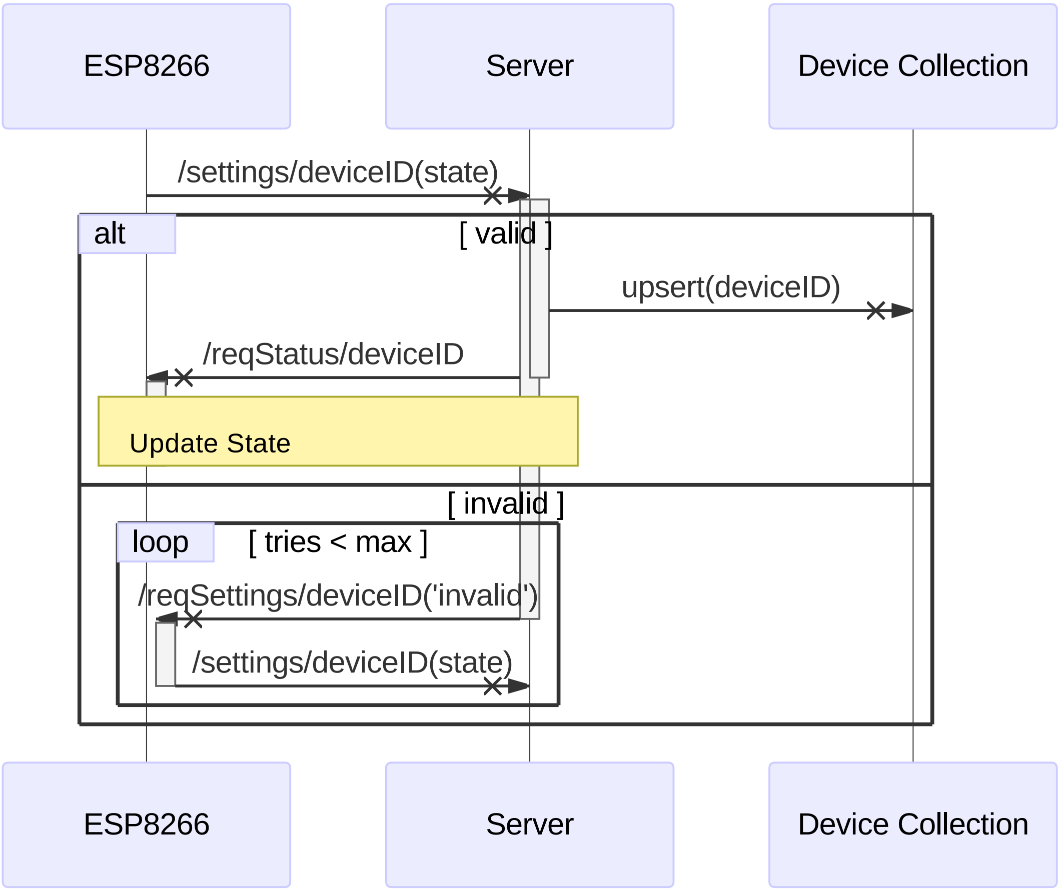
Device to Server Protocol

Handshake

sd Handshake



1. Upon connection, a device publishes a description of itself, including checkin behaviors, publish/subscribe topics, alterable settings etc.
2. The server, which subscribes to all topics, validates the devices announcement, and sets up the environment for it (adds/upserts a record of the device in table/collection of devices)
3. Server requests device state (publish to /reqStatus/deviceID)

Device Announcement/Declaration

A devices first communication with the server includes (in JSON):

1. a schema for future status updates

2. type of data reported (scalar/boolean, time-series/control-state)
3. checkin behavior (never, on-change, periodically etc)
4. deviceID
5. list of all topics published/subscribed to

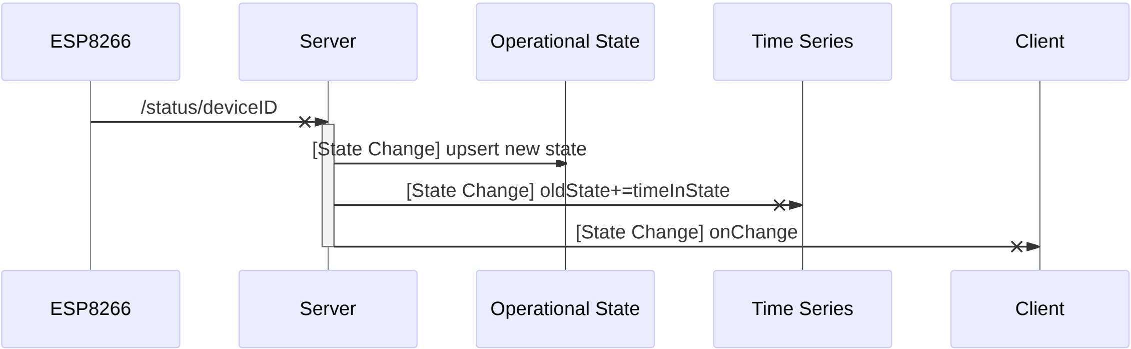
Example

```
{
  "deviceID": 12736004,
  "name": "Livingroom DHT",
  "primaryType": "digitalInput",
  "checkinFreq": 3600000,
  "type": "dht11",
  "topics": {
    "sub": {
      "settings": "/settings/12736004",
      "reqStatus": "/reqStatus/12736004"
    },
    "pub": {
      "status": "/status/12736004",
      "currentSettings": "/currentSettings/12736004"
    }
  },
  "schema": {
    "deviceID": 12736004,
    "timestamp": 1480192198673,
    "timeSeries": true,
    "type": "dht11",
    "purpose": "Temperature/Humidity Sensor",
    "state": {
      "status": "enum",
      "temp": "scalar",
      "humi": "scalar"
    }
  }
}
```

Status updates

- Upon receiving a status from a known device
  - Compares state to current operatonal state in database if changed store old data (including length of time in that state) to a time-series collection or database (if time-series data)

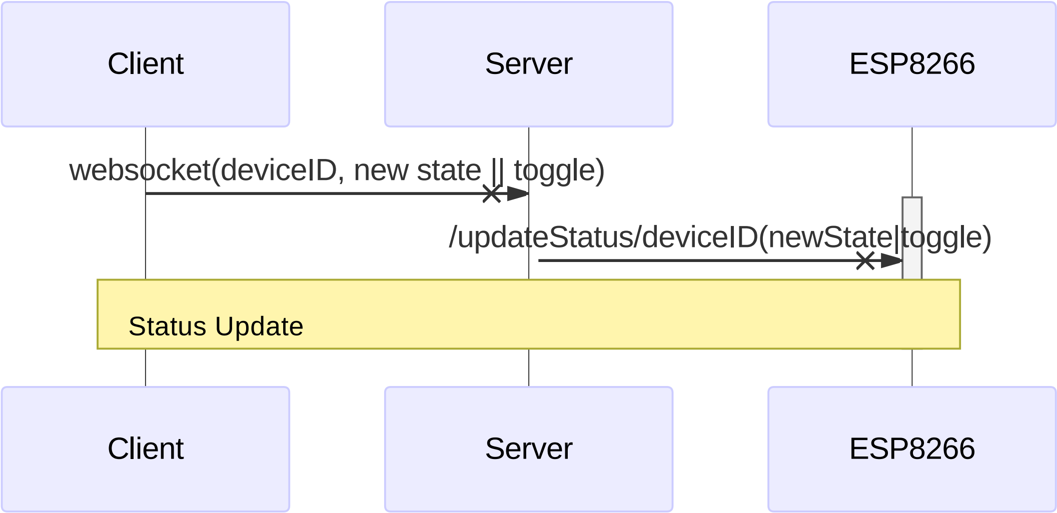
sd Status Update



Client Push

Need to define protocol for client/server initialization (client must abstract from mqtt)

sd Client Push


Client
Server
ESP8266
websocket(deviceID, new state || toggle)
/updateStatus/deviceID(newState|toggle)
Status Update
Client
Server
ESP8266