

研究開発用の M2M/IoT プロトタイプ用 アーキテクチャ推薦・評価システムの提案

山崎啓佑^{†1} 中島毅^{†1}

概要：近年、M2M/IoT システムというあらゆるものにセンサやタグを取り付け、それらの取得情報をネットワークに送信することで様々なサービスを提供するシステムの開発が盛んに行われている。しかし、一般に M2M/IoT 初学者には要求を満たす M2M/IoT システムを構築することは難しい。その要因の一つとして、M2M/IoT システムを構築するアーキテクチャの選択の困難さがある。これは、M2M/IoT の構成要素がソフトウェア、ハードウェア、通信技術など多岐にわたるため、これらの中のトレードオフを考慮した選択を行うには各分野の幅広い知識が必要なためである。そこで、本研究では初学者の開発を支援するために、研究開発用の M2M/IoT プロトタイプを対象としたアーキテクチャ推薦・評価システムの提案を行う。

キーワード：システムアーキテクチャ, M2M, IoT

Proposal of architecture recommendation and evaluation system for M2M/IoT prototype for research and development

KEISUKE YAMAZAKI ^{†1} TSUYOSHI NAKAJIMA ^{†1}

Abstract： In recent years, development of systems that provide various services by attaching sensors and tags to everything called M2M/IoT system and sending these obtained information to the network is being actively conducted. However, it is generally difficult for M2M/IoT primary scholars to build an M2M/IoT system that satisfies the requirements. One of the factors is the difficulty in choosing the architecture for constructing the M2M/IoT system. This is because the components of M2M/IoT range from software, hardware, communication technology and so on, so that a wide range of knowledge in each field is necessary to make a selection considering the trade-off between them. Therefore, in this research, we propose architecture recommendation and evaluation system for M2M/IoT prototype for research and development in order to support development of primary scholars.

Keywords： System Architecture, M2M, IoT

1. はじめに

M2M(Machine to Machine)/IoT(Internet of Things)とは、機械や機器、設備や備品などのあらゆるものにセンサやタグを取り付け、それらから得られる情報をネットワークに送信することで、機械の自動制御や監視、データ分析などの様々なサービスを提供するものである。これらの需要は年々高まっており、2020年には M2M/IoT システムに必要なセンサやアクチュエータなどの M2M/IoT デバイスの数が 200 億個に増加すると考えられている[1]。

一方で、M2M/IoT の開発を経験したことがない初学者にとって、M2M/IoT システム開発は難しく、機能要求を満たすシステムの構築を行えない場合や、機能要求を満たすシステムを構築できたとしても品質が悪い場合がある。なぜならば、M2M/IoT システムの開発には組み込み技術、ネットワーク技術、ソフトウェア技術、ハードウェア技術等の幅広い知識や理解や、それらに基づいた機能や品質のトレードオフを考慮したアーキテクチャの設計を行わなければならないからである。

この問題の解決策として、M2M/IoT システムのプロトタイプ

構築学習法[2]がある。この学習法は、M2M/IoT 初学者が、特定の機能と構成をもつ M2M/IoT システムのプロトタイプを手順通りに作成するによって、M2M/IoT システムの基本的な構成や各構成要素の動作を理解し、M2M/IoT システムを開発するための動機付けと基礎知識を身につけることを目標とした学習法である。

しかし、この学習法は、M2M/IoT システムの設計の考え方を省いており、また、アーキテクチャに関する獲得知識もプロトタイプのものに限定される。そのため、初学者が、異なる要求をもつ M2M/IoT システムを設計・構築する場合に、アーキテクチャの設計・選択において困難に直面する。無理にプロトタイプと同種のアーキテクチャを適用しようとすると、機能要求を満たさない、または品質が悪い M2M/IoT システムを構築してしまうことにつながる。

そこで、本研究では M2M/IoT 初学者が研究開発の目的でプロトタイプを作成する際に、開発したい M2M/IoT システムの要求と制約条件及びシステムで行われる処理を入力することで、インタラクティブに M2M/IoT アーキテクチャの推薦及びその品質を評価するツールの提案を行う。

以下、2 節で M2M/IoT システムの設計における課題、3 節で

^{†1} 芝浦工業大学
Shibaura Institute of Technology

課題解決に関する従来研究を述べ、4節で提案システム、5節でその使用例を示す。最後に6節でまとめと今後の課題について述べる。

2. M2M/IoT システムの設計課題

図1にM2M/IoTシステムの基本的な構成を示す。各要素の役割として、M2M/IoT デバイスはセンサデータの取得や送信、アクチュエータの制御等を行い、中継デバイスはセンサネットワークとインターネットをつなげる役割を持ち、クラウド・サーバはM2M/IoT デバイスが集めたデータの処理、分析を行い、その結果からM2M/IoT デバイスの制御やユーザーに対して有益な情報の提供を行う。しかし、このようなシステムを設計する場合には二つの課題がある。

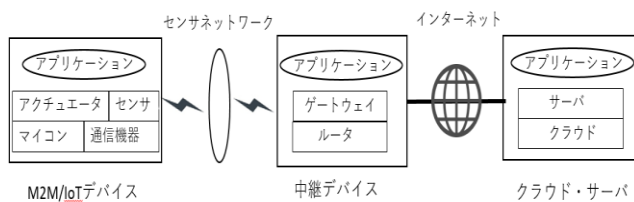


図1 M2M/IoT 基本構成図

第一の課題は、処理の分散配置なのである。基本的に、M2M/IoT システムは、費用や保守性の面から見れば、センサ情報の取得や通信の中継などの各要素固有の処理以外の処理を、クラウド・サーバ側で実装することが望ましい。しかし、すべての処理をクラウド・サーバ側で実装してしまうと、高速なデータ処理や分析、素早い応答などの要求がクラウド・サーバのスループットやネットワークトラフィック等の制約から満たせない場合がある。この問題を解決するために、フォグコンピューティング[3]やエッジコンピューティング[4]と呼ばれる、クラウド・サーバ以外のデバイスに処理を分散させる方法がある。しかし、処理を分散させると、M2M/IoT システム全体のコストの増加や保守性の低下がある。そのため、M2M/IoT システムの設計では、処理の分散配置に関するトレードオフを行う必要がある。

第二の課題は、M2M/IoT システムの構成要素の選択である。構成要素とは、M2M/IoT システムに使用されるハードウェア、基本ソフトウェア、ミドルウェア及び通信規格のことである。M2M/IoT システムの構成要素の選択には数多くの選択肢がある。センサネットワークに使用する無線規格だけを考えても、ZigBee, Bluetooth, Wi-Fi, Z-Wave, LoRa 等、様々な規格が存在しており、それぞれの特性を知らなければ開発する M2M/IoT システムの要求に対して適切な無線規格を選択することができない。不適切な構成要素を選択してしまうと、要求機能・品質を満たさないシステムを開発してしまうことにつながる。

上記の課題を解決し要求にあった M2M/IoT システムを、

初学者が支援なく設計することはハードルが高い。そのため、適切な処理分散のあり方と構成要素の選択を支援する、M2M/IoT システムのアーキテクチャに関する意思決定支援の機能が必要である。

3. 従来研究

3.1 アーキテクチャ推薦・評価システム

ソフトウェア・アーキテクチャの推薦システムとして、Archreco [6]がある。このシステムは、初学者を対象ユーザーとして、ユーザーが開発しようとしているソフトウェアに適したデザインパターンを推薦してくれる対話的な設計支援システムである。

我々が提案するシステムも、対話的に設計を行うことにより M2M/IoT システムのアーキテクチャ推薦及びアーキテクチャの品質評価を行うことを目指す。

3.2 M2M/IoT システム開発ツール

M2M/IoT の開発を補助するツールやフレームワークは数多く存在する。代表的なものに Node-RED[3]があり、用意されたコンポーネントを視覚的につなぎ合わせることで、M2M/IoT システムを開発することができる。

しかし、こうした開発ツールは、通信や簡単なソフトウェア処理の実装を（その詳細を隠して）可能にするツールであり、適切なアーキテクチャの選択などの設計に関する補助は行っていない。そのため、開発した成果物が要求を満たさないことが起こり得る。

3.3 M2M/IoT アーキテクチャ設計支援

M2M/IoT 初学者の設計支援に関する研究では、M2M/IoT システムのものづくりにおけるプロトタイプ構築学習法 [2]がある。この学習法は、機能と構成が決められた M2M/IoT システムのプロトタイプを手順通りに作成することにより、初学者が M2M/IoT システムの基本的な機能や各要素の役割を理解することを助ける。

しかし、M2M/IoT 初学者が作成したプロトタイプと異なる種類の M2M/IoT システムを作成する場合には、制約や要求が異なるため、この学習法をそのまま適用することができない点に課題がある。

4. 提案システム

4.1 提案システムの機能

本研究では、M2M/IoT システムの設計における課題である、処理の分散配置と構成要素の選択の解決方法として、M2M/IoT システムアーキテクチャ推薦・評価システムを提案する。

提案システムは、M2M/IoT 初学者を対象ユーザーにしており、入力としてユーザーが開発したい研究開発用の M2M/IoT システムの制約条件、要求、M2M/IoT システム上で行われる処理、M2M/IoT デバイスの振る舞いを受け取り、ユーザーと対話的な設計・評価を進める。ユーザーは開発したい M2M/IoT シ

システムに適切なアーキテクチャを設計することができる。
図 2. に提案システムの概念図を表す

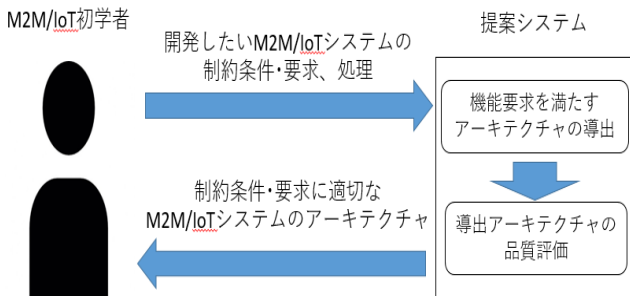


図 2 提案システムの概念図

ここで、研究開発用の M2M/IoT プロトタイプとは、Arduino や Raspberry Pi, Linux などのオープンなソフトウェア/ハードウェアを用いて開発を行うことにより、機能要求の確認や品質の確認、動作の検証を行うための試作品を指す。また、M2M/IoT 初学者とは、M2M/IoT の関連領域の知識はあり、簡単なソフトウェアの開発は可能だが、M2M/IoT システムを構成するソフトウェア以外の要素(組込み技術、ネットワーク技術、ハードウェア技術など)に関する知識が足りないために、2 節で述べた課題を解決できず、機能要求を満足する M2M/IoT システムの開発や、品質の良い M2M/IoT システムの開発を行えないような研究者を指す。

4.2 システム実現上の課題

本システムの設計をするにあたり、ユーザからどのような順番及び手順で制約条件や要求、処理を入力するのが課題となった。そこでこの課題の解決策として、4.3 利用シナリオ及び 4.4. 要求の入力に示すようにシステムとの対話的な制約条件、要求の入力及び、UML による処理の記述を用いることにした。

4.3 利用シナリオ

提案システムは以下の手順で利用されることを想定している。

S1 : 要求事項の入力

提案システムに対して、ユーザは作成したい M2M/IoT システムの制約条件と要求、M2M/IoT システム上で行われる処理及び M2M/IoT デバイスの振る舞いを入力する。

S2 : アーキテクチャ推薦

システムはユーザから入力された情報をもとにアーキテクチャの候補を複数導出、ユーザに対してそれぞれの特徴を品質とともに推薦する。

S3 : アーキテクチャ選択

ユーザは推薦された候補の中から自分の要求に合うアーキテクチャを選択する。

S4 : 詳細要求入力

ユーザは選択したアーキテクチャを詳細にするために、システムに対してさらに要求と制約条件を入力する。

S5 : 選択の繰り返し

S2 から S4 をアーキテクチャが出力されるまで繰り返す。ただし、S4 を行わない場合もある。

S6 : アーキテクチャ決定

最終的に、ユーザは作成したい M2M/IoT システムのアーキテクチャ及びその品質をシステムの出力として得ることができる。

4.4 要求事項の入力方法

利用シナリオの S1 : 提案システムへの入力において、情報を入力する手順とその方法を説明していく。

S1-1 : 環境情報の入力

まず、ユーザは M2M/IoT デバイスを設置する環境の数と名前(ここで言う環境とは、M2M/IoT デバイスが設置される場所のことである)を入力表に入力。その後、各環境の情報として、移動タイプ(環境が固定されているか、狭域移動をするか、広域移動をするか)とインターネットに接続可能かどうか、同じような環境がいくつあるか、及び、環境内に存在する M2M/IoT デバイスの名前、種類(センサデバイスであるか、アクチュエータデバイスであるか、センサデバイスかつアクチュエータデバイスであるか)、個数を入力表に入力する。また、図 3 は環境情報の入力の関係をクラス図として表したものである。

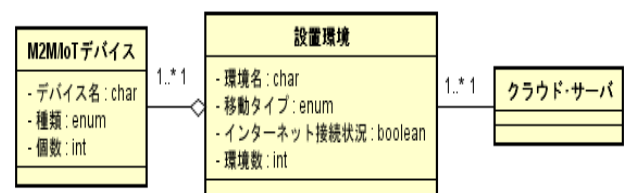


図 3 環境情報クラス図

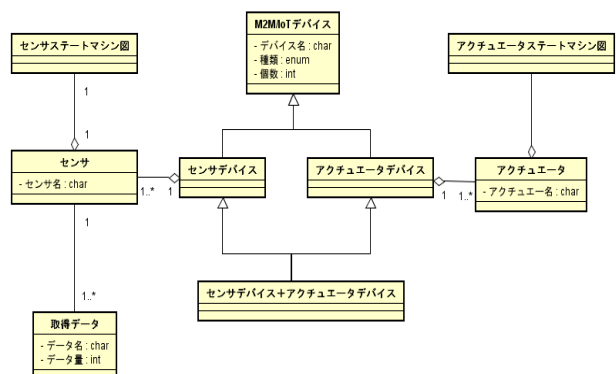


図 4 M2M/IoT デバイス情報クラス図

S1-2 : M2M/IoT デバイス情報の入力

次に,ユーザは各環境に設置する M2M/IoT デバイスの情報を入力表に入力する.M2M/IoT デバイスがセンサデバイスならば,センサデバイスが持っているセンサ及びそのセンサの取得データとデータ名,データ量を記入する.アクチュエータデバイスならば,アクチュエータデバイスに属しているアクチュエータの名前を入力する.

S1-3 : ステートマシン図の入力

次に,ユーザは各センサとアクチュエータの振る舞いをステートマシン図に記述する.表形式ではなく,ステートマシン図として記述する理由としては,

- ・表形式で入力するよりも図で入力するほうが視覚的,直感的にわかりやすいため,入力がしやすく,入力時のミス減らすことができる点.
- ・図として入力するので,入力に誤りがあった場合に修正することが容易である点.

以上二点があげられる.また,図 4 は M2M/IoT デバイスへの入力情報の関係をクラス図として表したものである.

S1-4 : 処理の入力

次に,ユーザは M2M/IoT システム上で行われる処理の入力を行う.入力情報として,処理の名前,処理の緊急性,処理の計算量を入力する.

S1-5 : シーケンス図の入力

次に,ユーザは処理の順番や条件,依存関係などの情報を入力するためにシーケンス図を記述する.シーケンス図の書き方としては,まず,ライフラインとして S1-3 で入力したデバイスの状態と,S1-4 で入力した M2M/IoT システム上での処理をライフラインとして定義するが,処理に関係のあるオブジェクトを新たにライフラインとして定義することもできる.新たにライフラインを定義する場合には,そのライフラインが M2M/IoT システムの内部に存在するのか外部に存在するのかも記述し,内部に存在するものならば,その場所も記述する.シーケンス図の記述方法としては S1-4 で入力した M2M/IoT システムの処理ごとにシーケンス図を作成する.また,シーケンス図の記述方法のルールとしては各処理のシーケンス図のメッセージは必ず各処理に対してメッセージを送信/受信されてなければならない.例えば,ある処理 A のシーケンス図を記述する際には処理 A にメッセージを送信するもの,もしくは処理 A からメッセージを受信するものしか処理 A のシーケンス図に記入することはできない.図にすると,処理 A のシーケンス図を記述する際に,図 5 のようなシーケンス図は入力として適切だが,図 6 のようなシーケンス図は入力として不適切である.図 6 の正しい書き方は,処理 A と処理 B のみを処理 A のシーケンス図として記述する書き方である.

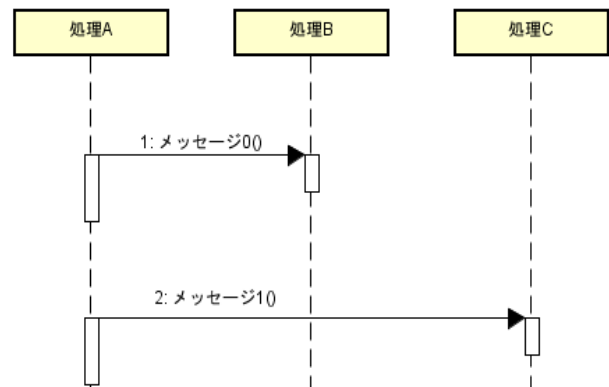


図 5 処理 A の記述として適切なシーケンス図

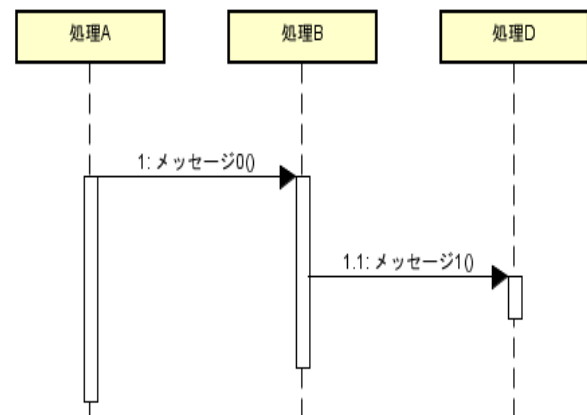


図 6 処理 A の記述として不適切なシーケンス図

シーケンス図をこのように記述させる事により,以下の効果が見込める.

- ・処理ごとに必要な情報や他の処理との依存等を吟味しながら入力を行うことができる.
- ・視覚的にわかりやすいので,ユーザが容易に入力することができる.

以上の S1-1 から S1-5 を行うことで,S1 : 提案システムの入力における M2M/IoT システムの制約条件と要求,M2M/IoT システム上で行われる処理及び M2M/IoT デバイスの振る舞いの入力完了する.

5. 提案システム使用例

提案システムの使用例として,車内見守りシステム[7]のアーキテクチャ推薦・評価がどのように行われるかを記述する.車内見守りシステムとは,子供やペットの車内放置による熱中症と車への車上荒らしの対策として考案された M2M/IoT システムである.このシステムの制約条件,要求及びシステム上の処理を以下に記す.

- ・車に温湿度センサデバイス(温度センサと湿度センサ搭載),人感センサデバイス,音センサデバイス,アラームデバイスを搭載
- ・熱中症対策として,人感センサで車内に人,もしくは動物がいるかの判定を行い,人もしくは動物がいた場合,

温度センサと湿度センサの値から WBGT(熱中症を予防することを目的として提案された指数[8])を計算、閾値を超えていた場合にはシステム利用者に車内の子供/動物が危険であるというメールを送信する。

- ・車上荒らし対策として、人感センサ及び音センサに反応があった場合には、車上荒らしが行われていると判断し、車のアラームを鳴らし、車上荒らしが行われていることを知らせるメールをユーザのスマートフォンにメールを送信する。また、アラームはユーザのスマートフォンから止めることができる。

以上の制約条件、要求、システム上の処理を本提案システムに入力し、出力としてアーキテクチャが得られるまでの流れを利用シナリオの流れに沿って示す。

S1 : 提案システムの入力

第4章で示した手順どおりに入力する。

S1-1 : 環境情報の入力

図2の項目を入力していく。まず、環境情報として、提案システムの設置環境名は車、移動タイプは広域移動、環境数が1000、環境内に存在するデバイスとして、温湿度センサ、音センサ、人感センサ、アラームと及びその属性を入力すると、図7示す情報が得られる。

S1-2 : M2M/IoT デバイス情報の入力

次に、ユーザは環境に設置する各デバイスの詳細情報を表形式で入力する。車内見守りシステムでは、温湿度センサデバイス、音センサデバイス、人感センサデバイス、アラームデバイスを車に設置する。入力形式及び、車内見守りシステムでの入力を表1に示す。

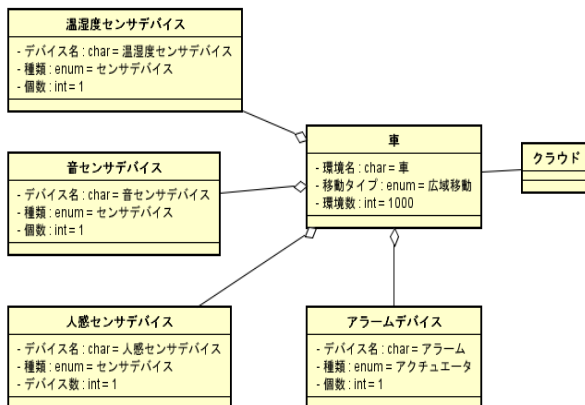


図7 S1-1 の入力から得られる情報

表1 車内見守りシステムの M2M/IoT デバイス情報

デバイス名	種類	センサ/アクチュエータ名	デバイス数	取得データ	データ量
温湿度センサデバイス	センサデバイス	温度センサ	1	温度データ	20byte
湿度センサデバイス	センサデバイス	湿度センサ	1	湿度データ	20byte
音センサデバイス	センサデバイス	音センサ	1	音データ	200byte
人感センサデバイス	センサデバイス	人感センサ	1	人感データ	20byte
アラームデバイス	アクチュエータデバイス	アラーム	1	-	-

S1-3 : ステートマシン図の入力

次に、各センサ、アクチュエータのステートマシン図を記述する。車内見守りシステムでは、温度センサ、湿度センサ、音センサ、人感センサ、アラームのステートマシン図を記述する。図8、図9にステートマシン図の記述例として温度センサとアラームのステートマシン図を記す。温度センサは60秒周期で温度データを取得することを表し、アラームはアラーム ON イベントが起こるとアラーム ON 状態に遷移し、アラーム OFF イベントが起こると、アラーム OFF 状態に遷移することを表す。

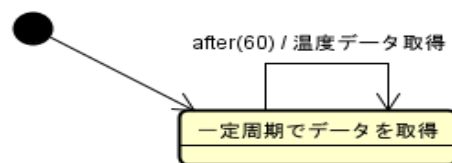


図8 温度センサステートマシン図

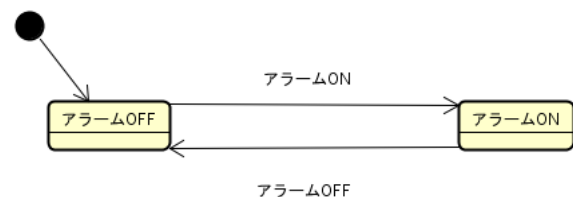


図9 アラームステートマシン図

S1-4 : 処理の入力

車内見守りシステム上で行う処理を処理の名前、処理の緊急性、処理の計算量とともに記述する。車内見守りシステムでは表2のようになる。

表2 車内見守りシステムの処理表

ライフライン名	場所
利用者のスマートフォン	システム外部
温度、湿度、WBGT データベース	クラウド・サーバ
利用者情報データベース	システム内部

S1-5 : シーケンス図の入力

次に,処理ごとにシーケンス図を記述していく.S1-3で入力された状態と処理をライフラインとしてシーケンス図を記述するが,それら以外の要素をライフラインとして用いたい場合には,表 3 のようにライフライン名とシステムの外部か内部,にあるかを記述することでライフラインとして用いることができる.これらの要素を組み合わせ,処理ごとにシーケンス図を記述していく.記述例としてメール通知(熱中症)の処理を図 10 に記す.以上で S1-5 が終了するので,S1 : 提案システムの入力が終了する.続いて,S2:アーキテクチャ推薦[一回目]を行う.

表 3 車内見守りシステムのライフライン定義

処理名	緊急度	計算量(オーダー)
アラーム起動	高い	1
メール通知(熱中症)	高い	1
メール通知(車上荒らし)	高い	1
WBGT 計算	高い	1
温度,湿度,WBGT 保存	低い	1

S2 : アーキテクチャ推薦[一回目]

システムは,ユーザに入力された情報をもとに,アーキテクチャの推薦を行う.アーキテクチャ推薦の一回目では,処理の分散配置についての推薦を行う.車内見守りシステムの例では,3種類のパターンが以下のように表示されるとともに,車内見守りシステム上で行われるそれぞれの処理がどこに配置されるかを記した配置図を出力する.

- ・パターン 1 : クラウド・サーバ処理パターン
特徴 : クラウド・サーバですべての処理を行う.
利点 : 実装が容易であり,保守性が高い.
欠点 : クラウド・サーバですべての処理を行うのでスループットの問題やネットワークトラフィックが課題となる可能性が高い.
- ・パターン 2 : フォグコンピューティングパターン
特徴 : 中継デバイス上で主要な処理を行う.
利点 : 中継デバイスで殆どの処理をするためにネットワークトラフィックやスループットが問題にならない場合が多い.
欠点 : クラウド・サーバ上ですべての処理を行う場合に比べてシステムが複雑になる.

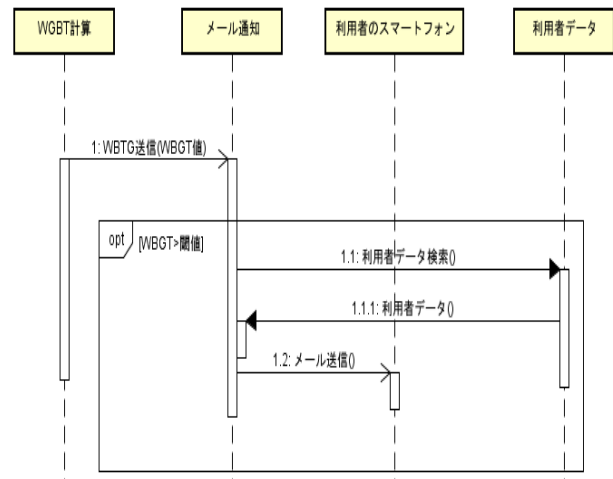


図 10 メール通知(熱中症)シーケンス図

- ・パターン 3 : エッジコンピューティングパターン
特徴 : M2M/IoT デバイス上で主要な処理を行う.
利点 : M2M/IoT 上でほとんどの処理を行うため,通信遅延による処理の実行の遅れが発生しにくい
欠点 : M2M/IoT デバイス上で処理を行わせるために他のパターンに比べて M2M/IoT デバイスが高性能でなくてはならないのでコストが高くなる.

S3 : アーキテクチャ選択[一回目]

ユーザは提示されたパターンの中から,自分の要求に合うパターンを選択する.今回の例では,パターン 2 : フォグコンピューティングパターンを選択したとする.

S4 : 詳細要求入力[一回目]

今回は行わない.その為,S5 を実行する

S5 : 選択の繰り返し[一回目]

S2 : アーキテクチャ推薦の二回目を実行

S2 : アーキテクチャ推薦[二回目]

次に,システムはフォグコンピューティングパターンにおける M2M/IoT デバイスと中継デバイス間の通信パターンの推薦を行う.今回の例では,提案システムは以下の 2 つのパターンを推薦する.また,それぞれのパターンがどのような構成になるかを図 11 と図 12 に記す.

・パターン 1 : LPWA パターン

- 特徴 : LPWA(Low Power Wide Area)と呼ばれる長距離無線通信に分類される無線規格を使用する.
- 利点 : 低消費電力で長距離通信が可能.
中継デバイスを基地局のように扱える.
- 欠点 : 低速で通信を行うので,緊急時に対応できない

可能性が高い。基本的には送信のみの片方向通信しかできない場合が多い。

・パターン2：近距離通信+インターネット通信パターン

特徴：環境にゲートウェイを搭載し、通信を行う。

利点：M2M/IoT デバイス間とゲートウェイ間でパターン1より高速な通信ができる。パターン1よりも複雑なネットワーク・トポロジーを取ることができ、障害に強い。

欠点：ゲートウェイを各環境に設置するコストがかかる。

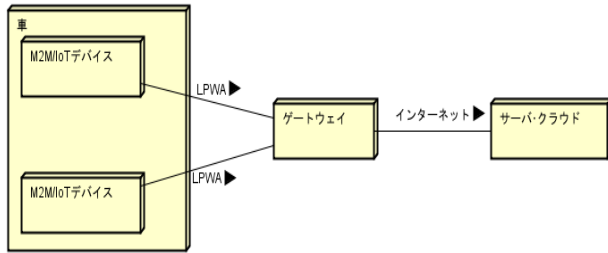


図 11 LPWA パターン図

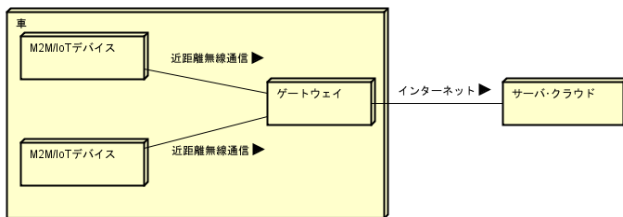


図 12 近距離通信+インターネット通信パターン図

S3：アーキテクチャ選択[二回目]

今回の選択では、ユーザはパターン2を選択したとする。

S4：詳細要求入力[二回目]

ユーザは次に、M2M/IoT システムで使用する通信規格についての要求を入力する。今回の場合は、環境の中で使用する近距離無線通信規格を選択するために、無線規格への要求を入力する。入力する項目は、以下ようになる。

- ・通信方法が TCP/IP 通信であるか
- ・最大同時接続数はいくつならばよいか
- ・周波数はいくつか
- ・ネットワーク・トポロジーは何を使用するか

これらの情報をその情報の概要(例えば、周波数は電波干渉や通信距離に関係しているなど)を提示し、ユーザはその情報をもとに要求を入力する。今回の例では表4のような入力を下と仮定する

S5：選択の繰り返し[二回目]

S2：アーキテクチャ推薦の三回目へ

表 4 車内見守りシステムの無線通信規格要求表

通信方法	TCP/IP 通信をしない
最大同時接続数	10
周波数	指定なし
ネットワーク・トポロジー	メッシュ

S2：アーキテクチャ推薦[三回目]

提案システムは、システムに保管されている近距離無線通信規格情報のリポジトリから、先程の S4-2 で入力した情報と、これまでに入力された情報から、条件を満たす規格をすべて推薦する。今回のレでは、推薦システムは、ZigBee と Z-Wave という無線規格を推薦する。

S3：アーキテクチャ選択[三回目]

ユーザは自分のシステムにどの通信規格がふさわしいかを、推薦された通信規格を比較して決定する。車内見守りシステムでは、ZigBee を選択したものとする。

S4：詳細要求入力[三回目]

入力を行わず、S5：選択の繰り返しへ。

S5：選択の繰り返し[三回目]

これまでの入力で、アーキテクチャが決定したので、繰り返し終了 S6：アーキテクチャ決定へ。

S6：アーキテクチャ決定

これまでのユーザとシステムの対話から、車内見守りシステムのアーキテクチャが決定した。そのアーキテクチャを図13に示す。また、決定したアーキテクチャに対して、性能効率性、信頼性、セキュリティ、保守性、移植性の面から評価し、その結果を出力するアルゴリズムを現在検討中である。

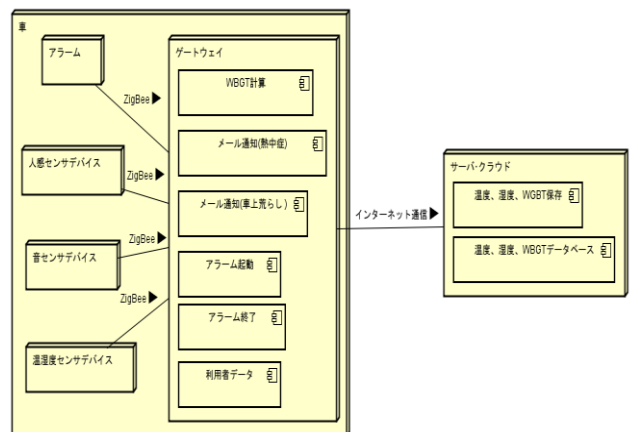


図 13 車内見守りシステムアーキテクチャ

6. まとめと今後の課題

本研究では M2M/IoT システムの設計における課題として、処理の分散配置と構成要素の選択について取り上げ、その解決方法として、M2M/IoT システムアーキテクチャ推薦・評価システムを提案した。

また、5章での車内見守りシステムを対象とした利用シナリオからも、M2M/IoT システムのアーキテクチャを推薦できることが明らかになったが、以下の課題も明らかになった。

- ・構成だけを決定できても実際に構築する際に必要なハードウェアやソフトウェアが不明であるため、これらの推薦や評価を行うための仕組みが必要である点
- ・推薦されたアーキテクチャに関して、特徴や利点、欠点のみを表すのではなく、ユーザから入力された情報をもとに、途中段階での品質の評価を行い、それをユーザに知らせる仕組みが必要である点。
- ・本研究では提案のみを行ったために、実際にシステムとして作成した場合に、シナリオどおりに推薦・評価を行うことができない可能性がある点。

以上の三点及び、アーキテクチャを評価するアルゴリズムについての検討も必要のため、今後はこれらの課題の解決を目指して研究を進めていく。

参考文献

- [1] 総務省平成 29 年版情報通信白書第一部特集
<http://www.soumu.go.jp/johotsusintokei/whitepaper/ja/h29/html/nc133100.html>
- [2] 大江信宏 M2M/IoT システムのものづくりにおけるプロトタイプ構築法とその実践評価
- [3] Bonomi, Flavio, et al. "Fog computing and its role in the internet of things." Proceedings of the first edition of the MCC workshop on Mobile cloud computing. ACM, 2012.
- [4] Shi, Weisong, et al. "Edge computing: Vision and challenges." IEEE Internet of Things Journal 3.5 (2016): 637-646
- [5] <https://nodered.org/>
- [6] Sielis, George A., Aimilia Tzanavari, and George A. Papadopoulos. "ArchReco: a software tool to assist software design based on context aware recommendations of design patterns." Journal of Software Engineering Research and Development 5.1 (2017): 2.
- [7] 森田剛史 M2M 技術を応用した”車内見守りシステム”：構築と検証
- [8] 環境省熱中症予防サイト : 暑さ指数(WBGT)とは?
<http://www.wbgt.env.go.jp/wbgt.php>