day-1 → Register the OPC UA Server in the MEC plaftorm
- Day-1 → Client get endpoint for the connection with the server

- Day-2 → Request node namespace
- Day-2 → Request to read a value form the namespace
- Day-2 → Change a valuie inside the server namespace