

データ連携基盤を活用したロケーションウェアサービスプラットフォームの提案

中橋 友郎[†] 陳 思楠[†] 中村 匡秀^{†,††} 佐伯 幸郎^{†††}

[†] 神戸大学 〒 657-8501 神戸市灘区六甲台町 1-1

^{††} 理化学研究所・革新知能統合研究センター 〒 103-0027 東京都中央区日本橋 1-4-1

^{†††} 高知工科大学 〒 782-8502 高知県香美市土佐山田町宮ノ口 185

E-mail: [†]tomorrow@ws.cs.kobe-u.ac.jp, ^{††}chensinan@gold.kobe-u.ac.jp, ^{†††}masa-n@cmds.kobe-u.ac.jp,
^{††††}saiki.sachio@kochi-tech.ac.jp

あらまし センシング技術の発展により、昨今では車両や人など移動体の位置情報を活用し、位置に応じたサービスを提供するロケーションウェア (LA) サービスの開発が著しい。しかしながら、現状では LA サービスの開発は、移動体の位置情報の管理や、位置情報に基づく条件の評価、条件を満たした際の処理を含んだシステムを個別で構築する必要がある。システムがサイロ化されることで、収集したデータの再利用性が低下し、また、機能の再開発のコストがかかる。本研究では、LA サービス開発を効率化する LA プラットフォームを提案する。LA プラットフォームは、従来では LA サービス毎に開発する必要があったオブジェクトの情報管理や条件の登録・評価機能などを LA サービスから切り出したサービスとして提供することで、LA サービスの実装範囲を低減し、開発コスト削減を実現する。ケーススタディでは、LA プラットフォームを活用し、人と部屋の位置関係に応じて入退室を通知する LA サービスの開発を行った。

キーワード スマートシステム, ECR ルール, ロケーションウェア, データ連携基盤, Pub/Sub

Developing Location-Aware Service Platform Using Data Integration Platform

Tomoro NAKAHASHI[†], Sinan CHEN[†], Masahide NAKAMURA^{†,††}, and Sachio SAIKI^{†††}

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

^{††} Riken AIP, 1-4-1 Nihon-bashi, Chuo-ku, Tokyo, 103-0027 Japan

^{†††} Kochi University of Technology, 185 Tosayamadacho Miyanakuchi, Kami, Kochi, 782-8502 Japan

E-mail: [†]tomorrow@ws.cs.kobe-u.ac.jp, ^{††}chensinan@gold.kobe-u.ac.jp, ^{†††}masa-n@cmds.kobe-u.ac.jp,
^{††††}saiki.sachio@kochi-tech.ac.jp

Abstract With the development of sensing technology, location-aware (LA) services, which utilize location information of moving objects such as vehicles and people to provide location-based services, have been remarkably developed in recent years. However, the development of Currently, the development of LA services requires the construction of individual systems that include the management of location information of mobile objects, evaluation of conditions based on location information, and processing when conditions are met. Siloed systems reduce the reusability of collected data and increase the cost of redevelopment of functions. In this study, we propose an LA platform that streamlines LA service development. The LA platform reduces the scope of LA service implementation and development costs by providing object information management, condition registration and evaluation functions, etc., as separate services from LA services, which conventionally had to be developed for each LA service. In the case study, the LA platform was used to develop an LA service that notifies users when they enter or leave a room based on the positional relationship between people and rooms.

Key words Smart System, ECR Rule, Location Awareness, Data Integration Platform, Pub/Sub

1. はじめに

ICT(情報通信技術)の発展に伴い、物理世界のモノ・コトの情報とクラウド上の計算資源を連携し、新たな付加価値サービスが提供されるシステム(スマートシステム)が実現する社会である超スマート社会[1]が、近年より我が国が目指すべき社会として提唱されている。超スマート社会では、あらゆる人が必要な時・必要な場所で適切なサービスを享受できることが重要視されている。これを実現するために、モノやコトの状況(コンテキスト)をセンサなどで認識(アウェア)し、状況に応じたサービスを提供するコンテキストアウェアサービス[2]の開発が盛んである。とりわけ、物理的に規模の大きなスマートシステムの実装例となるスマートシティ[3]において、現在位置や混雑状況など、地理的情報に基づく状況変化に応じたサービス提供の需要は高く、このようなサービスはロケーションアウェア(LA)サービス[4]と呼ばれる。

LAサービスのアーキテクチャは、ECA(Event・Condition・Action)ルール[5]に基づいたルールベースシステムとして構成される。ECAルールは、自動的に実行される処理を定義するためのルールであり、ルール実行のトリガー(Event)が生じた際、システムが評価する条件(Condition)が満たされれば、定義された処理(Action)が実行される。LAサービスにおいては、人や車両の位置情報などの変化がEventにあたり、位置情報やモノ同士の位置関係などに基づく条件がConditionにあたる。Actionは、ユーザへの通知や物理デバイスの動作、外部サービスの連携動作などである。

LAサービス開発の現状として、Eventの扱い(オブジェクトの状態変化の管理)・Conditionの評価・Actionの実行を含んだシステム構築する必要があり、サービス毎にサイロ化されたシステムが構築されている。データモデルやデータベースのアクセス先がサービス毎に異なることで収集したデータの再利用性が低下し、また、多くのLAサービスで共通しているオブジェクト管理やCondition評価の機能を再開発するコストがかかるという問題がある。

そこで我々は、オブジェクトの情報管理や、種々のLAサービス実現に求められるCondition評価の機能を外部サービスとして提供し、LAサービスから切り出すことで、LAサービス開発を低コスト化出来ないかと考えた。本研究では、LAサービスの開発を支援するLAプラットフォームを提案する。LAプラットフォームは、オブジェクトの情報管理機能と、Conditionの登録・評価機能をサービスとして提供する。開発者は、開発対象のLAサービスが取り扱う移動体や建物などのオブジェクトや、実行したいActionのトリガーとなるConditionをプラットフォームに登録する。LAプラットフォームは、オブジェクトの情報が増減した際、登録されたConditionを評価し、その評価結果を外部pub/subシステムにpublish(通知)する。LAサービスは、pub/subシステムにsubscribe(購読)することで、Conditionの評価結果を受け取ることができる。これにより、開発者が実装するLAサービスの機能は、実行するActionのみとなり、従来のようにEventの扱いやConditionの評価を含んだシステム

を構築する必要がなくなる。また、オブジェクトの情報管理を、AWS[6]・GCP[7]・FIWARE[8]などに代表されるデータ連携基盤を活用して行うことで、新たなLAサービスや、他分野データとの連携による付加価値サービスの実現など、データの再利用性の向上が期待できる。

本稿では、提案するLAプラットフォームを、オープンソースのデータ連携基盤FIWAREと、我々の研究グループが開発したpub/subシステムcs27pubsubをラップしたプロトタイプとして実装した。ケーススタディとして、実装したLAプラットフォームを活用したオブジェクト作成・Condition登録を行い、学生と部屋の距離に応じて入退室を通知するLAサービスを開発し、Event管理やCondition評価の実装を必要とせず、Actionのみのプログラム実装によるLAサービスの実現を確認した。

2. 準備

2.1 スマートシティ

ICTの発展により、交通・製造業・農業・医療・介護など多分野でモノの状態がセンシングされ、都市の最適な運用に役立てられている。多種多様な大規模データの収集が可能になり、これらのデータを統合管理し、利活用するための基盤の開発が進んでいる。大規模データを活用する基盤を都市に適用したものは都市OSと呼ばれ、行政・個人・企業データなど様々な物理世界の情報を収集・連携することによる新たな付加価値サービス(スマートサービス)の提供の実現を目指すスマートシティにおける中核を担う存在として、官民協同で導入に取り組んでいる。

2.2 ロケーションアウェアサービス

センシング技術の発展により、動的に変化するモノやヒトの状況(コンテキスト)をセンサで認識(アウェア)し、状況に応じた適切なサービスを提供するコンテキストアウェア・サービスの研究開発が活発である。特に、スマートシティにおいて、あらゆる人が必要な時・必要な場所でサービスを享受するために、ヒトや車両など地理的情報をもつ移動体の位置情報を活用したコンテキストアウェアサービスは、ロケーションアウェア(LA)サービスと呼ばれる。ロケーションアウェアサービスの例として、バスとバス停の距離の変化を認識して発着を知らせるバスロケーションシステム[9]や、観光地に近づくとその地に縁のあるクイズに挑戦できるまち歩きアプリ[10]などが提供されている。

2.3 LAサービス実現に必要な実装技術

状況に応じたサービス提供の実現における実装アプローチとして、ECAルールに基づいたルールベースシステムが挙げられる。ECAルールとは、自動的に実行される処理を定義するためのルール定義である。ルール実行のトリガー(Event)が生じた際、システムが評価する条件(Condition)が満たされれば、定義された処理(Action)が実行される。LAサービスにおいて、Eventは、移動体の位置情報などの状態変化を表す。Conditionは、モノ同士の位置関係などに基づく条件を表す。Actionは、ユーザへの通知や物理デバイスの動作、外部サービスの連携動作などを表す。バスロケーションシステムを例に挙げると、「バ

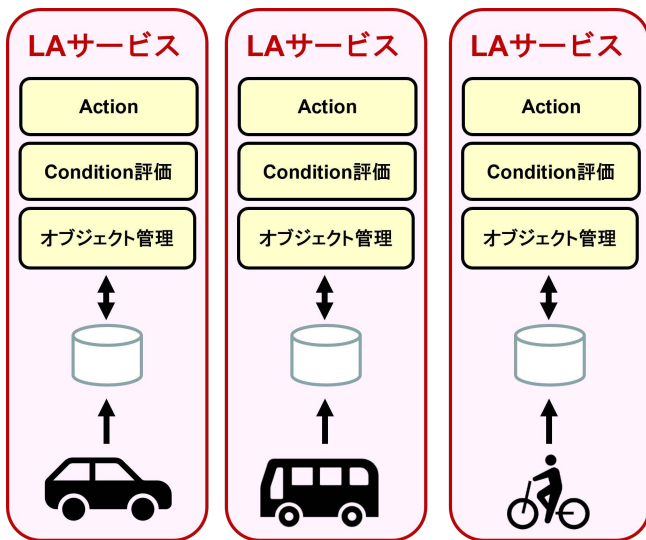


図1 LA サービス開発の現状

スが移動した」Event が、「バスとバス停の距離が一定以下になる」Condition で評価され、条件が満たされた場合「バス停に到着した旨を画面に表示する」Action が実行される。これらを踏まえて、LA サービスの実現に必要なとされる機能・システム構成は大きく以下の3点に分けられる。

S1: オブジェクトの情報管理

S2: Condition 評価

S3: Condition を満たした際の処理

S1 について、移動体や建物などのオブジェクトの状態をシステムが参照するために、必要な情報をデータとしてデータベースなどに保持する必要がある。また、ECA の Event にあたる、動的に変化するオブジェクトの状態をセンシングし、データをシステムに投入する仕組みが必要である。S2 は、オブジェクト同士の位置関係や移動距離に関する条件など、実現したいサービスにより評価すべき条件は様々である。3 は、Action に当たる部分であり、処理の例としてユーザへの通知や物理デバイスの動作、外部サービスの連携動作などが挙げられる。

2.4 LA サービスの開発上の問題点

図1に示す通り、現状では、LA サービスの開発は上述した S1,S2,S3 を1つのサイロ化されたシステムとして実装されている。サービス毎に個別システムを開発することによる問題点として以下の二点が挙げられる。

P1: データの再利用性が低い

P2: 共通機能を再開発するコストがかかる

P1 について、データモデルやデータベースのアクセス先がサービス毎に異なることで、他サービスからのデータ利活用が阻害されている。各 LA サービスが収集するデータが一元管理され、再利用性の向上が実現すれば、異種分野のデータを連携した新たな LA サービスの実現が期待できるようになる。FIWARE を筆頭としたデータ連携基盤は、抽象度の高いデータモデルを扱い、データ管理に関わる様々な機能群を備えているため、分野横断的なデータの連携に適していることから、データ連携基盤を活用してオブジェクトの状態管理をで行うことで

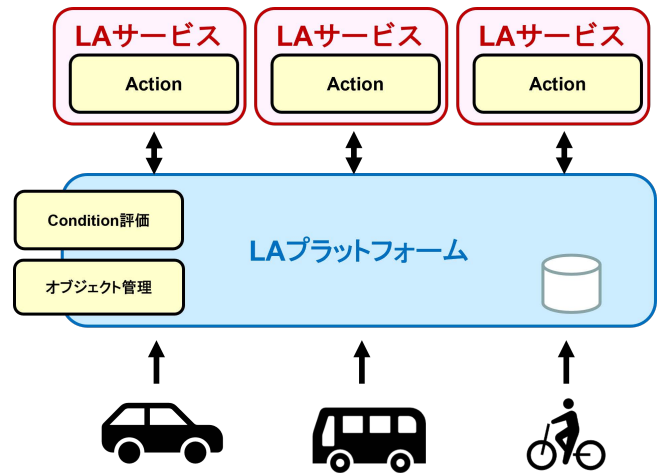


図2 LA プラットフォーム全体アーキテクチャ

P1 の解決が期待できる。

P2 について、2.3 で述べたように、LA サービスの実装におけるシステム構成は ECA ルールに基づいたシステムとして共通している点は多く、オブジェクトの管理や Condition 評価のロジックなど、各 LA サービスに共通で実装する必要のある機能をサービス毎に実装する必要があり、実装にコストがかかる。

3. 提案手法

3.1 研究目的とアプローチ

本研究の目的は、LA サービスの低コストな開発を実現することである。そのためのアプローチとして、データ連携基盤の活用によるオブジェクト管理と、位置情報に基づいた Condition の設定など、LA サービスに必要な機能をサービスとして提供する LA サービス開発支援基盤 LA プラットフォームを提案する。

3.2 全体アーキテクチャ

図2は、提案プラットフォームと LA サービスの関係を表す全体アーキテクチャである。イベント評価やオブジェクト管理の仕組みを LA がサービスとして提供することで、統合的なデータ管理と、再開発のコスト削減を実現する。図3は、提案プラットフォームの詳細アーキテクチャである。LA プラットフォームは以下の要素で構成される。

A1: オブジェクト定義

A2: Condition 設定

A3: Condition 評価・通知

PF ユーザである LA サービス開発者は、緯度・経度などの位置情報を持つオブジェクトをプラットフォームに登録する (A1)。次に、プラットフォームが提供する Condition のひな型に A1 で登録したオブジェクトを組み込み、Condition を設定する (A2)。Condition は、オブジェクトの情報が更新された際に評価され、条件を満たした場合、pub/sub システムに通知 (publish) される (A3)。これにより、LA サービスは、pub/sub システムに通知された情報を受け取り (subscribe)、Condition の評価結果に応じた処理を実行するといった、シンプルな実装で実現できる。以下では、A1～A3 の詳細を説明する。

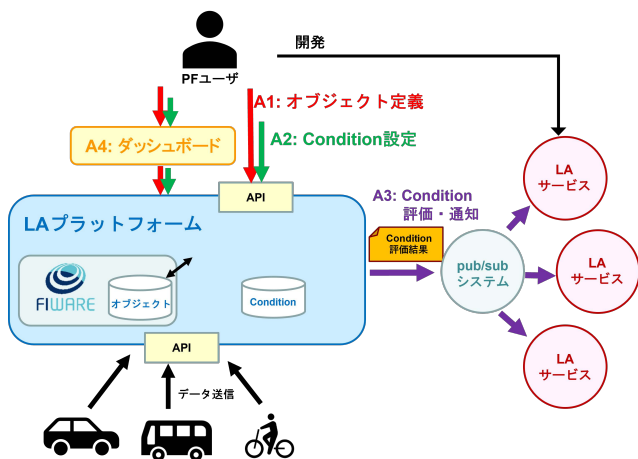


図3 LA プラットフォーム詳細アーキテクチャ

3.3 A1: オブジェクト定義

オブジェクト定義は2つのステップで行う。

一つ目は、オブジェクトの型を定義することである。型名・属性とそのデータ型を指定し、具体的な属性値を持つインスタンスを生成するためのひな型を作成する。型定義は JSON 形式で記述し、以下の項目を指定する。

- **serviceGroup**
 - 名前空間を分けるためのグループ名
- **objectType**
 - 型名
- **isMove**
 - 動的に位置が変化するかどうか
- **attributes**
 - オブジェクト固有の属性名・データ型のリスト

二つ目は、オブジェクトのインスタンスを生成することである。オブジェクト型名と、serviceGroup 内で一意なオブジェクト名を指定し、作成リクエストを LA プラットフォームに送信することで、オブジェクト型に設定された属性を持つオブジェクトがデータ連携基盤に登録・管理される。

3.4 A2: Condition 設定

PF ユーザは、実現したい LA サービスに応じて、LA プラットフォームが提供する Condition のひな型を選択し、A1 で設定したオブジェクトや値を組み込み、Condition を設定する。LA プラットフォームが提供する Condition のひな型の例を以下に示す。

- [オブジェクト 1] と [オブジェクト 2] の距離が [値] メートル以下になる
- [オブジェクト 1] の [属性名] の属性値が [値] になる

Condition 設定の記述は JSON 形式で行い、以下の項目を指定する。

- **serviceGroup**
 - 名前空間を分けるためのグループ名
- **conditionNumber**
 - Condition のひな型の種類
- **conditionObject**
 - ひな型に組み込むオブジェクトの Id
- **conditionValue**
 - ひな型に組み込む値

3.5 A3: Condition 評価・通知

データ連携基盤が管理するオブジェクトの情報が更新されると、LA プラットフォームは更新されたオブジェクトが関わる Condition を評価する。評価の結果、条件を満たした Condition があれば、評価結果を pub/sub システムを介して LA サービスに通知する。pub/sub システムとは、メッセージの出版者 (publisher) が Topic に対してメッセージを送信すると、Topic の購読者 (subscriber) にメッセージが配信される仕組みである。LA サービスへの通知を pub/sub システムを仲介して行うことで、LA プラットフォームが Condition の通知先となる LA サービスのエンドポイントを把握しなくとも、subscriber に Condition の評価結果を通知することができる。

4. 実装

4.1 LA プラットフォーム

本稿では、提案する LA プラットフォームのプロトタイプ実装として、オブジェクトの定義・管理、Condition 設定・評価、通知を RESTful API で提供するシステムを開発した。Condition のひな型として、2つのオブジェクトの距離が一定以下であれば真と評価する評価アルゴリズムを実装した。実装は Java 言語と SpringBoot を用いた。他サービスとの連携については、オブジェクトの登録・更新・削除を FIWARE の Orion Context Broker を介して行い、Condition の通知は我々の研究グループが開発した pub/sub システム cs27pubsub [11] を利用している。

4.2 ユーザインタフェース

LA プラットフォームの API を利用し、オブジェクトの管理を行うためのユーザインタフェースを開発した。実装は React フレームワークを用いた。スマートフォンなど位置情報の取得が可能な端末の web ブラウザでオブジェクト一覧ページ (図 4) からオブジェクトの位置情報更新ページ (図 5) を開いておくと、数秒毎に現在地を取得し、LA プラットフォームにオブジェクトの位置情報を送信することができる。

5. ケーススタディ

5.1 入退室通知サービス

LA プラットフォームを利用した LA サービス開発の例として、部屋との距離に応じて学生の入退室を機械音声で通知するデモアプリを開発した。位置情報の送信は、4.2 で実装した位置情報送信機能を利用した。機械音声での通知は、我々の研究グループが開発した音声読み上げサービスの web API を利用し

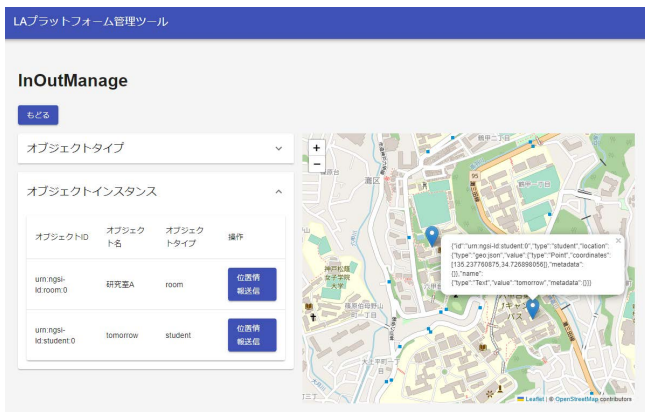


図4 オブジェクト一覧画面

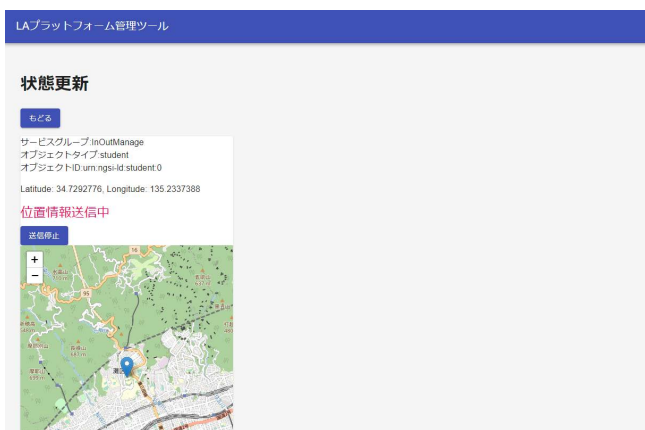


図5 位置情報送信画面

た。本サービスの開発は以下のステップで行った。

- step1: サービスグループの作成
- step2: オブジェクト型の作成
- step3: オブジェクトの作成
- step4: Condition の設定
- step5: サブスクリプションの設定
- step6: LA サービスの実装

5.1.1 step1: サービスグループを作成

サービスグループ "InOutManage" を LA プラットフォームに作成した。作成リクエストが送信されると、cs27pubsub に "InOutMangage" グループに登録された Condition の評価結果の通知先となる Topic が作成される。

"InOutMangage" グループ作成リクエストデータ

```
{ "name": "InOutManage" }
```

5.1.2 step2: オブジェクト型の作成

オブジェクト型 "student" と "room" を "InOutManage" グループに作成した。

"student" 型作成リクエストデータ

```
{ "serviceGroup": "InOutManage", "objectType": "student", "isMove": true, "attributeNames": [] }
```

"room" 型作成リクエストデータ

```
{ "serviceGroup": "InOutManage", "objectType": "room", "isMove": false, "attributeNames": [] }
```

5.1.3 step3: オブジェクトの作成

オブジェクト型 "student" のオブジェクト "tomo" と "room" のオブジェクト "laboA" を作成した。

"tomo" オブジェクト作成リクエストデータ

```
{ "serviceGroup": "InOutManage", "objectType": "student", "objectName": "tomorrow", "location": [0,0], "attributeValues": [] }
```

この操作で FIWARE に以下のようなオブジェクトのデータが作成される。

FIWARE に保存されるデータ

```
{ "id": "urn:ngsi-ld:student:0", "type": "student", "location": { "type": "geo:json", "value": { "type": "Point", "coordinates": [ 135.2337388, 34.7292776 ] }, "metadata": { } }, "name": { "type": "Text", "value": "tomo", "metadata": { } } }
```

5.1.4 step4: Condition の設定

"tomo" オブジェクトと "laboA" オブジェクトの距離が 10m 以下になった際に真と評価される Condition を作成した。

Condition 作成リクエストデータ

```
{ "serviceGroup": "InOutManage", "methodNumber": 1, "conditionObject": [ { "objectType": "student", "name": "tomo", "order": "1" }, { "objectType": "room", "name": "laboA", "order": "2" } ], "conditionValue": [ { "valueType": "double", "value": "10", "order": "1" } ] }
```

5.1.5 step5: サブスクリプションの設定

5.1.4 で作成した Condition の評価結果を開発する LA サービスが受け取るためのサブスクリプションを設定した。

サブスクリプション設定リクエストデータ

```
{ "serviceGroup": "InOutManage", "endpoint": [[LA サービスのエンドポイント]] }
```

"endpoint" に指定したエンドポイントに、Condition の評価結果が通知される。

5.1.6 step6: LA サービスの実装

5.1.5 で設定したエンドポイントに対する HTTP POST リクエストを受け取るサーバプログラムを実装した。以下は、実際に 5.1.4 で設定した Condition が評価された際に、サーバプログラムが受け取るリクエストデータである。

サーバプログラムが受け取るリクエストデータ

```
{ "evaluationResult": "true", "objects": [ { "id": "urn:ngsi-ld:student:0", "type": "student", "name": "tomo", "location": [135.2377608, 34.7268980] }, { "id": "urn:ngsi-ld:room:0", "type": "room", "location": [135.233772, 34.7268969], "name": "laboA" }, "values": [ [ { "value": 10, } ] ] }
```

上記データを受け、"evaluationResult" の値が "true" であれば、音声読み上げサービスの API を利用して、学生 "tomo" の入室を機械音声で通知するプログラムを実装した。図 6 に、実装

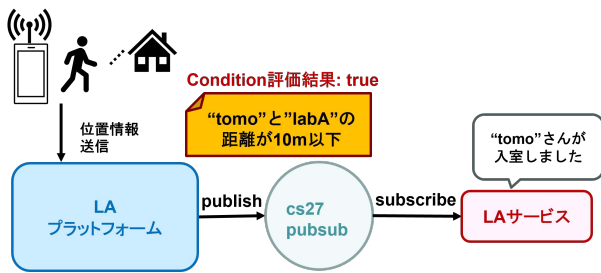


図6 入退室通知サービスアーキテクチャ

した入退室通知サービスのアーキテクチャを示す。

5.2 考察

以上より、学生と部屋の距離が近づいた際に、入室を通知する LA サービスが実装した。2.4 で述べた P1 への対応として、学生や部屋の情報を、入退室通知サービスが独自に保持せず、FIWARE が管理するオブジェクトを参照することで、データの再利用性の向上を確認した。P2 への対応として、LA プラットフォームにオブジェクトの状態管理や Condition 評価を任せることで、入退室通知サービスのプログラム実装は、Condition の評価結果を受け取り音声通知するというシンプルな実装で実現し、開発コストの削減が確認できた。

6. まとめ

本稿では、ECA ルールに基づいた LA サービスの開発を効率化することを目的とし、データ連携基盤の活用によるオブジェクト管理と、位置情報に基づいた Condition の設定などを実現した、LA プラットフォームを提案した。LA プラットフォームは、LA サービスの実現に必要な Condition のひな型を提供する。LA サービス開発者がひな型から Condition を設定し、評価を LA プラットフォームに任せることで、アクションのみを意識した実装が可能となる。本稿では、LA プラットフォームのプロトタイプ実装を行い、入退室通知サービスの開発を例に、LA プラットフォームを活用した低コストな LA サービス開発を示した。

今後の課題としては、LA プラットフォームの機能を拡張し、移動距離に関する条件など、より多くの Condition のひな型を提供することで、多様な LA サービスの開発を支援することが挙げられる。そのために、Condition 評価のアルゴリズムを外部から追加できるアーキテクチャの検討が必要であると考えている。

謝辞 本研究の一部は JSPS 科研費 JP19H01138, JP20H05706, JP20H04014, JP20K11059, JP22H03699, JP19K02973, 若手研究 23K17006 の助成を受けて行われている。

文献

- [1] 原辰次, 本多敏, “超スマート社会におけるシステム科学技術概論,” 計測と制御, vol.55, no.4, pp.284–287, 2016.
- [2] 坂本寛幸, 井垣宏, 中村匡秀, “コンテキストウェアアプリケーションの開発を容易化するセンササービス基盤,” 電子情報通信学会技術研究報告, vol.108, no.458, pp.381–386, 2009.
- [3] 藤井篤之, 中西華子, “スマートシティによる地方創生” 会津若松市,” 情報の科学と技術, vol.67, no.11, pp.566–572, 2017.
- [4] 松崎和賢, 吉岡信和, 本位田真一他, “Location scope: ロケーション

ウェアソフトウェア開発支援手法の提案,” 情報処理学会論文誌, vol.46, no.12, pp.2925–2939, 2005.

- [5] 中田匠哉, 中村匡秀, “コンテキストに基づくサービス連携を支援するイベントルーチングサービスの検討,” 電子情報通信学会技術研究報告; 信学技報, vol.121, no.157, pp.32–37, 2021.
- [6] “Aws,” <https://aws.amazon.com/jp/>. (Accessed on 06/18/2023).
- [7] “Gcp,” <https://console.cloud.google.com/>. (Accessed on 06/18/2023).
- [8] 石井和彦, 山中淳史, “Fiware を活用したスマートシティ向け共通プラットフォームの構築 (高松市事例)(データを活用した持続可能な都市経営特集)-(データ利活用型スマートシティの実証・実装事例),” NEC 技報=NEC technical journal, vol.71, no.1, pp.29–32, 2018.
- [9] 伊藤昌毅, 川村尚生, 菅原一孔, “スマートフォンを利用したバスロケーションシステムの開発,” 電子情報通信学会論文誌 D, vol.96, no.10, pp.2327–2339, 2013.
- [10] 明石拓弥, 中村匡秀, 大浦秀喜, 中井哲也, “クイズラリープラットフォームを用いた市民の地域に対する意識調査と行動分析,” 電子情報通信学会技術研究報告; 信学技報, vol.121, no.415, pp.41–47, 2022.
- [11] M. Nakamura, “cs27pubsub api マニュアル,” <https://wsapp.cs.kobe-u.ac.jp/cs27pubsub/swagger-ui.html>. (Accessed on 06/18/2023).