

M2Mサービスプラットフォームの開発

松田 尚久・佐藤 裕一
小関 昌一・永井 信行

要 旨

本稿では、今後主流となっていく水平統合型のM2Mビジネスを支えるCONNEXIVE M2Mサービスプラットフォームの要素技術について紹介します。

データモデル、ワークフローといったコーディングレス技術の採用や、M2Mという大規模トラフィックに耐える大規模高速化への取り組み、各種標準化への対応などを行い、NECのM2Mビジネスを支えるM2M基盤を提供します。

キーワード

- M2M ●サービスプラットフォーム ●水平統合
- データモデル ●ワークフロー

1. まえがき

従来、M2Mの領域においては垂直統合型で個別最適のシステム開発を行うのが主流でした。しかし、お客様の環境に統合ソリューションを導入したり、NECで自主クラウドビジネスを実施したりする場合には、個別最適のシステムではなく、

さまざまなサービスインタフェース、デバイスインタフェースを柔軟かつ容易に提供可能な水平統合型のプラットフォームが必要になってきます（図1）。

弊社では、水平統合型の製品としてM2Mサービスプラットフォーム（以下、M2M PFと略す）を開発しました。

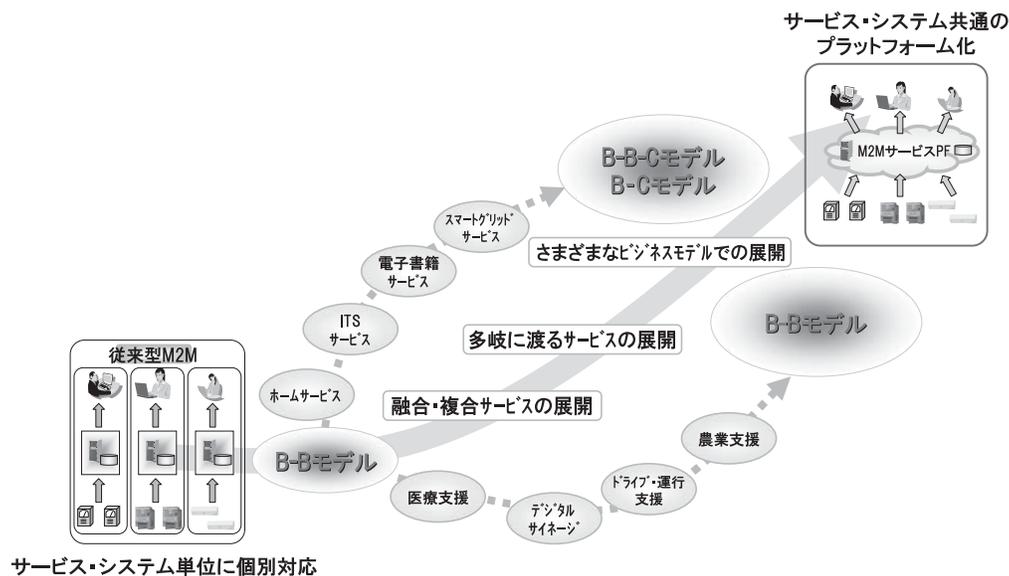


図1 垂直統合型から水平統合型へ

2. M2M PFの概要

2.1 M2M PFが解決すべき課題

M2M PFを開発するうえでの課題として、以下の5つが考えられます。弊社では、これらの課題を解決できる機能を開発しました。

- (1) 収集されるデータの種類やインタフェースの種類は膨大で、プログラムを書き換える場合にどこに影響があるのか把握できないため、修正の工数が膨大になる。
- (2) 固定のロジックで、個別にコーディングをしていると、数多くの統計処理や制御、イベント処理を実装するのにコストが掛かる。
- (3) 従来のようにサービスAPに独自のシステム間インタフェースを定義していると調整に労力が掛かる。また数多くのサービスがデータを共有するようになるとセキュリティを担保できる堅牢なインタフェースが必要。
- (4) 大量にモノがつながるシステムとなり、データベース関連の性能に不安が残る。
- (5) デバイスを開発しても、標準/独自インタフェースが大量にあり、サーバ側と確実につながる仕組みが必要になる。

2.2 M2M PFが実装している仕組み

上記の課題に対して、M2M PFでは以下のような機能を実装しています。

(1) 動作/モデル/インタフェース変換の論理的分割

デバイスプロキシ機能はデバイスと直接の通信を行う機能です。本機能にて、デバイスの仕様ごとの通信プロトコル（HTTP (Hypertext Transfer Protocol) などのアプリケーション層からICMP (Internet Control Message Protocol) などのネットワーク層）や、電文のシーケンスの差異を吸収します。

データゲートウェイ機能は、デバイスプロキシ機能より受け取ったデバイスのデータを解析し、システム内で扱いやすいデータ形式に正規化し、データベースに蓄積します。デバイスのデータの解析はデバイステンプレートと呼ぶXML形式のデータモデルに従って処理されます。デバイステンプレートの読み込みによって、データベーススキーマも自動的に生成されます。

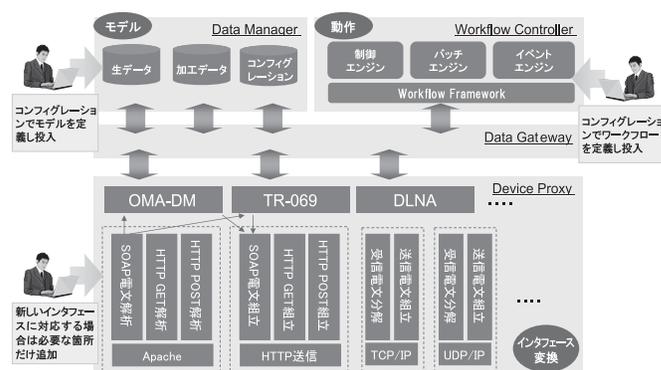


図2 ワークフローによる動作定義

上記の機能を通じてデータベースに追加されたデータは、動作定義を行うワークフロー機能によって利用されます（図2）。ワークフローについての詳細は後述します。

(2) ワークフローによる動作定義の実現

M2M PFが蓄積する生データに対し、サービスアプリケーション（以下、サービスAP）の要件に応じた加工データの生成を行います。この加工の動作定義を行うのがワークフローです。ワークフローが実行されるのは、サービスAPから要求を受け付けたとき（制御系ワークフロー）、デバイスからイベントを受信したとき（イベント系ワークフロー）、一定周期（バッチ系ワークフロー）の3種類が存在します。

M2M PFではデータの検索や挿入、更新といった単純なものから、合計や平均などを求めるといった多様な処理をワークフロー部品として取り揃え、各ワークフローはワークフロー部品を組み合わせることでデータの加工を可能にしています。そのために、DI (Dependency Injection) コンテナとしてSpring Framework（以下、Spring）を採用し、スケーラビリティやカスタマイズの実現に利用しています。SpringはBean定義ファイルと呼ぶXMLファイルに部品間の関係を記述し、実行時は定義ファイルに従ったインスタンスを生成することができます。部品間は共通のインタフェースに従った実装を行うことで、M2M PFは部品が追加されたり組み合わせが変わったりしても、コーディングをすることなく新しいワークフローが実現できます。

(3) サービスAPへの柔軟かつ堅牢なインタフェース

サービスAPへのインタフェースについては、既存製品であ

M2Mサービスプラットフォームの 開発

る「NC7000-WG」を使用します。NC7000-WGはオープンなWebベースのAPIを利用して、さまざまなサービスを容易かつ安全に市場へ提供できるWebサービス公開基盤ソフトウェアです。

M2M PFが提供するAPIをユーザーのニーズに合わせた形（SOAP、REST、JSON）で公開することにより、ユーザーの利便性を向上することができます。運用者に対しては、認証・認可・SLAのインタフェース、セキュリティ機能を提供し、安全なサービス公開を可能とします。またサービス管理画面を提供し、容易なサービス公開、管理を行うことができます。

(4) 大量データへの対応

M2M PFは、1台のサーバに複数の機能ブロックが同居するミニマム構成から、1つの機能ブロックがスケールアウトしてサーバ間通信を行う大規模構成にも対応します。現時点ではPostgreSQLを使用していますが、将来的にはApache Hadoopなどのデータストアへの対応も予定しています。

また、プログラム実装に関していえば、従来であれば機能ブロック間のインタフェースを大規模構成に合わせてRMI（Remote Method Invocation）で実装し、同一サーバ上で複数の機能ブロックを動作させる場合も処理コストの掛かるRMIで通信を行っていました。この領域でもSpringを活用することで、ミニマム構成であれば機能ブロック間をメ

ソッドコールで、大規模構成ではSpringが提供するRMI機能を利用して通信を行います。

(5) デバイスエージェントの提供

汎用デバイスエージェントは、デバイス上で動作するAPから呼び出されるオープンソースのライブラリとして提供し、M2M PFとの通信を容易に実現できます。現時点でJava及びC++版が存在します。すべてデバイスからM2M PFに向けて通信を行うため、NAT（Network Address Translation）超えなどの複雑な制限がありません。

この他、標準化に準拠したエージェントとしてOMA（Open Mobile Alliance）のDM（Device Management）に準拠したものや、Broadband ForumのTR-069に準拠したものを開発しています。

3. M2M PFを用いた構築手順

本章では、実際に弊社のM2M PFを活用するに当たり、どのように構築が簡易化されたか説明します。図3は主な準備内容を示したものです。

3.1 デバイスとのインタフェース定義

まずは、デバイスとのインタフェースを決める必要があります。

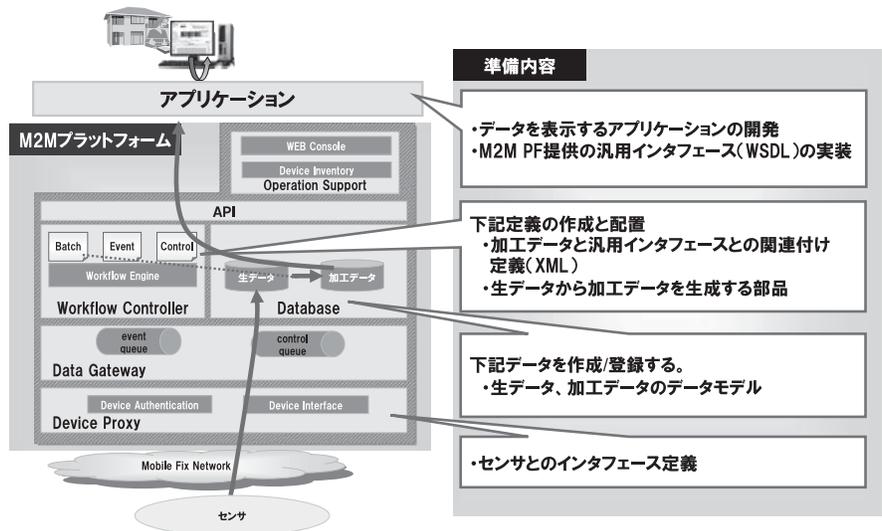


図3 M2M PFの準備内容

ます。デバイスとのインタフェースパターンについては、標準化に準拠したプロトコルを用いるパターン、配布する汎用デバイスエージェントを用いるパターン、独自のインタフェースに合わせてカスタマイズするの計3パターンから、一番適したパターンを選択します。新規に作成する場合は、標準化に準拠したプロトコルパターンを用いることや汎用デバイスエージェントを使用することにより、容易に接続することが可能です。

3.2 データモデルの定義

デバイスから送信されてくる生データ及び、サービスAPに見せたい加工データのモデルを定義します。例えば、1分ごとに送信されてくる温度について生データで定義し、1時間ごとの平均値を加工データで定義します。この定義はXMLベースで行い複雑なコーディングを必要としません。

3.3 ワークフローの定義

生データから加工データを生成するワークフローを定義するファイルを作成します。この作業は、コーディングではなくM2M PFで用意された部品を元にコンフィグレーションで実現できます。

3.4 サービスAPとのインタフェース定義

サービスAPへのインタフェースに関して、M2M PF側でSOAP、REST、JSONといったインタフェースを準備していることにより、デバイス側のインタフェースを意識することなく、容易にM2M PFに蓄積されたデータの取得が可能です。

4. M2M PFのロードマップ

2011年6月より、β版として実証実験を開始し、実証実験で得た経験より、利用シーンが多いと考えられるワークフロー部品の追加やデバイス/プロトコルの追加を行い、2011年9月にV1.0の商用版をリリースしました。

これに伴って2011年12月及び2012年3月と四半期ごとのバージョンアップを予定しており、データのアクセス高速化/大容量化やMVNO (Mobile Virtual Network Operator) に向けた対応、

サポートの充実を図るレポートビルダー対応、GUIによるデータ定義やワークフロー定義の構築自動化やGUIで容易に行えるオペレーション高度化などを予定しています。このように、利便性の高いプラットフォームに向けた機能拡充に取り組んでいきます。

また、対応プロトコルについては、さまざまなデバイスがつながるように、Continua、Z-Wave、ZigBeeなどに準拠したプロトコルを準備していきます。対応プロトコルが増えることにより、接続するデバイスの増加からアプリケーションの増加につなげ、さまざまな利用シーンに対して、最適かつ容易に利用できるプラットフォームを提供します。

5. むすび

情報を一元管理し、インタフェースを集約してエコシステムを構築する必要のあるM2Mビジネスにおいて、M2M PFは早く安く簡単にM2Mシステムを構築できる、弊社のM2Mクラウドビジネスを支える基盤製品です。

今後もお客様のフィードバックも活用しながら、より使いやすい製品を目指して新技術の投入を行ってまいります。

*Apache、Apache Hadoopは、Apache Software Foundationの商標または登録商標です。

*DLNAは、Digital Living Network Allianceの登録商標です。

*PostgreSQLは、PostgreSQLの商標です。

*Javaは、米国及びその他の国におけるOracle Corporation及びその子会社、関連会社の登録商標です。

*Continuaは、Continua Health Allianceの商標です。

*Z-Waveは、Sigma Designs, Inc.の登録商標です。

*ZigBeeは、ZigBee Alliance, Inc.の登録商標です。

*その他記載された会社名及び製品名などは、該当する各社の商標または登録商標です。

執筆者プロフィール

松田 尚久
キャリアサービス事業本部
第三キャリアサービス事業部
マネージャー

小関 昌一
NEC通信システム
ネットワークソフトウェア開発事業本部
第三ソフトウェア開発事業部
プラットフォーム開発センター長

佐藤 裕一
キャリアサービス事業本部
第三キャリアサービス事業部
主任

永井 信行
NEC通信システム
ネットワークソフトウェア開発事業本部
第三ソフトウェア開発事業部
ソフトウェアエキスパート

NEC 技報のご案内

NEC 技報の論文をご覧くださいありがとうございます。
ご興味がありましたら、関連する他の論文もご一読ください。

NEC技報WEBサイトはこちら

NEC技報(日本語)

NEC Technical Journal(英語)

Vol.64 No.4 Network of Things特集

Network of Things 特集によせて
NECが取り組むM2M事業

◇ 特集論文

M2M事業実現のための取り組み

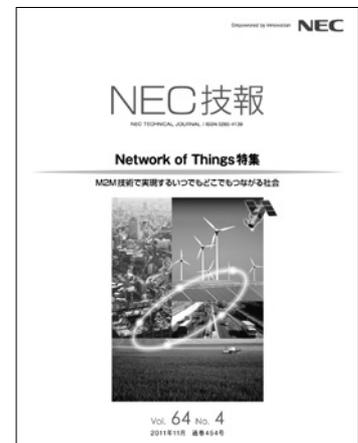
M2Mサービスの現状と展望
M2Mサービスプラットフォームの開発
M2Mグローバル展開の取り組み
M2M標準化動向と遠隔管理技術の標準化活動

M2Mサービス

農業ICTにおけるM2Mサービスプラットフォーム活用
「NECオートモーティブクラウド」への取り組み
ITSにおけるM2Mサービスプラットフォーム活用
M2Mを活用したxEMS(エネルギーマネジメントシステム)
宇宙からの地球観測とM2M～知の構造化に向けて～
産業機械・工作機械業界におけるM2M技術の活用
自販機電子マネー決済におけるM2Mの活用
M2Mクラウドによる業際ビジネスの実現に向けて

M2Mサービスを支えるデバイス及び要素技術

近距離無線規格「ZigBee」への取り組みと開発
M2Mサービスを支えるデバイス製品と活用事例
M2Mデバイスにおける組込みモジュールへの取り組み
エネルギーマネジメントに最適な「スマート分電盤」
M2Mサービスプラットフォームにおける大規模リアルタイム処理技術
画像認識を用いた個体識別による農産物のトレーサビリティ



Vol.64 No.4
(2011年11月)

特集TOP