

# 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

磯山和彦・佐藤正  
喜田弘司・吉田万貴子

## 要旨

近年のネットワークやデバイスなどの進化により多様なM2Mサービスが進展し、M2Mサービスプラットフォームに接続するデバイスの数は増え続けていくと考えられています。本稿では、大量のデバイスから集まるデータから、M2Mサービスにとって意味のある変化をリアルタイムに検出する、複合イベント処理技術について紹介します。

## キーワード

●M2M ●複合イベント処理 ●CEP ●スケーラビリティ ●負荷分散

## 1. まえがき

近年、電力消費量や交通情報など、さまざまな情報を収集するセンサや、ネットワークカメラやスマート家電などのようにネットワークを経由して遠隔操作を行えるデバイス（アクチュエータ）がネットワークにつながるようになってきています。それに伴い、センサからの情報を基に、人手を介することなくデバイスを制御するM2M（Machine to Machine）の技術が注目を集めています。

今後、このようなネットワークに接続されるデバイスは急増し、そこから発生する情報は膨大な量になり、M2Mサービスは大規模化していくことが予想されます。これに対処するため、サービスの大規模化に対応できるプラットフォームが必要になると考えられます。NECでは、デバイスからの大量の情報を扱い、大規模なM2Mサービスを実現するCONNEXIVE M2Mサービスプラットフォームを開発しました。本稿ではそのプラットフォームのなかで、情報の大規模リアルタイム処理を実現する技術を紹介합니다。

## 2. コンテキスト、イベントとは

「コンテキスト」とは、広い意味では「文脈」や「前後関係」と訳されますが、本稿では「人、モノの状態」のことを意味します。このようなコンテキストは、センサなどのデバイスから発せられる情報より推測することができます。例えばGPS（Global Positioning System）センサからの情報を基に、

ある人の位置を推測することができます。また、このようなコンテキストに基づいて最適なサービスを提供するサービスを「コンテキストウェアサービス」と呼びます。

また、上記のようなセンサから発せられる情報や、情報が発せられること自体を「イベント」と呼び、このイベントを基にコンテキストを推定したり、アクションを実行したりする処理を「イベント処理」と呼びます。特に、一つひとつの単純なイベントからは推測できないコンテキストを、複数のイベントの組み合わせから推測し、処理を行う技術を「複合イベント処理（Complex Event Processing：CEP）」と呼びます。

## 3. 要求処理能力

表は、あるサービスを想定した際、M2Mプラットフォームに要求されるイベント処理能力を算出したものです。この表によると、携帯端末や自動車を利用した全国規模のM2Mサービスを実現するためには、プラットフォームには100万イベント/秒以上の処理能力が要求されることが分かります。

例えば全国規模の携帯ユーザーを収容するリアルタイム位置情報サービスを考えると、携帯ユーザー5,000万人が1分に1回位置情報を発信すると、イベントトラフィックの総計は83万イベント/秒になります。

また、サービスのスモールスタート、将来の規模拡大に対応できるようにするため、プラットフォームには拡張可能な（＝スケーラブルな）アーキテクチャを採用することが要求されます。

表 M2Mサービスで必要となる処理能力

サービス	必要イベント数		ユーザーから求められるレスポンスタイム
	イベント数・概要	算出ロジック	
携帯電話 位置情報	・83万イベント/秒 イベント数は83万イベントと大容量になる	・5,000万人(携帯電話人口の半分)÷60秒(1分に1回送信)=83万イベント/秒	・1秒程度 位置情報を用いて利用するサービスではハイレスポンスが求められる
プローブカー	・133万イベント/秒 イベント数は100万イベント超と大容量になる	・自動車数8,000万台(自動車検査登録協会2008)÷60秒(1分に1回情報を収集)=133万イベント/秒	・10秒程度 渋滞情報の提供などで必要となるレスポンスタイムは10秒程度であると推定
家電・ エコモニタリング	・130万イベント/秒 制御タイミング次第でシステムに吸い上げる頻度が決まるため、用途が明確になれば、大容量になる可能性がある	・ハイエンドな一戸建て数40万戸(警備サービス加入世帯数と同程度と判断)×家電数(20個/1戸と仮定)×収集頻度(1分に1回)=約13万イベント/秒(電力モニタリングでは1msec単位で計測しているが、現状ネットワークに流すことができず測定機内に蓄積している)	・1秒程度 電気設備の制御を行うのであれば、ハイレスポンスが求められる
農業	・20万イベント/秒 農作地の状況データ(温湿度、風量、風向、日射量など)を広範囲で吸い上げるため大容量になると予想される	・農家300万戸の40%がフィールドサーバを導入。センサ数平均約10個、1分に1回情報を収集すると仮定すると20万イベント/秒	・1分程度 より高度なIT農業を行うのであれば、レスポンスタイムは1分程度である必要があると推定

次章ではNECが開発した、100万イベント/秒超のイベント処理を実現するプラットフォームアーキテクチャである、「SCTXPF (Scalable Context Platform)」を紹介します。

プラットフォームが実現可能となります。

## 4. SCTXPF

### 4.1 Publish/Subscribeアーキテクチャ

SCTXPFではPublish/Subscribe (通称Pub/Sub) アーキテクチャを採用しています。Pub/Subでは、プラットフォームにイベント処理してほしいアプリケーションは、イベント処理内容をイベント処理ルールとしてプラットフォームに登録(Subscribe) します。イベント処理ルールをSubscribeされたプラットフォームは、デバイスからイベント処理ルールにマッチするイベントが発信(Publish) されると、そのイベントを処理して、処理結果をそのアプリケーションに通知(Notify) します。

Pub/Subアーキテクチャを採用することにより、プラットフォームとアプリケーション、デバイスとのインターフェースが非同期となり、お互いが独立して動作可能となります。これにより、プラットフォームのサービスに対してアプリケーションやデバイスの加入・離脱が容易になり、多数のアプリケーションやデバイスが自由に加入・離脱可能なサービスプ

### 4.2 イベント処理の大規模化アーキテクチャ・コンセプト

CEPの処理を分解すると、シンプルな「ステートレス」な処理の部分と、複雑な「ステートフル」な処理に分解することができます。

例えば「イベントAとイベントBが1分以内に起きたら通知」というCEP処理があったとします。そのときイベントAまたはイベントBを大量のイベントの中から抽出する処理は、その瞬間に受信しているイベントのみを見て処理できます。これがステートレス処理に相当します。一方、イベントAとイベントBが1分以内に起きたことを発見するためには、1分以内に起きたイベントAとイベントBを記憶(これをステートと呼びます)しておかなければなりません。これがステートフル処理に相当します。

一般的に、ステートレス処理は受信したイベントのみを見て処理するので、高速に処理することができます。一方、ステートフル処理は、イベント受信時にステートを記憶し、過去のステートを検索する処理が必要なため、ステートレス処理より重く、遅くなってしまいます。

SCTXPFでは上記を考慮して、CEP処理をステートレス処理の部分とステートフル処理の部分に分解し、それぞれに最

適量のCPUリソースを割り当ててCEPの大容量処理を実現します。

### 4.3 アーキテクチャ

図1にSCTXPFのアーキテクチャを示します<sup>1~3)</sup>。SCTXPFはイベントプロセッサ (EP)、EPコントローラ (EP-CTL)、イベントディスパッチャ (ディスパッチャ) から構成されます。

ステートレス処理はディスパッチャが担当し、ステートフル処理はEPが担当します。また、CEP処理のステートレス処理とステートフル処理への分解はEP-CTLが担当します。以下にそれぞれの機能を示します。

#### (1) EP-CTL

EP-CTLは、アプリケーションからCEPルールを受け取り、CEPルールをEPに割り当てます。更にEP-CTLは、そのCEPルールの処理対象のイベントが、そのルールを設定されたEPに転送されるようにイベントディスパッチャルールを生成し、ディスパッチャに設定します。

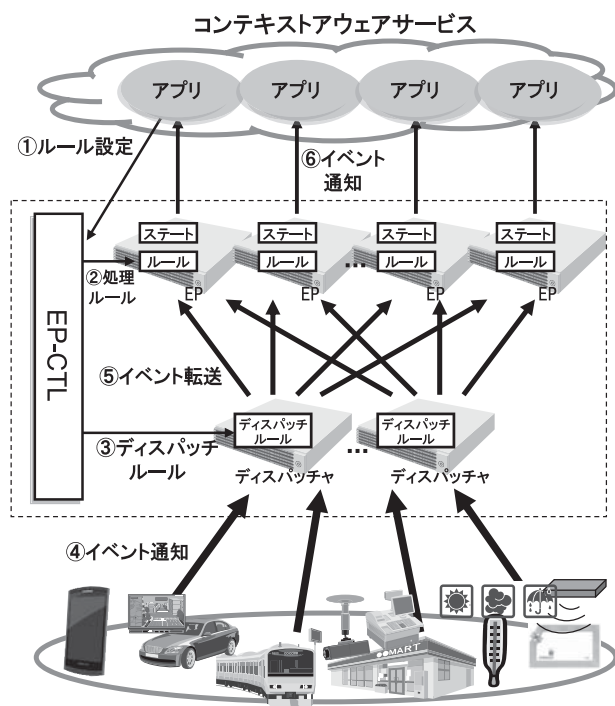


図1 SCTXPFアーキテクチャ

#### (2) EP

EPは、EP-CTLによって設定されたCEPルールに従ってイベント処理を行います。EPに設定されるCEPルールは、アプリケーションから受け取ったCEPルールそのままですが、後述するディスパッチャにおいてステートレス処理は行われるので、EPはステートフル処理のみ行えばよくなります。EPはCEPを行うために、イベントを受信したときにそのイベントにマッチするCEPルールを検索するルールエンジンと、ステートを保持するステート管理部を装備しています。

#### (3) ディスパッチャ

ディスパッチャは、EP-CTLによって設定されたディスパッチャルールに従って、イベントのフィルタリングとディスパッチングを行います。ディスパッチングとは、フィルタリングを通過したイベントを、そのイベントを処理するCEPルールが設定されたEPに振り分けることをいいます。イベントのフィルタリングとディスパッチングを行うために、イベントを受信したときにそのイベントにマッチするディスパッチャルールを検索するルールエンジンを装備しています。EPと異なり、ディスパッチャはステートフル処理を行わないのでステート管理部は装備していません。また、SCTXPFでは、図1のように複数のEP、ディスパッチャの間でステートを共有しないため、お互いに影響し合うことなく、サーバの数を増減することができ、処理能力を拡張することが可能です。よってSCTXPFは第3章で述べたスケーラブルなアーキテクチャであるといえます。

### 4.4 イベント処理

SCTXPFのイベントプロセッサ (EP) では、代表的なものとして、以下のイベント処理を行うことができます。

#### (1) シーケンシャルパターン検出

ある一定時間内にCEPルールで指定する複数のイベントが順番通りに出現したことを検知することができます。この機能を利用することにより、あるルートを人や車が通ったことを検出したり、電気機器の消費電力の異常なパターンを検出したりすることができます。

#### (2) ANDパターン検出

ある一定時間内にCEPルールで指定する複数のイベントがすべて出現 (シーケンシャルパターンと異なり順不同) したことを検知します。

(3)ORパターン検出

CEPルールで指定する複数のイベントのうち、いずれかが出現したことを検知します。

5. ITSカーナビゲーション向けリアルタイム情報提供サービスデモ

NECでは本稿のイベント処理システムを利用した、ITS (Intelligent Transport Systems) カーナビゲーション向けリアルタイム情報提供サービスのデモシステムを開発しましたので紹介します。

本デモシステムでは、全国を走る車にリアルタイムで情報を配信するサービスをエミュレートします。配信する情報としては、車の位置に応じた周辺の店舗の情報や、ユーザーの電子クーポンの使用履歴に応じた電子クーポン情報が挙げられます。

図2にデモシステムのシステム構成を示します。一番左のイベント発生装置が、車が発信するイベントを模擬して、イベントを発生させます。イベントの種類としては車の位置情報やバッテリー (EVを想定) の残量情報、クーポンの使用情報などがあります。イベント発生レートは、全国に稼働している車が5,000万台あり、それぞれの車が1分に1回イベントを発信するとして、それに相当するイベントを送出します。

図2で点線の四角で囲まれた部分が本稿のCEPシステムです。CEPでは、シーケンシャルパターン検出によって、車の位置情報イベントから車の進む方向を検出し、進行方向の店

舗情報を通知します。また、車のバッテリー残量によって近隣の充電ステーションの通知を行います (図3)。

図4に、デモシステムのイベント処理能力のグラフを示し

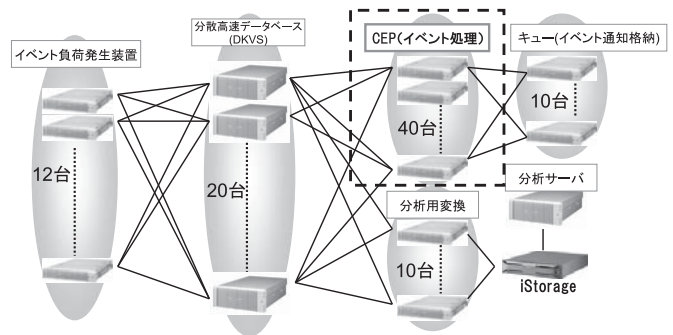


図2 デモシステム構成



図3 ITSカーナビゲーションのデモ画面

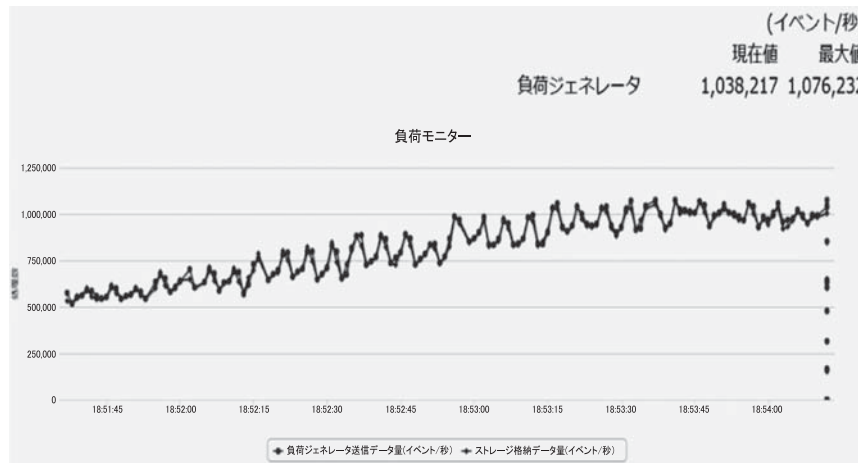


図4 デモにおける負荷処理量グラフ

ます。これを見るとシステムが100万イベント/秒の処理能力を達成していることが分かります。これにより、まずは表の要求処理能力を実現できることが言えます。また、本システムはスケーラビリティがあるのでサーバを増設することにより、更にイベント処理能力を上げることができます。

## 6. おわりに

本稿ではM2Mサービスにおける、スケーラブルで大規模、リアルタイムな複合イベント処理を実現する、イベント処理プラットフォーム技術について述べました。このような技術を採用することにより、今後、本格化し、大規模化していくM2Mサービスに対応できるイベント処理プラットフォームを提供できると考えています。

### 参考文献

- 1) 佐藤正ほか「大規模化可能なコンテキストプラットフォームの提案」情報処理学会、UBI研究会、2005-UBI-9、2005年11月
- 2) 佐藤正ほか「コンテキストウェアサービスのための分散複合イベント処理」情報処理学会、マルチメディア、分散、協調とモバイルシンポジウム (DICOMO2008)、2008年
- 3) K.Isoyama, et al. "SCTXPF: Scalable Context Delivery Platform" International Conference on Communications (ICC) 2011 Workshop on Embedding the Real World into the Future Internet, 2011

## 執筆者プロフィール

磯山 和彦  
サービスプラットフォーム研究所  
主任

喜田 弘司  
サービスプラットフォーム研究所  
主任研究員

佐藤 正  
サービスプラットフォーム研究所  
主任

吉田 万貴子  
サービスプラットフォーム研究所  
研究部長

# 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

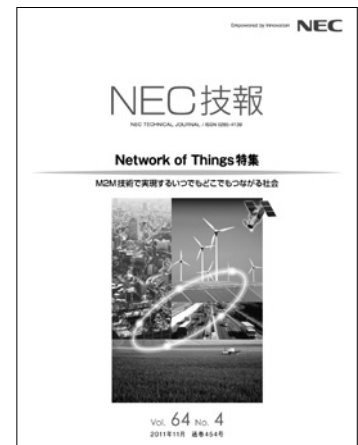
M2Mサービスの現状と展望  
M2Mサービスプラットフォームの開発  
M2Mグローバル展開の取り組み  
M2M標準化動向と遠隔管理技術の標準化活動

#### 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](#)