

## スマートコミュニティ実現に向けた IEEE1888 による 機器情報管理手法の検討

赤井和幸<sup>†</sup> 福田富美男<sup>†</sup> 福島伸夫<sup>†</sup>  
梁田龍治<sup>†</sup> 古川嘉識<sup>†</sup>

近年、施設管理用プロトコルをベースにして標準化された IEEE1888 は、異種プロトコルをゲートウェイによって吸収する仕組みと機器やシステムに依存しないメソッド体系により、多様な機器やアプリケーションを接続して新しいサービスを行うスマートコミュニティを実現するためのプロトコルとしても注目されはじめています。しかし既存の IEEE1888 に関連する研究や実証実験では機器同士の相互接続を中心に実施されることが多く、機器の管理・検索を行うレジストリとの通信及び機器情報管理手法への活用についての検討は始まったばかりである。本研究では IEEE1888 のレジストリ機能の実装を行い、実際にこの機能が IEEE1888 のシステムにおける機器情報管理機能として動作可能であることを複数ベンダ参加の実証実験において確認した。また、宅内に設置されるゲートウェイについては所定のリソースの範囲内で動作可能であることを検証するとともに、レジストリ機能と連携させることで他のコンポーネントと容易に接続できることを確認した。

### A Study on Device Management Method Using IEEE1888 in Smart Communities

KAZUYUKI AKAI<sup>†</sup> FUMIO FUKUDA<sup>†</sup>  
NOBUO FUKUSHIMA<sup>†</sup> RYUJI YANATA<sup>†</sup>  
YOSHINORI FURUKAWA<sup>†</sup>

Recently, IEEE1888 which was standardized based on the protocol for facility management has been attracted attention as a protocol for the smart communities because it has the architecture and the methods to communicate with devices in different protocols through the IEEE1888 Gateway. On the other hand, studies on the IEEE1888 Registry which has functions to search IEEE1888 Components and to manage IEEE1888 Components just began, because many of the researches and experiments related to IEEE1888 have been carried out to evaluate device interoperability. In this study, we have implemented the IEEE1888 Registry, and we have confirmed that the IEEE1888 Registry is effective in the device management function in the multi-vendor demonstration experiment. In addition, we have verified that the IEEE1888 Gateway in the home network can work within the certain resource limit, and connect other IEEE1888 Components easily by working with the IEEE1888 Registry.

#### 1. 背景

近年、ネットワークに対応した機器が一般家庭やオフィスに普及しつつあり、これらの機器やネットワークを活用したサービスとそれを実現するシステムの研究や提案がされている。特に、エネルギー使用量の増加により、エネルギーを節約するための様々なサービス、例えば電流センサから得られた情報の可視化（電力の見える化）や人感センサが不在と判定した場合に特定の機器の電源を OFF にするなどのセンサ情報と連動した機器制御サービスなどは多くの試作や製品[1][2]が世に広まりつつある。さらに、宅内やオフィス同士で相互に連携するシステムを構築することで地域内電力平準化などのサービス(スマートコミュニティ)なども可能になると考えられている。

これらのサービスを実現する上ではこれらの機器や管理を行なうシステムが通信するために、使用するプロトコル

や機器をあらかじめ決めておく必要がある。ビルなどのローカルエリアネットワークで構築される施設内では BACnet[3]や IEEE1888[4][5]などのプロトコルで統一してシステムを構築することが可能である。一方、宅内では DLNA[6]や ECHONET[7]に対応した機器が混在し、新しいプロトコルの機器などの追加や変更なども考えられる。また追加や変更時に対応するのは専門の知識を持った人間ではないこともありうる。さらにスマートコミュニティでのサービスは家同士を接続したり、家とビルなどの施設間で接続したりすることもあるため、想定していないさまざまな機器が扱う多種多様なプロトコルが混在する可能性が高くなる。そのためこれらの機器が連携するスマートコミュニティのシステムの実現はより困難になると想定される。そのため本研究では宅内同士及びクラウド上で提供されるシステムまたはサービス（センタ）が IEEE1888 プロトコルの仕組みとレジストリ機能を活用することで宅内機器を容易に管理・連携可能になることを確認することで、スマートコミュニティのシステムへの適用可能性を示す。

<sup>†</sup>NTT コムウェア株式会社 品質生産性技術本部研究開発部  
NTT Comware Corporation  
Research and Development Department  
Core Technology, Quality Management and Engineering Division

## 2. IEEE1888 プロトコル

### 2.1 IEEE1888 プロトコル

IEEE1888 プロトコル（以下 IEEE1888）は施設管理及びスマートコミュニティ用のプロトコルとして2011年3月に標準化されたプロトコルである。接続される機器やシステムをコンポーネントとして扱う。またそれらのコンポーネントの情報を管理するためのレジストリが存在する。IEEE1888 アーキテクチャの概要図を図 1 に示す。

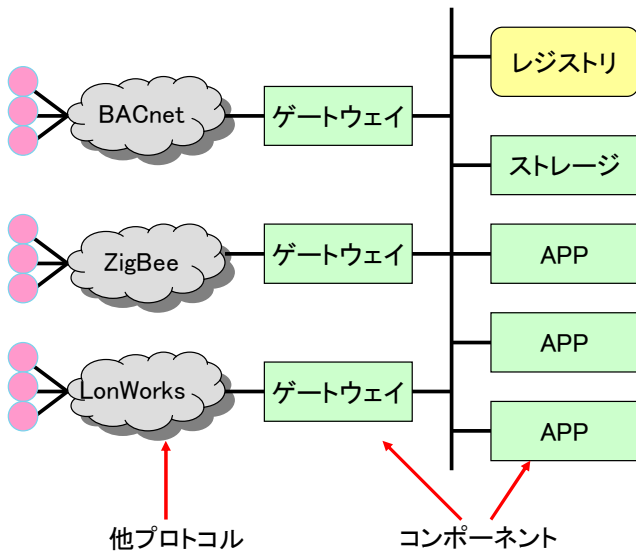


図 1. IEEE1888 アーキテクチャ概要図

IEEE1888 では3種類のコンポーネント(ゲートウェイ, ストレージ, APP) とそれらが相互に通信するための3つの手順(FETCH, WRITE, TRAP), 及びコンポーネントを管理するレジストリとコンポーネントが通信するための2つの手順(REGISTRATION, LOOKUP)が定義されている。これらの手順は IEEE1888 で定義されているメソッドを用いることで実現される。各種手順について表 1 に示す。

表 1. IEEE1888 で定義されている手順

コンポーネント間通信		
手順	メソッド	用途
FETCH	query	ストレージなどからデータを取得する。
WRITE	data	センサのデータなどを書き込む。またアクチュエータに対して操作をする。
TRAP	query	情報の変更などに対して指定したコンポーネントに通知するように設定する。
コンポーネント-レジストリ間通信		
REGISTRATION	registration	コンポーネントまたはポイントの情報をレジストリに登録する。
LOOKUP	lookup	コンポーネントまたはポイントの情報を検索する。

IEEE1888 では実際に通信対象としてアクセスできるコンポーネントの配下に、管理されている機器(機能)やデータを管理している「ポイント」という概念を持つ。これにより、コンポーネント配下に複数のセンサ機器が接続されている場合や、センサデータが違うコンポーネントに分散

して蓄積されている場合でも、その情報を一意に特定して適切な処理を行うことが可能となっている。

また、IEEE1888 は既存の設備管理のプロトコルと比較して以下のような特徴を持つ。

- 電力の見える化や制御などのエネルギーマネジメントに特化

電力値や温度などの環境データを蓄積し、共有するためのコンポーネントとしてストレージの概念が導入されている。そのため、他のセンサ機器やアクチュエータ機器に対する通信と同様のメッセージのやり取りでストレージに対するデータの蓄積や参照などを可能にする。またデータの参照方法(FETCH 手順)として、時間範囲を指定した連続するデータの参照やコンポーネント配下に管理されているデータの一括取得などが定義されている。このため一定期間の使用電力の推移を表示する電力の見える化サービスなどを容易に実現することが可能である。

- SOAP を用いた汎用的な通信と他プロトコルの差異を吸収するゲートウェイ

HTTP/SOAP を用いた汎用的なネットワーク上で動作するプロトコルとして定義されているため、LAN やインターネット環境上で容易に接続が可能である。また他プロトコルで構成されているシステムや機器に対してはゲートウェイでそれらの差異を吸収するということが仕様化されている。これにより、通信を行うコンポーネントはゲートウェイの配下に接続されている他のプロトコルそのものについて意識する必要がない。

- アプリケーション、ゲートウェイ、ストレージのコンポーネント化

センサ機器やアクチュエータだけでなく、データを蓄積するストレージや、そのデータを活用するアプリケーションなどもすべてコンポーネントという概念で扱われ、それらはすべて共通の手順(FETCH, WRITE, TRAP)によって通信が行われる。よって基本的には機器やシステムに依存したメソッドなどを実装する必要がない。

- 各コンポーネントの情報の管理を行う仕組み(レジストリ)

コンポーネントと通信するための URI(Uniform Resource Identifier)やそのコンポーネント配下で管理されている機器やデータの情報を一元管理するレジストリ機能が仕様化されている。このレジストリから必要な情報を参照することで、コンポーネントは事前に URI などの情報を設定しなくても通信先のコンポーネントを探し出したりすることが出来る。

### 2.2 IEEE1888 を用いたスマートコミュニティ

IEEE1888 は施設管理プロトコルから発展しているが、その応用例としてスマートコミュニティへの適応についても検討されている。IEEE1888 を用いることで、他プロトコルで構成されている機器群やシステムを容易に収容できると

ともに、アプリケーションやストレージを分散配置することが可能となり、多数のベンダのシステムやサービスの参入が容易になる環境が可能になると考えられる。IEEE1888で実現できるスマートコミュニティの一例を図2に示す。

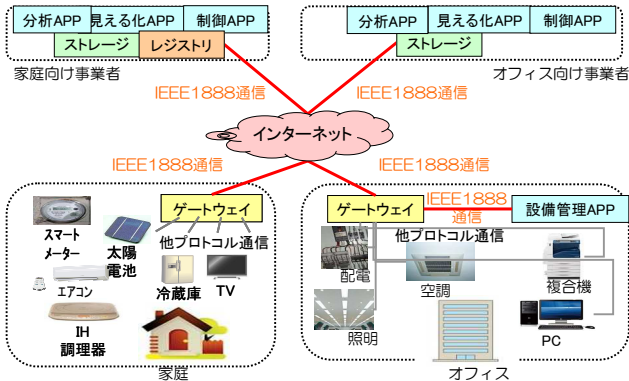


図2. IEEE1888を用いたスマートコミュニティのイメージ  
 施設内のローカルエリアネットワークであればあらかじめ接続される機器やシステムなどは把握できるため、コンポーネントやポイントの情報をレジストリで管理することは必須の条件ではない。一方、本研究でターゲットとしているスマートコミュニティのような環境では、機器の情報を管理して適切な情報を必要な他のコンポーネントに提供することが出来る機能としてレジストリが必要になると想定される。レジストリは機器の場所やセンサ種別などの属性情報を管理できるため、この情報の登録・参照を行うことで宅内機器の追加・変更が容易に対応することが可能になると考えられるためである。しかしながら、IEEE1888プロトコルの既存の研究・検証としてコンポーネント間通信については複数ベンダ間での相互接続検証なども行われており[8]、その妥当性と有用性については十分に知見が示されているものの、コンポーネント-レジストリ間通信やレジストリの活用方法についての検証は始まったばかりである。

### 3. IEEE1888のスマートコミュニティの要求条件と設計

#### 3.1 IEEE1888スマートコミュニティの要求条件

本研究ではスマートコミュニティシステムの一例として、宅内の機器とストレージやアプリケーションが連携するシステムについて検討する。本研究では図3に示すようなシステムを想定する。宅内ではセンサまたはアクチュエータが存在し、宅内のセンサから取得された情報はストレージに蓄積される。またアクチュエータは外部のアプリケーションから制御できる。ストレージは負荷分散や用途によって複数用意されることが考えられ、場所の変更や増設されることも想定される。またストレージに蓄積されたデータを、適切なアプリケーション(見える化APPや機器制御APPなど)が参照する。これによって宅内のゲートウェイ、センサのストレージ及びアプリケーションのそれぞれ

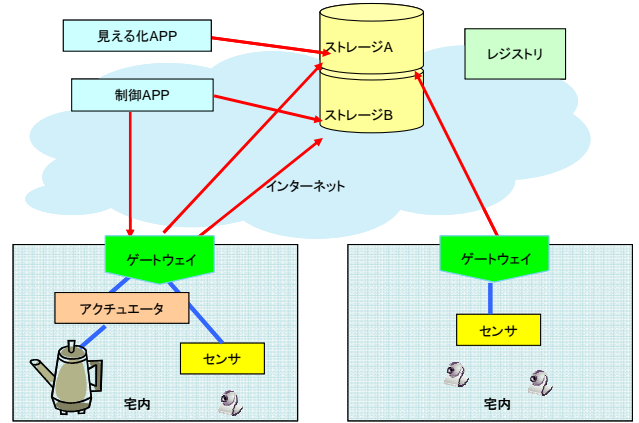


図3. 宅内機器とストレージ・アプリケーションの連携例のコンポーネントが連携して動作するようになる。このようなシステムに必要な要求条件をまとめると以下のようになった。

- 宅内のセンサ機器を扱うゲートウェイは、他のコンポーネントから情報を取得できるようにするため、(負荷分散や用途ごとに振り分けを行うアルゴリズムによって導き出された、適切な)ストレージにデータを蓄積する。
- 宅内のアクチュエータ機器を扱うゲートウェイは、他のコンポーネントからの制御を受け付けることができる。
- 宅内のセンサやアクチュエータを管理するゲートウェイは宅内に設置される機器に搭載され、その機器のリソース内で十分動作可能である。
- 宅内に接続されるセンサ機器やアクチュエータ機器は追加・変更されることを想定し、宅内のコンポーネント及びIEEE1888システムはレジストリを用いることで柔軟に対応できるようにする。

この要求条件に対して、本研究では(1)レジストリを活用した宅内機器の追加接続時の機器情報の管理及び宅内機器とストレージの接続容易化を実現するための通信方法、(2)宅内に設置されるゲートウェイに必要な実装、の2点から検討・考察を行う。

#### 3.2 レジストリの設計と宅内機器管理方式の検討

IEEE1888のレジストリは前述の通り、lookupメソッドとregistrationメソッドから構成される。それらで登録、参照されるデータは「コンポーネント情報」と「ポイント情報」の2つのデータが存在する。そのためlookupは引数によって2種類の使い方が存在することになる。1つ目のメソッドはlookup(point)であり、これはポイントに紐づく属性情報(センサ種別やロケーションなどの任意のパラメータ)をキーとすることでポイントID及びそれに付随する情報を取得するメソッドである。2つ目のメソッドはlookup(component)であり、これはコンポーネントの配下で管理しているポイントの情報(ポイントIDとデータの保持時間)によって、コンポーネントのアクセスURI及び付随

する情報を取得するメソッドである。これらのメソッドを用いることで、コンポーネントの情報について把握していなくてもポイントの属性情報などから最終的にコンポーネントのアクセス URI を取得し、通信することが出来るようになると考えられる。この結果、例えば図3のようなセンタ側に複数のストレージがある場合でも、レジストリを活用することでゲートウェイはそれらのストレージの URI 及び属性情報をすべて保持しておくことが不要になる。

本研究では、宅内の機器に追加・変更が発生したときにシステムが柔軟に対応することを示すことが出来る一例として、機器が追加されたときに適切なストレージにデータを蓄積するための手法を検討する。なお、適切なストレージを選ぶための手法自体については本研究では検討していないため省略する。データを蓄積すべき適切なストレージを決定するためのロジックを持つ場所として(1)ストレージ(2)レジストリ(3)ゲートウェイの3つが考えられる。それぞれの場合での通信方法(計4種類)について以下に示す。

(1) ストレージが蓄積場所決定ロジックを持つ場合

ストレージが宅内のセンサ機器の情報を蓄積するかどうかを判断して、データの蓄積を行う場合について考える。この手法の通信シーケンスの例について図4に示す。

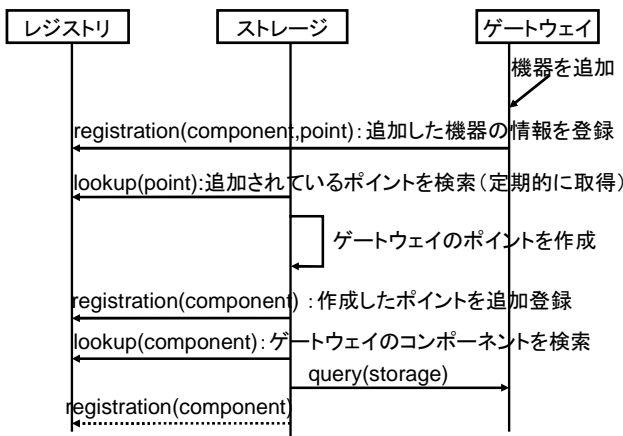


図4. ストレージが蓄積場所を決定する場合1(FETCH)

まず宅内にあるゲートウェイ配下に新しい機器が追加された場合、その情報(コンポーネント情報及びポイント情報)をレジストリに登録する(registrationメソッド)。一方ストレージは、レジストリに対して宅内のコンポーネントの情報に変化や追加があるかどうかを定期的に確認しておく(lookupメソッド)。確認時に蓄積すべきデータがあると判断されるポイントが追加されているとストレージが判断した場合、そのストレージは追加されたポイントの情報を自身のコンポーネント情報に追加してregistrationメソッドを呼び出す。その後、ストレージがゲートウェイのURIを取得するためにコンポーネント検索を行い(lookupメソッド)、その後そのURIに対してFETCH手順(query(storage))を用いてセンサのデータを取得する。その後ストレージは必要に応じてレジストリに自身が持っているポイントのデータの

情報を登録するようにする。なお、ストレージがデータ蓄積場所決定ロジックを持っている場合、データを蓄積するための通信方法は2種類あり、(1)-1、図4に示すFETCH手順によってストレージがゲートウェイからデータを取得する場合(1)-2、図5に示すWRITE手順(data())によってゲートウェイがストレージにデータを書き込む場合、が考えられる。

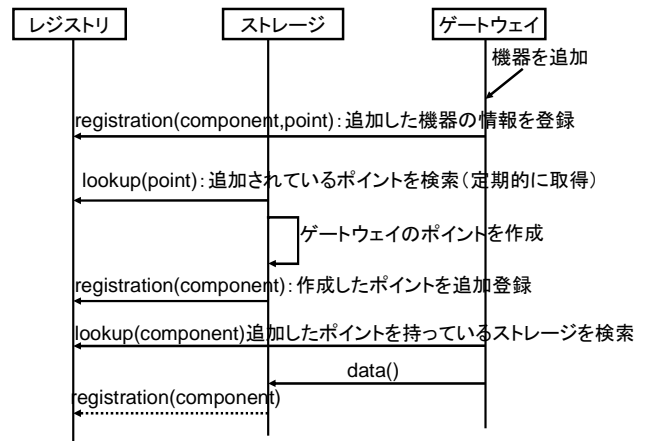


図5. ストレージが蓄積場所を決定する場合2(WRITE)

なお、(1)-2の方法を用いる場合、ゲートウェイはlookup(component)メソッドを用いてストレージを探す通信が必要となる。

(2) レジストリが蓄積場所決定ロジックを持つ場合  
 通信のシーケンスについて図6に示す。

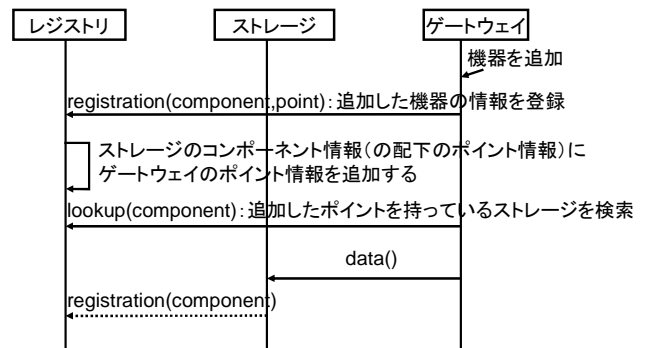


図6. レジストリが蓄積場所を決定する場合

ゲートウェイからregistrationメソッドによって新しくポイント情報が追加された場合、その情報から蓄積すべき適切なストレージをレジストリが判断する。その後レジストリは自身で、選択したストレージのコンポーネント情報にその情報を追加する。その後、ゲートウェイは手順(1)-2と同様にlookup(component)でストレージのコンポーネント情報を探し、その後WRITE手順によってゲートウェイがストレージにデータを書き込む。

(3) ゲートウェイが蓄積場所決定ロジックを持つ場合

ゲートウェイが書き込むべきストレージの情報を把握し、自身で判断して適切なストレージにデータを蓄積する通信シーケンスについて図7に示す。

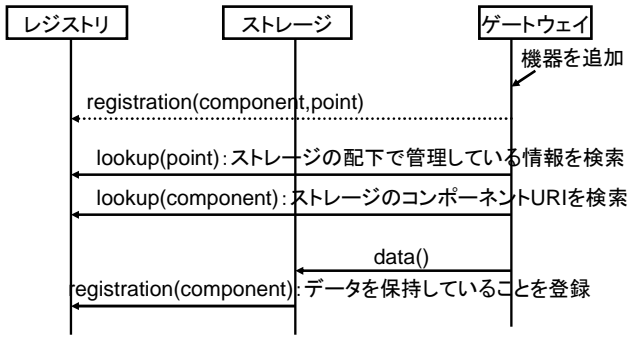


図7. ゲートウェイが蓄積場所を決定する場合

ゲートウェイは機器が追加された後、lookupメソッドを用いて、適切であると考えられるストレージの検索を行う。最初からコンポーネントの情報が分かっている場合はアクセスURIを検索するためのlookup(component)メソッドのみでよい。一方、今回想定するシステムのように最初から書き込むべきストレージを決めていない場合には、ゲートウェイとストレージとで予め決めておいたルールに従い、lookup(point)メソッドを用いることで適切なストレージの検索を行う。

### 3.3 宅内のゲートウェイの設計方針

宅内のセンサやアクチュエータを管理するゲートウェイは宅内に設置される機器に搭載されるため、そのリソース内で動作可能である必要がある。また機器の追加や取り外しに対して柔軟に対応できる仕組みが必要となる。本研究では宅内のゲートウェイを動作させる環境としてOSGiフレームワーク[9][10]を採用する。OSGiフレームワークはソフトウェアを「バンドル」という単位で、プログラムを部品化させることができ、バンドルの追加・削除・更新をする際にシステム全体を停止させることなく実施することが可能となる。そのため動的に機器が接続・取り外しがされる状況下では、システムを止めずにソフトウェアを変更することが可能なOSGiフレームワークを適用することが有用であると考えられる。OSGiフレームワーク上で動作するゲートウェイについて、デバイスを追加したときのイメージを図8に示す。

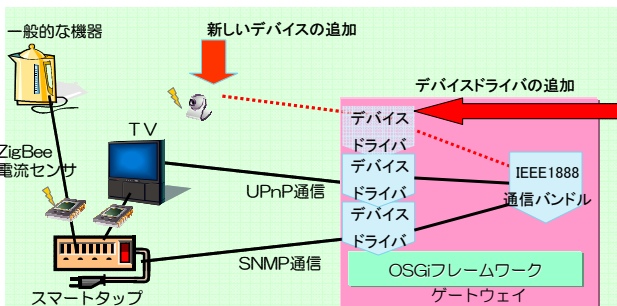


図8. OSGiフレームワーク上でのデバイスの追加イメージ

IEEE1888の通信を実際に行うゲートウェイバンドルと機器との通信を行うデバイスドライババンドルをそれぞれ独立させることが出来れば、機器の追加などが発生した場合にもシステムを停止させずにデバイスドライバだけを

システムに組み入れることが可能になる。さらに現在、宅内に接続されている機器の名称を特定するシステム及びアルゴリズムについて研究されており[11][12]、この技術を用いることで宅内に追加される機器を自動で特定することが可能になると考えられる。そのシステムを用いることで機器の追加から自動で機器の特定を行い、必要なデバイスドライバの追加・変更を行うことができることも可能であると考えられる。

ゲートウェイでは、要求条件を満たすためにIEEE1888の各メソッドを実装する必要がある。ただし、ストレージに対してデータを蓄積するための仕組みについては前述の通り、複数のパターンが考えられ、それによって実装すべきメソッドは異なる。また、宅内のゲートウェイを動作させる環境は性能の良い機器ではなく、使用できるリソースについても制限されているため実装されるバンドルのファイルサイズは出来るだけ小さいほうが良い[13]。具体的には、サービス利用時のファイルサイズ制限から1.5MBまたは3MB以内に収めることが望ましい。さらに図9のように、レジストリやストレージを宅内に設置するようなケースも考えられる。このような構成にすることでレジストリを宅内とインターネットで分散配置することで宅内機器と宅外の機器のアクセスコントロールなどが実現できると推測される。よって本検証ではレジストリ及びストレージもゲートウェイと同様のOSGiフレームワーク上での実装・評価を行う。

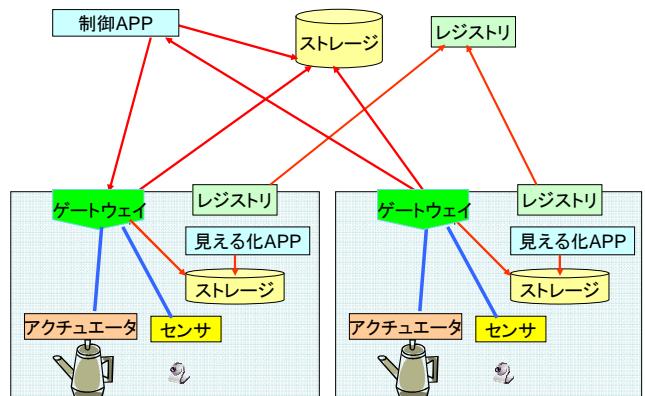


図9. 宅内のコンポーネントとセンサとの連携例

## 4. 検証と評価

### 4.1 検証環境

本検証で使用した環境は以下の通りである。

- OSGi Felix Framework Distribution 3.0.8  
データベース: H2 1.2.147
- Java SDK 1.4.2

また、宅内に接続されるセンサ及びアクチュエータ機器として以下の機器を検証のために使用した。

- リモート電源制御装置: APC7900
- アドソル日進製 ZigBee 人感センサ, 電流センサ



## 4.2 各メソッドの妥当性評価

IEEE1888 の各メソッド(data, query, registration, lookup)のクライアント及びサーバの機能を実装し、実装した各種メソッドが正しく実装・動作できているかどうか(妥当性)について2012年3月に東京大学で実施された相互接続実験[8]で確認をおこなった。具体的には、実装したレジストリサーバに対し、他ベンダ(5 団体)とのコンポーネント-レジストリ間接続が可能であることを検証した。またレジストリに登録される優先度のパラメータを用いた複数ストレージに対するアクセス制御実験を行い、優先度に従ってストレージのコンポーネントのURIを応答すること、これによってストレージの高可用性が実現可能であることを確認した。この実験ではレジストリのサーバ機能(registration, lookup)について公の場で初めて動作確認を行い、registration及びlookupメソッドの仕様の妥当性についても示すことが出来た。なお、相互接続実験では最終的にレジストリにコンポーネント情報、コンポーネント情報の配下に管理されているポイント情報、ポイント情報あわせて1820件のデータが管理され、そのデータファイルの大きさは14.0MBになったことを確認した。

## 4.3 ゲートウェイの実装

宅内のゲートウェイについては図10のような構成で設計・実装を行った。

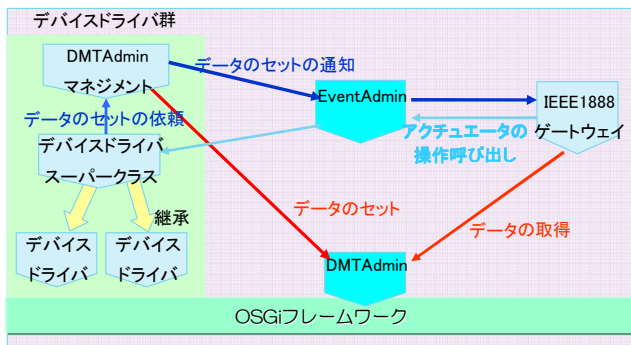


図10. 宅内ゲートウェイのバンドル構成

デバイスドライバとIEEE1888間のインターフェースとして、本検証ではOSGiフレームワークの標準バンドルであるDMTAdminバンドルとEventAdminバンドルを用いた。またデバイスドライバ側とのそれらの標準バンドルとの通信はデバイスドライバのスーパークラスとDMTAdmin マネジメントバンドルが仲介するように設計を行った。これによってデバイスドライバの開発者は、OSGiフレームワークおよびIEEE1888に関する知識がなくてもスーパークラスを継承するという制約だけでデバイスドライババンドルの開発が行えるようになる。また共通ライブラリを用いることで、IEEE1888以外のプロトコルをプロキシするようなバンドルを追加した場合でも共通で使用できるパラメータを一元管理することが可能となる。

センサとなる機器は種類によって「自ら値を送信するセンサ」と「問い合わせがあった場合に応答するセンサ」の

2種類がある存在することがわかった。前者については、データが送られてくるタイミングでWRITE手順(dataメソッド)にてデータを送信するとともに最新値をゲートウェイ内で保持しておけばFETCH手順(queryメソッド)にも対応できる。一方、後者は、そのままではWRITE手順にてデータを送信することが出来ない。そのため本研究ではIEEE1888ゲートウェイバンドル内部にタイマを実装し、そこから一定時間ごとにアクチュエータに対する操作と同様の手法(EventAdminバンドルによるアクチュエータ操作呼び出し)でデータを取得し、そのデータをWRITE手順でストレージに送信するように実装した。この実装方法によって、要求項目に必要なすべてのメソッドに対応できることが確認できた。また、デバイスドライバ追加時にゲートウェイバンドル及びDMTAdmin マネジメントバンドルの変更や追加が発生しないことを確認した。

## 4.4 コンポーネントの実装量

レジストリ、ストレージ、ゲートウェイのOSGiフレームワーク上でのそれぞれの実装量について確認を行い図11の結果を得た。

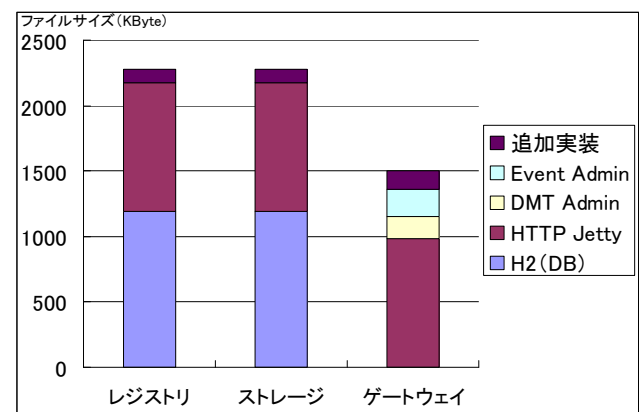


図11. コンポーネント及びレジストリのファイルサイズ要求条件を満たすためのコンポーネントについては標準ライブラリも含めてそれぞれ3MB以内で実装できることを確認した。特にゲートウェイの実装であれば1.5MB程度で実装可能であることを確認した。なお、本検証で用いたスマートタップ及びZigBeeのデバイスドライババンドルについてもそれぞれ16.4KB、20.2KBとなっていることを確認した。またqueryメソッド及びdataメソッドの開発ライン数はそれぞれ2.9KL、2.7KLであることを確認した。queryメソッドについては相互接続実験及びストレージのデータ蓄積を行うことが出来るための要件である最新値のデータ取得のみの実装であったが、その実装量はdataメソッドを上回っていることが分かった。またレジストリ及びストレージの機能についても基本的な実装であれば宅内で動作させることが出来る実装量であることが分かった。

## 4.5 考察

宅内の機器を適切に管理し、適切なストレージにデータを蓄積するための4つの方法についてはすべて実現可能で

はあることが分かった。また実際に検討・検証を行うことで以下のような知見を得ることが出来た。

- data メソッドよりも query メソッドのほうが電文にバリエーションがあるため(時間範囲を指定した連続データの取得やコンポーネント配下にある複数データの一括取得など)、より正確な実装を行うことは難しく、実装量がさらに増加することが予想される。
- FETCH 手順で宅内のゲートウェイからデータを取得する場合、ゲートウェイに対して適切なタイミングでデータ取得を行わなければ、取得できるはずのセンサーデータを取り損ねる場合や、不必要な通信が発生する可能性がある。
- ゲートウェイが適切なストレージを判断する方法では、その判断するロジックをホームゲートウェイ内に実装する必要があるため、実装量が大きくなる。またそのルールをゲートウェイに設定するため、変更時に宅内のゲートウェイすべてに更新をかける必要があるために影響が大きくなると推測される。

本検証の範囲では、ストレージもしくはレジストリでデータ蓄積ルールを決定するためのロジックを保持し、ゲートウェイからの WRITE 手順によってデータを蓄積する手法がよいと考えられる。なぜなら宅内のゲートウェイで query メソッドが不要になるために実装量の削減が見込めるためである。一方、その他の方法についても十分実現可能であることが確認できたため、用途や想定するシステムによっては適切な選択になりうると考えられる。

レジストリ及びストレージについても基本的な通信であれば宅内でも動作させることが可能である一方、管理・保持するデータ自体のサイズは大きくなる可能性があるため、それらのデータの一部はインターネット上で管理するなどの工夫が必要となってくると考えられる。

## 5. まとめ

IEEE1888 を用いたスマートコミュニティにおける宅内機器管理の一例として、宅内のゲートウェイとセンタのストレージを連携して容易に管理、接続する方法について検討を行った。IEEE1888 のレジストリ機能について実装を行い、その妥当性を確認することができたと同時に、実環境を想定したホームゲートウェイ上と同様の環境で検証を行うことで、スマートコミュニティのシステムの基盤が現在の環境で実現可能であることを示すことができた。一方 IEEE1888 は施設などの閉じられたネットワークエリアで利用されることを基本的には想定しているため、インターネットというオープンなネットワーク上で実現するにはアクセスコントロールやプライバシーなどのセキュリティに関連する課題もあり、これについては現在検討がされ始めている段階である。この課題の一部はレジストリを用いる

ことで解決できると考えられるため、セキュリティ問題を解決するためのレジストリ活用方法についても今後検討していきたい。

## 参考文献

- 1) パナソニック株式会社:“ライフィニティ”, <http://denko.panasonic.biz/Ebox/densetsu/lifinity/>
- 2) 株式会社東芝: “FEMINITY(フェミニティ)”, <http://feminity.toshiba.co.jp/feminity/>
- 3) ASHRAE BACnet: <http://www.bacnet.org/>
- 4) IEEE 1888: <http://standards.ieee.org/index.html>
- 5) 落合秀也: “Sensor Data Management and Transportation over Unreliable Networks”, 東京大学大学院情報理工学系研究科博士論文, Feb. 2011.
- 6) Digital Living Network Alliance(DLNA):<http://www.dlna.org/home>
- 7) エコーネットコンソーシアム, <http://www.echonet.gr.jp/>
- 8) 東大グリーン ICT プロジェクト, “<http://www.gutp.jp/>”
- 9) Open Service Gateway Initiative, <http://www.osgi.org/>
- 10) 川村, 前大道, 山崎, 森.: “OSGi(Open Service Gateway Initiative)の標準化動向について”, NTT 技術ジャーナル, Vol.19, No.11, pp.49-53, 2010
- 11) 佐藤さわ子, 梁田龍治, 前大道浩之.: “ホームネットワーク内機器同定方式の検討”, 電子情報通信学会技術報告, ICM2010-69, pp.87-92, March 11 2011
- 12) 美原義行, 山本隆二, 佐久間聡, 山崎毅文, 佐藤敦.: “ネットワーク接続機器の機器名特定システムの開発” 研究報告, コンシューマ・デバイス&システム (CDS), 2012-CDS-3(11),1-8 (2012-01-12)
- 13) 東日本電信電話株式会社 “フレッツ・ジョイント”, <http://flets.com/joint/>