

M2Mによる自動制御を実現するフレームワークの基礎提案

杉原 史人^{1,a)} 内藤 克浩² 鈴木 秀和³ 渡邊 晃³ 森 香津夫¹ 小林 英雄¹

概要：21世紀の社会では環境問題やエネルギー問題、高齢化社会の進展などさまざまな課題を抱えている。これらの問題の一部はICT技術により解決することが可能と考えられ、特に人の介在なしに自動で制御が行われることが望まれている。このようなサービスとしてM2M(Machine-to-Machine)が近年注目されているが、既存システムでは、機器専用のサーバなどを経由してサービスが提供されているものが多い。そのため、機器数や制御頻度に対する規模拡張性を維持するのが困難になりつつある。そこで、本稿では、M2M機器が直接通信を行うことにより、自動制御を実現可能なフレームワークを提案する。現在のネットワークにはNATなどの様々な制約が存在するため、機器間の直接通信を実現するためには、著者らが既に検討してきたNTMobile(Network Traversal with Mobility)を利用する。NTMobileはIPv4/IPv6混在環境において移動透過性と通信接続性を同時に実現可能であり、対象がどこにいても何であってもEnd-to-Endで安全に通信を行うことができる。また、提案フレームワークでは、異なる機器ベンダー間において、センサから得た情報等をトリガーに何か別のアクションを起こすという動作を規定し、複数のオブジェクトをトリガーに複数のアクションを行うことを可能にする。そして、Webサービスやスマートフォンのアプリを用いてユーザが自由自在にM2M機器の関連付けや自動制御ルールを容易に生成可能にする。

キーワード：M2M, エンド間通信, 連携制御, フレームワーク, NTMobile

1. はじめに

21世紀は環境問題をはじめ、エネルギー問題や高齢者社会など様々な課題を抱えている。これらの課題の一部はICT技術により解決が可能と考えられており、近年の情報通信技術や通信機器の発達により、機器同士が人を介さず相互に情報交換し、自動的に最適な制御が行われるシステムであるMachine-to-Machine(M2M)は解決策のひとつとして注目されている[1]。M2Mは住宅や医療、エネルギーなど幅広い分野での利用が期待されており、利用シーンに応じた様々なM2Mシステムが提案されている[2]。一般的なM2Mシステムはクラウドサービスを利用したクライアント・サーバ型のシステムとして構築される。既存のM2M機器の連携サービスでは、センサを実装した機器が

センシングしたデータをサーバに送信する。また、サーバは受信したデータの解析を行うことにより、所定のルールに従い、所望のM2M機器を制御するなどのフィードバックを行う[3]。M2M機器は今後益々増加することが予想され、既存のクライアント・サーバ型のシステムを採用した場合、サーバでのネットワーク負荷やデータ蓄積にかかるコストが増大するとともに、フィードバックに遅延が生じ、適切な制御を行うことができないといった規模拡張性の問題が存在する。また、クラウド上のデータは暗号化されていない場合があり、生活習慣などのプライバシーに関わる情報が流出する恐れもあり、可能な範囲でサーバを利用しないシステムが望まれている。

本稿では異なるベンダのM2M機器を連携させるためのフレームワークを提案する。提案フレームワークでは、M2M機器間の連携手順を定義するサーバを準備するが、機器間の連携処理は、必要に応じて機器が直接End-to-Endの通信を行うため、制御遅延の削減と規模拡張性の改善が可能である。提案フレームワークを利用することにより、センサなどから得られた情報をトリガーとして、他機器の制御を連携して行うことが可能となる。なお、機器間の連携手順は、Webサービス及び専用スマートフォンアプリケーションを用いて、ユーザが自由に連携手順を定義可能

¹ 三重大学大学院工学研究科電気電子工学専攻
Department of Electrical and Electronic Engineering, Mie University, Tsu, Mie 514-8507, Japan

² 愛知工業大学情報科学部
Faculty of Information Science, Aichi Institute of Technology, Toyota, Aichi 470-0392, Japan

³ 名城大学大学院理工学研究科
Graduate School of Science and Technology, Meijo University, Nagoya, Aichi 468-8502, Japan

a) fsugihara@com.elec.mie-u.ac.jp

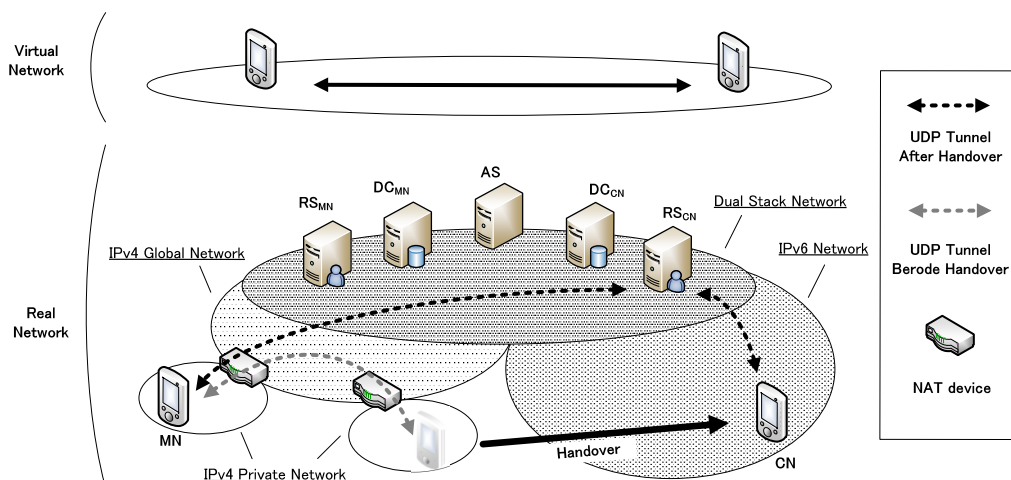


図 1 NTMobile のシステム概要

となる。また、作成した連携手順はインターネット上に置かれたサーバに保存することで、他のユーザが利用することも可能になる。

M2M 機器で利用可能な通信プロトコルを想定すると、現在は IPv4 が主に利用されている。一方、今後の M2M 機器数の増加と、M2M 機器で利用される機器性能などともない、M2M 機器などを想定した軽量プロトコルである 6LoWPAN や CoAP プロトコルなどが既に標準化されている。そのため、今後の M2M は IPv6 の対応も必要になると考えられる。機器間の End-to-End 通信を実現可能なプロトコルは多数提案されているが、IPv4 と IPv6 を共に対応する提案は限られている。そこで、提案フレームワークでは通信部分にこれまで著者らが開発してきた Network Traversal with Mobility (NTMoible) を使用する [4],[5],[6],[7]。NTMobile は IPv4/IPv6 混在環境において移動透過性を実現することが可能であり、NTMobile 用の仮想 IPv6 アドレスを利用することにより、いかなるネットワークへの通信接続性とネットワークを切り替えた場合でもコネクションを切断することなく通信を継続させる移動透過性を同時に実現可能である。そのため、提案フレームワークでは、NTMobile による End-to-End の通信経路を利用することにより、異なる機器ベンダ間においても連携制御を実現可能となる。

2. NTMobile

2.1 概要

図 1 に NTMobile のシステム概要を示す。NTMobile は、NTMobile を実装した端末 (NTMobile 端末) のほかに NTMobile 端末を管理する Direction Coordinator (DC)、NTMobile 端末が直接通信を行うことができない場合に通信を中継する Relay Server (RS) から構成される。NTMo-

bile 端末には NTMobile ネットワーク内で一意に識別可能な仮想 IP アドレスを割り当てることにより、アプリケーションは仮想 IP アドレスに基づいた通信を行う。そのため、実 IP アドレスの変化した場合にも、アプリケーションは仮想 IP アドレスを利用して通信を継続可能である。仮想 IP アドレスを用いた IP データグラムはカプセル化され、UDP トンネルを通して交換される。トンネルの通信経路は 2 種類あり、原則としては両端末間で直接トンネルを構築し、プロトコルバージョンが異なる場合などの両端末間で直接通信ができない場合は、RS を経由するトンネルを構築することで通信が可能となる。NTMobile ではアプリケーションは仮想 IP アドレスを利用することから、相手側のネットワークを意識することがなく、相互通信が可能となる。

- Direction Coordinator

NTMobile 端末の位置情報などを管理し、NTMobile 端末にトンネル構築に関わる各種処理の指示を出す装置である。各 DC は自身に割り当てられた仮想 IP アドレス空間を管理し、重複のないように NTMobile 端末に対して仮想 IP アドレスを割り当てる。また、DC は自身のデータベースに NTMobile 端末の Fully Qualified Domain Name (FQDN)、実 IP アドレス、仮想 IP アドレス、NAT の外側の実 IP アドレスとポート番号を記録している。DC は Domain Name System (DNS) の機能を持っており、DC ごとに異なるドメインを管理する。NTMobile では DC を探索するために DNS の機構を利用しているため、分散配置が可能である。NTMobile 端末は IPv4 および IPv6 を利用する可能性があるため、DC はデュアルスタックネットワークに設置されることを想定する。

- Relay Server (RS)

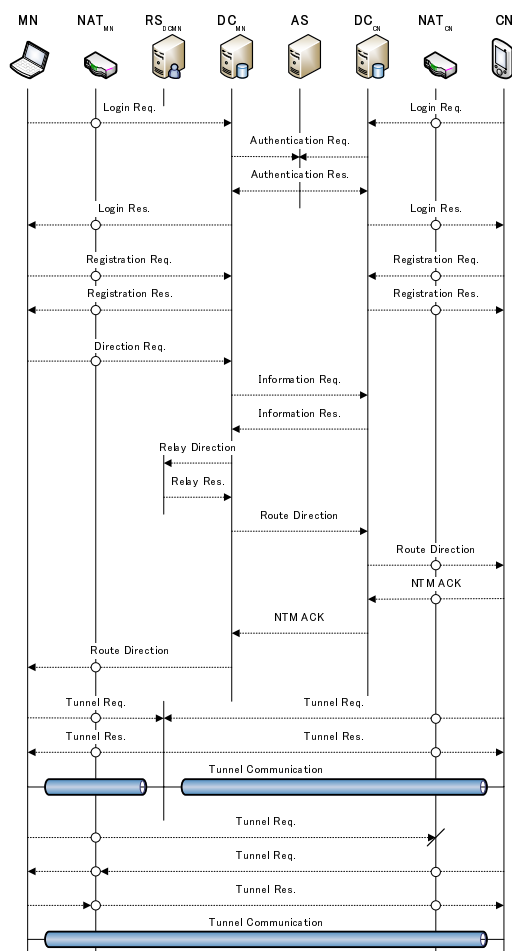


図 2 NTMobile のトンネル構築プロセス

NTMobile 端末が NAT 越え問題や IPv4/IPv6 ネットワークの混在により、直接通信を行うことができない場合に、通信の中継を行う。また、RS は DC により管理されており、DC の指示により中継処理を行うため、多数の RS を用いた分散処理も実現可能である。DC 同様に、RS は IPv4 および IPv6 間の中継も行うため、RS はデュアルスタックネットワークに設置されることを想定する。

● NTMobile 端末

アプリケーションが利用する仮想インタフェースを持ち、仮想インタフェースには、DC から割り当てられる仮想 IP が設定されることにより、アプリケーションは仮想 IP を用いた通信を行う。また、仮想 IP を用いた IP データグラムは UDP によるカプセル化されることで、実 IP アドレスの変化を隠蔽している。NTMobile 端末は実 IP アドレスが変化した場合、自身の DC に新たな実 IP アドレスの登録を行うことで、移動透過性と接続性を実現している。

3. 提案システム

3.1 概要

図 3 に提案システムの構成を示す。提案システムは、既存の NTMobile のサーバ群に加えて、M2M 機器を管理する各ベンダの M2M 機器管理サーバ、連携制御内容を管理するレシピサーバ、提案フレームワークと NTMobile の機能を持つ M2M 機器から構成される。なお、本研究では、連携制御内容をレシピと呼ぶ。

● レシピサーバ

ユーザのアカウントや M2M 機器、レシピの情報を管理するサーバである。NTMobile の端末の機能を有しており、デュアルスタックネットワーク上に設置することを想定している。レシピサーバは、様々なベンダへの対応を想定しており、M2M 機器管理サーバと通信を行うことにより、登録 M2M 機器とレシピサーバの紐付けと、登録 M2M 機器の機能情報を取得する。

● M2M 機器管理サーバ

M2M 機器の販売ベンダが運用するサーバであり、各 M2M 機器が持つ機能情報とレシピサーバとの紐付け処理を行う。なお、M2M 機器の認証はデジタル証明書などを用いることを想定する。

● M2M 機器

M2M 機器は NTMobile の機能と提案フレームワーク上のアプリケーションが実装される。提案フレームワークを利用したアプリケーションは、外部センサからの情報取得、外部機器の制御、レシピサーバとの通信、M2M 機器間の通信を行うことにより、連携動作を実現する。

提案フレームワークのレシピでは、M2M 機器内の制御及び機器間の制御を記述可能である。レシピでは、レシピの発動条件を示すトリガーと実行内容を示すアクションにより定義される。

● トリガー

M2M 機器に接続されている外部センサ及び Web サービスなどの外部サービスから得られる情報を、予め定義したルールにより評価することで発動の有無が決定される。

● アクション

トリガーの発動に対応する実行内容を示し、M2M 機器に接続されているアクチュエーターなどの外部機器の操作及び他 M2M 機器へのトリガーを生成可能である。

3.2 動作例

動作例として 2 つの M2M 機器が連携する様子を示すために、ベンダの異なる M2M 機器 A,B を例に動作手順を示す。手順は大きく 3 つに分かれており、各シーケンス図を

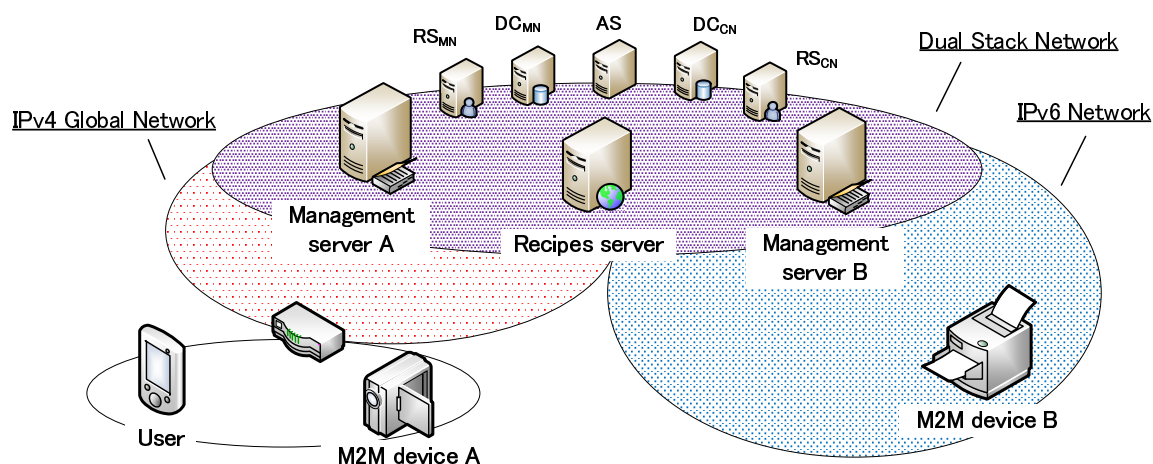


図 3 提案システム概要

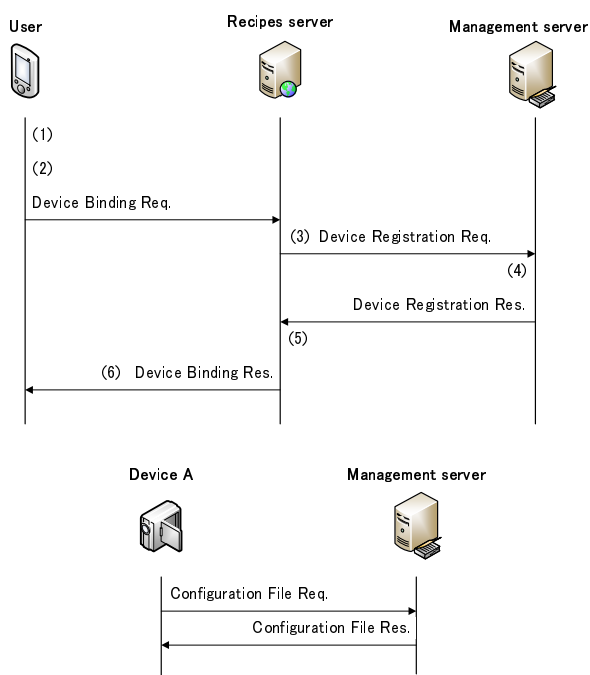


図 4 プロビジョニングシーケンス

用いて説明する。

3.2.1 プロビジョニング

図 4 は提案フレームワークの M2M 機器の設定手順のシーケンスを示す。これは、初めて利用する M2M 機器をユーザーのレシピサーバと紐付けるための手順である。各手順について、以下で説明する。

(1) 提案フレームワークに対応する M2M 機器には、ベンダと機器を一意に認識するための識別情報が提供されていることを前提とする。

(2) ユーザが M2M 機器 A 及び B を初期設定する際、ユーザは機器の識別情報を取得する。また、識別情報を Web アプリケーションなどを利用して、レシピサーバに登録する。

(3) レシピサーバは識別情報に基づき、対応するベンダ管理サーバにアクセスし、紐付ける M2M 機器の紐付け要求を行う。

(4) 要求を受けたベンダ管理サーバは、所望の機器の紐付けが完了した際に、レシピサーバに登録完了の返信を行う。

(5) レシピサーバはベンダ管理サーバにアクセスし、紐付けた機器が持つトリガに関する API とアクションに関する API のリストを取得する。

(6) レシピサーバは M2M 機器のレシピサーバへの紐付けが完了したことを、アプリケーションに通知することで、紐付け処理を完了する。

3.2.2 レシピ登録

図 5 は提案フレームワークにおける、M2M 機器 A 及び B 間の連携動作のレシピ設定のシーケンスを示す。また、各手順について、以下で説明する。

(1) ユーザはアプリケーション上で M2M 機器 A および B が持つ API を紐付けるレシピを任意に定義する。

(2) 本例では、M2M 機器 A がトリガーとなり、M2M 機器 B のアクションを引き起こすレシピを想定しており、M2M 機器 A と B の連携動作が必要となる。そこで、レシピサーバは定義されたレシピを各機器に転送を行う。

(3) レシピサーバは M2M 機器 A 及び B に対して NTM-Mobile の機能を用いてトンネル通信を開始する。

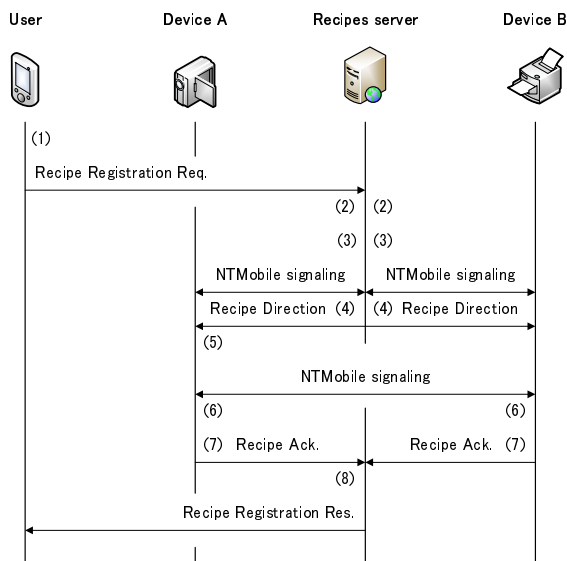


図 5 レシピ登録シーケンス

- (4) レシピサーバはトンネル構築後、M2M 機器 A 及び B に対して、Recipe Direction を送信することで、レシピを送信する。
- (5) レシピを受信した M2M 機器 A 及び B は、レシピが実行可能なのかを検証するため、テスト動作を機器間で行う。この際も、テスト動作実行前に NTMobile の機能を用いてトンネル通信を開始する。
- (6) トンネル構築後に、本例ではトリガー元となる M2M 機器 A から M2M 機器 B に向けて、Trigger test request メッセージを送信し、M2M 機器 B は Trigger test response メッセージを返信することで、レシピが実行可能であることを確認する。
- (7) M2M 機器 A 及び B は検証結果をレシピサーバに報告することにより、レシピが有効となる。
- (8) レシピサーバはアプリケーションにレシピが正常に登録されたことを通知する。

3.2.3 実際のレシピの動作

図 6 は提案フレームワークを用いてレシピに基づいた機器間連携を行う際のシーケンスを示す。また、各手順について、以下で説明する。

- (1) 本例では、M2M 機器 A で発生したトリガーに基づき、M2M 機器 B のアクションを引き起こす。そのため、M2M 機器 A はトリガー動作に基づき、M2M 機器 B への NTMobile のトンネル構築を行う。
- (2) M2M 機器間のトンネル構築が完了した後、M2M 機器 A と M2M 機器 B 上の提案フレームワークは相互に情報を交換することにより、トリガーに基づいたアクションを識別する。本例では、M2M 機器 A が M2M 機器 B に向けて、Action Request を送信する。

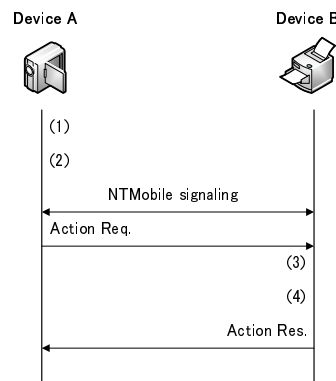


図 6 レシピ登録シーケンス

- (3) M2M 機器 B はレシピに基づいてアクションを行うことで、機器連携を実現する。
- (4) M2M 機器 B は Action Response を返信することで、レシピが正常に実行されたことを相互に確認する。

4. 実装計画

提案フレームワークを実装することを目的に、実装デザインを図 7 に示す。実装デザインに、M2M 機器の機能、デバイス管理サーバの機能、レシピサーバの機能と管理情報の連携性を示す。

4.1 デバイス管理サーバ

デバイス管理サーバは、M2M 機器の認証用のデータベース、レシピサーバとの紐付け用のデータベース、M2M 機器が持つ API リストのデータベースを持つ。認証用のデータベースは M2M 機器が正規の端末であるのか認証するために利用される。また、レシピサーバとの紐付け用データベースは、レシピサーバからの要請に応じて、M2M 機器を紐付けるために利用される。さらに、M2M 機器が持つ API リストのデータベースは、レシピサーバが利用可能な M2M 機器の機能を記述したものである。

4.2 M2M 機器

M2M 機器には、予め M2M 機器の認証用の情報が設定されている。この認証用情報を利用してデバイス管理サーバと通信する際に、正規機器であることを証明する。また、デバイス管理サーバ上のレシピサーバとの M2M 機器の紐付け情報であるプロビジョニング情報をデバイス管理サーバから取得する。また、レシピ実行用に、レシピ管理機能、トリガー機能、アクション機能を持つ。さらに、M2M 機器間の通信及びレシピサーバとの通信のために NTMobile の機能を持つ。レシピサーバからレシピを配布された場合、レシピ管理機能がトリガー機能とアクション機能に適切に情報を設定する。また、トリガー機能は外部からの情

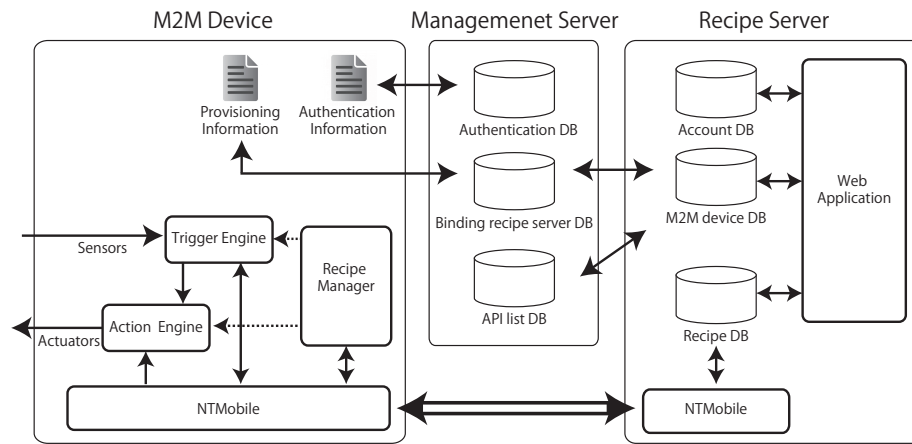


図 7 実装デザイン

報に応じて、対応したトリガーを発動させる。さらに、アクション機能は、トリガーをきっかけに、所定の処理を実行する。

4.3 レシピサーバ

レシピサーバには、ユーザを管理するアカウント管理用のデータベース、紐付けた M2M 機器を管理するデータベース、レシピを管理するデータベースが準備されている。ユーザは、Web アプリケーションなどから、これらのデータベースにアクセスを行う。また、レシピの情報は NTMobile の機能を利用して、M2M 機器に配布される。

5. まとめ

本稿では、M2M 機器が直接通信を行うことにより連携が可能なフレームワークの基礎設計を行った。提案フレームワークを利用することにより、ユーザーは異なるベンダの機器間の連携を容易に設定できる上、End-to-End の動作により規模拡張性も維持可能である。今後は、提案フレームワークのプロトタイプ実装を完成させることにより、具体的な性能評価を行う予定である。

謝辞 本研究の一部は農林水産省 革新的技術創造促進事業 (異分野融合共同研究), 科研費 (26330103, 15H02697) の助成を受けたものである。記して謝意を表する。

参考文献

[1] G. Wu, S. Talwqr, K. Johnsson, N. Himayat and K. D. Johnson, "M2M: From Mobile to Embedded Internet," IEEE Communications Magazine, Vol.49, No.4, pp.36-43, Apr. 2011.
 [2] 猿渡俊介, 森川博之, "モバイルネットワークを支える技術: 3.M2M の情報流," 会誌「情報処理」, Vol.55, No.11, pp.1269-1274, 2014.
 [3] 藤田隆史, 後藤良則, 小池新, "M2M アーキテクチャと技術課題," 電子情報通信学会, Vol.96, No.5, pp.205-312, 2013.
 [4] 鈴木秀和, 上酔尾一真, 納堂博史, 西尾拓也, 内藤克浩, 渡

邊晃, "IPv4/IPv6 混在ネットワークにおいて通信接続性と移動透過性を実現する NTMobile の研究," マルチメディア, 分散, 協調とモバイル (DICOMO2012) シンポジウム論文集, pp. 2391-2401, Jul.2012.
 [5] 上酔尾一真, 鈴木秀和, 内藤克浩, 渡邊晃, "IPv4/IPv6 混在環境で移動透過性を実現する NTMobile の実装と評価," マルチメディア, 分散, 協調とモバイル (DICOMO2012) シンポジウム論文集, pp. 1169-1179, Jul.2012.
 [6] 鈴木秀和, 上酔尾一真, 水谷智大, 西尾拓也, 内藤克浩, 渡邊晃, "NTMobile における通信接続性の確立手法と実装," 情報処理学会論文誌, Vol.54, No.1, pp.367-379, Jan. 2013.
 [7] 内藤克浩, 上酔尾一真, 西尾拓也, 水谷智大, 鈴木秀和, 渡邊晃, 森香津夫, 小林英雄, "NTMobile における移動透過性の実現と実装," 情報処理学会論文誌, Vol 54, No.1, pp.380-393, Jan. 2013.