

OM2M 기반의 OHP-M2M 오픈소스 설치 가이드

2015년 8월

경북대학교 통신프로토콜연구실

최예찬, 강형우

zns038053@gmail.com, hwkang0621@gmail.com

요 약

사물인터넷 (Internet of Things: IoT)이 이슈가 되면서 다양한 사물인터넷 기반의 오픈소스가 공개가 되고 있다. 본 문서에서는 그 중 Eclipse에서 공개한 OM2M 플랫폼을 기반으로 개발된 헬스케어 플랫폼인 OHP-M2M (Open Healthcare Platform based on oneM2M and ISO/IEEE 11073 PHD) 플랫폼에 대한 설치 및 실행방법에 대하여 설명한다.

목 차

1. 서론	2
2. OHP-M2M이란?	2
3. OHP-M2M 설치.....	3
3.1 UHEALTHSERVER 설치	4
3.2 M2MSERVER 설치	6
4. OHP-M2M 실행 방법	10
5. 결론	14
참고 문헌	15

1. 서론

최근 사물인터넷에 대한 관심이 증가하면서 그에 대한 내용을 쉽게 접할 수 있다. 스마트 시트, 스마트 홈, 스마트 카 등의 다양한 분야에서 사물인터넷에 대한 개념을 도입하여 프로젝트를 진행하고 있다. 이 중 사물인터넷 개념을 활용한 헬스케어 분야 역시 많은 연구가 진행 중이다. 우리나라에서의 헬스케어 분야는 아직까지 의료법 및 의료기기 규제로 인하여 많은 연구가 진행되지 못하고 있었지만, 사물인터넷의 등장과 함께 국가적인 차원에서 사물인터넷 기반의 헬스케어 서비스를 제공하기 위해 의료법 및 개인정보보호법 등을 개정을 추진하는 등의 지원을 아끼지 않고 있다.

본 문서에서는 ETSI의 사물인터넷 표준인 SmartM2M 기반으로 개발된 OM2M 플랫폼을 활용하고 헬스케어 국제표준인 ISO/IEEE 11073 표준을 접목하여 개발된 OHP-M2M에 대하여 소개하고, 설치 및 실행방법에 대하여 설명한다.

2. OHP-M2M이란?

OHP-M2M 플랫폼은 사물인터넷 기반 헬스케어 서비스를 제공하기 위해 개발 중인 플랫폼으로 프랑스 대표 연구기관인 LAAS/CNRS에서 시작한 프로젝트를 시작으로 현재 Eclipse에서 개발 중인 OM2M 플랫폼을 기반으로 만들어졌다. 또한 헬스케어 서비스를 지원하기 위해 헬스케어 국제표준인 ISO/IEEE 11073을 지원한다.

OHP-M2M 플랫폼은 산소포화도, 혈압 등의 다양한 헬스케어 디바이스 지원을 위한 헬스케어 디바이스 표준인 ISO/IEEE 11073-104xx 표준을 지원한다. 또한 IEEE 11073 DIM (Domain Information Model) 표준을 XML 형태로 구현하여 지원하며, IEEE 11073 SM (Service Model)은 센서 데이터 전송을 위해 개발된 CoAP (Constrained Application Protocol) 기반으로 개발되었다.

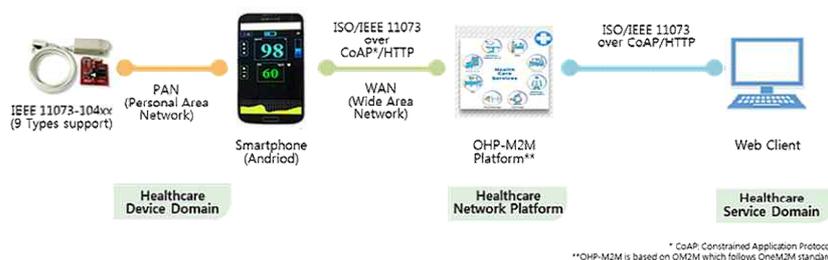


Figure 1. OHP-M2M project의 전체적인 구조

CONTROL

- [Home](#)
- [Current Events](#)
- [Recent Changes](#)
- [Discussion Forum](#)
- [Help](#)

Welcome to Open Healthcare Platform(OHP-M2M)

Open Healthcare Platform based on oneM2M and ISO/IEEE 11073 PHD (OHP-M2M) project, initiated by Advanced Information Network (AIN) Laboratory in Kyungpook National University, Korea, is an open source implementation of the healthcare M2M platform which is integrated with OneM2M and ISO/IEEE 11073 PHD standards. It provides a horizontal M2M healthcare service platform for developing services independently of the underlying network, with the aim to facilitate the deployment of vertical applications and heterogeneous devices in healthcare application.

Number of visitor: 2,183

Contents

- [1 Introduction](#)
- [2 Getting started](#)
- [3 Prerequisites](#)
- [4 Interesting links](#)
- [5 API specification](#)

Location of visitor

78 Visitors
19 Jun 2015 - 14 Aug 2015



Introduction



Various healthcare services have published around world. We developed the OHP-M2M platform based on ISO/IEEE 11073 PHD and oneM2M standards. You can make a platform based on international IoT healthcare standards by utilizing our OHP-M2M platform easily. The project provides a reliable way of

Figure 2. OHP-M2M project 홈페이지

3. OHP-M2M 설치

프로젝트를 진행하기 위해서는 먼저 GitHub에서 OHP-M2M와 UHealthServer를 다운받아야 한다. 아래의 경로를 통해 접속하고 Download ZIP 버튼을 눌러 다운 받는다.

- <https://github.com/AINLAB/OHP-M2M>

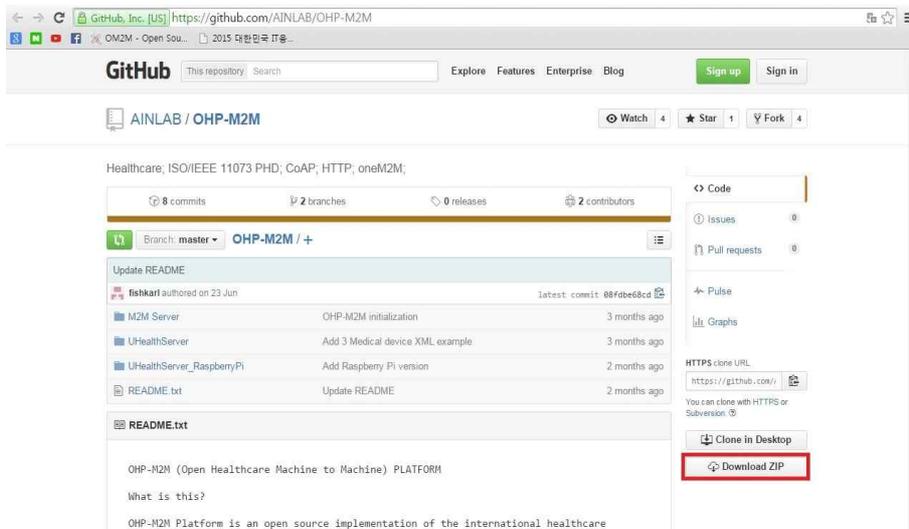


Figure 3. OHP-M2M GitHub 화면

M2M Server는 OM2M의 OSGi bundle로 구현된 플랫폼 서버이고, UHealthServer는 Android Application, UHealthServer_RaspberryPi는 UhealthServer의 Raspberry Pi 버전이다. 여기서는 안드로이드 버전의 UHealthServer만 다루도록 하겠다.

3.1 UHealthServer 설치

- a. Eclipse를 실행시켜 file 메뉴의 Import → Existing Android Code Into Workspace를 선택하여 프로젝트를 Import한다.

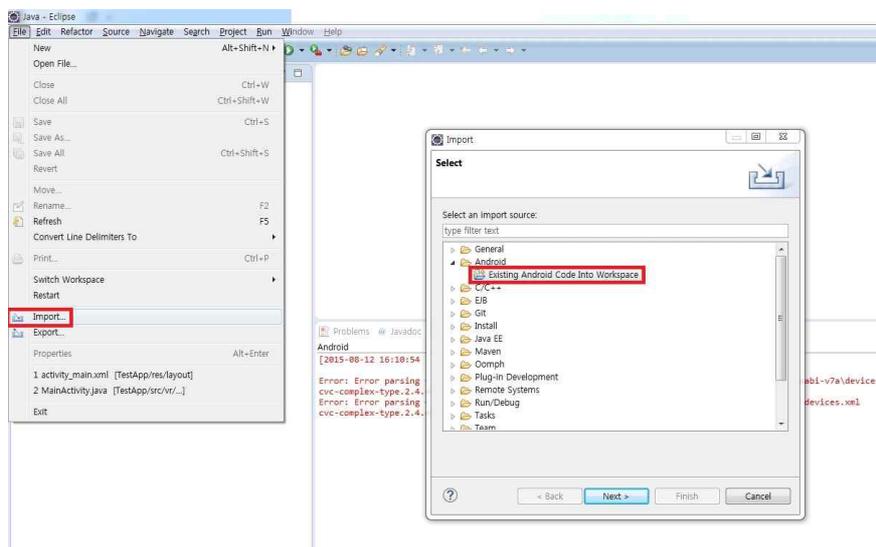


Figure 4. 안드로이드 프로젝트 Import (1)

- b. 위에서 다운받은 UHealthServer 프로젝트의 경로를 선택한 후 Finish 버튼을 누른다.

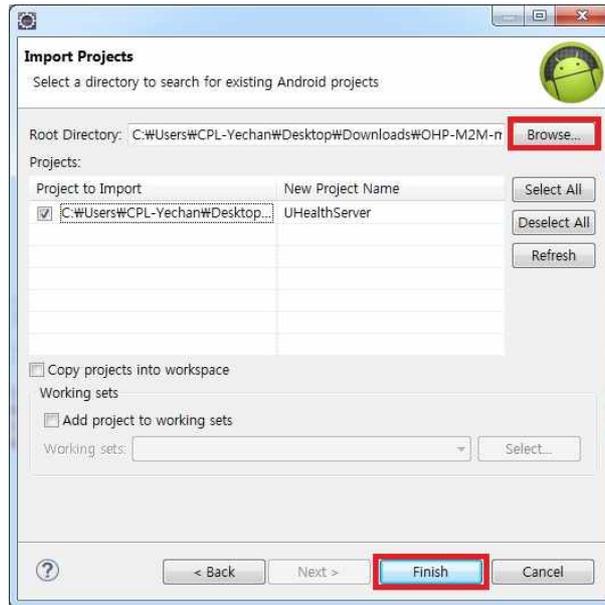


Figure 5. 안드로이드 프로젝트 Import (2)

- c. UHealthServer를 설치할 스마트폰을 연결한다. 프로젝트를 오른쪽 클릭하여 [Run as] → [1. Android Application]을 선택하여 스마트폰에 설치한다.

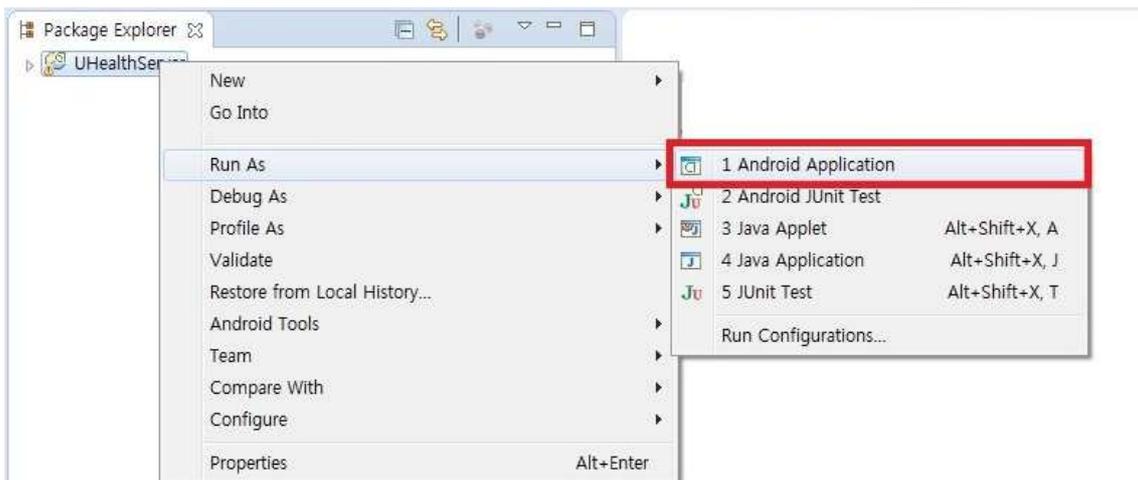


Figure 6. UHealthServer 설치

- d. 설치된 UHealthServer를 확인한다. 화면 왼쪽 상단에 서버 IP주소를 확인할 수 있다.

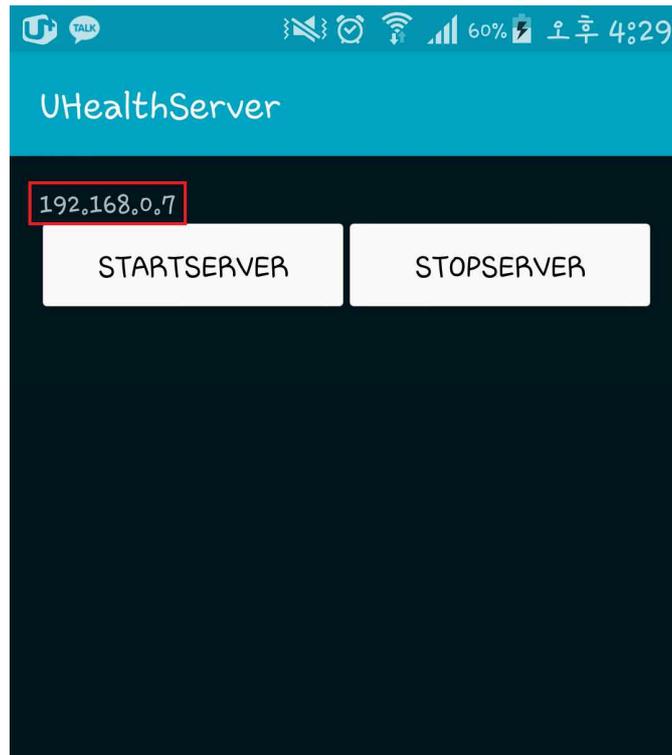


Figure 7. UHealthserver 실행

3.2 M2MServer 설치

먼저 OM2M이 설치해야 한다. OM2M 설치법은 OM2M 공식 홈페이지인 www.eclipse.org/OM2M 홈페이지 또는 저자가 속한 연구실 홈페이지인 protocol.knu.ac.kr 에서 찾을 수 있는 Technical Report (OM2M 오픈 소스 설치 가이드 문서)를 통해서 확인할 수 있다. 그리고 다운받은 M2M Server는 OM2M이 설치된 폴더 안에 있어야 한다. M2MServer 설치 순서는 다음과 같다.

- a. Eclipse를 실행시켜 file 메뉴의 Import → Existing Maven Projects로 프로젝트를 Import한다.

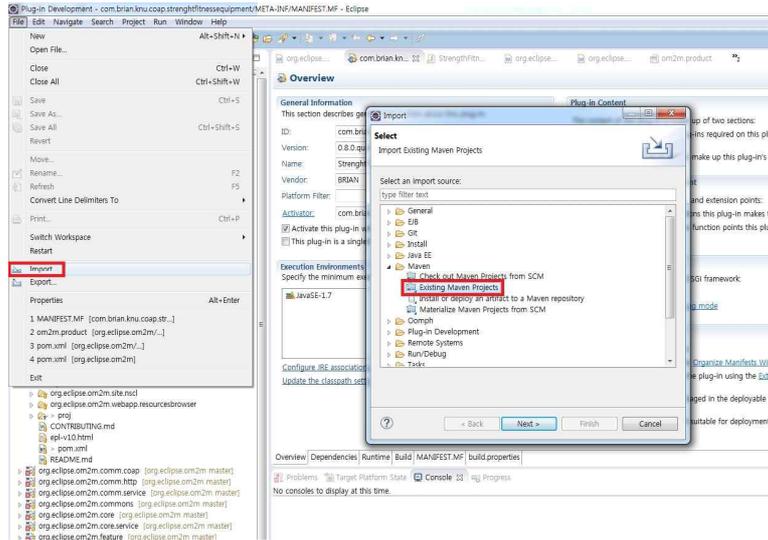


Figure 8. Maven 프로젝트 Import (1)

- b. OM2M폴더 내 M2M Server의 com.brian.knu.coap.strenghtfinessequipment 프로젝트를 선택하고 Finish버튼을 누른다.

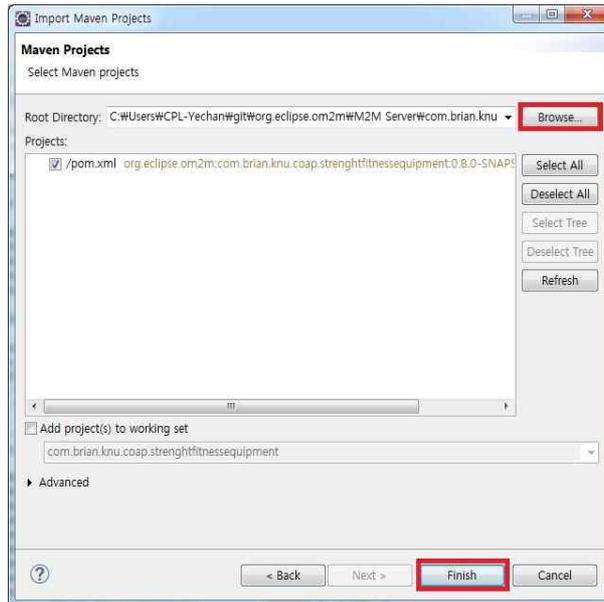


Figure 9. Maven 프로젝트 Import (2)

c. org.eclipse.om2m 내의 pom.xml파일을 선택하고 Modules에서 Add버튼을 누른다.

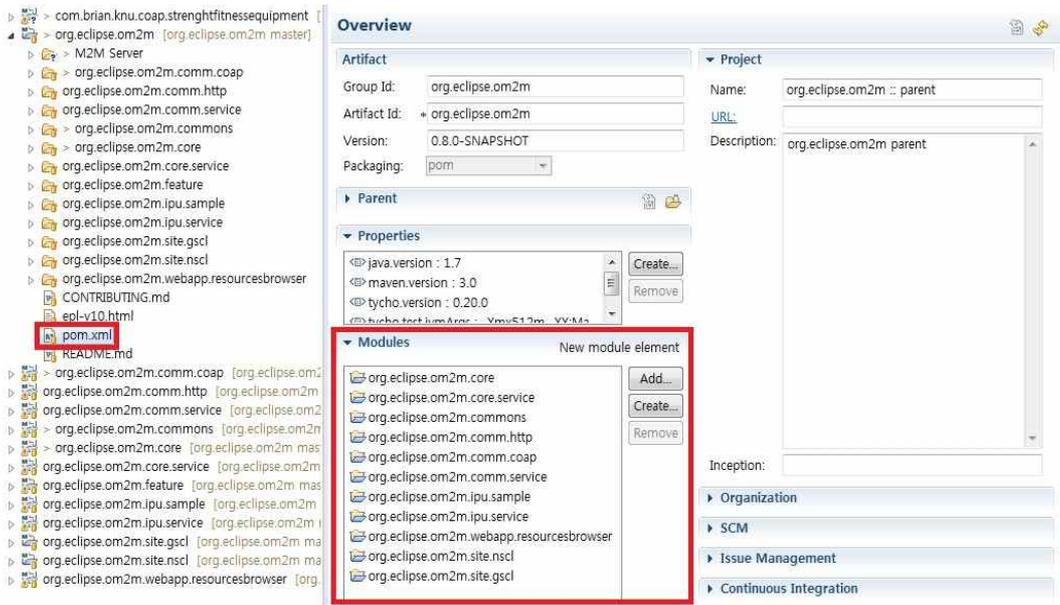


Figure 10. pom.xml 파일 수정 (1)

d. com.brian.knu.coap.strenghtfinessequipment를 선택하고 OK 버튼을 누른다.

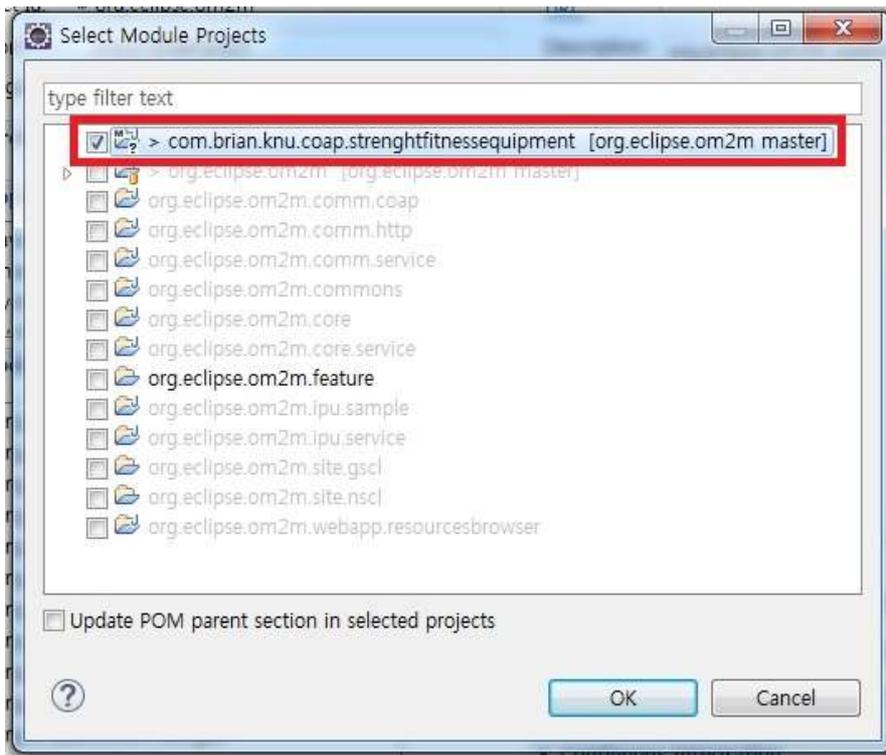


Figure 11. pom.xml 파일 수정 (2)

- e. org.eclipse.om2m.site.gscl 내의 om2m.product를 선택하고 dependencies 페이지에서 Add버튼을 눌러 com.brian.knu.coap.strenghtfinessequipment를 추가한다. (com.brian.knu.coap.strenghtfinessequipment의 META-INF 폴더 안에 있는 MANIFEST.MF 파일을 열어 해당 버전을 확인하고 dependencies의 버전을 수정해 준다.)

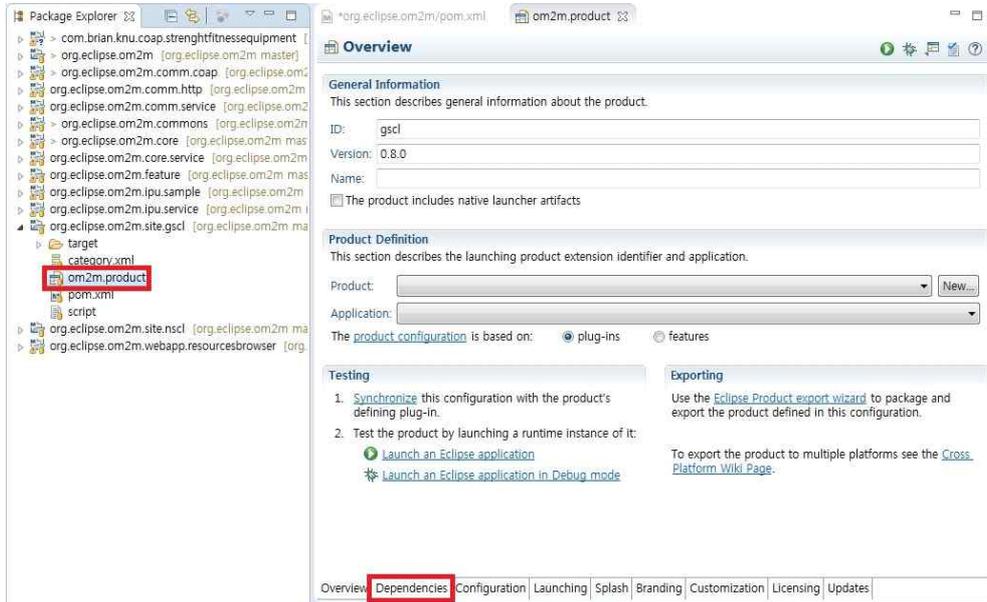


Figure 12. OM2M product Dependencies

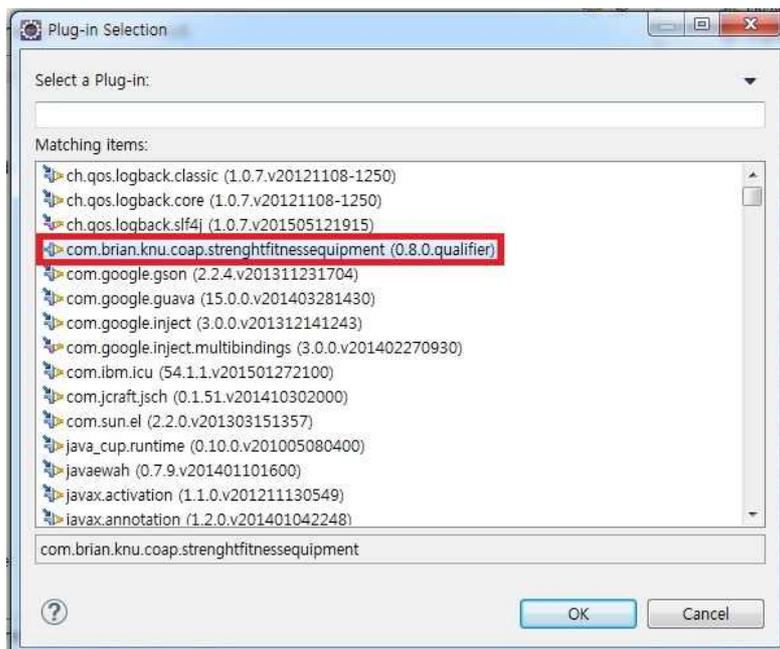


Figure 13. Plug-In 선택

4. OHP-M2M 실행 방법

- a. com.brian.knu.coap.strenghtfitnessequipment 프로젝트를 오른쪽 클릭하여 Maven → Update Project에 들어가서 해당 프로젝트를 선택 후 업데이트한다.

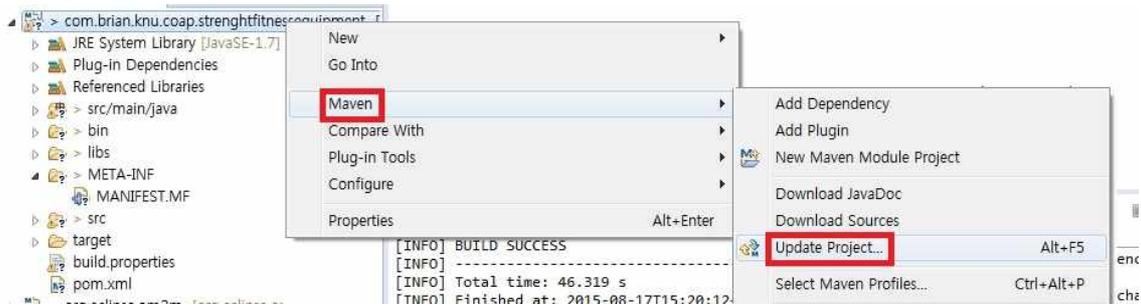


Figure 14. Maven Project 업데이트

- b. org.eclipse.om2m 프로젝트를 오른쪽 클릭하여 Run As -> Maven install을 누른다. 그리고 install이 제대로 되었는지 확인한다.

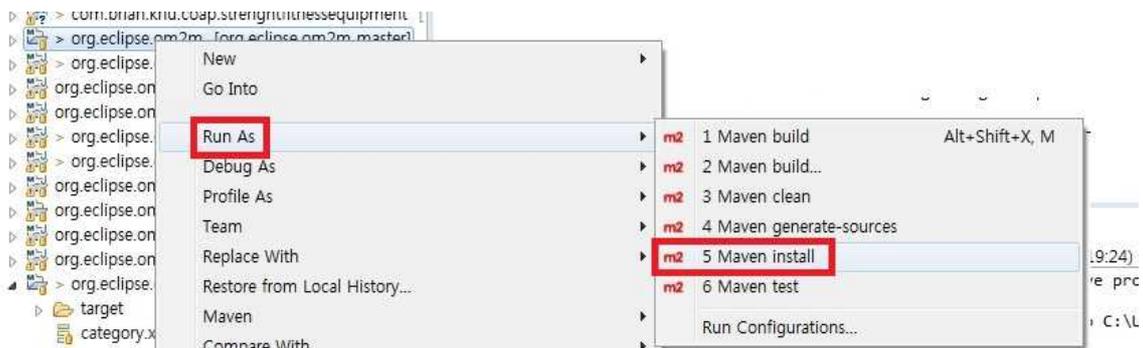


Figure 15. Maven Install

```

[INFO] -----
[INFO] Reactor Summary:
[INFO] org.eclipse.om2m :: parent ..... SUCCESS [ 1.182 s]
[INFO] org.eclipse.om2m :: commons ..... SUCCESS [ 2.917 s]
[INFO] org.eclipse.om2m :: comm service ..... SUCCESS [ 0.210 s]
[INFO] org.eclipse.om2m :: ipu service ..... SUCCESS [ 0.187 s]
[INFO] org.eclipse.om2m :: core service ..... SUCCESS [ 0.184 s]
[INFO] org.eclipse.om2m :: core ..... SUCCESS [ 1.782 s]
[INFO] org.eclipse.om2m :: comm http ..... SUCCESS [ 0.286 s]
[INFO] org.eclipse.om2m :: comm coap ..... SUCCESS [ 0.422 s]
[INFO] org.eclipse.om2m :: ipu sample ..... SUCCESS [ 0.651 s]
[INFO] org.eclipse.om2m :: webapp resourcesbrowser ..... SUCCESS [ 0.549 s]
[INFO] org.eclipse.om2m :: nscl product ..... SUCCESS [ 9.304 s]
[INFO] com.brian.knu.coap.strenghtfitnessequipment ..... SUCCESS [ 0.361 s]
[INFO] org.eclipse.om2m :: gscl product ..... SUCCESS [ 6.799 s]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 46.319 s
[INFO] Finished at: 2015-08-17T15:20:12+09:00
[INFO] Final Memory: 86M/633M
[INFO] -----

```

Figure 16. Maven Install 결과

- c. 안드로이드 UHealthServer를 실행하여 IP주소를 확인한다. 그리고 com.brian.knu.coap.strenghtfitnessequipment / StrengthFitnessMonitor.java파일을 열어서 확인한 IP주소를 Figure 17와 같이 넣어준다.

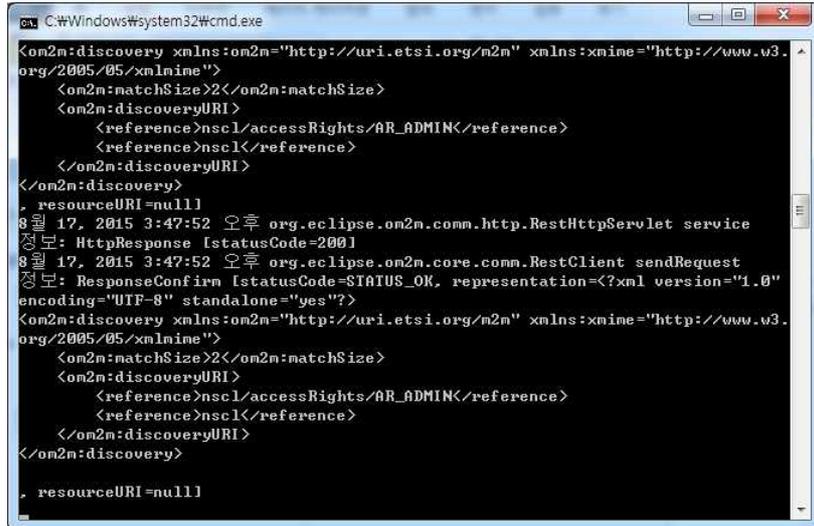
```

222     devConfigurationID.addSimple(sm);
223
224     cvDIM.addCompound(devConfigurationID);
225     break;
226     case 2650:
227         Compound systemTypeSpeclst = new SystemTypeSpeclst(deviceName);
228
229         makeCompound(systemTypeSpeclst, ele);
230         cvDIM.addCompound(systemTypeSpeclst);
231         break;
232     default:
233         break;
234     }
235 }
236
237 };
238
239 public void listenToCV() {
240     new Thread() {
241         public void observerStrengthFitness(String uri) {
242             CoapClient client = new CoapClient(uri + "/strenghtfitnessmonitor"); // This creates a coap client to
243                                     //listen to strenghtfitness
244
245             CoapObserveRelation relation1 = client
246                 .observe(myStrenghtFitnessHandler);
247         }
248     }
249 }
250
251 @Override
252 public void run() {
253     // TODO Auto-generated method stub
254
255     observerStrengthFitness("coap://192.168.0.7:5683");
256
257 }
258
259 }.start();
260 }
261 }
262 }
263 }

```

Figure 17. IP주소 및 Port번호 입력 및 확인

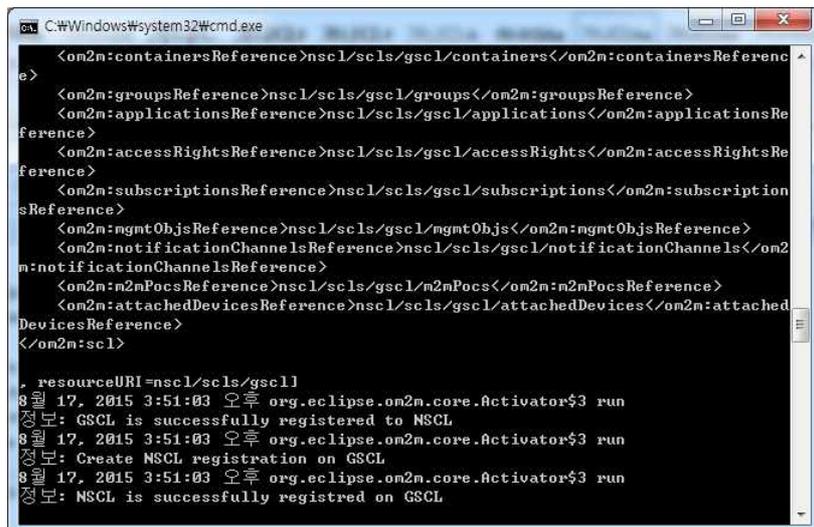
- d. C:\Users\UserName\git\org.eclipse.om2m\org.eclipse.om2m.site.nscf\target\products\nscf\win32\win32\x86_64 경로에 있는 NSCL 실행파일인 start.bat 파일을 실행한다.



```
C:\Windows\system32\cmd.exe
<om2m:discovery xmlns:om2m="http://uri.etsi.org/m2m" xmlns:xmime="http://www.w3.org/2005/05/xmime">
  <om2m:matchSize>2</om2m:matchSize>
  <om2m:discoveryURI>
    <reference>nscf/accessRights/AR_ADMIN</reference>
    <reference>nscf</reference>
  </om2m:discoveryURI>
</om2m:discovery>
, resourceURI=null
8월 17, 2015 3:47:52 오후 org.eclipse.om2m.core.http.RestHttpServlet service
정보: HttpResponse [statusCode=200]
8월 17, 2015 3:47:52 오후 org.eclipse.om2m.core.com.RestClient sendRequest
정보: ResponseConfirm [statusCode=STATUS_OK, representation=?xml version="1.0" encoding="UTF-8" standalone="yes"?]
<om2m:discovery xmlns:om2m="http://uri.etsi.org/m2m" xmlns:xmime="http://www.w3.org/2005/05/xmime">
  <om2m:matchSize>2</om2m:matchSize>
  <om2m:discoveryURI>
    <reference>nscf/accessRights/AR_ADMIN</reference>
    <reference>nscf</reference>
  </om2m:discoveryURI>
</om2m:discovery>
, resourceURI=null
```

Figure 18. NSCL 실행화면

- e. C:\Users\UserName\git\org.eclipse.om2m\org.eclipse.om2m.site.gscl\target\products\gscl\win32\win32\x86_64 경로에 있는 GSCL 실행파일인 start.bat 파일을 실행한다.



```
C:\Windows\system32\cmd.exe
<om2m:containersReference>nscf/scs/gscf/containers</om2m:containersReference>
<om2m:groupsReference>nscf/scs/gscf/groups</om2m:groupsReference>
<om2m:applicationsReference>nscf/scs/gscf/applications</om2m:applicationsReference>
<om2m:accessRightsReference>nscf/scs/gscf/accessRights</om2m:accessRightsReference>
<om2m:subscriptionsReference>nscf/scs/gscf/subscriptions</om2m:subscriptionsReference>
<om2m:mgmtObjsReference>nscf/scs/gscf/mgmtObjs</om2m:mgmtObjsReference>
<om2m:notificationChannelsReference>nscf/scs/gscf/notificationChannels</om2m:notificationChannelsReference>
<om2m:m2mPocsReference>nscf/scs/gscf/m2mPocs</om2m:m2mPocsReference>
<om2m:attachedDevicesReference>nscf/scs/gscf/attachedDevices</om2m:attachedDevicesReference>
</om2m:scf>
, resourceURI=nscf/scs/gscf
8월 17, 2015 3:51:03 오후 org.eclipse.om2m.core.Activator$3 run
정보: GSCL is successfully registered to NSCL
8월 17, 2015 3:51:03 오후 org.eclipse.om2m.core.Activator$3 run
정보: Create NSCL registration on GSCL
8월 17, 2015 3:51:03 오후 org.eclipse.om2m.core.Activator$3 run
정보: NSCL is successfully registered on GSCL
```

Figure 19. GSCL 실행화면

- f. GSCL에서 'SS'를 입력하면 Figure 20과 같이 리스트가 생성된다. 여기서 start 명령어와 함께 설치된 bundle을 실행시킬 수 있다. 실행하려는 bundle의 번호를 Start 번호와 같이 입력하여 실행한다. 예를 들면 start 28과 같다.

```

C:\Windows\system32\cmd.exe
osgi> SS
"framework is launched."

id      State      Bundle
0       ACTIVE    org.eclipse.osgi_3.10.2.v20150203-1939
1       STARTING  com.brian.knu.coap.strenghtfitnessequipment_0.8.0.20150817-0619
2       RESOLVED  javax.servlet_3.1.0.v20140303-1611
3       RESOLVED  org.apache.commons.codec_1.6.0.v201305230611
4       RESOLVED  org.apache.commons.httpclient_3.1.0.v201012070820
5       RESOLVED  org.apache.commons.logging_1.1.1.v201101211721
6       ACTIVE    org.apache.felix.gogo.command_0.10.0.v201209301215
7       ACTIVE    org.apache.felix.gogo.runtime_0.10.0.v201209301036
8       ACTIVE    org.apache.felix.gogo.shell_0.10.0.v201212101605
9       ACTIVE    org.eclipse.equinox.console_1.1.0.v20140131-1639
10      ACTIVE    org.eclipse.equinox.http.jetty_3.0.200.v20131021-1843
11      ACTIVE    org.eclipse.equinox.http.servlet_1.1.500.v20140318-1755
12      RESOLVED  org.eclipse.equinox.launcher_1.3.0.v20140415-2008
13      RESOLVED  org.eclipse.jetty.continuation_8.1.16.v20140903
14      RESOLVED  org.eclipse.jetty.http_8.1.16.v20140903
15      RESOLVED  org.eclipse.jetty.io_8.1.16.v20140903
16      RESOLVED  org.eclipse.jetty.security_8.1.16.v20140903
17      RESOLVED  org.eclipse.jetty.server_8.1.16.v20140903
18      RESOLVED  org.eclipse.jetty.servlet_8.1.16.v20140903
19      RESOLVED  org.eclipse.jetty.util_8.1.16.v20140903
20      ACTIVE    org.eclipse.om2m.comm.coap_0.8.0.20150817-0619
21      ACTIVE    org.eclipse.om2m.comm.http_0.8.0.20150817-0619
22      RESOLVED  org.eclipse.om2m.comm.service_0.8.0.20150817-0619
23      RESOLVED  org.eclipse.om2m.commons_0.8.0.20150817-0619
24      ACTIVE    org.eclipse.om2m.core_0.8.0.20150817-0619
25      RESOLVED  org.eclipse.om2m.core.service_0.8.0.20150817-0619
26      RESOLVED  org.eclipse.om2m.ipu.sample_0.8.0.20150817-0619
27      RESOLVED  org.eclipse.om2m.ipu.service_0.8.0.20150817-0619
28      ACTIVE    org.eclipse.om2m.webapp.resourcesbrowser_0.8.0.20150817-0619
29      RESOLVED  org.eclipse.osgi.services_3.4.0.v20140312-2051
osgi>
  
```

Figure 20. OSGi 콘솔에서의 설치된 번들 상태 출력

- g. 브라우저에 127.0.0.1:8080을 입력한다. OM2M페이지가 나타나면 username과 password에 'admin'을 입력하고 접속한다.



Figure 21. OM2M 웹 인터페이스

h. OM2M에 접속 후 GSCL Resource Tree에서 medical device 및 Resource를 찾을 수 있다.

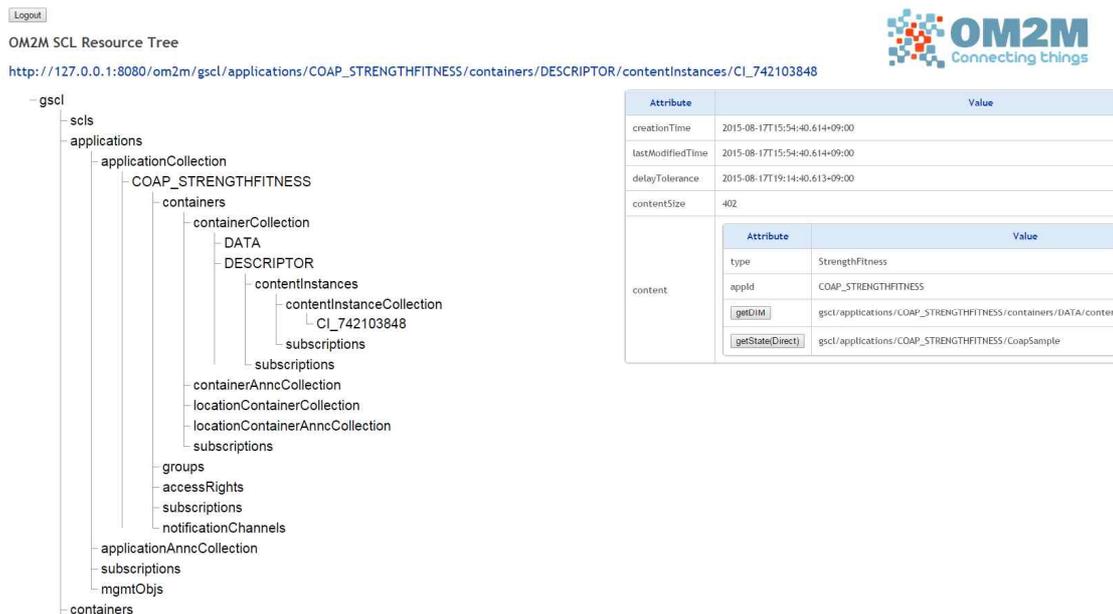


Figure 22. OM2M sclBase Resource

위와 같이, Medical Device에서 측정된 센서 데이터를 사물인터넷 표준에 맞게 Resource Tree를 만들어서 플랫폼에 저장하여 사물인터넷 기반의 헬스케어 서비스 제공을 위한 Web / Application 등을 만들 수 있다.

5. 결론

지금까지 본 고에서는 사물인터넷 표준기반의 OM2M 플랫폼을 활용하고 헬스케어 국제표준인 ISO/IEEE 11073 표준을 기반으로 하는 OHP-M2M의 설치 및 실행과정을 설명하였다. OHP-M2M은 사물인터넷 표준과 헬스케어 표준을 동시에 만족하는 플랫폼으로써, 표준을 만족하는 다른 플랫폼 및 어플리케이션과의 상호운용성 (Interoperability)을 보장한다. 위와 같이, OHP-M2M은 설치 및 실행방법이 간단하고 사물인터넷 및 헬스케어 표준을 만족하는 플랫폼으로써, 추후 사업자들이 사물인터넷 헬스케어 서비스를 개발하고 공개할 때 많은 도움이 될 것으로 기대된다.

참고 문헌

- [1] OM2M project homepage, <http://eclipse.org/OM2M/>
- [2] OHP-M2M project github, <https://github.com/AINLAB/OHP-M2M>
- [3] OM2M 오픈 소스 설치 가이드, protocol.knu.ac.kr/tech/CPL-TR-14-03-OM2M.pdf