

Device

Notes and Ideas

- EUI should form the key for Devices

API and Data Shape

Create Device

```
POST /api/devices
```

Body

```
{
  "deviceEUI": "string",
  "name": "string",
  "description": "deviceDescription",
  "applicationId": "string",
  "deviceProfileId": "string",
  "skipFrameCounterCheck": "boolean",
  "referenceAltitude": "number"
}
```

Returns 201

```
{}
```

Get Device

1. Do we want to return the network information with the device
2. If so does this impact Option 1 or 2 design?

```
GET /api/devices/:eui
```

Returns 200

```
{
  "deviceEUI": "string",
  "name": "string",
  "description": "deviceDescription",
  "applicationId": "string",
  "deviceProfileId": "string",
  "skipFrameCounterCheck": "boolean",
  "referenceAltitude": "number",
  "deviceStatusBattery": "number",
  "deviceStatusMargin": "number",
  "lastSeenAt": "date",
  "location": {
    "accuracy": "number",
    "altitude": "number",
    "latitude": "number",
    "longitude": "number",
    "source": "string"
  },
  "networkDeployment": [
    {
      networkId: "string",
      remoteDeviceId: "string"
    }
  ]
}
```

Get Many Devices

```
GET /api/devices?query=q
```

Returns 200

```
{
  "returned": "number",
  "available": "number",
  "results": [
    {
```

```
    "deviceEUI": "string",
    "name": "string",
    "description": "deviceDescription",
    "applicationId": "string",
    "deviceProfileId": "string",
    "skipFrameCounterCheck": "boolean",
    "referenceAltitude": "number",
    "deviceStatusBattery": "number",
    "deviceStatusMargin": "number",
    "lastSeenAt": "date"
  }]
}
```

Get an Application's Devices

```
GET /api/applications/:id/devices?query=q
```

Returns 200

```
{
  "returned": "number",
  "available": "number",
  "results": [
    {
      "deviceEUI": "string",
      "name": "string",
      "description": "appDescription",
      "applicationId": "string",
      "deviceProfileId": "string",
      "skipFrameCounterCheck": "boolean",
      "referenceAltitude": "number",
      "deviceStatusBattery": "number",
      "deviceStatusMargin": "number",
      "lastSeenAt": "date"
    }
  ]
}
```

Update Device

```
PUT /api/devices/:eui
```

Body

```
{
  "deviceEUI": "string",
  "name": "string",
  "description": "deviceDescription",
  "applicationId": "string",
  "deviceProfileId": "string",
  "skipFrameCounterCheck": "boolean",
  "referenceAltitude": "number"
}
```

Return 200

```
{
  "deviceEUI": "string",
  "name": "string",
  "description": "deviceDescription",
  "applicationId": "string",
  "deviceProfileId": "string",
  "skipFrameCounterCheck": "boolean",
  "referenceAltitude": "number",
  "deviceStatusBattery": "number",
  "deviceStatusMargin": "number",
  "lastSeenAt": "date"
}
```

Delete Device

```
DELETE /api/devices/:id
```

Body

```
{}
```

Return 204

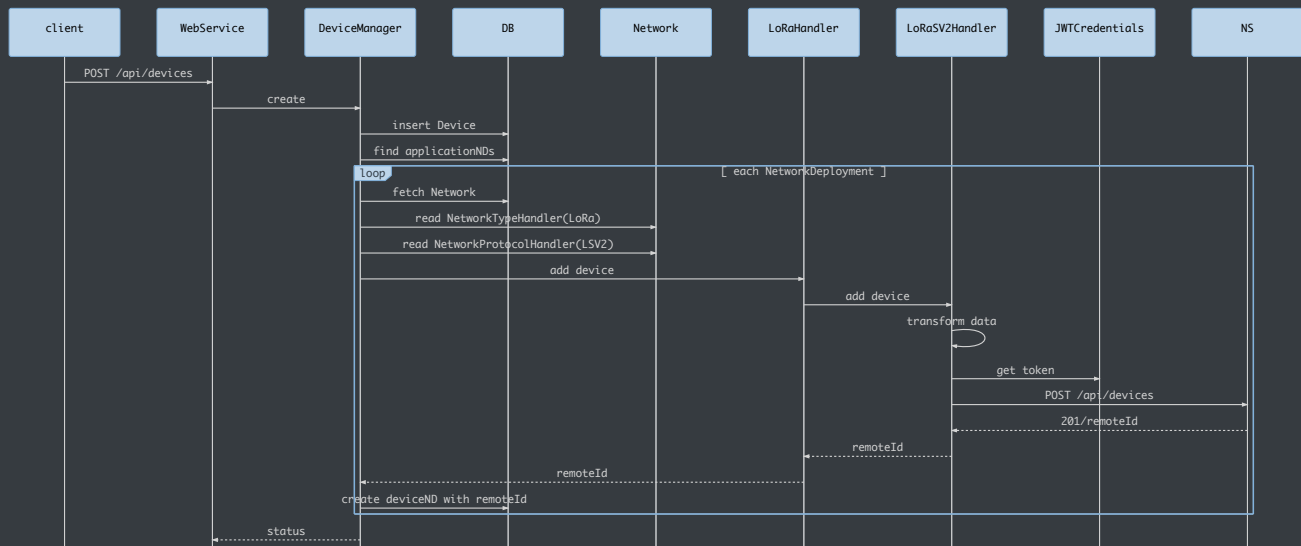
```
{}
```

Proposed Flow

New Device

Note: Used LoRaServerV2 as an example, could be any protocol

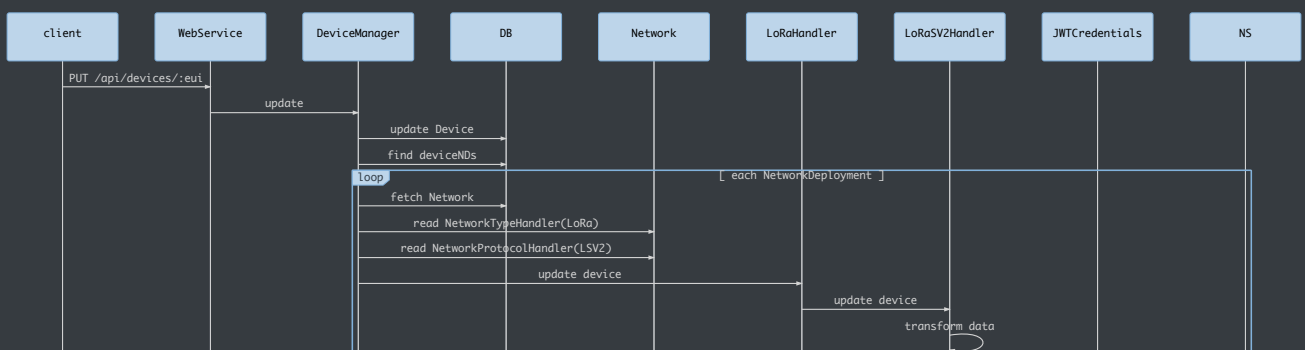
Question: Do we want to use Application to manage the networkDeployments or use them directly?

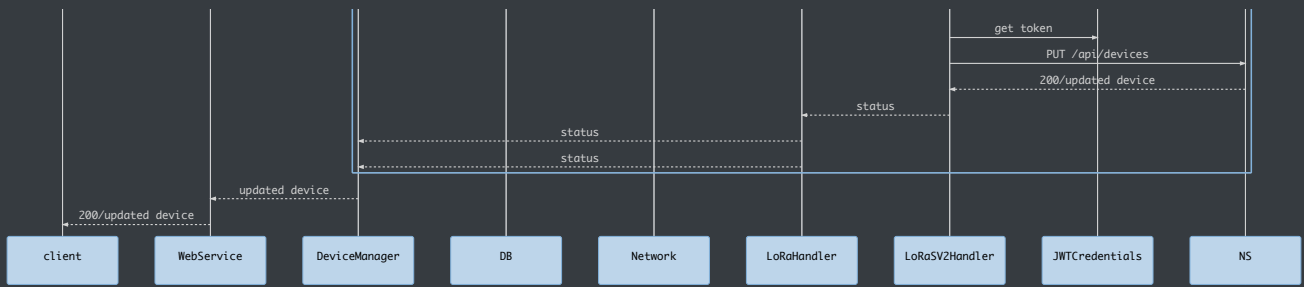




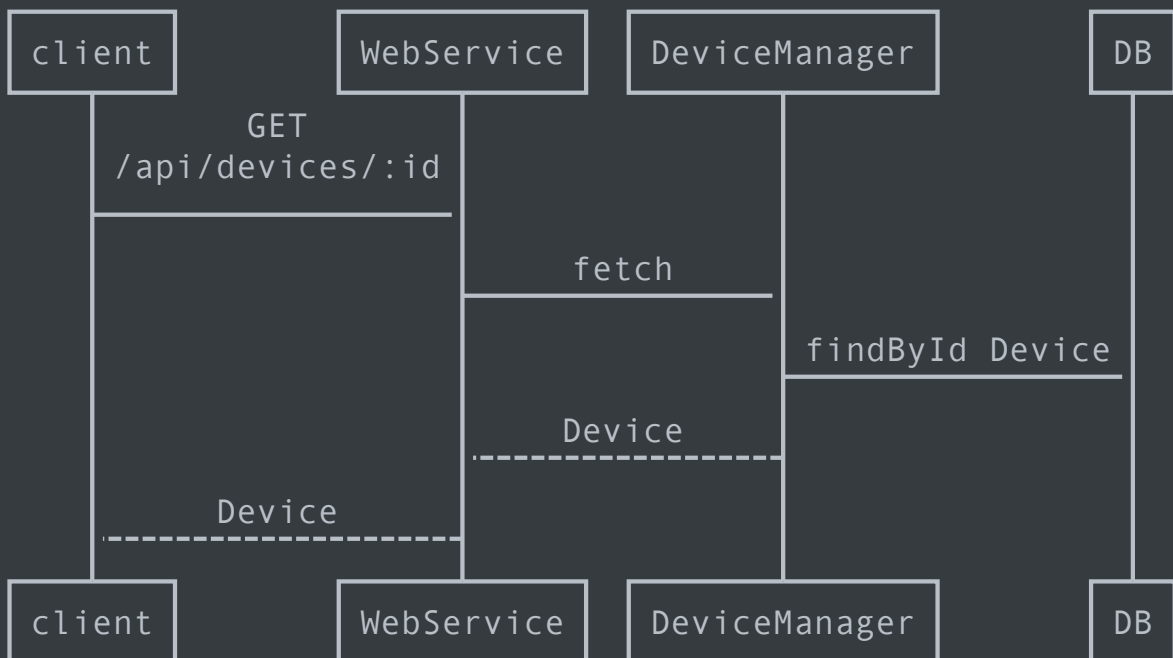
Update Device

Note: Used LoRaServerV2 as an example, could be any protocol





Get Device



Delete Device

