

クラウド型 HNS における家電制御サービスの検討

鷹取 敏志[†] 梶本 真佑[†] 佐伯 幸郎[†] 中村 匡秀[†]

[†] 神戸大学 〒657-8501 神戸市灘区六甲台町 1-1

E-mail: †takatori@ws.cs.kobe-u.ac.jp, ††{shinsuke,masa-n}@cs.kobe-u.ac.jp, †††sachio@carp.kobe-u.ac.jp

あらまし ホームネットワークシステム (HNS) とは、宅内の家電やセンサ等の機器をネットワークに繋ぎ、より便利で快適なサービスをユーザに提供するシステムのことである。従来の HNS では、HNS を構成する各種機器と HNS サービスが密に結合しているため、ユーザが自由に機器やサービスを選択することが困難である。我々は先行研究として、クラウドのコンセプトに基づき機器とサービスの疎結合を実現する新たな HNS 形態、クラウド型 HNS を提案している。クラウド型 HNS では、家電機器の操作やセンサ値の取得、マスタ情報の管理といった HNS サービスを実現するための基本的な機能を、ベンダに依存しない標準的なサービスとして提供する。本稿では、クラウド型 HNS の提供する基本サービスの内、宅内の家電を制御するサービス (Application Control as a Service: ACaaS) について検討する。具体的には、ACaaS を構成する階層アーキテクチャ、および家電機器とその制御方法の対応を動的に発見するためのデータモデルについて考察する。

キーワード ホームネットワークシステム (HNS)、クラウドコンピューティング、家電制御、ACaaS: Application Control as a Service、階層アーキテクチャ

Considering Appliance Control Service in A Cloud-Enabled Home Network System

Satoshi TAKATORI[†], Shinsuke MATSUMOTO[†], Sachio SAIKI[†], and Masahide NAKAMURA[†]

[†] Kobe University Rokko-dai-cho 1-1, Nada-ku, Kobe, Hyogo, 657-8501 Japan

E-mail: †takatori@ws.cs.kobe-u.ac.jp, ††{shinsuke,masa-n}@cs.kobe-u.ac.jp, †††sachio@carp.kobe-u.ac.jp

Abstract Home network system (HNS) provides various value-added services for home users by connecting household appliances and sensors. In the conventional HNS architecture, HNS appliances and services are tightly coupled. This coupling problem leads to difficulty for users to freely choose their favorite appliances and HNS services. In our previous work, we have proposed a novel HNS architecture, named Cloud HNS, that accommodates multi-vendor services by building on the concept of cloud computing. Cloud HNS provides standard and basic services such as appliance operation, environmental log collection and static data management. In this paper, we discuss and focus on the appliance operation, named Appliance Control as a Service (ACaaS), which is one of fundamental services provided by Cloud HNS. Specifically, we discuss about multilayered architecture of ACaaS and design of a data model for dynamically discovering a relationship between appliance and its controlling method.

Key words Home network system (HNS), cloud computing, appliance operation, ACaaS: Application Control as a Service, multilayered architecture.

1. はじめに

家電やセンサを家庭内のネットワークに接続することで、多種多様なサービスを提供するホームネットワークシステム (HNS) に注目が集まっている。HNS では、宅内に設置されたテレビやエアコン、扇風機などの家電機器、および温度計や湿度計などのセンサ機器をネットワークに接続することで、複数機器の連

携制御 [1] や、状況に応じたコンテキストウェアサービス [2] 等の付加価値サービス (以降、HNS サービスと呼ぶ) を提供することが出来る。

これまで長期にわたって研究開発が進められてきた HNS であるが、一般家庭への幅広い普及には至っていない。その要因の一つとして、現状の HNS では、HNS の構成要素 (家電機器やセンサ、ホームサーバ、通信プロトコルなど) と、ベンダが

提供する HNS サービスとが密結合していることが考えられる。この密結合のため、単一宅内において複数のベンダが提供する HNS サービスを同時に利用することは容易ではない。HNS 環境を制御するホームサーバを複数設置することで、マルチベンダ HNS 環境を実現することも可能ではある。しかしながら、この方法は導入コストが高く、また異なるベンダの HNS サービス間で特定の家電機器を共有することもできない。これらの問題により、ユーザは購入家電の選択肢が限られるのみならず、利用する HNS サービスの選択肢も限られる。その結果、HNS の導入が進まない状況に陥っている。

我々は現状の HNS の問題を解決するために、先行研究 [3] において、クラウド型 HNS を提案している。クラウド型 HNS とはクラウドのコンセプトに基づいた HNS の新たな形態である。クラウド型 HNS では、ホームサーバが実行していた HNS サービスの様々な機能（家電制御やセンサ値の取得、ログデータの管理など）を全てサービスとして抽象化し、クラウド上に集約する。この集約により、HNS 構成要素とベンダ提供のサービスとの間の密結合を解消し、宅内でのマルチベンダ HNS 環境が実現出来る。また、各ベンダの提供する HNS サービスは、クラウド型 HNS が公開する HNS-API を利用して各家庭の家電を遠隔制御する。これにより、宅内の構成情報や機器制御、ログ管理といったベンダ毎に保持していた共通の機能やサービスを一元管理することが可能となる。

先行研究では、クラウド型 HNS 全体のコンセプト、およびアーキテクチャの提案に止まっており、その具体的な実現方法については論及していなかった。特に、クラウド型 HNS を構成するコンポーネントの一つであり、また最も中心的な役割を果たす **HouseCloud** に関する検討が重要な課題となる。HouseCloud は、1. 住宅に関する静的な構成情報の管理、2. 住宅内で発生するログ情報の収集・管理、および 3. ネットワーク経由での家電機器の制御、という特定の用途に依存しない 3 つの標準的なサービスを持つ。各サービス実現のためには、それぞれ管理対象となるデータのスキーマ設計やアーキテクチャ設計などについて、より具体的な設計・検討が必要である。

本稿では、クラウド型 HNS における HouseCloud が提供する機能の一つである、**機器制御サービス (ACaaS: Appliance Control as a Service)** に着目し、そのアーキテクチャとデータモデル設計について考察する。

ACaaS は役割に応じた 5 つのレイヤ構造で構成されており、機器操作をサービス化し操作 API を提供するレイヤや、具体的な機器制御信号を発光するレイヤなどを持つ。このようにレイヤを構造を構成することで、各階層での処理の粗結合化と単純化が可能となり、保守性や拡張性を持たせつつ、複雑な機器サービスを容易に取り扱えるようになる。

2. 準備

2.1 ホームネットワークシステム (HNS)

HNS とは、TV や DVD, エアコン, 扇風機などの家電機器、および温度計や照度計などのセンサ機器をネットワークに接続し、様々な付加価値サービスを実現するシステムである。ス

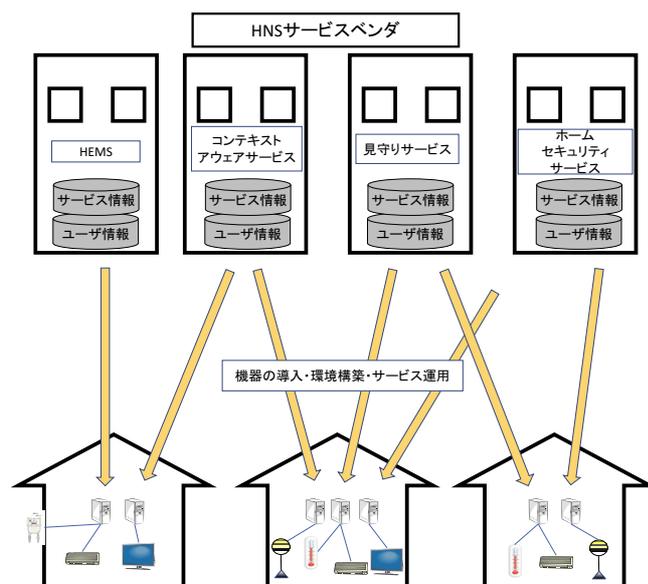


図 1 従来 HNS の全体アーキテクチャ

マートフォンを用いてシャッターを遠隔開閉するサービス [4] や、エアコンを自動的に節電運転するサービス [5] などが存在する。HNS は省エネや防犯、医療、介護等様々な分野での応用が期待されており、既にいくつかのサービスが市場に登場している。本稿では、これらの HNS 上で提供される付加価値サービスのことを **HNS サービス** と呼ぶ。

一般的な HNS のアーキテクチャを図 1 に示す。各家庭にはそれぞれホームサーバを中心とした HNS 環境が構築されており、家電機器やセンサ機器が宅内ネットワークに接続されている。これらの宅内機器をホームサーバからソフトウェア制御することで、HNS サービスは実現される。家電機器、およびセンサ機器の制御方法は HNS サービスベンダによって様々であるが、一般には赤外線や ZigBee [6] などの近距離無線規格や、イーサネット規格、ECHONET Lite プロトコルなどが利用される。HNS サービスベンダは、ホームサーバを経由して HNS サービスを提供する。HNS サービスの提供形態も様々であり、HNS サービスがプリインストールされたホームサーバ機器を販売する形態や、ネットワーク経由で HNS サービスをインストールする形態などが存在する。また、HNS サービス実行に必要な様々な情報（図中のサービス情報とユーザ情報）についても、ベンダ固有の環境で管理される。

上記の通り、HNS 環境をどのように実現するかは HNS サービスベンダに強く依存しており、HNS を構成する各種要素と HNS サービスは密に結合している。そのため、単一宅内の中に複数の HNS 環境を構築するようなマルチベンダ HNS の実現は容易ではない。図の場合、宅内にベンダ固有のホームサーバを複数導入することで、マルチベンダ HNS 環境を実現しているが、この方法は HNS 導入コストや管理コストの増加に繋がるほか、HNS サービス個々の情報が分散するという問題も発生する。

2.2 先行研究 [3]: クラウド型 HNS

クラウド型 HNS とは、従来 HNS の課題を解決するための

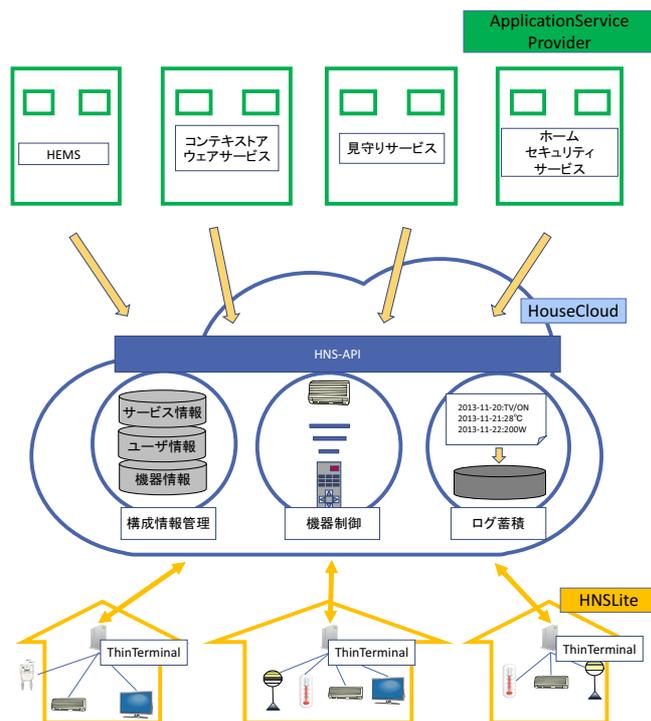


図2 クラウド型 HNS の全体アーキテクチャ

新たな HNS の形態である。HNS が提供する基本的な機能をサービスとして抽象化し、クラウド上で管理することで HNS の構成要素とサービスとの祖結合を図る。これにより、ユーザは手軽にマルチベンダの HNS サービスを利用することが出来るようになるほか、HNS 導入コストや管理コストの低減が見込める。

クラウド型 HNS の全体アーキテクチャを図 2 に示す。クラウド型 HNS は HNSLite, HouseCloud, Application-ServiceProvider の 3 つ要素から構成されている。

図 2 下段の HNSLite は従来の HNS 環境からホームサーバを取り除き、代わりに ThinTerminal が設置された軽量の HNS 環境である。ThinTerminal は HouseCloud からの命令に基づき、各家庭のログデータの取得や家電機器の実行を行う。ThinTerminal はインターネットと宅内ネットワークとのゲートウェイであり、宅内機器と HouseCloud を接続するための最低限の機能だけを持つ。これにより、宅内には物理空間との境界になる低レベルなデバイスしか存在しないことになる。従来のホームサーバで行われていた高級なサービスは、全てクラウド上で実行される。

HouseCloud (図中央) はクラウド型 HNS の中心的な役割を果たすレイヤである。HouseCloud は HNS サービスを実現するための基本的な 3 つの機能を持つ。

- **F1 構成情報の管理** : HouseCloud に接続される HNS 環境の静的な構成情報を管理する機能である。管理対象となる構成情報には、宅内の機器一覧や住民構成、利用しているサービスなど様々なマスタ情報が含まれる。
- **F2 家電機器の制御** : HNSLite に設置された宅内機器の操作を、抽象化したサービスとして ASP に提供する機能であ

る。HNS サービスからの機器操作リクエストを受信した場合に、操作対象の機器を特定し、具体的な操作コマンドに変換して HNSLite に指示する。

- **F3 住宅ログの収集** : HNSLite に設置されたセンサの情報や機器の状態ログを収集し、蓄積する機能である。膨大な量のログ情報を一元管理し、HNS-API 経由で HNS サービスにログ情報を提供する。

図 2 上段の ASP (Application Service Provider) とは、HNSLite 用の HNS サービスの開発・提供を行うプロバイダのことである。HNS サービスに必要な情報や操作は、HouseCloud の公開している HNS-API を通じて実行する。HNS ベンダに強く依存していた HNS 固有の機能や処理は、全て HouseCloud 上でサービスとして抽象化されるため、ベンダに依存せず HNS サービスの開発が可能となる。ユーザは HouseCloud に登録された様々なサービスから、自宅の HNS に適したものを自由に選ぶことが出来る。

2.3 本論文のScope

先行研究では、クラウド型 HNS 全体のコンセプトとアーキテクチャの提案に止まっており、その具体的な実現方法については論及していなかった。特にクラウド型 HNS 実現のためには、その中心となる HouseCloud の持つ 3 つの機能 (F1~F3) それぞれについて、より詳細な検討と設計が必須である。本稿では、F2 : 家電機器の制御に着目し、HouseCloud における機器制御サービスのアーキテクチャ、およびデータモデル設計について考察する。

3. 家電機器制御サービス (ACaaS: Appliance Controller as a Service)

3.1 キーアイディア

クラウド型 HNS において、各家庭で横断的な機器操作を実現するために、HouseCloud が提供する機器制御サービス (ACaaS: Appliance Controller as a Service) を検討する。各家庭内の機器制御をサービスとして抽象化し、HouseCloud において提供することで、HNS サービスと機器制御の疎結合化を図る。

機器制御サービスが家庭を横断して機器操作を可能にするために満たすべき要件は、ユーザ宅内 (HNSLite 内) にある機器の種類に関わらず、HouseCloud の提供する API から制御を可能とすることである。この要件を満たすために、機器制御サービスにおいて、機器を制御可能なコントローラを動的に解決する必要がある。コントローラとは各家庭の HNSLite 内に収容される、赤外線信号発光機や、ECHONET Lite 機器を制御するための信号を発信する機器を指す。各機器インスタンスは、機器自体を制御するためのコントローラと結びついている。機器の制御を実行する際に、動的にコントローラの解決を行うことで、どのような機器においても、同様のインタフェースから制御可能になる。また、各機器をコントローラと分離し、それらを自由に組み合わせることを可能にすることで、様々な機器を操作出来るようになる。

機器制御サービスにおけるコントローラの動的解決を実現す

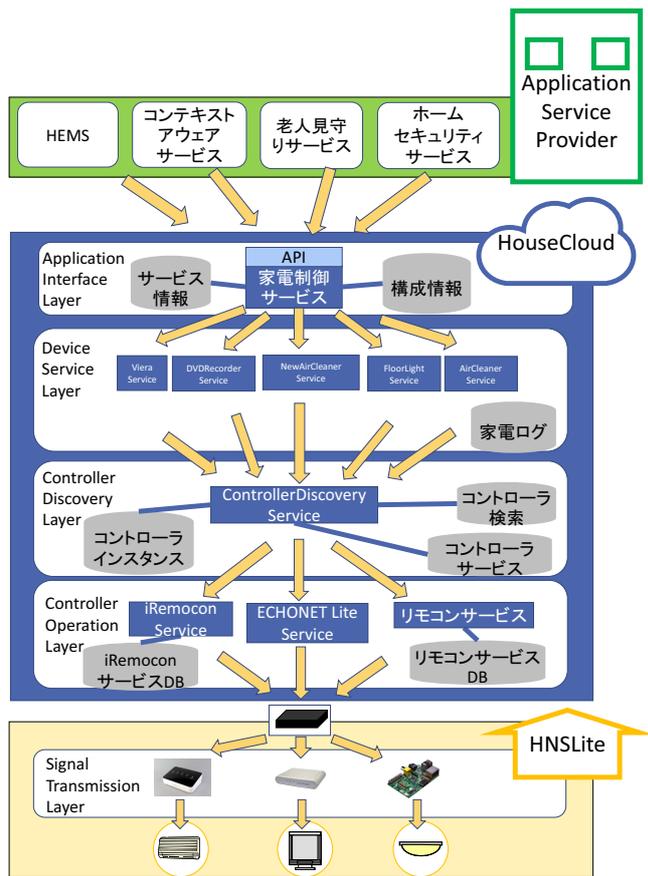


図 3 家電機器操作サービス (ACaaS) の階層アーキテクチャ

るための具体的な方法は、機器制御サービスの階層化である。機器制御サービスをインタフェース層、機器サービス層、コントローラ発見層、コントローラ操作層、信号発信層の、5つの階層に分離し、それぞれの階層において必要なパラメータを動的に解決する。インタフェース層において機器の ID 等の情報を解決し、機器サービス層でその機器特有の制御の特徴を吸収する、コントローラ発見層で機器に対応するコントローラを解決し、コントローラ操作層でコントローラに命令を送信、信号発信層でコントローラから信号を発信する。

これにより、コントローラや家電ベンダに依らずに、機器操作をサービスとして提供することが可能になる。また、コントローラの変更や機器の更新といった変化にも柔軟に対応可能になる。

3.2 アーキテクチャ

提案サービスのアーキテクチャを図 3 に示す。提案アーキテクチャはインタフェース層 (ApplicationInterfaceLayer)、機器サービス層 (DeviceServiceLayer)、コントローラ発見層 (ControllerDiscoveryLayer)、コントローラ操作層 (ControllerOperationLayer)、信号発信層 (SignalTransmissionLayer) の 5つのレイヤから構成される。以下で機器制御を実行する際の情報の流れを説明する。

ユーザや HNS サービスが機器制御を行う場合、ACaaS のインタフェース層が提供する API にリクエストを送信する。リクエストを受け取った ACaaS インタフェース層は、リクエスト

中のユーザ ID やサービス ID をもとに、HouseCloud 内に存在するユーザ宅内の機器構成情報 DB や、サービス情報 DB の情報を参照し、ユーザ認証および機器制御の許可判断を行う。また、制御する機器の ID 並びに機器クラスを取得し、機器サービス層のどのサービスに処理を引き継ぐか判断する。

次に、機器サービス層において、制御対象の機器に対応したサービスにより、機器ごとの制御の実装の違いを吸収する。細かな実装の違いをこの層で吸収することで、統一的なインタフェースの提供が可能になる。

その後、機器を実際に制御するために、機器と結びついているコントローラの解決を行う。これは、コントローラ発見層において行われる。コントローラ発見層では、機器 ID をもとに、その機器を制御するために必要な、コントローラ情報 (ID、種類、アドレス) を DB から取得する。また、取得したコントローラ情報から、コントローラを制御する為のサービスに関する情報 (必要パラメータの種類、エンドポイント) を取得する。そして、そのサービスに必要なパラメータを渡し、コントローラの制御を依頼する。

コントローラの解決が完了すると、コントローラ操作層で実際にコントローラを制御する命令を発行する。コントローラ操作層は様々な種類のコントローラを操作するサービス群から構成される。例えば、学習型リモコンである iRemocon を操作する iRemoconService、宅内の家電操作を実現するために、標準化が進められている、ECHONET Lite プロトコルを使用する ECHONETLiteService、我々の研究グループで開発中の RaaS (Remocon as a Service) などである。各サービスは、コントローラ操作層で解決された情報をもとに、HNSLite 内のコントローラインスタンスに信号送信命令を発する。

最後に、ユーザ宅内にあるコントローラインスタンスが受け取った信号送信命令に従い、実際に機器に対して信号を送る。

3.3 インタフェース層 (ApplicationInterfaceLayer)

インタフェース層は機器制御サービスの API を提供し、ユーザや HNS サービスから機器制御リクエストを受け付ける。HNSLite 内のあらゆる機器の制御は、全てこの API を通じて実行される。ユーザや HNS サービスが宅内の機器を制御する場合、制御対象となる機器に関する情報を含めたリクエストを API に送信する。インタフェース層では、まず、リクエストに含まれるユーザ ID や機器の種類に関する情報に基づき、HouseCloud のユーザ情報 DB に格納されている宅内の構成機器情報から、操作の対象となる機器 ID や機器クラスといった情報を取得する。ここで機器 ID とは、すべての機器に割り振られる一意の番号であり、機器クラスは、その機器の機種を特定するためのメーカー名や型番等の情報である。次に、取得した情報をもとに、制御対象の機器に対応する機器サービスを判断した後、操作名や機器 ID を渡し、処理を機器サービス層に引き継ぐ。

また、インタフェース層では、ユーザ認証および、HouseCloud 内の情報に対するアクセス権限、機器制御の実行権限の確認を行う必要がある。クラウド型 HNS では、宅内のあらゆる情報や、機器の制御をクラウド上で管理し、外部から利用出

来ようにする。このため、不正なアクセスにより、他人の家の情報を盗み取ったり、機器を無断で制御される可能性がある。従来サービスと異なり、影響が現実世界まで及ぶため、人体が危険にさらされる可能性があり、情報へのアクセス制御や、機器制御の実行判断が非常に重要になってくる。アクセス制限の方法は、ユーザが各サービスに使用可能な機器を設定することで実現する。予めユーザによってサービスごとに機器の使用権限を設定しておき、構成情報 DB で管理する。サービス利用中は、この設定情報をもとに、機器の制御リクエストが正常な権限に基づいたものであるかを逐次判断する。また、HouseCloud では、各 HNS サービスが利用する機器の種類や必要な情報種類等のサービス情報を管理している。ユーザはこの情報をもとに、サービスを利用するかを判断することが可能であり、自身の意図しないような機器の利用を防ぐことが出来る。

3.4 機器サービス層 (DeviceServiceLayer)

機器サービス層は機器クラスごとに作られた機器のサービスから構成される。機器サービスとは機器クラスごとに、つまり各機器の種類ごとに機器の機能を抽象化したサービスである。機器サービス層の目的は、機種ごとに異なる操作の実装を同一のインタフェースから利用出来るように、機種ごとのサービスで吸収することである。例えば、扇風機を操作する場合、風量を強にするために、ボタンを何回か押すトグル方式のタイプと、ボタン一つで操作出来るタイプがある。このような、同一の操作の実装上の違いを吸収することがこのレイヤーの役割である。上位層つまりインタフェース層では、機器クラスに基づき、各機器サービスに処理を割り振ることで、特定の機器の操作を同一の命令で実行可能にする。各機器サービスは機器を開発したベンダが作成し、HouseCloud に登録する。

その他に、機器サービス層では、指定された機器操作がその機種に存在するかの判定や機器操作のログ蓄積も行う。機器のログ情報はユーザの許可のもと HNS サービスに提供され、サービスの改善や実現に利用される。

3.5 コントローラ発見層 (ControllerDiscoveryLayer)

コントローラ発見層では、対象の機器を操作するためのコントローラの発見を行う。また、コントローラのアドレスや種類、コントローラを制御するサービスの解決も行う。各機器とコントローラの組み合わせの解決をこの層で行うことにより、どのような操作方法の機器でも制御可能になる。

コントローラ発見層ではコントローラ検索 DB、コントローラインスタンス DB、コントローラサービス DB の3つの DB を持つ。コントローラ検索 DB は機器とコントローラの対応を、コントローラインスタンス DB はコントローラのアドレスや種類等の情報を、コントローラサービス DB ではコントローラを制御するためのサービスに関する情報をそれぞれ保持している。

各 DB の設計情報を以下の図 4 に示す。上位層から、処理を任されたコントローラ発見層は、まず、コントローラ検索 DB を参照し、機器 ID をもとに、その機器を操作することが可能なコントローラの ID を取得する。例えば、機器 ID の tv001 を元に、コントローラ ID の c001 を取得する。次に、そのコントローラの ID を用いて、コントローラインスタンス DB か

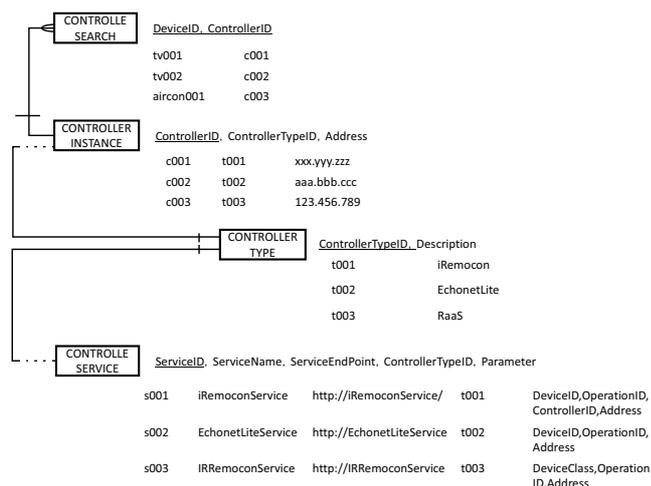


図 4 コントローラ発見層の DB

ら、コントローラの種類およびコントローラのアドレスを取得する。コントローラの種類とは iRemocon や ECHONET Lite 機器、RaaS 機器の家電機器に信号を発信出来るものを指す。コントローラのアドレスとはそのコントローラを一意に特定出来るものを指す。例えば、IP アドレスや MAC アドレスである。コントローラの特が完了した後、そのコントローラを制御するサービスに関する情報を解決する。コントローラの種類をもとに、そのコントローラを操作するサービスの詳細をコントローラサービス DB から取得する。コントローラサービス DB はサービス ID、サービス名、サービス URL、コントローラ Type、サービス実行に必要なパラメータの種類から構成されている。最後に、コントローラ発見層では、各コントローラサービスに必要なパラメータを渡し、機器の制御処理を引き継ぐ。

3.6 コントローラ操作層 (ControllerOperationLayer)

コントローラ操作層は機器の制御依頼を、コントローラの操作指示に変換し、コントローラの制御を行う。コントローラの種類ごとに、制御するためのサービスが設けられており、コントローラ操作層は、それらのサービスから構成されている。サービス例としては、iRemoconService, ECHONETLiteService, RaaS が挙げられる。各サービスはコントローラ発見層から渡されたパラメータをもとに、宅内のコントローラを実際に操作する命令を発する。各サービスは操作命令に必要な情報が格納された DB を独自にもつ。例えば、iRemocon であれば、各機器の操作ごとに赤外線信号とその信号番号が対応付けられているため、iRemoconService は、機器 ID、操作 ID、コントローラ ID、信号番号という情報を持った DB を持つ。機器を操作したい場合は、iRemoconService に必要な情報を渡し、iRemoconService が宅内の iRemocon に信号番号を送ることで、操作が可能になる。

3.7 信号発信層 (SignalTransmissionLayer)

信号発信層は、コントローラから信号を発信することで実際に機器操作を実行するレイヤである。特に、HNSLite 内のコントローラの実態のことを指す。各コントローラは HNSLite 内

の ThinTerminal に接続されており、HouseCloud から送られてくる命令は ThinTerminal を通じてコントローラに通達される。各コントローラはそれぞれアドレスを持ち、コントローラ操作層で決定された操作命令がそのアドレスの機器に送られる。信号発信層では、操作命令を受け取ったコントローラが命令を信号に変換して、あるいは送られてきた信号をそのまま、各機器に送信し、機器操作を行う。宅内の機器を操作可能な種類のコントローラを設置することで、様々な種類の機器を制御することが可能になる。例えば、ECHONET Lite に未対応な機器が存在していた場合、赤外線制御可能であるならば、コントローラを赤外線発光のものにすることで、容易に操作可能になる。また、信号情報をキャッシュとして ThinTerminal に保存したり、セキュリティ向上のために、暗号化のシステムの導入も必要となる。

4. 考 察

4.1 特 徴

ACaaS を利用するメリットとしては、以下の 2 点が考えられる。一つ目は、HNS 環境に対するコストの削減効果である。従来の HNS では、機器とサービスの結びつきが強く、HNS 導入の際に特定ベンダの機器を揃える必要があった。ECHONET Lite のように、家電制御プロトコルの標準化が進められているものの、ECHONET Lite 対応機器を設置する必要があり、新規 HNS 環境の設置コストは少なくない。また、機器操作のための API については標準化されておらず、機器制御 API と HNS サービスの結びつきは強い。ACaaS を使うことで HNS サービス開発者は、ユーザ宅内の機器を手軽に利用することが出来るようになり、サービス開発が活性化される。これにより、コストの低下も見込まれ、サービス料金の低下につながる。

二つ目は、コントローラの選択が自由になる点である。宅内に存在する機器に対応したコントローラを設置するだけで、機器制御が実現可能になる。ECHONET Lite 未対応機器でも赤外線可能ならば HNS に収容可能になる。また、複数の制御方式が混在していても、サービス側はその差異を意識することなくサービスを開発することが可能である。HNS の普及促進のためには、導入の敷居を低く、HNS サービスをすぐに使えるようにすることが重要である。ACaaS では、ECHONET Lite 機器を揃えることなく、HNS サービスを利用可能になる。

4.2 課 題

クラウド型 HNS においては、遠隔制御やセキュリティ、ネットワーク接続できない場合にサービスが利用できなくなるといった、クラウド固有の問題が常に存在する。まず、遠隔制御の問題について検討する。ACaaS では外部から機器の制御が可能となるため、ユーザが意図しない機器の遠隔制御が可能である。例えば、機器の誤操作や HNS サービスの設定ミスにより、エアコンの温度を危険な状態にまで上げるような操作が行われる可能性がある。一つの対応策としては、機器制御の実行時にその操作の安全性を検証するといった方法が考えられる。

次にセキュリティの確保手段について考察する。悪意のあるユーザへの対策としては、パスワード方式による単純な認証方

式に限らず、特定のハードウェアのみからアクセスを許可するといった方法が考えられる。一般に、セキュリティの高さと利便性はトレードオフの関係にあり、対象システムの性質に応じて安全性と利便性のバランスを考える必要がある。ACaaS は宅内機器の遠隔操作を許可するという性質上、高い安全性が求められるが、一方で日常生活と密接に結びついており同時に高い利便性も求められる。ACaaS でどのようにセキュリティを担保するかは、今後の重要な課題の一つである。

また、ネットワークに何らかの問題が発生した場合、ユーザが全く機器を操作できない状態に陥る可能性がある。対応策としては、全ての家電制御処理をネットワーク経由で実行するだけでなく、ThinTerminal を中心としたネットワーク不要のオフライン HNS モードを用意するという方法が考えられる。この方法は、宅内の高級なホームサーバを排除し導入コストを削減する、という ACaaS の要求と矛盾する部分もあるが、家電機器の On/Off などの日常的に利用する基本的な機器制御は常にオフラインでも利用出来るべきといえる。

5. おわりに

本稿では、クラウド型 HNS を実現するために、HNS における機器の基本操作を抽象化した ACaaS 実現のための検討およびアーキテクチャの提案を行った。提案アーキテクチャでは、機器操作サービスを 5 つの階層から構成する事により、様々な機器を家庭を横断して制御可能になる。今後の課題としては、まず、HouseCloud における機器制御サービスの実装を行う。その後、セキュリティや、HNS Lite との通信方法についての検討を行っていく予定である。

謝辞 この研究の一部は、科学技術研究費（基盤研究 C24500079, 基盤研究 C12877795, 基盤研究 B 23300009）、および、積水ハウスの研究助成を受けて行われている。

文 献

- [1] M. Nakamura, H. Igaki, H. Tamada, and K. Matsumoto, "Implementing integrated services of networked home appliances using service oriented architecture," Proceedings of the 2nd International Conference on Service Oriented Computing, pp.269–278, 2004.
- [2] 高塚広貴, 佐伯幸郎, 枡本真佑, 中村匡秀, "異種分散 web サービスに基づくコンテキストウェアサービスの管理フレームワークの提案," 電子情報通信学会技術報告, 第 113 巻電子情報通信学会, pp.1–6 Oct. 2013.
- [3] 鷹取敏志, 枡本真佑, 佐伯幸郎, 中村匡秀, "マルチベンダサービスを実現するクラウド型ホームネットワーク システムの提案," 電子情報通信学会技術報告, 第 113 巻電子情報通信学会, pp.53–58 Nov. 2013.
- [4] "TamaHome," <http://www.tamahome.jp/i-tamahome>.
- [5] "スマート HEMS," <http://www2.panasonic.biz/es/densetsu/aiseg/features/index.html>.
- [6] P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta, and Y. F. Hu, "Wireless sensor networks: A survey on the state of the art and the 802.15.4 and zigbee standards," Computer Communications, vol.30, no.7, pp.1655–1695, 2007.