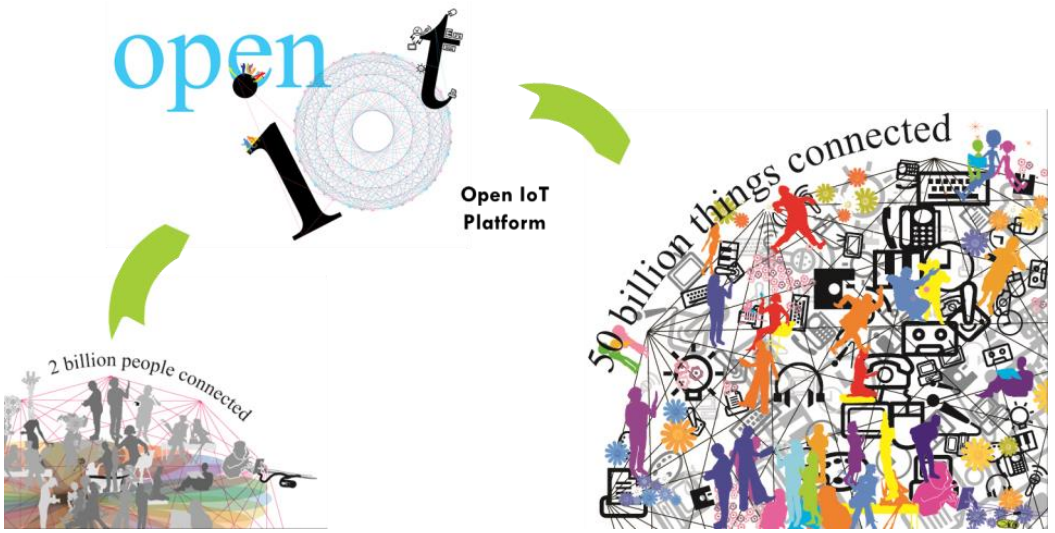
[KIOT 교육]

“사물인터넷 디바이스 개발”



KETI (Korea Electronic Technology Institute)

안 일 엽 책임

(iyahn@keti.re.kr)

1. 개 요

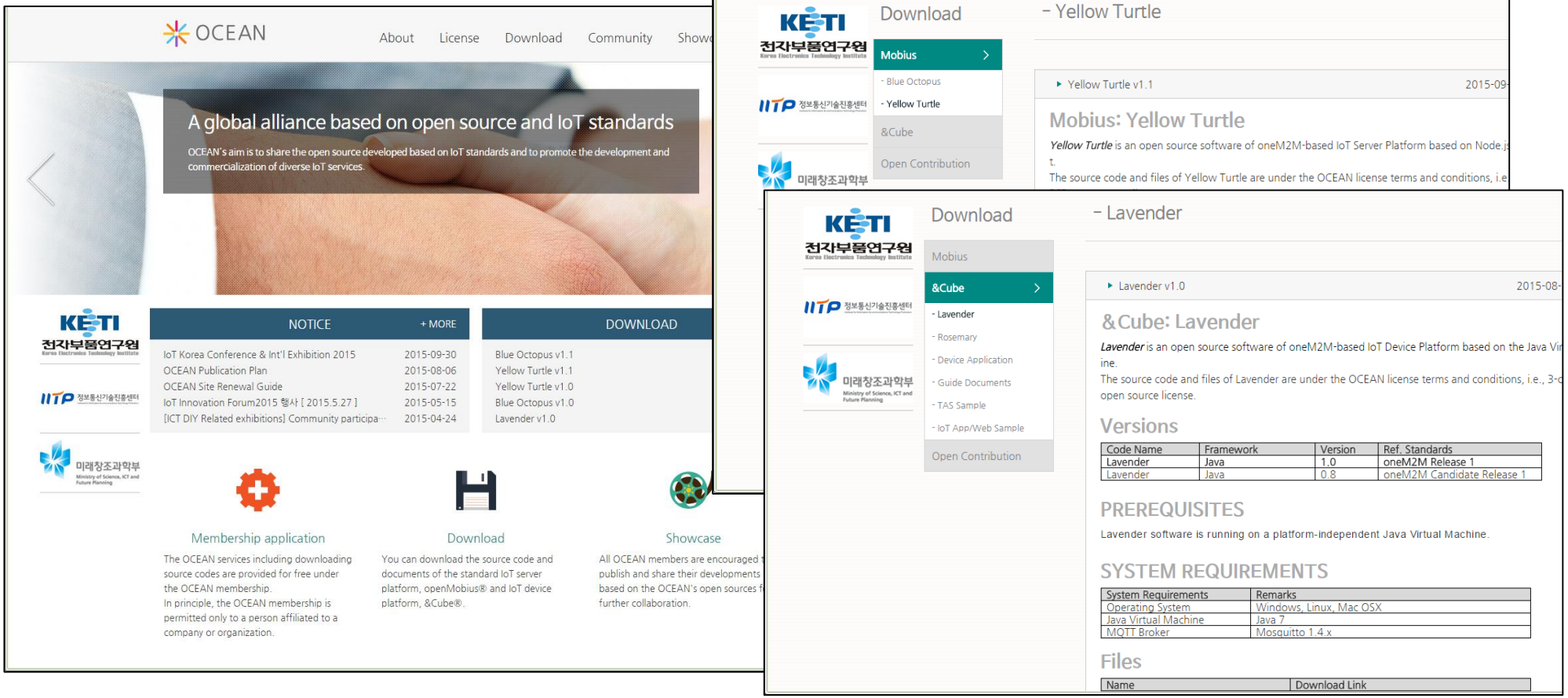
1.1 오픈소스 종류 및 oneM2M 표준 준수

1.2 &Cube: Thyme 활용 디바이스 개발 방법

1.3 &Cube 연동 구조

OCEAN Open Sources

- http://www.iotocean.org
- Have more platforms → become complicated



1.1 오픈소스 종류 및 oneM2M 표준 준수

■ 오픈소스 종류 별 oneM2M 표준 지원 여부

		AE	CSE			Framework
			ASN	MN	IN	
Mobius	Blue Octopus				√	Spring
	Yellow Turtle				√	Node.js
nCube	Rosemary			√		Java
				√		Node.js
	Lavender		√			Java
			√			Node.js
	Thyme	√				Java
		√				Node.js

■ Latest Release

	oneM2M Release 1				
	Software name	Framework	version	Publication date	Standard
Server(IN-CSE)	Mobius : Blue Octopus	Spring Framework	v1.1	2015-09-10	TS-0001 Functional Architecture v1.6.1 TS-0004 Service Layer Core Protocol v1.0.1 TS-0009 HTTP Protocol Binding v1.0.1 TS-0010 MQTT Protocol Binding v1.0.1
	Mobius : Yellow Turtle	Node.js	v1.1	2015-09-10	
Gateway(MN-CSE)	&Cube : Rosemary	Node.js	v1.0	2015-10	
Device(ASN-CSE)	&Cube : Lavender	Java	v1.0	2015-08-11	
Application(AE)	&Cube : Thyme	Node.js	v1.0	2015-10	

1.1 오픈소스 종류 및 oneM2M 표준 준수

■ Mobius: Blue Octopus

- Spring Framework 기반 IoT Server Platform (oneM2M IN-CSE)
- Java Virtual Machine 위에서 동작
- 시스템 요구사항

System requirements	Remarks
Operating System	WindowsX, Linux Redhat and CentOS
Java Virtual Machine	Java 7
Open Source Framework	Spring 3.06
Web Application Server	Spring MVC 3.0
Database	Tomcat 7
CoAP Framework	Mongo 2.6 Redis 2.8
CoAP Framework	Californium
MQTT Broker	Mosquitto 1.4.x

- 표준 리소스 지원
  - CSEBase, remoteCSE, AE, node, container, contentInstance, subscription, notification, mgmtCmd, execInstance, mgmtObj, group
- 표준 Primitive 지원
  - XML/Json 지원
  - Short-name 리소스 지원

1.1 오픈소스 종류 및 oneM2M 표준 준수

■ Mobius: Yellow Turtle

- Node.js Java Script 기반 IoT Server Platform (oneM2M IN-CSE)
- 경량/저용량 시스템으로 간편한 설치 가능
- 시스템 요구사항

System requirements	Remarks
Operating System	WindowsX, Linux Redhat and CentOS, Mac, Raspbian
Open Source Framework	Node.js
Web Application Server	Node.js
Database	MySQL
CoAP Framework	-
MQTT Broker	Mosquitto 1.4.x

- 표준 리소스 지원
  - CSEBase, remoteCSE, AE, container, contentInstance, subscription, notification, group
- 표준 Primitive 지원
  - XML/Json 지원
  - Long/Short-name 리소스 지원

1.1 오픈소스 종류 및 oneM2M 표준 준수

■ &Cube: Rosemary

- Node.js Java Script 기반 IoT Gateway Platform (oneM2M MN-CSE)
- 경량/저용량 시스템으로 간편한 설치 가능
- 시스템 요구사항

System requirements	Remarks
Operating System	WindowsX, Linux Redhat and CentOS, Mac, Raspbian
Open Source Framework	Node.js
Web Application Server	Node.js
Database	MySQL
CoAP Framework	-
MQTT Broker	Mosquitto 1.4.x

- 표준 리소스 지원
  - CSEBase, remoteCSE, AE, container, contentInstance, subscription, notification, group
- 표준 Primitive 지원
  - XML/Json 지원
  - Long/Short-name 리소스 지원

1.1 오픈소스 종류 및 oneM2M 표준 준수

■ &Cube: Lavender

- Java 기반 IoT Device Platform (oneM2M ASN-CSE)
- 경량/저용량 시스템으로 간편한 설치 가능
- Java Virtual Machine 위에서 동작
- 자체적으로 HTTP 서버를 소스 내부에 탑재하여 리소스 최소화
- 시스템 요구사항

System requirements	Remarks
Operating System	Windows, Linux, Mac OSX, Raspbian
Java Virtual Machine	Java 7 / 8
MQTT Broker	Mosquitto 1.4.x

- 표준 리소스 지원
  - CSEBase, remoteCSE, AE, node, container, contentInstance, mgmtCmd, execInstance, mgmtObj
- 표준 Primitive 지원
  - XML/Json 지원
  - Long/Short-name 리소스 지원

1.1 오픈소스 종류 및 oneM2M 표준 준수

■ &Cube: Thyme

- Node.js Java Script 기반 IoT Application (oneM2M AE)
- 경량/저용량 시스템으로 간편한 설치 가능
- 시스템 요구사항

System requirements	Remarks
Operating System	WindowsX, Linux Redhat and CentOS, Mac, Raspbian
Open Source Framework	Node.js
Web Application Server	Node.js
Database	MySQL
CoAP Framework	-
MQTT Broker	Mosquitto 1.4.x

- 표준 리소스 지원
  - CSEBase, remoteCSE, AE, container, contentInstance, subscription, notification, group
- 표준 Primitive 지원
  - XML/Json 지원
  - Long/Short-name 리소스 지원

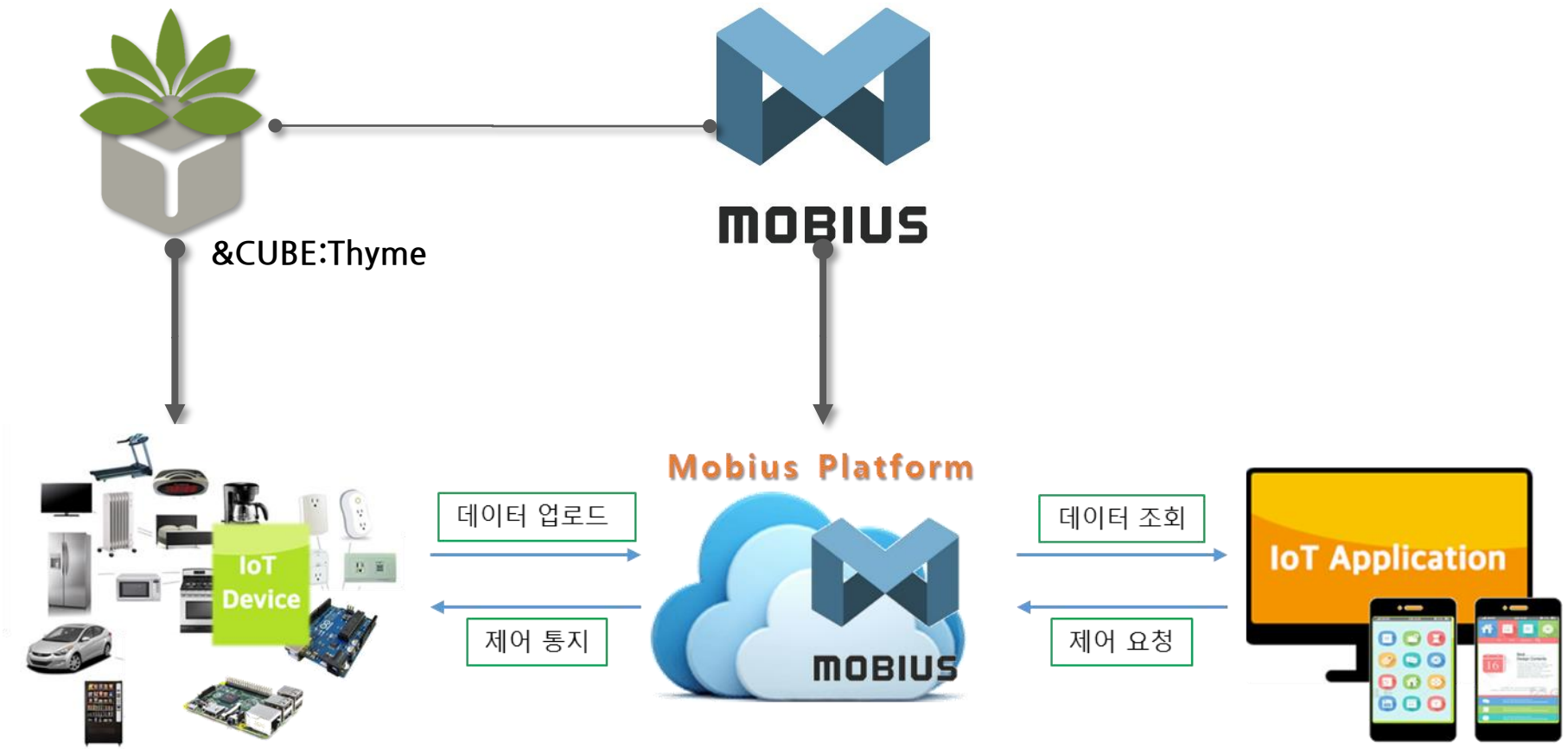
1. 개 요

1.1 오픈소스 종류 및 oneM2M 표준 준수

1.2 &Cube: Thyme 활용 디바이스 개발 방법

1.3 &Cube 연동 구조

1.2 &Cube:Thyme 활용 디바이스 개발 방법



Construction Server Platform

S/W name \ oneM2M Nodes		AE	CSE		IN	Framework
			ASN	MN		
Mobius	Blue Octopus				√	Spring
	Yellow Turtle				√	Node.js
&Cube	Rosemary			√		Java
				√		Node.js
	Lavender		√			Java
			√			Node.js
	Thyme	√				Java
		√				Node.js

Construction Device Platform

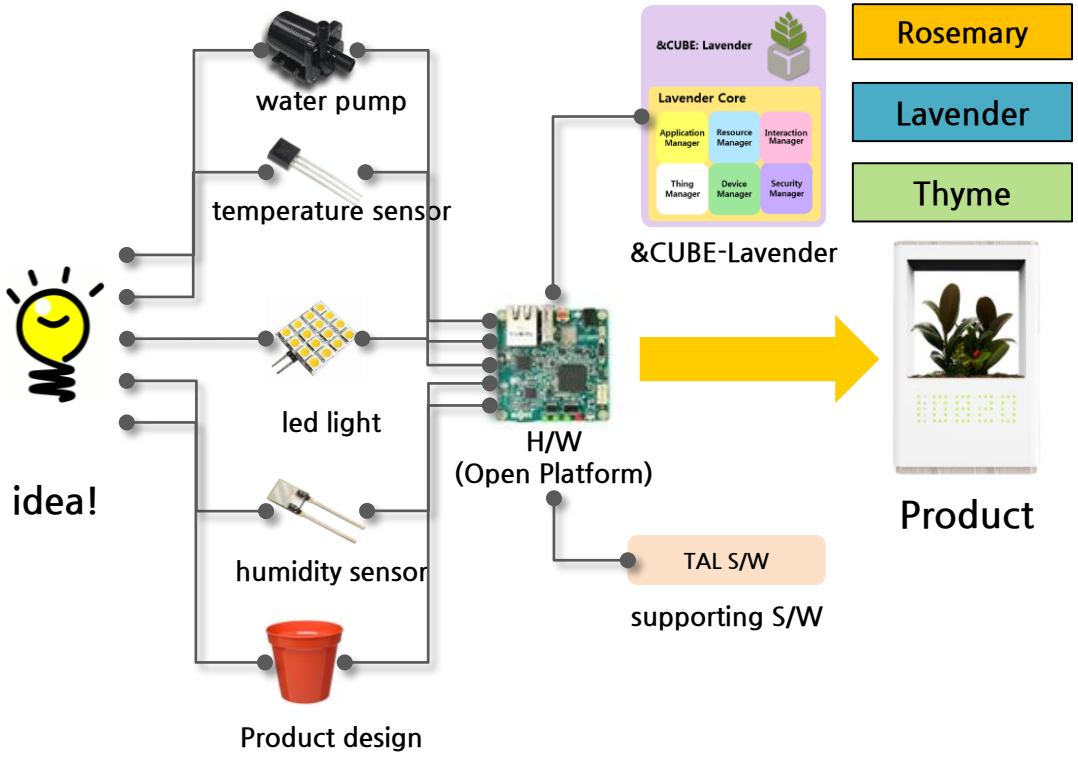
S/W name \ oneM2M Nodes		AE	ASN	CSE		Framework
				MN	IN	
Mobius	Blue Octopus				√	Spring
	Yellow Turtle				√	Node.js
	Rosemary			√		Java
				√		Node.js
&Cube	Lavender		√			Java
			√			Node.js
	Thyme	√				Java
		√				Node.js

Construction Device Application Platform

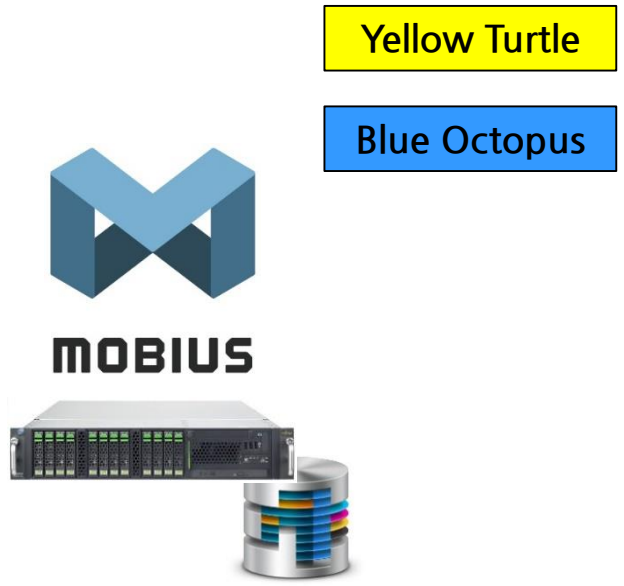
S/W name \ oneM2M Nodes		AE	CSE			Framework
			ASN	MN	IN	
Mobius	Blue Octopus				√	Spring
	Yellow Turtle				√	Node.js
&Cube	Rosemary			√		Java
				√		Node.js
	Lavender		√			Java
			√			Node.js
	Thyme	√				Java
		√				Node.js

1.2 &Cube:Thyme 활용 디바이스 개발 방법

Product = Idea + H/W + &CUBE + TAS



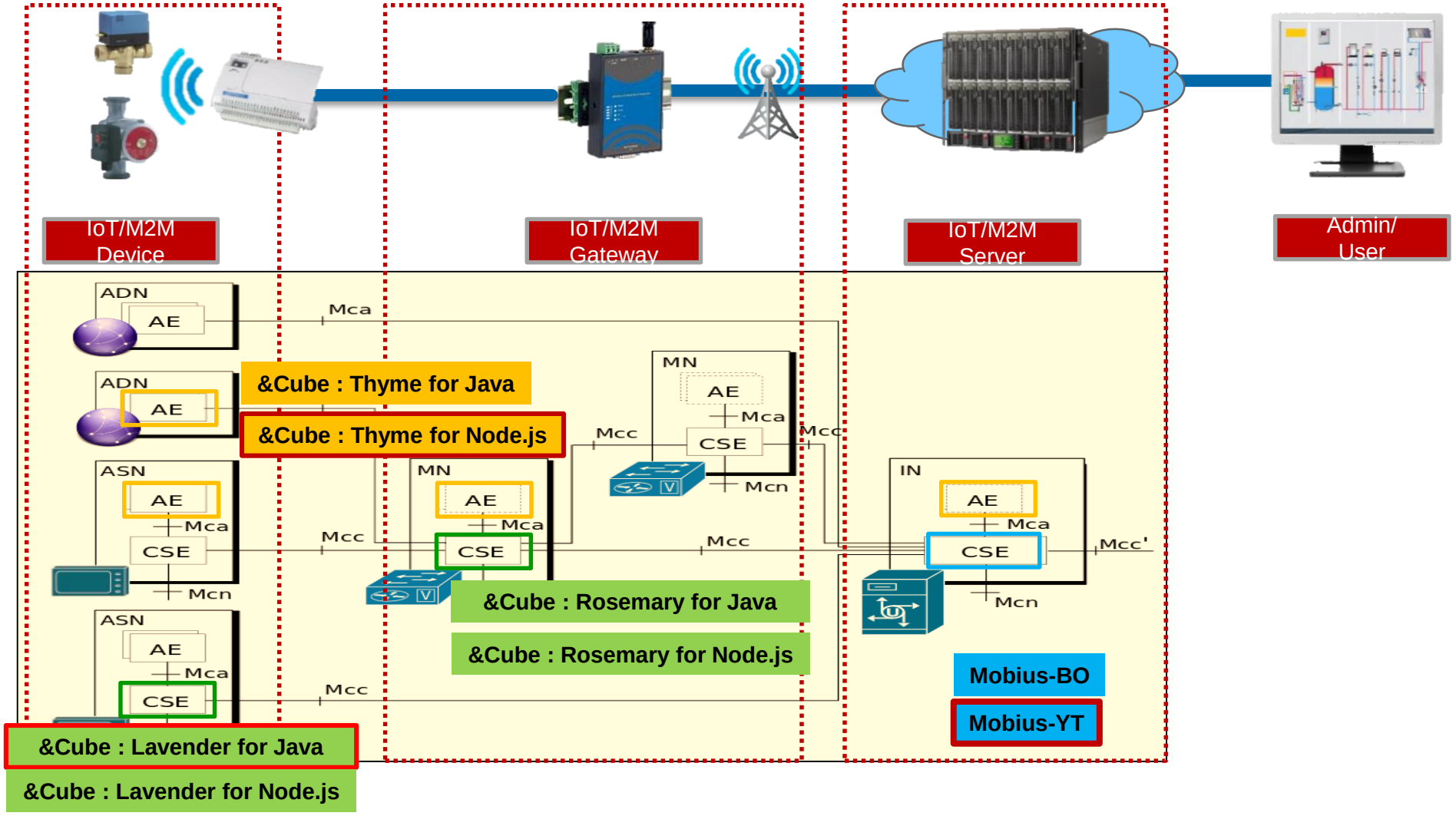
IoT Server = Server(H/W) + Mobius



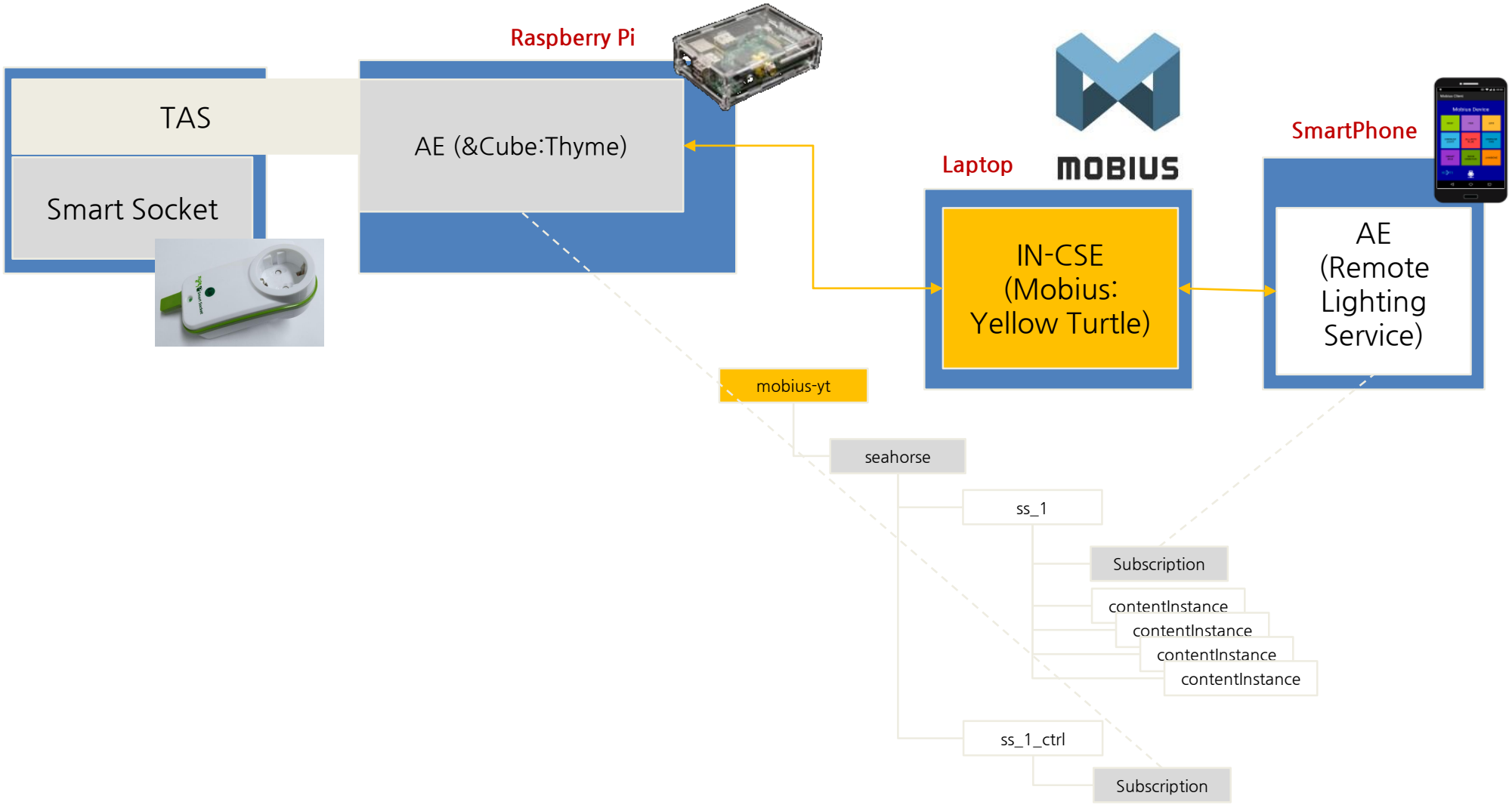
1. 개 요

1.1 오픈소스 종류 및 oneM2M 표준 준수
1.2 &Cube: Thyme 활용 디바이스 개발 방법
1.3 &Cube 연동 구조

1.3 Select open source platform in OCEAN to build the use case with oneM2M



1.3 &Cube 연동 구조


Raspberry Pi
TAS
Smart Socket
AE (&Cube:Thyme)
Laptop
MOBIUS
IN-CSE
(Mobius:
Yellow Turtle)
SmartPhone
AE
(Remote
Lighting
Service)
mobius-yt
seahorse
ss_1
Subscription
contentInstance
contentInstance
contentInstance
contentInstance
ss_1_ctrl
Subscription

2. 사물인터넷 디바이스 개발환경 구축

2.1 사물인터넷 디바이스 Open H/W 소개

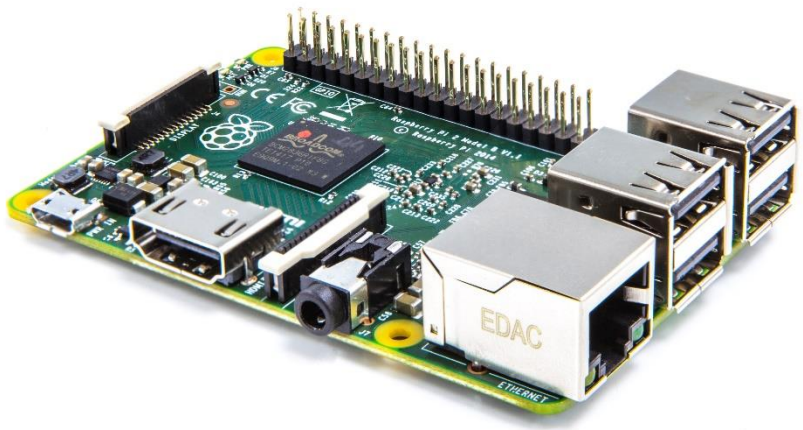
2.2 Open H/W 기반 개발환경 구축

2.3 &Cube: Thyme 구동환경 구축

2.1 사물인터넷 디바이스 Open H/W 소개

■ Raspberry Pi 2 Model B

- 영국의 라즈베리 파이 재단이 개발
- 기초 컴퓨터 과학 교육을 증진시키기 위해 만든 싱글 보드 컴퓨터
- Raspbian (Debian 계열 Linux) 운영체제 사용
- 기타 운영체제 포팅 가능
- 상세 정보는 http://www.raspberrypi.org/ 홈페이지에서 확인 가능



Item	Name	Feature
Processor	BCM2836	Broadcom BCM2836 SoC - 900Mhz ARM Cortex-A7 Quad core - Broadcom VideoCore IV Dual core - 1GByte Memory
Ethernet	LAN9514	SMSC LAN9514-jzx 10/100Mbps Ethernet Port - RJ-45 connector
USB	Host	USB 2.0 HS/FS/LS Host Port x 4
SD/MMC	microSD	1 microSD Socket (SDMMC2—BOOT)
Video Input	Connector	15Pin MIFI Camera interface connector x 2 - Raspberry Pi Camera - NoIR Camera
Video Output	HDMI	PAL/NTSC 640x350 ~ 1920x1200 resolution
	RCA	PAL/NTSC, Audio output
Audio Input	I2S	Inter-IC Sound, Integrated Interchip Sound
Audio Output	I2S	Inter-IC Sound, Integrated Interchip Sound - 3.5mm Audio jack - HDMI Digital Audio
Power	DC 5V	5V 800mA DC Input (Micro 5pin connector)
Expansion	Connector	GPIO 17 port, UART bus, I2C bus, SPI 2 port, I2S Audio port, +3.3V, +5V, Ground, HAT ID bus
Dimension	width x height	85.6 x 56.5 (mm)

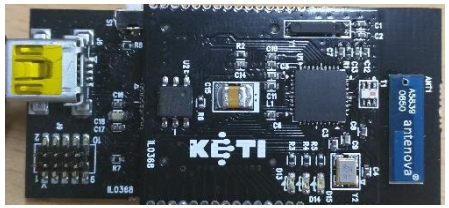
2.1 사물인터넷 디바이스 Open H/W 소개

■ Smart Socket

- 전력량 측정을 수행하고 ON/OFF 신호를 통해 전력차단이 가능한 디바이스
- 실제 전력 측정 및 전력차단을 수행하는 보드와 무선 통신으로 데이터 송신 및 제어 수신을 수행하는 보드의 2중 보드 형태
- TI CC2530을 이용한 ZigBee RF 무선 통신 수행
- 주기적으로 현재 전력량을 측정하여 Sink node로 전송
- Sink node로부터 제어 신호 수신 시 전원공급/차단 수행

■ Sink node

- TI CC2530
- CP2103 USB-to-Serial
- Mini USB 활용 연결 인터페이스 제공

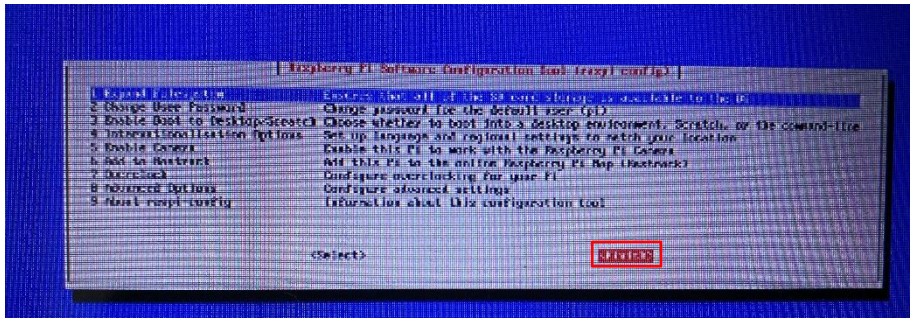


2. 사물인터넷 디바이스 개발환경 구축

2.1 사물인터넷 디바이스 Open H/W 소개
2.2 Open H/W 기반 개발환경 구축
2.3 &Cube: Thyme 구동환경 구축

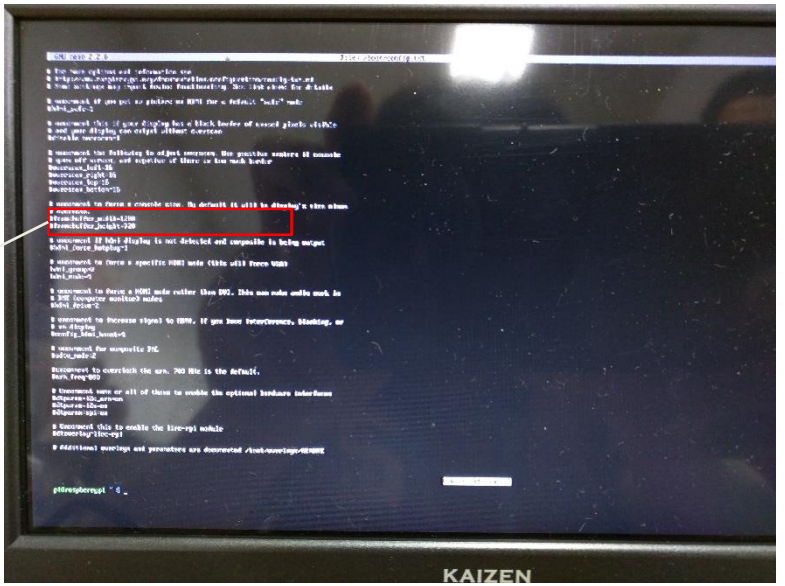
2.2 Open H/W 기반 개발환경 구축

- Raspberry-Pi 부팅
  - HDMI 케이블 및 전원케이블 연결
  - 초기 설정에서 Finish 선택



- Raspberry-Pi 해상도 설정 (# 삭제 및 숫자 변경)

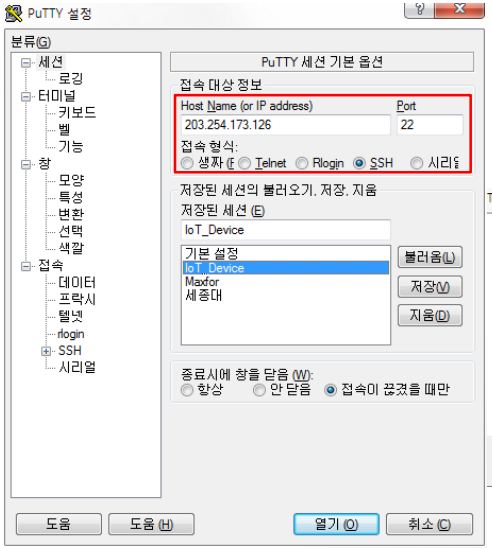
```
pi@raspberrypi ~ $ sudo nano /boot/config.txt
.....
framebuffer_width=800
framebuffer_height=480
.....
<Ctrl>+<X> → Y → <Enter>
```

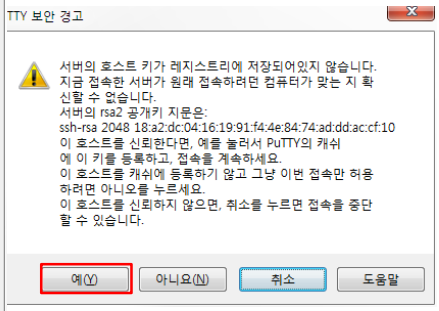


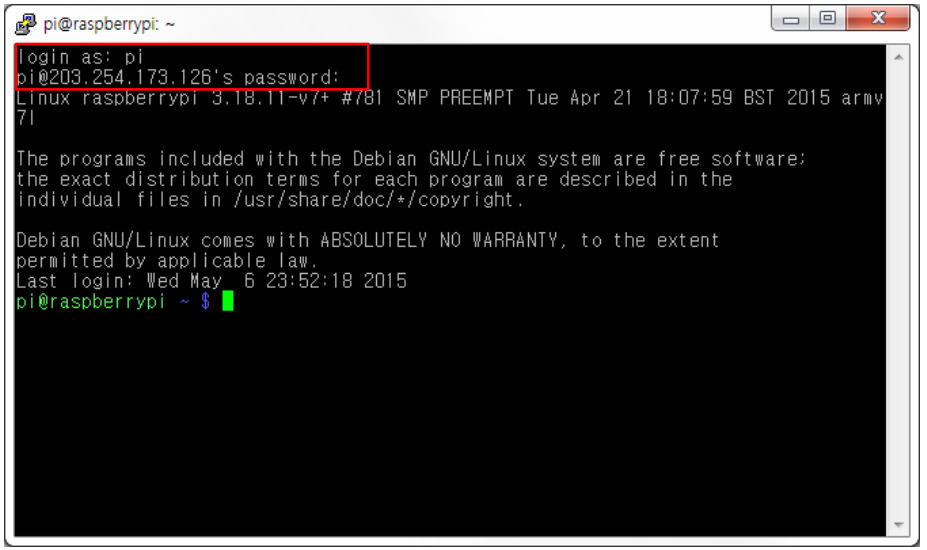
2.2 Open H/W 기반 개발환경 구축

■ 원격 접속환경 구축 (SSH client 사용)

- SSH client 프로그램 다운로드 및 설치 (PuTTY)
  - http://www.chiark.greenend.org.uk/~sgtatham/putty/
- PuTTY 프로그램 설정 및 연결
  - Raspberry-Pi의 IP address 입력을 통해 접속
  - 호스트 키 확인 메시지 출력 시 '예'를 눌러 캐쉬에 등록
  - 접속 완료 시 로그인 진행 (Raspberry-Pi 초기 설정 → ID : pi, PW : raspberry)







2.2 Open H/W 기반 개발환경 구축

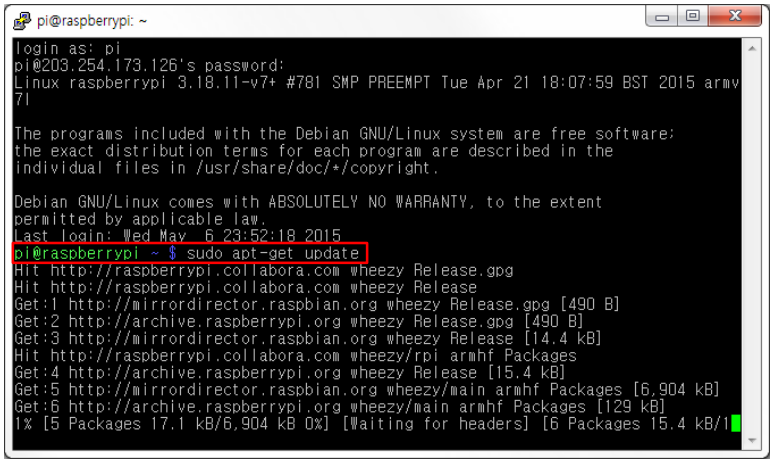
■ Samba 서버 구축

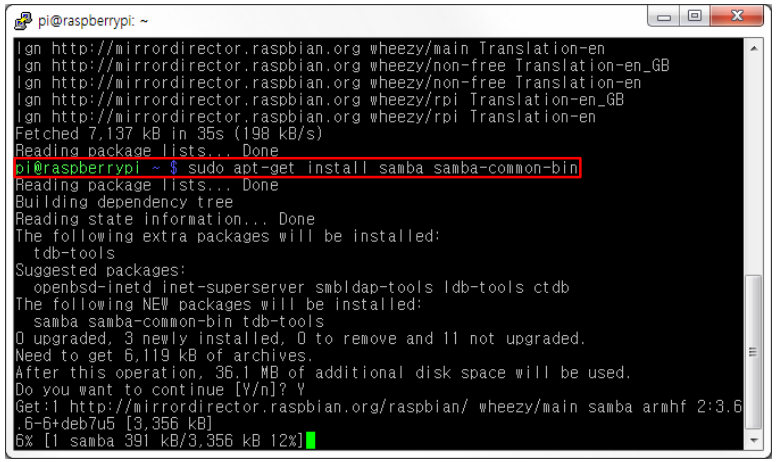
- Repository 업데이트

```
pi@raspberrypi ~ $ sudo apt-get update
.....
Reading package lists... Done
```

- Samba 서버 설치

```
pi@raspberrypi ~ $ sudo apt-get install samba samba-common-bin
.....
Do you want to continue [Y/n]? Y
```



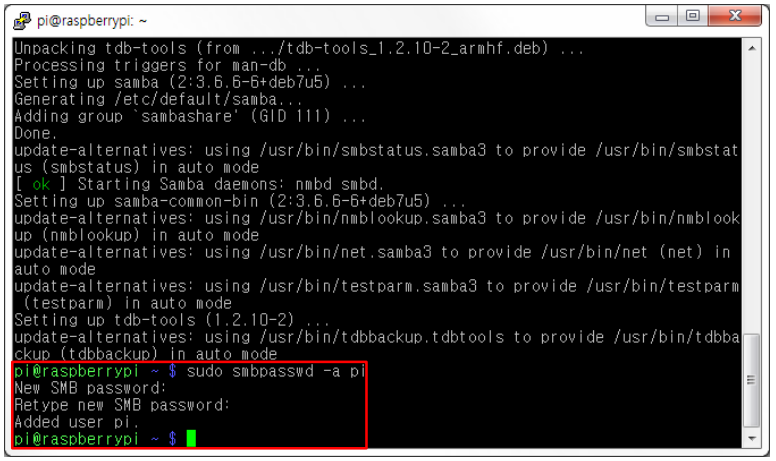


2.2 Open H/W 기반 개발환경 구축

■ Samba 서버 구축

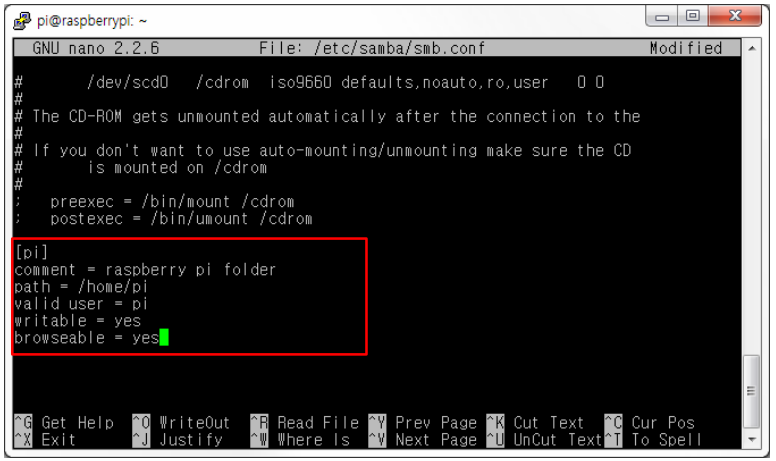
● Samba 서버 사용자 추가

```
pi@raspberrypi ~ $ sudo smbpasswd –a pi
New SMB password: (원하는 패스워드 입력)
Retype new SMB password: (원하는 패스워드 입력)
Added user pi.
```



● Samba 서버 사용자 설정

```
pi@raspberrypi ~ $ sudo nano /etc/samba/smb.conf
…… (가장 마지막 줄 밑에)
[pi]
comment = raspberry pi folder
path = /home/pi
valid user = pi
writable = yes
browseable = yes
<Ctrl>+<X> → Y → <Enter>
```

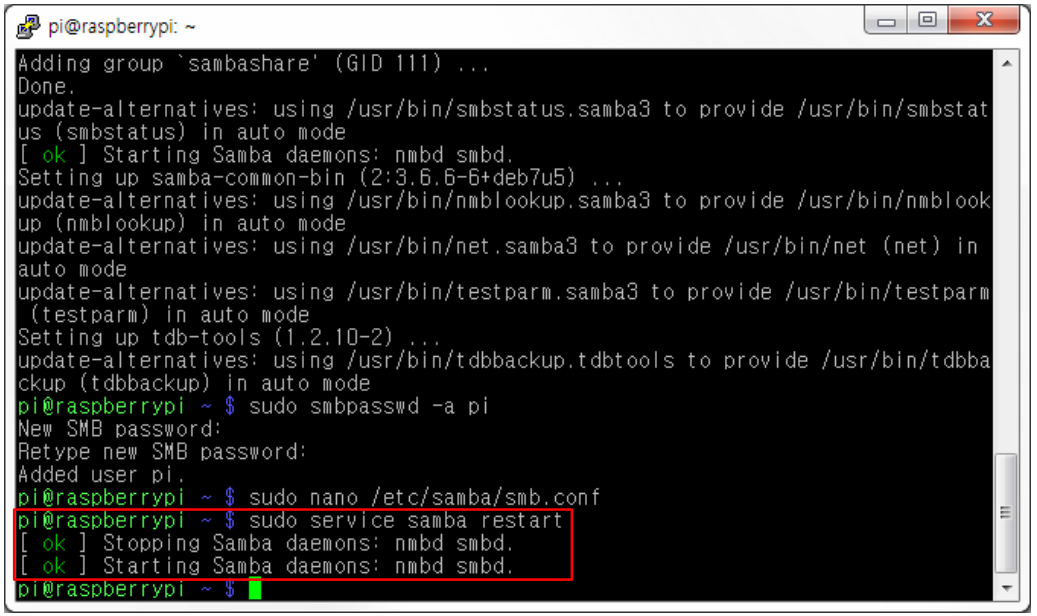


2.2 Open H/W 기반 개발환경 구축

■ Samba 서버 구축

● Samba 서버 재시작

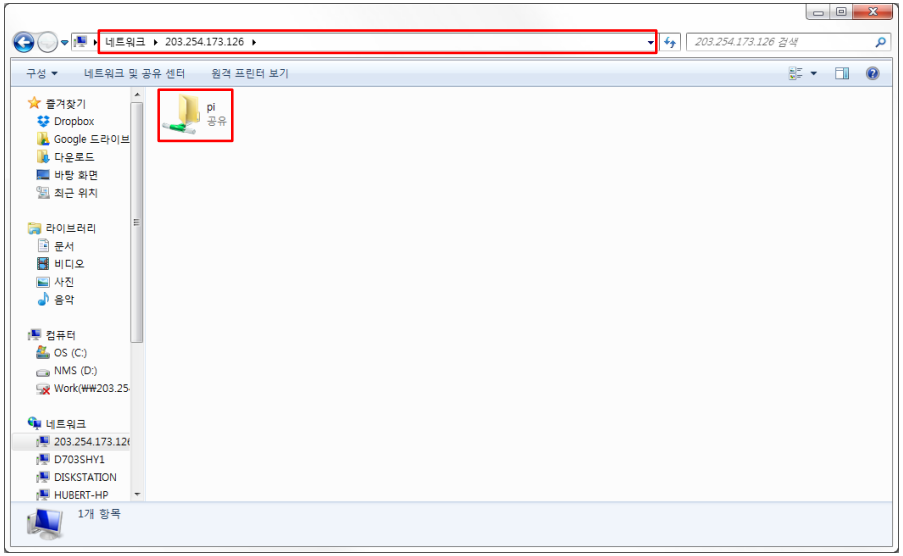
```
pi@raspberrypi ~ $ sudo service samba restart
[ ok ] Stopping Samba daemons: nmbd smbd.
[ ok ] Starting Samba daemons: nmbd smbd.
```

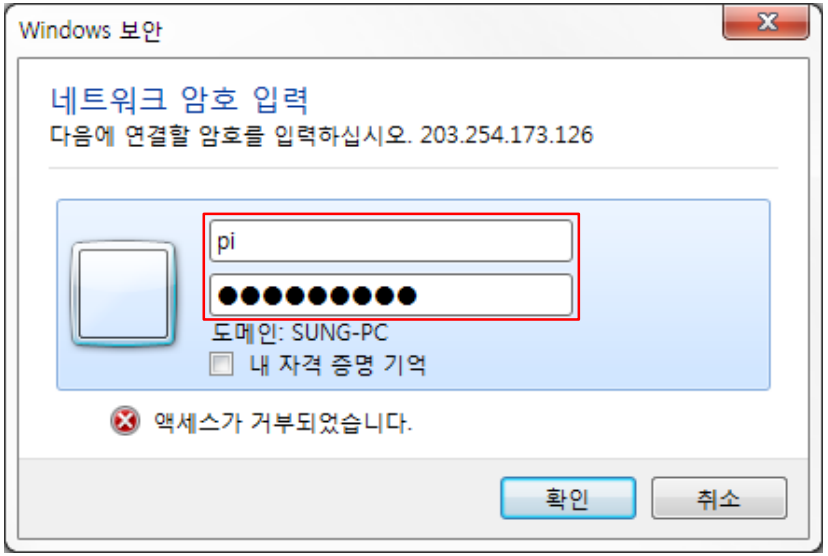


2.2 Open H/W 기반 개발환경 구축

■ Samba 폴더 연결

- Windows 탐색기 실행
- 주소입력 창에 ₩₩Raspberry-Pi IP 주소 입력
  - 예) ₩₩203.254.173.126
- pi 폴더 더블클릭
- 계정 및 패스워드 입력
  - 계정 : pi, 패스워드 : Samba서버 설치 시 지정한 패스워드





2. 사물인터넷 디바이스 개발환경 구축

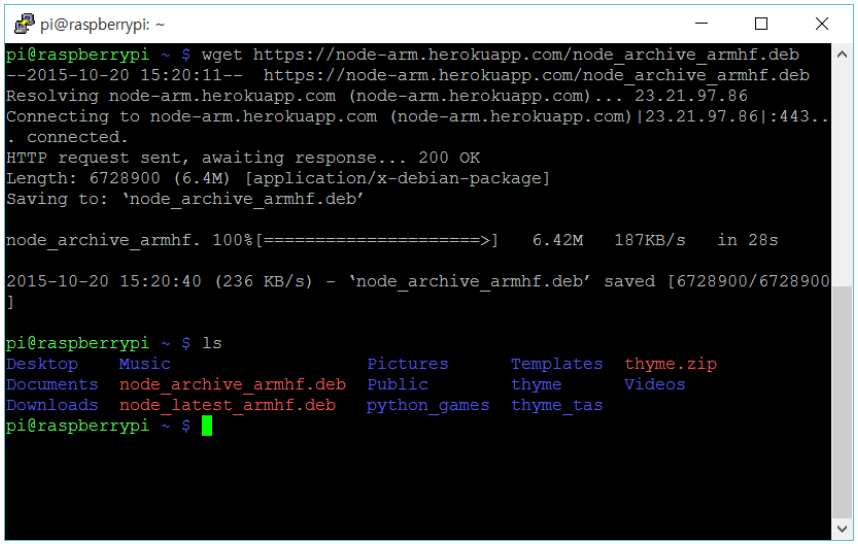
2.1 사물인터넷 디바이스 Open H/W 소개
2.2 Open H/W 기반 개발환경 구축
2.3 &Cube: Thyme 구동환경 구축

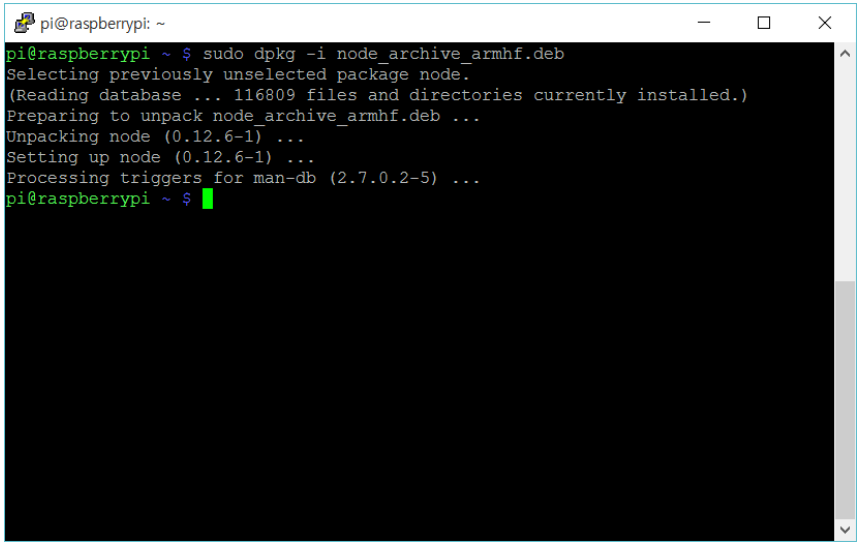
2.3 &Cube: Thyme 구동환경 구축

■ Node.js 패키지 설치

● Node.js 다운로드 및 설치

```
pi@raspberrypi ~ $ mkdir node
pi@raspberrypi ~ $ cd node
pi@raspberrypi ~/node $ sudo wget https://node-arm.herokuapp.com/node_archive_armhf.deb
pi@raspberrypi ~/node $ sudo dpkg –i node_archive_armhf.deb (패키지 설치 명령어)
pi@raspberrypi ~/node $ node –v (버전 확인 명령어)
pi@raspberrypi ~/node $ npm -v (추가 라이브러리 설치도구 버전 확인 명령어)
```



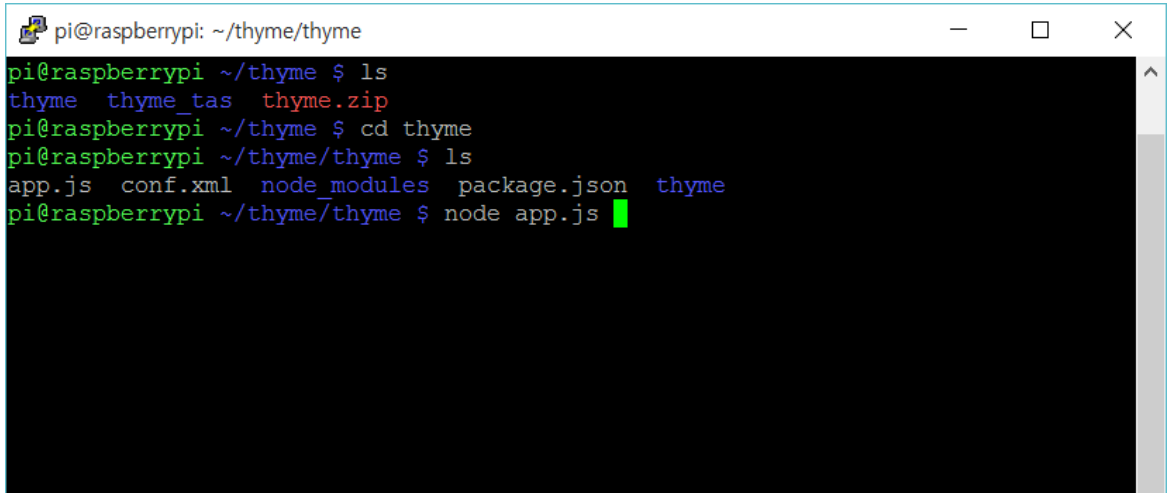


2.3 &Cube: Thyme 구동환경 구축

■ &Cube: Thyme 샘플 다운로드 및 구동 테스트

- www.iotocean.org 에서 &Cube:Thyme 다운로드
- 구동 테스트 → &Cube: Thyme 실행

```
pi@raspberrypi ~/node/thyme $ sudo npm install (추가 라이브러리 설치 명령)
pi@raspberrypi ~/node/thyme $ node app.js (실행 명령)
```



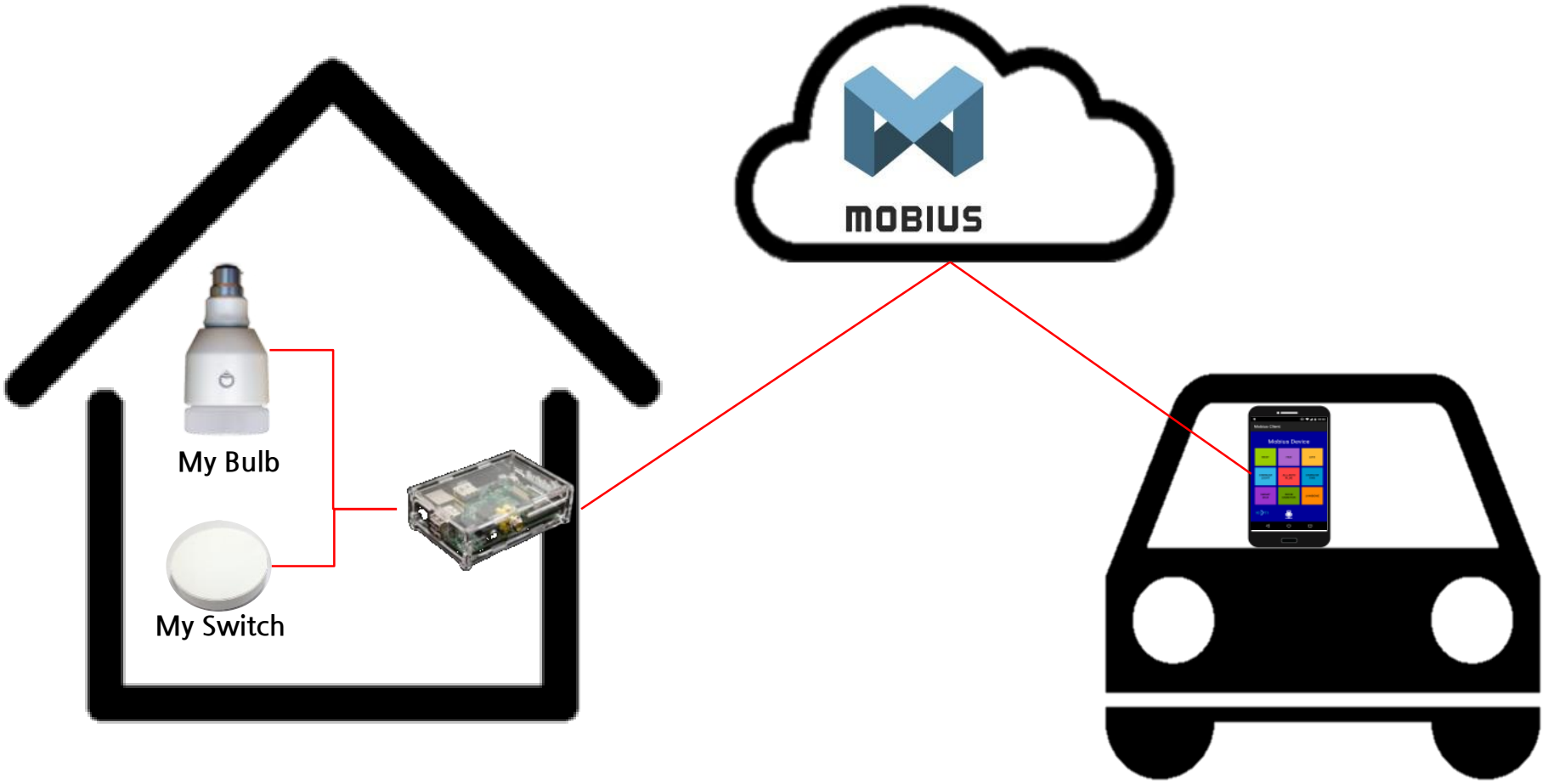
3. 사물인터넷 디바이스 개발 시연

3.1 서비스 시나리오

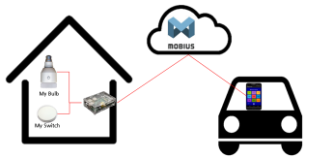
3.2 Yellow Turtle 구축 시연

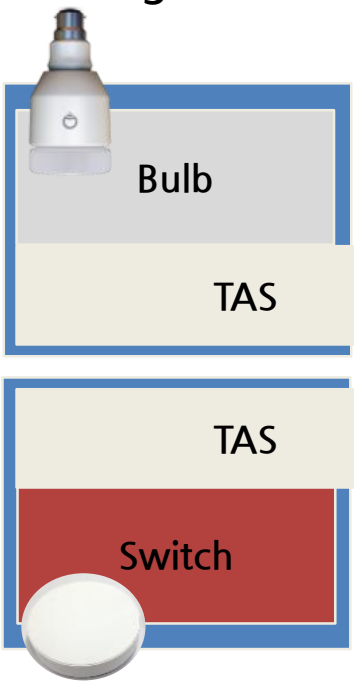
3.3 &Cube: Thyme 구동 시연

Case study for My Lighting Service



Configuration for My Lighting Service





Raspberry Pi

AE-Bulb (&Cube:Thyme)

AE-Switch (&Cube:Thyme)

AE-HomeLight (&Cube:Thyme)

ASN-CSE (&Cube:Lavendar for Java)

Laptop

MOBIUS

IN-CSE (Mobius: Yellow Turtle)

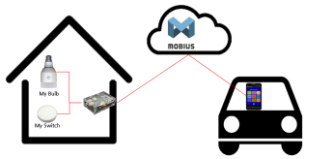
SmartPhone

AE (Remote Lighting Service)

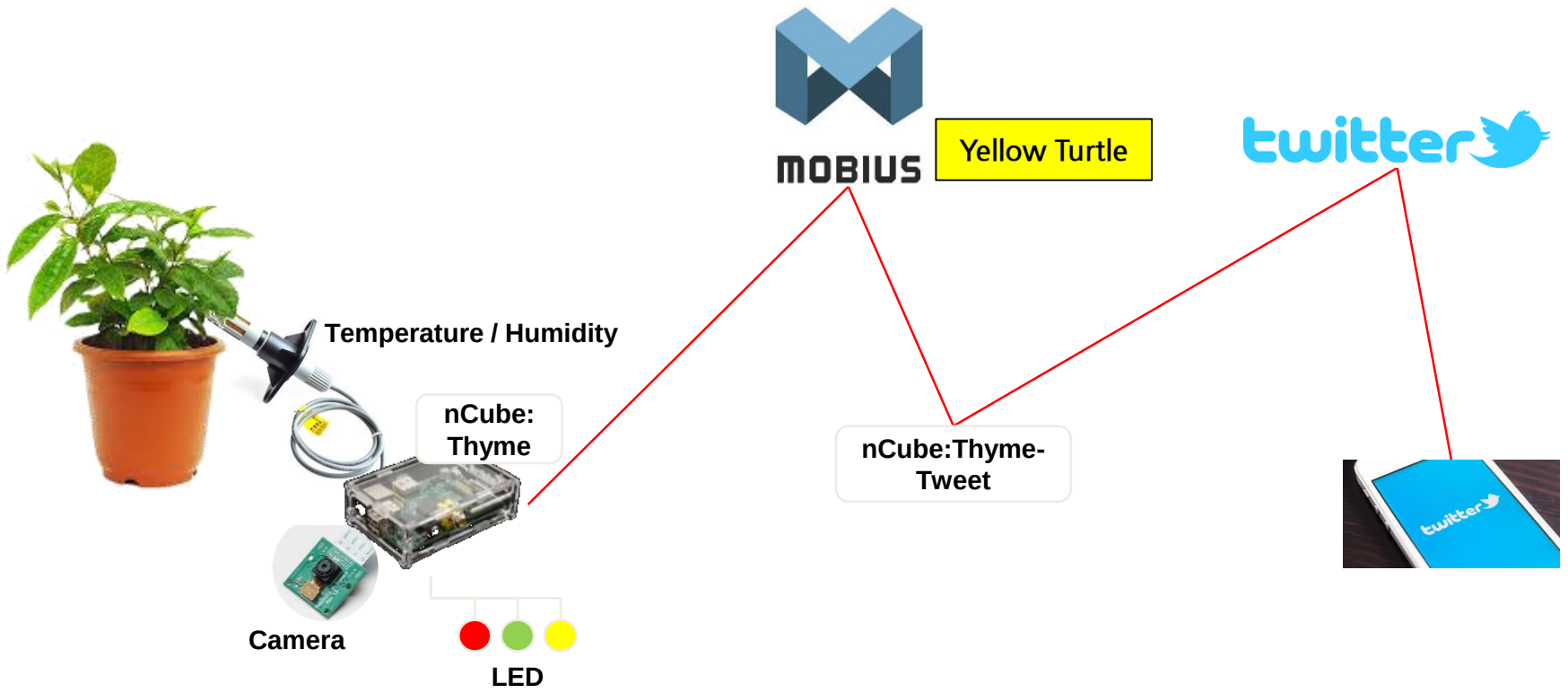
CSE-nCube
- AE-Bulb
  - CNT_Status
    - cin
  - CNT_control
    - Subscription
- AE-Switch
  - CNT_Status
    - cin
  - Subscription

CSE-Mobius
- CSE-nCube
- AE-HomeLight
  - CNT-MyLightStatus
    - Subscription
  - CNT-MyLightControl
    - Subscription

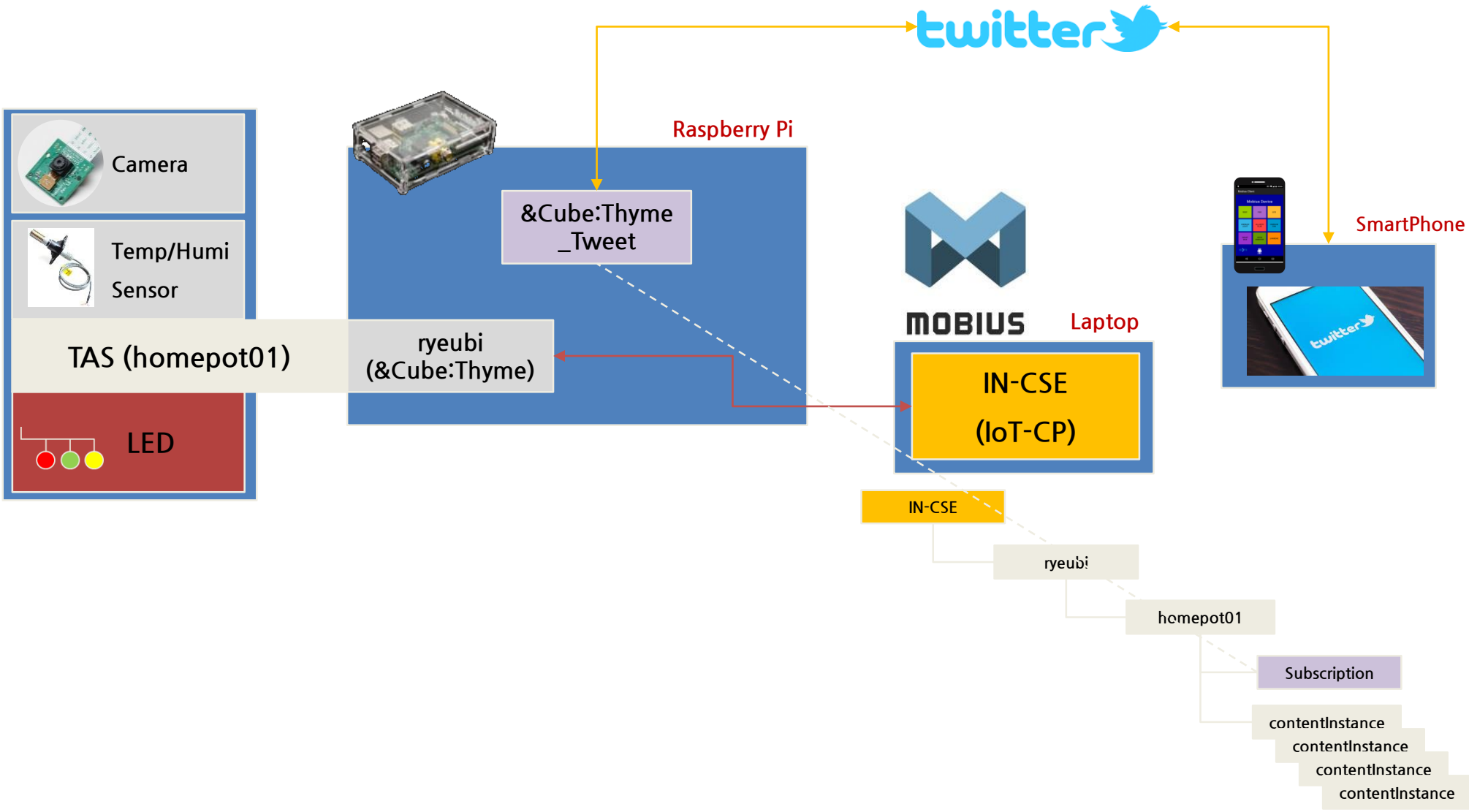
Configuration for My Lighting Service



Case study for Smart Pot Service (Demo)



Configuration for Smart Pot Service (Demo)

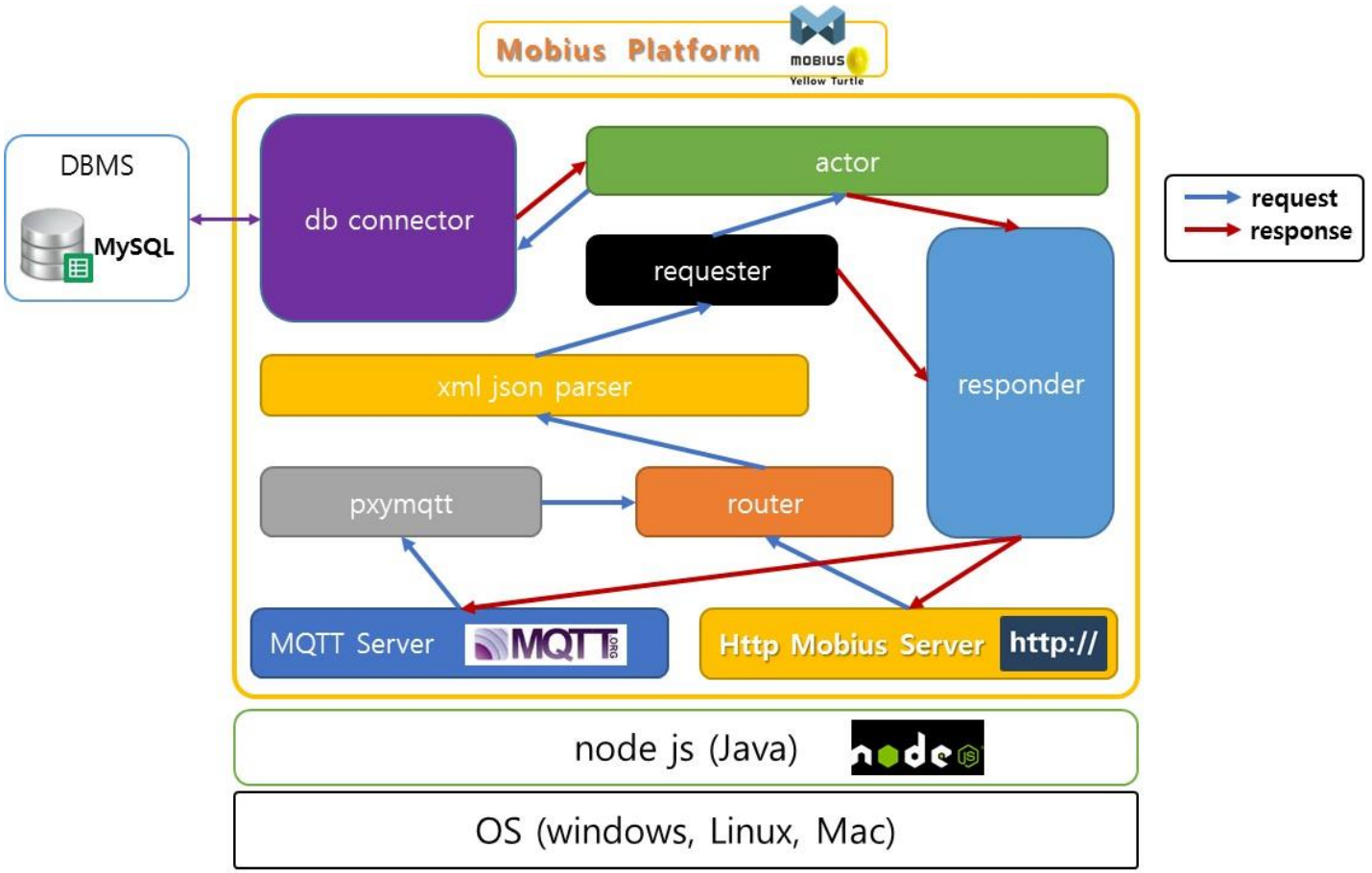


3. 사물인터넷 디바이스 개발 시연

3.1 서비스 시나리오
3.2 Yellow Turtle 구축 시연
3.3 &Cube: Thyme 구동 시연

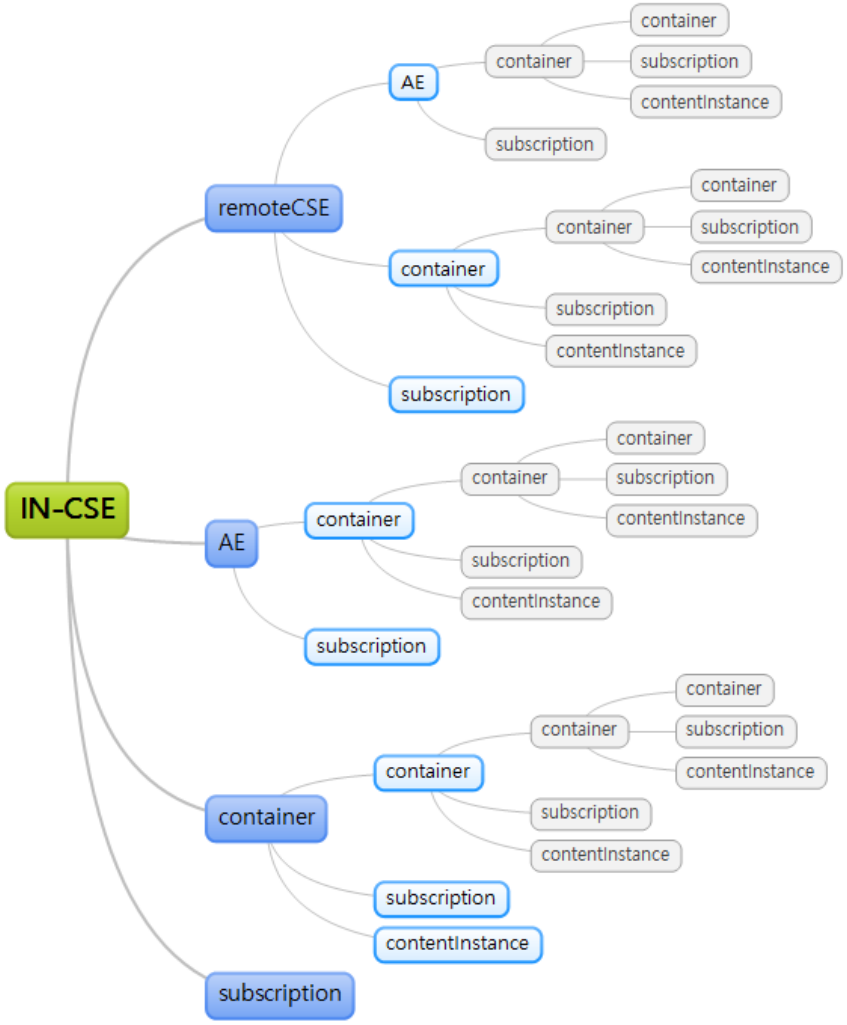
Mobius Yellow Turtle 서버 플랫폼 S/W 아키텍처

- Mobius Yellow Turtle : Node JS를 기반으로 Java Script로 개발
- DB는 MySQL을 지원하고 있으며 통신 프로토콜은 HTTP, MQTT를 지원



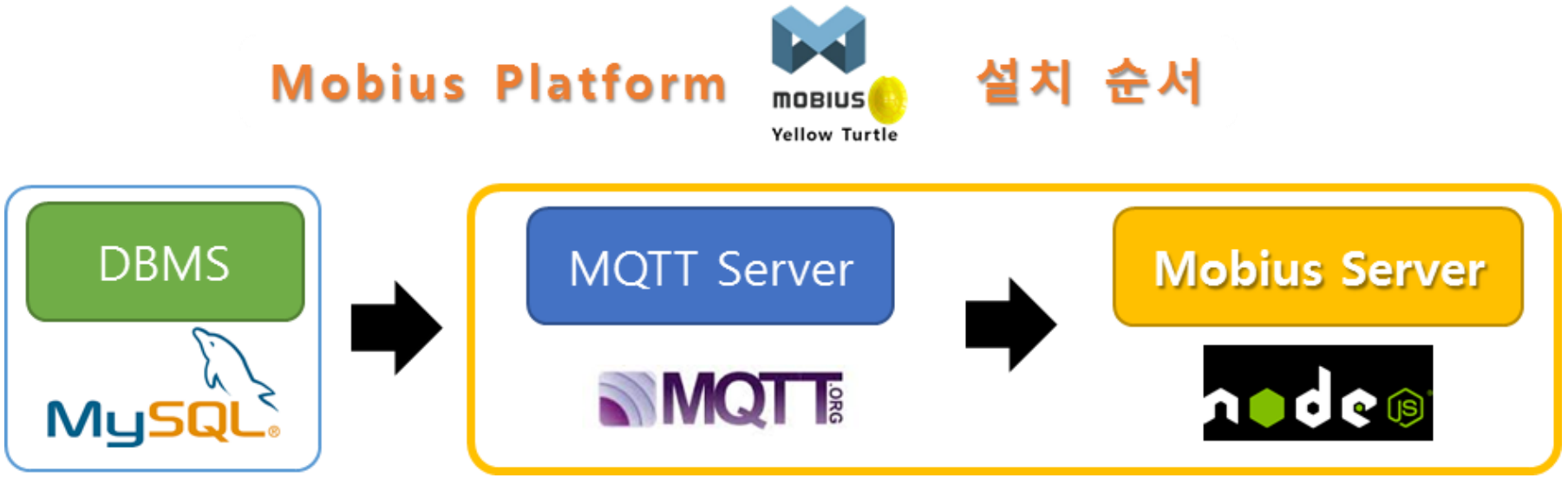
Mobius Yellow Turtle 서버 지원 Resource 구조

http://hostname:port/IN-CSE_name/AE_name/container_name/contentInstance_name



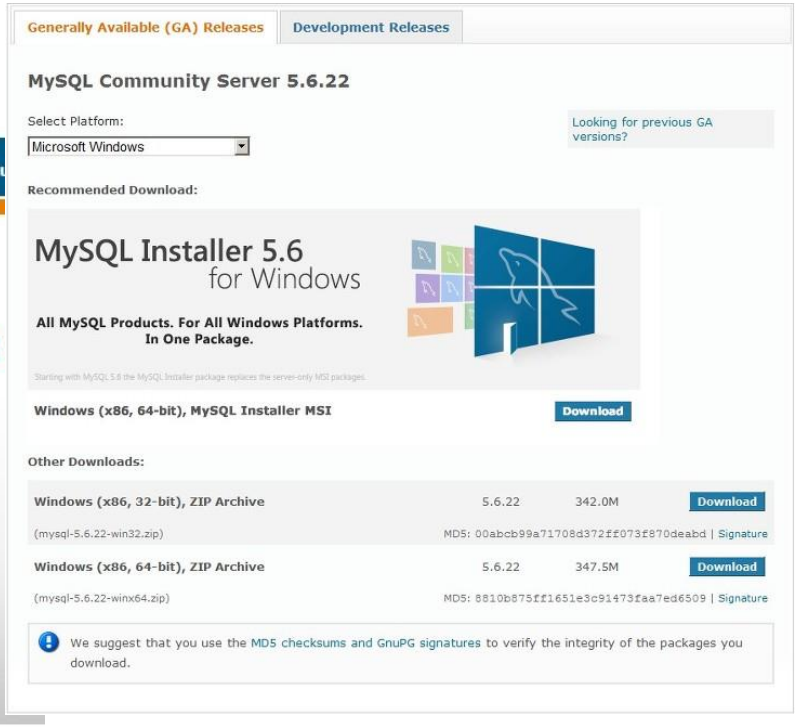
Construction Server Platform with Mobius : Yellow Turtle

■ Reference install guide of Yellow Turtle in OCEAN



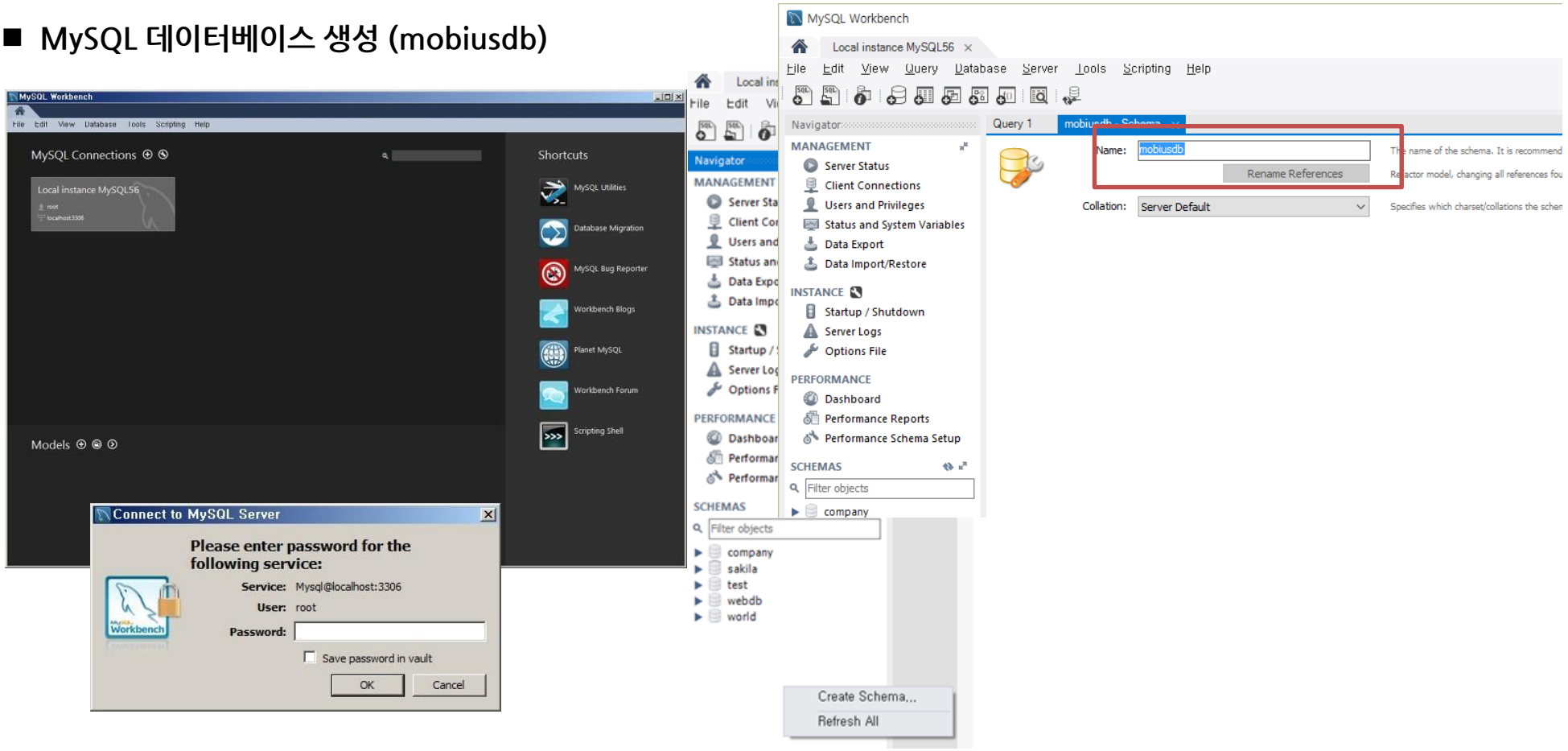
Construction Server Platform with Mobius : Yellow Turtle

■ MySQL 설치 (MySQL Server , MySQL Workbench)



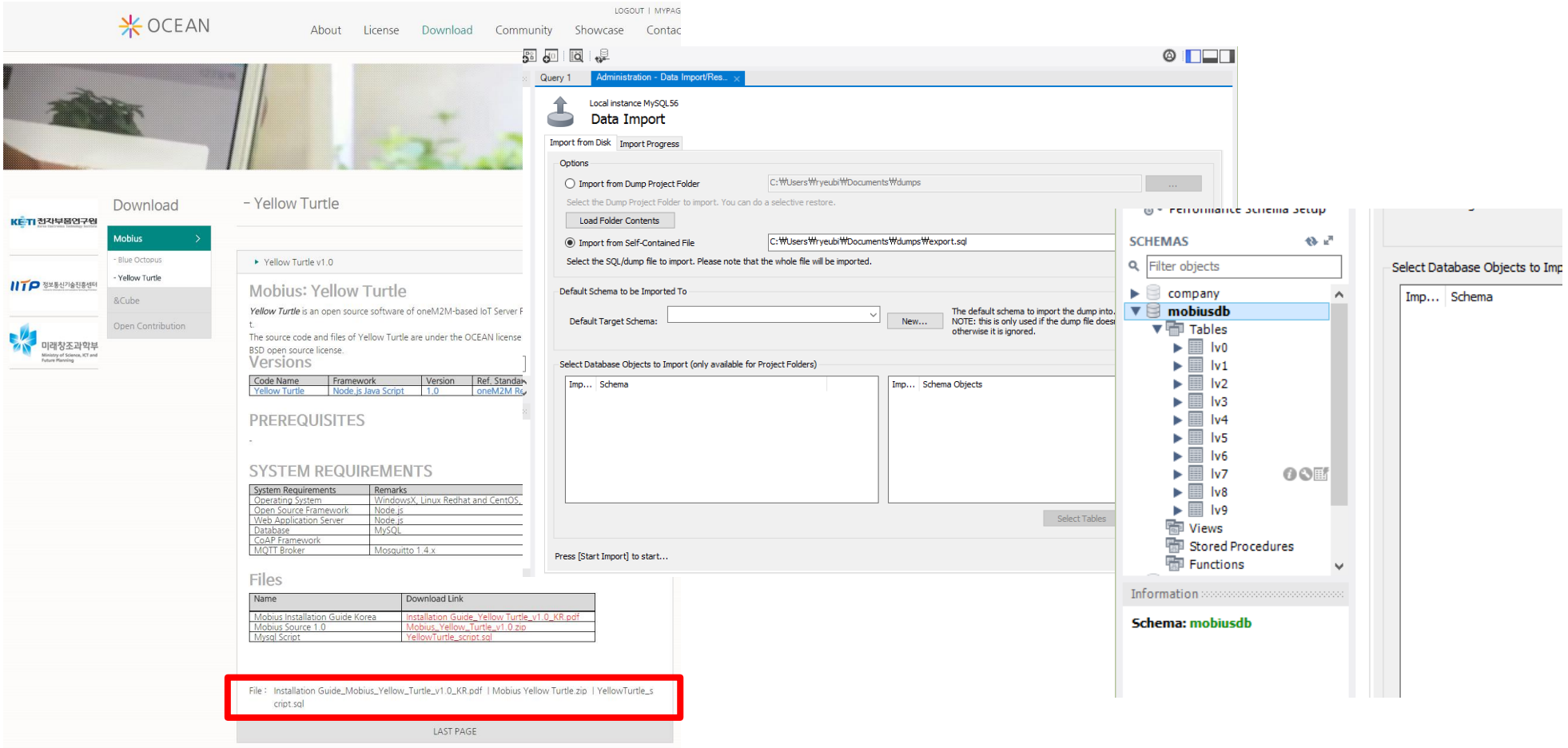
Construction Server Platform with Mobius : Yellow Turtle

■ MySQL 데이터베이스 생성 (mobiusdb)



Construction Server Platform with Mobius : Yellow Turtle

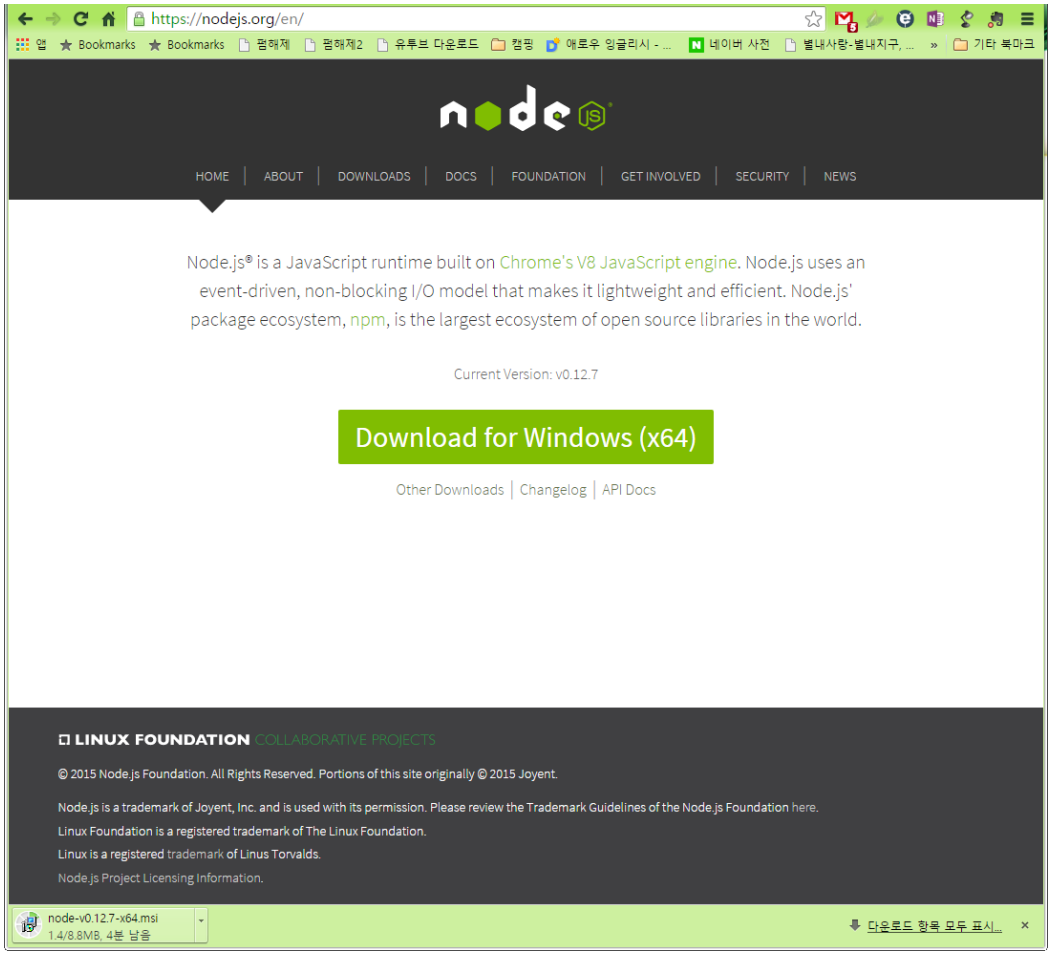
■ MySQL mobiusdb 테이블 가져오기 (www.iotocean.org)



Construction Server Platform with Mobius : Yellow Turtle

■ Node.js

- 고성능의 비동기 IO (Async/Non-blocking IO)를 지원하는 single thread 기반 네트워크 서버
- 2009년 Ryan Dahl에 의해 개발이 시작되었고 현재 수많은 지원 모듈을 가지고 있는 오픈 소스 프로젝트 중 하나
- Google Chrome V8 엔진으로 개발되어 있고 Event 기반의 프로그래밍 모델로써 프로그래밍 언어로는 Java script를 사용
- 현재 많은 인터넷 기업들이 node.js를 도입

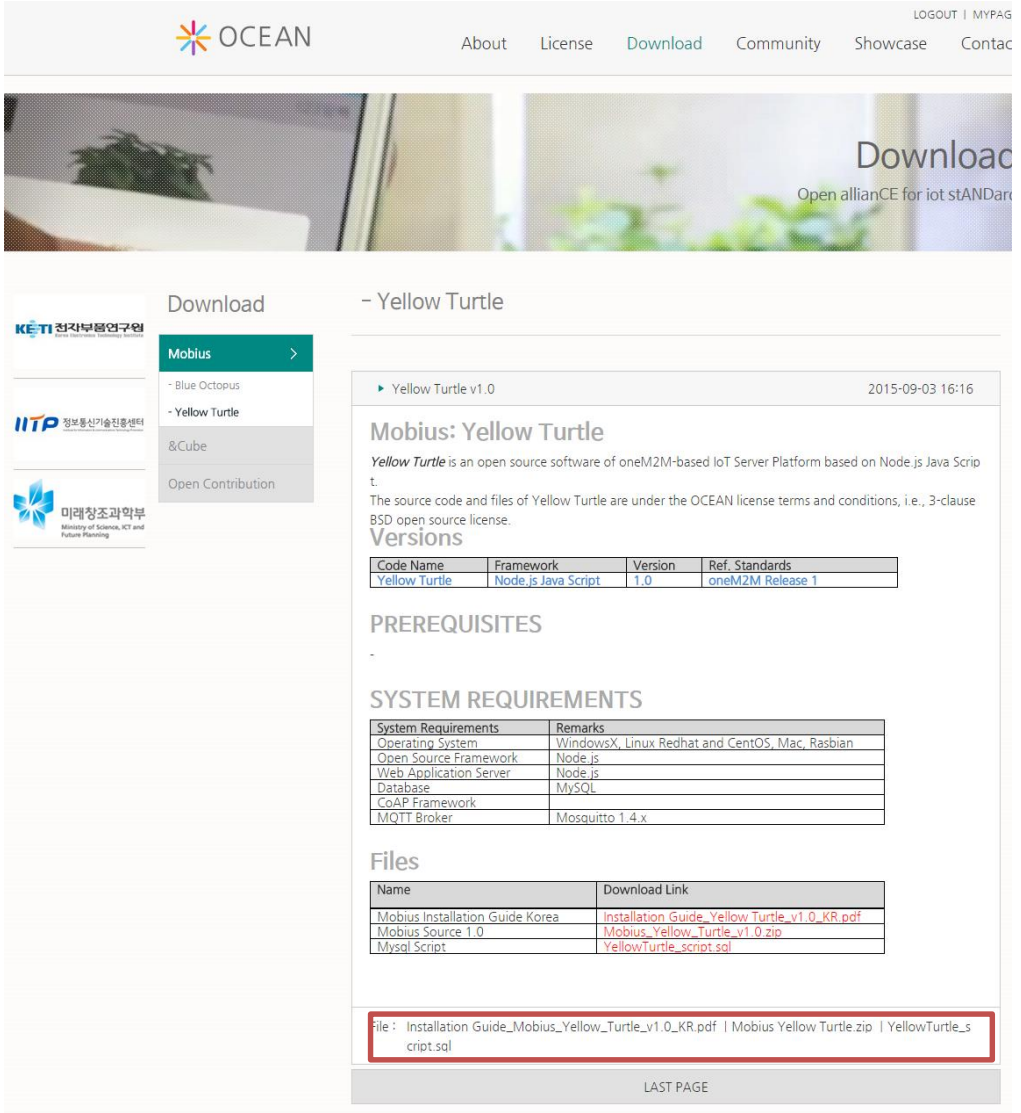


■ Node.js 설치

- http://www.nodejs.org

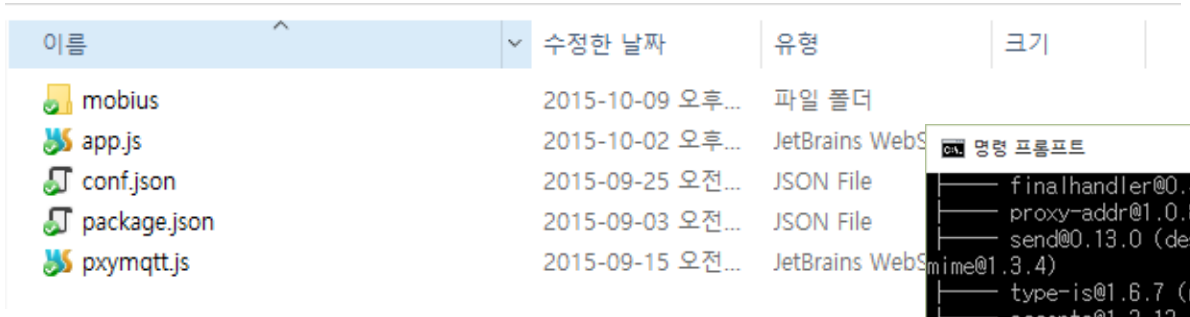
Construction Server Platform with Mobius : Yellow Turtle

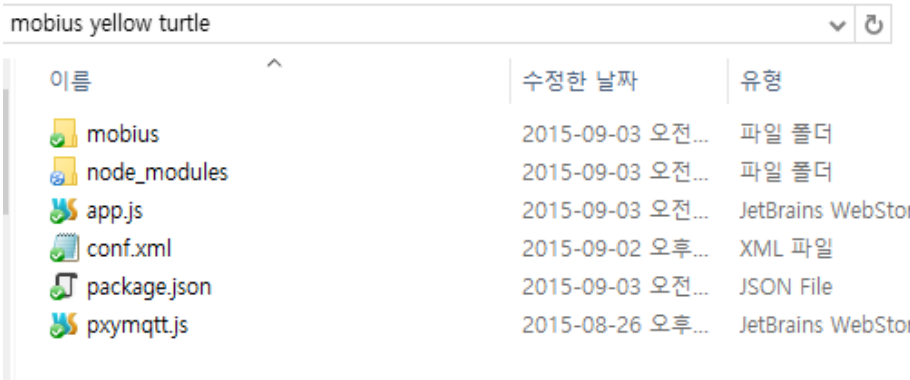
- Mobius-YT 서버 설치
  - OCEAN Alliance 사이트 (http://www.iotocean.org)
  - Download Mobius:Yellow Turtle from OCEAN

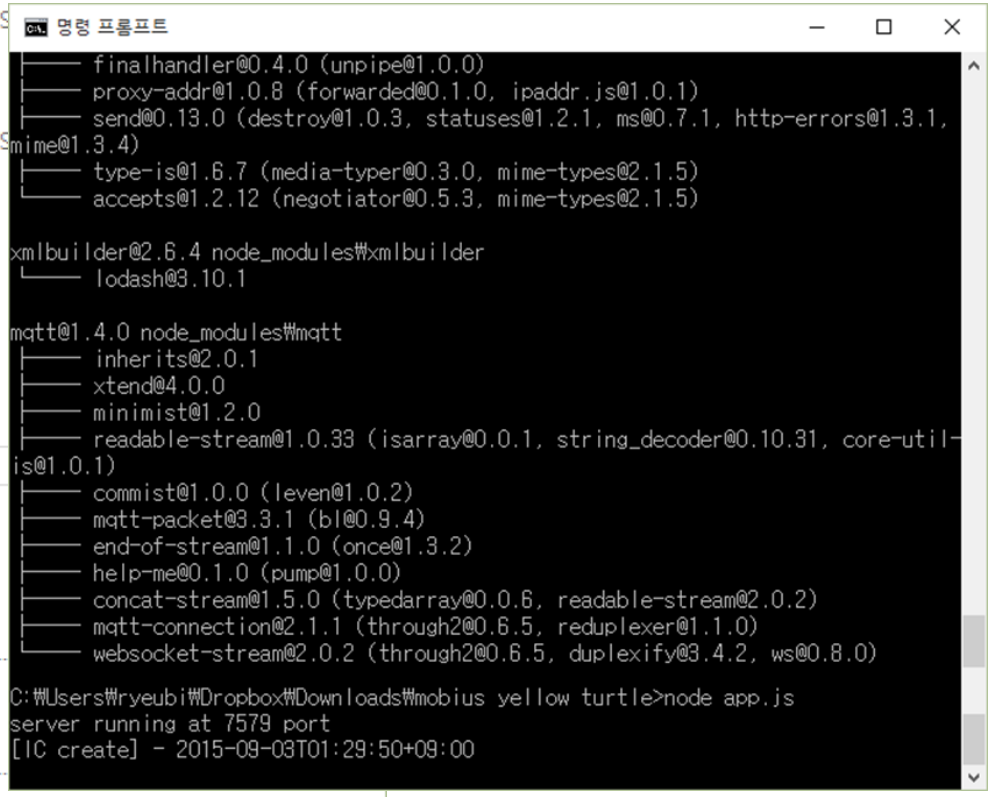


Construction of Mobius:Yellow Turtle

■ Extract zip and run 'npm install' in command prompt at the folder of mobius





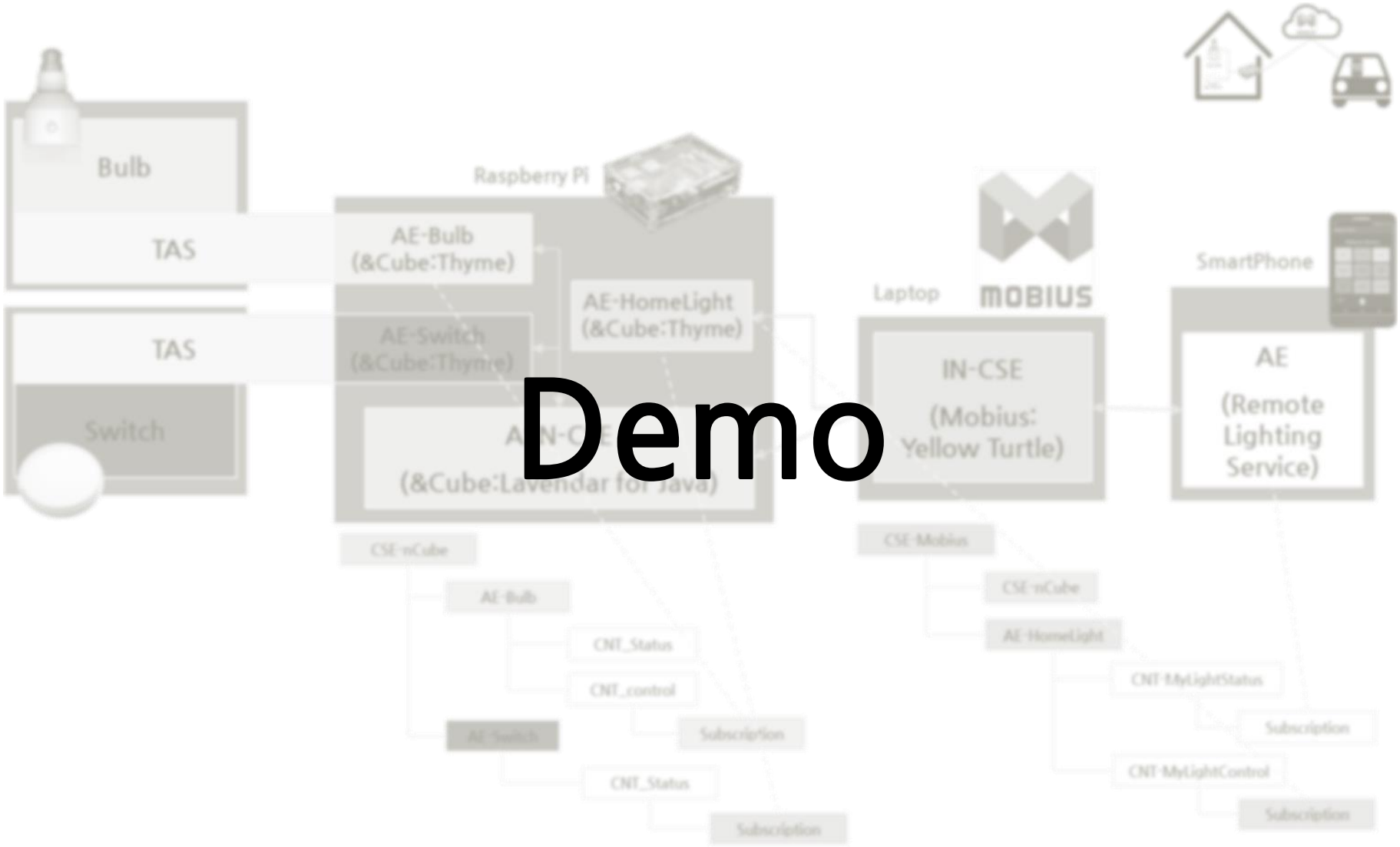


Construction of Mobius:Yellow Turtle

■ Setting of configuration file (conf.json) for mobius

- cbtype: CSE 타입 설정 (in or mn)
- nmtype: name type 설정 (long or short)
- bodytype: body content type 설정 (xml or json)
- in-cse: 서버의 정보 설정
  - cseid
  - csebase: 서버의 root 이름, root path
  - cseport : 서버가 오픈하는 포트 번호
  - dbhost: DB 서버 주소
  - dbpass: DB 접속 암호
  - mqttproxy: mqtt proxy 호스트 주소
  - mqttproxyport: mqtt proxy 포트 번호
- mn-cse: 미들노드일 때 in-cse의 정보
  - cbhost: in-cse 주소
  - cbport: in-cse 포트 번호
  - cbname: in-cse base name
  - cbcseid: in-cse의 cseid

```
{
    "m2m:conf": {
        "cbtype": "in",
        "nmtype": "short",
        "bodytype": "json",
        "in-cse": {
            "cseid": "/0.2.481.1.1.1.1",
            "csebase": "mobius-yt",
            "cseport": "7579",
            "dbhost": "localhost",
            "dbpass": "dksdlfduq2",
            "mqttproxy": "localhost",
            "mqttproxyport": "9726"
        },
        "mn-cse": {
            "cbhost": "203.253.128.150",
            "cbport": "9000",
            "cbname": "mobius",
            "cbcseid": "mobius"
        }
    }
}
```


Bulb
TAS
TAS
Switch
Raspberry Pi
AE-Bulb
(&Cube:Thyme)
AE-Switch
(&Cube:Thyme)
AE-HomeLight
(&Cube:Thyme)
(&Cube:Lavendar for Java)
Laptop
MOBIUS
IN-CSE
(Mobius:
Yellow Turtle)
SmartPhone
AE
(Remote
Lighting
Service)
CSE-nCube
AE-Bulb
CNT_Status
CNT_control
Subscription
AE-Switch
CNT_Status
Subscription
CSE-Mobius
CSE-nCube
AE-HomeLight
CNT-MyLightStatus
Subscription
CNT-MyLightControl
Subscription

Demo

3. 사물인터넷 디바이스 개발 시연

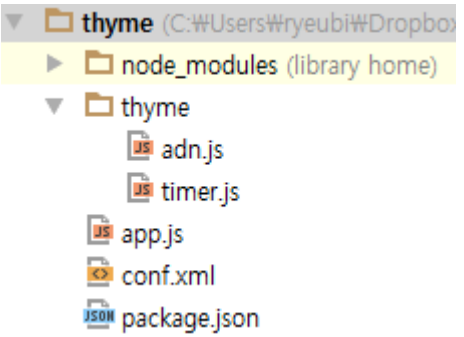
3.1 서비스 시나리오
3.2 Yellow Turtle 구축 시연
3.3 &Cube: Thyme 구동 시연

Development Device Platform with &Cube : Thyme

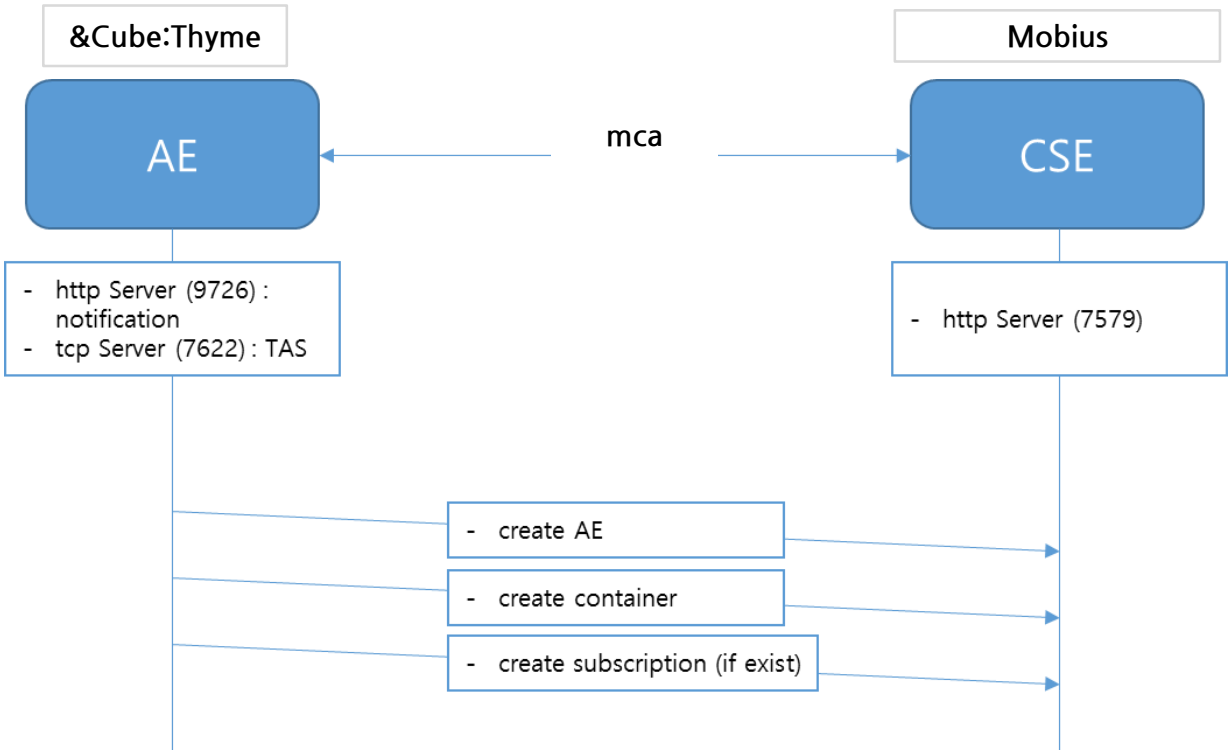
■ Reference development guide of &Cube : Thyme in OCEAN

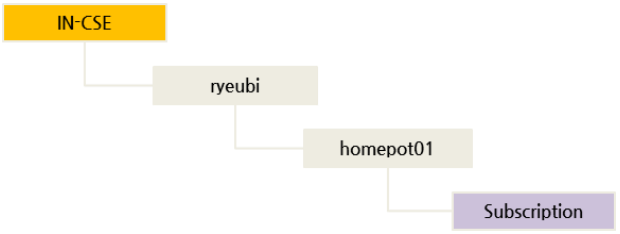
- Install Node.js in device and Download &Cube:Thyme source files into device from OCEAN
- Install modules needed with 'npm install' at prompt at device then Configure conf.xml
- Run &Cube:Thyme through 'node app.js'

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<m2m:conf xmlns:m2m="http://www.onem2m.org/xml/protocols" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <cse>
        <cbhost>localhost</cbhost>
        <cbport>7579</cbport>
        <cbname>mobius-yt</cbname>
    </cse>
    <ae>
        <appid>0.2.481.1.0001.001.75797579</appid>
        <appname>ryeubi</appname>
        <appport>9726</appport> <!-- for notification through http -->
        <appprotocol>json</appprotocol>
        <tasport>7622</tasport>
    </ae>
    <cnt>
        <ctname>homepot01</ctname>
    </cnt>
    <cnt>
        <ctname>cnt_2</ctname>
        <subname>sub_1</subname>
        <nu>mqtt://AUTOSET</nu>
    </cnt>
</m2m:conf>
```

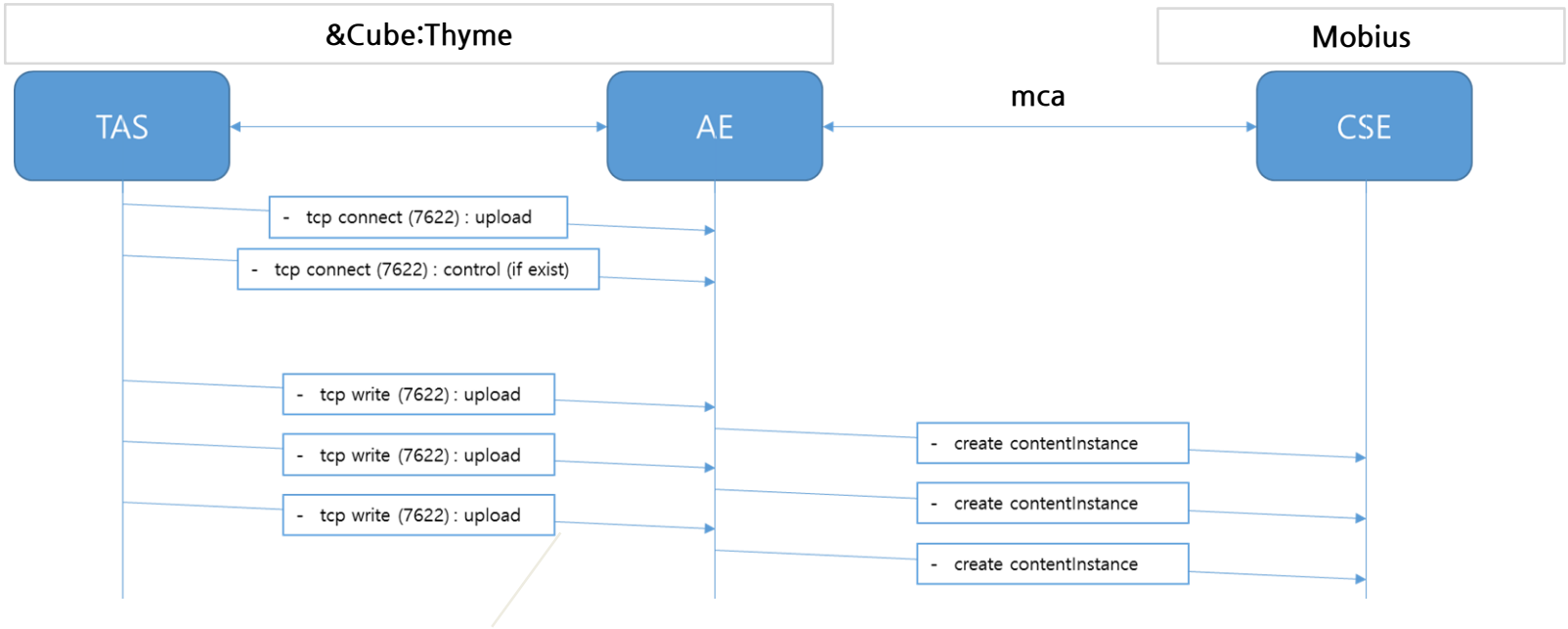


AE-CSE (&Cube-Mobius) MSC Example





TAS (Thing Adaptation Software) Example



{“ctname”:”<Target container name>”, “con”:”<data>”}

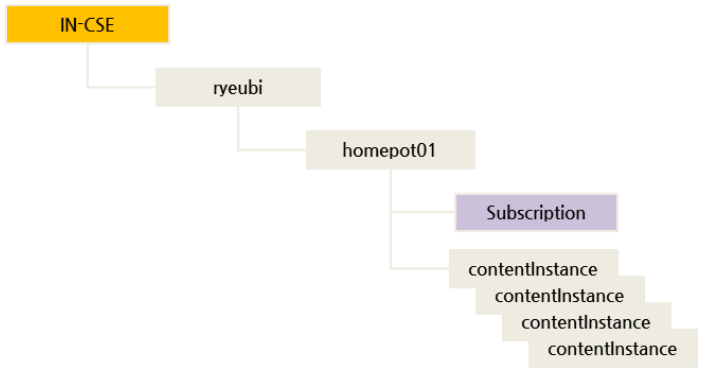
예)

{“ctname”:”cnt_1”, “con”:”TAS2015”}

{“ctname”:”temp”, “con”:”29.8”}

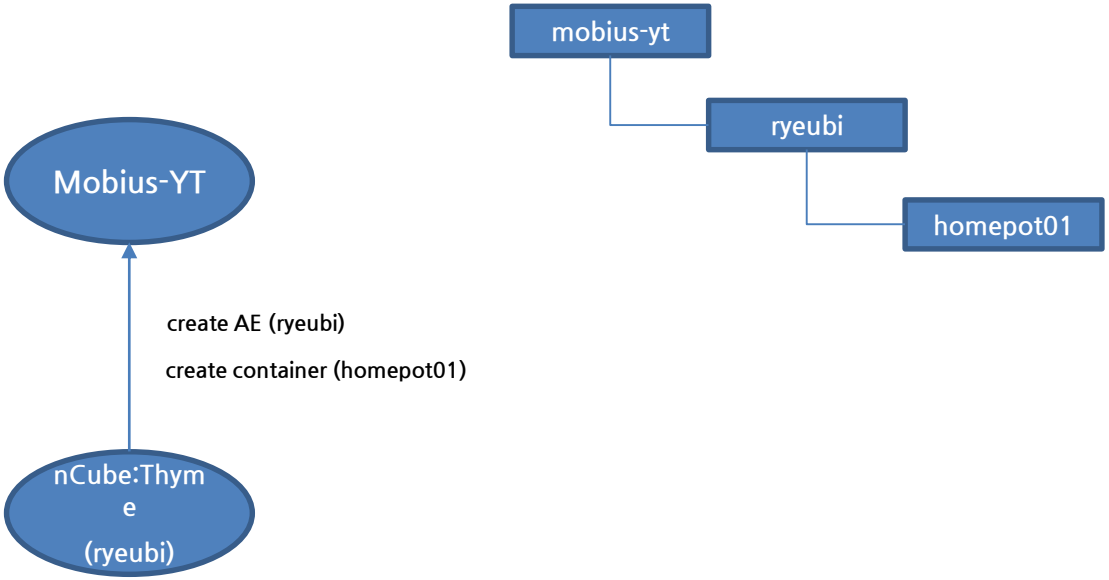
{“ctname”:”temphumi”, “con”:”29.8, 47”}

{“ctname”:”4491”, “con”:”29.8, 47”}

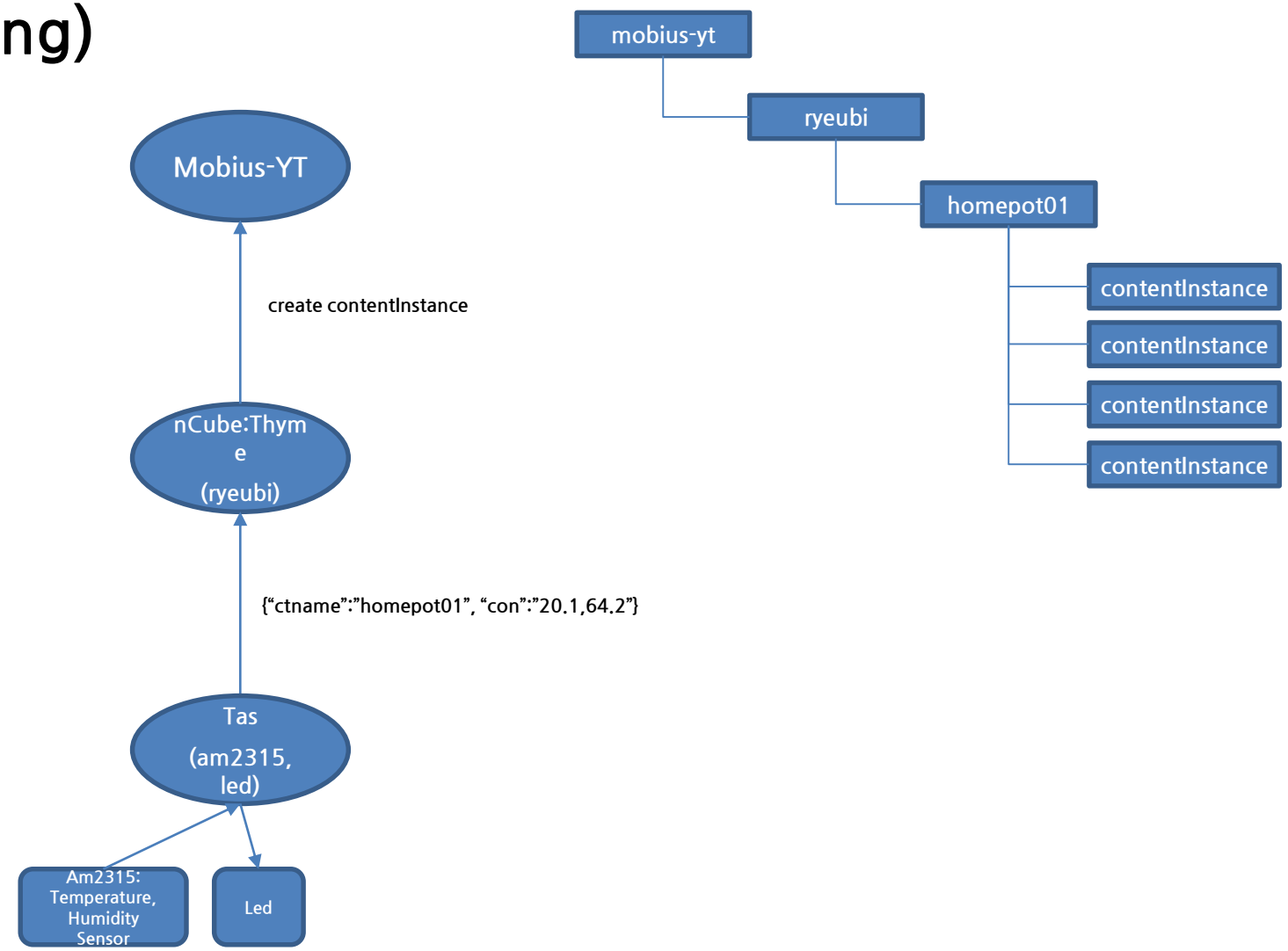


ToT (Tweet of Thing)

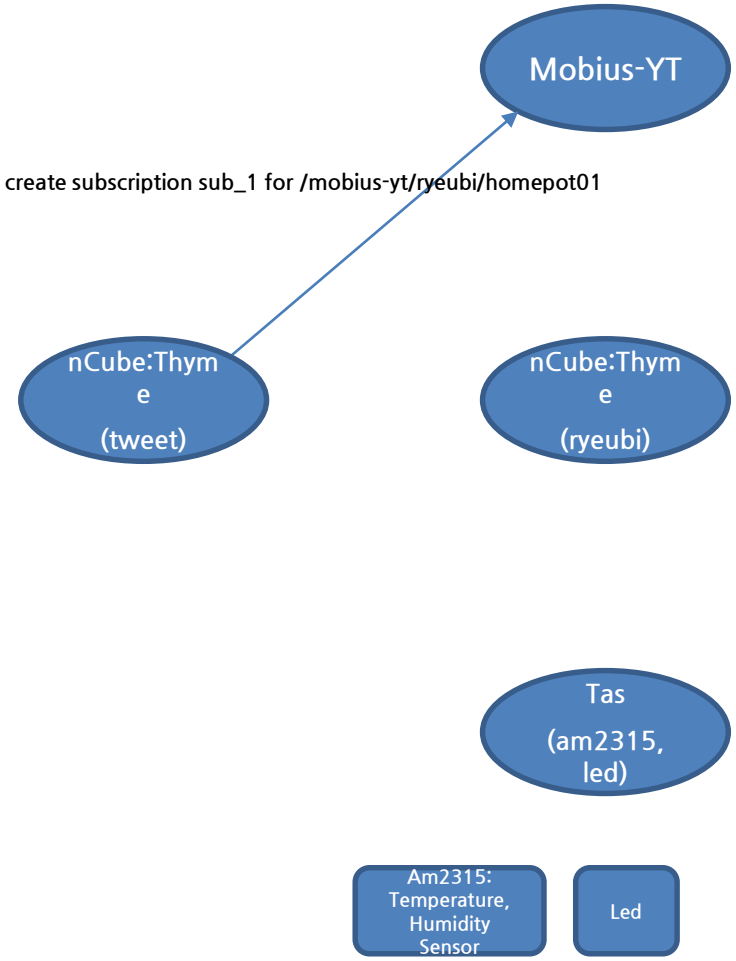
ToT (Tweet of Thing)

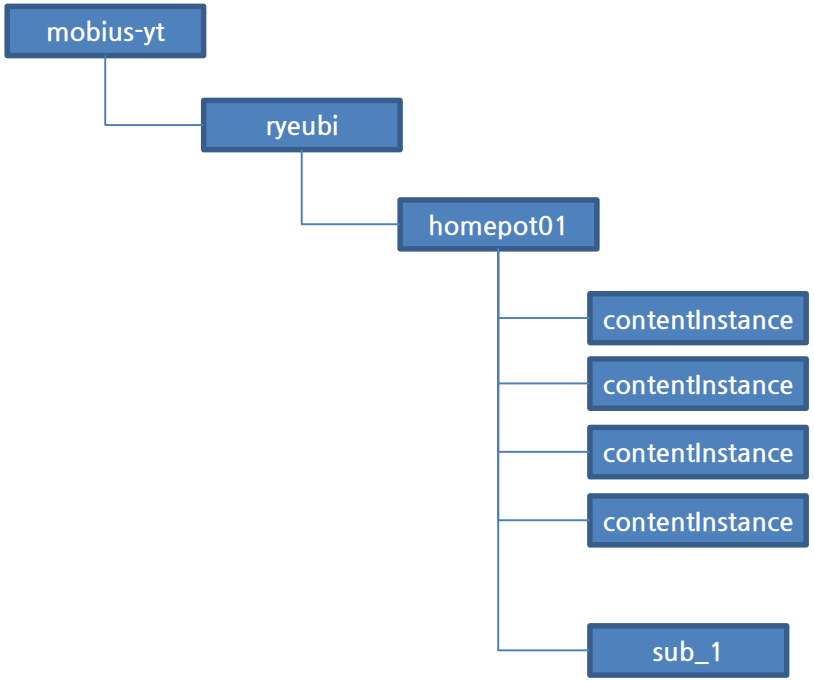


ToT (Tweet of Thing)

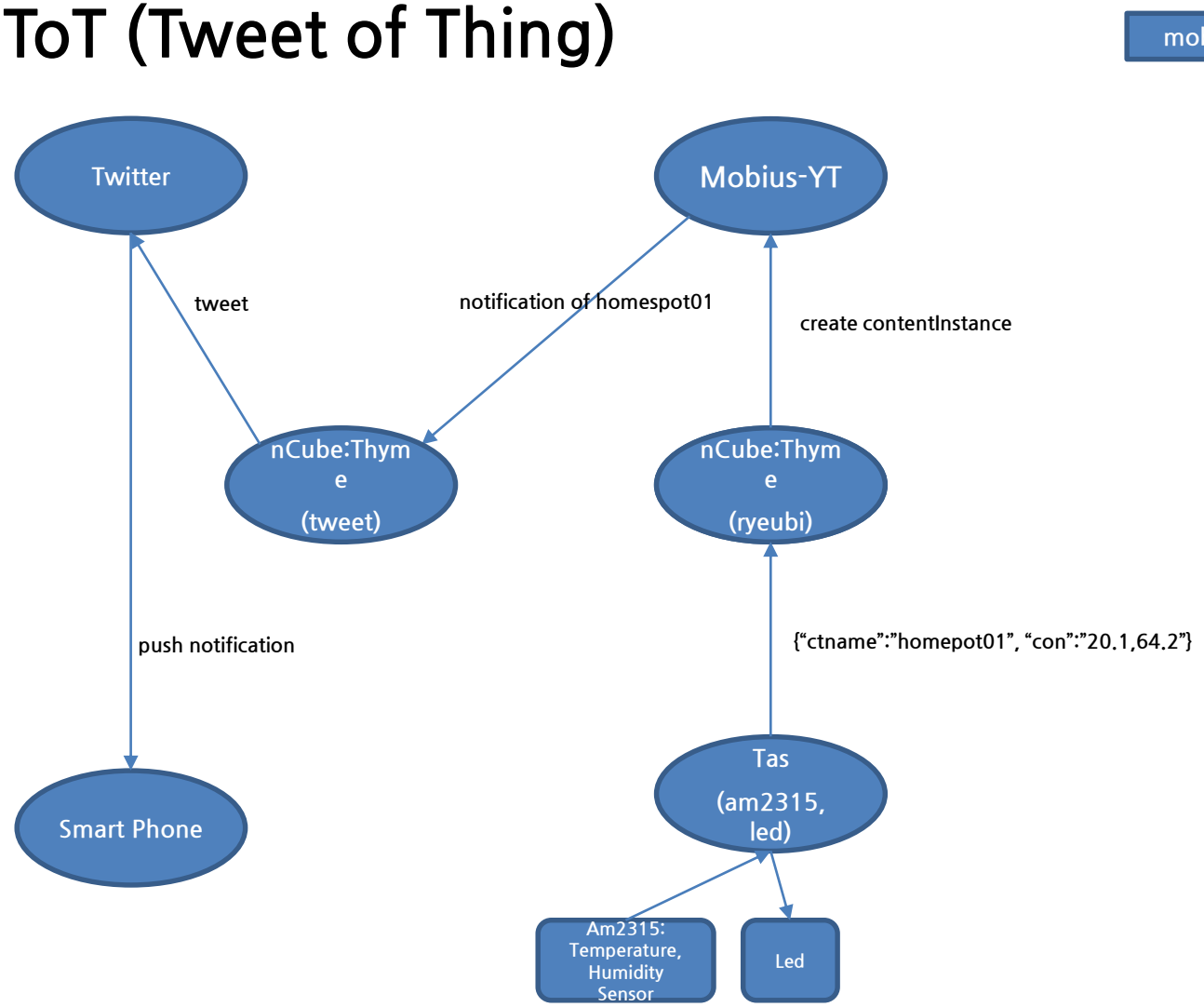


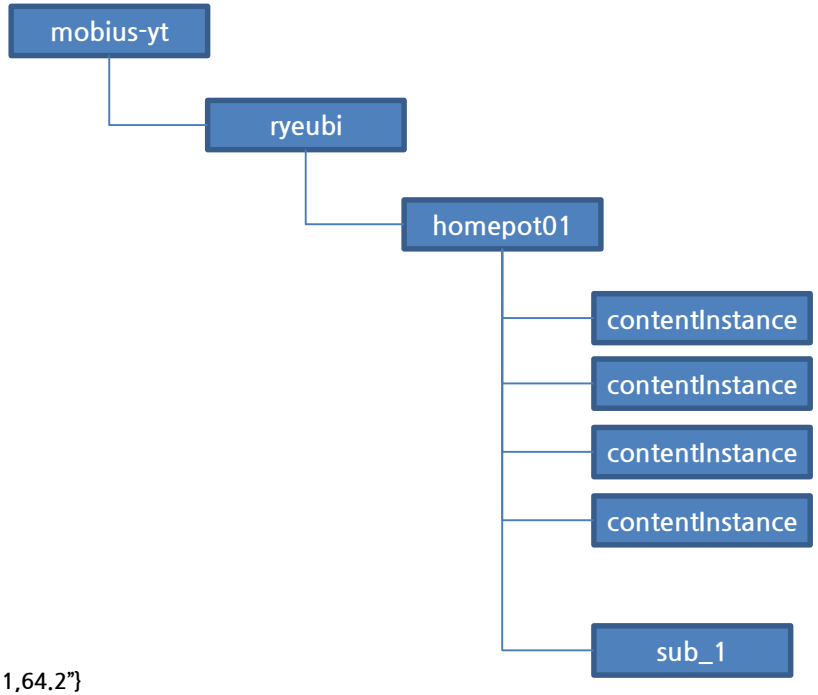
ToT (Tweet of Thing)

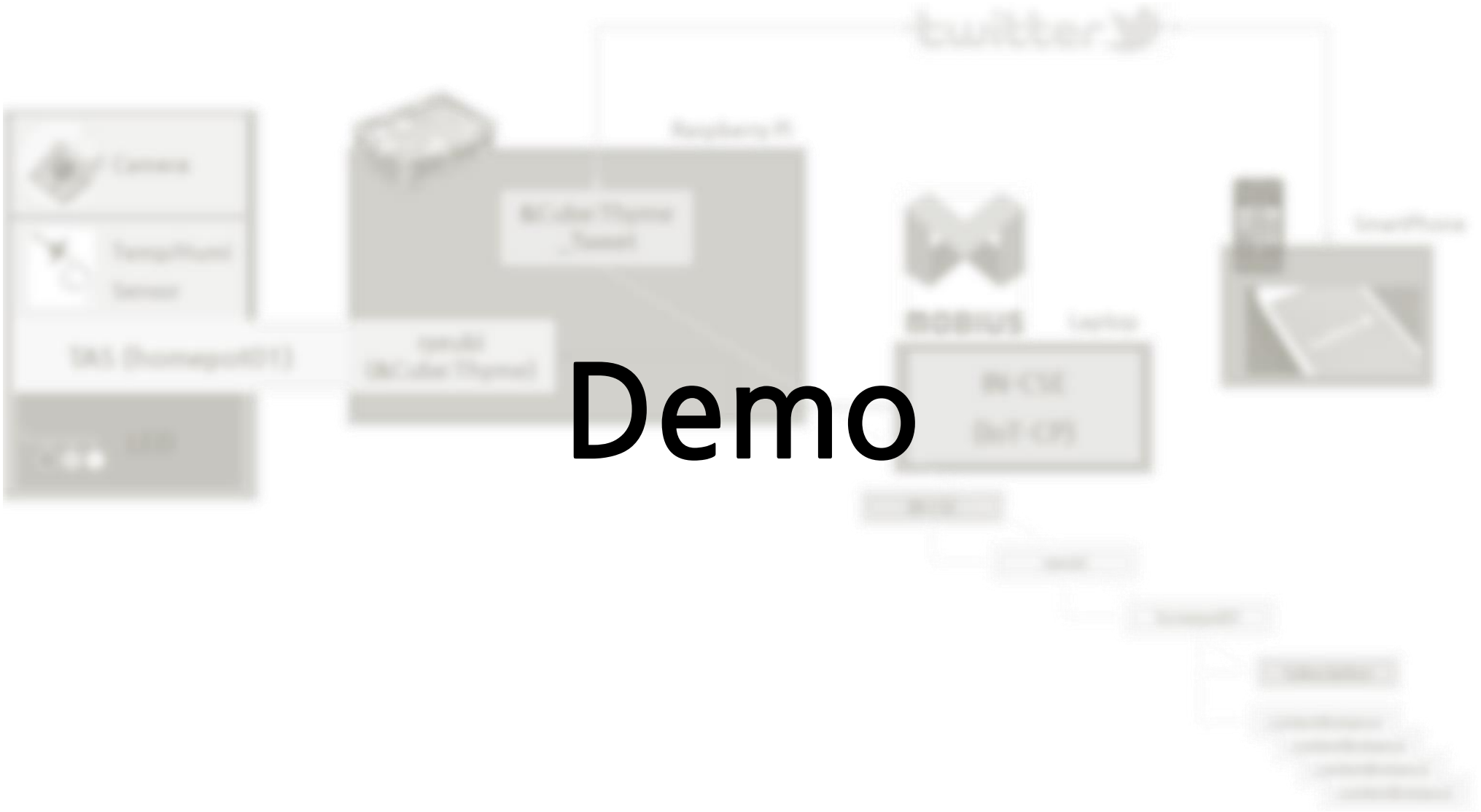




ToT (Tweet of Thing)



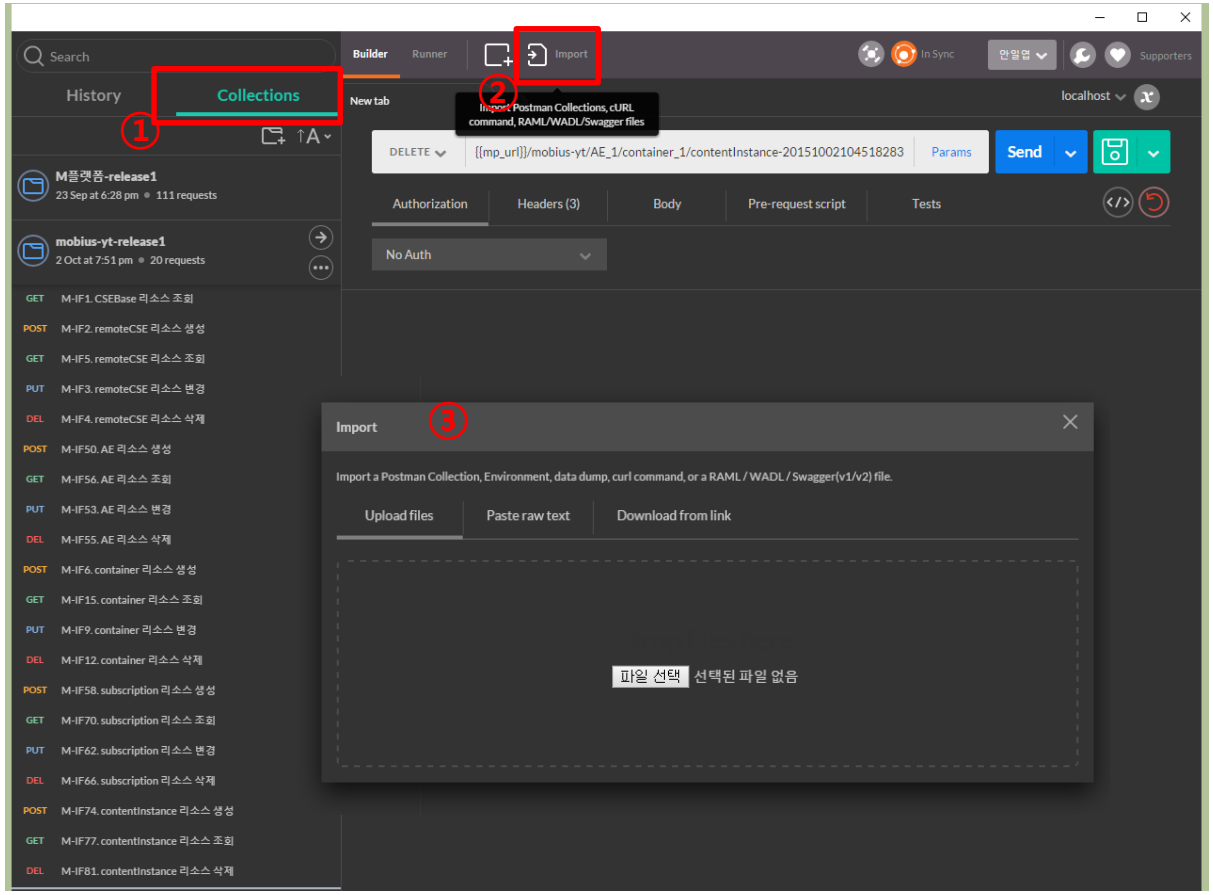



twitter
Raspberry Pi
Camera
Temp/Humi Sensor
SAS [homepot01]
LED
ROBIUS
Laptop
SmartPhone

Demo

Test for Server Platform constructed though Postman

- Test Server Platform working well
  - Download script file of postman from OCEAN



Conclusion - OCEAN (Open alliance for iot stANdard) - http://iotocean.org

Find,
Connect,
Control

Dream,
Do,
Realize

OCEAN

To Be Continue...

감 사 합 니 다.

안 일 엽 , iyahn@keti.re.kr

성 낙 명 , nmsung@keti.re.kr