

異種デバイス連携基盤を用いた センサ・家電制御アプリケーション

伊藤 崇洋^{†1} 加藤 悠一郎^{†1} 峰野 博史^{†2}
石川 憲洋^{†3} 水野 忠則^{†4}

本論文では、異なるネットワーク上に点在するセンサデバイスや家電機器などを Web を通して統一的に制御することの出来るセンサ・家電制御アプリケーションについて述べる。本アプリケーションは Web サーバー上に実装しており、ユーザはスマートフォンのブラウザなどを通して手軽に利用することが可能である。また、本アプリケーションはセンサデータを閲覧したり家電機器を操作するのみならず、あるセンサの値がある値を示した時に特定の家電を動作させる、といった、デバイスの連携機能も有するものとなっている。

Web application for controlling sensors and home appliances

TAKAHIRO ITO,^{†1} YUICHIRO KATO,^{†1} HIROSHI MINENO,^{†2}
NORIIHIRO ISHIKAWA^{†3} and TADANORI MIZUNO^{†4}

In this paper, we describe web application which can control the heterogeneous sensors and home appliances. Sensors and home appliances can be in a different network. This application is implemented on a WebServer, and user can use the application through the web browser easily. And, this application can be able to not only for monitoring the sensing datas and controlling the home appliances, but also for cooperating the device behavior such as when sensor detects some value, control the appliances.

1. はじめに

近年、様々な種類のスマートフォンがリリースされ、爆発的な普及を見せている。北米や西欧などの先進国では、2010年第4四半期において新規携帯電話販売の半数がスマートフォンに移り変わっている、との調査結果もある。これらのスマートフォンには、カメラやGPSなどと共に、加速度、方位、照度、温度、磁界、圧力などを測定する事のできる各種センサデバイスが搭載され、様々なアプリケーションを生み出す要素の1つとなっている。また、家庭内ゲーム機においても、WiiリモコンやPS Move、Kinectなどセンサを利用した様々なものがリリースされ、センサが我々のより身近なものとなりつつある。このように、徐々に利用されるシーンの増えているセンサであるが、現在利用されているセンサの多くはそのセンサの値を利用するデバイスとハードウェア的に接続されている場合が依然として多く、特定の場所における特定のアプリケーションの利用に特化したものとなっている。近年、環境モニタリングやホームセキュリティ、ヘルスケアなどを目的としてセンサネットワークに関する研究が盛んにおこなわれているが、この点に関しては同様である。センサネットワークは個々のセンサに無線技術を搭載しネットワークを形成したものであるが、そのネットワークは省電力や通信性能などを考慮して独自の通信・制御プロトコルを使っているものが多く、本来の目的以外の用途にも使うのは難しい。しかし、もし、それぞれのセンサネットワークを統一的に扱うことが出来るようになれば、センサ自身の有効活用に繋がるとともに、扱えるセンサ数の増大によって、信頼性の向上や応用アプリケーションの開発などにより質の高いサービスの提供が可能になると考えられる。

そこで我々は、これまでの研究¹⁾²⁾³⁾を基に開発した異種デバイス連携基盤技術を用い、各種センサデバイスや家電機器に対するユーザのアクセスポイントを Web に置いた新たなセンサ・家電アプリケーションを提案する。本アプリケーションでは様々な種類のセンサや家電機器の相互接続を異種デバイス連携基盤が担い、この機能を搭載した Web サーバーに

^{†1} 静岡大学大学院 情報学研究科

Graduate School of Informatics, Shizuoka University

^{†2} 静岡大学 情報学部

Faculty of Informatics, Shizuoka University

^{†3} 駒澤大学 グローバルメディアスタディーズ学部

Faculty of Global Media Studies, Komazawa University

^{†4} 静岡大学大学院 創造科学技術研究部

Graduate School of Science and Technology, Shizuoka University

においてそれらの機器への統一的なアクセスをユーザに提供する。ユーザは Web（本論文では特にスマートフォンのブラウザ）を通してセンサや家電にアクセスすることが出来、それらの機器の持っている情報を入手したり、あるセンサがある値を示した（イベント発生）時に特定の家電を動作させる（アクション）といった、イベントとアクションの関連付けを設定することも可能である。

以下、本論文の構成を示す。第 2 章では関連研究を紹介し、第 3 章にてシステムの提案を行う。そして、第 4 章で提案したシステムの実装について述べ、第 5 章でまとめをおこなう。

2. 関連研究

2.1 スマートハウス情報活用基盤整備フォーラム

スマートグリッド及びサービスまでを含めた社会システム（スマートコミュニティ）の普及を目指した取り組みとして、スマートコミュニティ・アライアンス⁴⁾がある。その中のワーキンググループの 1 つであるスマートハウスワーキンググループの活動の場として家庭内における異種デバイス連携などに関する議論がおこなわれているのが、スマートハウス情報活用基盤整備フォーラム⁵⁾である。本フォーラムでは、日立や大和ハウス、Panasonic などが積極的に実証実験をおこなっているようであるが、各種デバイスをつなぐ仕組みは、利用する製品全てを Bluetooth や独自の短距離無線プロトコルなど統一のもので扱う、というアプローチのようである。また、ユーザによる独自のイベント・アクション設定などに関する考察はおこなわれていないようである。

2.2 PUC

現在の情報家電や AV 機器、携帯端末などは Bluetooth や DLNA、WiFi などの共通の物理層プロトコルを利用していても、上位層のプロトコルの違いからお互いに通信を行うことはできない。このような問題を解決するための取り組みの 1 つとして、PUC(P2P Universal Computing Consortium)⁶⁾がある。PUC は Bluetooth や DLNA などの既存ネットワークの上にオーバーレイネットワークを形成することで様々なネットワークに存在する機器を相互接続・運用可能にしており、そのために必要なプロトコルとメタデータの仕様定義をおこなっている。PUC の仕様はオーバーレイネットワークにより異種ネットワークをつなぐため、既存のネットワークを統一する必要はないというメリットがある。

本論文では PUC の技術を利用して、異種デバイス連携環境の構築をおこなう。詳しい内容は、次章にて記す。

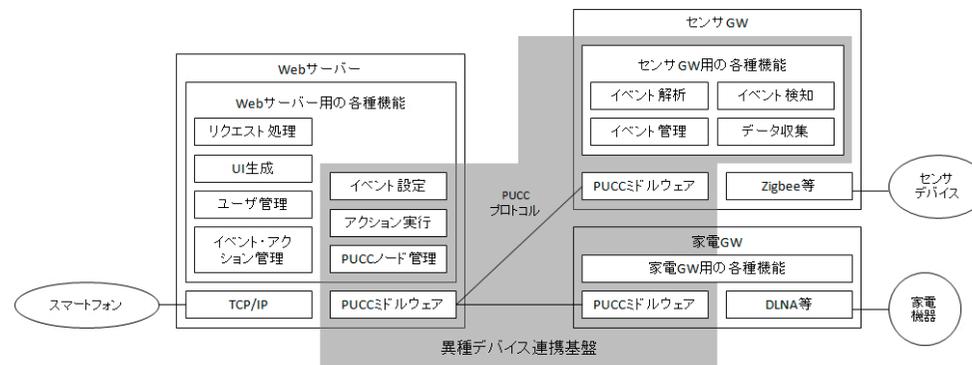


図 1 システム全体図

2.3 PIAX

P2P プラットフォームを用いてオーバーレイネットワークによってデバイスを接続可能とする技術として、PIAX⁷⁾もある。PIAX では、それぞれのデバイスが PIAX の P2P エージェントに接続することによって PIAX の P2P マルチオーバーレイネットワークに参加する。PIAX の P2P ネットワークには LL-Net や SkipGraph が用いられており、一般的な P2P よりも効率的に相手のピアを探索することが可能である。ユーザは PIAX のネットワークを通して、エージェントに接続されているセンサの値やストリーミングコンテンツ、口コミ情報などを取得することができる。現在は実証実験が盛んにおこなわれており、本システムの検証がおこなわれている。

3. 提案するシステム

まず初めに、Bluetooth や DLNA など異なるネットワーク上に存在する各種デバイスを統一的に扱えるようにするために必要な異種デバイス連携基盤について述べる。その後、この異種デバイス連携基盤を用いて異種ネットワーク上のデバイスの相互接続性を保障した上で、ユーザに対して「あるセンサがある値を示した（イベント発生）時に特定の家電を動作させる（アクション）」といった設定や、実際に設定されたイベント・アクションの定義に基づいて実行された結果などを通知するためのインターフェースを提供するセンサ・家電制御アプリケーションについて述べる。提案するシステムの全体図を図 1 に示す。



図 2 Pucc プロトコルスタック

3.1 異種デバイス連携基盤

異種デバイス連携基盤の中核は、第 2 章にも述べたように Pucc の仕様に基づいたものである。Pucc では相互に通信をおこなうためのプロトコルとメタデータを定義している。Pucc プロトコルスタックは図 2 のようになっており、IP や Bluetooth, Zigbee, IrDA など下位の通信プロトコルの上に経路制御やメッセージ制御をおこなう Pucc コアプロトコルを定義している。コアプロトコルの上にシステムごとに必要なメッセージを適宜定義しており、本論文ではセンサの値を用いて家電機器を操作することを想定しているため、そのために必要な各種メッセージについて次節にて述べる。これらの仕様に基づいて Pucc プロトコル及び各種機能を実装したプログラムを Pucc ミドルウェアと呼び、Pucc ミドルウェアを搭載したノードを Pucc ノードと呼ぶ。異種デバイス連携基盤上の各種ノードは全て Pucc ノードであり、センサを扱うためのノードならセンサのデータを収集する機能を Pucc ミドルウェアと共に実装し、センサゲートウェイ（センサ GW）と呼ぶ。また、家電機器を扱うためのノードなら家電機器を操作するための各種機能を Pucc ミドルウェアと共に実装し、家電制御ゲートウェイ（家電 GW）と呼ぶ。これら各種 GW を用いて異種ネットワーク上に存在する各種デバイスを相互に接続する仕組みをまとめて、異種デバイス連携基盤と呼ぶ。

3.1.1 Pucc メタデータ

Pucc メタデータは Pucc ノードに関する各種情報を記載した XML ファイルである。(図 3) Pucc ミドルウェアでは、このメタデータの送信・受信及び解析をすることによって、デバイスやサービスの探索をおこなっている。各要素の内容は以下のとおりである。

```
<Device type="URI of device type" id="Device ID" name="Device name">
<Specification>
<URLBase> Device's URI </URLBase>
<Manufacturer> manufacture name </Manufacturer>
:
</Specification>
<StateVariableList>
<StateVariable name="name" datatype="Data type" sendEvents="yes/no">
<DefaultValue> Default value </DefaultValue>
<AllowedValueList>
<AllowedValue> Value </AllowedValue>
</AllowedValueList>
</StateVariableList>
<ServiceList>
<Service name="Service name" type="URI of Service" />
<InputParameterList>
<Parameter name="Name of Input Parameter" type="Data type" />
</InputParameterList>
<OutputParameterList>
<Parameter name="Name of Output Parameter" type="Data type" />
</OutputParameterList>
</Service>
:
</ServiceList>
<EventConditionList>
<EventCondition id="Event ID">
<ConditionExpression> Event Condition Expression </ConditionExpression>
<NotificationVariableList>
<StateVariable name="Name of the function to be used" />
</NotificationVariableList>
</EventCondition>
:
</EventConditionList>
<PrimitiveDeviceList>
<PrimitiveDevice type="URI of device type" id="Device ID" name="Device name">
</PrimitiveDevice>
:
</PrimitiveDeviceList>
</Device>
```

図 3 Pucc メタデータ

- **Specification**
デバイスの名前や位置、製造番号などのデバイス自身に関するスタティックな情報を記載する。
- **StateVariable**
電源や動作モードなどデバイスの状態情報を記載する。
- **ServiceList**
イベント検知などのデバイスが提供可能なサービスの内容をリスト上に記載する。また、DVD デッキなど家電機器のメタデータの場合には、提供可能なサービスとして、録画、再生、早送りなどの機能を記載する。
- **EventConditionList**
センサデバイスのメタデータにのみ記載する。家庭内において温度を検知するデバイス

であった場合、50 度以上、といった明らかに異常と考えられる値を記述しておくことで万が一の事態に備える。

- **PrimitiveDeviceList**

デバイスがサブデバイスを保持していた場合、そのサブデバイスの要素をこの PrimitiveDeviceList 内に記載する。例えば、センサ GW が管理するセンサネットワーク上のデバイスはサブデバイスにあたるため、この要素内にセンサネットワーク上の各デバイスの情報を列挙する。

3.1.2 PUCC メッセージ

PUCC プロトコルには主に以下の 4 つのメソッドがあり、PUCC ミドルウェアを搭載し提供される API を利用することで以下の機能が実現できる。

- **DeviceDiscovery**

デバイスの探索に用いるメソッドである。探索条件を指定することで指定された条件にあったデバイスを探索することができる。デバイス探索をおこないたいノードは PUCC ネットワーク全体に DeviceDiscovery メッセージを送信し、メッセージを受けとったノードが自分が検索条件に合う端末であった場合にはメタデータを返す。

- **Subscribe**

ノード間で購読関係を結ぶために使用するメソッドである。センサ GW に対して Subscribe メッセージを送信した場合、イベントの登録をおこなうことができる。

- **Notify**

イベントの通知をおこなうメソッドである。実際に設定されているイベントが発生した際に、Subscribe 送信元にその事を知らせる場合などに用いる。

- **Invoke**

サービスもしくはアクションを実行する際に用いるメソッドである。Invoke 送信元は実行したいサービスをメッセージ内に記述し送信することで、このメッセージを受け取ったノードは指定されたサービスを実行する。

3.1.3 センサ GW

センサ GW は、センサデータを収集および管理する機能を有した PUCC ノードである。本論文では、センサデータの蓄積をこの GW 上でおこなう。また、Subscribe メッセージによって指定されたイベント条件式を解析し、判定をおこなうのもこの GW である。イベント条件の判定には、過去の研究成果²⁾ を利用する。

3.1.4 家電 GW

家電 GW は、家電機器を操作するための機能を有した PUCC ノードである。Invoke メッセージが送られてきた場合、指定されたデバイスの指定された機能を実行する。

3.2 センサ・家電制御アプリケーション

センサ・家電制御アプリケーションは PUCC ノードに Web サーバーの機能を持たせ、ユーザに対して各種センサデバイスや家電機器を制御するためのアクセスポイントとなるインタフェースを提供するアプリケーションである。また、一般的な Web サービスと同様、複数のユーザの情報を管理し、ユーザ独自のイベント・アクション定義などを管理する機能を有する。

センサ・家電制御アプリケーションの主要な機能は以下の 8 つである。

- **リクエスト処理**

ユーザデバイスからのリクエストを解析し、処理をおこなう。

- **UI 生成**

ユーザデバイスに表示するインタフェースを生成する。

- **ユーザ管理**

一般的な Web サービス同様、本アプリケーションにおいてもユーザ情報を Web サーバー側で持つ。

- **イベント・アクション管理**

ユーザが独自に設定したイベント・アクションの関連付けを管理する。本機能を用いることによって、イベント発生時に実行するアクションを決定する。また、上記のユーザ管理機能と連携して、イベント発生時に実行するアクションの排他制御を実現する。

- **イベント設定**

ユーザからの入力に従い、指定されたデバイスを有するノードに対して Subscribe メッセージの送信をおこなう。また、その際送信する Subscribe メッセージの生成もおこなう。

- **アクション実行**

ユーザからの入力に従い、指定されたデバイスを有するノードに対して Invoke メッセージの送信をおこなう。また、その際送信する Invoke メッセージの生成もおこなう。

- **PUCC ノード管理**

定期的に PUCC ネットワークに対して DeviceDiscovery メッセージを投げ、ネットワーク内に存在する PUCC ノードの情報を収集する。本機能によって集められ、管理され

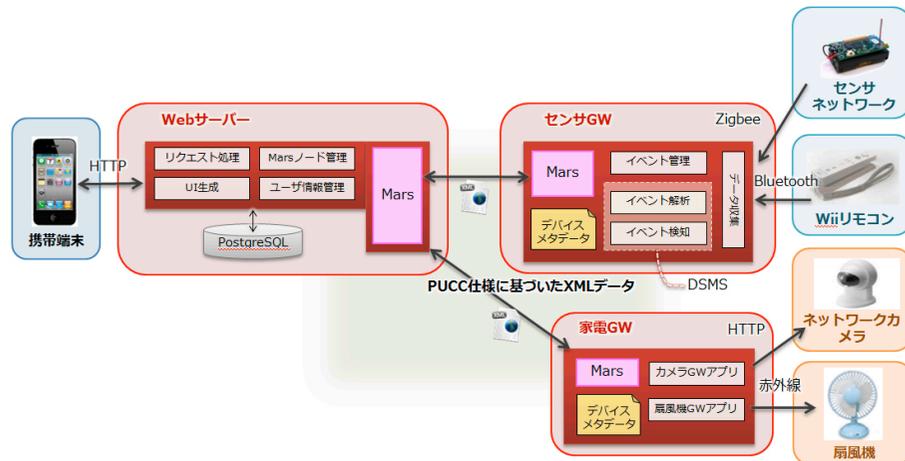


図4 全体図

ている情報を基に、ユーザに対して利用出来るデバイスの情報を提供する。

- ユーザに提供する情報の蓄積
イベント発生時に送られてきた Notify メッセージを解析し、ユーザにイベントが発生したことを通知するとともに、ログとして保管する。

4. 実装

本論文では実際に、異種デバイス連携基盤を用いてセンサ・家電制御アプリケーションを実装した。全体の構成図を図4に示す。

4.1 利用したデバイス

- ユーザデバイス：iPhone4
- センサデバイス1：Renesas Solutions 製のセンサボード
- センサデバイス2：Wii リモコン
- 家電機器1：Sony 製のネットワークカメラ (SNC-P5)
- 家電機器2：扇風機
- Web サーバー：MacBook Pro
- センサ GW：Windows XP 搭載のノート PC
- 家電 GW：Windows XP 搭載のノート PC



図5 ユーザインタフェース

4.2 PUCC ミドルウェア

本論文では、Mars という PUCC 仕様に基づいた独自の PUCC ミドルウェアを開発した。基本的には PUCC 仕様に準拠したもののだが、センサ及び家電機器の連携に必需なものだけに機能を削減している。言語は Java である。

4.3 センサ GW・家電 GW

共に、これまでの研究成果²⁾を用いて、新たに Mars に対応するよう改良したものである。第3章で述べた各種機能を実装している。

4.4 Web サーバー

Web サーバーの構成は、以下のとおりである。

- Apache Tomcat 6.0
- SAStruts 1.0.4sp8
- PostgreSQL 9.0
- Mars (PUCC middleware)

Super Agile Struts(SAStruts)は、現在、Java 言語における Web フレームワークとしてもっともよく利用されているフレームワークである Struts を完全にカプセル化し、開発効率の向上をはかったフレームワークである。

また、ユーザインタフェースは iPhone の Safari を使用し、ajax (json 形式) を用いて表示している。

- HTML5
- CSS3
- JavaScript (jQuery を含む)

これらによって、実際に表示されるユーザインタフェースのプロトタイプを図 5 に示す。現在、これらの画面を用いて、デバイスの探索、単純なイベント・アクションの設定、イベント発生時における通知などが実現出来ている。

5. おわりに

本論文では、異なるネットワーク上に点在するセンサデバイスや家電機器などを Web を通じて統一的に制御することの出来るセンサ・家電制御アプリケーションについて述べた。本アプリケーションは Web サーバー上に実装しており、ユーザは図 5 のような画面を通して手軽に利用することが可能である。また、本アプリケーションはセンサデータを閲覧したり家電機器を操作するのみならず、あるセンサの値がある値を示した時に特定の家電を動作させる、といった、デバイスの連携機能も有するものである。

今後は、Web サーバーにおける機能を API として公開し、iPhone アプリなどから本論文における各種機能を利用することを考えたい。また、その際における性能評価もおこないたいと考えている。さらに、センサ GW 自身の機能にも着目し、前後状態を考慮したりヘルスケアセンサにおけるイベント判定など、複雑なイベント条件への対応を進めていきたい。

参 考 文 献

- 1) 加藤悠一郎, 峰野博史, 角野宏, 石川憲洋, 水野忠則: ”携帯電話を用いた異種ネットワークデバイス連携システムの開発”, 2010-UBI-25, 2010.3.
- 2) Yuichiro Kato, Takahiro Ito, Hideki Kamiya, Masatoshi Ogura, Hiroshi Mineno, Norihiro Ishikawa, and Tadanori Mizuno: ”Home Appliance Control Using Heterogeneous Sensor Networks”, CCNC2009, 2009.1.
- 3) 伊藤崇洋, 小倉正利, 神谷英樹, 峰野博史, 石川憲洋, 水野忠則: ”携帯端末から制御可能なセンサ・家電連携環境の構築”, DICO2008, 2008.7.
- 4) スマートコミュニティ・アライアンス <http://www.smart-japan.org/>
- 5) スマートハウス情報活用基盤整備フォーラム <http://www.jipdec.or.jp/dupc/forum/eships/>
- 6) PUCC <http://www.pucc.jp/>
- 7) 吉田幹, 寺西祐一, 春本要, 下條真司,: ”マルチオーバーレイと分散エージェントの機構

を統合化した P2P プラットフォーム PIAX”, 2006-DPS-128, pp.43-48, Sep.2006.
8) SAStruts <http://sastruts.seasar.org/>