



oneM2Mを使用したIoTアプリケーション開発の始め方 —IoTアプリケーション開発者向けチュートリアル イベント開催—

クアルコム ジャパン株式会社 標準化担当部長

うちだ のぶゆき
内田 信行

株式会社 KDDI 総合研究所 スマートセキュリティグループ

おくい のりひろ
奥井 宣広

1. はじめに

昨今、IoT社会の実現に向けた動きが活発化し、2020年の導入を目指す次世代モバイル通信システム(5G)の開発と相まって、様々な新しい価値創造やビジネス展開への取り組みが行われている。こうした流れの中で、アプリケーション/プラットフォーム間のデータ共有・流通・連携を容易に行うことができ、IoTサービス構築に必要な機能をワンセットで規定する唯一のオープンな国際標準であるoneM2Mは、広く業界をまたがる新たなアプリケーション、ビジネスのイノベーション創出の原動力として期待されている。

このoneM2M技術の普及促進の一環として、ARIB^{*1}とTTC^{*2}が共催して、アプリケーション開発者がoneM2M技術を使って迅速かつ簡易に開発をスタートするためのチュートリアルイベントを開催した。本稿では、同イベントの講師でありoneM2M国際標準化活動にも貢献している筆者が、チュートリアルのエッセンスを紹介する。

2. oneM2Mの狙いと特徴

oneM2Mは、ARIB/TTCを含む世界の7つの主要な地域標準化団体を中心となって2012年に設立された(現在はインドTSDSIを加えた8団体)。oneM2Mでは、クロスドメインで相互接続可能なIoTエコシステムの実現を目指しており、サービスレイヤの各機能(CSF: Common Service

Function)に関して、仕様の策定を行っている。また、OCF(Open Connectivity Foundation)やW3C(World Wide Web Consortium)などの他のIoTに関連する標準化団体とも、インターワークやリエゾンにより、異なる規格との相互接続性もサポートし、お互いの強みを活かすことが可能である。

仕様策定の進捗に関しては、最低限の機能にフォーカスしたRelease1仕様書セットが2015年に発行され、実用化向けの機能を多く取り込んだRelease2仕様書セットが2016年に発行されている。現在は、開発側からのフィードバックの取り込みをメインとしてRelease3仕様の策定を行っており、2018年9月頃のリリースを予定している。

このように、oneM2Mに関しては、仕様の議論から開発の議論にシフトしてきており、oneM2M対応を謳うプラットフォーム製品やオープンソースなどが、海外を中心にリリースされている。一方で、日本の現状は、過去のARIB/TTC主催のoneM2Mセミナーにおいて、oneM2Mに興味があると回答したベンダーは多かったが、様々な理由により開発へとシフトしているとは言えない。

本チュートリアルは、oneM2M準拠のアプリケーション開発に興味がある開発者を対象として、必要最低限のoneM2M仕様を押さえ、オープンソースであるOM2M^{*3}とOS-IoT^{*4}を用いて、アプリケーション開発を始めるための一歩を手助けすることを目的としている。

3. アプリケーション開発者向けチュートリアル

3.1 oneM2Mサービスレイヤの概要

クアルコム ジャパン 内田信行(筆者)より、本チュートリアルで解説するデモを開発の際に使用するoneM2Mの仕様についての解説を行った。

oneM2M Release2では、サービスレイヤにおいて、図2に示す12種類のCSFを定義しており、これらのCSFを含むエンティティをCSE(Common Service Entity)として定義



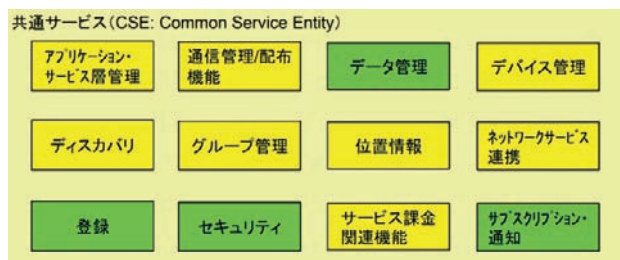
図1. oneM2M準拠OSSと準拠製品を商用化している企業

*1 一般社団法人 電波産業会 (<https://www.arib.or.jp>)

*2 一般社団法人 情報通信技術委員会 (<http://www.ttc.or.jp>)

*3 Eclipse OM2M (<https://www.eclipse.org/om2m/>)

*4 ATIS OS-IoT (<https://os-iot.org>)



■図2. oneM2Mプラットフォームが定義するサービス機能

■表1. デモで使用するリソース

リソース名	説明
〈cseBase〉	CSEのルートディレクトリ
〈accessControlPolicy〉	アクセス権限の設定
〈AE〉	アプリケーションの識別
〈container〉	〈AE〉配下に作成するデータの入れ物
〈contentInstance〉	〈container〉配下に作成するデータそのもの
〈subscription〉	他のエンティティに通知する条件

している。各CSFを実現するため、30以上のリソースが定義され、仕様書の分量も膨大なものとなっており、アプリケーション (AE: Application Entity) 開発においては、仕様書を読み解くことが一つのハードルとなっている。

本セッションでは、oneM2Mの仕組みを理解する上で必要な、プリミティブ、プロトコルバインディング、シリアルライゼーション、リソース構造について、デモの開発に最低限必要な箇所だけをピックアップして解説を行った。

oneM2Mのリソースは、RESTスタイルインタフェースを用いて、URIで一意に識別することができ、各リソースはCRUDN (Create/Retrieve/Update/Delete/Notify) により操作することができる。本チュートリアルで解説するデモでは、図2の緑のCSFに該当する6つのリソースだけで開発を行うことができる (表1)。このように、それほど多くのoneM2Mの知識がなくても、基本的なAE開発が十分可能であることを示した。

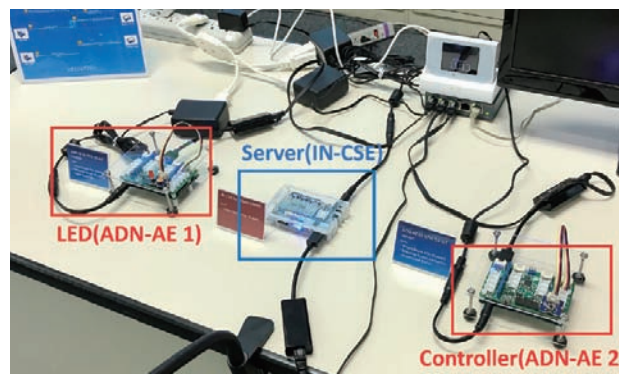
3.2 制御系IoTアプリケーション開発デモ

引き続き、クアルコム ジャパン 内田信行が、リモートによるLED制御をユースケースとした、制御系IoTアプリケーション開発デモについて解説した (写真1)。

デモの構成は写真2のとおりで、Firewallなどの設定を省き開発を容易にするために、LED (ADN-AE 1)、Controller (ADN-AE 2)、Server (IN-CSE) とともにローカルネットワーク内に置かれている。全てのエンティティはシングルボードコン



■写真1. クアルコム ジャパン 内田信行 (oneM2MプロトコルWG副議長)

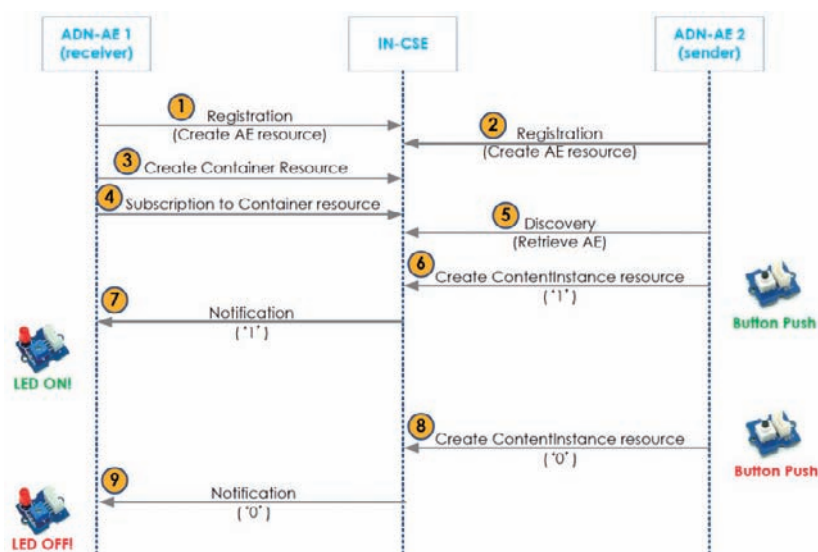


■写真2. 制御系アプリケーションデモ構成の外観

ピュータDragonBoard 410c上に構築されており、LEDとControllerの構築にはOS-IoTを、Serverの構築にはOM2Mを使用している。また、本デモでは、HTTPプロトコルバインディングとXMLによるシリアルライゼーションが用いられている。

本デモのフローは、図3に示すとおりで、各フローの内容は以下のとおりである。

- [手順1、2] LEDとControllerのServerへの登録
- [手順3] LEDの状態を管理するデータを格納するディレクトリの作成
- [手順4] LEDのディレクトリ内に生成されたデータをLEDに転送するための設定情報作成
- [手順5] ControllerがターゲットとするLEDの発見
- [手順6] LEDのディレクトリにLEDのONを示す「1」を書き込む
- [手順7] 手順6で作成された「1」をLEDに転送
- [手順8] LEDのディレクトリにLEDのOFFを示す「0」を書き込む



■図3. 制御系アプリケーションデモのシグナリングフロー



■写真3. KDDI総合研究所 奥井宣広



■写真4. 東京システムハウス 佐藤導吉氏

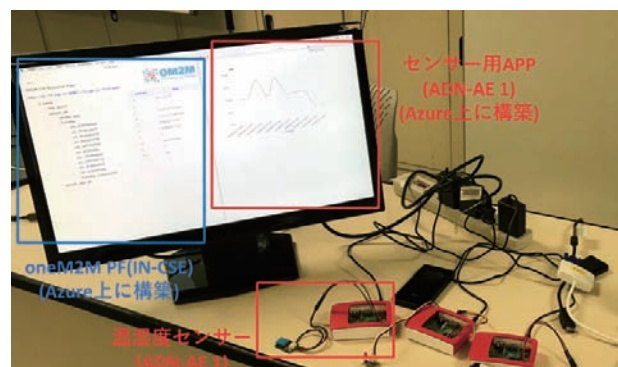
[手順9] 手順8で作成された「0」をLEDに転送

本セッションでは、本デモの構築に必要な手順として、DragonBoard 410cのセットアップ、OM2Mの設定、OS-IoTの使い方を中心に解説が行われた。また、本デモで利用したソースコードはGithub上に公開されており、誰でもダウンロードして試することができる。

<https://github.com/nuchida/arib-ttc-2018-02-02-demo>

3.3 センサー系IoTアプリケーション開発デモ

次に、KDDI総合研究所 奥井宣広（筆者）より、温湿度センサーの湿度情報を用いたセンサー系IoTアプリケーション開発デモの解説を行った（写真3）。また、東京システムハウス 佐藤導吉氏が、本デモを構築するためのアプリケーションの詳細を解説した（写真4）。



■写真5. センサー系アプリケーションデモ構成の外観

デモ構成は写真5のとおりで、温湿度センサーはRaspberry Pi 3 (ADN-AE 1) 上にGPIOで接続され、データを保管するoneM2M PF (IN-CSE) と、データの可視化を行うセンサー



用APP (ADN-AE 2) は、Microsoft Azure上に構築されている。開発においては、温湿度センサー部分はOS-IoTを、oneM2M PFはOM2Mを使用している。センサー用APPはWebアプリケーションとしてJavaScriptで構築されている。また、本デモでは、HTTPプロトコルバインディングとJSONによるシリアルライゼーションが用いられている。

デモのフローは図4に示すとおりであり、各フローの内容は以下のとおりである。

- [手順1、2] 温湿度センサーとセンサー用APPのIN-CSEへの登録
 - [手順3、4] センサー用APPが温湿度センサー配下に〈subscription〉を作成する権限を付与
 - [手順5] 湿度情報を格納するディレクトリの生成
 - [手順6] 温湿度センサーから湿度データが送信されてきた場合に、センサー用APPに転送するための設定情報作成
 - [手順7] oneM2M PFへの湿度データの書き込み
 - [手順8] 手順7で作成されたデータをセンサー用APPに転送
- デモ解説では、各シーケンスがoneM2Mの仕様においてどのような意味を持つか、各シーケンスのHTTPヘッダとJSONメッセージについて具体例を挙げて解説した。本デ

モでは、より実サービスのユースケースに近づけるために、oneM2Mのセキュリティ機能の1つであるアクセス制御を利用し、温湿度センサーやセンサー用APPに、必要最低限の権限を割り振るようにしている。デモ解説では、oneM2Mのアクセスコントロールポリシーの書き方について解説も行った。

アプリケーション解説では、以下の内容について、ステップバイステップで解説を行った。

- ・環境構築 (OM2M設定、Raspberry Pi 3設定、OS-IoTライブラリ生成)
- ・各シーケンスのソースコード解説
- ・デモで使用するOS-IoTのライブラリ解説 (表2)

また、実際にRESTクライアントを用いてOM2Mにメッセージを送信し、どのような応答が返ってくるかを実演し、OM2MとRESTクライアントで簡単にテストができることを示した。

4. おわりに

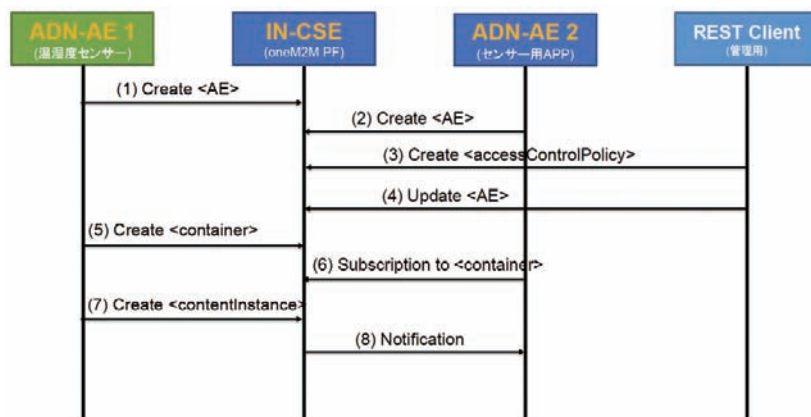
「oneM2Mを使用したIoTアプリケーション開発の始め方」と題してTTC会議室で開催したチュートリアルセッションは、75名の参加者を集め、大変な盛況のうちに終了した。また、ほとんど全ての参加者から、このような取組みを継続して実施することは有意義であるとのフィードバックを頂いた。

筆者はじめ、oneM2M標準化活動の推進、技術の普及を目指すメンバーは、ARIB / TTCを通じて、参加者に会場でアプリを試作していただくハンズオンセッションの実施なども念頭に、今後も開発者目線に立ったセミナー／ワークショップの継続的な実施を鋭意検討している。

なお、今回のチュートリアルセッションに使用した資料はTTCウェブサイト^{*5}より入手可能である。

■表2. OS-IoTのライブラリにおける主なメソッド

No	メソッド名	概要
1	::onem2m::initialize()	初期化処理を行う
2	::onem2m::terminate()	後処理を行う
3	::onem2m::setHostName()	oneM2Mサーバのホスト名を設定する
4	::onem2m::setFrom()	ヘッダーのX-M2M-Originの値を設定する
5	::onem2m::createResource()	リソースの生成を行う
6	::onem2m::retrieveResource()	リソースの取得を行う



■図4. センサー系アプリケーションデモのシグナリングフロー

*5 パスワード保護されていますので、資料ご希望の方は、TTC事務局 (i3cinfo [at] s.ttc.or.jp) までご連絡下さい。(送信の際には、「[at]」を「@」に置き換えて下さい。)