

# Zigbee Interworking

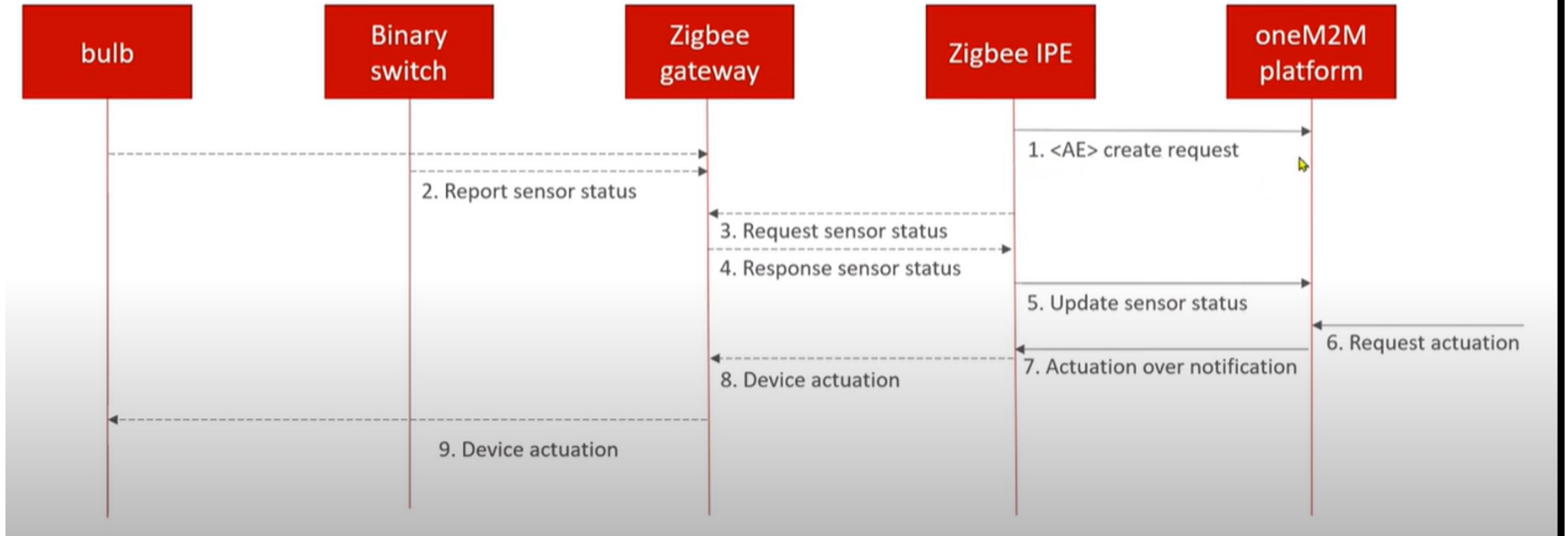
CS3

김상원  
안희찬  
이동훈

# 1. 예상 설계 시나리오

# Target

- Simple scenario



# Target

1. Zigbee IPE가 oneM2M의 request를 곧바로 zigbee protocol로 변환해야 한다.
2. 예를 들어, 기존 코드에서 oneM2M platform의 GET 요청에 대한 response를 얻기 위해, deconz로 보낸 GET의 response를 oneM2M platform에 보내는 코드를 찾아서 이를 수정해야 한다.
3. 수정해야 할 사항: oneM2M platform의 GET 요청을 곧바로 deconz 코드를 참고하여 zigbee로 요청해야 한다.

# Mobius GET method

```
---
110  this.get_device_temperature = function() {
111      setInterval(function(){
112          for (const [key, value] of Object.entries(resourceConfig)){
113              for(let i = 0; i < value.length; i++){
114                  for(let j = 0; j < value[i].deconzResource.state.length; j++) {
115                      if(value[i].deconzResource.state[j] === "temperature") {
116                          let temperature_value = deconz.sensordata(GATEWAY_ADDRESS, value[i].deconzId, value[i].deconzResource.state[j]);
117                          if (Last_temperature_level.get(value[i].oneM2MResource.deviceResourceName) != temperature_value) {
118                              Last_temperature_level.set(value[i].oneM2MResource.deviceResourceName, temperature_value);
119                              let device_temperature_value_path = "/" + CSE_NAME + "/" + IPE_RESOURCE_NAME + "/" + value[i].oneM2MResource.deviceResourceName + "/temperature";
120                              let device_temperature_value_resp = KETIMobius.put_fcmt(device_temperature_value_path, temperature_value);
121                              console.log(device_temperature_value_resp);
122                          }
123                      }
124                  }
125              }
126          }
127      }, ipeConfig.zigbeeGateway.sensingInterval);
128  }
---
```

- temperature\_value는 deconz.sensordata를 통해 deconz로 GET 요청을 통해 얻은 response이다.
- 따라서 deconz에서 이에 해당하는 부분을 찾아 수정하면 oneM2M의 GET 요청을 곧바로 Zigbee 요청으로 변경할 수 있다.

# deconz GET method

```
    */  
    ✓ int RestDevices::handleApi(const ApiRequest &req, ApiResponse &rsp)  
    {  
        // GET /api/<apikey>/devices  
        if (req.hdr.pathComponentsCount() == 3 && req.hdr.httpMethod() == HttpGet)  
        {  
            return getAllDevices(req, rsp);  
        }  
        // GET /api/<apikey>/devices/<uniqueid>  
        else if (req.hdr.pathComponentsCount() == 4 && req.hdr.httpMethod() == HttpGet)  
        {  
            return getDevice(req, rsp);  
        }  
        // PUT /api/<apikey>/devices/<uniqueid>/ddf/reload  
        else if (req.path.size() == 6 && req.hdr.method() == QLatin1String("PUT") && req.path[4] == QLatin1String("c  
        {  
            return putDeviceReloadDDF(req, rsp);  
        }  
        // GET /api/<apikey>/devices/<uniqueid>/ddf  
        else if (req.hdr.pathComponentsCount() == 5 && req.hdr.httpMethod() == HttpGet && req.hdr.pathAt(4) == QLat
```

# ApiRequest, ApiResponse

```
1012     */
1013     class ApiRequest
1014     {
1015     public:
1016         ApiRequest(const QHttpRequestHeader &h, const QStringList &p, QTcpSocket *s, const QString &c);
1017         QString apikey() const;
1018         ApiVersion apiVersion() const { return version; }
1019
1020         const QHttpRequestHeader &hdr;
1021         const QStringList &path;
1022         QTcpSocket *sock;
1023         QString content;
1024         ApiVersion version;
1025         ApiAuthorisation auth;
1026         ApiMode mode;
1027     };
1028
1029     /*! \class ApiResponse
1030
1031         Helper to simplify HTTP REST request handling.
1032     */
1033     class ApiResponse
1034     {
```

## 2. 한계



# 1. 소프트웨어 자체 개발의 제한사항

- Zigbee protocol을 사용하는 제품이라도 제조사가 다르면 통신하는데 제한요소가 많습니다.
- Zigbee 수신기를 개발하는 국내 회사 Tessol에 전화 문의 했을 때, 자신들 제품이 라즈베리파이 2와 샤오미 Zigbee 제품과 통신이 될 거라고 확신할 수 없기에 자신들 제품이 conbee 2를 대체할 수 있다고 말하기 어렵다고 하였습니다.

➤ 따라서 zigbee ipse 자체 개발을 포기하였습니다.

## 2. Conbee 2의 오픈소스 프로그램

- deconz를 활용한 개발의 제한사항:

Qt 프레임워크로 작성된 코드이기 때문에 해당 코드를 완전히 이해하기 어렵고, 코드의 복잡도가 높아서 지난주 교수님 미팅 이후로 오랜 시간 코드 리뷰를 하였지만 이해할 수 없었습니다.

➤ 따라서 open source를 수정하는 방식의 개발을 포기하였습니다.

# HTTP 2번 쓰려는 이유

## Architecture for IPE



- Interworking Proxy Entity(IPE) with others
  - IPE is a special AE that allows seamlessly interact between oneM2M system and other systems

