TTC標準
Standard

JT-Y2070

HEMS とホームネットワークサービスの要件とアーキテクチャ

Requirements and architecture of home energy management system and home network services

第 1 版

2015 年 8 月 27 日制定

一般社団法人
情報通信技術委員会

THE TELECOMMUNICATION TECHNOLOGY COMMITTEE

TTC Telecommunication Technology Committee

目　次

<参考>

1. 国際勧告等の関連

本標準は、2015 年 1 月に勧告化が承認された ITU-T 勧告 Y.2070 に準拠している。

2. 上記国際勧告等に対する追加項目等

2. 1 オプション選択項目

特になし

2. 2 ナショナルマター項目

特になし

2. 3 原標準に対する変更項目

特になし

2. 4 その他

特になし

2. 5 現勧告との章立て構成比較表

上記国際勧告との章立て構成の相違はない。

3. 改版の履歴

版数	制定日	改版内容
第 1.0 版	2015 年 8 月 27 日	制定

4. 工業所有権

本標準に関わる「工業所有権等の実施の権利に係る確認書」の提出状況は、TTC ホームページで御覧になれます。

5. 標準策定部門

次世代ホームネットワークシステム専門委員会（ITU-T Y.2070 および本標準の策定）

NGN&FN(Future Networks)専門委員会（ITU-T Y.2070 の策定）

１　本標準の範囲

本標準はホームエネルギーマネジメントシステム(HEMS)とホームネットワーク(HN)サービスの機能要件とアーキテクチャを規定する。HEMS はエネルギーの効率化やエネルギーの消費量削減に貢献するアプリケーションであり、HN サービスアーキテクチャを利用して HN に接続された家電、蓄電池、センサー等のデバイスの内部状態を監視したり、制御したりする。同様に、ホームセキュリティやヘルスケア等の他の HN サービスも HEMS と同じアーキテクチャで提供可能であり、各アプリケーションで必要となるデバイスの内部状態を参照し、制御する。本標準は、HEMS や他の HN サービスを実現するための機能要件、参照アーキテクチャ、機能間の関係を含む機能アーキテクチャを記載する。

本標準は以下を含む。

・HEMS や他の HN サービスのための HN サービスアーキテクチャの概要
・HN サービスアーキテクチャにおけるデバイス、ホームゲートウェイ(HGW)、管理プラットフォーム(PF)の機能要件、アーキテクチャに要求されるセキュリティ要件
・デバイスのタイプに合わせて、4 通りのデバイスと HGW との接続方法を持つ参照アーキテクチャ。デバイスのタイプは、IP または非 IP で接続されるベーシックデバイスと、HGW と直接またはアダプタ経由で接続される非ベーシックデバイスからなる
・デバイス、HGW、管理 PF、アプリケーションの 4 つのエンティティを持つ機能アーキテクチャ
・機能アーキテクチャの 3 種類の機能カテゴリ(デバイス操作、アプリケーション実行、システム管理)に関するエンティティ間の関係
・主として、HEMS に関する HN サービスに対するセキュリティモデルと機能

２　参考文献

下記の ITU-T 勧告および参考文献は、本標準の本文の中で参照され、本標準の規定となる。本標準発行時には、以下に示した版が有効であった。全ての標準および参考文献は改版される可能性がある。そのため、本標準の利用者は、以下に示した勧告および参考文献の最新版が適用可能かについて調査することが推奨される。現在有効な ITU-T 勧告のリストは定期的に発行される。

[ITU-T X.1111] Recommendation ITU-T X.1111 (2007), Framework of security technologies for home network

３　用語

3.1　他の文書で定義される用語

本標準では他文書で定義された以下の用語を利用している。

3.1.1　デマンドレスポンス [b-FG-Smart Terminology]

電力需要のピーク時において、電力需要家に対して電力消費の削減または消費パターンを変えさせるためのスマートグリッドの機能であり、通常は金銭的なインセンティブを与えることで実現する。すなわち、需要のピーク時もしくは電力供給が不安定なときに、公共施設、商業施設、産業施設、住宅に対して、何らかのインセンティブを与えてエネルギー消費を減少させるための仕組みを指す。デマンドレスポンスは

電力の需要と供給のバランスを最適化するために必須の機能である。

【補足】 スマートグリッド[b-FG-Smart Terminology]

センサーや制御機器が接続され、情報ネットワークと制御ネットワークに接続された双方向の電力ネットワーク。インテリジェントで効率的な電力ネットワークの最適化を実現する。

3.1.2 デバイスオブジェクト [b-ECHONET Lite]

センサー、エアコン、冷蔵庫等の設備機器や家電機器が保持する情報や遠隔から操作可能な制御項目を論理的にモデル化したものであり、リモート制御のためのインタフェース形式を統一したもの。個々の機器が持つ情報や制御対象をデバイスオブジェクトのプロパティとして規定し、これに対する操作方法（設定、参照）を規定する。

3.1.3 ホームネットワーク [b-ITU-T J.190]

2 つ以上のデバイスが何らかの標準的な制御方法に基づいて、情報を交換する居住環境向けに設計された近距離区間の通信システム。

3.1.4 プレゼンス [b-ITU-T Y.2720]

エンティティを特徴付ける属性のセット。属性として、主に現在の状況を持つ。

3.1.5 スマートメーター [b-FG-Smart Terminology]

スマートメーターは、デマンドレスポンス信号に基づいて、宅内機器の電力使用状況を監視したり、制御したりする建物内機器。ただし、プライバシに関するセキュリティポリシから、スマートメーターは個々の機器から直接制御することは推奨しない。個々の機器を制御したり、管理したりするためには、制御や管理機能を実現するホームゲートウェイやホームサーバのようなホームマネジメントシステムが必要となる。

3.1.6 web of things [b-ITU-T Y.2063]

物理的・仮想的なモノが World Wide Web を通じて接続され、制御される IoT を実現するための方法。

3.1.7 web リソース [b-W3C WACterms]

URI によって特定されるリソース。Web のコア機能の 1 つ。

3.2 本標準で定義される用語

本標準では以下の用語を定義する。

3.2.1 アダプタ

独自仕様の通信プロトコルを IP ベースのプロトコルに変換したり、独自仕様のデータモデルを抽象データモデルに変換したりすることで、非ベーシックデバイスのインタフェースを変換してホームゲートウェイに接続可能にするためのエンティティ。

3.2.2 デバイス管理

自動設定、動的なサービス配置、ファームウェアイメージ管理、ソフトウェアモジュール管理、状況監視、性能監視や診断機能等のデバイスを管理するための主要機能。

3.2.3 障害診断

メンテナンス作業の一例であり、ある目的のために実行される一連のメンテナンスの作業。

3.2.4 ホームコントローラ

エネルギー消費を抑制するために、家電や蓄電池等の家庭設備の監視や制御を行うホームエネルギーマネ

ジメントシステム用アプリケーションが動作する小型コンピュータ。

3.2.5　ホームエネルギーマネジメントシステム

共通的なサービスを提供するソフトウェアプラットフォーム(PF) と、家電や蓄電池等のデバイスを効果的に制御するアプリケーションで構成されるコンピュータシステム。アプリケーションは、最小コストでエネルギー供給における充分な安全性を確保する。

3.2.6　ホームゲートウェイ

ホームネットワーク上のデバイスを広域ネットワーク上のアプリケーションに接続する中継機器として振る舞う、常時起動・常時ネットワーク接続のデバイス。ホームネットワーク内のデバイスにアクセスするメッセージと同様に、ホームネットワークと広域ネットワーク間の双方向通信上のメッセージを扱い、必要なプロトコル変換を行う。

3.2.7　ホームネットワークリソース

家電や蓄電池、センサー等のデバイス、ホームネットワーク中のハブやアクセスポイント等のネットワークデバイス、ホームネットワークサービスが利用するデータ通信のためのネットワークをリソースとする。

3.2.8　インホームディスプレイ

住宅内のエネルギー消費情報を提示するための表示デバイス。ユーザは、このデバイスのユーザインタフェースを使って、住宅における電力消費量をグラフ表示したり、家庭内の機器を制御することができる。

3.2.9　管理エージェント

デバイス上で動作するソフトウェアであり、デバイスの設定情報や内部情報を収集する。管理エージェントは、管理プラットフォーム(PF)上のリソース管理機能からデバイスの構成に関する情報を取得し、遠隔管理や障害診断等を含む様々なホームネットワークサービスに関するデバイスの内部状態を送信する。

3.2.10　管理プラットフォーム(PF)

ホームネットワークアプリケーションに、インタフェースと管理機能を提供する共通機能を持つプラットフォーム(PF)。ホームゲートウェイ(HGW)とデバイスに対しては、仮想デバイス管理とリソース管理機能を提供する機能を持つ。

3.2.11　広域ネットワーク

インターネットを含む地理的に広いエリアをカバーする IP ベースの通信ネットワークで、デバイスやローカルエリアネットワークを収容する。

４　略語

本標準では以下の略語を用いる。既に日本語として広く使われている訳語を参考までに記載する。

API	Application Programing Interface	アプリケーションインタフェース
CPU	Central Processing Unit	中央制御装置
DB	Data Base	データベース
DR	Demand Response	デマンドレスポンス
EV	Electric Vehicles	電気自動車
HEMS	Home Energy Management System	ホームエネルギーマネジメントシステム
HGW	Home Gateway	ホームゲートウェイ
HN	Home Network	ホームネットワーク
HTTP	Hypertext Transfer Protocol	

IHD	In-Home Display	家庭内表示端末
IP	Internet Protocol	インターネットプロトコル
L2	Layer 2	レイヤ２
LAN	Local Area Network	ローカルエリアネットワーク
MAC	Message Authentication Code	
NAT	Network Address Translation	
PF	Platform	プラットフォーム
SOAP	Simple Object Access Protocol	
WAN	Wide Area Network	広域ネットワーク
WoT	Web of Things	
XML	Extensible Markup Language	
XMPP	Extensible Messaging and Presence Protocol	

５　規約

本標準では、記載される機能要件を「実装必須」「実装推奨」の 2 種類に分類し、以下の通りに定義する。

「実装必須」と記載されるものは、本標準に適合すると主張する場合には、厳密に従う必要があり、逸脱することが許されない機能要件であることを示す。

「実装推奨」と記載されるものは、本標準で任意に選択が許される機能要件であることを示す。この用語はベンダーの実装が推奨機能を含めて提供したうえで、ネットワーク事業者やサービスプロバイダがこれらの機能要件を選択可能であるという意味ではない。この用語は、ベンダーがこの機能要件を選択して提供し、標準に適合すると主張しても良いことを意味する。

６　概要

本章では、HN サービスアーキテクチャの概要を説明する。HN サービスは、例えば、人感センサーによって外部からの不審者を検知するホームセキュリティサービスのように、HN に接続されたデバイスを利用するものを指す。この定義では、HEMS は HN に接続される家電等を利用するサービスであるから、HN サービスの 1 つである。HEMS や他の HN サービスは、デバイスが HN に接続され、そのデバイスを制御するための HN サービスアーキテクチャによって提供される。以下、6.1 では、HN サービスアーキテクチャを説明する。6.2 では、HN アプリケーションとして HEMS を取りあげ、HEMS が 6.1 で述べた HN サービスアーキテクチャで実現できることを示す。

HN サービスの拡大や HN に接続されるデバイスの増加に伴い、アプリケーションの開発にはデバイスや通信プロトコルに関する多くの知識が必要になる。そのため、アプリケーション開発者が HN 向けアプリケーションがますます複雑になり、開発が難しくなっている。こうしたことから、アプリケーション開発者を支援する HN サービス向けアーキテクチャの提供が重要である。これが本標準の背景である。

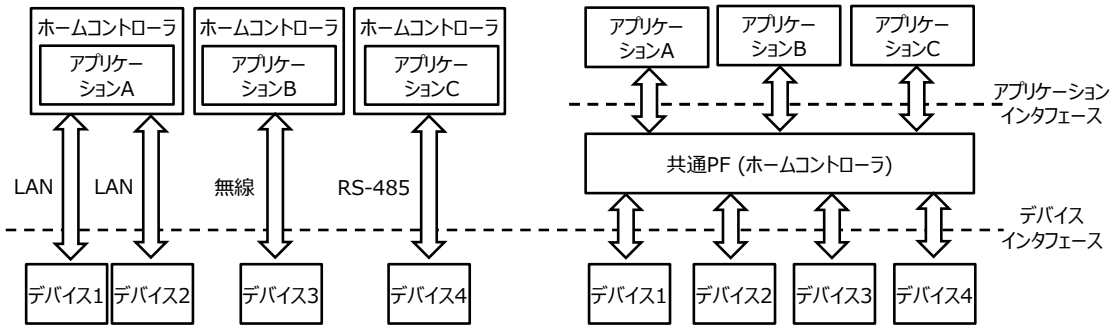
【補足】本章では、読者の理解のためにインターネットという用語を用いるが、第 7 章以降ではインターネットを含む広域ネットワーク(WAN)を用語として利用する。

6.1　HN サービスアーキテクチャ

HN アプリケーションは、宅内に設置される専用のホームコントローラ上で動作するように開発されてき

た。 図 1(a)に示すように個々のアクセスに示されるように、ホームコントローラは家電や蓄電池のような 1 つ以上のデバイス(宅内設備)と接続する。そして、それぞれのデバイスは独自の通信インタフェースを持っている。アプリケーションはデバイスの内部状態を参照したり、制御したりするために、各デバイスのインタフェースに合わせて開発する必要がある。

一方、デバイスの通信プロトコルの標準化が進んできたため、デバイスは図 1(b)に示されるようなホームコントローラ上で動作する共通 PF と標準プロトコルで接続されるようになりつつある。この共通 PF によりデバイスが接続される図では、共通 PF によって、デバイスインタフェースの抽象化が可能であり、デバイスはアプリケーションインタフェースで共通 PF に接続された任意の HN アプリケーションからアクセスされることが可能である。



(a) 独自 IF に合わせた接続　　(b) 共通 IF に合わせた接続

図 1 HN サービスへの 2 つのアクセスタイプ

図 2 は HN サービスアーキテクチャを示す。HN サービスアーキテクチャは 2 種類のアーキテクチャで構成される。

図 2(a)は、全てのデバイスとホームコントローラが宅内に設置されたアーキテクチャであり、アプリケーションと共通 PF はホームコントローラ上で動作する。本標準では、これを集約タイプのアーキテクチャと呼ぶ。図 2(b)は、デバイスが宅内に設置され、アプリケーションがインターネット上に配備されるアーキテクチャである。共通 PF の機能は宅内の HGW とホームコントローラではなく、インターネット上の管理 PF にも分散配備される。このアーキテクチャによって、アプリケーションがインターネットからデバイスにアクセスすることが可能となる。本標準では、このアーキテクチャを分散タイプのアーキテクチャと呼ぶ。

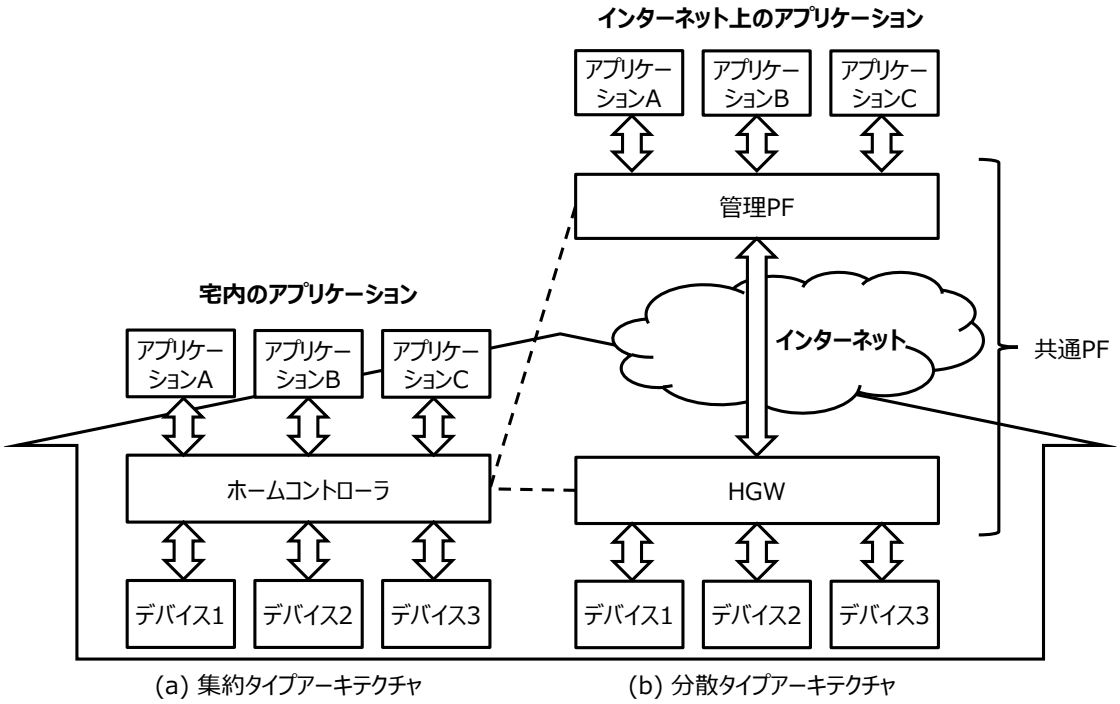


図 2　HN サービスアーキテクチャ

いずれのアーキテクチャも機能の配置位置は異なるが、共通 PF として同じ機能を持っているため、本標準のスコープ内である。両タイプのアーキテクチャは同じ機能からなるため、以下では分散タイプのアーキテクチャを中心に説明する。

6.2　HN サービスアーキテクチャに基づく HEMS

本節では、アーキテクチャの機能を明確にしながら、HN サービスアーキテクチャに基づく HEMS について説明する。

6.2.1　HEMS と HN サービスアーキテクチャ

HEMS は通常、以下のようなサービスを提供する。

・電力センサーやスマートメーターによる、家全体あるいは家電や蓄電池等の選択されたデバイスのエネルギー消費を可視化すること
・デバイスの内部状態の参照と制御を行い、需要のピーク時のエネルギーの効率利用あるいはエネルギー利用の削減をすること

HEMS アプリケーションは、HN アプリケーションの 1 つであるため、HEMS のアーキテクチャは HN サービスと同じアーキテクチャとなる。

HEMS に適用された HN サービスアーキテクチャを図 3 に示す。図 3 は、HN アプリケーションを HEMS アプリケーションとする、 図 2(b)で示される分散タイプのアーキテクチャである。

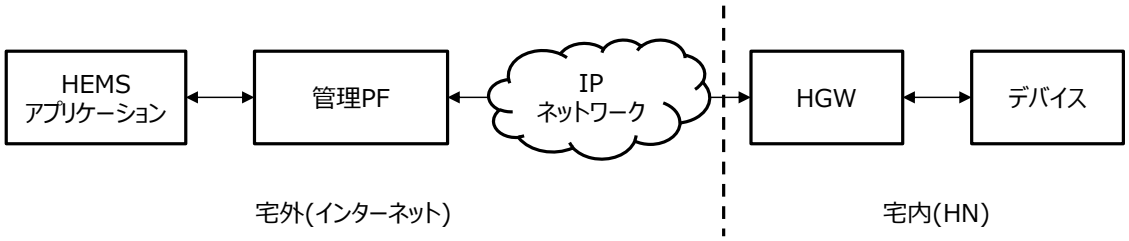


図 3　HN サービスアーキテクチャに基づく HEMS

図 3 では、HGW の左側が宅外のインターネット、右側が HN である。
HN に接続された家電、蓄電池、電力センサー等のデバイスは、インターネット上の HEMS アプリケーションから、内部状態を参照され、制御される。HGW は、インターネットと HN を中継すると同時に、デバイスとの通信に利用される様々な通信プロトコルを管理 PF との通信用にインターネット用のプロトコルに変換する。管理 PF はインターネットに配備され、Web ベースのアプリケーションインタフェースを提供する。HEMS アプリケーションは、このインタフェースを通じて実行する。
HEMS アプリケーションは、HGW と管理 PF によって、HN に接続されているデバイスを発見・特定でき、個々に割り当てられる ID を使ってデバイスにアクセスできる。このように、HEMS アプリケーションはデバイスを監視・制御し、このスキームにより HEMS が実現可能となる。
HGW と管理 PF 間で標準の通信プロトコルを利用することによって、HEMS アプリケーションはデバイスのインタフェースや、HGW とデバイス間の通信プロトコルを考慮する必要がなくなる。デバイスは、管理 PF によって Web リソースとして表現される。つまり、従来の Web アプリケーションと同様のアプリエーション開発が可能となる。したがって、本アーキテクチャによって、アプリケーション開発者はデバイスのインタフェースや通信プロトコルに関する深い知識を持たなくても、アプリケーションを開発できるようになる。

6.2.2　HEMS の構成例

本節では、アーキテクチャの機能を示すために、HN サービスアーキテクチャに基づいた 2 つの HEMS の例を説明する。
図 4 では、例えば、エアコン等の家電や電力センサー等のデバイスが、ECHONET Lite 等の標準プロトコルや独自プロトコル等のプロトコルで HGW に接続されている。HEMS アプリケーションはインターネット上で動作しており、HGW と管理 PF を介してデバイスと接続されている。このアーキテクチャでは、HEMS アプリケーションは HGW と管理 PF 経由で、電力センサーから家電の電力消費データを収集し、エネルギー消費量を可視化するために住宅内ディスプレイ(IHD)上の Web ブラウザに Web テキストベースのフォーマットでデータを送信する。また、エンドユーザが宅外からエネルギー消費量を参照できるように、スマートフォンの Web ブラウザにデータを送信することも可能である。

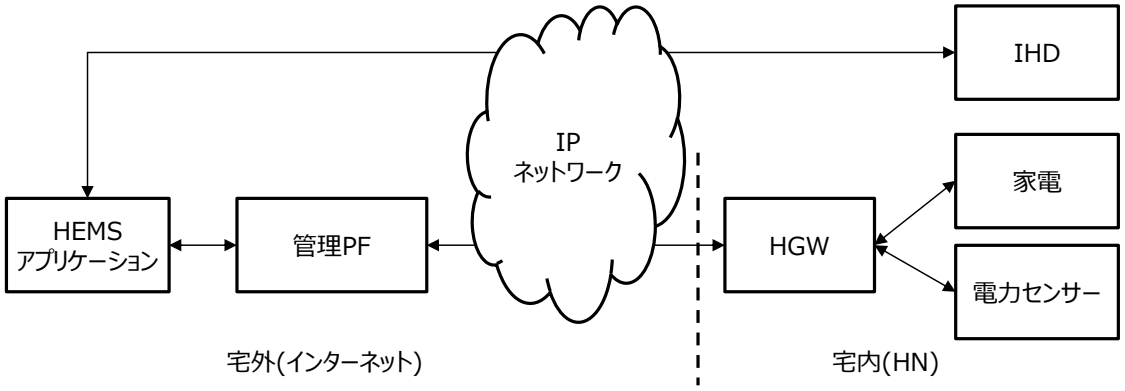


図 4　IHD を利用するエネルギー消費の可視化

このようなサービスは、エンドユーザに対して、例えば過去 1 週間のエネルギー消費量の変化をグラフとして可視化することができる。このサービスにとって、管理 PF は電力センサーから受信したデータを蓄積し、必要な時に HEMS アプリケーションに提供するものとなる。

図 5 では、HEMS アプリケーションが、電力会社が提供するデマンドレスポンス(DR)サービスを実現するアーキテクチャである。このアーキテクチャは、インターネットベースのサービスを結びつけることによって、新しいサービスを作ることができる。これがインターネット上にアプリケーションが配備される本アーキテクチャの主要な特徴である。

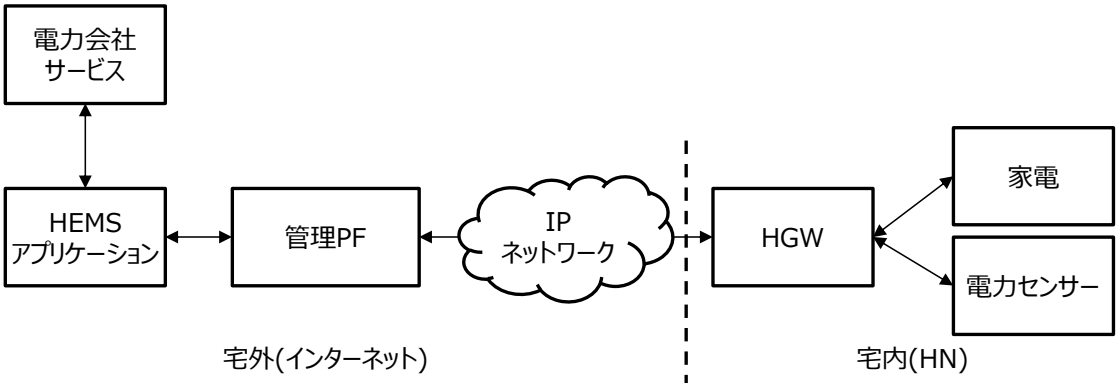


図 5　DR によるエネルギー消費制御

6.3　HN サービスアーキテクチャのメリット

6.2 では、HN を利用したサービスを開発するための HN サービスアーキテクチャと、このアーキテクチャを HEMS へ適用する方法について説明した。以下では、この HN アーキテクチャを採用するメリットについて整理する。

6.1 で説明したように、HN サービスアーキテクチャは集約タイプのアーキテクチャ(図 2(a))と分散タイプ

のアーキテクチャ(図 2(b))があり、2 つのタイプのアーキテクチャが共通 PF の同じ機能から構成されている。HN サービスアーキテクチャの 2 つのタイプのアーキテクチャに共通するメリットは、以下の①から③の通りである。

① アプリケーション開発者が、共通 PF が提供するアプリケーションインタフェースを使って、様々なサービスを提供するアプリケーションを開発することが可能となる。共通 PF のアプリケーションインタフェースは、集約タイプのアーキテクチャではホームコントローラ、分散タイプのアーキテクチャでは管理 PF で提供される。
② デバイス固有のプロトコル処理等の機能はプラグアンドプレイ機能により自動的に適用可能とすることで、エンドユーザにとってはデバイスの設置が自分自身で可能となり、サービスの導入が容易かつ低コストに実現可能となる。エンドユーザは、デバイスの接続に関して意識することは不要である。
③ システム管理者は、共通 PF を利用して、遠隔からシステム全体と HN リソースをメンテナンスすることが可能となる。共通 PF は、デバイスの自動コンフィギュレーションや HN 上で発生した障害検知機能を提供する。これらの機能により、低コストでのシステムの安定化やサービス提供が可能となる。

分散タイプのアーキテクチャは、異なるポリシーを持つネットワークで動作するデバイスとアプリケーションを協調させる機能を管理 PF 上に有する。そのため、分散タイプのアーキテクチャには、さらにいくつかのメリットがある。これらは以下の④から⑥の通りである。

④ HEMS に加えて、他の HN サービスを提供する場合、ホームコントローラでのソフトウェア処理の一部もしくは全部を管理 PF で実行することで、ホームコントローラ上の CPU やメモリ等のハードウェアリソースの増加を抑制し、サービスコストを抑えることができる。ハードウェアの制約を受けることなく、インターネット上のアプリケーションを変更したり、追加したりすることが可能である。
⑤ アプリケーションがデバイスにアクセスするための API を提供する管理 PF によって、容易に HN 上のデバイスの内部状態を参照したり、制御したりすることが可能となる。このアクセスはインターネットからの不法アクセスを防ぐためのファイアーウォールや NAT が存在する場合にも可能である。
⑥ デバイスと HGW に認証認可機能を提供し、HN を暗号化することによって、セキュリティが必要なアプリケーションの開発も容易に実現可能である。

7　機能要件

本章では、HN サービスアーキテクチャにおけるデバイス、HGW、管理 PF の機能要件と、セキュリティに関する要件を説明する。HEMS の機能を 6.2 で定義したため、機能要件は HEMS を参考にして抽出した。HN サービスアーキテクチャは、HEMS だけではなく、他の HN サービスにも適用可能であるため、以下は他の HN サービスの機能要件でもある。ただし、アプリケーションは、本アーキテクチャでは必須のエンティティであるが、機能要件はない。

7.1　デバイスの機能要件

デバイスの機能要件は以下の通りである。

⑴デバイス操作の機能要件

・デバイスオブジェクト

デバイスの機能を表現するための抽象的なデータモデルであるデバイスオブジェクトを持つこと（実装必須）

【補足】デバイスオブジェクトを持たないデバイスは、デバイスを直接収容するアダプタや HGW がデバイスの代わりにデバイスオブジェクトを持つこと

⑵管理の機能要件

・管理エージェント

HGW のリソース情報収集機能からの問い合わせに応答すること（実装必須）

障害診断のためにデバイス自身の内部状態をチェックすること（実装必須）

デバイスや HN のハブやアクセスポイント等のネットワークデバイスの設定を行うこと（実装必須）

7.2　HGW の機能要件

HGW の機能要件は以下の通りである。

⑴デバイス操作の機能要件

・データフォーマットとプロトコル(HTTP/IP)変換

デバイスオブジェクトのフォーマットを仮想デバイスのフォーマットに変換すること、及び WAN 上を安全な通信でデータを管理 PF へ送信するために、IP を HTTP に変換すること（実装必須）

⑵管理の機能要件

・リソース情報収集

新たに HN に接続されたデバイスを発見し、それぞれを特定し、個々の状態を管理すること（実装必須）

HN サービスが正常に動作しない場合に、障害の原因を決定するために、デバイスや他の HN リソースの内部状態や HN のトラフィック状態を収集すること（実装必須）

(3)アプリケーション実行の機能要件

・切断時用のアプリケーション

管理 PF とのネットワーク切断時に、バックアップ用のアプリケーションによって、自律的にデバイスを制御し続け、データを蓄積できること（実装推奨）

バックアップ用のアプリケーションによって、何らかのタスクを実行したり、管理 PF と協調動作をすること（実装推奨）

7.3 管理 PF の機能要件

管理 PF に対する機能要件は以下の通りである。

(1)デバイス操作の機能要件

・仮想デバイス

デバイスオブジェクトに対応する Web と親和性の高い表現方法を提供すること（実装必須）

仮想デバイスを監視・制御できること（実装必須）

1 つの物理的なデバイスを複数の機能として表現したり、複数の物理的なデバイスを 1 つの仮想デバイスとして表現する仮想デバイス表現をできること（実装推奨）

【補足】物理デバイスは HN に接続されるデバイスを指す。物理デバイスという用語は、仮想デバイスと区別するために用いる。

(2)管理の機能要件

・リソース管理

HGW に接続されたデバイスを発見し、活性化し、監視し、制御できること（実装必須）

アプリケーションからデバイスを唯一に識別できること（実装必須）

HGW のプロファイルを登録し、HGW に ID を付与し、HGW を所有するエンドユーザを登録し、認証されたユーザ情報によって HGW を識別できること（実装必須）

(3)アプリケーション実行の機能要件

・アプリケーション管理

エンドユーザによる許可により、デバイス接続に関するアプリケーションを認証認可する機能を持つこと

HGW を介して、デバイスから受信した情報を蓄積しても良い（実装推奨）

・アプリケーションインタフェース

HEMS や他の HN サービス向けのアプリケーションに対して、Web ベースのアプリケーションインタフェースを持つこと（実装必須）

7.4 セキュリティの機能要件

セキュリティに関する機能要件は以下の通りである。

・セキュアな通信

デバイスとアプリケーション間の WAN で安全に通信できること（実装必須）

【補足】HEMS を実現するうえで、安全な通信を実現するためのセキュリティ要件の詳細については付録 III の表 III.2 に示す。

・デバイス認証

管理 PF はデバイスを認証する機能を持つこと（実装必須）

８　参照アーキテクチャ

本章では、HEMS の参照アーキテクチャを説明する。本アーキテクチャは、他の HN サービスにも適用可能である。

図 6 は、分散タイプのアーキテクチャと参照点を示めす。

図 6 では、HGW から左側が WAN であり、右側が HN である。HN の接続は IP を基本とし、HGW は WAN と HN の通信を中継する。

この参照アーキテクチャには、ベーシックデバイスと非ベーシックデバイスに分類されたデバイスが示されている。ベーシックデバイスは、デバイスの機能を表現する抽象的なデータモデルであるデバイスオブジェクトを持つ。ベーシックデバイスのインタフェースは、HGW に標準で備わっており、ここでは抽象データモデルが利用される。また、抽象データモデルは管理 PF においても表現されており、アプリケーションが内部状態を参照し、制御する仮想デバイスとして扱われる。

ベーシックデバイスのインタフェースには、いくつかの標準仕様がある。そのうちのいくつかは IP ベースの通信プロトコルをサポート（図 6(a)）し、他のものは非 IP ベースのプロトコルをサポート（図 6(c)）している。非ベーシックデバイスは、内部にデバイスオブジェクトを持たない。非ベーシックデバイスは独自通信インタフェースで接続するため、この HGW に接続するためにはデバイスオブジェクトをサポートするアダプタを接続する（図 6(b)）ことが必要となる。デバイスと HGW が LAN 等を経由して直接接続可能な場合には、HGW にアダプタ機能を持つことも可能であり、この場合には非ベーシックデバイスは HGW に直接接続される（図 6(d)）。

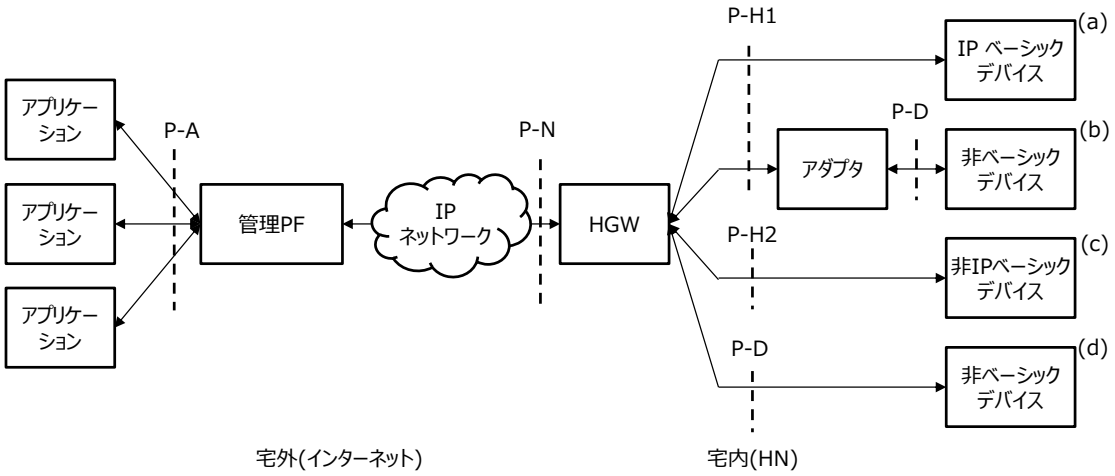


図 6　参照アーキテクチャと参照点

図 6 では、以下の参照点が定義される。図 1 に示されるアプリケーションインタフェースは、P-A 参照点に対応する。同じく図 1 に示されるデバイスインタフェースは、P-H1、P-H2、P-D 参照点のいずれかに対応する。

⑴P-A 参照点

P-A 参照点は、アプリケーションが Web ベースのアプリケーションインタフェースを通じて管理 PF にアクセスし、Web リソースとして扱われる仮想デバイスとして、HN に接続された物理デバイスの内部状態を参照し、制御できる。この参照点によって、アプリケーションは仮想デバイスの生成や削除、仮想デバイスのプロパティの参照や更新が可能となる。

⑵P-N 参照点

P-N 参照点は、管理 PF が WAN を介して宅内に設置された HGW にアクセスすることを可能とする。この参照点によって、管理 PF はリソース管理機能としてデバイスを利用可能な状態（アクティベート）し、その後、デバイスオブジェクトのプロパティ値を指定することで、デバイスの内部状態を取得し、制御することが可能となる。

⑶P-H1 参照点

P-H1 参照点は、デバイスが IP ベースの通信プロトコルで、デバイスオブジェクトをサポートする場合の HGW との接続点である。デバイスがアダプタ経由で接続する場合もこの参照点で接続される。この参照点によって、HGW はデバイスをアクティベートし、デバイスのステータスを取得し、プロパティの値を指定することでデバイスを制御可能となる。

⑷P-H2 参照点

P-H2 参照点は、デバイスが非 IP ベースの通信プロトコルで、デバイスオブジェクトをサポートする場合の HGW との接続点である。ベーシックデバイス(非 IP ベースのベーシックデバイス)にアクセスすることを可能とする。この参照点により、HGW はデバイスをアクティベートし、デバイスのステータスを取得し、プロパティの値を指定することによって、デバイスを制御可能となる。

⑸P-D 参照点

P-D 参照点は、アダプタや、HGW に実装されたアダプタの機能がデバイスインタフェースの独自通信プロトコルによって、非ベーシックデバイスにアクセスすることを可能とする。この参照点により、HGW はデバイスをアクティベートし、デバイスのステータスを取得し、プロパティの値を指定することによって、デバイスを制御可能となる。

以下では、図 6 に示すアーキテクチャの構成要素を、デバイス、HGW、管理 PF の順に説明する。
宅内の各デバイスは、4 つの方法のうちの 1 つの方法で HN を介して HGW に接続される。図 6 の(a)から(d)に示されるデバイスの接続方法について、以下に説明する。
デバイス(a)は IP ベースのベーシックデバイスであり、P-H1 参照点で HGW と直接接続する。デバイスは、プロトコルに合うインタフェースを持っているので、デバイスと HGW 間で IP ベースの通信プロトコルを利用する。[b-ECHONET Lite]は、このような通信プロトコルの 1 つの例である。
デバイス(b)は非ベーシックデバイスであり、独自のインタフェースを持つ。HGW に接続するためには、デバイスはアダプタを介して HN に接続する。アダプタは、デバイス自身がサポートする独自の通信プロ

トコルを IP ベースのプロトコルに変換するとともに、内部に組み込まれた独自のデータモデルを共通に扱われる抽象的なデータモデルに変換する。すなわち、デバイス(b)はアダプタを経由することで、ベーシックデバイスと同様に認識される。シリアルインタフェースで HN に接続される電気自動車(EV)の充電器は、アダプタを経由して接続することによりこの例に対応する。
デバイス(c)はベーシックデバイスであるが、非 IP 通信プロトコルのみを利用可能であるため、HGW と直接通信するためには非 IP ベースの通信プロトコルが利用される。抽象化されたデータモデルを内部に持っており、データモデルの変換は行われない。Bluetooth や ZigBee は、このデバイスの例である。
デバイス(d)は非ベーシックデバイスであり、デバイス(b)と同様である。しかし、このケースではデバイスがサポートするネットワーク媒体を HGW がサポートしており、ネットワークとしては直接接続可能なケースである。この場合には、HGW は内部にデバイス(d)に対するアダプタの機能を持つことができるため、アダプタを外部に接続必要がなく、デバイスは HGW に直接接続される。

HGW は、新たに接続されたデバイスを自動的に発見する機能を持つ。HGW は、デバイスの接続通知を受信し、機能不全時にはアラームを受信する。これによって、システムの信頼性が向上する。また、エンドユーザが自分自身でデバイスを接続し、HEMS や他の HN サービスで利用することも容易になる。HGW は、デバイスで利用される様々な通信プロトコルを、P-N 参照点で管理 PF がサポートする WAN 向けの通信プロトコルに変換する。例えば、デバイスが ECHOET Lite をサポートする場合には、ECHONET Lite を HTTP ベースのプロトコルに変換する。
HGW とデバイス(アダプタ経由を含む)間の通信は、IP ベースあるいは非 IP ベースに関わらず、暗号化せずに使われる可能性がある。しかし、WAN を経由する HGW と管理 PF の間の通信はセキュアであることが必須であり、認証や暗号化機能を備え、安全なコミュニケーションを保証する HTTP のような通信プロトコルを利用する必要がある。たとえば、HTTP は[b-BBF TR-069]のような標準のデバイス管理プロトコルを利用することもできる。

管理 PF は、WAN 上に提供されるサーバ機能である。管理 PF は、Web ベースのアプリケーショインタフェースを備え、アプリケーションとは P-A 参照点で接続され、このインタフェースを介してアプリケーションが実行される。HGW と管理 PF 間では標準の通信プロトコルを利用することによって、アプリケーションはデバイスのインタフェースや HN で利用されている通信プロトコルを考慮する必要がなくなる。HN に接続されるデバイスは、管理 PF によってその機能が抽象データモデルとして表現され、Web リソースとして扱われる。この機能によって、アプリケーション開発者はデバイスに関する深い知識を持たなくても、アプリケーションを開発することができる。
【補足】[b-ITU-T Y.2063]で規定される WoT の機能配置モデルを付録 1 に示す。

９　機能アーキテクチャ

本章では機能アーキテクチャについて説明する。本アーキテクチャは、HEMS の他、一般的な HN サービスにも適用可能である。

図 7 は、IP ベースのベーシックデバイスに対する分散タイプの機能アーキテクチャを示す。このアーキテクチャ内の機能は、デバイス操作、アプリケーション実行、システム管理の 3 つのカテゴリで構成される。それぞれのカテゴリの詳細は、第 10 章で説明する。図 7 では、デバイスは IP ベースのベーシックデバイスとして表記されている。アダプタを利用することで、全てのデバイスをベーシックデバイスとして扱うことができることは 9 章で説明したので、図 7 ではデバイスの構成については簡略化した表現となっている。各タイプのデバイスに対する詳細アーキテクチャは、10.1.1 から 10.1.4 に記載した。管理 PF とアプリケーション間のアーキテクチャは、図 7 に示すように全てのデバイスで共通である。

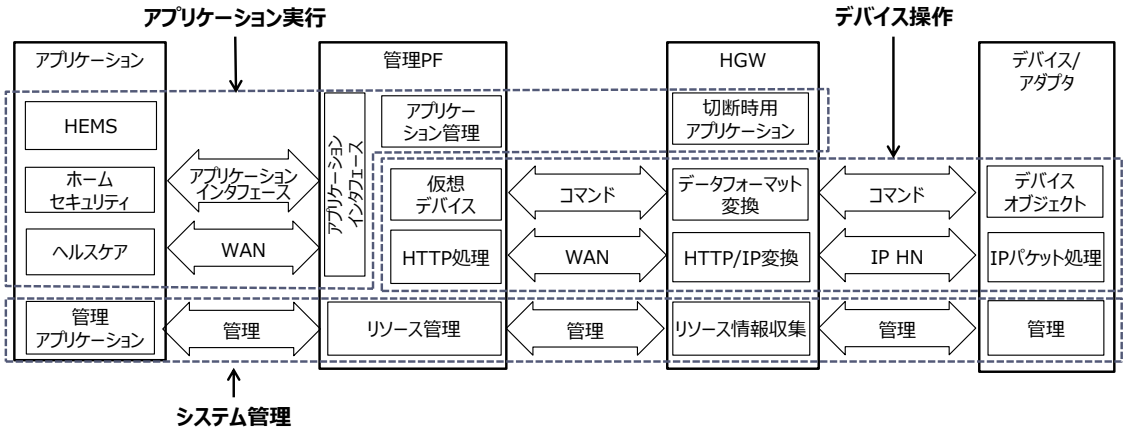


図 7　IP ベースのベーシックデバイスに対する機能アーキテクチャ

デバイスはそれぞれ固有の機能を持つ。これらの機能はプロファイルとして定義され、HGW や管理 PF を通じて、アプリケーションに通信プロトコルによって送信される。この結果、本アーキテクチャでは、アプリケーションがデバイスの内部状態を参照し、制御することが可能である。

図 7 は、IP ベースのベーシックデバイスに対する IP ベースの通信プロトコルを示しているが、第 8 章で説明したように、HGW とデバイス間では IP ベース、非 IP ベース、独自の通信プロトコルが利用される。HGW とデバイス間でやりとりされるコマンドは、HGW がデバイスに対して、内部状態を取得するための”GET”（取得）、プロパティを指定して情報を設定する”SET”（参照）、内部状態やイベントが更新したときに通知するように要求する”INFORM”（通知）のような制御メソッドを提供する。HGW は、通信プロトコルを HTTP に変換し、WAN を介して、管理 PF と通信する。

[b-BBF TR-069]のようなデバイス管理のための通信プロトコルが、HGW と管理 PF 間で利用される。[b-BBF TR-069]は、[b-BBF TR-181]で規定された XML で一般化されたデバイスデータモデルを参照しており、このデータモデルが”GET”、”SET”、”INFORM”のようなコマンドで利用されている。XML フォーマットは、HGW とデバイス間で利用される通信プロトコル毎に規定されるデバイスモデルに従う。

管理 PF は、HGW を経由して取得したデバイスのステータスや構成管理データを蓄積する。管理 PF は、

仮想デバイスを管理し､Web ベースのアプリケーションインタフェースを介して仮想デバイスをアプリケーションに提供する。アプリケーション開発者は物理的なデバイスを Web リソースとして制御することでアプリケーションを開発できる。
以下の節では、図 8 に示す各エンティティの詳細を説明する。デバイスや HGW の機能は、IP ベースのベーシックデバイスに関して説明する。他の種類のデバイスの機能は 10.1.2 から 10.1.4 で示す。管理 PF とアプリケーションの機能は、全てのタイプのデバイスに対して共通である。
【補足】HN アプリケーションの例は、付録 2 で説明する。

9.1　デバイス

IP ベースのベーシックデバイスは以下の機能を提供する
⑴デバイス操作機能
・デバイスオブジェクト
・IP パケット処理
⑵管理機能
・管理エージェント

上記のうち、デバイスオブジェクトと管理エージェントについて説明する。IP パケット処理は IP パケットでの通信する機能であり、ここでは割愛する。

9.1.1　デバイスオブジェクト

ベーシックデバイスは、デバイスオブジェクトを持つ。デバイスオブジェクトは、実装とは独立したデバイス機能を規定するプロパティ群で構成される。プロパティは、デバイス内部状態を取得したり、デバイス機能を制御したりするための論理的な内部機能項目であり､遠隔のアプリケーションからアクセスされ、制御される。遠隔制御に対するデータ構造は、<プロパティ、値>の組で規定される。デバイスオブジェクトは、それぞれの種類のデバイス毎(家電、蓄電池等)に規定されるので、異なる製造者による既存の家電は、同様の方法で遠隔から制御できる。
例えば、エアコンは[b-ECHONET Lite]で、プロパティとして定義される稼働状態、温度設定、操作モードのプロパティを持つ。[b-SEP 2.0]や[b-ISO/IEC 14543-3-x]も同様のプロパティ構造を定義している。HGW はプロパティを指定して、プロパティからデータ(値)を取得する。プロパティを設定したり、制御したりするには、プロパティを指定して、適切な値を設定する。例えば、エアコンの目標温度を設定するには、目標温度に応じたプロパティを指定し、適切な値(例えば 25 度)を設定する。

9.1.2　管理エージェント

管理エージェントは､HN を安定動作させるための管理機能である｡情報の欠落により障害検知がされず、原因の特定を難しくする。したがって、HN リソースのあらゆる情報を取得可能にしておくことが重要である。管理エージェントはデバイスの内部状態を保持し、必要に応じて HGW のリソース情報収集機能を通じて、管理 PF のリソース管理機能に送信する。詳細は 10.3 で説明する。

9.2 HGW

HGW は、WAN と HN を中継する。HGW は、IP ベースのベーシックデバイスと接続する際に、以下の機能を提供する。

⑴デバイス操作機能

・データフォーマットとプロトコル(HTTP/IP)変換

⑵管理機能

・リソース情報収集機能

⑶アプリケーション実行機能

・WAN 切断時のアプリケーション実行環境

9.2.1 データフォーマットとプロトコル(HTTP/IP)変換

この機能は、通信プロトコルを変換するためのデバイス操作機能である。WAN では、HTTP が通信プロトコルとして、よく利用されている。そこで、HGW は HN で利用されている通信プロトコルを HTTP に変換する。

物理的なデバイスの<プロパティ、値>の組は、HN を通じて HGW に送信され、HGW で管理 PF と通信するために HTTP に変換される。[b-BBF TR-069]が SOAP ベースで<プロパティ、値>形式の情報をやり取りする通信プロトコルを規定していることはよく知られている。これは、HGW と管理 PF 間の通信プロトコルの 1 つの候補である。プレゼンスの通信プロトコルである XMPP も同様の通信方式を有しており、候補の 1 つである。

9.2.2 リソース情報収集機能

この機能は、個々の HGW に対する HN リソースの情報を収集し、その情報を管理 PF 上のリソース管理機能に通知するための管理機能である。この機能はまた、新たに HGW に接続されたデバイスを発見し、それらのデバイスに対する初期設定を行う。HGW は、個々のデバイスに対してユニークな ID を付与する。

9.2.3 WAN 切断時のアプリケーション実行環境

切断時用のアプリケーション機能は、WAN が何らかの理由で切断された場合に、HN に接続されているデバイスを動作し続けるための代替アプリケーション機能を提供する。WAN が切断されると、バックアップ目的のアプリケーションは WAN 上で動作しているアプリケーションに代わって、状況を満たすように正しい設定を行う。このバックアップ目的のアプリケーションに対するアプリケーションインタフェースは、HTTP に基づいており、デバイスオブジェクトを管理する API に変換されたデータフォーマットを提供する。この機能を、デバイスで発生するデータの前処理として利用することも可能である。

9.3 管理 PF

管理 PF は HN に接続される物理デバイスを仮想デバイスとして管理し、アプリケーションに対して、アプリケーションインタフェースを通じて Web リソースとして仮想デバイスを制御可能とする。管理 PF は以下の機能を備える。

(1)デバイス操作機能

・仮想デバイス

・HTTP 処理

(2)管理機能

・リソース管理

(3)アプリケーション実行機能

・アプリケーション管理

・アプリケーションインタフェース

これらの機能のうち、HTTP 処理以外の機能を以下の節で説明する。HTTP 処理は HTTP でデバイス情報を通信する機能を提供するものであり、ここでは説明を割愛する。

9.3.1 仮想デバイス

仮想デバイスは、HN に接続されたベーシックデバイスの、デバイスオブジェクトに対応するデバイス表現である。デバイスのプロパティは、Web アプリケーションで容易に扱うことができるように、XML 等のフォーマットで表現される。この機能の詳細は 10.3 で説明する。

仮想デバイスは、デバイスの抽象化に関して、2 つの機能を提供する。1 つ目の機能はデバイスのプロパティと通信プロトコルの抽象化である。例えば、ベンダーA とベンダーB が、エアコンの同じ機能に異なるプロパティ名を定義している場合、管理 PF ではデバイス(エアコン)のそれぞれのプロパティ名を同じプロパティ名になるように変換して、仮想デバイスを生成する。

2 つ目の機能は、1 つの物理的なデバイスに複数の機能が備わっている場合に、それぞれを複数の仮想デバイスとして分離する機能である。例えば、人感センサーによって自動的に機能を制御するエアコンは、人感センサーによる人がいるかどうか、どこにいるか等の検知データに関するプロパティを持つ。そこで、エアコンは 2 つの機能、すなわち空調機能と人感検知機能を持つと定義することができる。これにより、2 つの仮想デバイス(空調デバイスと人感検知デバイス)が、1 つの物理的なデバイス(エアコン)から生成される。この機能により、エアコンの人感検知デバイスとしての機能のみを利用し、他のデバイスの制御をするアプリケーションを実現することも可能となる。

9.3.2 リソース管理

管理 PF 上のリソース管理は、管理エージェントが収集した HN リソースの情報を集める機能を提供する。また、障害を検知し、障害対処を行うために、デバイスやネットワークデバイスの内部状態、HGW 毎のネットワーク帯域を管理する。詳細については 10.3 で説明する。

9.3.3 アプリケーション管理

アプリケーション管理は、アプリケーションの情報を登録し、アプリケーションとデバイス間の関係を保持する。この機能は、デバイスから適切なアプリケーションにデータを送信する。さらに、アプリケーションの代わりに、デバイスから受信したデータを蓄積する履歴データ管理機能を持つ。この機能は、例えば、アプリケーションの要件に応じて、過去 24 時間のデバイスデータ等を提供する。

9.3.4　アプリケーションインタフェース

アプリケーションインタフェースは、HGW を通じてデバイスの内部状態を参照し、制御するための Web ベースのインタフェースである。このインタフェースでは、アプリケーションが管理 PF 上の仮想デバイスを Web リソースとしてアクセスする。

9.4　アプリケーション

本標準のアーキテクチャでは、アプリケーションに要求される共通の機能はない。本節では、管理 PF の機能を利用し、アプリケーションがどのように実現されるかを記載する。以下に記載する 3 つの機能操作が、アプリケーションに提供されるべき機能である。それらは図 8 で示されるように、デバイス管理機能、デバイス操作機能、障害診断機能である。図 7 に示されるアプリケーションエンティティ内のアプリケーションは、これらの機能を備えていると考えてよい。

3 つの機能は、以下に対応する操作として説明される。

⑴機能：デバイス管理

管理 PF の仮想デバイスにアクセスするアプリケーションは、管理 PF のアプリケーション管理機能に登録される。この操作は、アプリケーションのデバイス管理機能によって実行される。登録することにより、デバイスの制御に関してアプリケーション間の排他制御が可能となり、仮想デバイスで発生するデータを自動的に送信するアプリケーションを指定することができる。

⑵機能：デバイス操作

アプリケーションにおけるデバイス操作機能は、管理 PF で管理される仮想デバイスの設定情報の取得や設定をすることによって、HN に接続される物理デバイスの内部状態の参照し、制御することである。デバイスのオン/オフ制御は、1 つの例である。この操作は、10.2 で説明する管理 PF における仮想デバイスの内部状態を参照・制御することにより実行される。仮想デバイスから HN 上の物理デバイスへの操作方法は、10.1 で説明する。

⑶機能：障害診断

アプリケーションにおける障害診断機能は HN リソースの内部状態や設定情報を取得する。この機能は管理 PF のリソース管理に接続し、リソース管理によって検出された HN に関連する情報を取得し、その情報を必要とするアプリケーションに通知する。

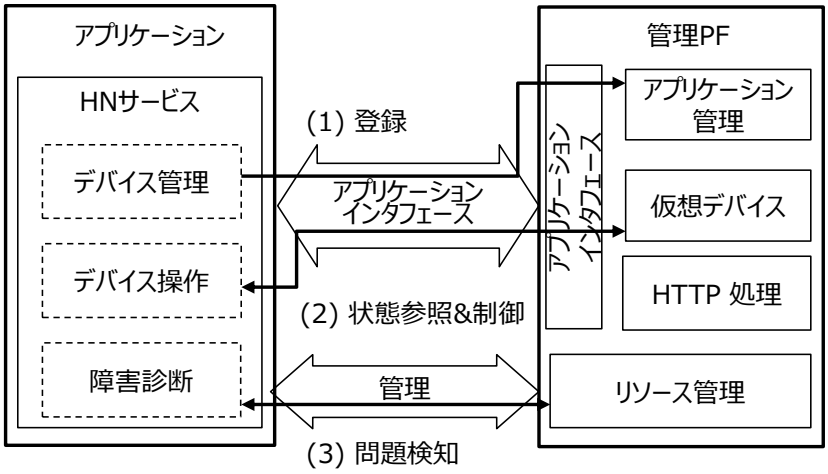


図 8　HN アプリケーションに対する 3 つの操作

１０　機能関連性

本章では、エンティティ間の関係を明確にするため、図 7 に示すデバイス操作、アプリケーション実行、システム管理の 3 つの機能カテゴリから詳細を説明する。

10.1　デバイス操作

デバイス操作は、管理 PF からデバイスの内部状態を参照・制御する機能を提供する。図 6 に示すように、HGW からデバイスに接続するには、デバイスの種類に応じて、4 つの方法があるため、以下ではそれぞれの接続方法について説明する。

10.1.1　IP ベースのベーシックデバイスの操作

図 9 は、IP ベースのベーシックデバイス(図 6 のデバイス(a))の操作に対する機能アーキテクチャを示している。IP ベースのベーシックデバイスは、デバイスオブジェクトと IP パケット処理の 2 つの機能を持つ。管理 PF 上の仮想デバイスは、ベーシックデバイスのデバイスオブジェクトに対応するデバイスの表現である。アプリケーションは、アプリケーションインタフェースを通して仮想デバイスにアクセスする。そして、仮想デバイスのプロパティを指定することによって、遠隔から実際のデバイスの内部状態を参照し、制御することができる。

<プロパティ、値>の組は、デバイスが持つ機能にアクセスするためのデータ形式である。デバイスコマンドは、HN では IP ベースの通信プロトコルで HGW に送信され、WAN では HTTP ベースのプロトコルで管理サーバに通知される。HN のプロパティ組の形式は、内容は同じであるが、WAN 上の形式と表現が異なるので、HGW は HN と WAN の間で相互に変換する。したがって、HGW はそのプロパティ組の形式を変換するデータフォーマット変換と、HN 上の通信プロトコルを HTTP に変換するプロトコル(HTTP/IP)変換の 2 つの機能を持つ。

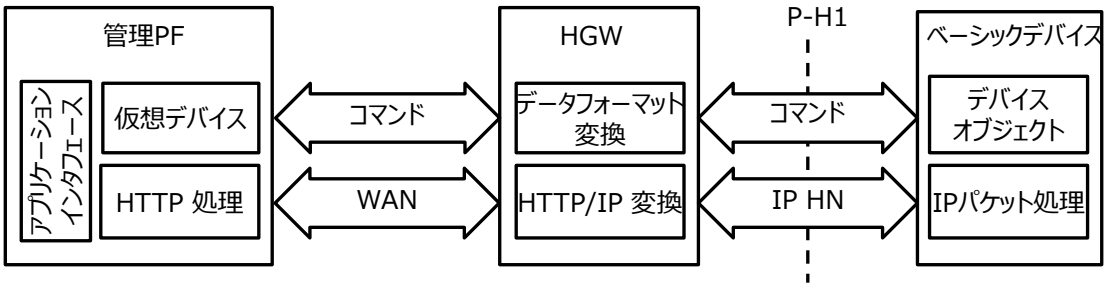


図 9　IP ベースのベーシックデバイス操作のための機能アーキテクチャ

10.1.2　非 IP ベースのベーシックデバイスの操作

図 10 は、非 IP ベースの通信プロトコルで HGW と接続するベーシックデバイス(図 6 のデバイス(c))に対する機能アーキテクチャである。このケースでは、デバイスコマンドは P-H2 参照点で変換される。L2 フレーム処理機能が、デバイスのサポートする非 IP ベースの通信プロトコルを HGW 内で IP ベースのプロトコルに変換するために利用される。HGW と管理 PF 間の通信プロトコルは、図 9 の IP ベースのベーシックデバイスの場合と同じである。

たとえば、[b-SEP 2.0]のデバイスインタフェースは、P-H2 参照点をサポートする。

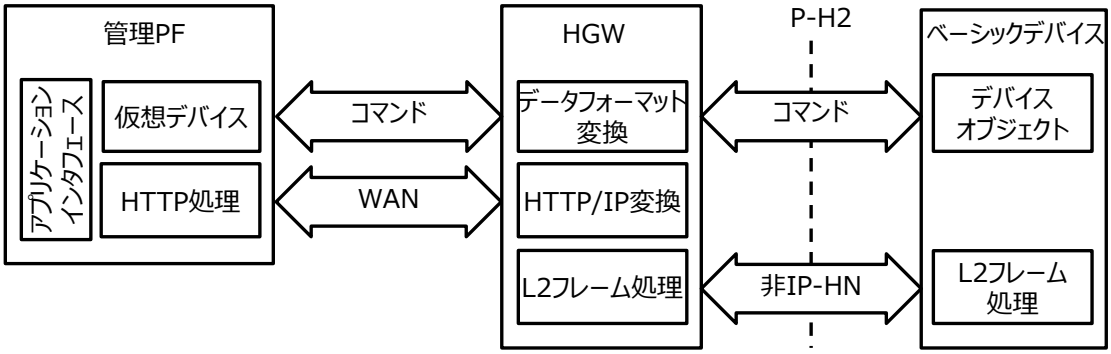


図 10　非 IP ベースのベーシックデバイス操作のための機能アーキテクチャ

10.1.3　アダプタによる非ベーシックデバイスの操作

図 11 は、アダプタによる非ベーシックデバイス(図 6 のデバイス(b))のための機能アーキテクチャである。非ベーシックデバイスは、デバイスオブジェクトを持たない既存のデバイスである。多くの非ベーシックデバイスは、独自のインタフェースを持つ。参照点 P-D の例は、でシリアルインタフェースである。そこで、デバイスと HGW 間に設置されたアダプタは、参照点 P-H1 でベーシックデバイスとして認識されるように、非ベーシックデバイスを変換してベーシックデバイスとして動作する。

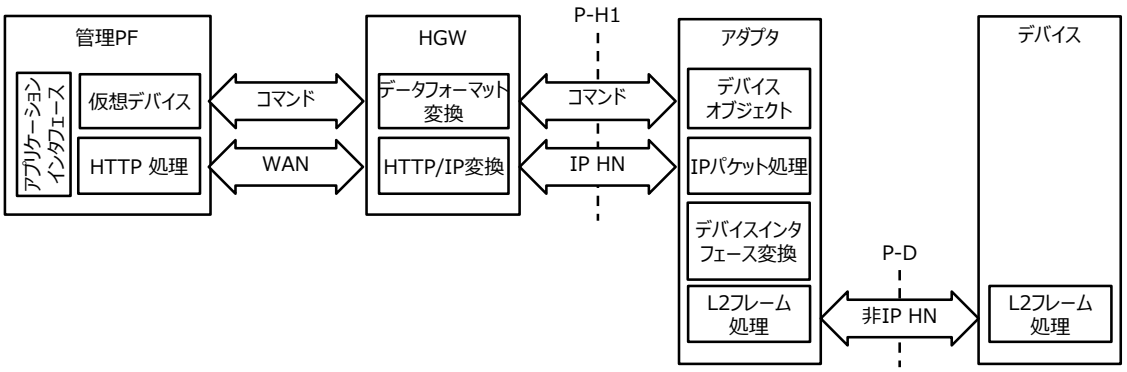


図 11　アダプタによる非ベーシックデバイス操作のための機能アーキテクチャ

10.1.4　HGW 内のアダプタ機能による非ベーシックデバイスの操作

図 12 は、HGW に直接接続している非ベーシックデバイス(図 6 のデバイス(d))の機能アーキテクチャである。非ベーシックデバイスにとって、HGW はデバイスを直接接続するためにアダプタを配置するのではなく、アダプタの機能(デバイス抽象化機能)を具備する。HGW 内部のデバイス抽象化機能は、ベーシックデバイスと同じインタフェースを提供する。デバイス抽象化機能はまた、参照点 P-D で操作手続きをサポートし、非ベーシックデバイスは図 12 で示されるように直接 HGW に接続する。

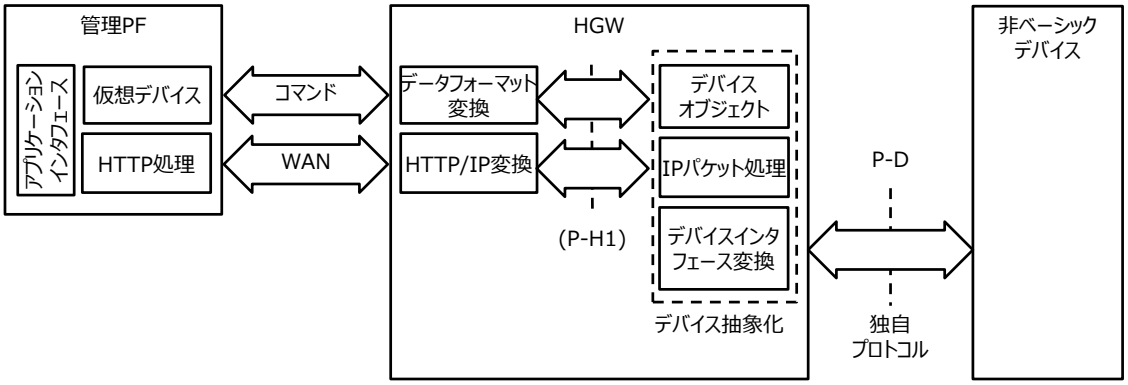


図 12　HGW 内にアダプタ機能を持つ非ベーシックデバイス操作のための機能アーキテクチャ

10.2　アプリケーション実行

アプリケーションによる管理 PF 上の仮想デバイスのプロパティ値の設定や取得は、結果的に HN に接続された物理的なデバイスの内部状態の参照や制御を行う。図 13 に示されるように、管理 PF のアプリケーションインタフェースは、仮想デバイスを Web リソースに変換し、結果として、アプリケーションは HTTP プロトコルで物理的なデバイスの内部状態を参照し、制御できる。

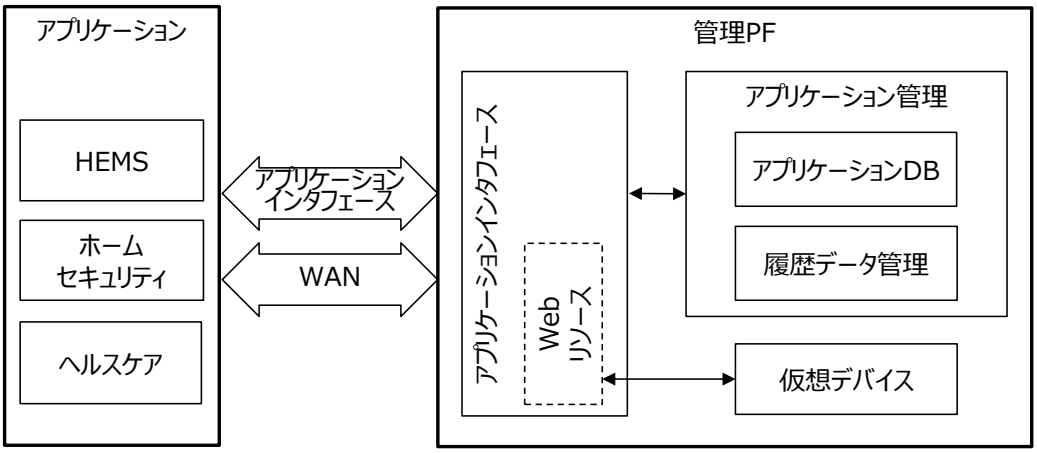


図 13　アプリケーション実行のための機能アーキテクチャ

アプリケーション管理は、アプリケーションデータベース(DB)と履歴データ管理の 2 つの機能で構成される。

アプリケーション DB は、WAN を介して管理 PF に接続されているアプリケーションのリストを持つ。アプリケーション DB は、HGW や他のアプリケーションから、ターゲットとなるアプリケーションにデータを送信するために利用される。

履歴データ管理機能は 9.3.3 に記載した通り。

10.3　システム管理

HN は、多くの異なる技術が共存するため、非常に複雑になることがある。HN に接続されたデバイスは、様々な領域で利用される。HN はいくつかの HN リソース(デバイスやアクセスポイント)が組み合わされ、複雑なトポロジとなる。エンドユーザにとって、家庭内には管理者や技術者がいないため、HN を管理し、維持することは困難である。したがって、管理者や遠隔管理者が不要で、容易な構成管理を含む、様々な障害特定処理や障害復旧処理を実現するために、リソース管理機能が提供される。

図 14 は、ベーシックデバイスを持つ HN の管理のための機能アーキテクチャを示す。管理エージェントを持たない非ベーシックデバイスでは、アダプタや HGW が管理エージェント機能を提供する。

管理 PF にはリソース管理機能があり、その機能は HN リソースの情報や構成情報を保持する。HGW は、リソース情報収集機能を持つ。HGW は、HN リソースのステータス、性能、構成情報を取得し、障害を検出し、障害処理を提供する。また、エンドユーザに対して、HN リソースの容易な構成管理手順を提供する。HGW は、必要とされる操作に関する最小の設定で HN リソースを設定する。デバイス上の管理エージェントは、HGW 上のリソース情報収集機能からの指示によって、宅内の環境情報を設定して収集する。管理アプリケーションは、コールセンターやカスタマーサポートのような遠隔管理者用のアプリケーションである。それは、障害診断用の完全なリソース情報を表示し、障害からの復旧操作のために特定のプロパティを設定する機能を持つ。

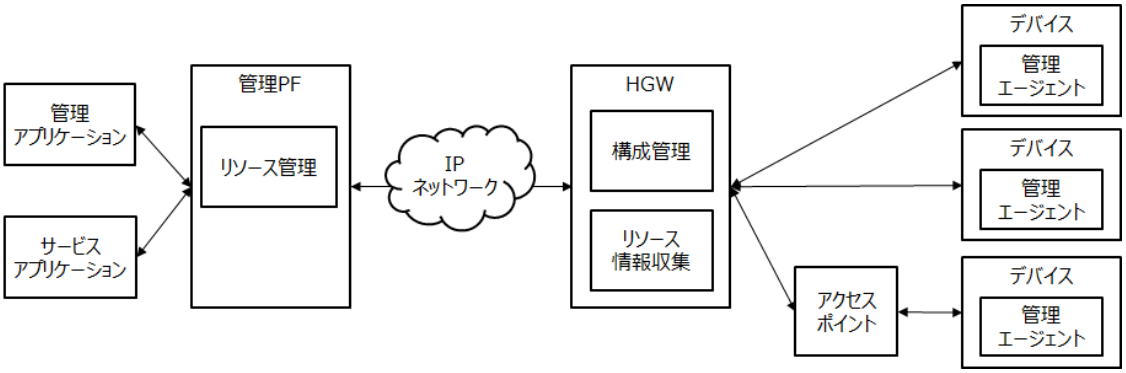


図 14　ベーシックデバイスの管理を実現する機能アーキテクチャ

１１　セキュリティサポート

本章では、HN サービス、特に HEMS に関するセキュリモデルと機能を説明する。HN に関する一般的なセキュリティの要件や技術は、[ITU-T X.1111]で説明されている。本標準は、WAN を介してデバイスとアプリケーション間で安全な通信を実現するために、[ITU-T X.1111]の技術を HN サービスに適用する。HEMS のセキュリティモデルは、11.1 で説明する。付録 III の HMES モデルに関するセキュリティの考慮の結果として、11.2 でセキュリティ機能アーキテクチャが示される。

11.1　HEMS のセキュリティモデル

本節では、図 4 に示された分散タイプのアーキテクチャに基づく HEMS のセキュリティモデルを、図 15 を用いて説明する。HEMS は HN サービスの 1 つのため、このモデルは一般的な HN モデルに適用可能である。

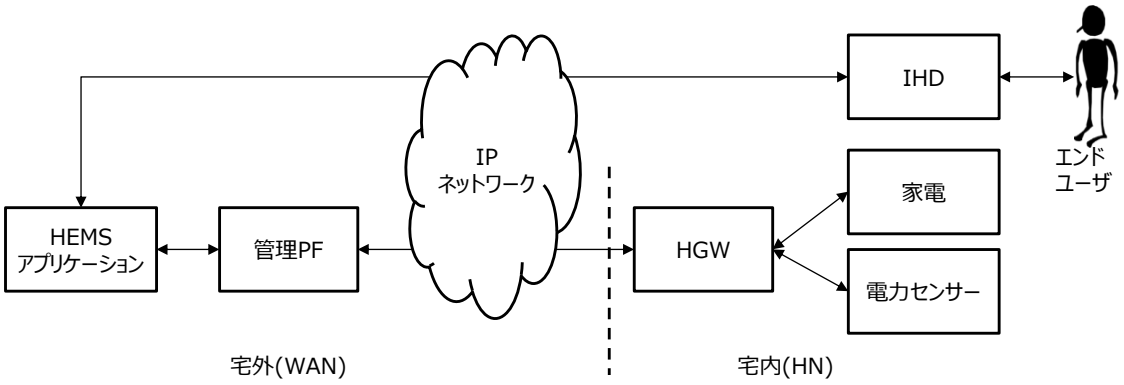


図 15　HEMS のセキュリティモデル

図 15 では、エンドユーザは、HEMS アプリケーションにセキュアな通信で接続するために、IDH 上の Web ブラウザを利用している。HEMS アプリケーションは、デバイスのデータ(例えば、家電のプロパティや電力センサーのプロパティ)を収集し、エンドユーザの指示に応じてデバイスを制御する。

HN のセキュリティは、[ITU-T X.1111]で規定されており、本標準では HN におけるセキュリティ技術のフレームワークとして[ITU-T X.1111]を参照する。

本モデルで扱うエンティティの機能は[ITU-T X.1111]のそれとは異なるため、最初に定義する。本モデル

で扱われるエンティティには、エンドユーザ、IHD、HEMS アプリケーション、管理 PF、HGW、家電機器や電力センサーのようなデバイスの 6 つがある。また、エンティティ間の関係としては、ユーザと IHD、IHD と HEMS アプリケーション、HEMS アプリケーションと管理 PF、管理 PF と HGW、HGW とデバイスの 5 つがある。

付録 III では、本モデルのセキュリティに関する考察が、[ITU-T X.1111]に記載される流れに基づいて行われている。最終的なモデルは、このモデルは表 III3 に示されている。

11.2　セキュリティ機能

デバイスは[ITU-T X.1111]で規定されるセキュリティ機能を備えることが期待される。図 16 は、[ITU.T X.1111]に基づき付録 III の考慮の結果から導かれたセキュリティ機能アーキテクチャである。図 16 では、実線と点線で示されたセキュリティ機能は、それぞれ実装必須と実装推奨を表している。付録 III の表 III.3 では 9 つのセキュリティ機能が規定されているが、図 16 のセキュリティ機能は[b-ISO/IEC27000]を考慮して、3 つの機能(耐可用性、メッセージ認証、エンティティ認証)に簡略化されている。耐可用性は、認証されていないエンティティの攻撃から防御するための機能である。メッセージ認証は、送信される情報の正当性と完全性を維持するために変更を防御するための機能である。エンティティ認証は、認証されていないエンティティがエンティティの情報を利用することを防御するための機能である。

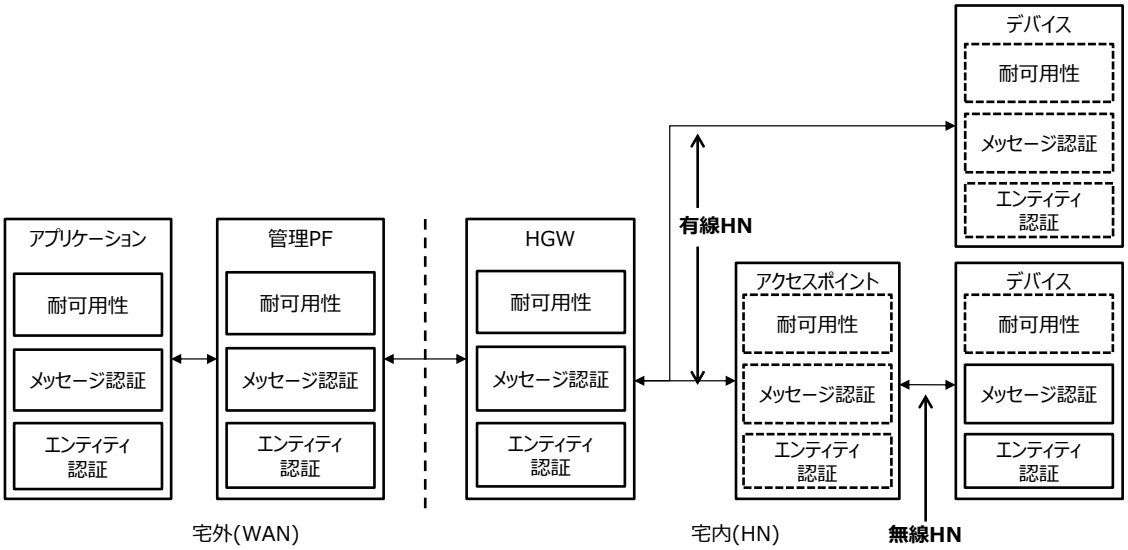


図 16　セキュリティ機能アーキテクチャ

家電やセンサー等の実際のデバイスは、演算性能が低いため、多くの場合は完全なセキュリティ機能を持つことができない。そのため、デバイスのセキュリティ機能の制限を補完する 2 つの解決策を以下に示す。1 つめは、管理 PF と HGW が、図 16 に示されるセキュリティ機能を実現することである。また、通常は WAN と HN 間に設置されるブロードバンドルーター(図 16 には記載されていない)は通常のファイアーウォール機能を持つ。このように、HGW に有線 HN で接続しているデバイスは、通常セキュリティ機能がなくても問題はなく、無線 HN で接続しているデバイスは、無線接続で要求される最小限のセキュリティ機能でも問題はない。したがって、図 16 に示されるように、デバイスとアクセスポイント間のメッセー

ジ認証機能とエンティティ認証機能がセキュアなワイヤレス接続のために必要となる。
【補足】図 16 に示されるように、で示されているアクセスポイント上のセキュリティ機能は有線 HN を介した HGW から必要とされるが、それらはオプション機能である。

もう 1 つの解決策は、9.3 で説明した管理 PF 内のリソース管理機能によって、HGW やデバイスに対して、以下のエンティティ認証機能を提供する。HN がセキュアに保たれる場合、これはエンティティ認証を提供するための単純で有効な手法である。

⑴HGW 認証

HGW の認証情報は、HGW が初めて管理 PF に接続するまでに、管理 PF 登録されている。HGW が最初に接続した際に、リソース管理機能が認証情報と事前に登録された情報を比較する。認証情報はリソース管理機能で管理される。

⑵デバイス認証

デバイスを特定する情報は、デバイスがセキュリティ機能を持たない場合には、HGW に接続するまでに、管理 PF に登録される。デバイス認証は管理 PF 内で事前に登録された情報と HGW に接続された場合に取得できるデータが一致するかどうかを調べる機能として提供される。

⑶デバイスのアクセス制御

物理的なデバイスに対するエンティティ認証機能は、管理 PF の仮想デバイスに対するアクセス制御機能として提供される。

付録 I　WoT に基づく機能配置モデル

[b-ITU-T Y.2063]では、WoT での物理的なデバイスは制約付きの(constrained)デバイスと完全な(fully-fledged)デバイスの 2 つのカテゴリに分類される。

・制約付きのデバイス：制約付きのデバイスはインターネットに接続することができず、Web の機能を持っていない。デバイスは WoT ブローカーのエージェントとインタラクションする。
・完全なデバイス：完全なデバイスは Web の機能を持つ。デバイスは WoT ブローカーだけではなく、Web 上のサービスとインタラクション可能である。

制約付きのデバイスと完全なデバイスは、デバイスオブジェクトを持っていないため、本標準では非ベーシックデバイスに分類される。制約付きのデバイスはアダプタを介して HGW、さらには管理 PF と通信する。完全なデバイスもまた、HGW を経由して、管理 PF と通信することができる。
本標準では、リソース情報収集機能を持つ HGW は、HN を通じて接続される制約付きのデバイスと完全なデバイスを含む、全てのデバイスを管理する。
[b-ITU-T Y.2063]は WoT ブローカーを定義しており、物理的なデバイスはアプリケーションから WoT ブローカーを通じて、Web リソースとしてアクセスされる。この機能アーキテクチャは、サービスレイヤとアダプテーションレイヤに分類され、本標準ではサービスレイヤのアーキテクチャは管理 PF、アダプテーションレイヤのアーキテクチャは HGW に対応する。
WoT ブローカーの Web アダプテーション機能は、物理的なデバイスと WoT サービス間の通信のため、通信プロトコルの Web プロトコルへのアダプテーションのみを行う。したがって、Web ブローカーは、本標準の P-D 参照点のみをサポートする。図 I.1 は、HGW は制約付きデバイスと完全なデバイスに接続された機能配置モデルを示す。

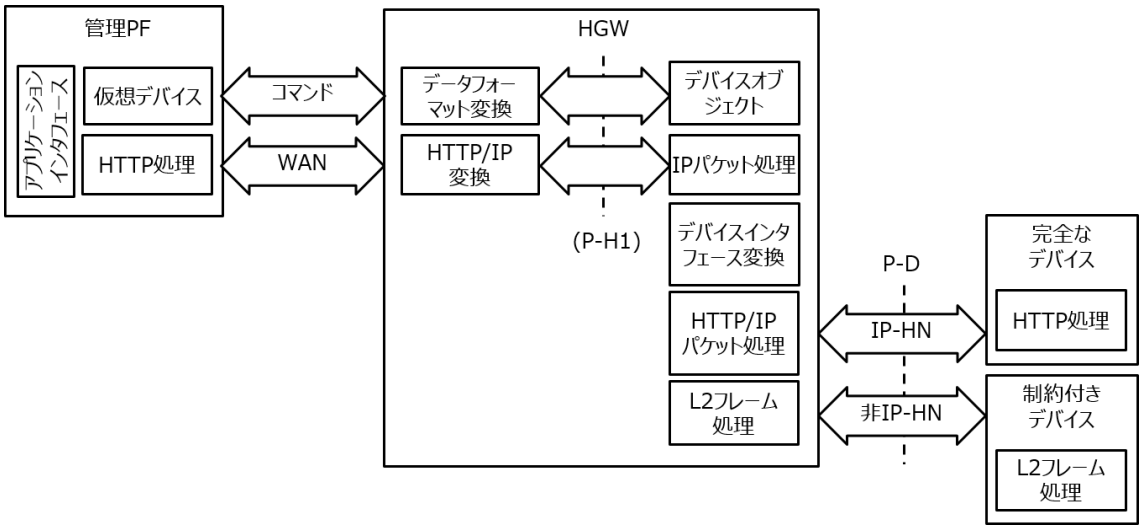


図 I.1　WoT の機能配置モデル

管理 PF はアプリケーション開発者に、Web ベースのアプリケーションインタフェースを通じて、Web リソースとして扱われる仮想デバイスを提供する。その結果、アプリケーション開発者は[b-ITU-T Y.2063]で定義された、WoT ブローカー内の WoT サービスと WoT ブローカー外部の Web サービスを融合したマッシュアップサービスのアプリケーションを開発することができる。
このように、本標準のアーキテクチャは、WoT で HEMS や他の HN サービスを提供する。

付録 II　HN アプリケーションの例

分散タイプのアーキテクチャは、様々な HN サービスに適用可能である。以下は、これらのサービスを実現する HN アプリケーションの例である。

(1)ホームセキュリティ

ホームセキュリティアプリケーションは、宅内に設置されたセンサーから脅威を検知し、警備員を派遣するために、セキュリティ会社に問題を通知する。不審者を検知する人感センサーや、家事を件とする火災センサー等のセンサーが、HGW や管理 PF を通じて、ホームセキュリティアプリケーションに接続される。

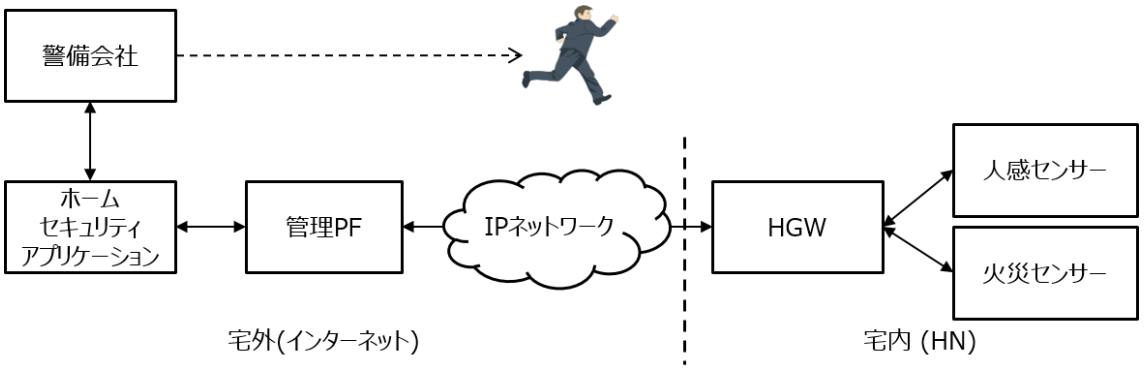


図 II.1　ホームセキュリティアプリケーションのアーキテクチャ

(2)デバイスへの正しいアクセス権制御を持ったカスタマーサポート

様々な種類のデバイスが接続される HN はますます複雑になっているため、カスタマーサポートは重要なサービスである。サービスが正しく機能しない場合、なぜ、どこで障害が起こっているかを把握することは困難である。図 II.2 で示された例は、企業 A、企業 B によって製造されたデバイス(家電)のそれぞれのサポートのために、2 つの異なるカスタマーサポートサービスが同じ管理 PF を通じて、別々に提供されている想定である。

2 つ以上の企業によって製造されたデバイスが HN に設置されている場合、ある企業に製造されたデバイスに関する障害情報を他の企業が入手できるという問題がある。ほとんどのデバイス企業は、他の企業がそのような情報を取得することを拒む。

図 II.2 には、管理 PF 上で、3 つのサービスが動作している。この場合、HN サービスプロバイダのアプリケーションは、管理 PF を通じて、ネットワーク切断のような診断情報を取得することができる。カスタマーサポートサービスは、サポート対象のデバイスに関する詳細な診断情報を取得する(例えば、企業 A のカスタマーサポートサービスは企業 A が製造したデバイスに関する詳細な障害情報を取得する)。このサービスは、管理 PF が適切なサービスのみへの情報配信機能を持っているため、実現可能である。

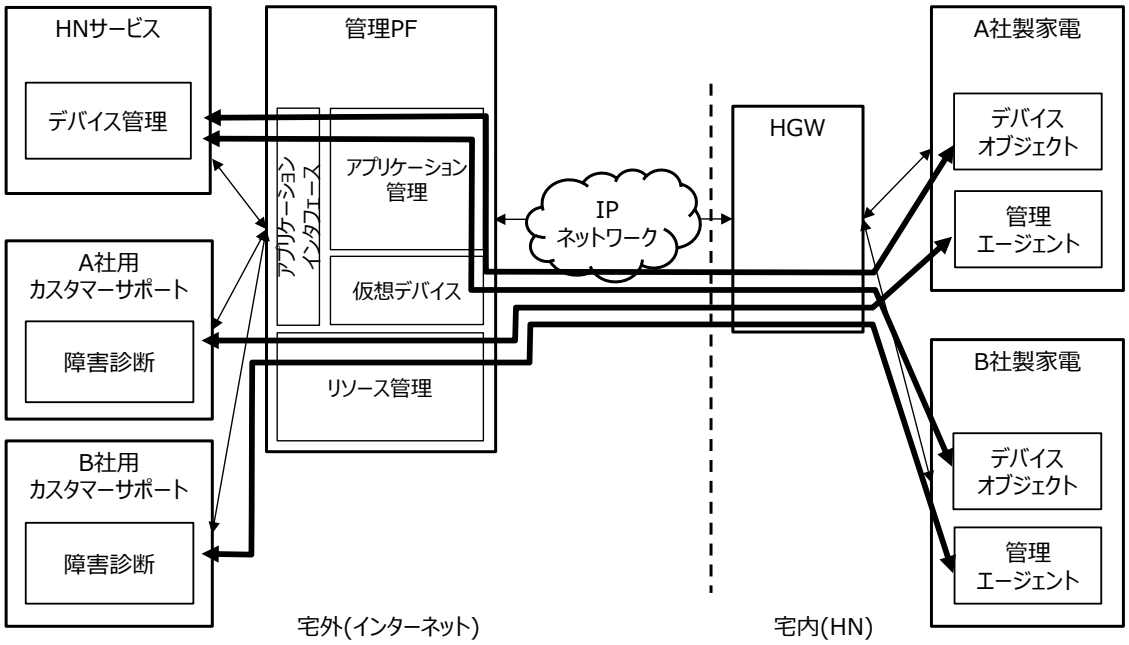


図 II.2　カスタマーサポートアプリケーションのアーキテクチャ

(3)より良い睡眠のための室内設備の調整

より良い睡眠のための室内設備調整サービスは、エアコンや照明機器を制御することで、部屋の温度、湿度、明るさを調整し、睡眠環境を適切に保ち続ける。ユーザ毎の良い睡眠の条件を取得するために、睡眠センサーは睡眠パターン、心拍数、呼吸数、ユーザのいびきを取得し、より良い睡眠のための条件を求める。

睡眠センサーの情報は非常にセンシティブなため、管理 PF は厳重に管理し、正しいサービスに通知する必要がある。この場合、睡眠モニターサービスは睡眠センサーから通知されたデータを独占的に取得し、より良い睡眠のための適切な環境条件に関する情報を、外部インタフェースを介して、室内設備調整サービスにメタデータとして提供する。その後、室内設備調整サービスは、エアコンや照明機器を制御する。

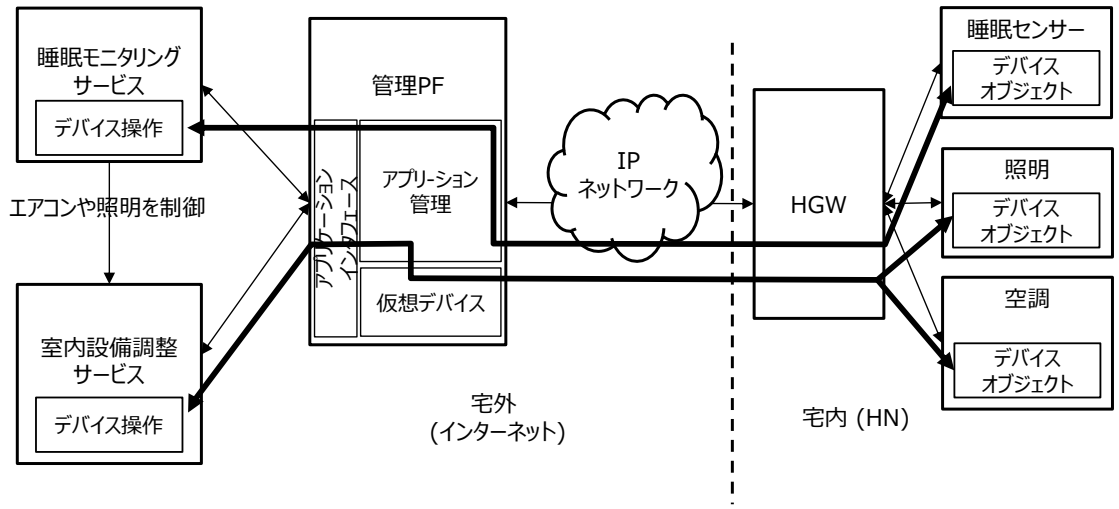


図 II.3　より良い睡眠のための室内設備調整アプリケーションのアーキテクチャ

付録 III　[ITU-T X.1111]に基づいたセキュリティに関する考慮

[ITU-T X.1111]はエンティティ、エンティティ間の関係、セキュリティ脅威、セキュリティ要件を定義している。また、それらに基づいたセキュリティ機能を説明している。図 15 に、エンドユーザからデバイスまでの 6 つのエンティティと、それらの間の 5 つの関係を示している。表 III.1 は、HEMS モデルに対するセキュリティ脅威の関係を示す。

表 III.1 HEMS モデルに対するセキュリティ脅威の関係

エンティティ・関係	一般的なセキュリティ脅威					
	情報開示/盗聴	妨害	変更/注入	非認証アクセス	否認	パケット異常転送
デバイス	Y	Y	Y	Y		
HGW	Y	Y	Y	Y		Y
MPF	Y	Y	Y	Y		
アプリ	Y	Y	Y	Y		
IHD	Y	Y	Y	Y		
User/IHD				Y		
IHD/アプリ	Y	Y	Y	Y		
アプリ/MPF	Y	Y	Y	Y		
MPF/HGW	Y	Y	Y	Y	Y	
HGW/デバイス	Y	Y	Y	Y	Y	

【補足】MPF とユーザは、それぞれ管理 PF とエンドユーザを示す。”エンティティあるいは関係”の列における”xxx/yyy”という表記は、xxx と yyy の間の関係を意味している。セル中の”Y”は、そのエンティティや関係に対して、脅威が存在することを示す。

[ITU-T X.1111]ではモバイル指向のセキュリティ脅威が記述されているが、この脅威は HN アプリケーションからデバイスへのセキュリティにフォーカスしている本標準ではスコープ外である。表 III.2 は、HEMS のセキュリティ要件間の関係、[ITU-T X.1111]で記載されているセキュリティ要件に基づく脅威と機能を示している。

表 III.2 HEMS のセキュリティ要件、脅威、機能の関係

セキュリティ要件	一般的なセキュリティ脅威	セキュリティ機能
データ機密性	情報開示/盗聴 非認証アクセス	暗号化 アクセス制御 鍵管理

データ完全性	変更/注入 パケット異常転送	完全性 MAC デジタル署名 公証 鍵管理
認証	情報開示/盗聴 妨害 変更/注入 非認証アクセス 否認	MAC デジタル署名 公証 鍵管理
否認不可	否認	デジタル署名 公証 鍵管理
アクセス制御/認証	情報開示/盗聴 妨害 変更/注入 非認証アクセス	暗号化 MAC エンティティ認証 デジタル署名 アクセス制御 鍵管理
可用性	妨害	MAC エンティティ認証 デジタル署名 アクセス制御 鍵管理 耐可用性
プライバシセキュリティ	情報開示/盗聴	暗号化 MAC エンティティ認証 デジタル署名 アクセス制御 鍵管理
通信フローセキュリティ	パケット異常転送	完全性 MAC エンティティ認証 アクセス制御 鍵管理

セキュリティ機能間の関係と、HEMS モデルのセキュリティ(図 15)は、表 III.2 に記載されたセキュリティ機能に基づいて、表 III.3 に含まれる。

表 III.3 セキュリティ機能とモデルの関係

エンティティ・関係		セキュリティ機能								
		暗号化	完全性	MAC	エンティティ認証	デジタル署名	公証	アクセス制御	鍵管理	耐可用性
蓄積データ	デバイス	Y	Y	Y	Y	Y	-	-	-	Y
	HGW	Y	Y	Y	Y	Y	Y	Y	Y	Y
	MPF	Y	Y	Y	Y	Y	Y	Y	Y	Y
	アプリ	Y	Y	Y	Y	Y	Y	Y	Y	Y
	IHD	Y	Y	Y	Y	Y	Y	Y	Y	Y
通信データ	ユーザ/IHD	Y	-	-	Y	Y	-	-	Y	Y
	IHD/アプリ	Y	Y	Y	Y	Y	-	Y	Y	Y
	アプリ/MPF	Y	Y	Y	Y	Y	Y	Y	Y	Y
	MPF/HG	Y	Y	Y	Y	Y	-	Y	Y	Y
	HGW/デバイス	Y	Y	Y	-	-	-	-	-	Y

【補足】MPF とユーザは、それぞれ管理 PF とエンドユーザを示す。”エンティティあるいは関係”の列における”xxx/yyy”という表記は、xxx と yyy の間の関係を意味している。セル中の”Y”は、セキュリティサービスが対応するセキュリティ機能によって提供されることを示す。

表 III.3 のセキュリティ機能とモデル間の関係を考慮した、本標準で規定されたアーキテクチャにおけるエンティティに必要なセキュリティ機能が図 16 に示されている。
セキュリティ機能は、[b-ISO/IEC 27000]を考慮し、図 16 では 3 つの機能に単純化されている。それらは、耐可用性、メッセージ認証、エンティティ認証機能である。暗号化や鍵管理のような共通機能は省略されている。他の機能は、この 3 つの機能にマージされている。これらの機能はそれぞれ[b-ISO/IEC 27000]で規定された可用性、完全性、機密性という情報セキュリティ要件に対応している。

【補足】[b-ISO/IEC 27000]は、情報セキュリティ管理システムを規定している。その要件は以下のように定義されている。

・**可用性**：認証済みのエンティティによる要求に対して、アクセス・利用可能な特性
・**完全性**：正確であり改竄されていないこと
・**機密性**：認証されていない個人、エンティティ、プロセスに、情報が利用できないあるいは開示されないこと

参考文献

[b-BACnet] ANSI/ASHRAE, *Standard 135-2004*.

[b-BBF TR-069] Broadband forum, *TR-069 Amendment, 4CPE WAN Management Protocol.*

[b-BBF TR-181] Broadband forum, *TR-181 Issue 2 Amendment 6, Device Data Model for TR-069.*

[b-ECHONET Lite] ECHONET Consortium, *ECHONET Lite Specification Version 1.10.*

[b-FG-Smart Terminology] ITU-T FG Smart Deliverable (2011), *Smart Grid Terminology*.

[b-ISO/IEC 14543-3-x] ISO/IEC 14543-3-x-series (2006), *Information technology Home electronic system (HES) architecture*.

[b-ISO/IEC 27000] ISO/IEC 27000 (2014), *Information technology – Security techniques – Information security management systems – Overview and vocabulary*.

[b-ITU-T J.190] Recommendation ITU-T J.190 (2002), *Architecture of MediaHomeNet that supports cable-based services*.

[b-ITU-T Y.2063] Recommendation ITU-T Y.2063 (2012), *Framework of the web of things*.

[b-ITU-T Y.2720] Recommendation ITU-T Y.2720 (2009), *NGN identity management framework*.

[b-SEP 2.0] ZigBee Alliance, *Smart Energy Profile 2.0 Application Protocol.*

[b-W3C WACterms] W3C (1999), *Web Characterization Terminology & Definitions Sheet*.