

国土交通省

Project LINKS × 地域交通DX推進プロジェクト「COMmmONS(コモンズ)」

2025年度 GTFS-Flex及びGTFS-Ondemandの技術実証プロジェクト

GTFS-Flex/Ondemand活用システム 技術検証レポート付録



Project LINKS

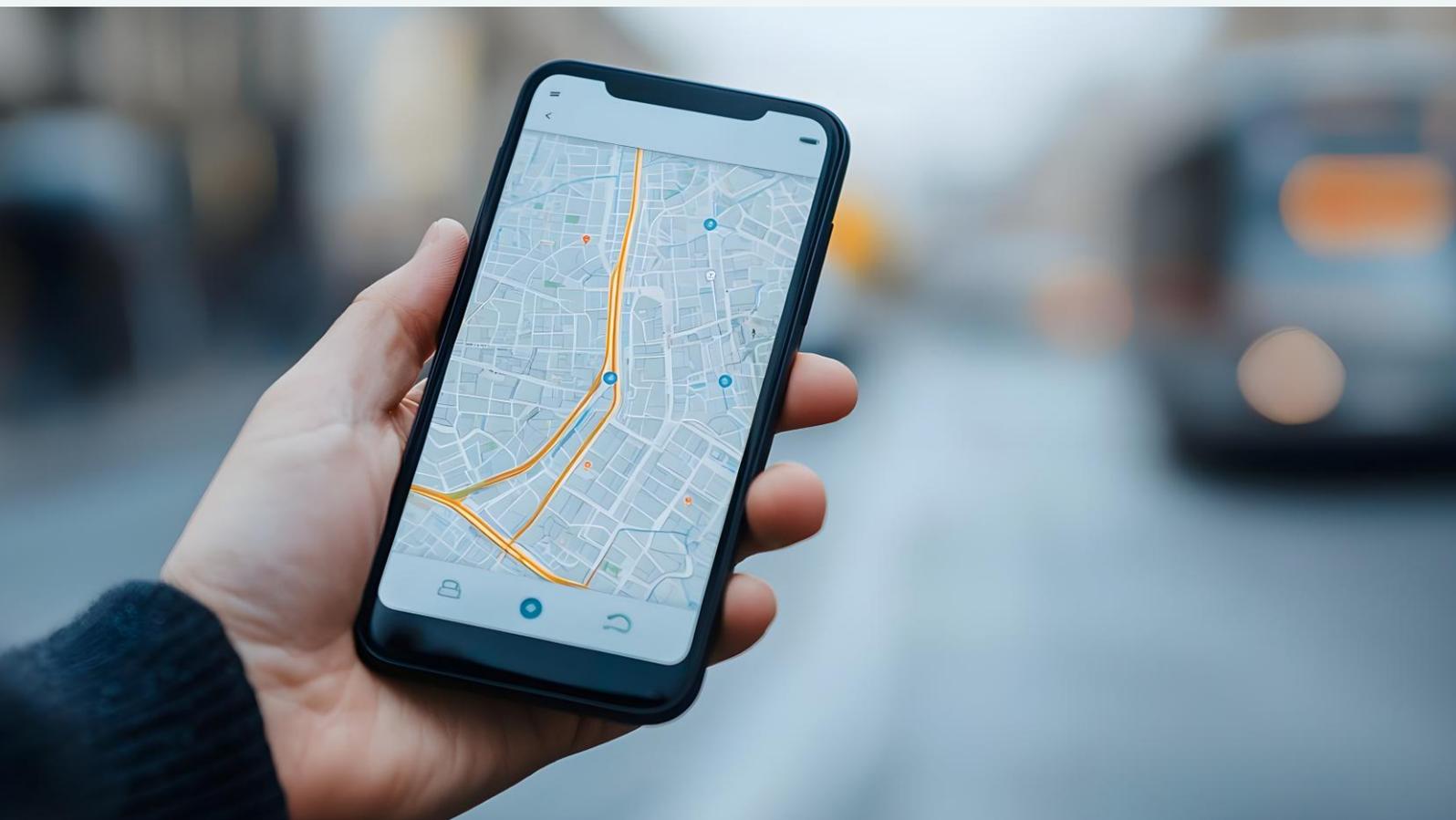


COMmmONS
by MLIT

GTFS-Flex/Ondemand活用システム システム設計書

国土交通省 総合政策局 情報政策課/モビリティサービス推進課

No.009-01



目次

1. 開発スコープ	- 1 -
1-1. 概要	- 1 -
1-2. システムを利用する業務全体像とシステム利用フロー	- 3 -
2. 開発するシステム：機能要件（FN/SL/AL/CO/HW/IF/UI）	- 6 -
2-1. システム機能（FN）	- 6 -
2-1-1. システムアーキテクチャ	- 6 -
2-1-2. システム機能一覧	- 7 -
2-1-3. システム機能の詳細	- 10 -
2-1-4. ソフトウェア・ライブラリ（SL）の詳細	- 42 -
2-1-5. 数理モデル・アルゴリズム（AL）の詳細	- 53 -
2-2. システムコンポーネント（CO）	- 65 -
2-2-1. システムコンポーネント図	- 65 -
2-2-2. システムコンポーネント一覧	- 65 -
2-3. ハードウェア（HW）	- 66 -
2-3-1. ハードウェアアーキテクチャ	- 66 -
2-3-2. ハードウェア一覧	- 66 -
2-3-3. ハードウェアの詳細	- 67 -
2-4. データインターフェース（IF）	- 69 -
2-4-1. データアーキテクチャ	- 69 -
2-4-2. データインターフェース一覧	- 71 -
2-4-3. データインターフェースの詳細	- 73 -
2-5. ユーザーインターフェース（UI）	- 91 -
2-5-1. 画面遷移図	- 91 -
2-5-2. ユーザーインターフェース一覧	- 92 -
2-5-3. ユーザーインターフェースの詳細	- 94 -
3. 開発するシステム：非機能要件（NF）	- 116 -
3-1. 非機能要件一覧	- 116 -
3-2. 非機能要件の詳細	- 117 -
4. 実証調査に利用するデータ（DT）	- 118 -
4-1. 実証調査に利用するデータ一覧	- 118 -
4-2. 実証調査に利用するデータの詳細	- 119 -
5. 用語集	- 138 -

1. 開発スコープ

1-1. 概要

本プロジェクトでは、経路検索サービスにおいてデマンド型交通を取り扱うための最新の国際標準データ仕様の有用性検証を行うため、経路検索アプリケーション及びオンデマンドバス管理システムの開発を行う。具体的には、国際標準化された GTFS-Flex と、現在国際コミュニティで新たに提案されている GTFS-Ondemand の二つのデータ仕様について、データ作成、アプリケーション開発、サービス実証を行う。開発は、経路検索アプリケーションとして「駅探乗換案内」を、オンデマンドバス管理システムとして「SAVS」をそれぞれ改修して実装する。システム全体の主要な処理概要は以下となる。

- ・ 利用者は、駅探乗換案内の UI から定時定路線とデマンド型交通を統合したマルチモーダルな乗換経路を検索する。
- ・ SAVS は、GTFS-Ondemand のリアルタイムメッセージを活用し、駅探乗換案内へオンデマンドバスが配車されるまでのリアルタイムの待ち時間と運賃情報を連携する。
- ・ 駅探乗換案内は、定時定路線交通の乗換検索結果と SAVS から得たオンデマンドバスのリアルタイム情報をつなぎ合わせることで、マルチモーダルな乗換経路を検索する。
- ・ 利用者は、駅探乗換案内の UI 上に示されたマルチモーダル乗換検索結果から、シームレスに SAVS へオンデマンドバスの配車予約を行う。
- ・ オンデマンドバスのドライバーは、SAVS からリアルタイムに指示された配車予約に従ってオンデマンドバスを運行する。

オンデマンドバスを効率良く運行するためには、可能な限り複数の利用者を乗り合わせ、全体として走行経路を最適化しなければならない。しかし際限なく乗り合いをさせては、当初の目的地への到着時間が際限なく遅延することになり、利用者の乗車体験が悪化し、後続の公共交通への乗り換えが失敗する原因にもなり得る。そのため本システムでは、GTFS-Ondemand のリアルタイムメッセージを活用し、オンデマンドバスの出発地及び目的地のリアルタイムに算出した発着日時だけでなく、最遅の場合を想定した発着日時もあわせて SAVS から駅探乗換案内へ連携する。これにより、駅探乗換案内は最遅の状況でのマルチモーダル乗換検索も実施できるようになり、最遅の場合の乗換経路も利用者へ提示が可能となる。

また、駅探乗換案内が実施するマルチモーダル乗換検索は、グラフアルゴリズムを用いた定時定路線交通の経路検索と、GTFS-Ondemand のリアルタイムメッセージを用いて SAVS が配信するリアルタイム配車情報をつなぎ合わせることで実現する。この際、駅探乗換案内の UI は、サービス実証にて実際に運行するオンデマンドバスの情報を網羅的に記載して作成した GTFS-Flex のデータを読み込み、オンデマンドバスのサービス時間や大型荷物の持込可否、平均的な待ち時間などオンデマンドバスの予約に関する情報を利用者へ提示する。なお、GTFS-Flex の静的なデータだけでなく、GTFS-Ondemand のリアルタイムメッセージを用いることでマルチモーダル乗換検索の精度が向上し、利用者のユーザビリティが向上するというを示すために、本プロジェクトでは対照実験として GTFS-Ondemand のリアルタイムメッセージを用いないマルチモーダル乗換検索の UI/UX も検証する。そのため、GTFS-Ondemand のリアルタイムメッセージの利用の有無を切り替えられるようにする。

本プロジェクトではサービス実証として、以下 3 つのユースケースの実証を行う。そのため本システムで

は、GTFS-Flex にて定義されるオンデマンドバスの乗降スポットとサービスエリアを読み込み、出発地や目的地を乗降スポットの一覧やサービスエリアの地図から選択できるようにし、加えて定時定路線交通の駅・バス停からも選択できるようにすることで、定時定路線交通とオンデマンドバスとをまたがった乗り換えを伴う乗車体験の実証ができるようにする。

- ・ Checkpoint 型：時刻表とルートが定まっておらず、事前に定められた停車位置のどこからでも乗降できる
- ・ Zone-to-point 型：サービスエリア内のどこからでも乗車でき、事前に定められた停車位置のどこかで降車できる
- ・ Point-to-zone 型：事前に定められた停車位置のどこかで乗車でき、サービスエリア内のどこでも降車できる

1-2. システムを利用する業務全体像とシステム利用フロー

1. 業務フロー

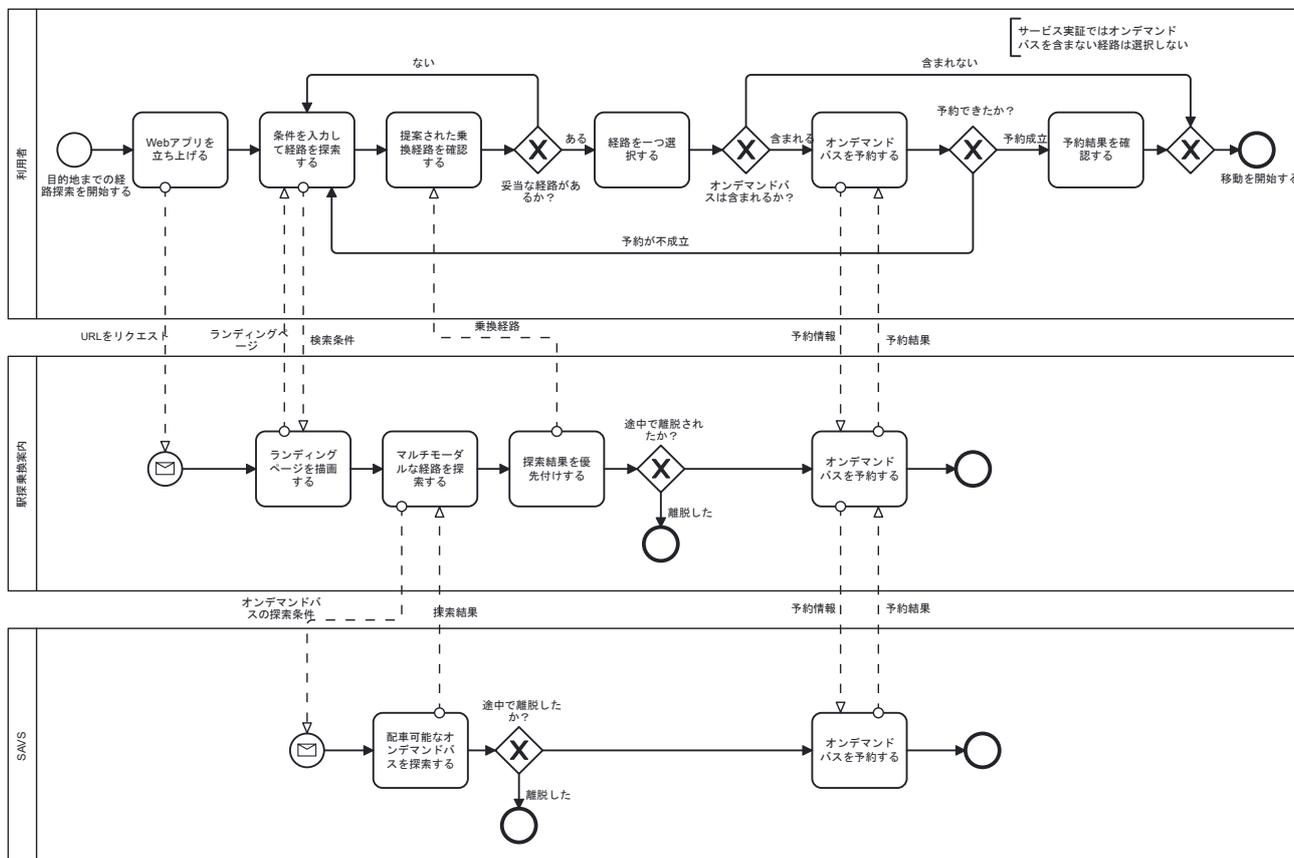


図 1-1 業務フローの全体像

2. システムシーケンス図

① マルチモーダルな経路探索

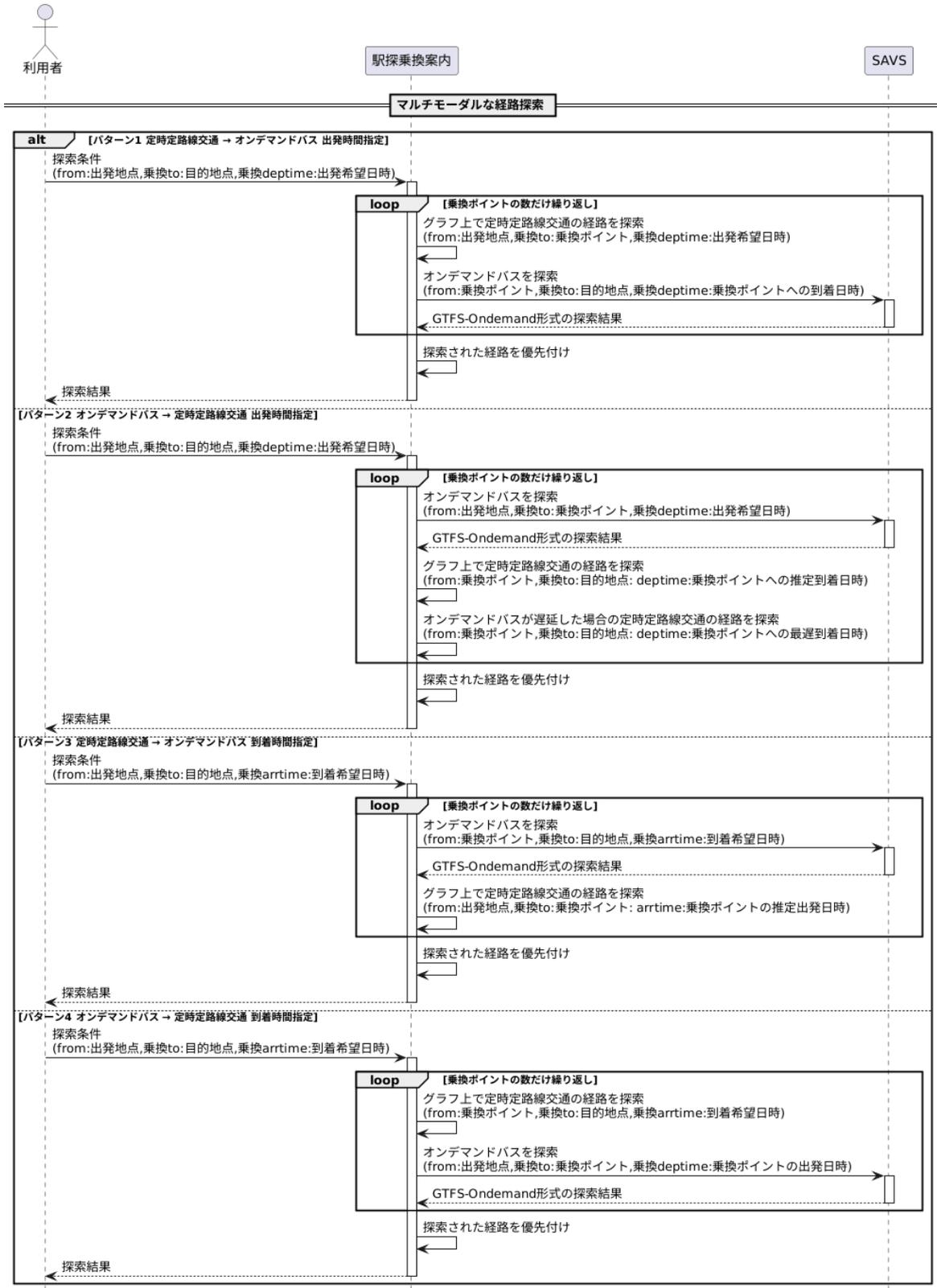


図 1-2 マルチモーダルな経路探索

② オンデマンドバス配車予約

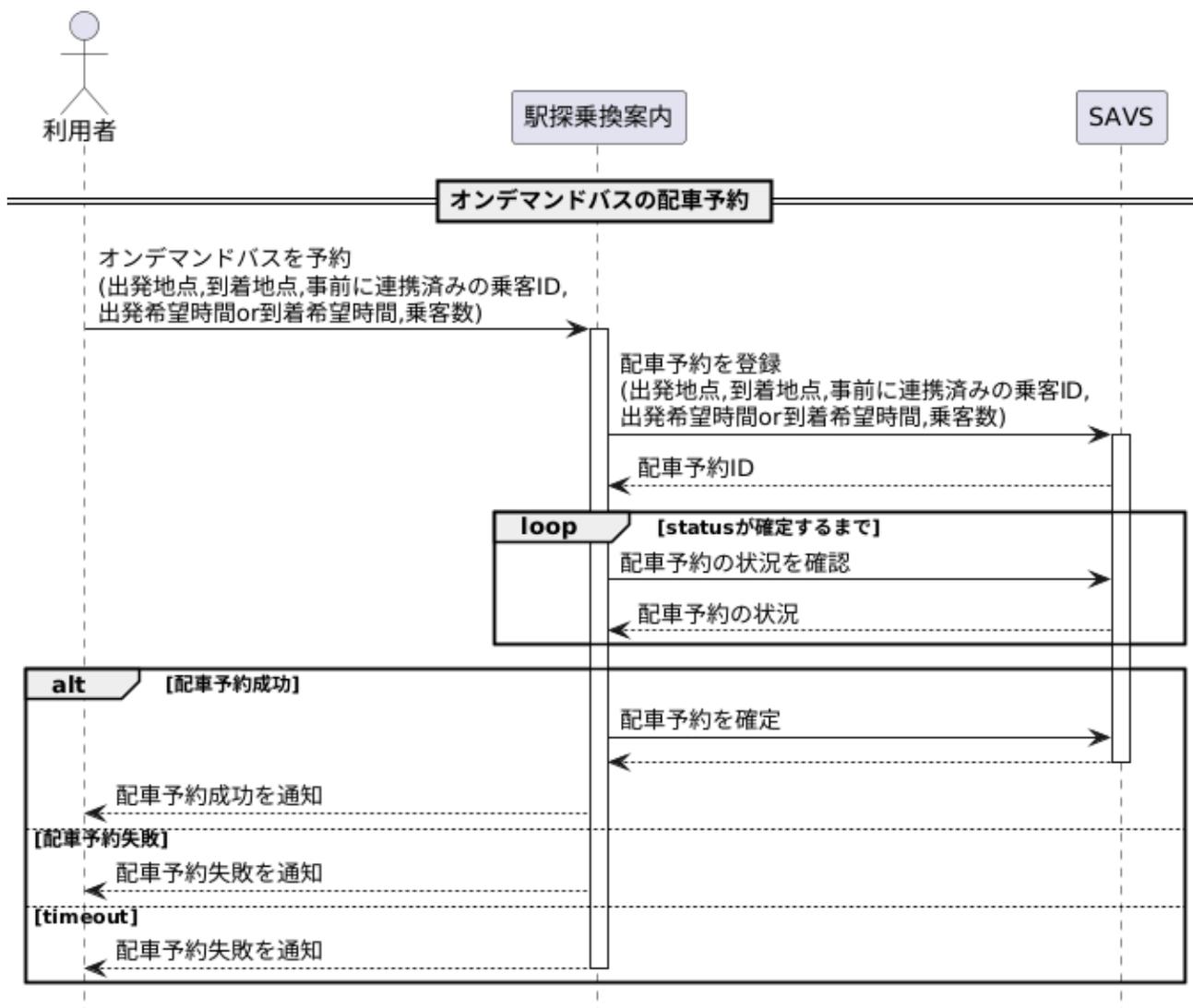


図 1-3 オンデマンドバス配車予約

2. 開発するシステム：機能要件 (FN/SL/AL/CO/HW/IF/UI)

2-1. システム機能 (FN)

2-1-1. システムアーキテクチャ

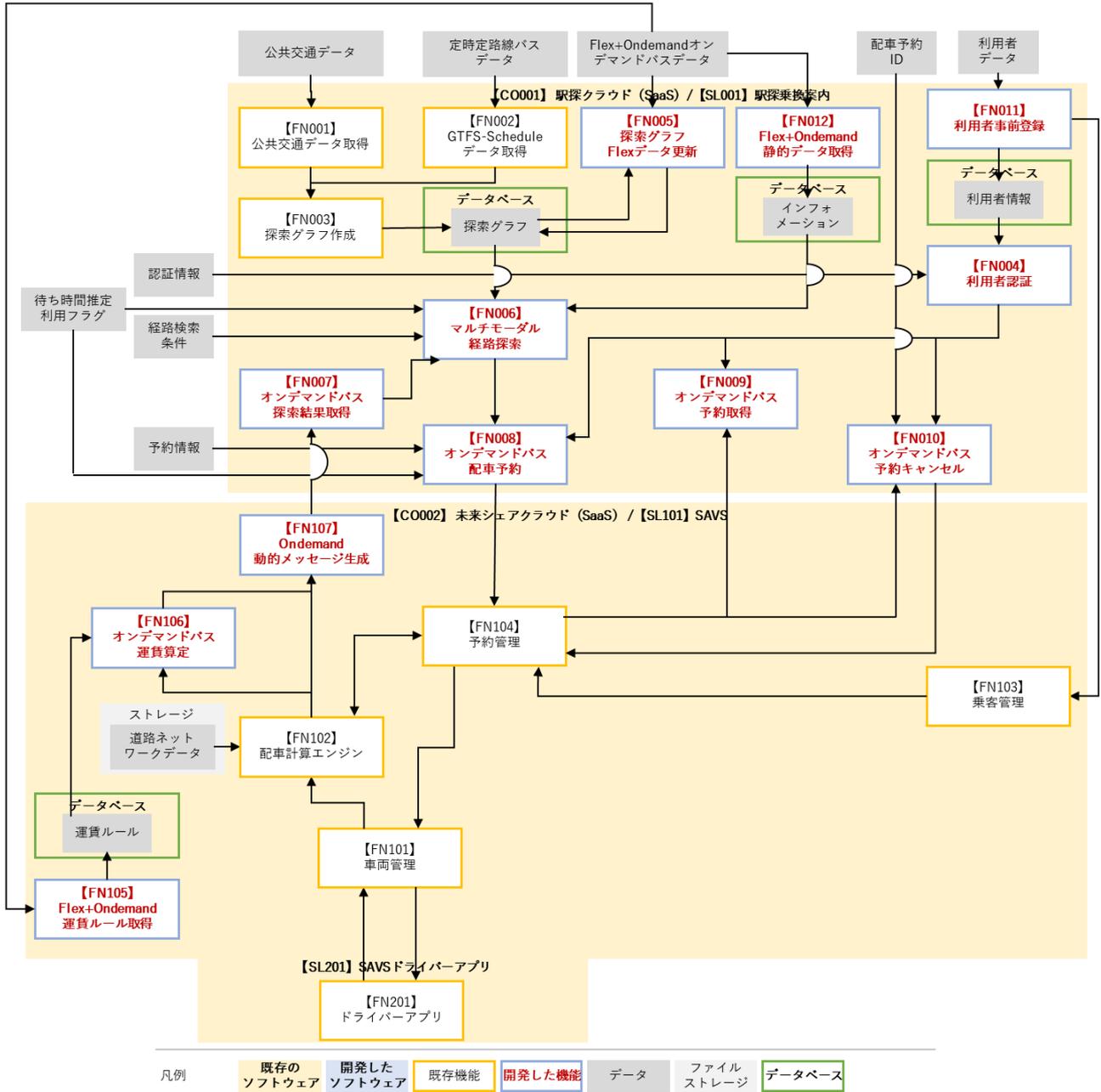


図 2-1 システムアーキテクチャ

2-1-2. システム機能一覧

表 2-1 【SL001】 駅探乗換案内機能一覧

※朱文字：新規開発・既存改修

ID	機能名	機能説明
FN001	公共交通データ取得	● 交通新聞社やJTBパブリッシング等の事業者から取得した公共交通データを取り込む
FN002	GTFS Schedule データ取得	● GTFS Schedule データを取り込む
FN003	探索グラフ作成	● 取り込んだ公共交通のデータを基に、定時定路線交通の経路探索を行うためのグラフを構成する
FN004	利用者認証	● マルチモーダル乗換探索やオンデマンドバス配車予約を行う利用者の認証を行う
FN005	探索グラフ Flex データ更新	● GTFS-Flex 及び GTFS-Ondemand の静的データを取り込み、その Flex 拡張部分に含まれるオンデマンドバスの乗換ポイントに関する情報から、オンデマンド交通の乗換ポイントとなるノードを探索グラフへ追加する
FN006	マルチモーダル経路探索	● 入力された探索グラフ上での定時定路線交通の最短経路探索とオンデマンドバス配車予約アルゴリズムを連携させ、マルチモーダルな乗換探索を実行する
FN007	オンデマンドバス探索結果取得	● 条件に従って推定したオンデマンドバスの待ち時間と運賃を、GTFS-Ondemand のリアルタイムデータ (WaitTimeUpdate と FareUpdate) として取り込む
FN008	オンデマンドバス配車予約	● 探索結果から選択された経路と入力された予約情報を基に、オンデマンドバスの配車予約機能呼び出す
FN009	オンデマンドバス予約取得	● 利用者の配車予約の一覧あるいは詳細を取得する
FN010	オンデマンドバス予約キャンセル	● 乗車前の配車予約をキャンセルする

GTFS-Flex/Ondemand 活用システム システム設計書

FN011	利用者事前登録	<ul style="list-style-type: none"> ● 実証開始前に、利用者の情報をデータベースへ直接登録する
FN012	GTFS-Flex 及び GTFS-Ondemand 静的データ取得	<ul style="list-style-type: none"> ● GTFS-Flex 及び GTFS-Ondemand が拡張した静的データを取り込み、利用者へ提示するインフォメーションデータを取得する

表 2-2 【SL101】SAVS 機能一覧

※朱文字：新規開発・既存改修

ID	機能名	機能説明
FN101	車両管理	<ul style="list-style-type: none"> ● 【FN201】ドライバーアプリから取得した車両情報を【FN102】配車計算エンジンの配車パラメータのひとつとして与える ● 今後予定されている行き先を【FN201】ドライバーアプリへ配信する
FN102	配車計算エンジン	<ul style="list-style-type: none"> ● 対象地域の道路ネットワークデータ、車両の現在位置や今後の行き先、配車予約の条件といった配車パラメータを組み合わせ、AI 配車計算を行う
FN103	乗客管理	<ul style="list-style-type: none"> ● 配車予約の主体となる乗客を登録し管理する
FN104	予約管理	<ul style="list-style-type: none"> ● 出発地点と到着地点、出発時間と到着時間のどちらを指定するか等の配車予約の条件を基に、【FN102】配車計算エンジンを用いてリアルタイムな配車計算を行い、配車予約を登録する ● 登録済みの配車予約情報を取得する ● 登録済みの配車予約をキャンセルする
FN105	GTFS-Flex 及び GTFS-Ondemand 運賃ルール取得	<ul style="list-style-type: none"> ● GTFS-Flex 及び GTFS-Ondemand が拡張した静的データから運賃算定に必要なルールを取得する
FN106	オンデマンドバス 運賃算定	<ul style="list-style-type: none"> ● 【FN105】で得た運賃ルールと【FN102】から得た配車計画を基に、オンデマンドバスの運賃を算定する
FN107	Ondemand 動的メッセージ生成	<ul style="list-style-type: none"> ● 配車計画データから出発地と到着地の待ち時間を推定し、【FN106】で算出した運賃とあわせて WaitTimeUpdate と FareUpdate メッセージで構成される Ondemand メッセージを組み立てる

表 2-3 【SL201】SAVS ドライバーアプリ機能一覧

※朱文字：新規開発・既存改修

ID	機能名	機能説明
FN201	ドライバーアプリ	<ul style="list-style-type: none"> ● 現在位置や乗客の乗車状況など、車両の情報を【FN101】車両管理へ連携する ● 【FN101】車両管理から得た今後予定されている行き先や、車両の現在位置等をドライバーへ提示する

2-1-3. システム機能の詳細

開発機能の詳細要件を記す。なお、本業務において開発や改修を行う内容については、**朱文字**で示す。

【FN001】公共交通データ取得

- 本システム機能の概要
 - 交通新聞社及び JTB パブリッシング社から入手する定時定路線交通のデータを入力し、探索グラフを構成するための中間データ形式に変換する
 - 交通新聞社及び JTB パブリッシング社のデータレイアウトは GTFS 形式とは異なるが、定義されているデータの内容は GTFS 形式で定義された定時定路線交通のデータと相違ないため、【FN002】で扱う路線バスの GTFS Schedule データで十分な検証結果を得られる
- フローチャート

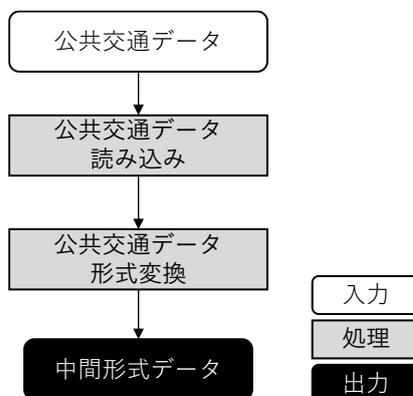


図 2-2 公共交通データ取得のフローチャート

- 本システム機能の処理の詳細
 - 公共交通データ読み込み
 - ◇ 処理内容
 - ファイルとして与えられた公共交通データを読み取る
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
 - 公共交通データ形式変換
 - ◇ 処理内容
 - 読み取った公共交通データを探索グラフ作成機能で用いる中間データ形式に変換する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様

- 入力
 - ◇ 公共交通データ
 - データの内容
 - 交通新聞社や JTB パブリッシング社から入手する公共交通の時刻表データ
 - データの形式
 - 事業者の独自形式ファイル
 - 利用するデータインターフェース
 - 【IF001】公共交通データファイル IF
- 出力
 - ◇ 中間形式データ
 - データの内容
 - 定時定路線交通の探索グラフを構成するための中間形式データ
 - データの形式
 - メモリ上に格納
 - 利用するデータインターフェース
 - 駅探乗換案内の既存機能をそのまま利用

【FN002】GTFS Schedule データ取得

- 本システム機能の概要
 - GTFS Schedule のデータを入力し、探索グラフを構成するための中間データ形式に変換する
- フローチャート

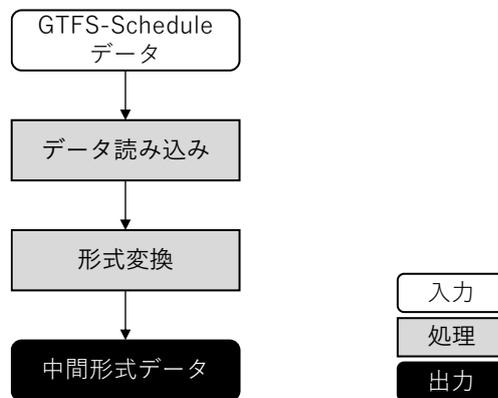


図 2-3 GTFS Schedule データ取得のフローチャート

- 本システム機能の処理の詳細
 - データ読み込み
 - ◇ 処理内容
 - ファイルあるいは情報配信基盤の URL として与えられた GTFS Schedule 形式の路線バスデータを読み取る
 - ◇ 利用するライブラリ
 - なし

- ◇ 利用するアルゴリズム
 - なし
- 形式変換
 - ◇ 処理内容
 - 読み取ったデータを探索グラフ作成機能で用いる中間データ形式に変換する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ GTFS Schedule データ
 - データの内容
 - GTFS-JP 第 3 版に準拠した路線バスデータ
 - データの形式
 - CSV 形式
 - 利用するデータインターフェース
 - 【IF003】GTFS Schedule 定時定路線バスデータファイル IF
 - 出力
 - ◇ 中間形式データ
 - データの内容
 - 定時定路線交通の探索グラフを構成するための中間形式データ
 - データの形式
 - メモリ上に格納
 - 利用するデータインターフェース
 - 駅探乗換案内の既存機能をそのまま利用

【FN003】探索グラフ作成

- 本システム機能の概要
 - 定時定路線交通の駅・バス停の位置、路線や便、時刻表情報を基に、定時定路線交通の乗換経路を
するための重み付き連結グラフを作成する
- フローチャート

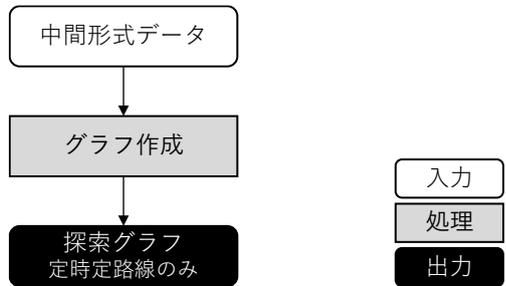


図 2-4 探索グラフ作成のフローチャート

- 本システム機能の処理の詳細
 - グラフ作成
 - ◇ 処理内容
 - 駅探乗換案内の既存機能をそのまま利用
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 中間形式データ
 - データの内容
 - 定時定路線交通の探索グラフを構成するための中間形式データ
 - データの形式
 - メモリ上に格納
 - 利用するデータインターフェース
 - 駅探乗換案内の既存機能をそのまま利用
 - 出力
 - ◇ 探索グラフ（定時定路線のみ）
 - データの内容
 - 単一始点最短経路探索アルゴリズムを実行するための非負数重み付き連結グラフ
 - データの形式
 - データベースに格納
 - 利用するデータインターフェース
 - 駅探乗換案内の既存機能をそのまま利用

【FN004】利用者認証<新規開発>

- 本システム機能の概要
 - 本実証に参加する利用者の認証を行う
- フローチャート

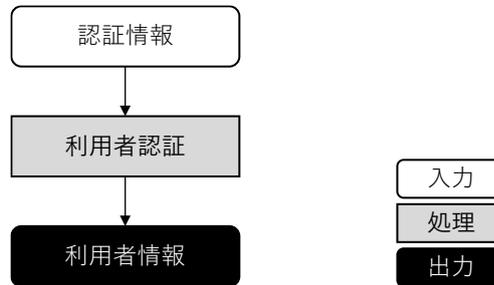


図 2-5 乗客登録のフローチャート

- 本システム機能の処理の詳細
 - 利用者認証
 - ◇ 処理内容
 - 登録済みの利用者情報を用いて利用者を認証する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 認証情報
 - データの内容
 - 利用者の認証に必要な情報
 - データの形式
 - Web フォームデータ
 - 利用するデータインターフェース
 - 【IF014】利用者認証情報入力 IF
 - 出力
 - ◇ 利用者情報
 - データの内容
 - ログインしている利用者の情報
 - ◇ 利用者 ID (文字列)
 - ◇ 対応する SAVS の乗客 ID (文字列)
 - データの形式
 - HTTP セッション
 - 利用するデータインターフェース

➤ なし

【FN005】 探索グラフ Flex データ更新<新規開発>

- 本システム機能の概要
 - GTFS-Flex 及び GTFS-Ondemand の静的データを取り込み、その Flex 拡張部分に含まれるオンデマンドバスの乗換ポイントに関する情報から、オンデマンド交通の乗換ポイントとなるノードを探索グラフへ追加する
- フローチャート

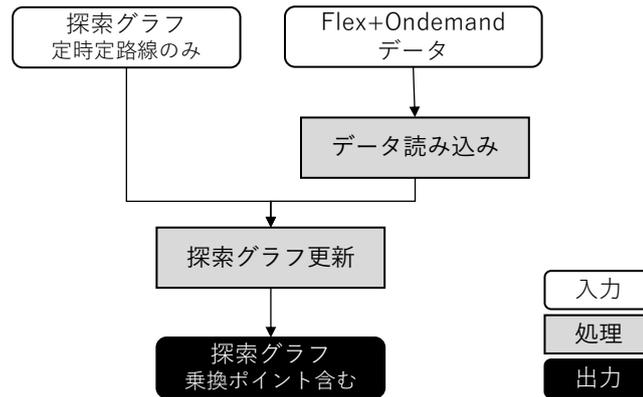


図 2-6 探索グラフ Flex データ更新のフローチャート

- 本システム機能の処理の詳細
 - データ読み込み
 - ◇ 処理内容
 - ファイルあるいは情報配信基盤の URL として与えられた GTFS-Flex 及び GTFS-Ondemand 形式のオンデマンドバスデータを読み取る
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
 - 探索グラフ更新
 - ◇ 処理内容
 - GTFS-Flex 及び GTFS-Ondemand データのうち、Flex 拡張部分に含まれるオンデマンドバスの乗換ポイントに関する情報から、定時定路線交通とオンデマンド交通の一つ以上の乗換ポイントを探索グラフへノードとして追加する
 - 近隣の定時定路線交通の駅・バス停から乗換ポイントまでエッジを接続し、徒歩移動を前提とした所要時間を重みとして設定する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし

- 本システム機能の入出力データの仕様
 - 入力
 - ◇ GTFS-Flex 及び GTFS-Ondemand データ
 - データの内容
 - GTFS-Flex 及び GTFS-Ondemand 形式のオンデマンドバスデータ
 - データの形式
 - CSV 形式及び geojson 形式
 - 利用するデータインターフェース
 - 【IF004】 GTFS-Flex 及び GTFS-Ondemand オンデマンドバスデータファイル IF
 - ◇ 探索グラフ（定時定路線のみ）
 - データの内容
 - 定時定路線交通のデータから【FN003】で出力した、単一始点最短経路探索アルゴリズムを実行するための非負数重み付き連結グラフ
 - データの形式
 - データベースに格納
 - 利用するデータインターフェース
 - 駅探乗換案内の既存機能をそのまま利用
 - 出力
 - ◇ 探索グラフ（乗換ポイント含む）
 - データの内容
 - オンデマンドバスの乗換ポイントノードが追加された、単一始点最短経路探索アルゴリズムを実行するための非負数重み付き連結グラフ
 - データの形式
 - データベースに格納
 - 利用するデータインターフェース
 - 駅探乗換案内の既存機能をそのまま利用

【FN006】 マルチモーダル経路探索 <新規開発>

- 本システム機能の概要
 - 利用者から与えられた探索条件を基に、定時定路線交通とオンデマンドバスを組み合わせたマルチモーダルな乗換経路を探索する
 - ◇ 待ち時間推定利用フラグが ON に設定されている場合、GTFS-Ondemand のリアルタイム情報を用いたオンデマンドバスの推定待ち時間情報と定時定路線交通の経路探索を組み合わせた、乗換待ち時間の推定精度が高いマルチモーダルな乗換経路を探索する。また同時に、推定されたオンデマンドバスの経路を基に算出された運賃も得る
 - ◇ 待ち時間推定利用フラグが OFF に設定されている場合、オンデマンドバスのリアルタイム情報は用いず、定時定路線交通の経路探索とオンデマンドバスの静的な配車可能経路を組み合わせたマルチモーダルな乗換経路を探索する。この場合、運賃は扱わない

- GTFS-Flex 及び GTFS-Ondemand 拡張された静的データから得た利用者向けのインフォメーションを取得し、画面へ表示する

● フローチャート

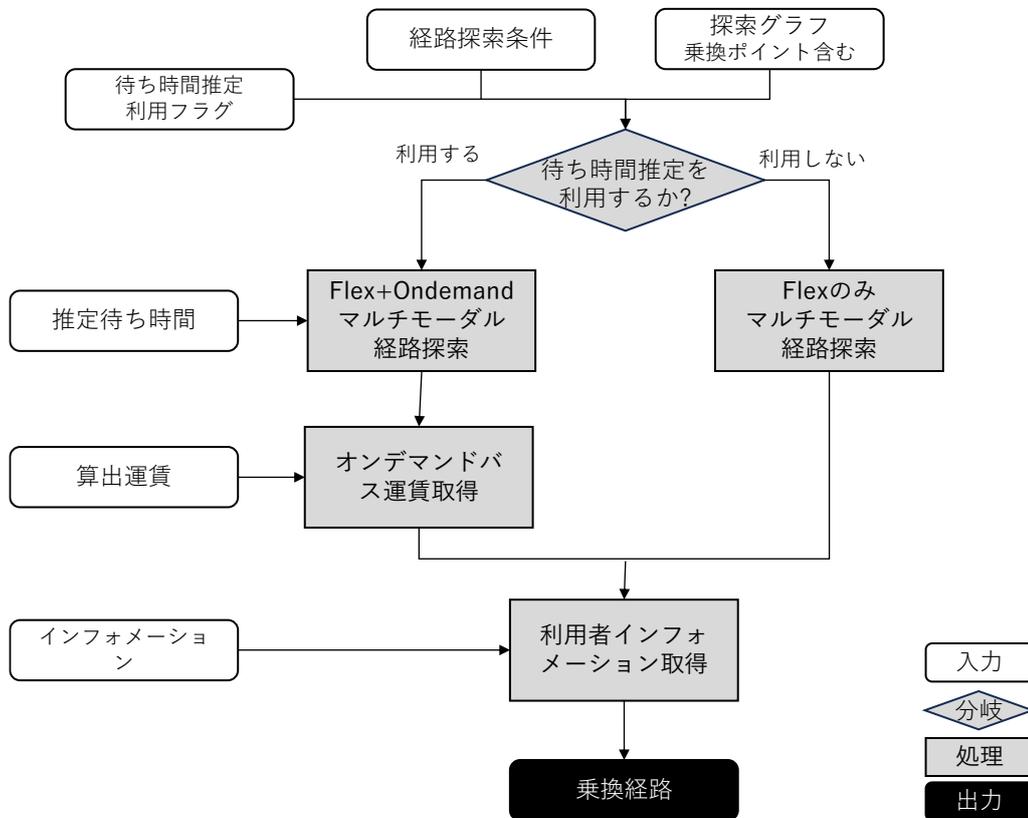


図 2-7 マルチモーダル経路探索のフローチャート

● 本システム機能の処理の詳細

- GTFS-Flex 及び GTFS-Ondemand マルチモーダル経路探索

◇ 処理内容

- 利用者から与えられた探索条件を基に、GTFS-Ondemand のリアルタイム情報を用いたオンデマンドバスの推定待ち時間情報と定時定路線交通の経路探索を組み合わせた、マルチモーダルな乗換経路を探索する
- 定時定路線交通とオンデマンドバスの、どちらからどちらに乗り換えるか、及び出発時間と到着時間のどちらを指定するかの組み合わせで、4つの乗換パターンのいずれかを実行する
 - パターン 1: 出発時間指定 定時定路線交通 → オンデマンドバス
 - パターン 2: 出発時間指定 オンデマンドバス → 定時定路線交通
 - パターン 3: 到着時間指定 定時定路線交通 → オンデマンドバス
 - パターン 4: 到着時間指定 オンデマンドバス → 定時定路線交通

◇ 利用するライブラリ

- なし

◇ 利用するアルゴリズム

- 【AL001】 GTFS-Flex 及び GTFS-Ondemand マルチモーダル経路探索アルゴリズム
 - Flex のみマルチモーダル経路探索
 - ◇ 処理内容
 - 利用者から与えられた探索条件を基に、定時定路線交通の経路探索とオンデマンドバスの静的な配車可能経路を組み合わせたマルチモーダルな乗換経路を探索する
 - 経路探索時点ではオンデマンドバスの配車時間を得ることができないため、定時定路線交通の経路探索をした後に、オンデマンドバスの静的な配車可能経路を付け加えることでマルチモーダルな乗換経路を成すため、以下の2つの乗換パターンのみ実行できる
 - パターン1: 出発時間指定 定時定路線交通 → オンデマンドバス
 - パターン4: 到着時間指定 オンデマンドバス → 定時定路線交通
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - 【AL002】 Flex のみマルチモーダル経路探索アルゴリズム
 - オンデマンドバス運賃取得
 - ◇ 処理内容
 - GTFS-Flex 及び GTFS-Ondemand マルチモーダル検索を行う際に、あわせて算出されるオンデマンドバスの運賃を取得する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
 - 利用者インフォメーション取得
 - ◇ 処理内容
 - 【FN012】 GTFS-Flex 及び GTFS-Ondemand 静的データ取得によって得られた利用者向けインフォメーションをデータベースから取得し、乗換経路と合わせて利用者へ提示する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 経路探索条件
 - データの内容
 - 利用者が入力する乗換経路を探索する条件
 - データの形式
 - Web フォームからの入力

- 利用するデータインターフェース
 - 【IF015】 マルチモーダル経路検索条件入力 IF
- ◇ 探索グラフ（乗換ポイント含む）
 - データの内容
 - オンデマンドバスの乗換ポイントノードが追加された、単一起点最短経路探索アルゴリズムを実行するための非負数重み付き連結グラフ
 - データの形式
 - データベースに格納されたグラフ構造データ
 - 利用するデータインターフェース
 - 駅探乗換案内の既存機能をそのまま利用
- ◇ 推定待ち時間
 - データの内容
 - 【FN007】 から得られる、出発地点及び到着地点でのオンデマンドバスの推定待ち時間
 - データの形式
 - GTFS-Ondemand メッセージ (WaitTimeUpdate メッセージ)
 - 利用するデータインターフェース
 - 【IF006】 GTFS-Ondemand リアルタイムデータ API
- ◇ 算定運賃
 - データの内容
 - 【FN007】 から得られる、出発地点と到着地点から算定されたオンデマンドバスの運賃
 - データの形式
 - GTFS-Ondemand メッセージ (FareUpdate メッセージ)
 - 利用するデータインターフェース
 - 【IF006】 GTFS-Ondemand リアルタイムデータ API
- ◇ インフォメーション
 - データの内容
 - GTFS-Flex 及び GTFS-Ondemand 拡張の静的データから得られた、利用者向けのインフォメーション
 - データの形式
 - データベースに格納
 - 利用するデータインターフェース
 - 【IF005】 GTFS-Flex 及び GTFS-Ondemand インフォメーション IF
- ◇ 待ち時間推定利用フラグ
 - データの内容
 - 待ち時間推定の利用を ON/OFF するフラグ
 - データの形式

- ブラウザの Local Storage にフラグ名をキーとして格納
 - ◇ demand_estimation_gtfs_flg = True / False
- 利用するデータインターフェース
 - 【IF009】 待ち時間推定利用フラグ IF
- 出力
 - ◇ 乗換経路
 - データの内容
 - 優先度でソートされた経路探索の結果
 - データの形式
 - Web 画面として表示
 - ◇ 【UI009】 検索結果画面を参照
 - 利用するデータインターフェース
 - なし

【FN007】 オンデマンドバス探索結果取得<新規開発>

- 本システム機能の概要
 - 【FN006】 の 4 パターンごとに算出されるオンデマンドバスの配車検索条件を基に、【FN107】 Ondemand 動的メッセージ生成機能から得られるオンデマンドバスの待ち時間 (WateTimeUpdate メッセージ) 及び運賃 (FareUpdate メッセージ) を取り込む
- フローチャート

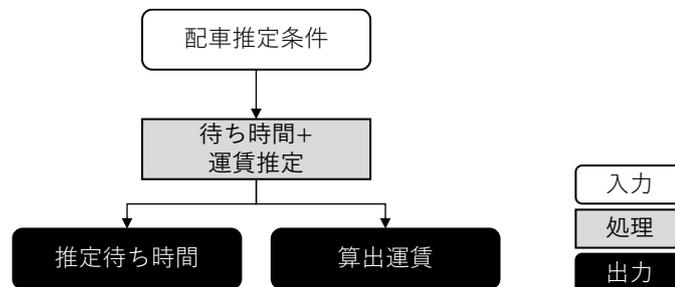


図 2-8 オンデマンドバス探索結果取得のフローチャート

- 本システム機能の処理の詳細
 - 待ち時間+運賃推定
 - ◇ 処理内容
 - 【FN006】 の 4 パターンごとに算出されるオンデマンドバスの配車推定条件を基に、【FN107】 Ondemand 動的メッセージ生成機能呼び出す
 - 【FN107】 から戻されるオンデマンドバスの待ち時間及び算定された運賃を GTFS-Ondemand のリアルタイムデータとして取り込み、出発地点及び到着地点でのオンデマンドバスの推定待ち時間及び算定運賃として【FN006】に戻す
 - ◇ 利用するライブラリ
 - なし

◇ 利用するアルゴリズム

- 【AL001】 GTFS-Flex 及び GTFS-Ondemand マルチモーダル経路探索アルゴリズム

● 本システム機能の入出力データの仕様

➤ 入力

◇ 配車推定条件

- データの内容
 - オンデマンドバスの配車推定をするための条件
- データの形式
 - 関数の引数としてメモリ上に保持
 - ◇ 【IF006】GTFS-Ondemand リアルタイムデータ API のリクエストパラメータと同等
- 利用するデータインターフェース
 - なし

➤ 出力

◇ 推定待ち時間

- データの内容
 - 出発地点及び到着地点でのオンデマンドバスの推定待ち時間
- データの形式
 - GTFS-Ondemand メッセージ (WaitTimeUpdate メッセージ)
- 利用するデータインターフェース
 - 【IF006】 GTFS-Ondemand リアルタイムデータ API

◇ 算出運賃

- データの内容
 - 出発地点と到着地点から算出されるオンデマンドバスの運賃
- データの形式
 - GTFS-Ondemand メッセージ (FareUpdate メッセージ)
- 利用するデータインターフェース
 - 【IF006】 GTFS-Ondemand リアルタイムデータ API

【FN008】 オンデマンドバス配車予約 <新規開発>

- 本システム機能の概要
 - マルチモーダル経路探索結果から利用者が選択した乗換経路と、同乗者数や車いすの有無等の利用者が入力した予約情報を基に、オンデマンドバスの配車予約機能呼び出して配車予約を行う
- フローチャート

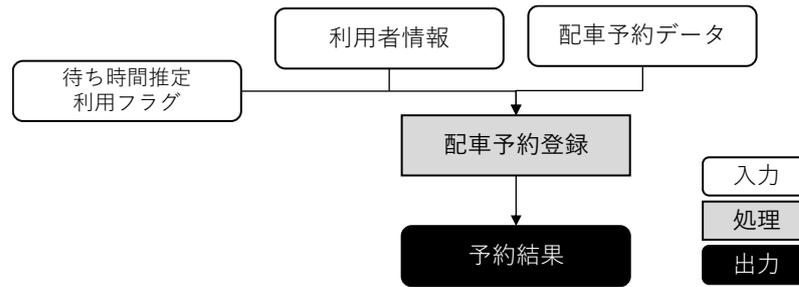


図 2-9 オンデマンドバス配車予約のフローチャート

- 本システム機能の処理の詳細
 - 配車予約登録
 - ◇ 処理内容
 - 配車予約データを基に、【FN104】を呼び出してオンデマンドバスの配車予約を登録する
 - 待ち時間推定利用フラグが ON の場合、配車予約を確定する前に、マルチモーダル経路探索を行った際に利用者へ提示した条件と同等の配車ができることを確認するステップを実行する
 - ほかの利用者の配車予約と競合して到着後の乗り換えに間に合わなくなってしまう場合など、経路探索時に利用者へ提示した条件と同等の配車ができない場合、配車予約を取り消し、利用者へは再び経路探索から実行してもらうように促す
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - 【AL003】オンデマンドバス配車予約アルゴリズム
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 利用者情報
 - データの内容
 - ログインしている利用者の情報
 - ◇ 利用者 ID (文字列)
 - ◇ 対応する SAVS の乗客 ID (文字列)
 - データの形式
 - HTTP セッション
 - 利用するデータインターフェース
 - なし
 - ◇ 配車予約データ
 - データの内容
 - オンデマンドバスの配車を予約するための条件
 - データの形式
 - Web フォーム

- 利用するデータインターフェース
 - 【IF006】 GTFS-Ondemand リアルタイムデータ API
- ◇ 待ち時間推定利用フラグ
 - データの内容
 - 待ち時間推定の利用を ON/OFF するフラグ
 - データの形式
 - ブラウザの Local Storage にフラグ名をキーとして格納
 - ◇ demand_estimation_gtfs_flg = True / False
 - 利用するデータインターフェース
 - 【IF009】 待ち時間推定利用フラグ IF
- 出力
 - ◇ 予約結果
 - データの内容
 - オンデマンドバスの配車予約結果
 - データの形式
 - 画面に表示
 - ◇ 【UI011】 配車予約完了画面を参照
 - 利用するデータインターフェース
 - なし

【FN009】 オンデマンドバス予約取得<新規開発>

- 本システム機能の概要
 - 利用者が予約した配車予約の一覧あるいは詳細を取得する
- フローチャート

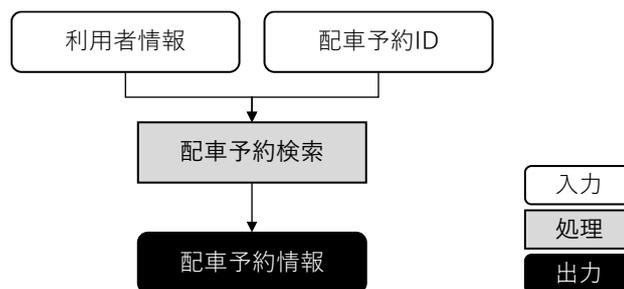


図 2-10 オンデマンドバス予約取得のフローチャート

- 本システム機能の処理の詳細
 - 配車予約検索
 - ◇ 処理内容
 - 【FN104】 予約管理機能呼び出し、ログインしている利用者の配車予約の一覧あるいは指定された配車予約 ID の配車予約を取得する
 - ◇ 利用するライブラリ

- なし
- ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 利用者情報
 - データの内容
 - ログインしている利用者の情報
 - ◇ 利用者 ID (文字列)
 - ◇ 対応する SAVS の乗客 ID (文字列)
 - データの形式
 - HTTP セッション
 - 利用するデータインターフェース
 - なし
 - ◇ 配車予約 ID
 - データの内容
 - 取得する配車予約の ID
 - ◇ 配車予約 ID (整数)
 - データの形式
 - Web フォーム
 - 利用するデータインターフェース
 - なし
 - 出力
 - ◇ 予約結果
 - データの内容
 - 配車予約の一覧あるいは詳細
 - データの形式
 - 画面に表示
 - ◇ 【UI003】 ホーム画面あるいは【UI012】 配車予約詳細画面を参照
 - 利用するデータインターフェース
 - なし

【FN010】 オンデマンドバス予約キャンセル<新規開発>

- 本システム機能の概要
 - 乗車前であれば、利用者が予約した配車予約をキャンセルする
- フローチャート

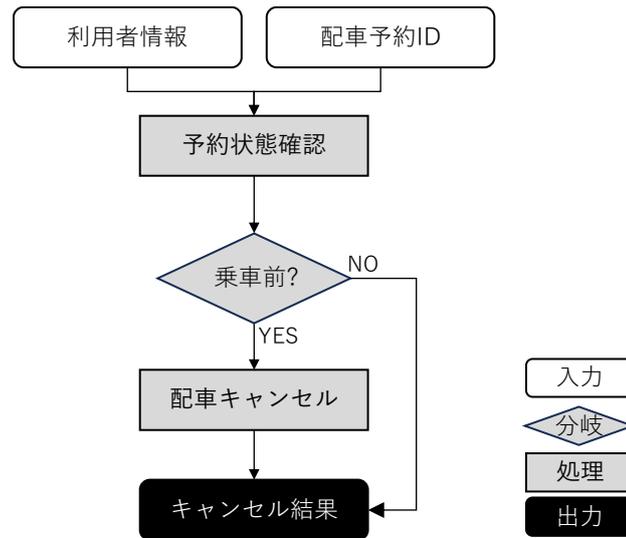


図 2-11 オンデマンドバス予約キャンセルのフローチャート

- 本システム機能の処理の詳細
 - 予約状態確認
 - ◇ 処理内容
 - 【FN104】 予約管理機能呼び出し、指定した配車予約 ID の配車予約の状態を確認する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
 - 配車キャンセル
 - ◇ 処理内容
 - 【FN104】 予約管理機能呼び出し、指定した配車予約 ID の配車予約をキャンセルする
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 利用者情報
 - データの内容
 - ログインしている利用者の情報
 - ◇ 利用者 ID (文字列)
 - ◇ 対応する SAVS の乗客 ID (文字列)
 - データの形式
 - HTTP セッション
 - 利用するデータインターフェース

- なし
- ◇ 配車予約 ID
 - データの内容
 - キャンセルする配車予約の ID
 - ◇ 配車予約 ID (整数)
 - データの形式
 - Web フォーム
 - 利用するデータインターフェース
 - なし
- 出力
 - ◇ キャンセル結果
 - データの内容
 - 配車キャンセルの結果
 - データの形式
 - 画面に表示
 - ◇ **【UI013】配車キャンセル結果画面を参照**
 - 利用するデータインターフェース
 - なし

【FN011】利用者事前登録<新規開発>

- 本システム機能の概要
 - 本実証に参加する被験者が使用する利用者情報を事前登録する
- フローチャート

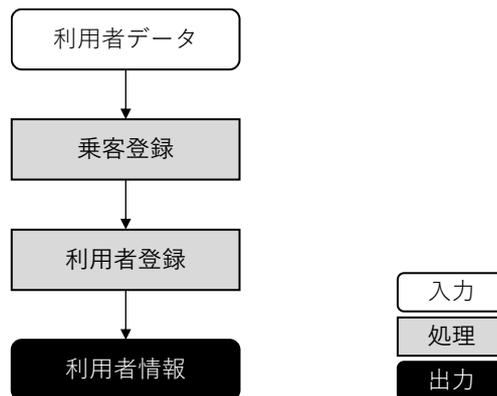


図 2-12 利用者事前登録のフローチャート

- 本システム機能の処理の詳細
 - 乗客登録
 - ◇ 処理内容
 - 利用者データを基に **【FN103】乗客管理**を呼び出し、SAVS の乗客情報を登録する
 - ◇ 利用するライブラリ

- なし
- ◇ 利用するアルゴリズム
 - なし
- 利用者登録
 - ◇ 処理内容
 - 利用者データを基に、SAVS の乗客 ID と対応付けて駅探乗換案内の利用者情報をデータベースへ登録する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 利用者データ
 - データの内容
 - 駅探乗換案内及び SAVS に登録する利用者/乗客の情報
 - データの形式
 - CSV ファイル
 - 利用するデータインターフェース
 - 【IF019】 事前登録用利用者データファイル IF
 - 出力
 - ◇ 利用者情報
 - データの内容
 - 駅探乗換案内の利用者情報

名称	説明	型
id	駅探乗換案内の ID	文字列
password	パスワードを一方方向ハッシュ関数で暗号化した文字列	文字列
savs_id	対応する SAVS の ID	文字列
name	ダミーの利用者名	文字列
phone	ダミーの電話番号	文字列

- データの形式
 - データベースに格納
- 利用するデータインターフェース
 - なし

【FN012】 GTFS-Flex 及び GTFS-Ondemand 静的データ取得<新規開発>

- 本システム機能の概要
 - GTFS-Flex 及び GTFS-Ondemand が拡張した静的データを取り込み、利用者へ提示するインフォ

メーションデータを取得する

- フローチャート

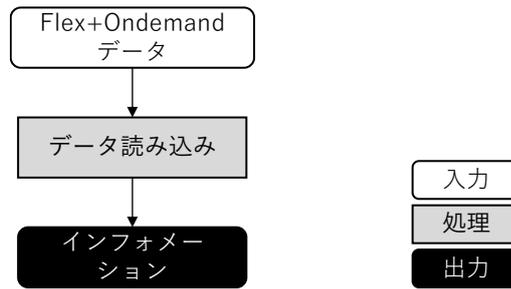


図 2-13 GTFS-Flex 及び GTFS-Ondemand 静的データ取得のフローチャート

- 本システム機能の処理の詳細
 - データ読み込み
 - ◇ 処理内容
 - GTFS-Flex 及び GTFS-Ondemand 形式のオンデマンドバスデータを読み取り、利用者に提示すべきインフォメーションデータを取り出してデータベースへ格納する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ GTFS-Flex 及び GTFS-Ondemand データ
 - データの内容
 - GTFS-Flex 及び GTFS-Ondemand 形式のオンデマンドバスデータ
 - データの形式
 - CSV 形式及び geojson 形式
 - 利用するデータインターフェース
 - 【IF004】 GTFS-Flex 及び GTFS-Ondemand オンデマンドバスデータファイル IF
 - 出力
 - ◇ インフォメーション
 - データの内容
 - GTFS-Flex 及び GTFS-Ondemand 拡張の静的データから得られた、利用者向けのインフォメーション
 - データの形式
 - データベースに格納
 - 利用するデータインターフェース
 - 【IF005】 GTFS-Flex 及び GTFS-Ondemand インフォメーション IF

【FN101】車両管理

- 本システム機能の概要
 - 【FN201】ドライバーアプリから取得した車両情報を【FN102】配車計算エンジンの配車パラメータのひとつとして与える
 - 今後予定されている行き先を【FN201】ドライバーアプリへ配信する
- フローチャート

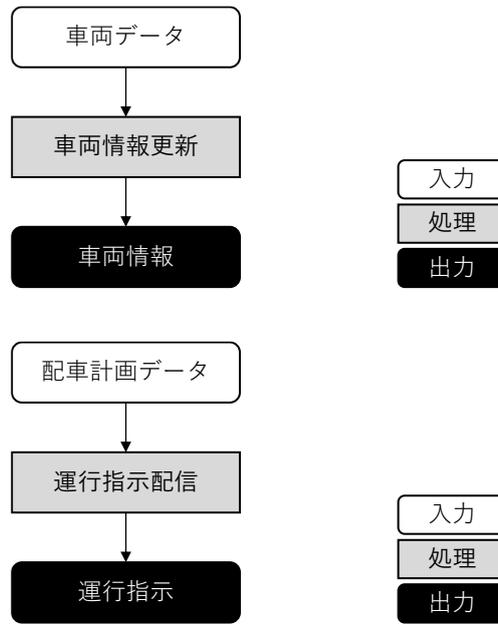


図 2-14 車両管理のフローチャート

- 本システム機能の処理の詳細
 - 車両情報更新
 - ◇ 処理内容
 - 【FN201】ドライバーアプリから取得した車両情報を【IF029】を経て【FN102】配車計算エンジンの配車パラメータのひとつとして与える
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
 - 運行指示配信
 - ◇ 処理内容
 - 【FN102】配車計算エンジンがリアルタイムに計算した配車計画データを基に、今後予定されている行き先を【FN201】ドライバーアプリへ配信する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし

- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 車両データ
 - データの内容
 - 現在位置など【FN201】ドライバーアプリが送信する車両のリアルタイムデータ
 - データの形式
 - リクエストパラメータ
 - 利用するデータインターフェース
 - 【IF012】車両情報更新 API
 - ◇ 配車計画データ
 - データの内容
 - 【FN102】配車計算エンジンがリアルタイムに計算した配車計画
 - データの形式
 - メモリ上に格納
 - 利用するデータインターフェース
 - SAVS の既存機能をそのまま利用
 - 出力
 - ◇ 車両情報
 - データの内容
 - 【FN201】から得た車両データによって随時更新される車両の現在位置等の情報
 - データの形式
 - 関数の引数としてメモリ上に保持
 - 利用するデータインターフェース
 - 【IF029】車両情報 IF
 - ◇ 運行指示
 - データの内容
 - 停車すべき乗降ポイントとその順序など、車両が運行すべき経路の指示
 - データの形式
 - リクエストパラメータ
 - 利用するデータインターフェース
 - 【IF013】運行指示 API

【FN102】配車計算エンジン

- 本システム機能の概要
 - 対象地域の道路ネットワークデータ、車両の現在位置や今後の行き先、配車予約の条件といった配車パラメータを組み合わせ、AI 配車計算を行う
- フローチャート

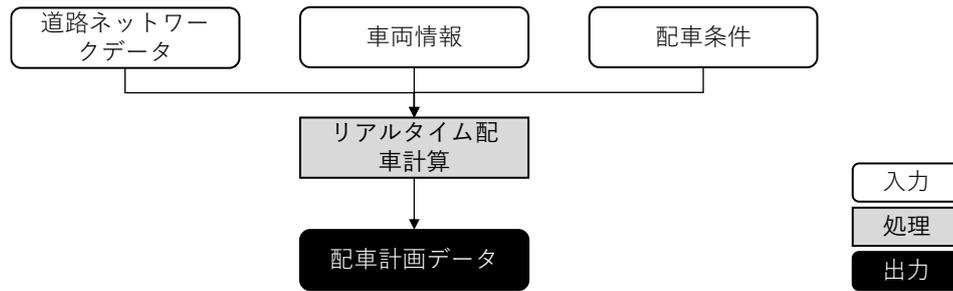


図 2-15 配車計算エンジンのフローチャート

- 本システム機能の処理の詳細
 - リアルタイム配車計算
 - ◇ 処理内容
 - 対象地域の道路ネットワークデータ、車両の現在位置や今後の走行予定経路及び利用者からの配車条件に従い、リアルタイムに最適な便乗配車計算を行う
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - 逐次最適挿入法によるデマンド発生時の即時配車決定
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 道路ネットワークデータ
 - データの内容
 - 乗合配車による時間の遅れや迂回時間等を考慮した出発推定日時、到着推定日時を計算するために用いるデータ
 - データの形式
 - OpenStreetMap 形式
 - 利用するデータインターフェース
 - 【IF002】道路ネットワークデータ
 - ◇ 車両情報
 - データの内容
 - 【FN101】によって随時更新される車両の現在位置等の情報
 - データの形式
 - 関数の引数としてメモリ上に保持
 - 利用するデータインターフェース
 - 【IF012】車両情報更新 API
 - ◇ 配車条件
 - データの内容
 - オンデマンドバスの配車を計算するための条件
 - データの形式

- 関数の引数としてメモリ上に保持
 - ◇ 【IF006】GTFS-Ondemand リアルタイムデータ API のリクエストパラメータと同等
 - 利用するデータインターフェース
 - なし
- 出力
 - ◇ 配車計画データ
 - データの内容
 - 計算した結果の配車計画
 - データの形式
 - メモリ上に格納
 - 利用するデータインターフェース
 - 【IF028】リアルタイム配車計算結果 IF

【FN103】乗客管理

- 本システム機能の概要
 - 配車予約の主体となる乗客を登録し管理する。
- フローチャート

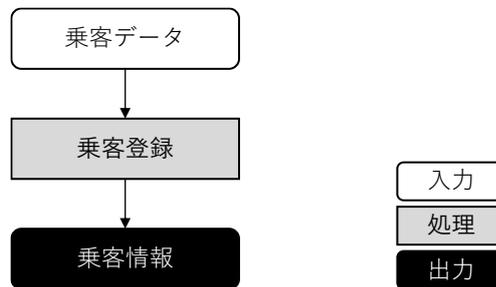


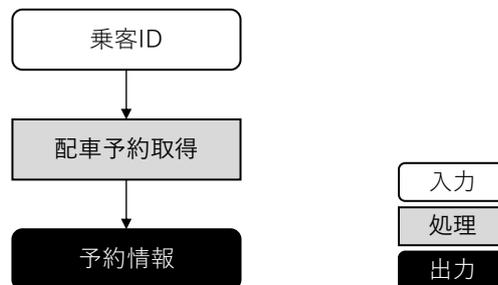
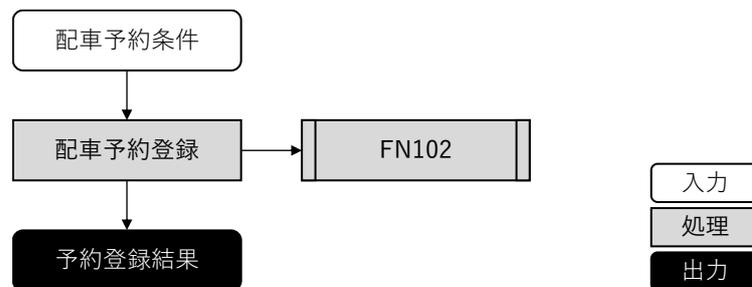
図 2-16 乗客管理のフローチャート

- 本システム機能の処理の詳細
 - 配車予約確認
 - ◇ 処理内容
 - 配車予約の主体となる乗客を登録する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 乗客データ
 - データの内容

- 乗客登録に必要な情報
 - データの形式
 - リクエストパラメータ
 - 利用するデータインターフェース
 - 【IF007】乗客登録 API
- 出力
 - ◇ 乗客情報
 - データの内容
 - 乗客情報
 - データの形式
 - データベースに格納
 - ◇ 【IF007】乗客登録 API のレスポンスと同等
 - 利用するデータインターフェース
 - なし

【FN104】予約管理

- 本システム機能の概要
 - 出発地点と到着地点、出発時間と到着時間のどちらかを指定するか等の配車予約の条件を基に、
 - 【FN102】配車計算エンジンを用いてリアルタイムな配車計算を行い、配車予約を登録する
 - 登録済みの配車予約情報を取得する
 - 登録済みの配車予約をキャンセルする
- フローチャート



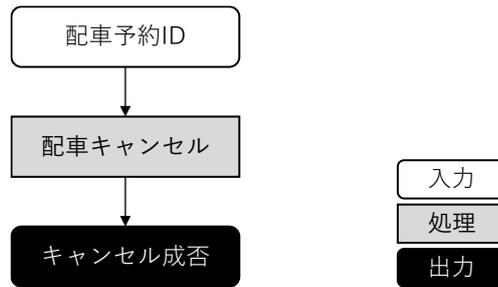


図 2-17 予約管理のフローチャート

- 本システム機能の処理の詳細
 - 配車予約登録
 - ◇ 処理内容
 - 【FN102】を呼び出してリアルタイムに配車計算を行い、配車予約を登録する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
 - 配車予約取得
 - ◇ 処理内容
 - 乗客 ID に対応する配車予約の状態を一覧で取得する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
 - 配車キャンセル
 - ◇ 処理内容
 - 指定された配車予約 ID の配車予約をキャンセルする
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 配車予約条件
 - データの内容
 - オンデマンドバスの配車を予約するための条件
 - データの形式
 - リクエストパラメータ
 - 利用するデータインターフェース

- 【IF008】配車予約登録 API
- ◇ 乗客 ID
 - データの内容
 - 配車予約を検索する乗客 ID
 - データの形式
 - リクエストパラメータ
 - 利用するデータインターフェース
 - 【IF017】配車予約取得 API
- ◇ 配車予約 ID
 - データの内容
 - 配車登録時に得た配車予約 ID
 - データの形式
 - リクエストパラメータ
 - 利用するデータインターフェース
 - 【IF018】配車キャンセル API
- 出力
 - ◇ 予約登録結果
 - データの内容
 - 配車予約登録の結果
 - データの形式
 - リクエストレスポンス
 - 利用するデータインターフェース
 - 【IF008】配車予約登録 API
 - ◇ 予約情報
 - データの内容
 - 当該乗客 ID に紐づく配車予約の一覧
 - データの形式
 - リクエストレスポンス
 - 利用するデータインターフェース
 - 【IF017】配車予約取得 API
 - ◇ キャンセル成否
 - データの内容
 - 配車キャンセルの成否
 - データの形式
 - リクエストレスポンス
 - 利用するデータインターフェース
 - 【IF018】配車キャンセル API

【FN105】 GTFS-Flex 及び GTFS-Ondemand 運賃ルール取得<新規開発>

- 本システム機能の概要
 - GTFS-Flex 及び GTFS-Ondemand が拡張した静的データから運賃算定に必要なルールを取得する
- フローチャート

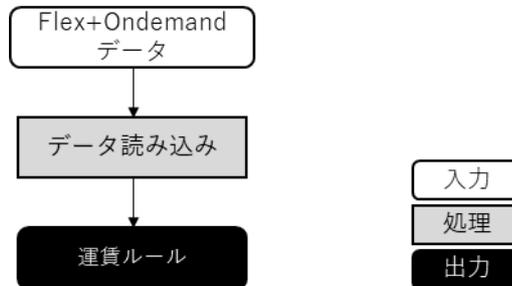


図 2-18 GTFS-Flex 及び GTFS-Ondemand 運賃ルール取得のフローチャート

- 本システム機能の処理の詳細
 - データ読み込み
 - ◇ 処理内容
 - GTFS-Flex 及び GTFS-Ondemand 形式のオンデマンドバスデータを読み取り、運賃ルールを取得してデータベースへ格納する
 - 本実証では、以下の運賃ルールを定義する
 - 乗車地点~降車地点が 0.5km 以内 = 開始料金のみ (100 円)
 - 乗車地点~降車地点が 0.5km 以上 = 最初の 0.5km までは開始料金 (100 円) + 0.5km 以降は 0.25km ごとに 10 円を加算 (最大料金はなし)
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ GTFS-Flex 及び GTFS-Ondemand データ
 - データの内容
 - GTFS-Flex 及び GTFS-Ondemand 形式のオンデマンドバスデータ
 - データの形式
 - CSV 形式及び geojson 形式
 - 利用するデータインターフェース
 - **【IF004】** GTFS-Flex 及び GTFS-Ondemand オンデマンドバスデータファイル IF
 - 出力
 - ◇ 運賃ルール
 - データの内容

- オンデマンドバスの運賃を算出するために用いる運賃ルール
- データの形式
 - データベースに格納
- 利用するデータインターフェース
 - 【IF010】 GTFS-Flex 及び GTFS-Ondemand 運賃ルール IF

【FN106】 オンデマンドバス運賃算定 <新規開発>

- 本システム機能の概要
 - 【FN105】で得た運賃ルールと【FN102】から得た配車計画を基に、オンデマンドバスの運賃を算定する
- フローチャート

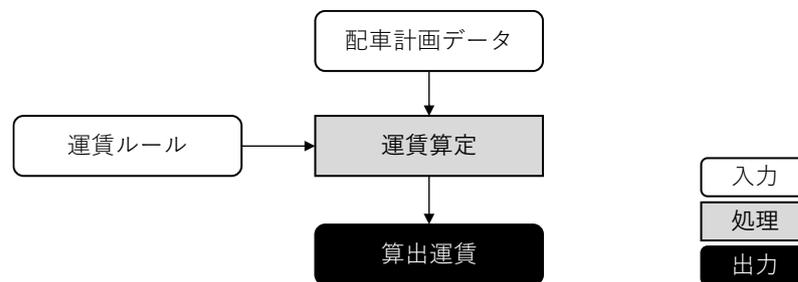


図 2-19 オンデマンドバス運賃算定のフローチャート

- 本システム機能の処理の詳細
 - 運賃算定
 - ◇ 処理内容
 - 【FN102】によって得られた配車計画データと【FN105】から得た運賃ルールを基に、出発地点から到着地点までのオンデマンドバス運賃を算定する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
- 本システム機能の入出力データの仕様
 - 入力
 - ◇ 配車計画データ
 - データの内容
 - 計算した結果の配車計画
 - データの形式
 - メモリ上に格納
 - 利用するデータインターフェース
 - 【IF028】リアルタイム配車計算結果 IF
 - ◇ 運賃ルール
 - データの内容

- オンデマンドバスの運賃を算出するために用いる運賃ルール
 - データの形式
 - データベースに格納
 - 利用するデータインターフェース
 - 【IF010】 GTFS-Flex 及び GTFS-Ondemand 運賃ルール IF
- 出力
 - ◇ 算定運賃
 - データの内容
 - 出発地点から到着地点までで算定されたオンデマンドバスの運賃
 - データの形式
 - メモリ上に格納
 - 利用するデータインターフェース
 - 【IF011】 算定運賃 IF

【FN107】 Ondemand 動的メッセージ生成<新規開発>

- 本システム機能の概要
 - 【FN102】 で推定したオンデマンドバスの待ち時間と 【FN106】 で算出した運賃を、GTFS-Ondemand が拡張した WaitTimeUpdate と FareUpdate メッセージとして返す
- フローチャート

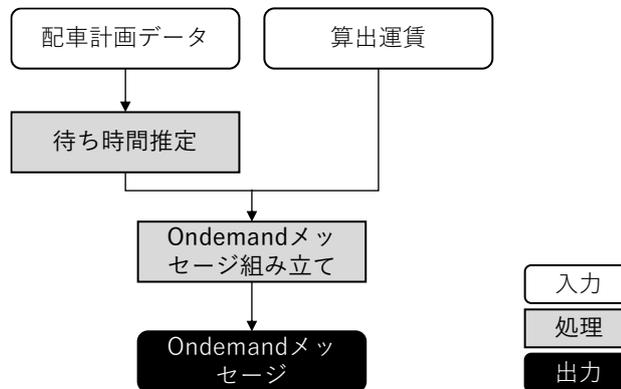


図 2-20 Ondemand 動的メッセージ生成のフローチャート

- 本システム機能の処理の詳細
 - 待ち時間推定
 - ◇ 処理内容
 - 配車計画データを基に出発地及び到着地でのオンデマンドバスの推定待ち時間を推定する
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし

- Ondemand メッセージ組み立て
 - ◇ 処理内容
 - 出発地と到着地の待ち時間、及び【FN106】で算出した運賃を基に WaitTimeUpdate と FareUpdate メッセージで構成される Ondemand メッセージを組み立てる
 - ◇ 利用するライブラリ
 - なし
 - ◇ 利用するアルゴリズム
 - なし
 - 本システム機能の入出力データの仕様
 - 入力
 - ◇ 配車計画データ
 - データの内容
 - 計算した結果の配車計画
 - データの形式
 - メモリ上に格納
 - 利用するデータインターフェース
 - 【IF028】リアルタイム配車計算結果 IF
 - ◇ 算定運賃
 - データの内容
 - 出発地点から到着地点までで算定されたオンデマンドバスの運賃
 - データの形式
 - メモリ上に格納
 - 利用するデータインターフェース
 - 【IF011】推定運賃 IF
 - 出力
 - ◇ Ondemand メッセージ
 - データの内容
 - オンデマンドバスの推定待ち時間と算出運賃を表す、GTFS-Ondemand が拡張したリアルタイムメッセージ
 - データの形式
 - GTFS-Ondemand が拡張した GTFS Realtime メッセージ形式
 - 利用するデータインターフェース
 - 【IF006】GTFS-Ondemand リアルタイムデータ API
- 【FN201】ドライバーアプリ
- 本システム機能の概要
 - 現在位置や乗客の乗車状況など、車両の情報を【FN101】車両管理へ連携する
 - 【FN101】車両管理から得た今後予定されている行き先や、車両の現在位置等をドライバーへ提示

する

- フローチャート

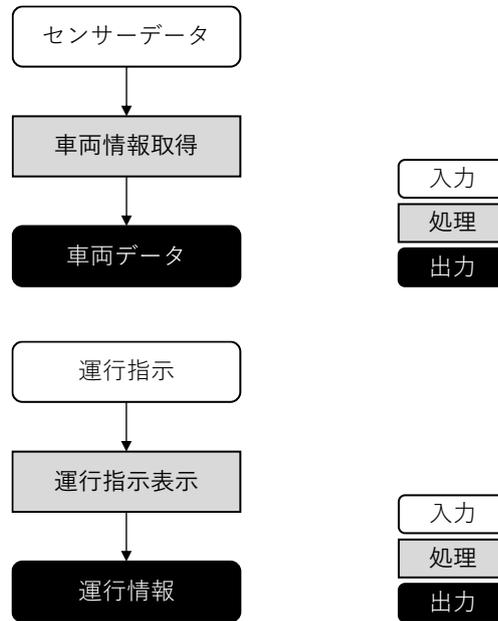


図 2-21 車両情報取得のフローチャート

- 本システム機能の処理の詳細

- 車両情報更新

- ◇ 処理内容

- 現在位置など車両の情報を車載タブレットのセンサーが取得し、乗客の乗車状況なども合わせて車両データとして【FN101】へ随時送信する

- ◇ 利用するライブラリ

- Android OS の位置情報サービス

- ◇ 利用するアルゴリズム

- なし

- 運行指示表示

- ◇ 処理内容

- 【FN101】から得た運行指示を基に、車載タブレットへ最新の運行ルートを表示する

- ◇ 利用するライブラリ

- なし

- ◇ 利用するアルゴリズム

- なし

- 本システム機能の入出力データの仕様

- 入力

- ◇ センサーデータ

- データの内容
 - 車載タブレットに搭載された GPS データ

- データの形式
 - 車載タブレットの OS が提供するデータ
 - ◇ Android OS の位置情報サービスを参照
- 利用するデータインターフェース
 - なし
- ◇ 運行指示
 - データの内容
 - 停車すべき乗降ポイントとその順序など、車両が運行すべき経路の指示
 - データの形式
 - リクエストパラメータ
 - 利用するデータインターフェース
 - 【IF013】 運行指示 API
- 出力
 - ◇ 車両データ
 - データの内容
 - 現在位置など車載タブレットのセンサーが取得した車両のリアルタイムデータ
 - データの形式
 - リクエストパラメータ
 - 利用するデータインターフェース
 - 【IF012】 車両情報更新 API
 - ◇ 運行情報
 - データの内容
 - 停車すべき乗降ポイントとその順序など、車両が運行すべき経路の指示
 - データの形式
 - 画面に表示
 - ◇ 【UI105】 ホーム画面（割当）を参照
 - 利用するデータインターフェース
 - なし

2-1-4. ソフトウェア・ライブラリ (SL) の詳細

表 2-4 ソフトウェア・ライブラリー一覧

※朱文字：新規開発・既存改修

ID	名称	バージョン	内容
SL001	駅探乗換案内	-	<ul style="list-style-type: none"> ● GTFS Schedule と GTFS-Flex、及び GTFS-Ondemand を活用してマルチモーダルな乗換探索を行い、指定された経路のオンデマンドバス配車予約を行うソフトウェア ● 駅探乗換案内の商用サービスを改修して実装
SL002	FastAPI	-	<ul style="list-style-type: none"> ● 駅探乗換案内にてマルチモーダル経路検索 API に使用するフレームワーク
SL003	gtfs-realtime-bindings	-	<ul style="list-style-type: none"> ● Protocol Buffers でバイナリ表現された GTFS Realtime (Ondemand) データの解析ライブラリ
SL004	protobuf	-	<ul style="list-style-type: none"> ● Protocol Buffers のメカニズムに基づいて構造化データをシリアライズ/デシリアライズするライブラリ
SL005	express	-	<ul style="list-style-type: none"> ● 駅探乗換案内にて利用者用 UI のサーバサイドフレームワーク
SL006	Vue.js	-	<ul style="list-style-type: none"> ● 駅探乗換案内にて利用者用 UI を構成するフロントエンドフレームワーク
SL007	PostgreSQL	-	<ul style="list-style-type: none"> ● 駅探乗換案内にて各種配信するデータを格納するオープンソースのリレーショナルデータベース
SL008	PostGIS	-	<ul style="list-style-type: none"> ● 駅探乗換案内にて PostgreSQL で位置情報を扱うことを可能とする拡張機能
SL101	SAVS	1.6.6	<ul style="list-style-type: none"> ● 駅探乗換案内と連携し、出発地点から目的地点までのオンデマンドバスの配車計算をリアルタイムで実行し、出発地点及び目的地点での待ち時間を推定する機能及び、出発地点から目的地点までオンデマンドバスの配車予約を行い、SAVS ドライバーアプリを経由してドライバーへ運行指示を出すソフトウェア ● 未来シェア SAVS を改修して実装
SL201	SAVS ドライバーアプリ	1.6.6	<ul style="list-style-type: none"> ● オンデマンドバスに据え付けられた車載タブレットにインストールされるソフトウェア ● 現在位置などオンデマンドバスの車両情報を SAVS へ送信すると共に、【SL101】から受信したオンデマンドバスの運行指示をドライバーへ伝える ● 未来シェア SAVS の商用アプリをそのまま利用

ソフトウェア・ライブラリの詳細を記す。なお、本業務において開発（新規・改修）を行うソフトウェア・ライブラリを**朱文字**で示す。

【SL001】 駅探乗換案内

- ベンダー
 - 株式会社 駅探
- 公式サイト
 - <https://ekitan.com/>
- 本ソフトウェア・ライブラリの概要
 - GTFS Schedule と GTFS-Flex、及び GTFS-Ondemand を活用してマルチモーダルな乗換探索を行い、指定された経路のオンデマンドバス配車予約を行うソフトウェア
 - 駅探 乗換案内の商用サービスを改修して実装
- 開発するシステムにおいて利用する機能の詳細
 - 公共交通データ取得
 - GTFS Schedule データ取得
 - 探索グラフ作成
 - 利用者登録・認証
 - 探索グラフ Flex データ更新
 - マルチモーダル経路探索
 - オンデマンドバス探索結果取得
 - オンデマンドバス配車予約
 - オンデマンドバス予約取得
 - オンデマンドバス予約キャンセル
 - GTFS-Flex 及び GTFS-Ondemand 静的データ取得
- イメージ

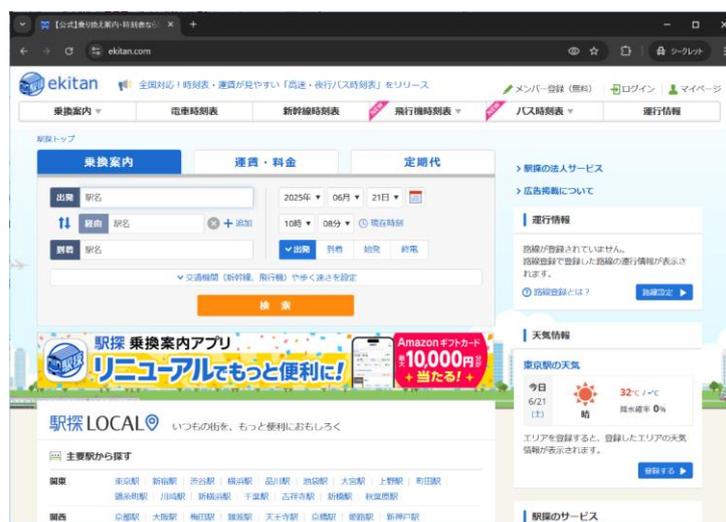
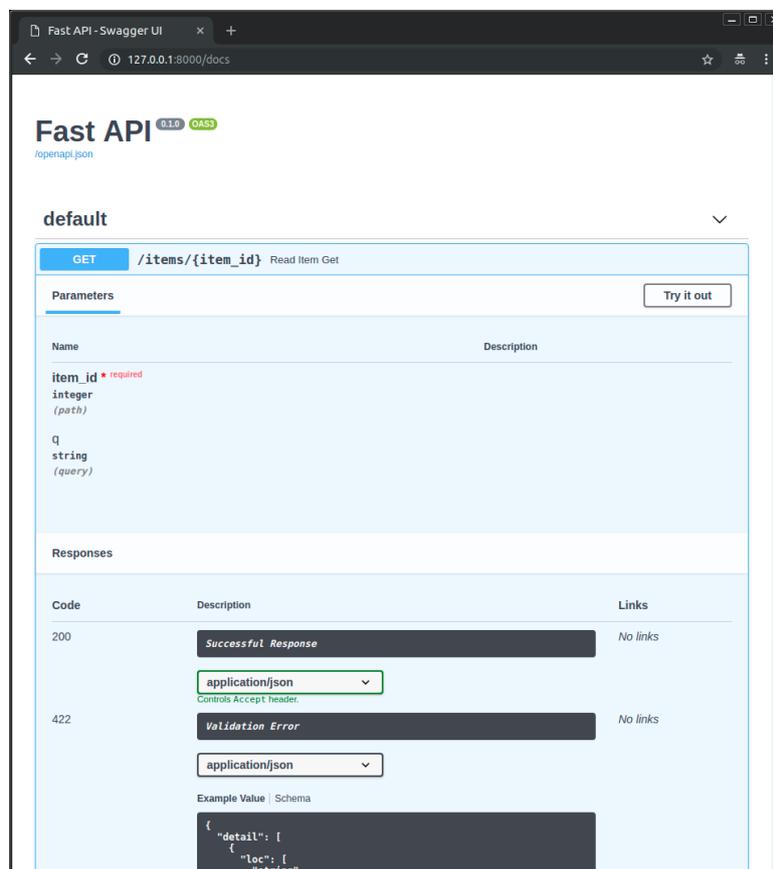


図 2-22 駅探乗換案内のイメージ

【SL002】 FastAPI

- ベンダー
 - Sebastián Ramírez 氏 / コミュニティ
- 公式サイト
 - <https://fastapi.tiangolo.com>
- 本ソフトウェア・ライブラリの概要
 - Python で API を開発するための Web フレームワーク
 - Python の型ヒントを活用し、読みやすくメンテナンスしやすいコードを記述できる
 - 高速でモダンな API 開発に特化しており、非同期処理に強みがある
- 開発するシステムにおいて利用する機能の詳細
 - 型アノテーションによるバリデーション：Python 型ヒントによるバリデーションと補完
 - 自動 API ドキュメント生成：OpenAPI (Swagger UI / ReDoc) を自動生成
- イメージ



☒ 2-23 FastAPI のイメージ

【SL003】 gtfs-realtime-bindings

- ベンダー
 - MobilityData
- 公式サイト
 - <https://github.com/MobilityData/gtfs-realtime-bindings>
- 本ソフトウェア・ライブラリの概要
 - GTFS Realtime データを扱うための Python,、 Java など言語別バインディングライブラリ
- 開発するシステムにおいて利用する機能の詳細
 - GTFS Realtime プロトコル対応：GTFS のリアルタイム仕様に対応
 - protobuf と統合：Protocol Buffers で定義された.proto ファイルからリアルタイムフィードを扱う
- イメージ

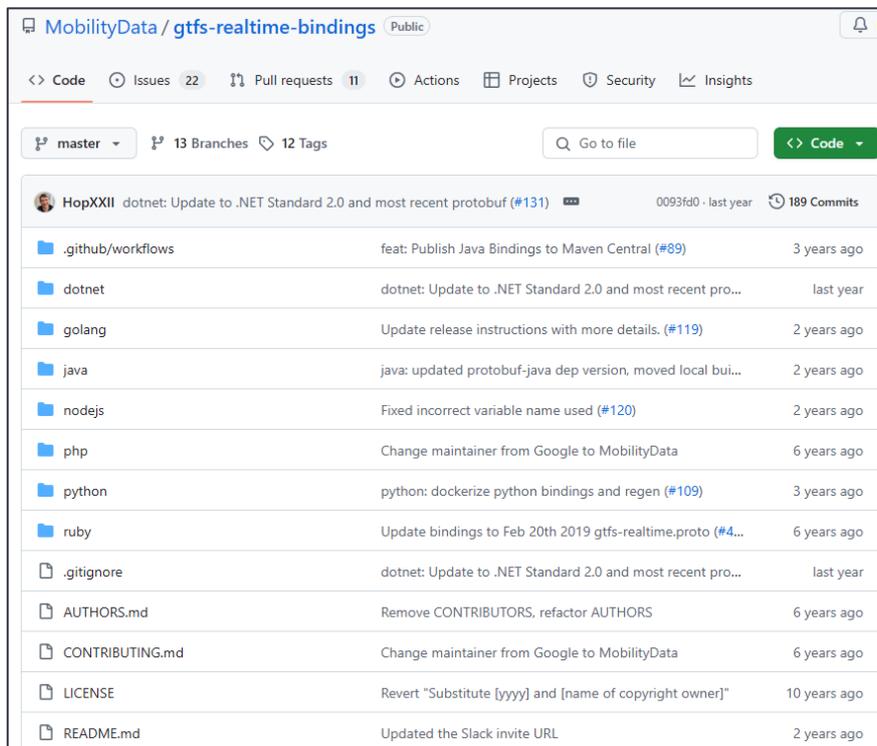


図 2-24 gtfs-realtime-bindings のイメージ

【SL004】protobuf

- ベンダー
 - Google
- 公式サイト
 - <https://protobuf.dev/>
- 本ソフトウェア・ライブラリの概要
 - 構造化データをシリアル化するためのプラットフォームに依存しない拡張可能なメカニズム
 - 様々なデータストリームに構造化データを簡単に読み書きできる
- 開発するシステムにおいて利用する機能の詳細
 - データの直列化／逆直列化：構造化データをバイナリフォーマットで軽量化
 - スキーマ定義：.proto ファイルでデータ構造を定義
 - 複数言語対応：Python、Ruby、Java、Goなどをサポート
- イメージ

```
import google3
from google.protobuf import descriptor as _descriptor
from google.protobuf import descriptor_pool as _descriptor_pool
from google.protobuf import runtime_version as _runtime_version
from google.protobuf import symbol_database as _symbol_database
from google.protobuf.internal import builder as _builder
_runtime_version.ValidateProtobufRuntimeVersion(
    _runtime_version.Domain.GOOGLE_INTERNAL,
    0,
    20240502,
    0,
    '',
    'main.proto'
)
# @@protoc_insertion_point(imports)

_sym_db = _symbol_database.Default()

DESCRIPTOR = _descriptor_pool.Default().AddSerializedFile(b'\n\nmain.proto\x12\x08tutor

_globals = globals()
_builder.BuildMessageAndEnumDescriptors(DESCRIPTOR, _globals)
_builder.BuildTopDescriptorsAndMessages(DESCRIPTOR, 'google3.main_pb2', _globals)
if not _descriptor._USE_C_DESCRIPTORS:
    DESCRIPTOR._loaded_options = None
    _globals['_PERSON']._serialized_start=25
    _globals['_PERSON']._serialized_end=316
    _globals['_PERSON_PHONENUMBER']._serialized_start=122
    _globals['_PERSON_PHONENUMBER']._serialized_end=210
    _globals['_PERSON_PHONETYPE']._serialized_start=212
    _globals['_PERSON_PHONETYPE']._serialized_end=316
    _globals['_ADDRESSBOOK']._serialized_start=318
    _globals['_ADDRESSBOOK']._serialized_end=365
# @@protoc_insertion_point(module_scope)
```

図 2-25 protobuf のイメージ

【SL005】 express

- ベンダー
 - StrongLoop/IBM
- 公式サイト
 - <https://expressjs.com/>
- 本ソフトウェア・ライブラリの概要
 - Node.js のためのシンプルかつ柔軟な Web アプリケーションフレームワーク
 - 数多くの HTTP ユーティリティメソッドとミドルウェアを利用できる
- 開発するシステムにおいて利用する機能の詳細
 - ルーティング定義：HTTP メソッドとパスに対する処理を簡潔に記述可能
 - ミドルウェア構成：リクエスト/レスポンス処理を段階的に制御
 - エラーハンドリング：エラー処理の一元化が可能
- イメージ

```
Respond with Hello World! on the homepage:

app.get('/', (req, res) => {
  res.send('Hello World!')
})

Respond to POST request on the root route (/), the application's home page:

app.post('/', (req, res) => {
  res.send('Got a POST request')
})

Respond to a PUT request to the /user route:

app.put('/user', (req, res) => {
  res.send('Got a PUT request at /user')
})

Respond to a DELETE request to the /user route:

app.delete('/user', (req, res) => {
  res.send('Got a DELETE request at /user')
})
```

図 2-26 express のイメージ

【SL006】 Vue.js

- ベンダー
 - Evan You 氏／コミュニティ
- 公式サイト
 - <https://vuejs.org>
- 本ソフトウェア・ライブラリの概要
 - Web アプリケーションのユーザーインターフェースを構築するための JavaScript フレームワーク
 - 学習が容易で、柔軟性が高く、高速なパフォーマンスが特徴
 - シングルページアプリケーション(SPA)やサイトの一部改修など、様々な規模のプロジェクトに対応
- 開発するシステムにおいて利用する機能の詳細
 - 双方向バインディング：v-model によりデータと UI を同期
 - リアクティブなデータ管理：データの変更が自動的に DOM に反映
 - コンポーネント指向：再利用可能な UI 部品の構築
- イメージ

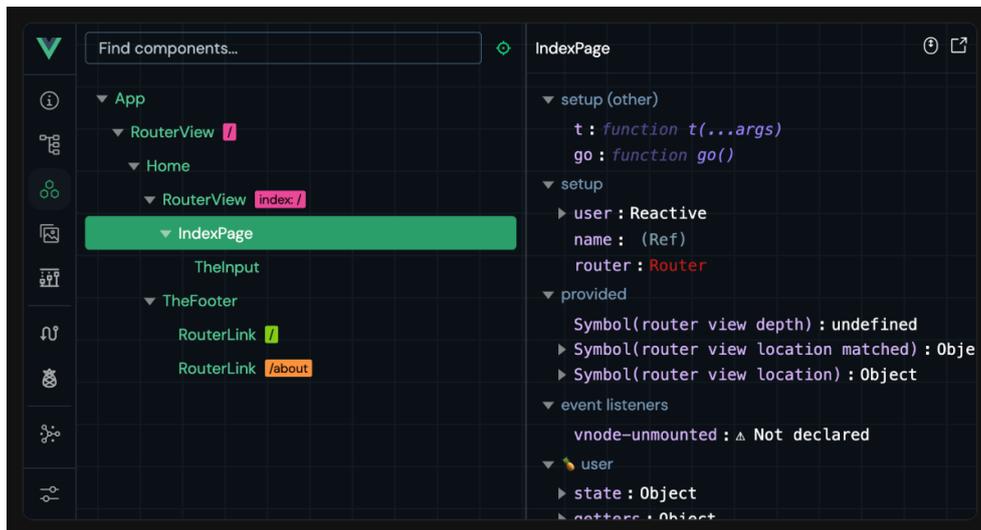


図 2-27 Vue.js のイメージ

【SL007】 PostgreSQL

- ベンダー
 - PostgreSQL Global Development Group
- 公式サイト
 - <https://www.postgresql.org>
- 本ソフトウェア・ライブラリの概要
 - オープンソースの高機能リレーショナルデータベース
- 開発するシステムにおいて利用する機能の詳細
 - リレーショナルデータベース：表形式でデータを管理
 - SQL 言語：SQL を使ってデータの操作や検索を行う
 - 拡張機能：ユーザー定義型や関数、拡張モジュールで機能を拡張
- イメージ

```
-- Create the table
CREATE TABLE employees (
  ID SERIAL PRIMARY KEY,
  Name VARCHAR(50),
  Age INT,
  Salary DECIMAL(10, 2),
  City VARCHAR(50),
  Country VARCHAR(50)
);

-- Insert data into the table
INSERT INTO employees (Name, Age, Salary, City, Country) VALUES
('Ravi', 21, 2000.00, 'Maryland', 'Germany'),
('Farhan', 30, 5000.00, 'New York', 'USA'),
('Teja', 28, 4500.00, 'Muscat', 'Dubai'),
('Yash', 27, 2500.00, 'Kolkata', 'India'),
('Ashin', 26, 3500.00, 'Bhopal', 'India');
```

図 2-28 PostgreSQL のイメージ

【SL008】 PostGIS

- ベンダー
 - PostGIS Project
- 公式サイト
 - <https://postgis.net>
- 本ソフトウェア・ライブラリの概要
 - PostgreSQL データベースに地理空間情報を格納し、操作するための地理空間拡張モジュール
 - PostgreSQL に空間データ型、空間関数、空間インデックスなどを追加することで、地理情報システム (GIS) を実現するためのツールとして利用できる
- 開発するシステムにおいて利用する機能の詳細
 - 空間インデックス：GiST インデックスにより高速な空間検索
 - 空間演算：距離、交差、包含などを SQL で記述可能
 - GeoJSON や KML 形式との相互変換：Web 地図との連携が容易
- イメージ

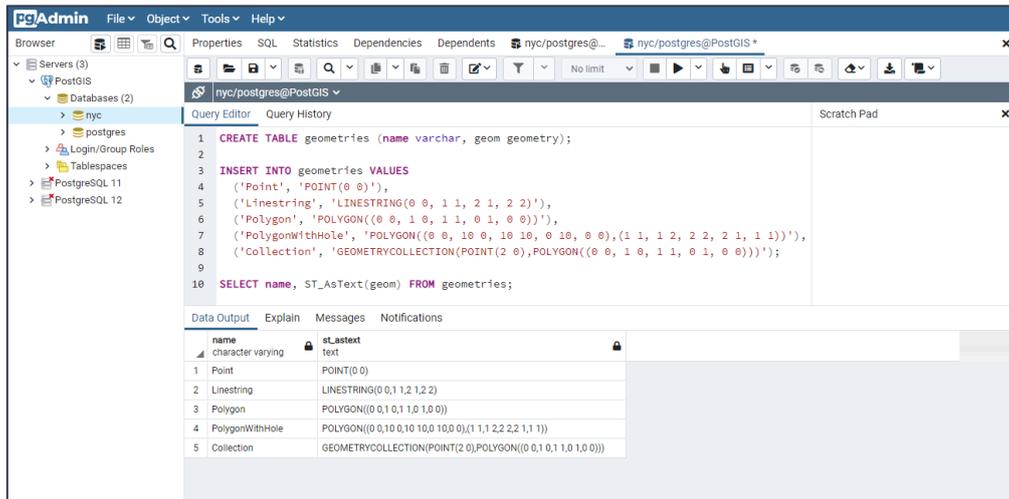


図 2-29 PostGIS のイメージ

【SL101】 SAVS

- ベンダー
 - 株式会社未来シェア
- 公式サイト
 - <https://www.miraishare.co.jp/>
- 本ソフトウェア・ライブラリの概要
 - 【SL001】 駅探乗換案内と連携し、出発地点から目的地点までのオンデマンドバスの配車計算をリアルタイムで実行し、出発地点及び目的地点での待ち時間を推定する
 - また出発地点から目的地点までオンデマンドバスの配車予約を行い、【SL201】を経由してドライバーへ運行指示を出す
 - 未来シェア SAVS の商用サービスを改修して実装

GTFS-Flex/Ondemand 活用システム システム設計書

- 開発するシステムにおいて利用する機能の詳細
 - 車両管理
 - 配車計算エンジン
 - 乗客管理
 - 予約管理
 - GTFS-Flex 及び GTFS-Ondemand 運賃ルール取得
 - オンデマンドバス運賃算定
 - Ondemand 動的メッセージ生成
- イメージ

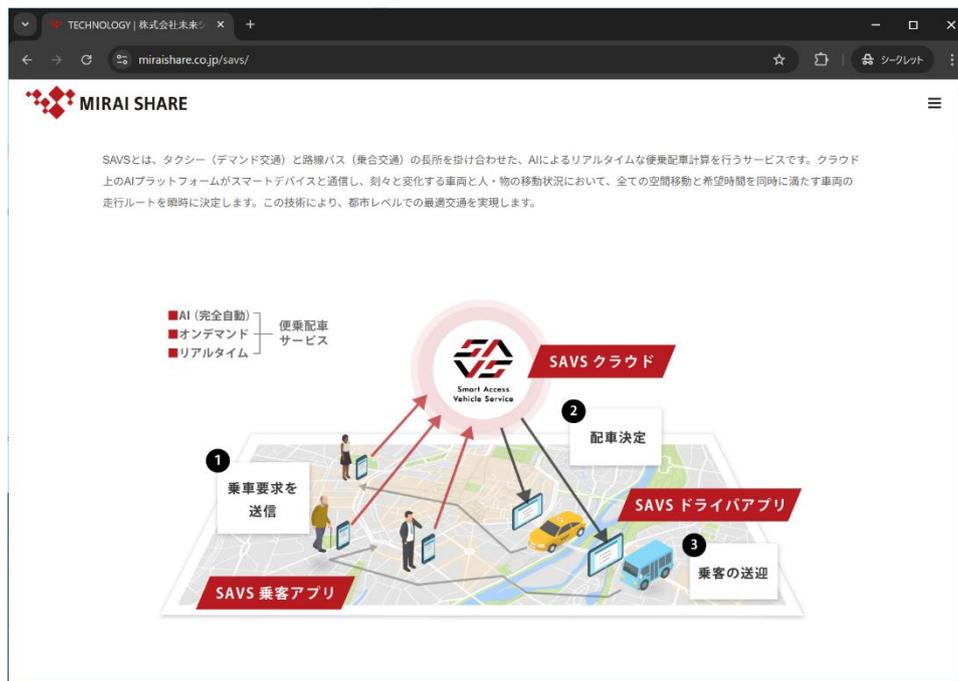


図 2-30 SAVS のイメージ

【SL201】 SAVS ドライバーアプリ

- ベンダー
 - 株式会社未来シェア
- 公式サイト
 - <https://www.miraishare.co.jp/>
- 本ソフトウェア・ライブラリの概要
 - オンデマンドバスに据え付けられた車載タブレットにインストールされるソフトウェア
 - 現在位置などオンデマンドバスの車両情報を【SL101】へ送信すると共に、【SL101】から受信したオンデマンドバスの運行指示をドライバーへ伝える
 - 未来シェア SAVS の商用アプリをそのまま利用
- 開発するシステムにおいて利用する機能の詳細
 - 車両情報取得

➤ 運行指示表示

● イメージ

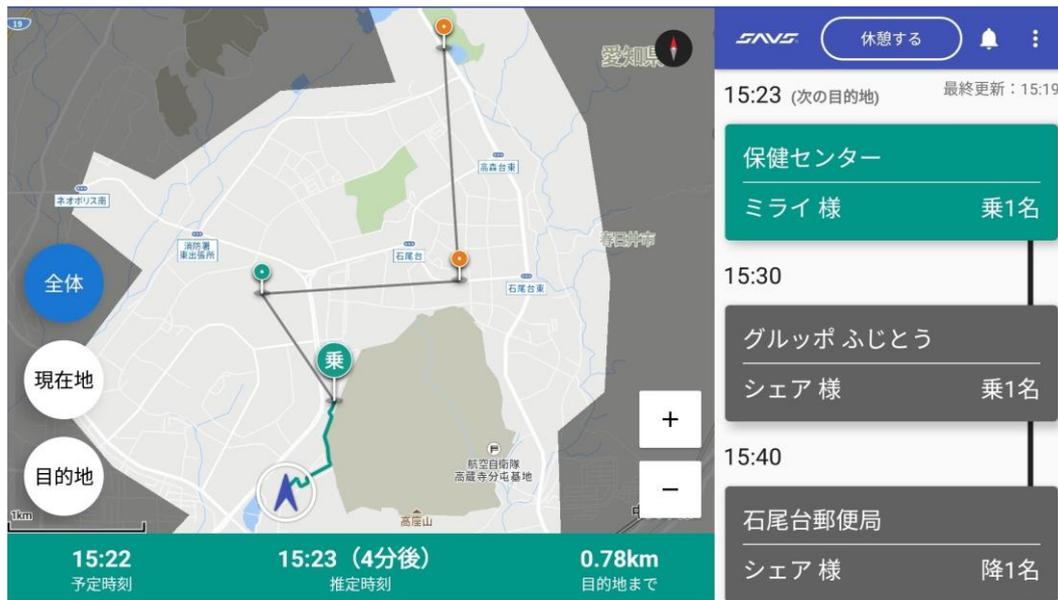


図 2-31 SAVS ドライバーアプリのイメージ

2-1-5. 数理モデル・アルゴリズム (AL) の詳細

表 2-5 数理モデル・アルゴリズム一覧

※朱文字：新規開発・既存改修

ID	名称	説明	アルゴリズムを利用した機能
AL001	GTFS-Flex 及び GTFS- Ondemand マルチモー ダル経路探 索アルゴリ ズム	● GTFS-Flex 及び GTFS-Ondemand を利用するマルチモーダル経路探索のアルゴリズム	FN006, FN007
AL002	Flex のみ マルチモー ダル経路探 索アルゴリ ズム	● GTFS-Flex のみを利用するマルチモーダル経路探索のアルゴリズム	FN006
AL003	オンデマン ドバス配車 予約アルゴ リズム	● オンデマンドバス配車予約サービスの API を活用したオンデマンドバスの配車予約を確定させるアルゴリズム	FN008

数理モデル・アルゴリズムの詳細を記す。なお、本業務において開発（新規・改修）を行う数理モデル・アルゴリズムを**朱文字**で示す。

【AL001】 GTFS-Flex 及び GTFS-Ondemand マルチモーダル経路探索アルゴリズム

- 本アルゴリズムの概要
 - GTFS-Flex 及び GTFS-Ondemand を利用するマルチモーダル経路探索のアルゴリズム
- 本アルゴリズムを利用した機能
 - **【FN006】** マルチモーダル経路探索
 - **【FN007】** オンデマンドバス待ち時間推定
- 本アルゴリズムの計算・処理の詳細
 - 本実証において開発するマルチモーダル経路探索アルゴリズムは、定時定路線交通の駅・バス停をノードとしたグラフ上での最短経路探索と、オンデマンドバス配車予約サブシステムから得るオンデマンドバスの所要時間推定を組み合わせることで実現する

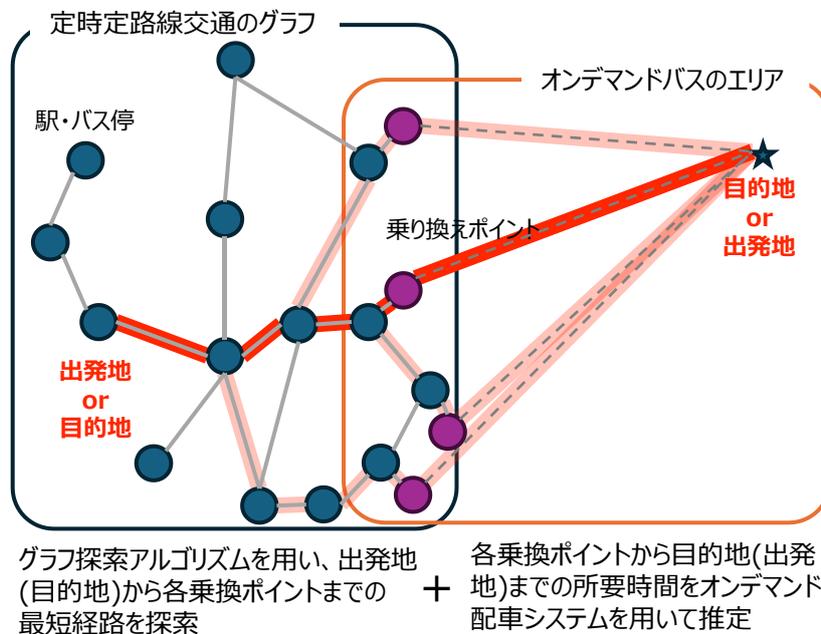


図 2-32 GTFS-Flex 及び GTFS-Ondemand マルチモーダル乗換探索のイメージ

- 定時定路線交通の探索グラフの作成
 - ◇ 定時定路線交通の最短経路の探索用に、以下のステップで連結グラフを作成する
 - 駅・バス停をノードとして定義する
 - 定時定路線交通の便や徒歩で乗り換えができるノード間を、移動手段ごとにエッジで接続する
 - 時刻表データや距離等を基に、移動所要時間を重みとして各エッジに設定する
- オンデマンドバスの乗換ポイントの設定
 - ◇ 定時定路線交通の探索グラフへ、オンデマンドバスとの乗換ポイントとなるノードを追加する
 - 1~20 ノード程度の乗換ポイントを設定し、ノードとして定義する

- 乗換ポイントから徒歩圏内にある近隣のノードから、徒歩移動を前提とした重みでエッジを設定する
- 定時定路線交通の最短経路の探索
 - ◇ 定時定路線交通とオンデマンドバス間を乗り換える場合、必ずいずれかの乗換ポイントを経由するルールを定めることで、オンデマンドバスの所要時間を推定する回数を乗換ポイント数以下に限定することができる
 - ◇ 定時定路線交通の探索グラフ上の出発地（目的地）と各乗換ポイント間の最短経路を、単一始点最短経路探索アルゴリズムである Dijkstra 法¹を用いてそれぞれ算出する
 - 優先度付きキューを利用することで Dijkstra 法の計算量は以下程度と見込まれ、実用的な速度で動作することが期待できる

$$\text{式 2-1 } O((V + E) \log(V)) \quad V: \text{ノード数}, E: \text{エッジ数}$$

- オンデマンドバスの所要時間の推定
 - ◇ オンデマンドバスの配車エリア上の目的地（出発地）と各乗換ポイント間の所要時間をそれぞれリアルタイムに算出する
 - オンデマンドバスの所要時間推定は、インターネット越しの API 呼び出しが必要となる重い処理だが、計算量は以下となり現実的な速度で動作すると見込まれる

$$\text{式 2-2 } O(n) \quad n: \text{乗換ポイント数}$$

- マルチモーダルな乗換経路全体の所要時間の推定
 - ◇ 乗換ポイントごとに定時定路線交通の所要時間とオンデマンドバスの推定所要時間を加算することで、当該乗換経路全体の所要時間を推定する
- オンデマンドバスの乗合による遅延の考慮
 - ◇ オンデマンドバスは、移動中にほかの乗客との乗合が発生し、目的地への到着時間が当初の推定よりも遅延する場合がある
 - ◇ ただし本実証で利用するオンデマンドバス配車予約サービスは、迂回許容時間を設定することで、最遅の到着時間を推定することができる²。GTFS-Ondemand のリアルタイムメッセージでは、*wait_time*（乗客がある地点で待つ時間）だけでなく *max_wait_time*（乗客がある地点で待つ最大待ち時間）も設定することができるため、これらの情報を活用して乗り換えに失敗しない適切な乗換経路を提案することができる

¹ Dijkstra, E. W. (1959). "A note on two problems in connexion with graphs". Numerische Mathematik. 1: 269–271.

² <https://www.miraishare.co.jp/savs/#spec>

```

header{
  timestamp: 1761458412
}
entity{
  on_demand{
    wait_time_update{
      wait_location: 出発地点
      trip_id: t005
      wait_time: 120
      ...
    }
    wait_time_update{
      wait_location: 到着地点
      trip_id: t005
      wait_time: 1260
      max_wait_time: 2160
      ...
    }
  }
}

```

図 2-33 GTFS-Ondemand を用いた推定待ち時間のリアルタイムメッセージのイメージ

- 定時定路線交通とオンデマンドバスの乗換パターンは次の 4 種類であるため、パターンごとに乗合が発生した場合のアルゴリズムを定める
 - パターン 1: 出発時間指定 定時定路線交通 → オンデマンドバス
 - ◇ 特に問題なくオンデマンドバスと乗り換えられるパターン
 - 定時定路線交通の出発駅・バス停から各乗換ポイントまでの経路をグラフ上で探索し、各乗換ポイントの到着時間を算出する
 - 乗換ポイントの到着時間を基に、各乗換ポイントから目的地までオンデマンドバスで移動する場合の所要時間を推定する
 - オンデマンドバスに乗合があった場合、目的地の到着時間が推定した最遅日時まで遅延する可能性がある。後続の乗り換えが無いので支障はないが、最遅の到着日時も合わせて利用者に提示する
 - ある乗換ポイントから目的地まで、配車可能なオンデマンドバスが存在しない場合、その乗換経路は候補から外す
 - 各乗換ポイント経由の乗換経路を集め、なるべく早く目的地に到着する順序で乗換経路を優先付ける

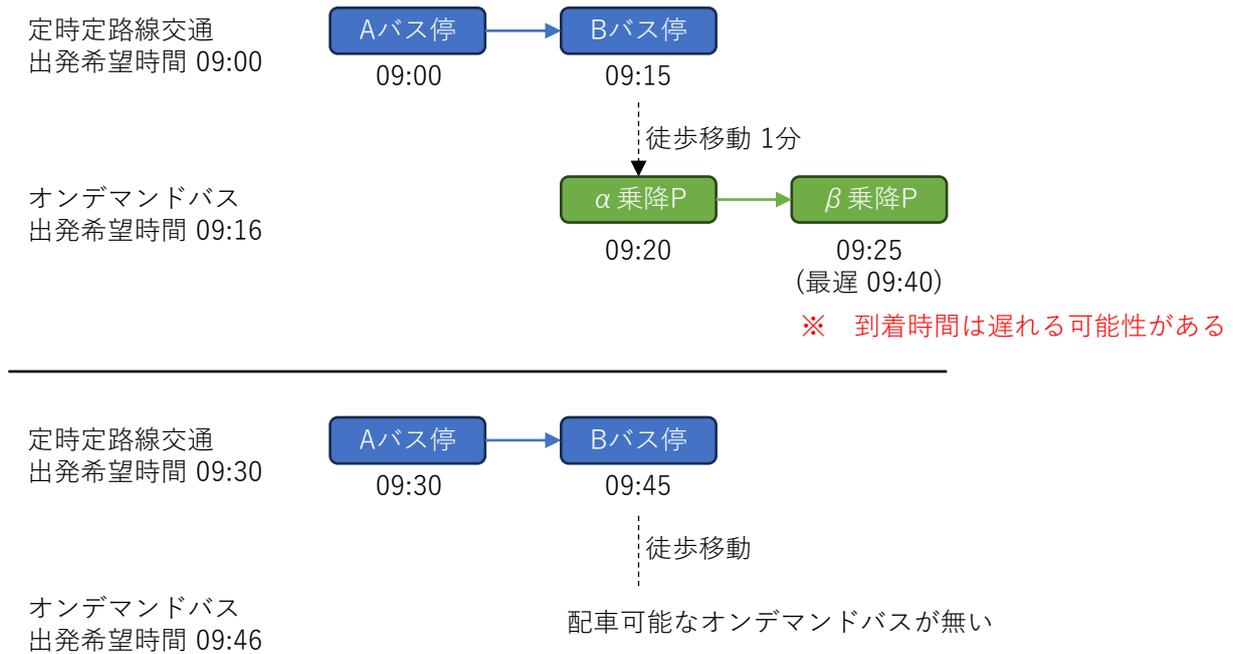
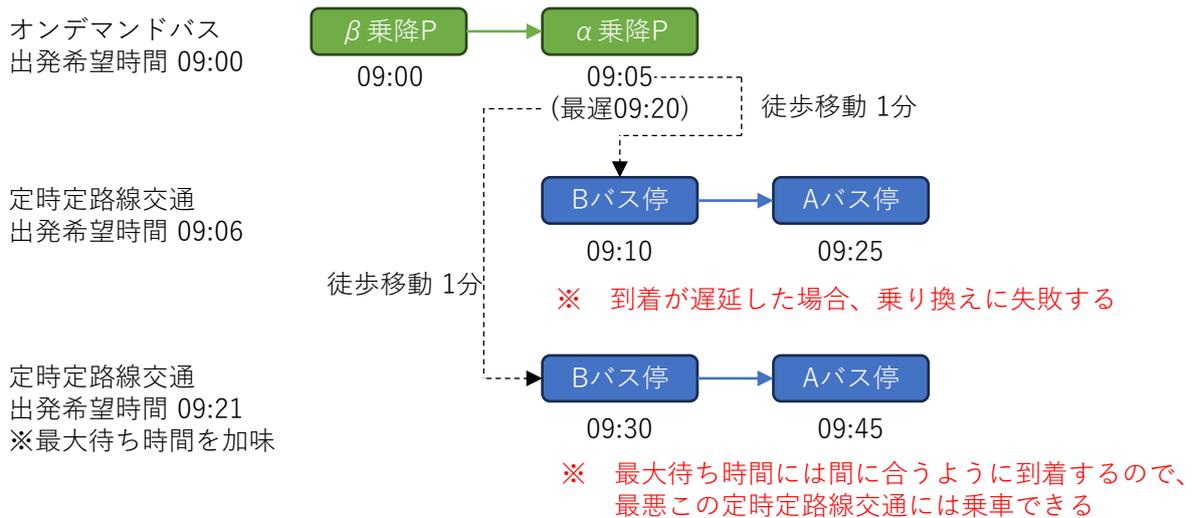


図 2-34 パターン 1 の乗換経路のイメージ

➤ パターン 2: 出発時間指定 オンデマンドバス → 定時定路線交通

◇ 乗合による遅延が影響するパターン

- オンデマンドバスの出発地点から各乗換ポイントまでオンデマンドバスで移動する場合の所要時間を推定する
- オンデマンドバスに乗合があった場合、各乗換ポイントへの到着時間は最遅推定日時まで遅延する可能性があるため、定時定路線交通の探索グラフ上で以下二つの経路を探索する。
 - 各乗換ポイントへのオンデマンドバスの推定到着日時を基に、各乗換ポイントから定時定路線交通の目的駅・バス停までの経路を探索する
 - 各乗換ポイントへのオンデマンドバスの最遅到着日時を基に、各乗換ポイントから定時定路線交通の目的駅・バス停までの経路を探索する
- 出発地点からある乗換ポイントまで配車可能なオンデマンドバスが存在しない場合、その乗換経路は候補から外す。さらに、ある乗換ポイントから目的駅・バス停まで、推定到着日時だけでなく最遅到着日時での定時定路線交通の経路が探索できなかった場合、その乗換経路も候補から外す
- 各乗換ポイント経由の乗換経路を集め、到着推定時間を基にした場合になるべく早く目的駅・バス停に到着する順序で乗換経路を優先付ける



オンデマンドバス
出発希望時間 09:30

配車可能なオンデマンドバスが無い

定時定路線交通 (探索不要)

図 2-35 パターン 2 の乗換経路のイメージ

- パターン 3: 到着時間指定 定時定路線交通 → オンデマンドバス
- ◇ 乗合が無かった場合、乗り換えの待ち時間が長くなるパターン
 - 到着時間からさかのぼり、オンデマンドバスの到着地点から各乗換ポイントまでオンデマンドバスで移動する場合の所要時間を推定する
この際オンデマンドバスが相乗りされる場合を考慮し、各乗換ポイントへの推定出発日時は乗合があった場合の迂回許容時間を見込んだ日時で推定される
 - 各乗換ポイントの推定出発時間を基に、各乗換ポイントから出発駅・バス停までの定時定路線交通の経路をさかのぼって探索する
 - 到着地点からある乗換ポイントまで、配車可能なオンデマンドバスが存在しない場合、その乗換経路は候補から外す
 - 各乗換ポイント経由の乗換経路を集め、なるべく遅く出発駅・バス停を出発する順序で乗換経路を優先付ける



オンデマンドバス
到着希望時間 18:30

配車可能なオンデマンドバスが無い

定時定路線交通 (探索不要)

図 2-36 パターン 3 の乗換経路のイメージ

- パターン 4: 到着時間指定 オンデマンドバス → 定時定路線交通
- ☆ 乗合が無かった場合、乗り換えの待ち時間が長くなるパターン
 - 定時定路線交通の到着駅・バス停から各乗換ポイントまでの経路をグラフ上でさかのぼって探索し、各乗換ポイントの出発時間を算出する
 - 乗換ポイントの出発時間を基に、各乗換ポイントから出発地までオンデマンドバスで移動する場合の所要時間を推定する
 - オンデマンドバスに乗合があった場合を考慮し、オンデマンドバスの推定出発日時は乗合があった場合の迂回許容時間を見込んだ日時で推定される
 - そのため出発地、あるいは乗換ポイントで多少の待ち時間が発生する可能性がある。
 - ある乗換ポイントから出発地まで、配車可能なオンデマンドバスが存在しない場合、その乗換経路は候補から外す
 - 各乗換ポイント経由の乗換経路を集め、なるべく遅く出発地を出発する順序で乗換経路を優先付ける

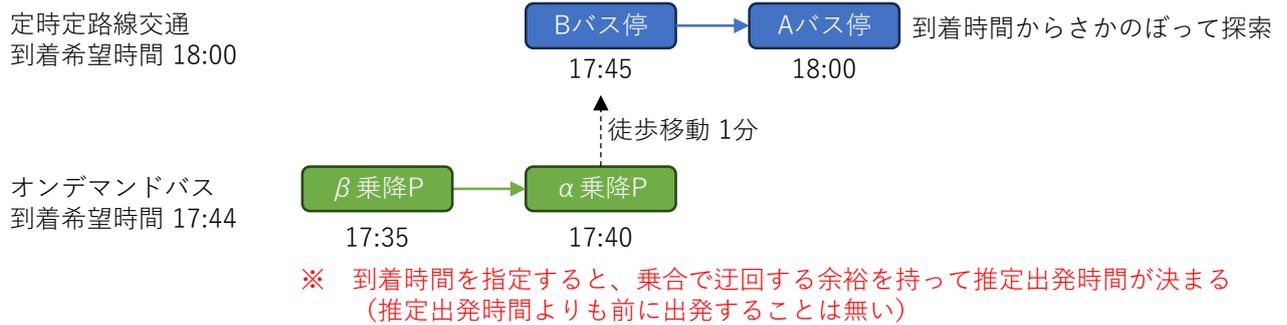


図 2-37 パターン 4 の乗換経路のイメージ

【AL002】 Flex のみマルチモーダル経路探索アルゴリズム

- 本アルゴリズムの概要
 - GTFIS-Flex のみを利用するマルチモーダル経路探索のアルゴリズム。
- 本アルゴリズムを利用した機能
 - 【FN006】 マルチモーダル経路探索
- 本アルゴリズムの計算・処理の詳細
 - 本実証において開発するマルチモーダル経路探索アルゴリズムは、定時定路線交通の駅・バス停をノードとしたグラフ上での最短経路探索と、GTFIS-Flex によるオンデマンドバスの静的な配車定義を組み合わせることで実現する

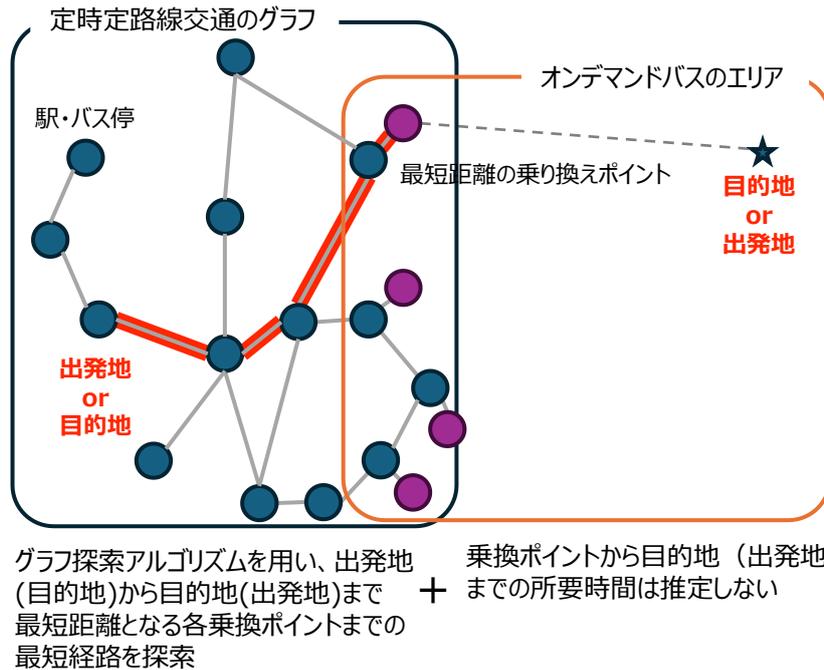
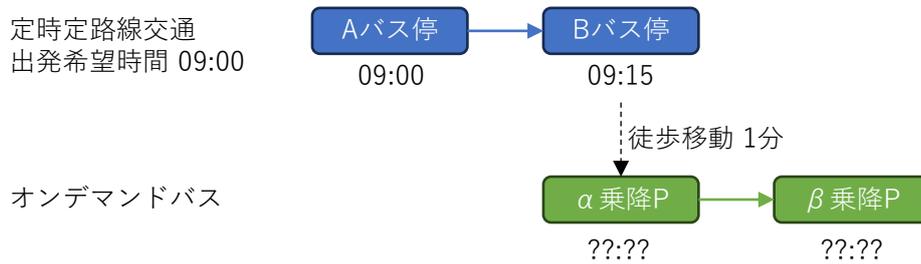


図 2-38 Flex のみマルチモーダル乗換探索のイメージ

- 定時定路線交通の探索グラフの作成
 - ◇ 【AL001】と同様
- オンデマンドバスの乗換ポイントの設定
 - ◇ 【AL001】と同様
- 定時定路線交通の最短経路の探索
 - ◇ GTFS-Flex のみを用いる場合、オンデマンドバスの配車エリア上の目的地 (出発地) と各乗換ポイント間の所要時間をリアルタイムに算出することができないため、目的地 (出発地) と直線距離 (ユークリッド距離) が最短の乗換ポイントからオンデマンドバスを利用するものとする
 - ◇ 定時定路線交通の探索グラフ上の出発地 (目的地) と、目的地 (出発地) から最短ユークリッド距離となる乗換ポイント間の最短経路を、【AL001】と同様に Dijkstra 法を用いて算出する
- 乗換パターンの考慮
 - ◇ Flex のみを用いる場合、オンデマンドバスの配車時間をリアルタイムに推定することができない。そのため、オンデマンドバスの目的地 (出発地) と乗換ポイント間の所要時間を推定した後に、その結果を用いて定時定路線交通の出発地 (目的地) と乗換ポイント間の最短経路を算出するステップを踏む経路探索は実行できない。よって【AL002】では以下の 2 パターンの乗換アルゴリズムを実装する
- パターン 1' : 出発時間指定 定時定路線交通 → オンデマンドバス
 - ◇ 定時定路線交通からオンデマンドバスに乗り換えられるが、到着時間がわからないパターン
 - 定時定路線交通の出発駅・バス停から目的地まで最短距離にある乗換ポイントまでの経路をグラフ上で探索し、その乗換ポイントの到着時間を算出する

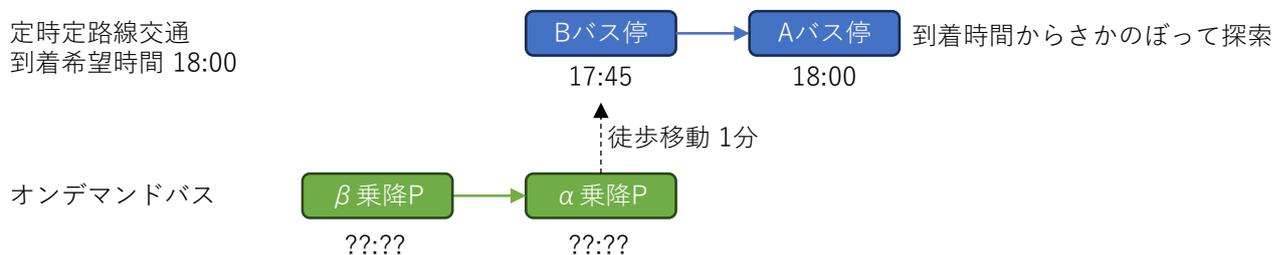
- 乗換ポイントから目的地までの所要時間はわからないため、利用者には推定到着時間を提示しない。また配車可能なオンデマンドバスの有無も確認しない



※ オンデマンドバスの出発推定時間、到着推定時間はわからない

図 2-39 パターン 1' の乗換経路のイメージ

- パターン 4' : 到着時間指定 オンデマンドバス → 定時定路線交通
 - ◇ オンデマンドバスから定時定路線交通に乗り換えられるが、出発時間がわからないパターン
 - 定時定路線交通の到着駅・バス停から出発地まで最短距離にある乗換ポイントまでの経路をグラフ上でさかのぼって探索し、その乗換ポイントの出発時間を算出する
 - 乗換ポイントから出発地までの所要時間はわからないため、利用者には推定出発時間を提示しない。また配車可能なオンデマンドバスの有無も確認しない



※ オンデマンドバスの出発推定時間、到着推定時間はわからない

図 2-40 パターン 4' の乗換経路のイメージ

【AL003】 オンデマンドバス配車予約アルゴリズム

- 本アルゴリズムの概要
 - オンデマンドバス配車予約サービスの API を活用したオンデマンドバスの配車予約を確定させるアルゴリズム
- 本アルゴリズムを利用した機能
 - **【FN008】** オンデマンドバス配車予約
- 本アルゴリズムの計算・処理の詳細
 - オンデマンドバスの配車予約処理は、既存予約の再計算を伴う重い処理であるため、**【SL001】** 駅探乗換案内からの API 接続を短時間で完結できるように、予約の仮登録処理は非同期で実行しなければならない

- 【AL001】で示したように、GTFS-Flex 及び GTFS-Ondemand でマルチモーダルな乗換経路を推定するステップでは、一回の乗換経路探索で複数のオンデマンドバスの配車待ち時間を推定し、一つを除いた残りの推定結果は全て捨てることになる
- 加えて利用者がオンデマンドバスの予約をせずに離脱した場合、全ての推定結果は捨てる必要があるが、利用者が離脱したことをシステムは直接検知することができないため、推定結果を捨てる判断がつくまである程度の時間は待つ必要がある
- そのため、乗換経路探索時点で推定した配車計画を全て予約して車両を拘束してしまうことは、オンデマンドバス配車予約サービスの効率を低下させるため、避ける必要がある
- これらの理由より、オンデマンドバスの配車予約の確定までは、以下のアルゴリズムで処理を行う

※ 経路検索時の推定結果は、実際の配車予約とは無関係

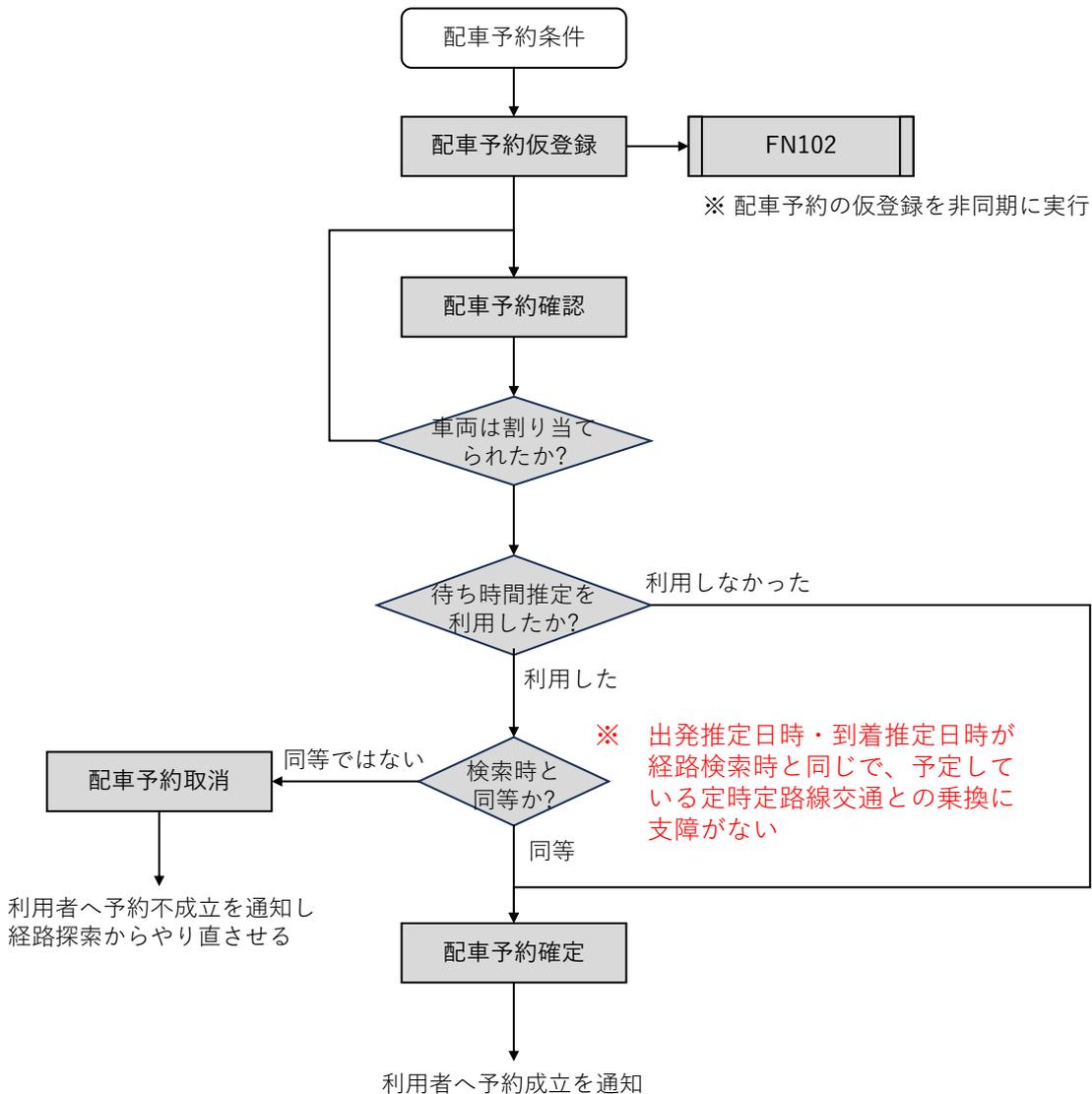


図 2-41 オンデマンドバス配車予約確定/取消のフローチャート

- 【FN008】オンデマンドバス配車予約がオンデマンドバスの配車予約を行うと、【FN102】配車計算エンジンが非同期で呼び出され、配車予約条件に合致した車両を配車可能か計算

する

- 配車予約条件に合致する配車可能な車両があった場合、車両が割り当てられ、配車予約が仮登録される
- 【FN008】 オンデマンドバス配車予約は仮登録が成立するまで待機し、車両が割り当てられた配車予約の出発推定日時や到着推定日時、最遅到着日時等を得る
- GTFS-Ondemand のリアルタイムデータによる推定待ち時間を用いずに経路探索を行っていた場合（Flexのみマルチモーダル経路探索の実行時）は、この時点で配車予約を確定し、利用者へ予約成立を通知する
- GTFS-Ondemand のリアルタイムデータを用いた推定待ち時間を考慮して経路探索を行っていた場合（GTFS-Flex 及び GTFS-Ondemand マルチモーダル経路探索の実行時）は、仮予約された出発推定日時や到着推定日時、最遅到着日時と経路探索時の推定待ち時間を比較し、経路探索時と同等の配車予約ができていることを確認する
 - 出発推定日時、到着推定日時、最遅到着日時が、マルチモーダル乗換探索時に予定していた定時定路線交通との乗換で支障を起こさない
 - 仮予約の出発推定日時、到着推定日時、最遅到着日時と、マルチモーダル乗換探索時に得たそれぞれの日時との差分が、事前に定めた閾値以内に収まっている
 - ◇ 要件定義時点では、閾値を 60 秒とする
 - 結合テストとシステムテストを通じて、適切な閾値にチューニングする
 - ただしオンデマンドバスから定時定路線交通に乗り換える前提で、現在日時を出発希望時間として乗換探索を行った場合、配車可能なオンデマンドバスが偶然近くにいたならば、オンデマンドバスの出発日時が乗換探索の操作を行った直後に推定される可能性がある。この際、配車予約の操作が完了するまでに時間を要すると、実際に仮予約された出発推定日時と乗換探索時の出発推定日時に閾値以上の差分が出る可能性がある。このような状況では、後続の定時定路線交通との乗換で支障を起こさないのであれば、出発推定日時の差分が閾値を超えていても同等の配車予約が成立したとみなす
- GTFS-Flex 及び GTFS-Ondemand マルチモーダル経路探索を実行しており、経路探索時と同等の配車予約ができなかった場合、配車予約を取り消し、利用者へ予約不成立を通知して経路探索から再実行を促す。経路探索時と同等の配車予約ができた場合、配車予約を確定し、利用者へ予約成立を通知する

2-2. システムコンポーネント (CO)

2-2-1. システムコンポーネント図

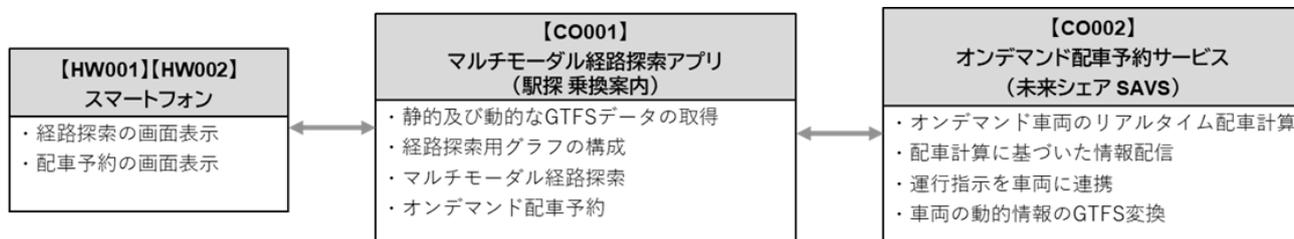


図 2-42 システムコンポーネント図

2-2-2. システムコンポーネント一覧

表 2-6 システムコンポーネント一覧

ID	種別	コンポーネント名	用途
CO001	クラウド (SaaS)	駅探 マルチモーダル経路検索アプリ	<ul style="list-style-type: none"> ● 利用者が利用するUI (Web アプリ) にて、出発地、目的地、出発又は到着時間等の探索条件を入力させることで、その条件に合致したマルチモーダルな乗換経路の候補を表示する
CO002	クラウド (SaaS)	未来シェア オンデマンド配車予約サービス	<ul style="list-style-type: none"> ● 利用者からの配車予約と車両の現在位置、空き席数、走行予定経路等を基に、リアルタイムに最適な配車計算を行う

2-3. ハードウェア (HW)

2-3-1. ハードウェアアーキテクチャ

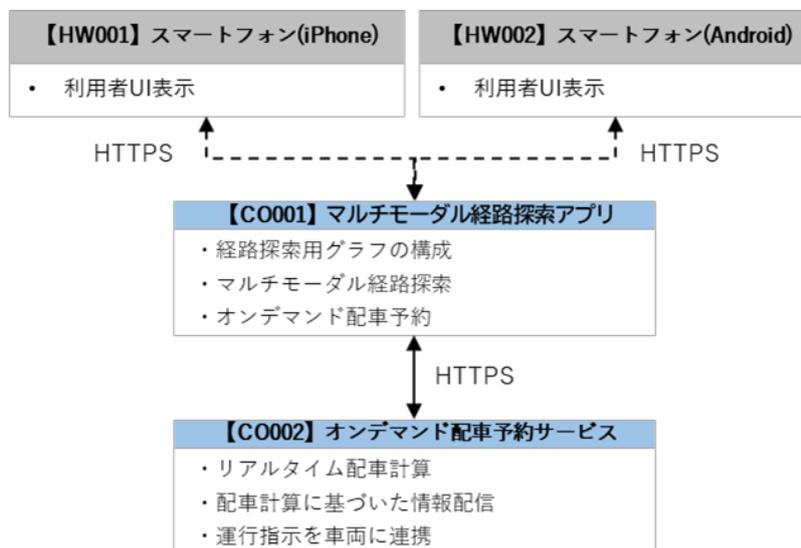


図 2-43 ハードウェアアーキテクチャ

2-3-2. ハードウェア一覧

表 2-7 ハードウェア一覧

※朱文字：新規開発・既存改修

ID	種別	ベンダー	品番	用途
HW001	スマートフォン (iPhone)	Apple	-	<ul style="list-style-type: none"> ● 実証の被験者が利用者用 UI (Web アプリ) を使用するために用いる ● iOS18 以降
HW002	スマートフォン (Android)	Google 等	-	<ul style="list-style-type: none"> ● 実証の被験者が利用者用 UI (Web アプリ) を使用するために用いる ● Android14 以降

2-3-3. ハードウェアの詳細

ハードウェアの詳細を記す。なお、本業務において開発（新規・改修）を行うハードウェアを**朱文字**で示す

【HW001】スマートフォン (iPhone)

- 本ハードウェアの概要
 - 実証に参加する被験者が所持する iOS18 以降の iPhone
 - ブラウザからマルチモーダル乗換探索アプリの WebUI を使用する
- 本ハードウェアを提供するベンダー
 - Apple
- 本ハードウェアの仕様・スペック (iPhone16 の場合)
 - SoC : Apple A18
 - OS : iOS18
 - メモリ : 非公開
 - ストレージ : 128GB/256GB/512GB
 - ディスプレイ : 6.1 インチ 2,556 x 1,179 ピクセル
 - 通信方式 : 国内キャリアの 4G、5G に対応
- イメージ



図 2-44 iPhone の例 (iPhone 16) ³

³ 公式 HP より抜粋 : <https://www.apple.com/jp/iphone-16/>

【HW002】スマートフォン(Android)

- 本ハードウェアの概要
 - 実証に参加する被験者が所持する Android14 以降の Android スマートフォン
 - ブラウザからマルチモーダル乗換探索アプリの WebUI を使用する
- 本ハードウェアを提供するベンダー
 - Google、サムスン、シャープ等
- 本ハードウェアの仕様・スペック（Google Pixel9 の場合）
 - SoC：Google Tensor G4
 - OS：Android14
 - メモリ：12GB
 - ストレージ：128GB/256GB
 - ディスプレイ：6.3 インチ 2,424 x 1,080 ピクセル
 - 通信方式：国内キャリアの 4G、5G に対応
- イメージ



図 2-45 Android スマートフォンの例（Google Pixel9）⁴

⁴ 公式 HP より抜粋：https://store.google.com/jp/product/pixel_9?hl=ja

2-4. データインターフェース (IF)

2-4-1. データアーキテクチャ

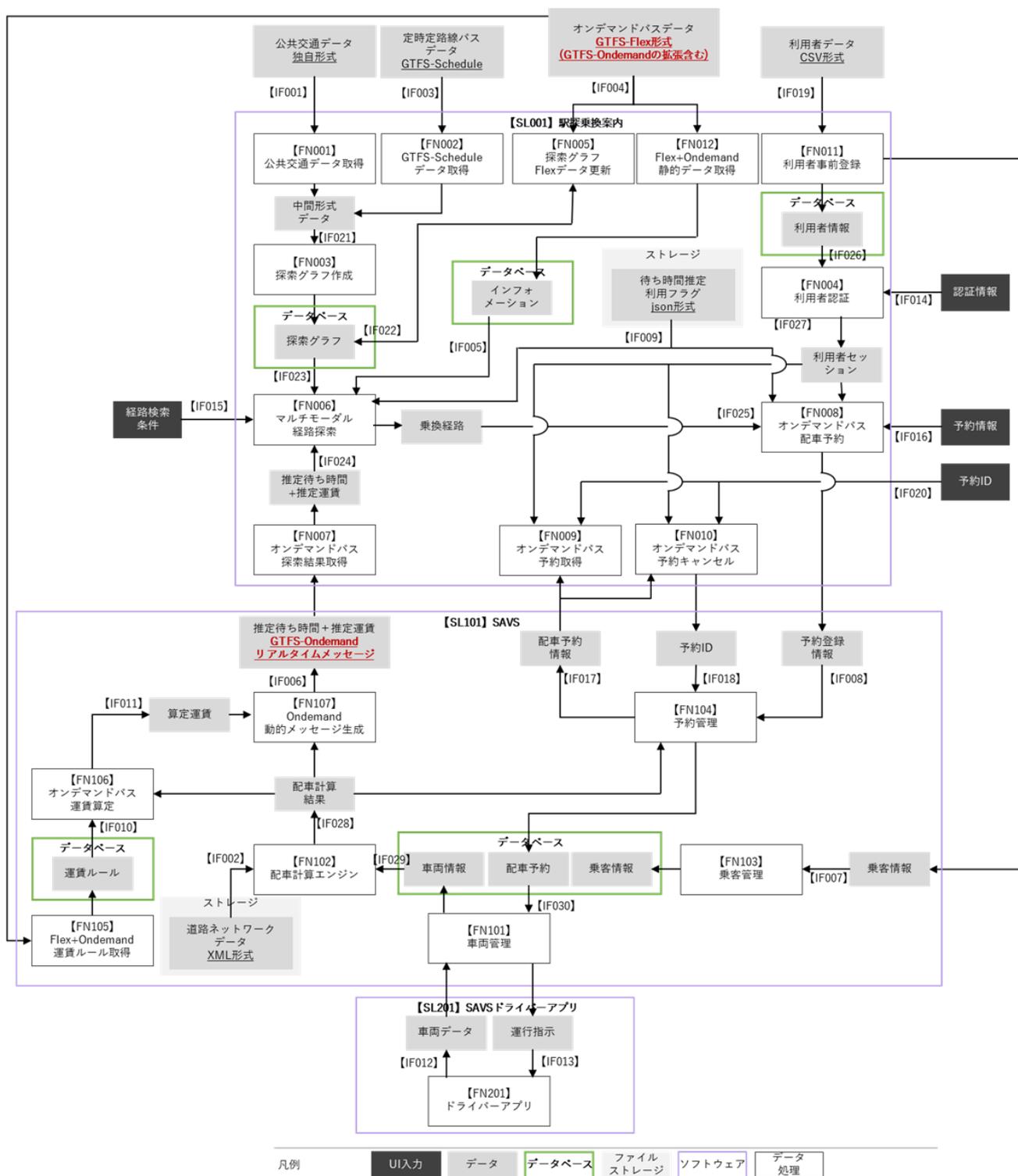


図 2-46 データアーキテクチャ

2-4-2. データインターフェース一覧

表 2-8 データインターフェース一覧

※朱文字：新規開発・既存改修

ID	名称	出力側 ID	入力側 ID
IF001	公共交通データファイル IF	-	FN001
IF002	道路ネットワークデータ	-	FN102
IF003	GTFS Schedule 定時定路線バスデータファイル IF	-	FN002
IF004	GTFS-Flex 及び GTFS-Ondemand オンデマンドバスデータ ファイル IF	-	FN005 FN012 FN105
IF005	GTFS-Flex 及び GTFS-Ondemand インフォメーション IF	FN012	FN006
IF006	GTFS-Ondemand リアルタイムデータ API	FN107	FN007
IF007	乗客登録 API	FN011	FN103
IF008	配車予約登録 API	FN008	FN104
IF009	待ち時間推定利用フラグ IF	-	FN006 FN008
IF010	GTFS-Flex 及び GTFS-Ondemand 運賃ルール IF	FN105	FN106
IF011	算定運賃 IF	FN106	FN107
IF012	車両情報更新 API	FN201	FN101
IF013	運行指示 API	FN101	FN201
IF014	利用者認証情報入力 IF	-	FN004
IF015	マルチモーダル経路検索条件入力 IF	-	FN006
IF016	予約情報入力 IF	-	FN008
IF017	配車予約取得 API	FN104	FN009 FN010
IF018	配車キャンセル API	FN010	FN104
IF019	事前登録用利用者データファイル IF	-	FN011
IF020	予約 ID 入力 IF	-	FN009 FN010
IF021	定時定路線交通中間形式データ IF	FN001 FN002	FN003
IF022	探索グラフ更新 IF	FN005	FN005
IF023	定時定路線乗換探索 IF	FN003	FN006
IF024	オンデマンドバス探索 IF	FN007	FN006
IF025	乗換経路探索結果 IF	FN006	FN008
IF026	利用者情報 IF	FN011	FN004

GTFS-Flex/Ondemand 活用システム システム設計書

IF027	利用者セッション IF	FN004	FN008 FN009 FN010
IF028	リアルタイム配車計算結果 IF	FN102	FN104 FN106 FN107
IF029	車両情報 IF	FN101	FN102
IF030	運行指示 IF	FN104	FN101

2-4-3. データインターフェースの詳細

データインターフェースの詳細を記す。なお、本業務において開発（新規・改修）を行うデータインターフェースを**朱文字**で示す。

【IF001】公共交通データファイル IF

- 本インターフェースの概要
 - 交通新聞社や JTB パブリッシング社から入手する JR 北海道や札幌市営地下鉄、札幌市電の時刻表データ
- 本インターフェースを利用する機能
 - **【FN001】公共交通データ取得**
- 本インターフェースを利用してやり取りを行うデータの詳細
 - 事業者の独自形式ファイル

【IF002】道路ネットワークデータ

- 本インターフェースの概要
 - 乗合配車による時間の遅れや迂回時間等を考慮した出発推定日時、到着推定日時を計算するために用いる道路ネットワークのデータ
- 本インターフェースを利用する機能
 - **【FN102】配車計算エンジン**
- 本インターフェースを利用してやり取りを行うデータの詳細
 - OpenStreetMap による道路ネットワークのデータ
 - ◇ ノード (node)
 - 緯度と経度、ID を持ち、地上にある特定の地点を意味する
 - 参照: <https://wiki.openstreetmap.org/wiki/Node>
 - ◇ ウェイ (way)
 - 2 から 2000 までのノードを順序付きで並べたもの
 - 道路や河川、鉄道路線を表したり、始点と終点のノードが同一の閉じたウェイとしてエリアを表したりすることができる
 - ウェイにはキーと値から構成されるタグを複数定義することができ、highway タグを付けられたウェイは道路を意味する
 - highway 以外にも、name (道路名)、maxspeed (最高速度制限)、lanes (車線数)、length や width (道路の長さ) と幅)、oneway (一方通行か?)、bridge (橋か?)、tunnel (トンネルか?) 等のタグが良く用いられる
 - 参照: <https://wiki.openstreetmap.org/wiki/Way>

【IF003】GTFS Schedule 定時定路線バスデータファイル IF

- 本インターフェースの概要

- 実証実験エリアを運行する定時定路線バスの時刻表データ
- GTFS-JP 第3版に準拠
- 本インターフェースを利用する機能
 - 【FN002】 GTFS Schedule データ取得
- 本インターフェースを利用してやり取りを行うデータの詳細
 - GTFS-JP 第3版のリファレンスを参照
<https://www.mlit.go.jp/sogoseisaku/transport/content/001419163.pdf>

【IF004】 GTFS-Flex 及び GTFS-Ondemand オンデマンドバスデータファイル IF

- 本インターフェースの概要
 - 実証実験エリアを運行するオンデマンドバスのサービスエリアやサービス時間、オンデマンドバスの乗降スポット、運賃ルール等を定義するデータ
- 本インターフェースを利用する機能
 - 【FN005】 探索グラフ Flex データ更新
 - 【FN012】 GTFS-Flex 及び GTFS-Ondemand 静的データ取得
 - 【FN105】 GTFS-Flex 及び GTFS-Ondemand 運賃ルール取得
- 本インターフェースを利用してやり取りを行うデータの詳細
 - 2024年3月に採用された GTFS-Flex 仕様、及び 2025年4月時点での GTFS-Ondemand 仕様案に準拠
 - データモデルと利用する値の詳細は、要件定義参考資料_GTFS-Flex 及び GTFS-Ondemand の拡張項目の一覧と画面表示対応を参照
 - ◇ 2024年3月に採用された GTFS-Flex 仕様
<https://github.com/google/transit/pull/388>
 - ◇ 2025年4月時点での GTFS-Ondemand 仕様案
<https://share.Mobilitydata.org/gtfs-ondemand>

【IF005】 GTFS-Flex 及び GTFS-Ondemand インフォメーション IF

- 本インターフェースの概要
 - GTFS-Flex 及び GTFS-Ondemand の静的データから得た利用者へ提示する予約関連情報を表す IF
- 本インターフェースを利用する機能
 - 【FN006】 マルチモーダル経路探索
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-9 GTFS-Flex 及び GTFS-Ondemand インフォメーション IF

名称	説明	型	必須
trips.trip_type	乗り合いされるか(一緒に旅行を調整しなかった多くの乗客がおり、出発地と目的地が異なる可能性がある)、又はプライベートか (1人の乗客とおそらくその同伴者のみが車両に乗り、出発地と	整数	

	目的地は同一) を定義する。 0 又は空 - 乗り合い / 1 - プライベート		
trips.luggage_allowed	乗客が荷物を持って車両に乗車できるかどうかを示す。 0 又は空 - 情報が利用できない / 1 - 荷物を持って乗車可 / 2 - 荷物を持って乗車不可	整数	
trips.pets_allowed	乗客がペットと一緒に車両に乗車できるかどうかを示す。 0 又は空 - 情報が利用できない / 1 - ペットを連れて乗車可 / 2 - ペットを連れて乗車不可 3 - ペットを連れて乗車できるが、キャリーバッグに入れる必要がある	整数	
trips.service_animal_allowed	乗客が介助動物と一緒に車両に乗車できるかどうかを示す 0 又は空 - 情報が利用できない / 1 - 介助動物を連れて乗車可 / 2 - 介助動物を連れて乗車不可 / 3 - 介助動物を連れて乗車できるが、キャリーバッグに入れる必要がある	整数	
trips.priced_itinerary	有料の旅程（高速料金やフェリー料金等）を使用できるかどうかを定義する 0 又は空 - 情報が利用できない / 1 - 有料の旅程を使用する場合がある / 2 - 有料の旅程は使用しない	整数	
stop_times. start_pickup_drop_off_window	オンデマンドサービスが利用可能になる時間	文字列	
stop_times. end_pickup_drop_off_window	オンデマンドサービスの利用が終了になる時間	文字列	
stop_times. pickup_proximity_level	リクエスト場所から乗車場所までの近接レベルを定義する 0 又は空 - 情報が利用できない / 1 - リクエスト場所にできるだけ近い場所で乗車 2 - リクエスト場所から少しの距離の場所（近い交差点等）で乗車する場合がある	整数	
stop_times. drop_off_proximity_level	リクエスト場所から降車場所までの近接レベルを定義する 0 又は空 - 情報が利用できない / 1 - リクエスト場所にできるだけ近い場所で降車 2 - リクエスト場所から少しの距離の場所（近い	整数	

	交差点等) で降車する場合がある		
booking_rules.booking_type	どの程度前から予約できるかを指定 0:リアルタイム予約 1:事前通知による当日予約まで 2:前日予約まで	整数	○
booking_rules.prior_notice_duration_min	booking_type=1 の場合に最小で何分前までに予約が必要か	整数	
booking_rules.prior_notice_duration_max	booking_type=1 の場合に最大で何分前から予約ができるか	整数	
booking_rules.prior_notice_last_day	booking_type=2 の場合に何日前まで予約ができるか	整数	
booking_rules.prior_notice_last_time	booking_type=2 の場合に、prior_notice_last_day の何時まで予約ができるか	文字列	
booking_rules.prior_notice_start_day	何日前から予約ができるか booking_type=0 の場合は禁止 booking_type=1 で prior_notice_duration_max が定義されている場合は禁止	整数	
booking_rules.prior_notice_start_time	prior_notice_start_day の何時から予約ができるか	文字列	
booking_rules.prior_notice_service_id	prior_notice_last_day 又は prior_notice_start_day をカウントするサービス日を示す booking_type=0 や 1 では禁止	文字列	
booking_rules.message	オンデマンドサービスを予約するときに示されるサービスを利用する乗客へのメッセージ	文字列	
booking_rules.phone_number	予約するために電話をかける電話番号	文字列	
booking_rules.info_url	予約ルールに関する情報を提供する URL	文字列	
booking_rules.booking_url	予約リクエストを行うことができるインターネット予約システムの URL	文字列	
booking_rules.download_app_url	予約リクエストを行うことができるアプリへの URL (Web サイトやアプリマーケット)	文字列	
wait_rules.start_time	待機ルールの適用が開始される時間を示す	文字列	
wait_rules.end_time	待機ルールの適用が終了される時間を示す	文字列	
wait_rules.mean_wait_time	stop_time で乗客が待つ平均待ち時間 (分)	整数	
wait_rules.safe_wait_time	stop_time で乗客が待つ安全な待ち時間 (分)。安全とは、過去のデータに基づいた 95 パーセントイル値の待ち時間を意味する	整数	
wait_rules.max_wait_time	stop_time で乗客が待つ最大待ち時間 (分)	整数	
booking_deep_link.	Android のディープリンクをサポートするために	文字列	

android_uri	Android の Intent と共に Android アプリに渡される URI。Android App Links が推奨される		
booking_deep_link. ios_uri	iOS で利用されるディープリンクの URI。iOS Universal Links が推奨される	文字列	
booking_deep_link. web_url	ウェブブラウザで利用される URL	文字列	

【IF006】 GTFS-Ondemand リアルタイムデータ API

- 本インターフェースの概要
 - 乗車地点と降車地点の緯度経度、出発希望日時あるいは到着希望日時等を指定して呼び出すことで、配車可能なオンデマンドバスが存在するかを判定し、存在する場合は乗車地点を出発する推定日時、降車地点に到着する推定日時と最遅の推定日時、及び算定した運賃を GTFS-Ondemand が拡張した GTFS Realtime 形式で返す API
- 本インターフェースを利用する機能
 - **【FN007】** オンデマンドバス待ち時間推定
 - **【FN107】** Ondemand 動的メッセージ生成
- 仕様
 - プロトコル
 - ◇ HTTPS
 - メソッド
 - ◇ POST
 - リクエストデータ形式
 - ◇ application/json
 - レスポンスデータ形式
 - ◇ application/x-protobuf (protocol buffers のバイナリ形式)
 - 文字コード
 - ◇ UTF-8
 - URI
 - ◇ <http://example.co.jp/demand-estimation-gtfs>
 - 使用例
 - ◇ <http://example.co.jp/demand-estimation-gtfs>
 - リクエストパラメータ

名称	説明	値	必須
tripId	GTFS-Flex にて定義された、オンデマンドバスの trip_id	文字列	○
pickUpLocationId	GTFS-Flex にて定義された、乗車地点の stops.stop_id、location_groups.location_group_id、あるいは location.geojeson の id	文字列	○

dropOffLocationId	GTFS-Flex にて定義された、降車地点の stops.stop_id、location_groups.location_group_id、あるいは location.geojeson の id	文字列	○
pickUpPosition.lat	乗車地点の緯度	浮動小数点数	○
pickUpPosition.lng	乗車地点の経度	浮動小数点数	○
dropOffPosition.lat	降車地点の緯度	浮動小数点数	○
dropOffPosition.lng	降車地点の経度	浮動小数点数	○
spaces[*].name	リクエストするスペースの種別名	SEAT, WHEEL_CHAIR	○
spaces[*].value	リクエストするスペースの数	整数	○
shareable	乗合を許容するか (本実証では true)	true, false	○
pickUpTime	乗車希望日時	日時 (RFC3339 形式)	※1
dropOffTime	降車希望日時	日時 (RFC3339 形式)	※1

※1 pickUpTime と dropOffTime のどちらか一方を必ず指定 (両方指定することはできない)

➤ リクエストボディサンプル

```
{
  "tripId": "trip_001",
  "pickUpLocationId": "pickUp_0001",
  "dropOffLocationId": "dropOff_0021",
  "pickUpPosition": {
    "lat": 35.457362,
    "lng": 139.633486
  },
  "dropOffPosition": {
    "lat": 35.457362,
    "lng": 139.633486
  },
  "spaces": [
    {
      "name": "SEAT",
      "value": 1
    }
  ],
  "shareable": true,
  "pickUpTime": "2021-01-22T10:00:00+09:00",
  "dropOffTime": null
}
```

➤ レスポンスパラメータ

名称	説明
header.gtfs_realtime_version	GTFS Realtime のバージョン (最新は 2.0)
header.timestamp	このメッセージが作成された POSIX time
entity.id	Feed 内で一意の ID
entity.on_demand[*] .wait_time_update.wait_location	リクエストで与えた乗車地点もしくは降車地点の stops.stop_id、location_groups.location_group_id、あるいは location.geojeson の id
entity.on_demand[*] .wait_time_update.trip_id	リクエストで与えた trip_id
entity.on_demand[*] .wait_time_update.wait_time	乗車地点もしくは降車地点で乗客が待つ推定待ち時間 (秒)
entity.on_demand[*] .wait_time_update.safe_wait_time	乗車地点もしくは降車地点で乗客が待つ待ち時間の 95 パーセントイル値 (秒)
entity.on_demand[*] .wait_time_update.max_wait_time	乗車地点もしくは降車地点で乗客が待つ最大推定待ち時間 (秒)
entity.on_demand[*] .wait_time_update.vehicle_availability	乗車地点もしくは降車地点での車両の利用可能性 NO_VEHICLE, LOW, MEDIUM, HIGH
entity.on_demand[*] .fare_update.fare_leg_id	運賃計算で用いる fare_leg_rules.fare_leg_id
entity.on_demand[*] .fare_update.origin	乗車地点の stops.stop_id、location_groups.location_group_id、 あるいは location.geojeson の id
entity.on_demand[*] .fare_update.destination	降車地点の stops.stop_id、location_groups.location_group_id、 あるいは location.geojeson の id
entity.on_demand[*] .fare_update.amount	リアルタイムに算定した運賃のうち、fare_leg_rules に定義された固定運賃部分の金額
entity.on_demand[*] .fare_update.fare_variable_amount[*] .fare_variable_id	リアルタイムに算定した運賃のうち、fare_variable_rules に定義された変動運賃部分の ID
entity.on_demand[*] .fare_update.fare_variable_amount[*] .amount	リアルタイムに算定した運賃のうち、fare_variable_rules に定義された変動運賃部分の金額

➤ レスポンスサンプル

(protocol buffers のバイナリを JSON として表現)

```
{
  "header": {
    "gtfs_realtime_version": "2.0",
```

```

    "timestamp": 1748738603
  },
  "entity": {
    "id": "001",
    "on_demand": [
      {
        "wait_time_update": {
          "wait_location": ["pickUp_0001"],
          "trip_id": "trip_001",
          "wait_time": 612,
          "max_wait_time": 1512
        }
      },
      {
        "wait_time_update": {
          "wait_location": ["dropOff_0021"],
          "trip_id": "trip_001",
          "wait_time": 1812,
          "max_wait_time": 2712
        }
      },
      {
        "fare_update": {
          "fare_leg_id": "fare_leg_005",
          "origin": ["pickUp_0001"],
          "destination": ["dropOff_0021"],
          "amount": 100.0,
          "fare_variable_amount": [
            {
              "fare_variable_id": "fare_variable_001",
              "amount": 25.3
            }
          ]
        }
      }
    ]
  }
}

```

【IF007】 乗客登録 API

- 本インターフェースの概要
 - 配車予約の主体となる乗客を登録する API
- 本インターフェースを利用する機能
 - **【FN011】** 利用者事前登録
 - **【FN103】** 乗客管理
- 本インターフェースを利用してやり取りを行うデータの詳細
 - SAVS の乗客情報（既存機能をそのまま利用）

【IF008】 配車予約登録 API

- 本インターフェースの概要
 - 配車予約を登録する API
- 本インターフェースを利用する機能
 - **【FN008】** オンデマンドバス配車予約
 - **【FN104】** 予約管理
- 本インターフェースを利用してやり取りを行うデータの詳細
 - SAVS の予約情報、予約登録結果（既存機能をそのまま利用）

【IF009】 待ち時間推定利用フラグ IF

- 本インターフェースの概要
 - マルチモーダル経路探索及びオンデマンドバス配車予約において、GTFS-Flex によるオンデマンドバスのサービス定義情報と GTFS-Ondemand のリアルタイムデータによる待ち時間推定の双方を利用するか、あるいは GTFS-Flex によるオンデマンドバスのサービス定義情報のみ用いるかを切り替えるフラグを与える IF
- 本インターフェースを利用する機能
 - **【FN006】** マルチモーダル経路探索
 - **【FN008】** オンデマンドバス配車予約
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-10 待ち時間推定利用フラグ IF

名称	説明	型	必須
demand_estimation_gtfs_flg	待ち時間推定利用可否を設定するフラグ True は利用する、False は利用しない	真偽値	○

【IF010】 GTFS-Flex 及び GTFS-Ondemand 運賃ルール IF

- 本インターフェースの概要
 - GTFS-Flex 及び GTFS-Ondemand の静的データから得た、オンデマンドバスの運賃ルールを定義す

る IF

- 本インターフェースを利用する機能
 - 【FN106】 オンデマンドバス運賃算定
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-11 GTFS-Flex 及び GTFS-Ondemand 運賃ルール IF

名称	説明	型	必須
fare_leg_rules.amount	運賃区間の運賃を示す。運賃変動ルールがこの区間に割当てられている場合（つまり、行に variable_group_id が定義されている場合）、運賃の固定価格（又は開始価格）として機能する	数値	○
fare_leg_rules.trip_cap_amount	開始料金と変動料金を合算した乗客が支払う最大料金	数値	
fare_leg_rules.trip_min_amount	開始料金と変動料金を合算した乗客が支払う最低料金	数値	
fare_leg_rules.price_adjustment	運行状況（例：需要の急増）に対応するために料金調整が行われる可能性があるかどうかを示す 0（又は空白） - 情報がない / 1 - 料金調整が行われる可能性がある / 2 - 料金調整は行われない	整数	
fare_leg_rules.tax_inclusion	税金が含まれるかどうかを示す 0（又は空白） - 情報がない / 1 - 税金が含まれる / 2 - 税金は含まれない	整数	
fare_leg_rules.tip_acceptance	チップを受け取るかどうかを示す 0（又は空白） - 情報がない / 1 - チップを受け取ることができる / 2 - チップは受け取らない	整数	
fare_variable_rules.fare_variable_type	変動料金の基準となる変数項目 0 - キロメートル（距離精算） / 1 - 分（時間精算） / 2 - アクティブ分（車両が走行している時間で精算） / 3 - アイドル分（車両が停止している時間で精算） / 4 - 乗客数 / 5 - 荷物 / 6 - ペット	整数	○
fare_variable_rules.interval	変動料金が適用される間隔。interval が 0 の場合、1 回のみ課金される fare_variable_type が 0, 1, 2, 3 の場合は必須。それ以外は禁止	数値	
fare_variable_rules.start	変動料金の適用が開始される値。このフィールドが空の場合、乗車開始と同時に金額が請求される	数値	
fare_variable_rules.end	変動料金の適用が終了する値。このフィールドが空の場合、乗車が終了するまで金額が請求される	数値	
fare_variable_rules.amount	単位当たりの変動料金	数値	

fare_variable_rules.currency	通貨	文字列	
fare_variable_rules. price_adjustment	運行状況（例：需要の急増）に対応するために料金調整が行われる可能性があるかどうかを示す 0（又は空白） - 情報がない / 1 - 料金調整が行われる可能性がある / 2 - 料金調整は行われない	整数	
fare_variable_rules. tax_inclusion	税金が含まれるかどうかを示す 0（又は空白） - 情報がない / 1 - 税金が含まれる / 2 - 税金は含まれない	整数	

【IF011】 算定運賃 IF

- 本インターフェースの概要
 - **【FN106】** オンデマンドバス運賃算定機能が推定したオンデマンドバスの運賃情報を連携するための IF
- 本インターフェースを利用する機能
 - **【FN107】** Ondemand 動的メッセージ生成
- 本インターフェースを利用してやり取りを行うデータの詳細
 - **【IF006】** GTFS-Ondemand リアルタイムデータ API のレスポンスの fare_update と同等（下記イメージを参照）

```
{
  "fare_update": {
    "fare_leg_id": "fare_leg_005",
    "origin": ["pickUp_0001"],
    "destination": ["dropOff_0021"],
    "amount": 100.0,
    "fare_variable_amount": [
      {
        "fare_variable_id": "fare_variable_001",
        "amount": 25.3
      }
    ]
  }
}
```

【IF012】 車両情報更新 API

- 本インターフェースの概要
 - オンデマンド配車予約で利用するための車両位置を保存する API
 - 車載端末より呼び出しを行う
- 本インターフェースを利用する機能

- 【FN101】 車両管理
- 【FN201】 ドライバーアプリ
- 本インターフェースを利用してやり取りを行うデータの詳細
 - SAVS の現在位置等の車両情報（既存機能をそのまま利用）

【IF013】 運行指示 API

- 本インターフェースの概要
 - オンデマンド配車予約で配車計算した結果より作成した、車両運行指示用データを取得する API
 - 車載端末より呼び出しを行う
- 本インターフェースを利用する機能
 - 【FN101】 車両管理
 - 【FN201】 ドライバーアプリ
- 本インターフェースを利用してやり取りを行うデータの詳細
 - SAVS の運行すべき経路情報（既存機能をそのまま利用）

【IF014】 利用者認証情報入力 IF

- 本インターフェースの概要
 - 【SL001】 駅探乗換案内を使用する利用者の登録、及び利用者の認証を行う Web UI の IF
 - 【UI001】 ログイン画面を参照
- 本インターフェースを利用する機能
 - 【FN004】 利用者認証
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-12 利用者登録・認証情報入力 IF

名称	説明	値	必須
ID	利用者 ID	文字列	○
Password	認証用パスワード	文字列	○

【IF015】 マルチモーダル経路検索条件入力 IF

- 本インターフェースの概要
 - マルチモーダル経路探索の探索条件を入力する Web UI の IF
 - 【UI004】 検索条件入力画面を参照
- 本インターフェースを利用する機能
 - 【FN006】 マルチモーダル経路探索
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-13 マルチモーダル経路検索条件入力 IF

名称	説明	値	必須
----	----	---	----

出発地	<ul style="list-style-type: none"> ● 定時定路線からオンデマンドバスへの乗り換え <ul style="list-style-type: none"> ➤ 駅・バス停名 ● オンデマンドバスから定時定路線への乗り換え <ul style="list-style-type: none"> ➤ 乗降ポイント名 (Checkpoint 型) ➤ 緯度・経度 (Zone-to-point 型) 	文字列	○
到着地	<ul style="list-style-type: none"> ● 定時定路線からオンデマンドバスへの乗り換え <ul style="list-style-type: none"> ➤ 乗降ポイント名 (Checkpoint 型) ➤ 緯度・経度 (Point-to-Zone 型) ● オンデマンドバスから定時定路線への乗り換え <ul style="list-style-type: none"> ➤ 駅・バス停名 	文字列	○
出発日時	出発日時	日時	※3
到着日時	到着日時	日時	※3

※3 出発日時と到着日時のどちらか一方を必ず指定（両方指定することはできない）

【IF016】 予約情報入力 IF

- 本インターフェースの概要
 - オンデマンドバスの配車予約をするために必要な情報を入力する Web UI の IF
 - **【UI004】** 検索条件入力画面を参照
- 本インターフェースを利用する機能
 - **【FN008】** オンデマンドバス配車予約
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-14 予約情報入力 IF

名称	説明	値	必須
人数	同行者含めオンデマンドバスに乗車する人数	整数	○
車椅子	オンデマンドバスに乗りこむ車椅子の台数 (※4)	整数	○
大型荷物	オンデマンドバスに持ち込む大型荷物の数 (※4)	整数	○

※4 車椅子の利用や大型荷物の持ち込みにはアルゴリズム上対応しているが、本実証の範囲ではこれらのパラメータは利用しない

【IF017】 配車予約取得 API

- 本インターフェースの概要
 - 条件に合致する配車予約を取得する API
- 本インターフェースを利用する機能

- 【FN009】 オンデマンドバス予約取得
- 【FN010】 オンデマンドバス予約キャンセル
- 【FN104】 予約管理
- 本インターフェースを利用してやり取りを行うデータの詳細
 - SAVS の乗客に紐づく配車予約の一覧（既存機能をそのまま利用）

【IF018】 配車キャンセル API

- 本インターフェースの概要
 - 配車予約をキャンセルする API
- 本インターフェースを利用する機能
 - 【FN010】 オンデマンドバス予約キャンセル
 - 【FN104】 予約管理
- 本インターフェースを利用してやり取りを行うデータの詳細
 - SAVS の配車予約キャンセルの成否（既存機能をそのまま利用）

【IF019】 事前登録用利用者データファイル IF

- 本インターフェースの概要
 - 事前登録する利用者の情報を記載したファイルを連携する IF
- 本インターフェースを利用する機能
 - 【FN011】 利用者事前登録
- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-15 事前登録用利用者データファイル IF

名称	説明	型	必須
id	駅探乗換案内にログインする ID	文字列	○
password	駅探乗換案内にログインするパスワード	文字列	○
name	ダミーの利用者名	文字列	○
phone	ダミーの電話番号	文字列	○

【IF020】 予約 ID 入力 IF

- 本インターフェースの概要
 - オンデマンドバスの配車予約の確認やキャンセルを行うために必要な配車予約 ID を入力する Web UI の IF
- 本インターフェースを利用する機能
 - 【FN009】 オンデマンドバス予約取得
 - 【FN010】 オンデマンドバス予約キャンセル

- 本インターフェースを利用してやり取りを行うデータの詳細

表 2-16 予約 ID 入力 IF

名称	説明	値	必須
配車予約 ID	配車予約時にシステムから通知される配車予約の ID	整数	○

【IF021】 定時定路線交通中間形式データ IF

- 本インターフェースの概要
 - 【IF001】 公共交通データファイル IF や 【IF003】 GTFS Schedule 定時定路線バスデータファイル IF から得た定時定路線公共交通のデータから変換された、駅探乗換案内独自の中間形式データを連携するための IF
 - この中間形式データを基に、駅探乗換案内独自の探索グラフを構成する
- 本インターフェースを利用する機能
 - 【FN001】 公共交通データ取得
 - 【FN002】 GTFS Schedule データ取得
 - 【FN003】 探索グラフ作成
- 本インターフェースを利用してやり取りを行うデータの詳細
 - 駅探乗換案内独自の中間形式データ（既存機能をそのまま利用）

【IF022】 探索グラフ更新 IF

- 本インターフェースの概要
 - 駅探乗換案内独自の定時定路線交通用の探索グラフ上へ、オンデマンドバスとの乗換ポイントとなるノードとエッジを追加する IF
- 本インターフェースを利用する機能
 - 【FN005】 探索グラフ Flex データ更新
- 本インターフェースを利用してやり取りを行うデータの詳細
 - 定時定路線交通用の探索グラフ

【IF023】 定時定路線乗換探索 IF

- 本インターフェースの概要
 - 探索グラフ上で定時定路線交通の乗換経路を探索するための IF
- 本インターフェースを利用する機能
 - 【FN006】 マルチモーダル経路探索
- 本インターフェースを利用してやり取りを行うデータの詳細
 - 定時定路線交通用の探索グラフ（既存機能をそのまま利用）

【IF024】 オンデマンドバス探索 IF

- 本インターフェースの概要

- 【FN007】 が SAVS から得たオンデマンドバスの待ち時間情報を連携するための IF
- 本インターフェースを利用する機能
 - 【FN006】 マルチモーダル経路探索
- 本インターフェースを利用してやり取りを行うデータの詳細
 - 【IF006】 GTFIS-Ondemand リアルタイムデータ API のレスポンスと同等（下記イメージを参照）

```
{
  "header": {
    "gtfs_realtime_version": "2.0",
    "timestamp": 1748738603
  },
  "entity": {
    "id": "001",
    "on_demand": [
      {
        "wait_time_update": {
          "wait_location": "pickUp_0001",
          "trip_id": "trip_001",
          "wait_time": 612,
          "max_wait_time": 1512
        }
      },
      {
        "wait_time_update": {
          "wait_location": "dropOff_0021",
          "trip_id": "trip_001",
          "wait_time": 1812,
          "max_wait_time": 2712
        }
      },
      {
        "fare_update": {
          "fare_leg_id": "fare_leg_005",
          "origin": ["pickUp_0001"],
          "destination": ["dropOff_0021"],
          "amount": 100.0,
          "fare_variable_amount": [
            {
              "fare_variable_id": "fare_variable_001",
```


ョンに持たせた利用者情報を連携する IF

- 本インターフェースを利用する機能
 - 【FN008】 オンデマンドバス配車予約
 - 【FN009】 オンデマンドバス予約取得
 - 【FN010】 オンデマンドバス予約キャンセル
- 本インターフェースを利用してやり取りを行うデータの詳細
 - 利用者 ID (文字列)
 - 対応する SAVS の乗客 ID (文字列)

【IF028】 リアルタイム配車計算結果 IF

- 本インターフェースの概要
 - リアルタイム配車計算を行った結果を連携する IF
- 本インターフェースを利用する機能
 - 【FN102】 配車計算エンジン
 - 【FN104】 予約管理
 - 【FN106】 オンデマンドバス運賃算定
- 本インターフェースを利用してやり取りを行うデータの詳細
 - SAVS の計算した結果の配車計画 (既存機能をそのまま利用)

【IF029】 車両情報 IF

- 本インターフェースの概要
 - リアルタイム配車計算に必要な車両の位置情報等を連携する IF
- 本インターフェースを利用する機能
 - 【FN102】 配車計算エンジン
- 本インターフェースを利用してやり取りを行うデータの詳細
 - SAVS の車両の現在位置、乗客の乗車状況 (既存機能をそのまま利用)

【IF030】 運行指示 IF

- 本インターフェースの概要
 - 計算された最新の配車予約情報を基に、車両が運行すべき経由地を指示する IF
- 本インターフェースを利用する機能
 - 【FN101】 車両管理
 - 【FN104】 予約管理
- 本インターフェースを利用してやり取りを行うデータの詳細
 - SAVS の運行すべき経路 (既存機能をそのまま利用)

2-5. ユーザーインターフェース (UI)

2-5-1. 画面遷移図

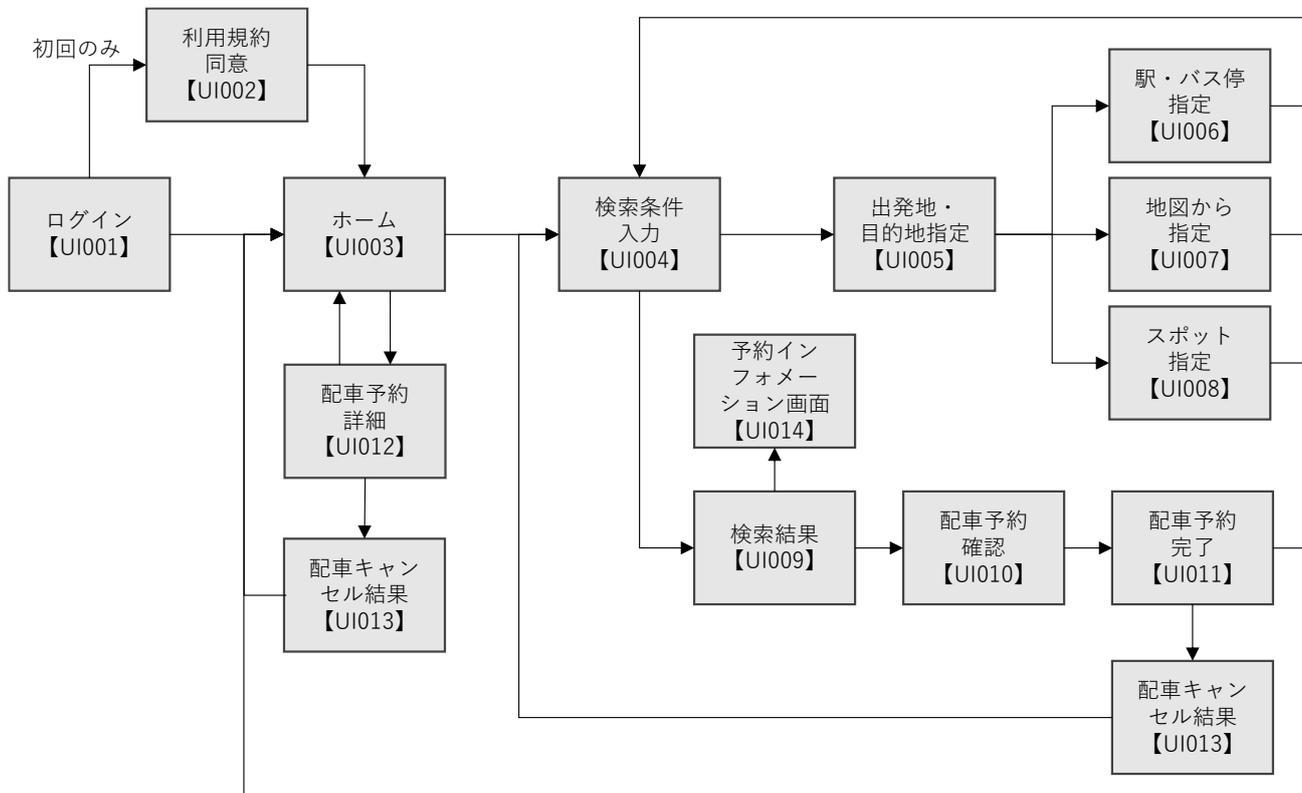


図 2-47 【SL001】 駅探乗換案内用画面遷移図

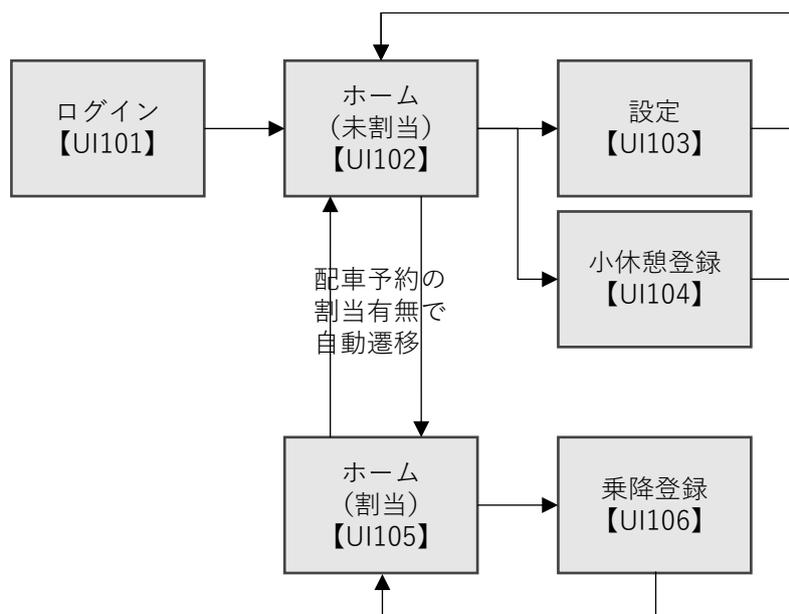


図 2-48 【SL201】 SAVS ドライバーアプリ用画面遷移図

2-5-2. ユーザーインターフェース一覧

表 2-18 【SL001】 駅探乗換案内用画面一覧

※朱文字：新規開発・既存改修

ID	画面名	説明	本画面から利用する機能
UI001	ログイン画面	● 被験者に割当てられた ID とパスワードで乗換探索アプリにログインする	FN004
UI002	利用規約同意画面	● 本実証の利用規約を表示し、同意を求める	FN004
UI003	ホーム画面	● 乗換探索アプリのホーム画面 ● すでに配車予約が成立している場合、予約の概要を表示する	FN006 FN009
UI004	検索条件入力画面	● マルチモーダル乗換探索の検索条件、及び配車予約に必要な条件を入力する	FN006
UI005	出発地・目的地指定画面	● 出発地及び目的地を指定する方法（駅・バス停から指定/スポットから指定/地図から指定）を選択する	FN006
UI006	駅・バス停指定画面	● 出発地、あるいは目的地として、定時定路線の駅・バス停を指定する	FN006
UI007	地図から指定画面	● 出発地、あるいは目的地を地図から指定する	FN006
UI008	スポット指定画面	● 出発地、あるいは目的地として、観光スポット等のランドマークを指定する	FN006
UI009	検索結果画面	● 検索条件を基にマルチモーダル乗換探索を実行して得た乗換経路を表示する	FN006 FN008
UI010	配車予約確認画面	● 選択した経路に含まれるオンデマンドバスの出発地や目的地、乗車条件などを確認し、配車を予約する	FN008
UI011	配車予約完了画面	● 配車予約の結果を表示する ● 乗車前であれば配車予約をキャンセルすることもできる	FN008
UI012	配車予約詳細画面	● 成立済みの配車予約の詳細を表示する ● 乗車前であれば配車予約をキャンセルすることもできる	FN009
UI013	配車キャンセル結果画面	● 配車予約をキャンセルした結果を表示する	FN010
UI014	予約インフォメーション画面	● GTFS-Flex 及び GTFS-Ondemand 拡張した静的ファイルから得た、予約に関するインフォメーションを利用者へ提示する	FN006

表 2-19 【SL201】 SAVS ドライバーアプリ用画面一覧

※朱文字：新規開発・既存改修

ID	画面名	説明	本画面から利用する機能
UI101	ログイン画面	<ul style="list-style-type: none"> ● 各運転手に割当てられたIDとパスワードでSAVSドライバーアプリにログインする 	FN201
UI102	ホーム画面 (未割当)	<ul style="list-style-type: none"> ● SAVS ドライバーアプリのホーム画面 ● 配車予約が割当てられていない場合はこの画面が表示される 	FN201
UI103	設定画面	<ul style="list-style-type: none"> ● ホーム画面での地図の表示方法の設定や、車両の運行を停止する場合などに用いる 	FN201
UI104	小休憩登録画面	<ul style="list-style-type: none"> ● 運転手が小休憩を登録する 	FN201
UI105	ホーム画面 (割当)	<ul style="list-style-type: none"> ● SAVS ドライバーアプリのホーム画面 ● 配車予約が割当てられると自動的に切り替わり、乗車地点や降車地点といった車両がたどるべき目的地が順番に指示される 	FN201
UI106	乗降登録画面	<ul style="list-style-type: none"> ● 目的地にて乗客が乗降したことを記録する 	FN201

2-5-3. ユーザーインターフェースの詳細

ユーザーインターフェース（画面）の詳細を記す。なお、本業務において開発（新規・改修）を行うユーザーインターフェース（画面）を**朱文字**で示す。

1) 【SL001】 駅探乗換案内用画面

【UI001】 ログイン画面

札幌MaaS

ID・パスワードを入力してください。

ID (必須)

ID

パスワード (必須)

パスワード

ログインする

図 2-49 ログイン画面

- 本画面の概要
 - 被験者に割当てられた ID とパスワードで乗換探索アプリにログインする
 - ログインボタンを押下すると、初回のみ【UI002】利用規約同意画面に遷移する
 - ログインボタンを押下すると、2回目以降は【UI003】ホーム画面へ直接遷移する
- 本画面から利用する機能
 - 【FN004】利用者登録・認証

【UI002】 利用規約同意画面

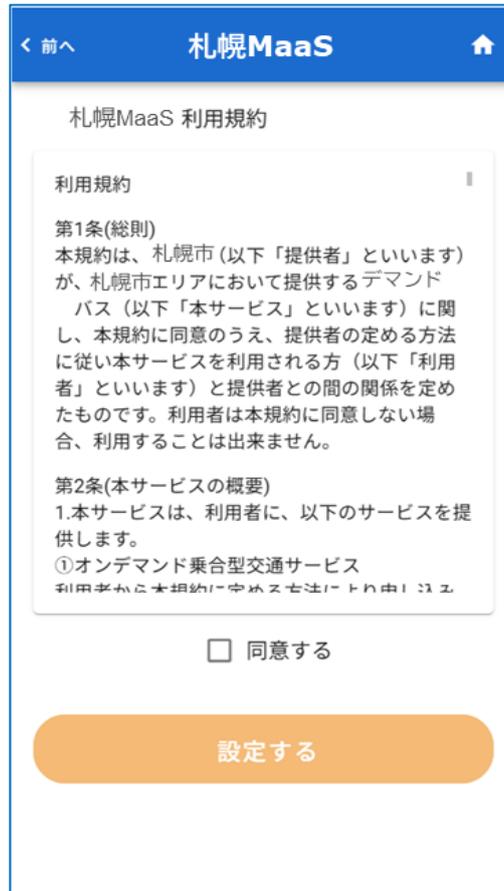


図 2-50 利用規約同意画面

- 本画面の概要
 - 本実証の利用規約を表示し、同意を求める
 - 同意するにチェックが入ると、設定するボタンが押せるようになり、押すとホーム画面へ遷移する
- 本画面から利用する機能
 - 【FN004】 利用者登録・認証

【UI003】 ホーム画面



図 2-51 ホーム画面

- 本画面の概要
 - 乗換案内アプリのホーム画面
 - メニューから GTFS-Flex 及び GTFS-Ondemand にてマルチモーダル乗換探索を行うか、GTFS-Flex のみでマルチモーダル乗換探索を実施するかを選択することができる
 - メニューからログアウトすることができる
 - すでに配車予約が成立している場合、予約の概要の画面が表示される
 - ルート検索ボタンを押下すると、【UI004】 検索条件入力画面に遷移する
- 本画面から利用する機能
 - 【FN006】 マルチモーダル経路探索
 - 【FN009】 オンデマンドバス予約取得

【UI004】 検索条件入力画面

図 2-52 検索条件入力画面

- 本画面の概要
 - マルチモーダル乗換探索の検索条件、及び配車予約に必要な条件を入力する
 - ◇ マルチモーダル乗換探索の検索条件
 - 出発地

- 目的地
- 乗車希望日時あるいは降車希望日時
- 乗換時の移動速度（ゆっくり/普通/はやめ）
- ◇ 配車予約の条件
 - 人数
 - 車椅子の台数
 - 大型荷物の数
- ◇ その他
 - 検索結果の表示順序
- 出発地ボタン（あるいは目的地ボタン）を押下すると、【UI005】出発地・目的地指定画面へ遷移する
- ルート検索ボタンを押下すると、入力された条件に従って【FN006】マルチモーダル乗換探索が乗換ルートを検索し、その結果を【UI009】検索結果画面にて表示する
- 本画面から利用する機能
 - 【FN006】マルチモーダル経路探索

【UI005】 出発地・目的地指定画面



図 2-53 出発地・目的地指定画面

- 本画面の概要
 - 出発地（あるいは目的地）の指定方法を選択する
- 本画面から利用する機能
 - 【FN006】 マルチモーダル経路探索

【UI006】 駅・バス停指定画面

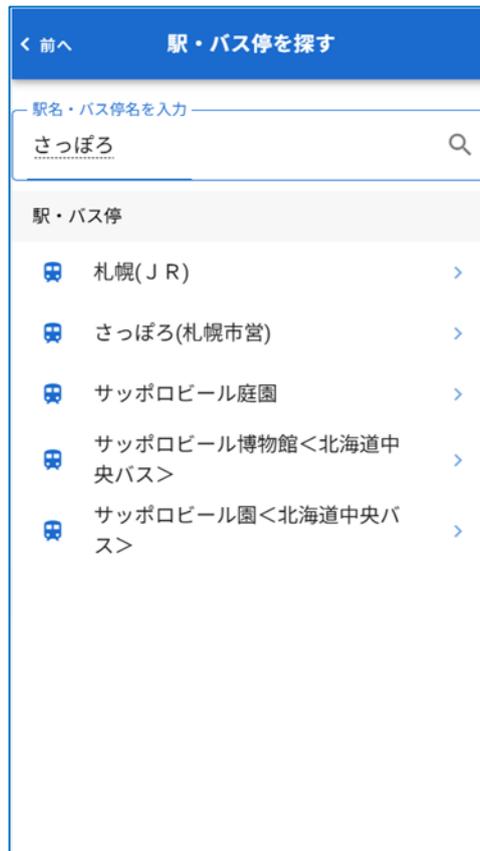


図 2-54 駅・バス停指定画面

- 本画面の概要
 - 入力フィールドに駅名やバス停名を入力すると、インクリメンタルサーチを実施して合致する駅・バス停の候補を表示する
 - 表示された駅・バス停を押下すると、該当の駅・バス停が出発地（あるいは目的地）として指定された状態で【UI004】 検索条件入力画面へ遷移する
- 本画面から利用する機能
 - 【FN006】 マルチモーダル経路探索

【UI007】 地図から指定画面



図 2-55 地図から指定画面

- 本画面の概要
 - 地図をタップして出発地（あるいは目的地）にピンを立てる
 - ピンを立てた後に出発地にするボタン（あるいは目的地にするボタン）を押下すると、ピンの位置の緯度・経度が出発地（あるいは目的地）として指定された状態で【UI004】 検索条件入力画面へ遷移する
- 本画面から利用する機能
 - 【FN006】 マルチモーダル経路探索

【UI008】 スポット指定画面

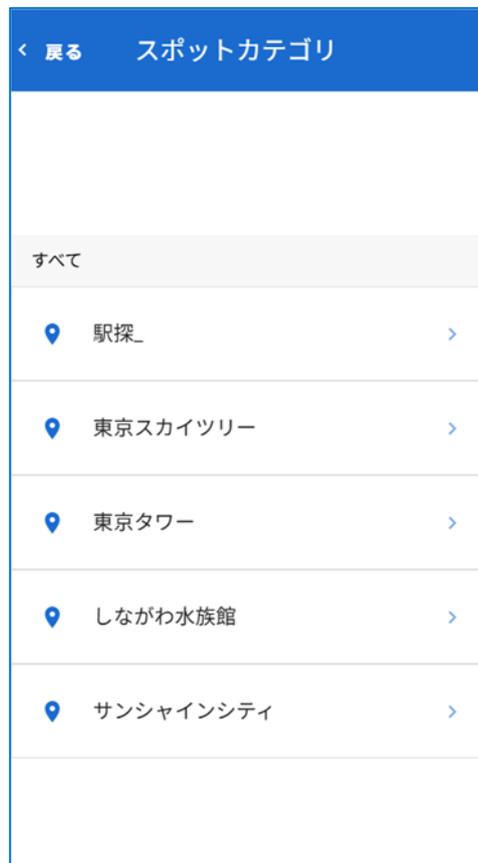


図 2-56 スポット指定画面

- 本画面の概要
 - 【DT007】 実証用オンデマンドバス静的データの stops.txt に定義された Checkpoint 型の乗降ポイントを抽出し、スポットとして一覧表示する
 - スポットを押下すると、該当スポットが出発地（あるいは目的地）として指定された状態で【UI004】 検索条件入力画面へ遷移する
- 本画面から利用する機能
 - 【FN006】 マルチモーダル経路探索

【UI009】 検索結果画面

The screenshot shows a route search interface with the following details:

- 出発 (Departure):** 中島公園 (Nakajima Park)
- 到着 (Arrival):** 千歳鶴酒ミュージアム (Chikuzen Sake Museum)
- 時間 (Time):** 2025年11月30日 15:00 出発
- Route Options:**
 - Option 1: 15:03 → 15:15 (12分) 乗換1回
 - Option 3: 15:03 → 15:20 (17分) 乗換1回
 - Option 4: 15:03 → 15:28 (25分) 乗換1回
- Selected Route (Option 1):**
 - 15:03 発 - 15:15 着 340円
 - 12分 標準ルート 最遅ルート
 - 15:03 発 中島公園 210円
 - 1分 札幌市営地下鉄南北線 普通麻生行
 - 15:04 着 すすきの(市営)
 - 15:04 発 すすきの(市営)
 - 2分 徒歩
 - 15:06 着 すすきの 130円
 - 15:06 発 すすきの
 - 9分 デマンド交通 デマンドバス予約 100円
 - 検索モード: GTFS-Ondemand有効
 - 予約タイプ: リアルタイム予約
 - 乗合: 許可
 - 荷物可否: 可
 - ペット可否: 不可
 - 平均待ち時間: 15分
 - 実証実験の為、運賃は表示のみで取受いたしません。予約時間を表示しています。
 - 15:15 着 千歳鶴酒ミュージアム
- Option 2:** 15:03 発 - 15:20 着 210円 (17分)
- Option 3:** 15:03 発 中島公園 210円

Callouts from the left side of the image point to specific features:

- GTFS-Ondemandリアルタイムメッセージから得た、標準的な待ち時間の場合と最遅の場合の切り替え (Switching between standard and latest waiting times based on real-time messages).
- GTFS-Flexで定められた予約情報 (重要なものを抜粋) (Booking information defined by GTFS-Flex, excerpting important items).
- GTFS-Flex (booking_rules) で設定した利用者向けメッセージ (Message for users set in GTFS-Flex (booking_rules)).

An inset map on the right shows the route path from Nakajima Park to Chikuzen Sake Museum, highlighting the demand bus segment.

図 2-57 検索結果画面

- 本画面の概要
 - 検索条件を基にマルチモーダル乗換探索を実行して得た乗換経路を表示する
 - オンデマンドバスの経路として表示される時間が予測時間であることを明示する
 - 【SL101】 SAVS から得たオンデマンドバスの距離によって変動する運賃を表示する
 - オンデマンドバスが標準的な待ち時間で運行した場合の乗換経路と、最遅の待ち時間で運行した場合の乗換経路を切り替えて表示できる

- オンデマンドバス特有の Notice を利用者へ表示するために、GTFS-Flex (booking_rules) にて定義されている利用者向けメッセージを表示する
- 地図ボタンを押下すると、出発地と目的地をプロットした地図を表示する
- 予約ボタンを押下すると、【UI010】配車予約確認画面に遷移する
- 情報ボタンを押下すると、【UI014】予約インフォメーション画面をポップアップ表示する
- 本画面から利用する機能
 - 【FN006】マルチモーダル経路探索
 - 【FN008】オンデマンドバス配車予約

【UI010】配車予約確認画面

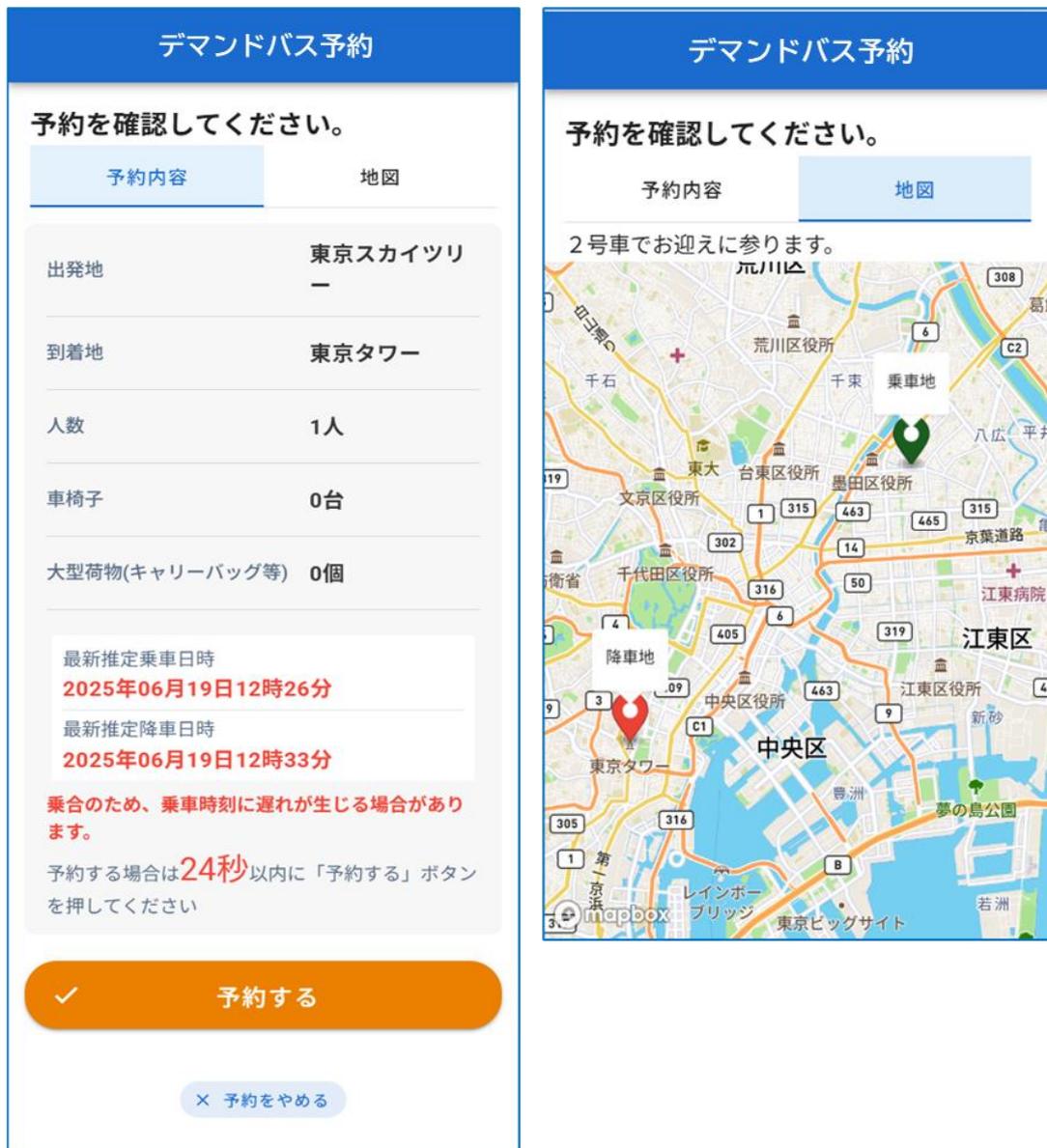


図 2-58 配車予約確認画面

- 本画面の概要
 - 【UI004】検索条件入力画面にて与えられた条件を基に探索した、配車可能なオンデマンドバスの情

報を確認する

- 地図ボタンを押下すると、オンデマンドバスの乗車地と降車地がプロットされた地図を確認することができる
- 制限時間内に予約ボタンを押下すると、【FN008】オンデマンドバス配車予約機能が【SL101】オンデマンド配車予約と連携し、オンデマンドバスの配車予約を確定させた後に【UI011】に遷移し、配車予約の成立結果を表示する
- 本画面から利用する機能
 - 【FN008】オンデマンドバス配車予約

【UI011】配車予約完了画面



図 2-59 配車予約完了画面

- 本画面の概要
 - 配車予約の確定結果を表示する
 - 地図ボタンを押下すると、オンデマンドバスの乗車地と降車地がプロットされた地図を確認することができる
 - オンデマンドバスに乗車する前であれば、配車予約をキャンセルすることもできる
- 本画面から利用する機能
 - 【FN008】オンデマンドバス配車予約

【UI012】配車予約詳細画面



図 2-60 配車予約詳細画面

- 本画面の概要
 - 成立済みの配車予約の情報を表示する
 - 地図ボタンを押下すると、オンデマンドバスの乗車地と降車地がプロットされた地図を確認することができる
 - オンデマンドバスに乗車する前であれば、配車予約をキャンセルすることもできる
- 本画面から利用する機能
 - 【FN009】オンデマンドバス予約取得

【UI013】配車キャンセル結果画面



図 2-61 配車キャンセル結果画面

- 本画面の概要
 - 成立済みの配車予約をキャンセルした結果を表示する
- 本画面から利用する機能
 - 【FN010】 オンデマンドバス予約キャンセル

【UI014】予約インフォメーション画面

閉じる 予約情報		閉じる 予約情報		閉じる 予約情報	
予約タイプ	リアルタイム予約	予約タイプ	当日予約	予約タイプ	前日までに予約
サービス開始時間	09:00	予約受付時間	8時間前から1時間前まで	予約開始日時	7日前の09:00
サービス終了時間	16:00	サービス開始時間	09:00	予約終了日時	前日の16:00 (土日祝日を問わずにカウント)
乗合	許可	サービス終了時間	16:00	サービス開始時間	09:00
荷物可否	可	乗合	許可	サービス終了時間	16:00
ペット可否	不可	荷物可否	可	乗合	許可
介助動物可否	不可	ペット可否	不可	荷物可否	可
有料旅程可否	不可	介助動物可否	不可	ペット可否	不可
平均的な待ち時間	15分	有料旅程可否	不可	介助動物可否	不可
安全な待ち時間	15分	平均的な待ち時間	15分	有料旅程可否	不可
最大の待ち時間	30分	安全な待ち時間	15分	平均的な待ち時間	15分
リクエストにできる限り近い場所で乗車可		最大の待ち時間	30分	安全な待ち時間	15分
リクエストにできる限り近い場所で降車可		リクエストにできる限り近い場所で乗車可		最大の待ち時間	30分
リクエストにできる限り近い場所で降車可		リクエストにできる限り近い場所で降車可		リクエストにできる限り近い場所で乗車可	
予約問い合わせ電話番号	080-xxxx-xxxx	リクエストにできる限り近い場所で降車可		リクエストにできる限り近い場所で降車可	
予約システムURL	(駅探システムjのURL)	予約問い合わせ電話番号	080-xxxx-xxxx	予約問い合わせ電話番号	080-xxxx-xxxx
予約ルールURL	(駅探システムjのURL)	予約システムURL	(駅探システムjのURL)	予約システムURL	(駅探システムjのURL)
予約アプリダウンロード	https://~~~~ (ダミー)	予約ルールURL	(駅探システムjのURL)	予約ルールURL	(駅探システムjのURL)
予約アプリ起動 (Android)	https://~~~~ (ダミー)	予約アプリダウンロード	https://~~~~ (ダミー)	予約アプリダウンロード	https://~~~~ (ダミー)
予約アプリ起動 (iOS)	https://~~~~ (ダミー)	予約アプリ起動 (Android)	https://~~~~ (ダミー)	予約アプリ起動 (Android)	https://~~~~ (ダミー)
予約アプリ起動 (Web)	https://~~~~ (ダミー)	予約アプリ起動 (iOS)	https://~~~~ (ダミー)	予約アプリ起動 (iOS)	https://~~~~ (ダミー)
		予約アプリ起動 (Web)	https://~~~~ (ダミー)	予約アプリ起動 (Web)	https://~~~~ (ダミー)

booking_type=0の場合

booking_type=1の場合

booking_type=2の場合

図 2-62 予約インフォメーション画面

- 本画面の概要
 - オンデマンドバスの予約に関するインフォメーションを利用者へ提示する
- 本画面から利用する機能
 - 【FN006】マルチモーダル経路探索

2) 【SL201】SAVS ドライバーアプリ用画面

【UI101】ログイン画面

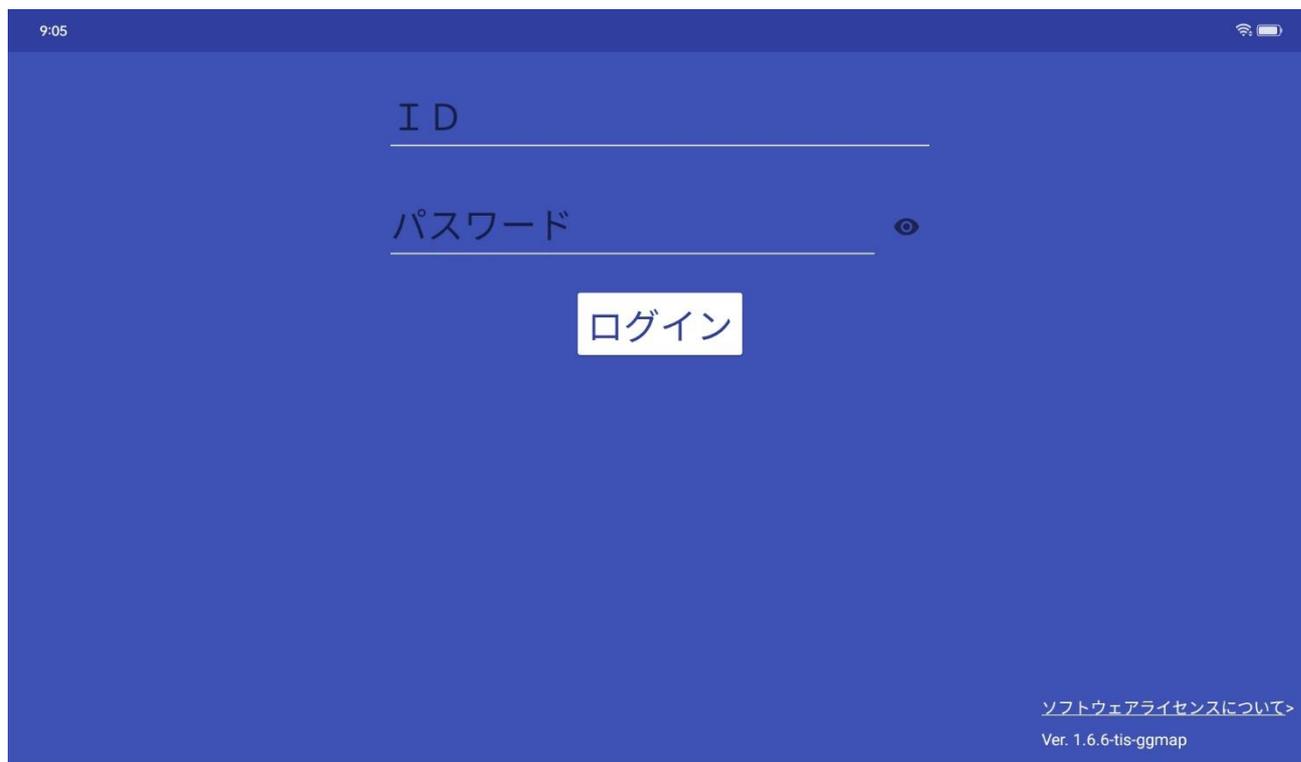


図 2-63 ログイン画面

- 本画面の概要
 - 各運転手に割当てられた ID とパスワードで SAVS ドライバーアプリにログインする
- 本画面から利用する機能
 - 【FN201】ドライバーアプリ

【UI102】ホーム画面（未割当）



図 2-64 ホーム画面（未割当）

- 本画面の概要
 - SAVS ドライバーアプリのホーム画面
 - 当該車両へ配車予約が割当てられていない場合はこの画面が表示される
- 本画面から利用する機能
 - 【FN201】ドライバーアプリ

【UI103】設定画面

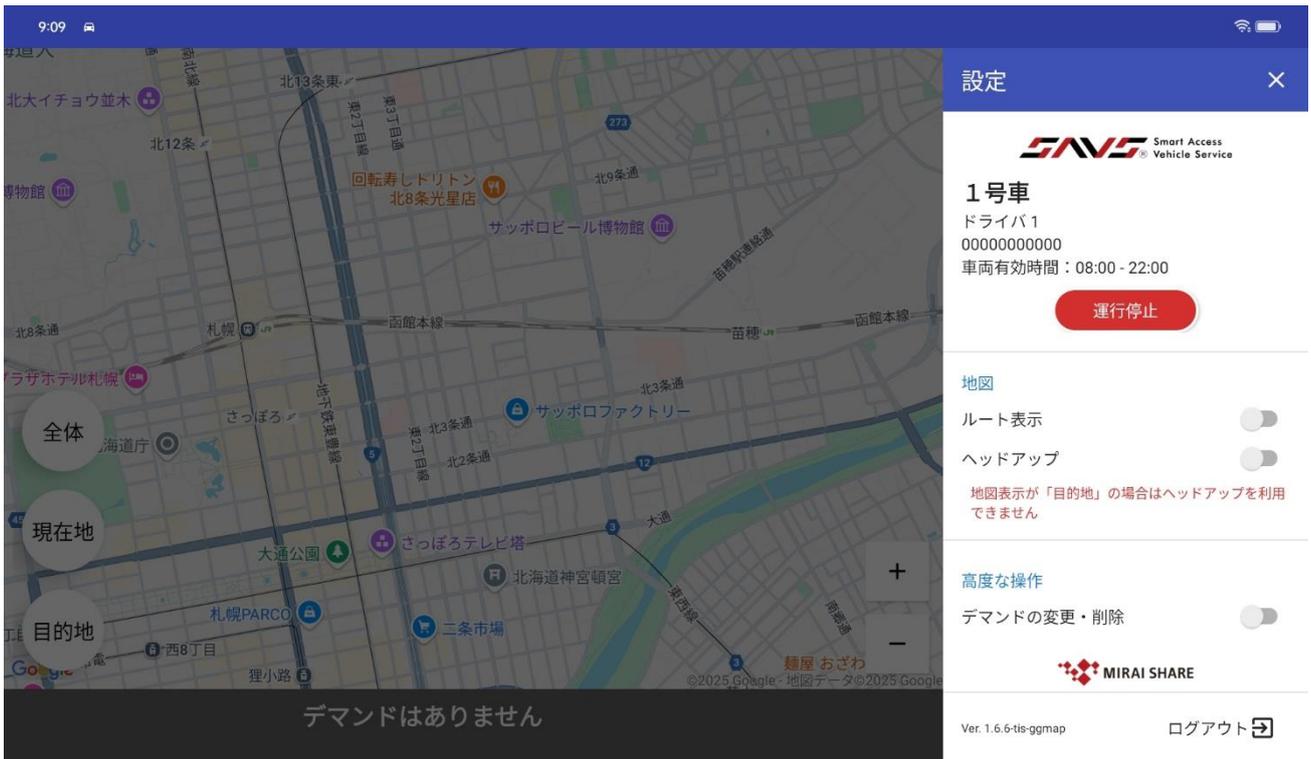


図 2-65 設定画面

- 本画面の概要
 - 車両の運行ができなくなった場合に、運行停止ボタンを押すことで当該車両への配車予約の割当てを停止することができる
 - ホーム画面で表示される地図の表示方法をカスタマイズすることができる
 - 当該車両に割当てられた配車予約を、運転手の変更・削除しても良いか切り替えられる
 - SAVS ドライバーアプリからログアウトすることができる
- 本画面から利用する機能
 - 【FN201】ドライバーアプリ

【UI104】小休憩登録画面



図 2-66 小休憩登録画面

- 本画面の概要
 - 運転手が小休憩を取りたい場合、この画面を使って小休憩を登録する
 - 当該車両に割り当てられている配車予約を勧案し、取得可能な時間に小休憩を取るよう指示される
- 本画面から利用する機能
 - 【FN201】ドライバーアプリ

【UI105】 ホーム画面（割当）

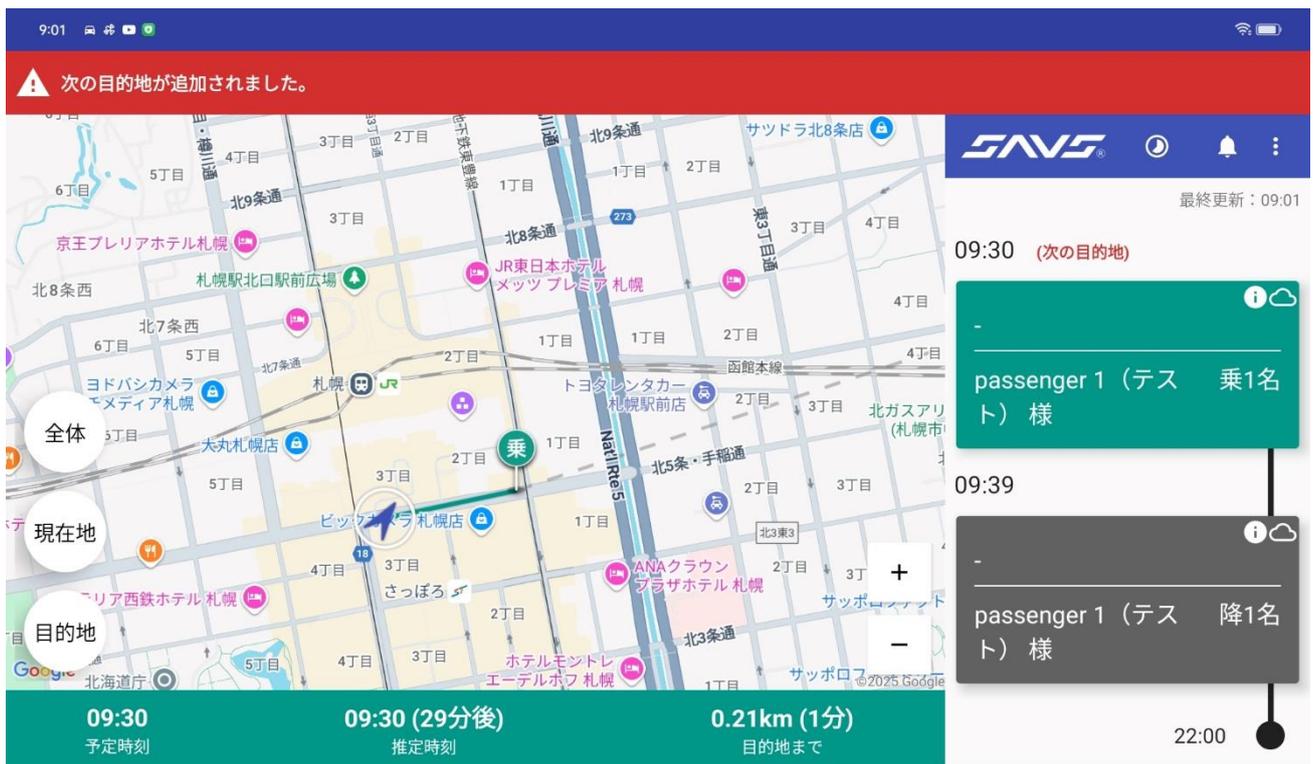


図 2-67 ホーム画面（割当）

- 本画面の概要
 - 当該車両に配車予約が割当てられると SAVS ドライバーアプリの【UI102】ホーム画面（未割当）が【UI105】ホーム画面（割当）へ自動的に切り替わる
 - 乗車地点や降車地点といった車両がたどるべき目的地が順番に指示される
 - 次の目的地に到着すべき推定時刻と、移動に要する時間の予測も表示される
 - 配車予約の変更・削除が運転手に許可されている場合、割当てられた配車予約をこの画面から変更・削除することができる
 - 目的地に到着した際に、運転手が右ペインに表示されている当該目的地のボックスを押すと、その目的地の【UI106】乗降登録画面へ遷移する
- 本画面から利用する機能
 - 【FN201】ドライバーアプリ

【UI106】乗降登録画面

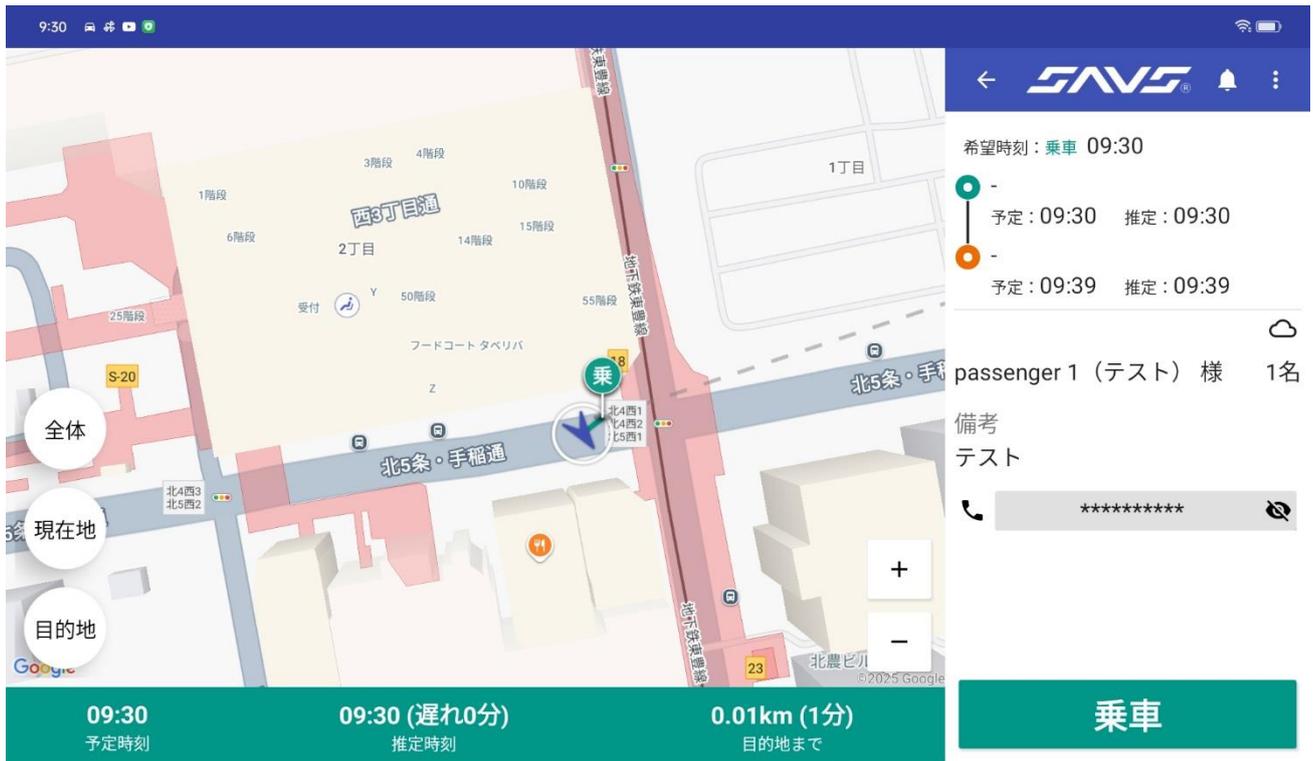


図 2-68 乗降登録画面

- 本画面の概要
 - 指示された目的地で乗客が乗降した際に、運転手がこの画面で乗車や降車を登録する
 - 乗車や降車が登録されると【UI105】ホーム画面（割当）に戻り、次の目的地への移動が指示される
 - 最後の目的地で乗客の降車を登録した後に、【UI102】ホーム画面（未割当）へ自動的に遷移する
- 本画面から利用する機能
 - 【FN201】ドライバーアプリ

3. 開発するシステム：非機能要件（NF）

3-1. 非機能要件一覧

表 3-1 非機能要件一覧

カテゴリ	ID	非機能項目	要件詳細
可用性	NF001	安定稼働	<ul style="list-style-type: none"> ● 実証を実施するに足る品質でシステムが安定稼働することを目標に、実証用システム全体の稼働率を95%/月以上にすること ● 実証中に障害が発生した場合、ログ等を解析して障害原因を突き止め、システムアーキテクチャやコンポーネントの改善点を見出すこと
性能・拡張性	NF002	応答速度	<ul style="list-style-type: none"> ● マルチモーダル乗換探索アプリのランディングページの応答時間が90パーセンタイル値で3秒以下であること ● マルチモーダル乗換探索アプリの乗換探索結果ページの応答時間が90パーセンタイル値で7秒以下であること
その他	NF003	商用サービスの活用	<ul style="list-style-type: none"> ● 本実証用に新たにシステムを開発して実証するのではなく、商用として運用されている乗換案内サービスやオンデマンドバス配車予約サービスを活用して実証を行うことで、社会実装に近い商用グレードの可用性、保守性、利用性、拡張性、セキュリティを担保すること

3-2. 非機能要件の詳細

【NF001】安定稼働

- 本非機能要件の概要
 - 実証を実施するに足る品質でシステムが安定稼働することを目標に、実証用システム全体の稼働率を計測し評価する
 - KPI としては 95%/月以上と設定
 - 実証中に障害が発生した場合、ログ等を解析して障害原因を突き止め、システムアーキテクチャやコンポーネントの改善点を見出す
- 設定理由
 - 10 月及び 11 月の実証期間にはそれぞれ予備日 1 日を見込んでおり、95%/月以上の稼働率でシステムが動作すれば、滞りなく実証を行うことができるため

【NF002】応答速度

- 本非機能要件の概要
 - マルチモーダル乗換探索アプリの利用者が途中離脱しないことを目標に、ランディングページの応答時間を計測し評価する
 - KPI としては 90 パーセンタイル値で 3 秒以下を設定
 - 同様に、乗換探索結果ページの応答時間を計測し評価する
 - KPI としては 90 パーセンタイル値で 7 秒以下を設定
- 設定理由
 - ランディングページの応答時間として 3 秒を設定した理由は、一般的に 3 秒を越えるとページからの離脱率が上がると言われているため
 - 乗換探索結果ページの応答時間として 7 秒を設定した理由は、【AL001】マルチモーダル乗換探索アルゴリズムが動作する最も重い処理であり、駅探及び未来シェアの過去事例から 7 秒程度と見込まれるため

【NF003】商用サービスの活用

- 本非機能要件の概要
 - 本実証用に新たにシステムを開発して実証するのではなく、商用として運用されている乗換案内サービスやオンデマンドバス配車予約サービスを活用して実証を行う
 - 乗換案内サービスは（株）駅探のサービスを、オンデマンドバス配車予約サービスは（株）未来シェアの SAVS を利用する
- 設定理由
 - 本実証用に新たにシステムを開発して実証するのではなく、商用サービスを改修して実証することで、社会実装に近い UI/UX やシステムアーキテクチャで実証を行うことができるため

4. 実証調査に利用するデータ (DT)

4-1. 実証調査に利用するデータ一覧

表 4-1 実証調査に利用するデータ一覧

※朱文字：本実証で変換・作成するデータ

ID	データ名称	データ形式	出所	データを利用する ID
DT001	JR 北海道時刻表データ	独自形式	交通新聞社	FN001
DT002	札幌市営地下鉄時刻表データ	独自形式	JTB パブリッシング	FN001
DT003	札幌市電(路面電車)時刻表データ	独自形式	JTB パブリッシング	FN001
DT004	JR 北海道バス時刻表データ	GTFS Schedule	札幌市	FN002
DT005	中央バス時刻表データ	GTFS Schedule	札幌市	FN002
DT006	じょうてつバス時刻表データ	GTFS Schedule	札幌市	FN002
DT007	実証用オンデマンドバスデータ	GTFS-Flex 及び GTFS-Ondemand	本実証用に作成	FN005 FN012 FN105

4-2. 実証調査に利用するデータの詳細

実証調査に利用するデータの詳細を記す。なお、本業務において変換・生成を行うデータを**朱文字**で示す。

【DT001】 JR 北海道時刻表データ

- 本データの概要
 - 実証エリア近傍の JR 北海道の路線（函館本線、千歳線、札沼線）の時刻表データ
- データ定義

表 4-2 JR 北海道時刻表データのサンプル・イメージ

No.	日本語名称
1	列車番号
2	駅名
3	到着時刻
4	出発時刻
5	順序
6	運賃

- 本データの形式
 - 独自形式
- 出所
 - 交通新聞社 (<https://www.kotsu.co.jp/>)

【DT002】 札幌市営地下鉄時刻表データ

- 本データの概要
 - 実証エリア近傍の札幌市営地下鉄の路線（南北線、東豊線、東西線）の時刻表データ
- データ定義

表 4-3 札幌市営地下鉄時刻表データのサンプル・イメージ

No.	日本語名称
1	列車番号
2	駅名
3	到着時刻
4	出発時刻
5	順序
6	運賃

- 本データの形式
 - 独自形式
- 出所
 - JTB パブリッシング (<https://jtbpublishing.co.jp/>)

【DT003】札幌市電（路面電車）時刻表データ

- 本データの概要
 - 実証エリア近傍の札幌市電（路面電車）の路線の時刻表データ
- データ定義

表 4-4 札幌市電（路面電車）時刻表データのサンプル・イメージ

No.	日本語名称
1	列車番号
2	駅名
3	到着時刻
4	出発時刻
5	順序
6	運賃

- 本データの形式
 - 独自形式
- 出所
 - JTB パブリッシング (<https://jtbpublishing.co.jp/>)

【DT004】JR 北海道バス時刻表データ

- 本データの概要
 - 実証エリア近傍を走行する定時定路線バスである JR 北海道バスの時刻表データ
 - GTFS-JP 第 3 版に従い以下の項目を設定した GTFS Schedule データへ変換して使用
- データ定義

表 4-5 JR 北海道バス時刻表データのサンプル・イメージ (agency.txt)

No.	日本語名称	必須	書式・選択肢など
1	事業者 ID	○	● 事業者の法人番号を設定
2	事業者名称	○	● 経路検索で案内する場合に適切な名称を設定。正式名称である必要はなく、旅客が交通機関を識別しやすい名称を設定
3	事業者 URL	○	● 事業者 HP のトップページの URL を設定
4	タイムゾーン	○	● 「Asia/Tokyo」を設定
5	言語	○	● 「ja」を設定
6	電話番号	-	● 窓口電話番号を設定

表 4-6 JR 北海道バス時刻表データのサンプル・イメージ (stops.txt)

No.	日本語名称	必須	書式・選択肢など
1	停留所・標柱 ID	○	● 事業者が内部的に使用しているコードをそのまま設定
2	停留所・標識名称	○	● 停留所名を設定。その地域の住民や旅行者が理解できる名前を使用
3	緯度	○	● 標柱が設置されている場所の緯度を設定
4	経度	○	● 標柱が設置されている場所の経度を設定
5	運賃エリア ID	○	● 運賃エリアを設定
6	停留所・標柱区分	-	● 停留所なのか標柱なのか設定 ● 0:標柱 ● 1:停留所

表 4-7 JR 北海道バス時刻表データのサンプル・イメージ (routes.txt)

No.	日本語名称	必須	書式・選択肢など
1	経路 ID	○	● 事業者が内部的に使用しているコードをそのまま設定
2	事業者 ID	○	● agency から参照
3	経路略称	※1	● 系統番号あるいは路線名称等、当該系統を識別可能な略称等を設定
4	経路名	※1	● 経由地や目的地等を含んだ経路に関する詳細な情報を設定
5	経路タイプ	○	● 「3」を設定

※1 経路略称と経路名のどちらかは必ず設定する。

表 4-8 JR 北海道バス時刻表データのサンプル・イメージ (trips.txt)

No.	日本語名称	必須	書式・選択肢など
1	経路 ID	○	● routes から参照
2	運行日 ID	○	● calendar から参照
3	便 ID	○	● 便を特定する ID を指定
4	便行先	-	● 便としての行先と経由を設定
5	往復区分	-	● その便の往復区分を指定 ● 0:復路 ● 1:往路

表 4-9 JR 北海道バス時刻表データのサンプル・イメージ (stop_times.txt)

No.	日本語名称	必須	書式・選択肢など
1	便 ID	○	● trips から参照
2	到着時刻	○	● その便のその標柱への到着時刻を設定 ● 起点はその標柱からの出発時刻と同じ時刻を設定
3	出発時刻	○	● その便のその標柱からの出発時刻を設定 ● 終点はその標柱への到着時刻と同じ時刻を設定

4	標柱 ID	○	● stops から参照
5	通過順位	○	● その便での該当標柱の通過順序を指定 ● 通過順に昇順で数値を設定 ● 必ずしも連番である必要はない
6	乗車区分	-	● 0:通常の乗車地 1:乗車不可能
7	降車区分	-	● 0:通常の降車地 1:降車不可能

表 4-10 JR 北海道バス時刻表データのサンプル・イメージ (calendar.txt)

No.	日本語名称	必須	書式・選択肢など
1	運行日 ID	○	● 運行区分を表す値を設定
2	月曜日	○	● 0:月曜日に非運行 1:月曜日に運行
3	火曜日	○	● 0:火曜日に非運行 1:火曜日に運行
4	水曜日	○	● 0:水曜日に非運行 1:水曜日に運行
5	木曜日	○	● 0:木曜日に非運行 1:木曜日に運行
6	金曜日	○	● 0:金曜日に非運行 1:金曜日に運行
7	土曜日	○	● 0:土曜日に非運行 1:土曜日に運行
8	日曜日	○	● 0:日曜日に非運行 1:日曜日に運行
9	サービス開始日	○	● 適用を開始する日付を指定 ● YYYYMMDD 形式
10	サービス終了日	○	● 適用を終了する日付を指定 ● YYYYMMDD 形式

表 4-11 JR 北海道バス時刻表データのサンプル・イメージ (calendar_dates.txt)

No.	日本語名称	必須	書式・選択肢など
1	運行日 ID	○	● calendar から参照
2	日付	○	● 運行日 ID で指定される運行区分の利用タイプを設定する日付を指定 ● YYYYMMDD 形式
3	利用タイプ	○	● date で指定された日に、運行日 ID で指定されている運行区分が適用されるかを指定 ● 1:運行区分適用 ● 2:運行区分非適用

表 4-12 JR 北海道バス時刻表データのサンプル・イメージ (fare_attributes.txt)

No.	日本語名称	必須	書式・選択肢など
1	運賃 ID	○	● fare_rules と紐づけるための ID を設定
2	運賃	○	● 運賃 (円) を指定
3	通貨	○	● 「JPY」を設定

4	支払いタイミング	○	<ul style="list-style-type: none"> ● 運賃の支払いタイミング指定 ● 0:乗車後に支払う 1:乗車前に支払う
5	乗換	○	<ul style="list-style-type: none"> ● 料金で許可される乗換回数を指定 ● 0:乗換不可 ● 1:一度の乗換が可能 ● 2:二度の乗換が可能 ● 空白:乗換回数に制限がない
6	乗換有効期限	-	<ul style="list-style-type: none"> ● 乗換が可能な場合、乗換期限が切れるまでの時間を秒数で指定 ● 乗換を認めない場合、ここでの設定値は運賃の有効期限となる ● 意図的な期限を設定しない場合、空白か値を指定しない

表 4-13 JR 北海道バス時刻表データのサンプル・イメージ (fare_rules.txt)

No.	日本語名称	必須	書式・選択肢など
1	運賃 ID	○	● fare_attributes から参照
2	経路 ID	-	● routes から参照
3	乗車地ゾーン	-	● 乗車地の zone_id を設定
4	降車地ゾーン	-	● 降車地の zone_id を設定
5	通過ゾーン	-	● 使用しない

表 4-14 JR 北海道バス時刻表データのサンプル・イメージ (feed_info.txt)

No.	日本語名称	必須	書式・選択肢など
1	提供組織名	○	● データを公開する組織の正式名称を指定
2	提供組織 URL	○	● データ公開組織の URL を指定
3	提供言語	○	● 「ja」設定
4	有効期間開始日	-	<ul style="list-style-type: none"> ● データが有効な期間の開始日を設定 ● YYYYMMDD 形式
5	有効期間終了日	-	<ul style="list-style-type: none"> ● データが有効な期間の終了日を設定 ● YYYYMMDD 形式
6	提供データバージョン	-	● 提供しているデータのバージョンを設定

表 4-15 JR 北海道バス時刻表データのサンプル・イメージ (translations.txt)

No.	日本語名称	必須	書式・選択肢など
1	テーブル名	○	<ul style="list-style-type: none"> ● 翻訳するフィールドを含む情報ファイルを指定 ● agency, stops, routes, trips, stop_times, feed_info
2	フィールド名	○	<ul style="list-style-type: none"> ● 翻訳するフィールド名を指定
3	言語	○	<ul style="list-style-type: none"> ● 日本語とよみがなを設定 ● ja:日本語 ● ja-Hrkt:よみがな
4	翻訳先言語	○	<ul style="list-style-type: none"> ● 翻訳先の文言 ● よみがなは、原則としてそのままの読みを記載
5	レコード ID	○	<ul style="list-style-type: none"> ● 翻訳対象のフィールドに対応する ID を設定 ● agency - agency_id ● stops - stop_id ● routes - route_id ● trips - trip_id ● stop_times - trip_id
6	レコードサブ ID	○	<ul style="list-style-type: none"> ● record_id で参照されている情報ファイルに一意的 ID が存在しない場合、フィールドを含む項目を翻訳する際に使用 ● stop_times - stop_sequence

- 本データの形式
 - CSV
- 出所
 - 札幌市
- 変換方法（処理内容・手順）
 - データベースから出力された生データを CSV として取得し、python を用いて変換

【DT005】 中央バス時刻表データ

- 本データの概要
 - 実証エリア近傍を走行する定時定路線バスである、北海道中央バスの時刻表データ
 - GTFS-JP 第 3 版に従い、以下の項目を設定した GTFS Schedule データへ変換して使用
- データ定義
 - 【DT004】 JR 北海道バス時刻表データと同様
- 本データの形式
 - CSV
- 出所
 - 札幌市
- 変換方法（処理内容・手順）
 - データベースから出力された生データを CSV として取得し、python を用いて変換

【DT006】 じょうてつバス時刻表データ

- 本データの概要
 - 実証エリア近傍を走行する定時定路線バスである、じょうてつバスの時刻表データ
 - GTFS-JP 第3版に従い、以下の項目を設定した GTFS Schedule データへ変換して使用
- データ定義
 - 【DT004】 JR 北海道バス時刻表データと同様
- 本データの形式
 - CSV
- 出所
 - 札幌市
- 変換方法（処理内容・手順）
 - データベースから出力された生データを CSV として取得し、python を用いて変換

【DT007】 実証用オンデマンドバスデータ

- 本データの概要
 - 実証エリアを走行するオンデマンドバスのサービスエリアや乗降ポイントを設定したデータ
 - GTFS-Ondemand によって拡張された GTFS-Flex データとして作成
- データ定義

表 4-16 実証用オンデマンドバス静的データのサンプル・イメージ (agency.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	事業者 ID	○	● 事業者 (TIS) の法人番号を設定	-
2	事業者名称	○	● 事業者 (TIS) の名称を設定	-
3	事業者 URL	○	● 事業者 (TIS) HP のトップページの URL を設定	-
4	タイムゾーン	○	● 「Asia/Tokyo」を設定	-
5	言語	○	● 「ja」を設定	-
6	電話番号	-	● 事業者 (TIS) の窓口電話番号を設定	-

表 4-17 実証用オンデマンドバス静的データのサンプル・イメージ (stops.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	停留所・標柱 ID	○	● オンデマンドバスの乗降ポイントを識別する一意な ID を指定 ● stops.stop_id、location_groups.location_group_id、location.geojesson の id を通じて一意でなければならない	一意性の制約の追加は Flex
2	停留所・標識名称	○	● オンデマンドバスの乗降ポイント名を設定	-
3	緯度	○	● オンデマンドバスの乗降ポイントの緯度を設定	-
4	経度	○	● オンデマンドバスの乗降ポイントの経度を設定	-

表 4-18 実証用オンデマンドバス静的データのサンプル・イメージ (routes.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	経路 ID	○	● 一意な ID を設定	-
2	事業者 ID	○	● agency から参照	-
3	経路略称	-	● 経路名 (route_long_name) を設定するため使用しない	-
4	経路名	○	● 本実証名を設定	-
5	連続乗車	-	● 「1」(利用しない) を指定	Flex
6	連続降車	-	● 「1」(利用しない) を指定	Flex
7	経路タイプ	○	● 「13」(自動車) を設定	Ondemand
8	予約ディープリンク ID	-	● オンデマンドサービスのアプリを開くために使用するディープリンクを示す ID	Ondemand

表 4-19 実証用オンデマンドバス静的データのサンプル・イメージ (trips.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	経路 ID	○	● routes から参照	-
2	運行日 ID	○	● calendar から参照	-
3	便 ID	○	● 便を特定する ID を指定	-
4	便タイプ	-	<ul style="list-style-type: none"> ● 乗合されるか否かを指定 ● 0 又は空:乗合 ● 1:プライベート ● 本実証では「0」を指定 	Ondemand
5	荷物可否	-	<ul style="list-style-type: none"> ● 乗客が荷物を持って車両に乗車できるかどうかを示す ● 0 又は空:情報が利用できない ● 1:荷物を持って乗車可 ● 2:荷物を持って乗車不可 ● 本実証では「1」を指定 	Ondemand
6	ペット可否	-	<ul style="list-style-type: none"> ● 客がペットと一緒に車両に乗車できるかどうかを示す ● 0 又は空:情報が利用できない ● 1:ペットを連れて乗車可 ● 2:ペットを連れて乗車不可 ● 3:ペットを連れて乗車できるが、キャリーバッグに入れる必要がある ● 本実証では「2」を指定 	Ondemand
7	介助動物可否	-	<ul style="list-style-type: none"> ● 乗客が介助動物と一緒に車両に乗車できるかどうかを示す ● 0 又は空:情報が利用できない ● 1:介助動物を連れて乗車可 ● 2:介助動物を連れて乗車不可 ● 3:解除動物を連れて乗車できるが、キャリーバッグに入れる必要がある ● 本実証では「2」を指定 	Ondemand
8	有料旅程可否	-	<ul style="list-style-type: none"> ● 有料の旅程（高速料金やフェリー料金等）を使用できるかどうかを定義する ● 0 又は空:情報が利用できない ● 1:有料の旅程を使用する場合がある ● 2:有料の旅程は使用しない ● 本実証では「2」を指定 	Ondemand

表 4-20 実証用オンデマンドバス静的データのサンプル・イメージ (stop_times.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	便 ID	○	● trips から参照	-
2	停留所グループ ID	※2	● location_groups から参照	Flex
3	ロケーション ID	※2	● locations.geojson から参照	Flex
4	通過順位	○	● 便 ID と通過順位を組み合わせて主キーとなるように設定	-
5	オンデマンドサービス開始時間	○	● この場所でオンデマンドサービスの利用が開始する時間 HH:MM:SS 形式	Flex
6	オンデマンドサービス終了時間	○	● この場所でオンデマンドサービスの利用が終了する時間 HH:MM:SS 形式	Flex
7	乗車タイプ	-	● 乗車方法を指定 ● 0 又は空:定時定路線での乗車 ● 1:乗車不可 ● 2:電話やアプリによって代理店に乗車の手配が必要 ● 3:ドライバーと調整して乗車の手配が必要 ● 4:代理店あるいはドライバーのどちらかと乗車の手配が必要 ● 本実証では「1」あるいは「2」を設定	Flex アプリ手配については Ondemand
8	降車タイプ	-	● 降車方法を指定 ● 0 又は空:定時定路線での降車 ● 1:降車不可 ● 2:電話やアプリによって代理店に降車の手配が必要 ● 3:ドライバーと調整して降車の手配が必要 ● 4:代理店あるいはドライバーのどちらかと降車の手配が必要 ● 本実証では「1」あるいは「2」を設定	Flex アプリ手配については Ondemand
9	連続乗車	-	● 「1」(利用しない)を指定	Flex
10	連続降車	-	● 「1」(利用しない)を指定	Flex
11	乗車ルール ID	-	● booking_rules から参照する乗車ルールの ID	Flex
12	降車ルール ID	-	● booking_rules から参照する降車ルールの ID	Flex
13	乗車近接レベル	-	● リクエスト場所から乗車場所までの近接レベルを定義する ● 0 又は空:情報が利用できない ● 1:リクエスト場所にできるだけ近い場所で乗車 ● 2:リクエスト場所から少しの距離の場所(近い交差点等)で乗車する場合がある	Ondemand

			<ul style="list-style-type: none"> ● stop_id が GeoJSON を参照する場合のみ利用可能 	
14	降車近接レベル	-	<ul style="list-style-type: none"> ● リクエスト場所から降車場所までの近接レベルを定義する ● 0 又は空:情報が利用できない 1:リクエスト場所にできるだけ近い場所で降車 ● 2:リクエスト場所から少しの距離の場所(近い交差点等)で降車する場合がある ● stop_id が GeoJSON を参照する場合のみ利用可能。 	Ondemand
15	平均待ち時間	-	<ul style="list-style-type: none"> ● stop_time で乗客が待つ平均待ち時間 (分) ● stop_id が GeoJSON をあるいは location_group を参照し、pickup_type=2 あるいは 4 の場合のみ利用可能 	Ondemand
16	安全待ち時間	-	<ul style="list-style-type: none"> ● stop_time で乗客が待つ安全な待ち時間 (分) ● 安全とは、過去のデータに基づいた 95 パーセンタイル値の待ち時間を意味する ● stop_id が GeoJSON あるいは location_group を参照し、pickup_type=2 あるいは 4 の場合のみ利用可能 	Ondemand
17	最大待ち時間	-	<ul style="list-style-type: none"> ● 乗客が待つ最大待ち時間 (分)。最大待ち時間はサービスプロバイダの SLA に基づいて定義される ● stop_id が GeoJSON あるいは location_group を参照し、pickup_type=2 あるいは 4 の場合のみ利用可能 ● 本実証では SAVS の迂回許容時間を指定 	Ondemand
18	待機ルール ID	-	<ul style="list-style-type: none"> ● wait_rule_id が定義されている場合、その wait_rules の値で mean_wait_time、safe_wait_time、max_wait_time は上書きされる ● stop_id が GeoJSON あるいは location_group を参照し、pickup_type=2 あるいは 4 の場合のみ利用可能 	Ondemand

※2 停留所グループ ID とロケーション ID のどちらかは必ず設定する

表 4-21 実証用オンデマンドバス静的データのサンプル・イメージ (calendar.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	運行日 ID	○	● 運行区分を表す値を設定	-
2	月曜日	○	● 0:月曜日に非運行 1:月曜日に運行	-
3	火曜日	○	● 0:火曜日に非運行 1:火曜日に運行	-
4	水曜日	○	● 0:水曜日に非運行 1:水曜日に運行	-
5	木曜日	○	● 0:木曜日に非運行 1:木曜日に運行	-
6	金曜日	○	● 0:金曜日に非運行 1:金曜日に運行	-
7	土曜日	○	● 0:土曜日に非運行 1:土曜日に運行	-
8	日曜日	○	● 0:日曜日に非運行 1:日曜日に運行	-

9	サービス開始日	○	<ul style="list-style-type: none"> ● 適用を開始する日付を指定 ● YYYYMMDD 形式 	-
10	サービス終了日	○	<ul style="list-style-type: none"> ● 適用を終了する日付を指定 ● YYYYMMDD 形式 	-

表 4-22 実証用オンデマンドバス静的データのサンプル・イメージ (location_groups.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	停留所グループ ID	○	<ul style="list-style-type: none"> ● 停留所グループを識別する一意な ID を指定 ● stops.stop_id、location.geojeson の id、location_groups.location_group_id を通じて一意とならなければならない 	Flex
2	停留所グループ名	-	<ul style="list-style-type: none"> ● 停留所グループ名を設定 	Flex

表 4-23 実証用オンデマンドバス静的データのサンプル・イメージ (location_group_stops.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	停留所グループ ID	○	<ul style="list-style-type: none"> ● location_groups から参照 	Flex
2	停留所・標柱 ID	○	<ul style="list-style-type: none"> ● stops から参照 ● この location_group に属する乗降ポイントを意味する 	Flex

表 4-24 実証用オンデマンドバス静的データのサンプル・イメージ (locations.geojson)

No.	フィールド名	必須	書式・選択肢など	拡張規格
1	type	○	● 「FeatureCollection」を指定	Flex
2	features[*].id	○	● エリアを識別する一意な ID を指定 ● stops.stop_id、location.geojson の id、location_groups.location_group_id を通じて一意とならなければならない	Flex
3	features[*].type	○	● 「Feature」を指定	Flex
4	features[*].properties	○	● エリアの属性を指定するオブジェクト	Flex
5	features[*].properties.stop_name	-	● エリアの名前を設定	Flex
6	features[*].properties.stop_desc	-	● エリアの説明を設定	Flex
7	features[*].geometry	○	● エリアの形状を指定するオブジェクト	Flex
8	features[*].geometry.type	○	● エリアが一つの領域で囲まれている場合は Polygon、エリアが複数の領域に分割されている場合は MultiPolygon を指定	Flex
9	features[*].geometry.coordinates	○	● Polygon を構成する経度・緯度の配列を設定 ● MultiPolygon の場合は、Polygon を示す経度・緯度の配列を設定する	Flex

locations.geojson の例

```
{
  "type": "FeatureCollection",
  "features": [
    {
      "id": "area_001",
      "type": "Feature",
      "geometry": {
        "type": "Polygon",
        "coordinates": [
          [
            141.35417852542855, 43.07316565788088
```

```

    ],
    [
      141.37459507543088, 43.07581484460556
    ],
    [
      141.376842259334, 43.065569033569794
    ],
    [
      141.368860882023, 43.06228548314663
    ],
    [
      141.3648911482074, 43.05722798132312
    ],
    [
      141.35849676175533, 43.056600837195234
    ]
  ]
],
"properties": {}
}
]
}

```

表 4-25 実証用オンデマンドバス静的データのサンプル・イメージ (booking_rules.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	予約ルール ID	○	● 予約ルールを識別する ID	Flex
2	予約タイプ	○	● どの程度前から予約できるかを指定 ● 0:リアルタイム予約 ● 1:事前通知による当日予約まで ● 2:前日予約まで	Flex
3	事前通知最小期間	-	● booking_type=1 の場合に最小で何分前までに予約が必要か	Flex
4	事前通知最大期間	-	● booking_type=1 の場合に最大で何分前から予約ができるか	Flex
5	事前通知	-	● booking_type=2 の場合に何日前まで予約ができるか	Flex

	最終日			
6	事前通知 最終時間	-	<ul style="list-style-type: none"> ● booking_type=2 の場合に、prior_notice_last_day の何時まで予約ができるか 	Flex
7	事前通知 開始日	-	<ul style="list-style-type: none"> ● 何日前から予約ができるか ● booking_type=0 の場合は禁止 ● booking_type=1 で prior_notice_duration_max が定義されている場合は禁止 	Flex
8	事前通知 開始時間	-	<ul style="list-style-type: none"> ● prior_notice_start_day の何時から予約ができるか 	Flex
9	事前通知 サービス日	-	<ul style="list-style-type: none"> ● prior_notice_last_day 又は prior_notice_start_day をカウントするサービス日を示す ● booking_type=0 や 1 では禁止 	Flex
10	メッセージ	-	<ul style="list-style-type: none"> ● オンデマンドサービスを予約するときに示されるサービスを利用する乗客へのメッセージ 	Flex
11	電話番号	-	<ul style="list-style-type: none"> ● 予約するために電話をかける電話番号 	Flex
12	予約情報 URL	-	<ul style="list-style-type: none"> ● 予約ルールに関する情報を提供する URL 	Flex
13	予約アプリ URL	-	<ul style="list-style-type: none"> ● 予約リクエストを行うことができるインターネット予約システムの URL (Ondemand 拡張にてアプリへの URL は禁止となった) 	Ondemand
14	アプリダウンロード URL	-	<ul style="list-style-type: none"> ● 予約リクエストを行うことができるへのアプリへの URL (Web サイトやアプリマーケット) 	Ondemand

表 4-26 実証用オンデマンドバス静的データのサンプル・イメージ (wait_rules.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	待機ルール ID	○	<ul style="list-style-type: none"> ● 待機ルールを識別する一意な ID を指定 	Ondemand
2	停留所・標柱 ID	○	<ul style="list-style-type: none"> ● 待機ルールが適用される stops.stop_id、location.geojeson の id、location_groups.location_group_id を示す 	Ondemand
3	運行日 ID	-	<ul style="list-style-type: none"> ● 待機ルールが適用されるカレンダーを示す 	Ondemand
4	適用開始時間	-	<ul style="list-style-type: none"> ● 待機ルールの適用が開始される時間を示す ● (stop_times.start_pickup_dropoff_window、stop_times.end_pickup_dropoff_window) 間隔内でなければならない ● このフィールドが指定されていない場合、待機ルールは (stop_times.start_pickup_dropoff_window、stop_times.end_pickup_dropoff_window) 間隔内のいつでも適用される 	Ondemand

			<ul style="list-style-type: none"> ● 本実証では実証開始時間を指定 	
5	適用終了時間	-	<ul style="list-style-type: none"> ● 待機ルールの適用が終了される時間を示す ● (stop_times.start_pickup_dropoff_window、stop_times.end_pickup_dropoff_window) 間隔内ではない ● このフィールドが指定されていない場合、待機ルールは (stop_times.start_pickup_dropoff_window、stop_times.end_pickup_dropoff_window) 間隔内のいつでも適用される ● 本実証では実証終了時間を指定 	Ondemand
6	平均待ち時間	-	<ul style="list-style-type: none"> ● stop_time で乗客が待つ平均待ち時間 (分) ● このフィールドに値が入力されると、stop_times.mean_wait_time の値がオーバーライドされる ● safe_wait_time 及び max_wait_time が指定されていない場合は必須 	Ondemand
7	安全待ち時間	-	<ul style="list-style-type: none"> ● stop_time で乗客が待つ安全な待ち時間 (分)。安全とは、過去のデータに基づいた 95 パーセンタイル値の待ち時間を意味する ● このフィールドに値が入力されると、stop_times.safe_wait_time の値がオーバーライドされる ● mean_wait_time 及び max_wait_time が指定されていない場合は必須 	Ondemand
8	最大待ち時間	-	<ul style="list-style-type: none"> ● stop_time で乗客が待つ最大待ち時間 (分) ● 最大待ち時間はサービスプロバイダの SLA に基づいて定義される ● このフィールドに値が入力されると、stop_times.max_wait_time の値がオーバーライドされる ● mean_wait_time 及び safe_wait_time が指定されていない場合は必須 	Ondemand

表 4-27 実証用オンデマンドバス静的データのサンプル・イメージ (booking_deep_links.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	予約ディープリンク ID	○	<ul style="list-style-type: none"> ● オンデマンドサービスのアプリを開くために使用するディープリンクを示す ID 	Ondemand
2	Android URI	-	<ul style="list-style-type: none"> ● Android のディープリンクをサポートするために Android の Intent と共に Android アプリに渡される URI ● Android App Links が推奨される 	Ondemand
3	iOS URI	-	<ul style="list-style-type: none"> ● iOS で利用されるディープリンクの URI ● iOS Universal Links が推奨される 	Ondemand
4	Web URL	-	<ul style="list-style-type: none"> ● ウェブブラウザで利用される URL 	Ondemand

表 4-28 実証用オンデマンドバス静的データのサンプル・イメージ (fare_leg_rules.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	運賃区間 ID	-	<ul style="list-style-type: none"> ● 運賃区間を示す ID 	Ondemand
2	運賃	-	<ul style="list-style-type: none"> ● 運賃区間の運賃を示す ● 運賃変動ルールがこの区間に割当てられている場合（つまり、行に variable_group_id が定義されている場合）、運賃の固定価格（又は開始価格）として機能する 	Ondemand
3	最大料金	-	<ul style="list-style-type: none"> ● 開始料金と変動料金を合算した乗客が支払う最大料金 	Ondemand
4	最低料金	-	<ul style="list-style-type: none"> ● 開始料金と変動料金を合算した乗客が支払う最低料金 	Ondemand
5	運賃調整可否	-	<ul style="list-style-type: none"> ● 運行状況（例：需要の急増）に対応するために運賃調整が行われる可能性があるかどうかを示す ● 0 (又は空白):情報がない ● 1:運賃調整が行われる可能性がある ● 2:運賃調整は行われない 	Ondemand
6	税込み/税抜き	-	<ul style="list-style-type: none"> ● 税金が含まれるかどうかを示す ● 0 (又は空白):情報がない ● 1:税金が含まれる ● 2:税金は含まれない 	Ondemand
7	チップ	-	<ul style="list-style-type: none"> ● チップを受け取るかどうかを示す。 ● 0 (又は空白):情報がない ● 1:チップを受け取ることができる ● 2:チップは受け取らない 	Ondemand
8	変動運賃グループ ID	-	<ul style="list-style-type: none"> ● 変動運賃グループの ID 	Ondemand

表 4-29 実証用オンデマンドバス静的データのサンプル・イメージ (fare_variable_rules.txt)

No.	日本語名称	必須	書式・選択肢など	拡張規格
1	変動料金 ID	-	<ul style="list-style-type: none"> ● 変動料金の設定を識別する ID 	Ondemand
2	変動料金グループ ID	-	<ul style="list-style-type: none"> ● 変動料金のグループを識別する ID 	Ondemand
3	変動料金種別	○	<ul style="list-style-type: none"> ● 変動料金の基準となる変数項目 ● 0:キロメートル (距離精算) ● 1:分 (時間精算) ● 2:アクティブ分 (車両が走行している時間で精算) ● 3:アイドル分 (車両が停止している時間で精算) ● 4:乗客数 ● 5:荷物 ● 6:ペット 	Ondemand
4	変動料金間隔	-	<ul style="list-style-type: none"> ● 変動料金が適用される間隔 ● interval が 0 の場合、1 回のみ課金される ● fare_variable_type が 0, 1, 2, 3 の場合は必須 ● それ以外は禁止 	Ondemand
5	課金開始値	-	<ul style="list-style-type: none"> ● 変動料金の適用が開始される値 ● このフィールドが空の場合、乗車開始と同時に金額が請求される 	Ondemand
6	課金終了値	-	<ul style="list-style-type: none"> ● 変動料金の適用が終了する値 ● このフィールドが空の場合、乗車が終了するまで金額が請求される 	Ondemand
7	単位当たり運賃	-	<ul style="list-style-type: none"> ● 単位当たりの変動料金 ● fare_variable_rules.min_amount と fare_leg_rules.max_amount が空の場合は必須 ● fare_variable_rules.min_amount と fare_leg_rules.max_amount が定義されている場合は禁止 	Ondemand
8	通貨	-	<ul style="list-style-type: none"> ● 運賃の通貨を指定する ● fare_variable_rules.amount、fare_variable_rules.min_amount、fare_variable_rules.max_amount のいずれかが定義されている場合は必須 ● それ以外は禁止 	Ondemand
9	運賃調整可否	-	<ul style="list-style-type: none"> ● 運行状況 (例:需要の急増) に対応するために運賃調整が行われる可能性があるかどうかを示す ● 0 (又は空白):情報がない ● 1:運賃調整が行われる可能性がある 	Ondemand

			<ul style="list-style-type: none"> ● 2:運賃調整は行われたい 	
10	税込み/税抜き	-	<ul style="list-style-type: none"> ● 税金が含まれるかどうかを示す ● 0 (又は空白):情報がない ● 1:税金が含まれる ● 2:税金は含まれない 	Ondemand

- 本データの形式
 - CSV (locations.geojson のみ GeoJSON 形式)
- 出所
 - 本実証用に作成

5. 用語集

用語	定義・説明
GTFS-Flex	デマンドレスポンス形式の公共交通の発見可能性を促進することを目的に、GTFS Schedule を拡張した仕様
GTFS-Ondemand	オンデマンド交通への対応を目的に、GTFS-Flex や GTFS Realtime を拡張する仕様案
GTFS Realtime	公共交通事業者のデータのうち、現在の到着時刻や出発時刻、車両位置、サービスアラートといった動的な情報を記述する標準フォーマット
GTFS Schedule	公共交通事業者のデータのうち、停留所や路線系統、便、時刻表といった定時定路線の静的な情報を記述する標準フォーマット
オンデマンドバス	利用者からの配車予約と車両位置から AI を用いて最適な運行ルートを決定し、利用者が乗合しつつもおおむね希望どおりに移動ができるバス
定時定路線交通	鉄道や路線バスに代表される、事前に定められた時刻表に従い、事前に定められた経路を走行する公共交通



Project LINKS



COMmmONS
By MLIT

**GTFS-Flex/Ondemand活用システム システム設計書
Ver1.0**

発行日: 2026年3月

委託者: 国土交通省 総合政策局
情報政策課、モビリティサービス推進課

受託者: TIS株式会社