資料４－２

おもてなしインフラ

報告書（案）

平成 28 年 2 月 10 日

目次

1　背景と目的

2020 年には 3000 万人ともいわれる、数多くの海外からの観光客の来日が予想されている。言葉による意思疎通の難しいケースでも、各自の属性とニーズに応じた適切なサービスを、情報通信技術（ICT）を利用して実現する「おもてなし」がいま望まれている。さらに 2020 年に向けて 4K8K テレビやデジタルサイネージ等、ICT を利用した多くの先進的サービスが計画されているが、それらを点としてではなく都市のサービスとして連携させるためにも、サービス間でオープンにユーザーの属性情報を連携させて使える仕組みをインフラとして構築することが必要である。このインフラによって、外国人観光客などのユーザーが、属性（言語、宗教・文化、身体特性）と、空間情報（現在地）や目的（行き先など）などに応じた、その時その場で最適な情報とサービスを、言葉の壁なく簡単に得られるようにする。また、サービサーがサービス対象を認識し、その属性に応じた最適のサービスを容易に提供できるようにする。このような仕組みを提供するインフラを「おもてなしインフラ」と名付ける。

おもてなしインフラでは、交通系 IC カードをカードリーダーにタッチするだけで様々なサービスを透過的に利用できることを目指す。サービスを享受するための認証方式には、交通系 IC カード以外にも非接触 IC カード、スマートフォン内蔵の NFC や生体認証など、様々な認証方式に対応することを目指す。おもてなしインフラと様々なサービスと、IC カードやスマートフォンとの関連を図 1 に示す。

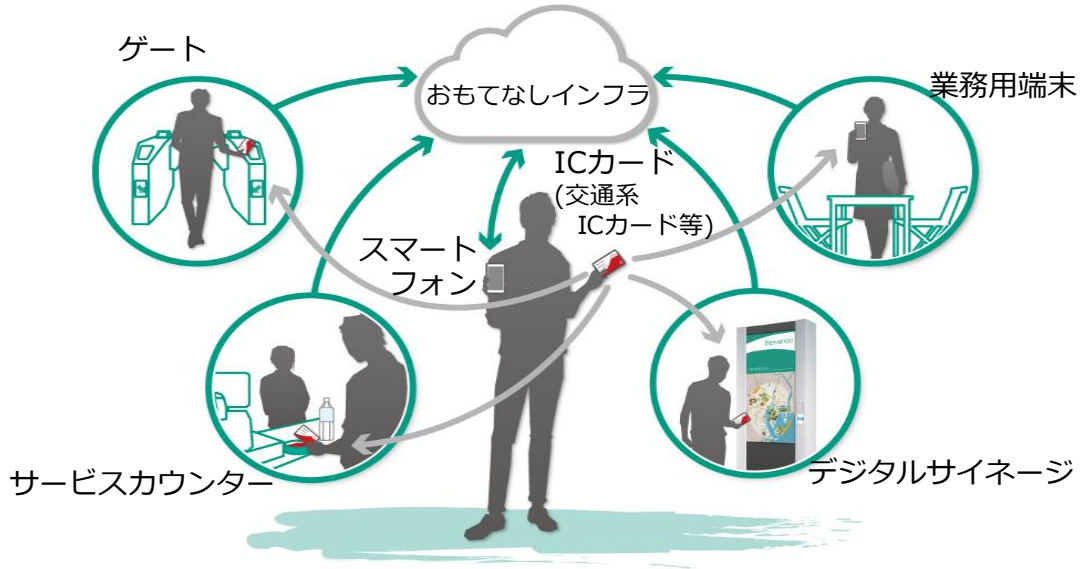

図 1 さまざまなサービスの利用を 1 枚のカードやスマートフォンに集約

2020 年にはおもてなしインフラが広く普及している状態となることを目指すために、2016 年度には一部の地域にて実証事業が予定されている。本報告書では、この地域実証に向けておもてなしインフラに求められる機能を明らかにする。

2　実現する機能

2.1　　インフラの提供する機能の概要

おもてなしインフラでは、おもてなしインフラで管理するユーザーの属性情報と、様々なサービス上で提供されるユーザーの属性情報をユーザー及びサービサーの合意を得た上で、一括で検索・取得できることを目指す。おもてなしインフラにて管理するユーザーの属性情報をユーザー基本属性情報と定義し、サービス事業者がそのサービス上から取得するユーザーの属性情報をユーザー属性情報と定義する。

なお、おもてなしインフラからユーザー属性情報を検索・取得する問い合わせがあった場合でも、その提供については、サービサー側にてその可否を判断するものである。

実証実験では、下記のユーザー基本属性情報の例のうち、実サービス化に向け、どの項目が必要となるかを明確化する。また、ユーザーが全ての項目をおもてなしインフラに登録することを必須とするか、ユーザーが登録の可否を選択できることとするかについても検証を行うこととする。

(ユーザー基本属性情報例)

- 性別
- 年齢・年代
- 出身（言語）
- 目的地
- 希望するユーザーインタフェース（画面表示、音声案内等）
- アクセシビリティに関する情報（車いす等）
- 食の禁忌
- メールアドレス
- パスポート関連情報
- オリジナル ID(ユーザーに割り当てられるグローバルでユニークな識別子)　　等

2.1.1　　インフラの利用者・管理者

おもてなしインフラの利用者・管理者として下記 5 者を想定する。

- ユーザー
  - おもてなしインフラを活用したサービスを利用するユーザー
- サービサー
  - サービスを提供する事業者又はその委託を受けてシステムを開発する事業者（もしくは、サービスを提供する事業者からシステム運用の受託者）
- おもてなしインフラ管理者
  - おもてなしインフラ自体の管理者
- おもてなしインフラサービサー管理者

- おもてなしインフラ管理者のうち、サービサーとサービスの管理のみ可能な管理者
- デバイス管理者
  - おもてなしインフラに登録したデバイスの情報を編集できるユーザー

2.1.2 インフラに対する要件

ユーザー属性情報を様々なサービサー間で連携するために、おもてなしインフラには以下のような要求が存在する。なお実証実験時にサービサーからの要望等に応じて、以下の要求を変更する可能性がある。

検索・取得について

- 様々なサービサーが提供するサービス上でユーザーを一意に識別できること
- サービスが提供するユーザー属性情報を検索・取得できること
  - おもてなしインフラ自体が提供するユーザー基本属性情報に関しては、個人を特定しないものでユーザーの同意及びサービサーの合意を得たもののみで、共通事項となるものについて、おもてなしインフラ上で検索・取得できること
- 表記や単位の異なる情報もサービスの要求に応じて変換して取得できること
  - おもてなしインフラと、あるサービスが提供する情報の単位を、サービスの要求する単位系に変換して提供すること
  - おもてなしインフラと、あるサービスが提供する情報の表記を、サービスの要求する表記方式に変更すること
    - ✧ 例えば電話番号の表記方式の変更や、日時の表記方式の変更を想定する

セキュリティ・プライバシーについて

- ユーザーに無許可でサービスが情報を取得できないこと
  - 公序良俗に反するサービスは、おもてなしクラウドへの接続の対象としないが、仮にこうしたサービスが接続してきた場合でも、ユーザーの ID を悪用して他サービス等から無許可で情報を取得できないこと
- サービスに対して提供するユーザー属性情報をユーザーが細かく制御できること
  - ユーザーがサービスに対して提供するユーザー属性情報は、ユーザーが任意のタイミングで利用許可を取り消すことができること
  - ユーザー属性情報の利用をユーザーが許可する際には、サービスが必要とするユーザー属性情報を一括で許可、個別に選択して許可、の両方ができること
- ユーザーの属性情報に対するアクセス権限を継承できること
  - ユーザー属性情報に対する権限の継承をユーザーがサービサーに対して許可すると、そのサービサーから任意のサービサーへアクセス権限を継承できること

- 権限の多段継承が許可されている場合、権限を継承したサービサーが別のサービサーにアクセス権限を継承できること
  - 例えば、A 社が B 社に事業の委託と共に、どの権限を委譲するかを明示してアクセス権限の継承を行い、その権限をさらに B 社が C 社へ事業の委託と共に A 社に指定された権限の中から特定の権限を継承すると、C 社が作成したサービスには B 社が指定した A 社の権限を利用できることを想定している。なお継承時にはユーザーの承認を必要とする。
- あるユーザーまたはサービサーが任意のサービサーから権限を剥奪した場合、そのサービサーから継承された該当ユーザーに関する権限が剥奪されること

利便性について

- あらかじめユーザーの許可があれば、サービス利用の度にユーザーに許可を求めることなくユーザー属性情報の提供を行うこと
  - 例えばユーザーの言語情報など、確認なく利用可能であるとあらかじめユーザーが指定したユーザー属性情報は、新たなサービスを利用する場合でも、ユーザーの確認なくユーザー属性情報を提供すること
- 様々な認証方式に対応すること
  - 認証方式には、交通系 IC カードの他、ユーザー名とパスワードの組み合わせや NFC スマートフォン、指紋、虹彩などが考えられる
- 多様な認証のレベルに対応すること
  - 提供するユーザー属性情報や、ユーザー情報を利用する場面による、ユーザー属性情報取得時の認証の強度の違いに対応できること
- おもてなしインフラ対応デバイスの情報を取得できること
  - デバイスとは、デジタルサイネージなど、街や施設内などに設置されるおもてなしインフラの API に準拠したハードウェアを指す
  - おもてなしインフラ対応のデバイスが広く普及した場合、現在地周辺のデバイスの検索をおもてなしインフラで実現する等、おもてなしインフラとデバイスが連携することにより、ユーザーが積極的にデバイスを活用できるため、利便性の向上に繋がる

2.1.3 インフラに求められる機能

以上の要求を実現するためおもてなしインフラでは以下の機能を提供する。なお機能の網羅性については、実証実験を通じて検証する必要がある。

- セマンティクスに基づくユーザー属性情報の取得を実現する
  - ここでいうセマンティクスとは、単位変換、表記変換を実現できることを表す

- ✧ 例えば、センチメートル単位で登録されているデータを、インチ単位で出力できること等を想定する
- ➢ ただし、単位変換、表記変換するためのルールは、任意に追加・削除可能とするべきである

- サービスが提供するユーザー属性情報を取得可能にする
  - ➢ あらかじめおもてなしインフラにサービスを登録すれば、登録したサービスが提供するユーザー属性情報も検索・取得可能とする
  - ➢ 地域実証では実現しないが、将来的には、おもてなしインフラ自体が管理するユーザー属性情報を動的に拡張して、サービスの過程で生成される情報なども取得できることを想定する
- ユーザーに一意な ID（オリジナル ID）を割り当て、全てのユーザー属性情報はこのオリジナル ID に関連づける
  - ➢ ID に関する詳細は 4.1.2 を参照されたい
- サービスにオリジナル ID を提供する際には、ユーザーがオリジナル ID をサービスに提供することに明示的に同意する必要があることとする
  - ➢ オリジナル ID の提供に同意がない場合は、おもてなしインフラで新たな ID（リンク ID）を発行してオリジナル ID の代わりにサービスへ提供する
  - ➢ リンク ID はサービス毎に異なる ID とする
  - ➢ オリジナル ID 1 個に対してリンク ID は複数対応づけ可能とする
- サービサーに信頼度を付与する
  - ➢ 信頼度の付与はおもてなしインフラ管理者が行う
- サービサーとサービスにユーザー属性情報のアクセス権限を付与する
  - ➢ アクセス権限の付与はユーザーが行う
- 対応する認証方式を拡張可能とする
  - ➢ 地域実証事業では少なくとも、交通系 IC カードとして採用実績のある FeliCa と、ユーザーID・パスワードの組み合わせに対応する
  - ➢ 最終的には、生体認証や二段階認証など、多様な認証方式に対応する
- 認証方式に信頼度を付与する
  - ➢ 信頼度の適切なレベルは、本実証実験を通じて明らかにする必要がある

2.1.4 地域実証事業に対する支援機能

2.1.4.1 おもてなしポータルアプリ

おもてなしインフラに対してユーザーがアクセスするためのアプリが、おもてなしポータルアプリである。おもてなしポータルアプリでは、以下の機能を提供する。なお機能の網羅性については、実証実験を通じて検証していく必要がある。

- サービスランチャー
  - おもてなしインフラに登録されているサービサーが提供するサービスの一覧を表示する
    - 日時や現在地などに対応して、関連の深い順でサービス一覧を表示する
  - サービスのランチャーとして機能する
    - ただし、端末内の任意のアプリを起動できないなど、ランチャーを技術的に実現不可能な端末ではこの機能を実現する必要はない
- プライバシーダッシュボード
  - ユーザーが各サービスやサービサーに対して提供しているユーザー属性情報の一覧を表示・変更する（アクセス権限の一覧を表示・変更する）インタフェースを提供する

上述の機能を提供するおもてなしポータルアプリを提供するプラットフォームとして下記を想定するが、実証実験を通じて他にも有用なプラットフォームが存在する場合には、そのプラットフォームにも対応することが必要である。

- スマートフォン
  - Android 端末と iPhone を想定する
    - 実証実験を実施する時期を考慮して Android のバージョンは 4.4 以降を想定する
    - 実証実験を実施する時期を考慮して iPhone の iOS のバージョンは 9 以降を想定する
  - OS の提供時期や端末の普及状況を考慮して、実現可能であれば Windows 10 Mobile にも対応を行う
- ウェブブラウザ
  - Internet Explorer 最新版、Microsoft Edge 最新版、Firefox 最新版、Chrome 最新版、Safari 最新版を想定する

2.1.4.2 開発者サイト

開発者サイトは、おもてなしインフラを活用したサービスをサービサーが開発することを促す Web サービスである。以下の機能を提供する他、開発者の疑問に答えるサポートページ等、開発を助けるための機能を出来るだけ盛り込むことを想定する。なお機能の網羅性については、実証実験を通じて明らかにする。

- API ドキュメントの提供
  - おもてなしインフラにて提供する API について、API 仕様や使い方に関する説

明を記載したドキュメントを掲載する

- 標準モジュールの提供
  - 後述するデバイス側標準モジュールと、サービス側標準モジュールなど、標準モジュールを提供する
- サービサーの登録・更新・削除
  - サービサーの登録は、サービサーに関する情報をおもてなしインフラに登録し、おもてなしインフラ管理者またはおもてなしインフラサービサー管理者によるサービサーの審査を通過すると完了する
  - サービサーはサービサー登録時に登録したユーザーID とパスワードを利用して開発者サイトにログインを行う
- サービスの検索・登録・更新・削除
  - サービスの検索はログインしなくとも可能であるが、サービスの登録・更新・削除はログインしている場合のみ可能である
  - サービスの更新・削除は、サービスを登録したサービサーと、おもてなしインフラ管理者、おもてなしインフラサービサー管理者のみ可能である
  - サービスに対して発行されているアクセストークン等を管理するインタフェースも提供する

2.1.4.3 デバイス側標準モジュール

2.1.4.3.1 機能概要

デバイスとは、おもてなしインフラを利用してサービスを提供するデジタルサイネージのようなハードウェアを指す。デバイス側標準モジュールとは、デバイスで一般的に使う機能をまとめて、対応システムを簡単に構築できる API を提供するモジュールである。おもてなしインフラとの連携を容易にするため、デバイスに ID を付与することを想定する。本モジュールはおもてなし対応デバイスを構築するためのサンプルソースとして公開予定とする。

2.1.4.3.2 機能要件

デバイス側標準モジュールの機能要件を以下に定義する。なお、以下の機能が全てとは限らない。機能の網羅性については実証実験を通じて検証していく必要がある。

- デバイス認証機能
  - おもてなしインフラと接続するデバイスが正当であるかを認証する
  - 一例としてデバイス証明書などが考えられる
- ユーザー認証機能
  - デバイスで読み取った交通系 IC カードの情報を認証マネージャ、アクセス制

御マネージャと連携の上、 ユーザーが誰であるかを認証する

2.1.4.3.3 非機能要件

デバイス側標準モジュールの非機能要件を以下に定義する。なお、以下の非機能要件が全てとは限らない。非機能要件の網羅性については実証実験を通じて検証していく必要がある。

- おもてなしインフラとの通信が暗号化されていること
- サンプルソースの追加／削除（バージョン管理）が容易であること
- サンプルソースは、メンテナンス性を考慮した構造とすること
  - 具体的には、ディレクトリ構造が開発者にわかりやすくなっていることや、ドキュメントが整っていること、ソースコード中にコメントが十分含まれていること等を想定する

2.1.4.4 サービス側標準モジュール

2.1.4.4.1 機能概要

おもてなしインフラは広く一般に利用されることを目的としており、サービサーがおもてなしインフラの仕様に準拠したシステムを可能な限り容易に構築できる必要がある。サービス側標準モジュールは、この目的を達成するため、サービスが一般的に使う機能をまとめて、対応システムを簡単に構築できる API を提供するモジュールである。本モジュールはおもてなし対応サービスを構築するためのサンプルソースとして公開予定とする。

2.1.4.4.2 機能要件

サービス側標準モジュールの機能要件を以下に定義する。なお、以下の機能が全てとは限らない。機能の網羅性については実証実験を通じて検証していく必要がある。

- ユーザー属性情報アクセス機能
  - ユーザー属性情報を登録、更新、削除する
  - ユーザー属性情報を検索・取得する

- 認証機能
  - サービスの認証、ユーザーの認証（交通系 IC カード、ユーザ名・パスワード）を認証マネージャ、アクセス制御マネージャと連携の上で実施する

2.1.4.4.3 非機能要件

サービス側標準モジュールの非機能要件を以下に定義する。なお、以下の非機能要件が

全てとは限らない。非機能要件の網羅性については実証実験を通じて検証していく必要がある。

- おもてなしインフラとの通信が暗号化されていること
- サンプルソースの追加／削除（バージョン管理）が容易であること
- サンプルソースは、メンテナンス性を考慮した構造とすること
  - 具体的には、ディレクトリ構造が開発者にわかりやすくなっていることや、ドキュメントが整っていること、ソースコード中にコメントが十分含まれていること等を想定する

3　アーキテクチャ

3.1　　全体構成

おもてなしインフラのアーキテクチャを図 2 に示す。おもてなしインフラは、以下から構成される。

- おもてなしユーザー属性情報提供
  - おもてなしインフラ内や外部サービスのユーザー属性情報を検索・取得する
- おもなてしアイデンティティ管理
  - ユーザーの認証やアクセス制御やアカウント管理などを実現する
- おもてなしインフラ管理
  - おもてなしインフラの監視などを実現する
- おもてなしポータルアプリ
  - おもてなしインフラを利用して構築されたサービスのポータルを実現する
- デバイスレジストリ
  - サービスとデバイスの対応付けやデバイスの設置場所等、デバイスに関する情報を登録する

また、おもてなしインフラ外ではあるが、サービサー向けの開発者サイトも開発対象である。おもてなしアプリは、各地域実証によって実現する内容が異なる可能性が考えられることから、おもてなしインフラでは開発対象外とする。

おもてなしインフラが ID をベースとしたインフラであるため、図中に ID 管理プラットフォームを記載している。

おもてなしインフラ上で収集されたユーザーの属性情報へのアクセス履歴など、おもてなしインフラに蓄積されたログの一部は、個人が識別できないように匿名化した上でオープンデータとして、おもてなしインフラから何らかのオープンデータ提供プラットフォームへデータ提供する可能性を検討する。

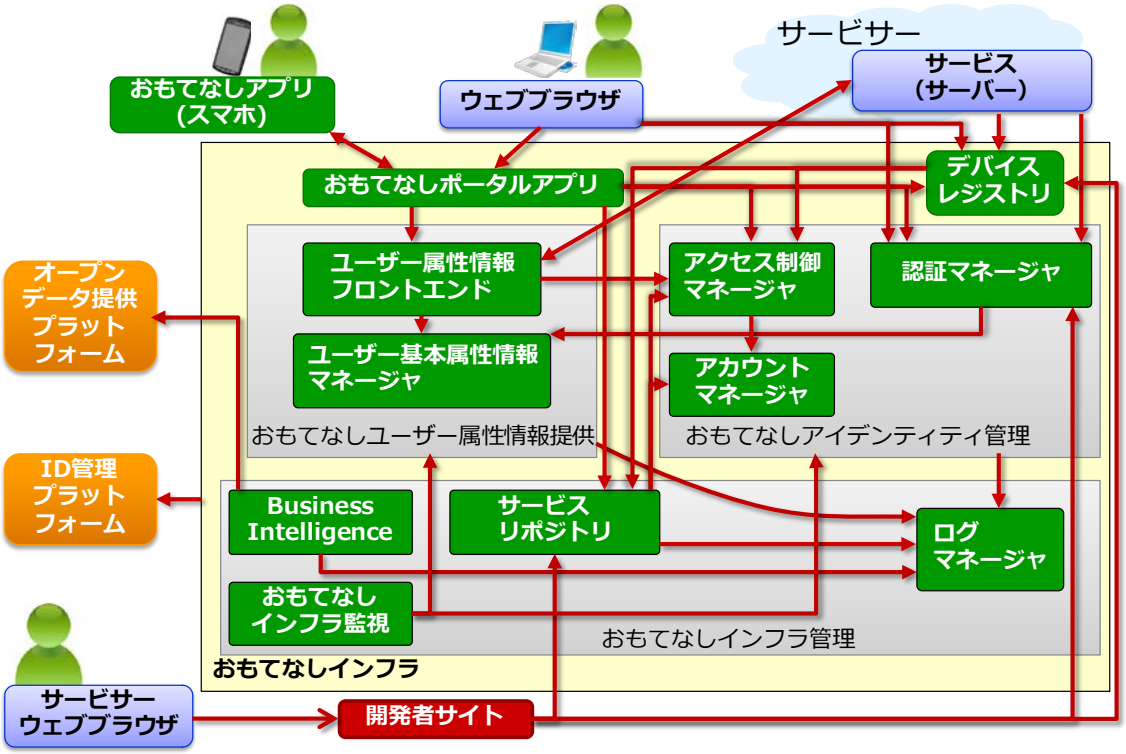

図 2 おもてなしインフラアーキテクチャ

以下、各モジュールについて説明する。

3.1.1 おもてなしユーザー属性情報提供

- ユーザー属性情報フロントエンド
  - おもてなしポータルアプリやサービスからユーザー属性情報の取得要求を受け付け、アクセス権限を確認し、適切な処理を行う
  - 取得要求の種類として、外部のサービスが提供するユーザー属性情報を代理取得してレスポンス中に埋め込むパターンと、外部のサービス ID のみ返却するパターンの 2 種類が指定可能であることを想定する
  - サービスの要求に基づき、ユーザー属性情報の単位変換、表記変換を実施する
  - リンク ID をオリジナル ID と関連づけて管理する
- ユーザー基本属性情報マネージャ
  - ユーザー基本属性情報を返却する
    - ユーザー基本属性情報の内容は 2.1 に示した通りである

3.1.2 おもてなしアイデンティティ管理

- アクセス制御マネージャ
  - アクセス権限の管理を行う

- サービスがユーザー属性情報に対して適切なアクセス権限があるか確認を行う
- 認証マネージャ
  - ユーザーの認証とサービスの認証を行う
  - 様々な認証方式に対応する
    - 本実証実験では少なくとも、ユーザーの認証に交通系 IC カード（FeliCa）と、ユーザー名・パスワードを想定する
    - サービスの認証には OpenID connect[1]やクライアント証明書などの方式が考えられるが、実証実験では広く利用されている手法を判断して採用する
  - 認証要求時には、必要とする信頼度を指定できることとし、結果には認証方式の信頼度も合わせて返却する必要がある
  - ユーザーを識別する ID として、オリジナル ID とリンク ID を利用し、ユーザーの許可に基づいていずれかを返却する
  - ユーザー登録・認証方式とユーザーとの関連付けを実現する
- アカウントマネージャ
  - 実証実験では実現しないが、サービサーの登録を行う際や、API アクセス数に基づいたサービサーの課金ができることを想定する

3.1.3 おもてなしインフラ管理

- サービスリポジトリ
  - サービサーとサービサーによって開発されたサービスを管理する
  - サービサー、サービスの登録時には、それぞれに対応する ID を発行する
    - サービサー登録時にはサービサーID を発行する
    - サービス登録時にはサービス ID を発行する
  - サービスがおもてなしインフラ準拠として提供する API の仕様を記述したサービスプロファイルを管理する
    - サービスプロファイルはサービス登録時に提供される
  - サービスの検索は誰でも実施可能であるが、サービスの登録にはサービサーとして登録済であることが必要である
  - サービサーの登録時に、おもてなしインフラサービサー管理者によるサービサーの審査の必要性について、実証実験で明らかにする必要がある
- ログマネージャ
  - おもてなしインフラ内のモジュールに関する全てのログを集約し、ログ取得・検索可能な API を提供する
- Business intelligence
  - ログマネージャのログを取得して、ユーザーがサービスを利用している状況や、

[1] http://openid.net/connect/

おもてなしインフラの性能などを解析する

- おもてなしインフラ監視
  - おもてなしインフラの死活監視を行い、異常発生時や障害発生時には、該当モジュールの再起動やおもてなしインフラ管理者へのアラートなどを行う
  - Web ユーザインタフェースによる設定画面や監視画面なども提供する

3.1.4 その他

- おもてなしアプリ
  - 本ドキュメントの 2.1.4.1 に記載したおもてなしポータルアプリのスマートフォン用アプリケーションである
  - 実証実験では、実証を行う地域毎におもてなしアプリが開発されることを想定するため、おもてなしアプリはおもてなしインフラの開発対象としない想定とする
- 開発者サイト
  - 本ドキュメントの 2.1.4.2 を参照のこと

3.2 利用フロー

おもてなしインフラを実際に利用する際のフローを、いくつかシチュエーションを例に説明する。なお利用フローは現在の想定であり、実証実験を進めるにあたって修正する可能性がある。

3.2.1 ユーザー登録時の利用フロー

ユーザーの登録は、ユーザー登録用のページでの登録や、登録用に設置するキオスク端末での登録などを想定している。

例えば、ユーザー登録用のページでは、下記手順を完了することにより、交通系 IC カード等での認証も可能となる。

1. ユーザーは認証マネージャの提供するページにてメールアドレスを登録すると、おもてなしインフラによって送信されたメールアドレスの確認メールを受信する
2. ユーザーは、届いたメール内に記載された URL にアクセスすると、ログインのためのパスワードとユーザー基本属性情報を登録するページが表示されるため、必要事項を記載して登録を行う
   - ユーザー基本属性情報は実証実験にて明らかにする
   - 登録時に ID 管理プラットフォームから新たな ID を発行し、ユーザー基本属性情報の一部としてオリジナル ID を保存する
3. ユーザーは、認証マネージャの提供するページにて他の認証方式の関連付けを行う

- 本ページにてユーザーアカウントと交通系 IC カード等との関連付けを実現する

3.2.2 サービス登録時の利用フロー

サービスの登録には、あらかじめサービサーの登録が完了していることが前提である。

サービスの登録には、以下の 5 ステップが必要である。サービス登録時のフローを図 3 に示す。

1. サービサーが開発者サイトへログインする
2. 開発者サイトのサービス登録フォームに必要事項を記入し、サービス登録要求をサービスリポジトリに対して発行する
   - なおサービス登録時に入力が必要な項目は実証実験にて明らかにする
3. クライアント ID 等、サービスの認証に必要な情報と、サービサーがサービスを登録可能かについて、アクセス制御マネージャに問い合わせる
4. サービス登録時にアカウントを記録する。
5. サービス登録時に、ID 管理プラットフォームから新規 ID の割り当てを受けて、サービスに対してサービス ID を付与し、登録を完了する

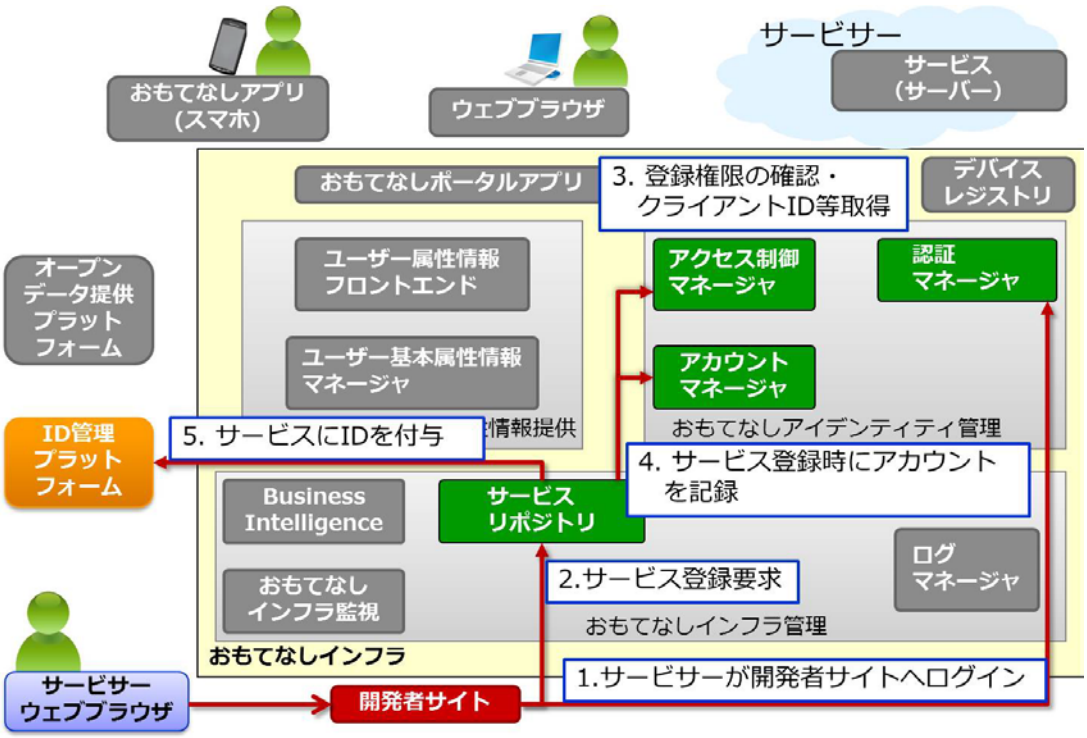

図 3 サービス登録時のフロー

3.2.3　おもてなしアプリ利用時のフロー

ユーザー属性情報に基づいておすすめのサービスリストをおもてなしアプリで表示する場合を例に、おもてなしアプリ利用時のフローを説明する。なお既におもてなしインフラにユーザー登録が完了していることを前提としている。このフローを図 4 に示す。

1. おもてなしアプリにログインする
2. 認証マネージャにてユーザー認証を実施する
   - 認証に成功すると、トークンを発行しておもてなしアプリに返却する
3. おもてなしアプリでサービスリストの表示要求を行う
4. おもてなしポータルアプリからユーザー属性情報の取得要求を行う
5. ユーザー属性フロントエンドでは、要求されたユーザー属性情報に対するアクセス権限をアクセス制御マネージャにて確認する
6. アクセス権限がある場合、ユーザー属性情報を提供するモジュールにアクセスする
7. サービスリストの取得を行い、ユーザー属性情報に基づいたサービスリストの優先順位付け等を行う
8. おもてなしアプリにておすすめサービスリストを表示する

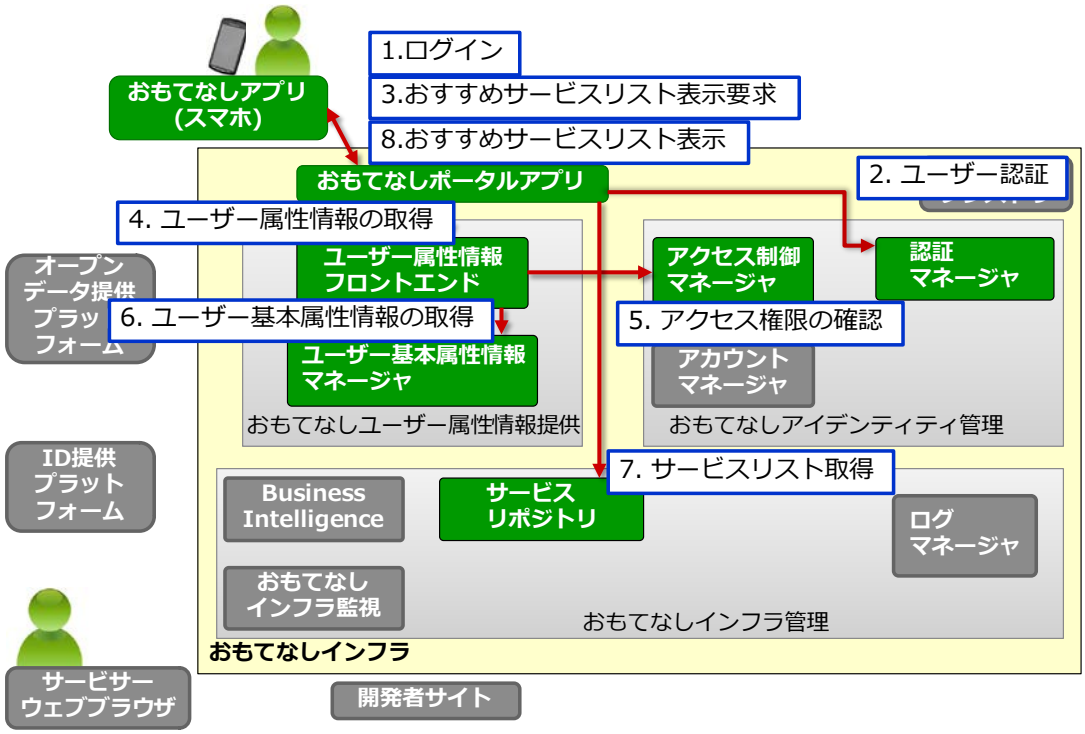

図 4 おもてなしアプリ利用時のフロー

3.2.4　デバイス利用時のフロー

サイネージを例におもてなしインフラのデバイスに関するフローを説明する。なおあらかじめ3.2.2の手順に基づいてデバイスを管理するサービスの登録が完了しているものとする。

3.2.4.1　デバイスの登録

おもてなしインフラに以下の手順でデバイスの情報を登録することを推奨する。デバイスをおもてなしインフラに登録することで、おもてなしポータルアプリでデバイスの案内が実現するなど、おもてなしポータルアプリにてデバイスが紹介できるため、デバイス設置者にとってメリットがある。おもてなしインフラにデバイスを登録するためには、デバイスに対してグローバルでユニークに識別可能な ID を割り当てることが必要である。

1. デバイスまたはサービスが、デバイスとサービスの対応付け等の情報と共に、おもてなしインフラへデバイス登録要求を送る
2. おもてなしインフラはデバイスの登録を行い、デバイスに対してデバイス ID を割り当てる
3. デバイス管理者が、デバイスの設置場所などの情報をおもてなしインフラに登録する
4. サービスのサーバーへのデバイスの登録が完了する

3.2.4.2　ユーザー属性情報に基づくサービスの提供

ユーザー属性情報に基づいてサイネージ等のデバイスがユーザーにサービスを提供するフローは図 5 に示すとおりである。

1. ユーザーが自身の所持する交通系 IC カードをデバイスにタッチすると、デバイスからサービスへログイン要求が発生する
2. タッチされた IC カードの情報を認証マネージャに送る
   - 交通系 IC カードが未登録の場合、ユーザー属性情報を登録するページか、既存のユーザーID と関連付けを行うページへ誘導する
   - 交通系 IC カードが登録済みの場合、アクセストークンと、オリジナル ID かリンク ID を返却する
3. サービスはユーザー属性情報の取得をユーザー属性情報フロントエンドへ要求する
4. ユーザー属性情報フロントエンドでは、利用するユーザー属性情報に対するアクセス権限が本サービスにあるかアクセス制御マネージャに確認し、アクセス権限がない場合はユーザーに許可を依頼する
5. アクセス権限がある場合、ユーザー属性情報を取得する

6. ユーザー属性情報フロントエンドは、取得したユーザー属性情報をサービスへ返却する
7. 取得したユーザー属性情報を基にユーザーへサービスを提供する

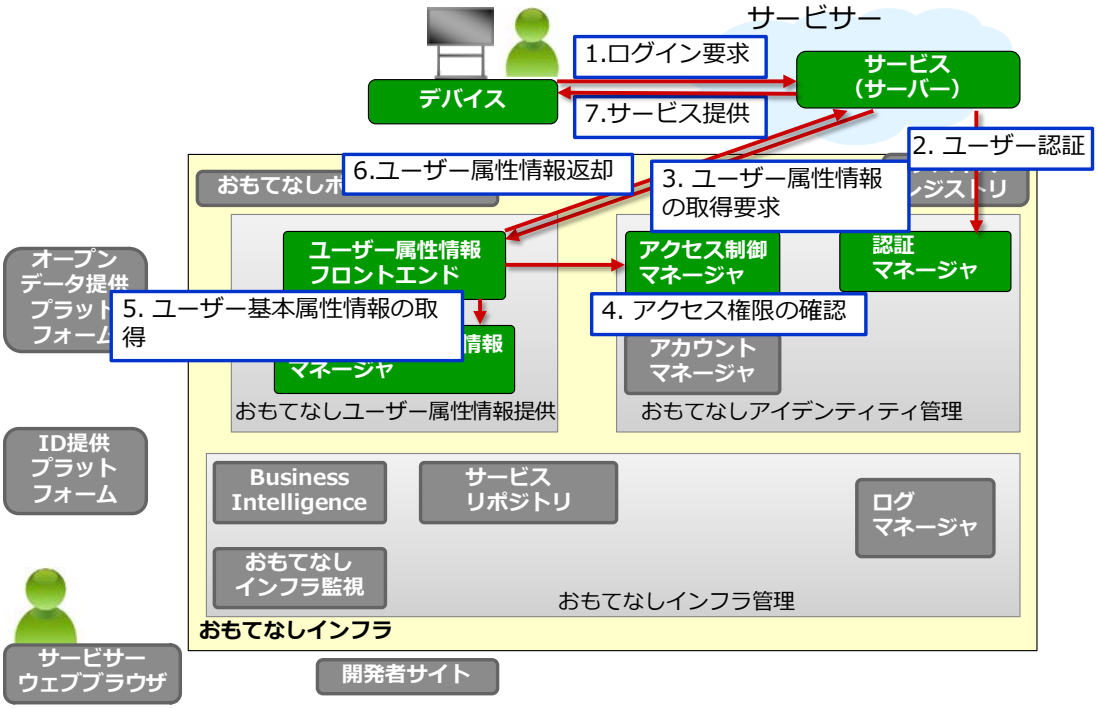

図 5 デジタルサイネージなどのデバイスで属性情報に応じた情報提供などを行うフロー

3.2.5 サービス利用時のフロー

おもてなしインフラのサービスに関するフローを説明する。なお、あらかじめ 3.2.2 の手順に基づいてサービスの登録が完了しているものとする。ユーザー属性情報に基づいてユーザーにサービスを提供するフローを図 6 に示す。ユーザー認証のフローは、説明のために一部簡略化している。

1. ユーザーが自分自身の所持する交通系 IC カードをタッチするか、ユーザー名とパスワードを入力すると、ブラウザからサービスへログイン要求が発生する
2. サービスは入力された認証情報を認証マネージャに送る
   - ユーザーアカウントが未登録の場合、アカウント登録ページを表示する
   - ユーザーアカウントが登録済みの場合、アクセストークンと、オリジナル ID かリンク ID を返却する
3. サービスはユーザー属性情報の取得をユーザー属性情報フロントエンドへ要求する
4. ユーザー属性情報フロントエンドでは、利用するユーザー属性情報に対するアクセス権限が本サービスにあるかアクセス制御マネージャに確認し、アクセス権限がな

い場合はユーザーに許可を依頼する

5. アクセス権限がある場合、ユーザー属性情報を取得する
6. ユーザー属性情報フロントエンドは取得した情報をサービスへ返却する
7. サービスはユーザー属性情報を基にウェブブラウザ向けのコンテンツを提供する

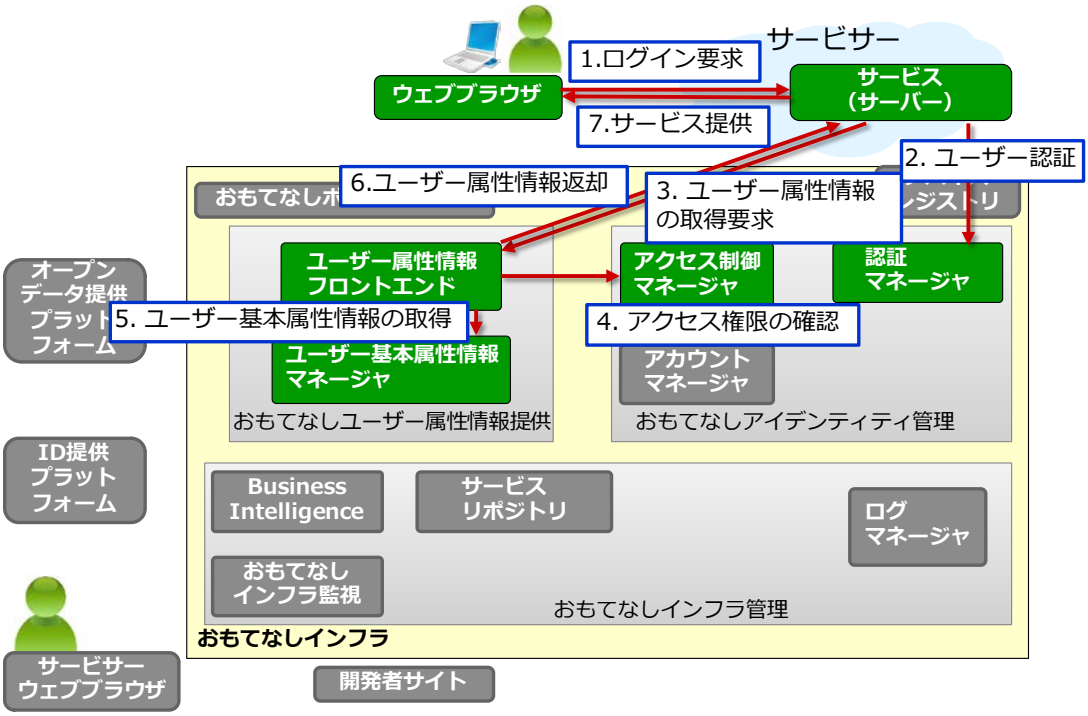

図 6 免税手続きなどのサービスにおけるおもてなしインフラの利用フロー

3.3 アクセス権限のモデル

アクセス権限として 3 種類定義する。なおここで定義するアクセス権限が十分であるかどうかについては、実証実験を通じて明らかにする必要がある。

- サービスアクセス権限
  - 特定のサービスがどのユーザー属性情報に対してアクセスできるかを表現する
- サービサーアクセス権限
  - サービスアクセス権限と同様の内容と、特定のサービサーがユーザー属性情報に対するアクセス権限を他のサービサーに継承できるかと、アクセス権限の多段継承できるかを表現する
- 信頼度別サービサーアクセス権限
  - サービサーの信頼度によって、サービサーの作成したサービスがアクセス可能なユーザー属性情報を表現する
  - これによりユーザが新たに利用するサービスがあらかじめ許可したユーザー属

性情報のみ利用する場合、ユーザー属性情報の利用確認を省略できる

アクセス権限には、以下の特性が必要であると考えられるが、実証実験によって以下の特性で十分かどうかを確認する必要がある。

- アクセス権限の状態には、許可・拒否・未回答の 3 種類が存在する
  - 未回答の場合は、他のアクセス権限に記載されている許可または拒否を採用し、全てのアクセス権限で未回答であれば、拒否とする
- サービスアクセス権限はサービサーアクセス権限よりも優先される
- サービサーアクセス権限は信頼度別サービサーアクセス権限よりも優先される
- サービサーアクセス権限を複数のサービサー間で権限を継承する場合、継承元のサービサーアクセス権限がなくなると、継承先のサービサーアクセス権限も失う

3.4 認証方式

ユーザー認証とサービス認証の方式として OpenID connect が広く普及していることから、OpenID connect 1.0 をベースとした手法に対応することが適切であると考えられるが、実証実験を通して対応する方式を検討すること。またサービス認証の方式としてクライアント証明書を利用する方式も採用事例があることから、クライアント証明書に基づく方式も検討すること。

3.5 運用・保守方式

おもてなしインフラはユーザー数が多いことが予想されるため以下の対応が必要となる。

- おもてなしインフラ内のモジュールの状況を把握するために、誰が何をしたかに関する記録や、それぞれのモジュールの動作状況をログマネージャに集約する
- クラウドサービスを活用して、災害対策のためにマルチリージョンにサーバーを設置することやサーバーの保守にかかる負担を削減する

4 技術要件

4.1 利用する技術

4.1.1 交通系 IC カード

交通系 IC カードとして日本国内で一般的に利用されているのは FeliCa[2]である。交通系 IC カードが保持し、おもてなしインフラに利用できる情報は下記が存在する。

- IDm

[2] http://www.sony.co.jp/Products/felica/about/index.html

- IDm はカードに搭載された製造 ID で、リーダ／ライタが通信相手のカードを識別するために用いられる
- IDm は容易に読み出せることから、当該 IDm と同じふるまいをする装置を製作し使用することにより偽装することが容易に可能であり、IDm の真正性を担保することが難しい
- IDm に任意の情報を関連づけるためには、インターネット接続が必要である

- FeliCa ポケットに独自データを保存
  - FeliCa ポケットとは、カード内の保存領域であり、条件を満たせばこの領域にデータを書き込みできる
  - セキュアにデータの読み書きができる
  - インターネット接続ができなくともデータの読み書きができる
  - IC カードは記憶領域が限られていることから、FeliCa ポケットに対してデータを読み書きできるアプリケーション数やデータサイズは制約がある

おもてなしインフラでは、上記の特徴や、実際のアプリケーションを考慮して認証方法を検討する必要がある。

4.1.2 グローバルユニーク識別子

おもてなしインフラでは、複数のサービス間でユーザー属性情報を連携する必要があるため、ユーザー属性情報やサービス等、おもてなしインフラで扱う対象に対してグローバルでユニークな識別子を付与する必要がある。具体的な要件の例は下記の通りである。

（例）

- 国際標準規格化された識別子の体系であること
- 固定長であること （効率的な処理のため）
- 世界中の人物、サービス、デバイス、場所等を識別するための十分な空間として 128bit 以上の番号空間をもつこと世界中の人物、サービス、デバイス、場所等を識別するための十分な空間をもつこと
- 発行される識別子についてグローバルに唯一性が保証できること
  （同じ ID が 2 度以上発行されないことを保証できること）
- 人物、サービサー、デバイス、サービス等を含め、様々な対象に付与できること
- 管理についてのポリシが公表されており、平等性をもった内容であること
- 本件で想定するサービスを含む各種サービスにおける実用化例を豊富に有すること。特に、サービス間での連携を含む応用事例を持っていること

おもてなしインフラ内では図 7 に示すように、様々な認証方式によってユーザーを認証し、様々なサービス間でユーザー属性情報の連携を実現する。2.1.3 でも説明しているよう

に、オリジナル ID をユーザーに割り当て、このオリジナル ID を様々なサービス間で ID として利用することで、サービスの垣根を超えたデータ連携を実現する。プライバシーの問題に配慮するため、リンク ID も利用する。

下記は一例であるが、様々な ID とオリジナル ID を対応づけることが可能であるため、様々な方式によるユーザーの認証が実現できる。

- IDm
  - FeliCa に割り当てられる ID である
- 旅券番号
  - パスポートに記録されている ID である
- 端末識別子
  - スマートフォン等の端末を表す ID である

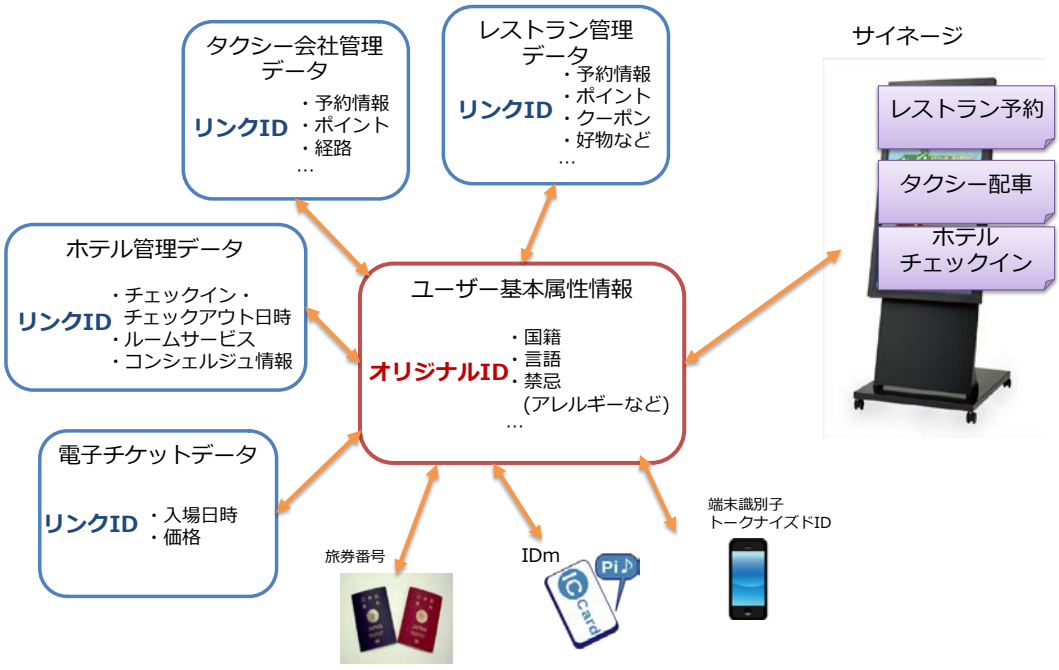

図 7 ID に基づくサービス間のデータ連携

4.2 拡張性

おもてなしインフラを利用するエンドユーザーや参入サービサーの増減予測は困難であると考えられる。また、実証実験のコストを肥大化させないためにも、最小限の構成で検証を開始することが求められる。

つまり、必要に応じて容易に拡張ができることが求められるため、オートスケールに対応したプラットフォームの採用とアプリケーションの実装が必要である。なお、スケーラビリティ実現に際し、事前にプロトタイプ検証による基礎値測定を行い、オートスケール

の発動条件（閾値）を定義する必要がある。

4.3　オープン性

4.3.1　オープンソースソフトウェアの活用

OS、ミドルウェア、データベース、開発言語は一般的に広く採用されている標準技術を採用することが必要である。アプリケーションフレームワークを採用する場合、オープンソースソフトウェアを極力採用する。ただしセキュリティに関する領域は、様々な認証方式に対応する必要があり、認証のためにライセンス料等が必要な方式も存在することから対象外も許容する。

4.3.2　既存決済系システムとの関係

おもてなしインフラではアカウントマネージャというモジュールを実装することを想定するが、おもてなしインフラ自身は決済システムを提供しない。おもてなしインフラは、オープンなシステムであることを標榜していることから、既存の決済系システムを利用できることを目指す。

4.4　パフォーマンス

おもてなしインフラは広く一般に利用されることが想定されるため、クライアントのデバイスの種類を問わず、エンドユーザーからのリクエストに対し、ストレスのないレスポンスを返す必要がある。ストレスのないレスポンス時間の定義は実証実験を通じて具体的にしていく必要があるため、レスポンスタイムを測り、分析する仕組みを検討する必要がある。

4.5　セキュリティ

おもてなしインフラを運用するにあたり、扱う情報のレベルに応じたセキュリティ対策が必要となる。レベルは大きく 2 種類（High、Low）に分類され、個人情報のような秘匿性の高い情報を High、公開が可能なレベルの情報を Low として扱う。なお High、Low のレベルは実証実験を通じ、必要に応じて細分化される可能性がある。

Low レベルの情報をおもてなしインフラ外部に提供する場合には、最低限、名寄せが不可能であるための対策、情報が特定できないためのマスキング処理などを施す必要がある。

High レベルの情報に関しては、不正アクセス、情報漏えい、マルウェア感染、データ改ざん、なりすまし等、世間一般に想定されるセキュリティリスクに対する策を講じる必要がある。具体的な対策として、ウイルス対策や、不正通信のブロックを行う仕組みや、万が一不正通信を行われた場合もそれを迅速に検知する仕組みや、追跡調査が行えるための仕組み（ログの取得／保存など）を導入する必要がある。

情報セキュリティに関するコンプライアンスとして、本サービスの開発者、運営者など、

プロジェクトに関わるメンバーに対しては、セキュリティガイドライン（アクセス権の付与も含む）を制定し、それを遵守させる必要がある。

4.6　信頼性

おもてなしインフラは広く一般に利用されることが想定されるため、以下の要件を満たす必要がある。

- トラブル発生時のデータ復旧が可能であること
- 高いレベルの MTBF （平均故障間隔） と MTTR（平均復旧時間）を保持すること
- 各コンポーネントは多重化され、いずれかのコンポーネントに障害が起きても可能な限りサービスを提供できること

なお、必ずしも実証実験の開始時から高信頼なシステムを構築する必要はないが、必要に応じて容易に信頼性を高められるシステム構成を検討しておく必要がある。また、信頼性を高める時期を開発担当組織が明確にする必要がある。

4.7　可用性

おもてなしインフラは広く一般に利用されることが想定されるため、以下の要件を満たす必要がある。

- 高稼働率と高パフォーマンスレベルの業務継続性を保有すること
  - ただし、サービスに優先順位を定め、縮退可能とする
- 単一障害時はサービスを停止せず、業務の継続を担保すること
- 業務停止時の目標復旧水準は、可及的速やかに障害発生時点まで復旧が可能なシステム構成とすること
- 大規模災害時の目標復旧水準は、世代管理された複製（バックアップ）を保管・管理し、災害時は複製から回復可能なシステム構成とすること

なお、必ずしも実証実験の開始時から高可用性であるシステムを構築する必要はないが、必要に応じて容易に可用性を高められるシステム構成を検討しておく必要がある。また、可用性を高める時期を開発担当組織が明確にする必要がある。

4.8　保守性

おもてなしインフラは広く一般に利用されることが想定されるため、以下の要件を満たす必要がある。

- システム、データのバックアップをデータの重要度・更新頻度等を考慮して設計し、実施すること
- 履歴など参照頻度の低い情報は、一定期間経過後、テープなどの長期保管に適したメディアに保管すること
- 災害等に備えて全体バックアップを定期的に行い、遠隔地に保管すること
- ハードウェア、ネットワーク、ソフトウェアについて、死活監視、エラー監視、リソース監視を行うこと
- 監視項目は、その重要度に応じてアラームレベルを設定できること
- パトランプやメール送付等でおもてなしインフラの運用者にアラームを連絡する手段を設けること
- 環境面は検証系・開発系を必要とするが、冗長度等は最低限のシステム構成とすること
- 通常のメンテナンス、パッチ適用、プログラムリリースでは原則としてシステムを停止させないこと

なお、必ずしも実証実験の開始時から保守性のあるシステムを構築する必要はないが、必要に応じて容易に保守性を高められるシステム構成を検討しておく必要がある。また、保守性を高める時期を開発担当組織が明確にする必要がある。

5　開発要件

5.1　　開発体制

本ドキュメントに記載されている全ての事項について、実現の妥当性と創造性の検討／検証が遂行でき、開発スケジュール、予算計画等を含めた開発計画が無理なく効率的に組まれ、それを踏まえ確実なプロジェクト運営が見込める開発体制が必要である。

上記を踏まえ、交通系 IC カードを活用したクラウド型サービス及び認証に関する開発・商用展開の実績又それと同等の能力を有しており、開発体制における配置要員の役割、責任分担を明確にできる組織に開発を委託することが必要である。

5.2　　開発スケジュール要件

2016 年 11 月からのサービス実証実験に合わせて本基盤の稼動が可能な開発スケジュールである必要がある。ついては、上記要件の実現に向けてマイルストーンを設定し、提示すること。また、設定したマイルストーンについて進捗状況を定期的に確認し、開発上の課題・問題点を早期に把握し、開発期間内に開発終了させるよう調整すること。

プロジェクト運営においては、課題管理表（課題内容、対応者、対応方針、対応結果等）・進捗報告書を作成した上で適宜課題管理と進捗管理を行い、進捗報告は進捗状況に応じて 1～2 回／月程度実施する必要がある。

6　運用要件

6.1　　運用体制

本サービスを運営するにあたり、以下の体制が必要であると考えられる。

- 統合運用窓口
- エンドユーザー向け問い合せ窓口
- サービサー向け問い合せ窓口
- サービス維持管理　（ポータル管理、お知らせ広報）
- システム維持管理　（システム監視、基盤メンテナンス）
- アプリケーション保守　（バグ対応、バージョンアップ対応）

上記体制は実証実験を通じ効率的に運営するために、必要に応じてその役割の統廃合を行う場合がある。そのため、実証実験を行う者は実証実験の段階から運用体制も整備した上で実証実験を行う必要がある。

6.2　　運用スケジュール

2016年11月時点で6.1に記載の運用体制を整備した上で実証実験を開始することが求められる。2016 年 11 月時点で運用体制を整備できない場合、運用体制を整備できる時期を開発担当組織が明確にする必要がある。

7　普及展開

7.1　　事業可能性

2016年度の実証実験のシステムは適切な拡張により、2020年に向けて3000万人程度のユーザー数にも対応できるシステムとし、さらにそれを事業化することが強く望まれる。そのため、以下のような項目で、拡張の実現と事業の確立、さらにその後の継続性の妥当性が検討／検証が遂行できることが求められる。

7.1.1　　協力体制

事業可能性の検討／検証が遂行できるように、システム拡張ための協力体制、また初期にビジネスインするサービサーなど、企業との協力の調整を行い現実性の検証ができる具体的な企業名を含む協力体制。

7.1.2　　ビジネスモデル

2020年以後も持続できる体制確立の妥当性の検討／検証が遂行できるように、事業化のためのマネタイズを中心とした継続可能なビジネスモデル。

7.2　システム拡張と普及展開のスケジュール

本実証実験を通し、サービサーと連携して必要な機能を洗い出して進めていく必要がある。2020 年に向けて 3000 万人程度のユーザー数にも対応できるシステム拡張を進めていく必要がある。

この前提で、前記の事業体制およびビジネスモデルを想定した上で以下のようなマイルストーンを整備できる時期を開発担当組織が明確にする必要がある。

- 最終的に想定されている仕様で、本年度実証で実現されない機能の実装
  - 動的属性（サービス過程で生成・取得される属性）データの利用等
- 各種外部システムの連携時期
  - クレジットカードシステムとの連携等
- 想定する事業体制の確立
- 想定する継続的ビジネスモデルの実施
- 3000 万人体制に対応できるスケールの実現

8　付録: 外部 API