

# シェアサイクル ポート共有API標準仕様書 ガイダンス



2026年2月  
国土交通省 総合政策局 公共交通政策部門 モビリティサービス推進課

## 文書管理情報

項目	内容
文書名称	シェアサイクルポート共有API標準仕様書ガイダンス
文書ID	commmmons_doc_002
発行元	国土交通省 総合政策局 公共交通政策部門 モビリティサービス推進課
最新版数	v1.0
制定年月	2026年2月
適用範囲	シェアサイクル事業者間でのポート共有と利用・精算データの連携
備考	本仕様書は2025年度の地域交通DX推進プロジェクト「COMmmmons」における「モビリティ・データ標準化プロジェクト」により作成されました。

## 改訂履歴

版数	年月日	改訂内容
v1.0	2026/02/13	初版制定
v1.1	2026/03/23	「1.6. API標準化がもたらす便益」 実証の結果を反映  「3.5. 業務フロー/業務一覧」 業務フロー（v1.1）の更新をFL5、FL6へ反映

# 目次

## 1. プロジェクトの概要

- 1.1. はじめに
- 1.2. シェアサイクルサービスについて
- 1.3. シェアサイクルサービスにおける現状課題
- 1.4. シェアサイクルサービスのAPI標準化による課題解決
- 1.5. API標準化のプロセス
- 1.6. API標準化がもたらす便益
- 1.7. 標準ドキュメントの種類

## 2. 標準ドキュメントの読み方

- 2.1. システムアーキテクチャの読み方
- 2.2. APIの読み方
- 2.3. ERDの読み方
- 2.4. 業務フロー/業務一覧の読み方

## 3. 標準ドキュメントの解説

- 3.1. 標準化の範囲と概要
- 3.2. システムアーキテクチャの解説
- 3.3. APIの解説
- 3.4. ERDの解説
- 3.5. 業務フロー/業務一覧(抜粋)の解説
- 3.6. ポート共有で得た収益を互いに清算する



# 1. プロジェクトの概要

---

- 1.1. はじめに
- 1.2. シェアサイクルサービスについて
- 1.3. シェアサイクルサービスにおける現状課題
- 1.4. シェアサイクルサービスのAPI標準化による課題解決
- 1.5. API標準化のプロセス
- 1.6. API標準化がもたらす便益
- 1.7. 標準ドキュメントの種類

## 1.1. はじめに

本ガイドスは、国土交通省が推進する地域交通DX推進プロジェクト「COMmmONS (コモンズ)」における、2025年度「シェアサイクルポート共有API標準化プロジェクト」の成果を解説するものです。

### プロジェクトの目的

国土交通省では、地域交通DX推進プロジェクト「COMmmONS (コモンズ)」として、2025年度に「シェアサイクルポート共有API標準プロジェクト」を実施しました。

このプロジェクトは、事業者ごとに分断されているシェアサイクルのポートネットワークやシステム仕様の標準化を通じ、事業者の垣根を越えたポートの相互利用（貸出・返却）を実現し、利用者にとっての移動の連続性と利便性の向上を図ることを目的としています。

具体的には、システム間連携に必要な標準APIや収益按分等の業務モデルを確立することで、限られた都市空間におけるポート確保競争によるコスト増を抑制するとともに、事業者間のシステム連携コストを削減し、持続可能な地域交通の構築やデータ活用による高度な政策立案が可能な環境の実現を目指します。

### 本ガイドスの構成

本ガイドスは、プロジェクトの背景と意義を説明する「プロジェクトの概要（第1章）」、技術資料の読み方を解説する「標準ドキュメントの読み方（第2章）」、標準仕様の内容を説明する「標準ドキュメント（第3章）」の3章により構成されています。

1章

2章

3章

プロジェクトの概要

標準ドキュメントの読み方

標準ドキュメント

本プロジェクトの目的や標準化の意義を説明

標準ドキュメントの読み方を解説

標準ドキュメントの内容を説明

### 本ガイドスの対象読者

本ガイドスは、ポート共有モデルの導入を検討されているシェアサイクルアプリ事業者様や自治体の交通政策担当者様、シェアサイクルシステムとの連携を検討されているMaaSアプリ事業者様や交通事業者様に向けた資料となります。

想定読者



交通事業者様



自治体関係者様



開発担当者様

## 1.2. シェアサイクルサービスについて

シェアサイクルサービスとは、街中に設置された複数のサイクルポートで自転車を借りたり返したりできるサービスです。利用者は専用アプリやカードを用いて、好きな場所で自転車を借り、目的地近くのポートに返却することができます。

### サービスの特徴

街中に設置された複数の拠点（ポート）において、自転車の貸出・返却ができるサービスです。従来のレンタサイクルと異なり、「借りた場所とは異なるポートへ返却できる」点が特徴であり、通勤・観光・買い物等における柔軟な移動手段として活用されます。

### 利用手順

利用は専用アプリを用い、以下の手順で行います。

登録：アプリにて利用者情報および決済方法を登録する。

選択：アプリでポートを検索し、希望車両を予約する。

貸出：現地で解錠して利用を開始する。返却時は目的地のポートへ駐輪・施錠する。

1

登録

2

選択

3

貸出



### 活用メリット

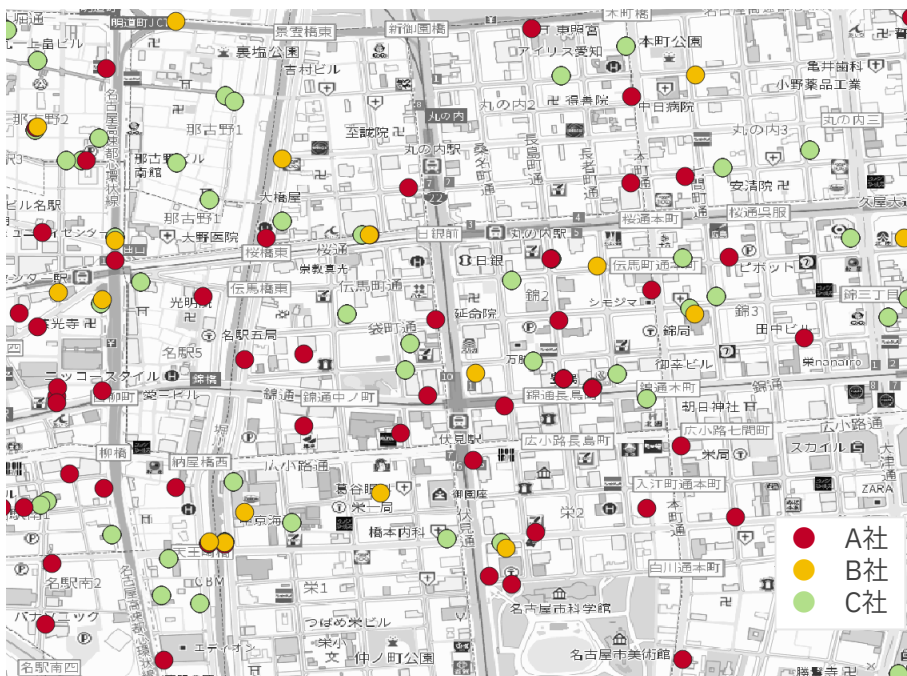
利用者へのメリット	移動の効率化	駅やバス停から最終目的地までの移動を円滑にする。
	快適性の向上	電動アシスト自転車により、坂道や長距離移動の負担が軽減される。また短時間の利用であればタクシー等と比較して安価である。
	利便性の確保	自転車を所有・管理する手間がなく、必要な時に即座に利用可能である。
地域へのメリット	公共交通網の補完	既存の交通機関ではカバーしきれない地域へのアクセスを向上させる。
	観光振興	観光地における周遊性を高め、地域の魅力発信に寄与する。
	環境負荷の低減	CO <sub>2</sub> を排出しない移動手段として、脱炭素社会の実現に貢献する。

### 1.3. シェアサイクルサービスにおける現状課題

シェアサイクルサービス等のマイクロモビリティのシェアリングサービスは公共交通を補完する手段として普及しつつあります。他方、公有地を利用したポート用地の不足や、事業者ごとにサービスやアプリが分断されていることによる利便性の低下が課題となっています。

#### 解決すべき課題

- 近年、マイクロモビリティのシェアリングサービスの普及が進みつつあり、従来の公共交通と組み合わせたラストワンマイルの補完や外出機会創出効果が注目されつつある。
- シェアサイクルサービスなど国内のシェアモビリティサービスは、貸出・返却のポート用地を定義する形式が原則であり、交通結節点や主要な施設などでポート用地を確保することが、シェアモビリティサービスの利便性や事業性の確保、公共交通の補完の観点から重要となる。
- 一方で、事業者ごとに構築したポートネットワークは展開地域やアプリごとに分断されており、地域をまたいだワンアプリでの利用が難しく、利便性を損なっている。
- また、交通政策等と連携して限られた公有地を利用するケースでは、ポート不足が課題となる場合もある。



#### 【課題】

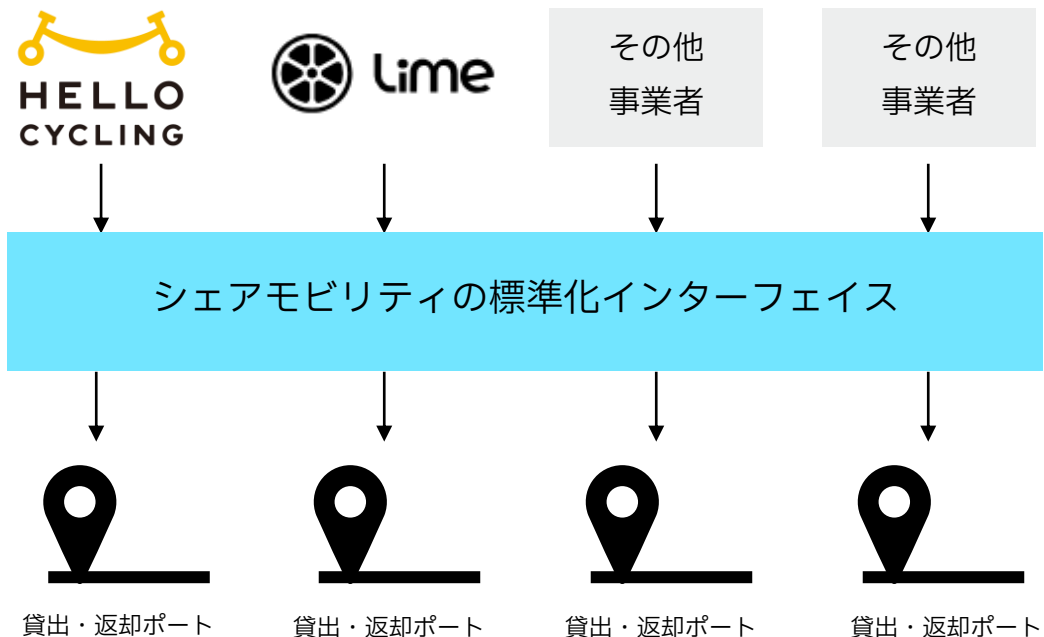
- ✓ 同じエリアに複数社がサービス展開しているため、複数のアプリを導入する必要がある。
- ✓ 同じ会社のポート同士でしか貸出・返却できず、利用者からのアクセシビリティに課題。
- ✓ 限られた公有地を有効活用できない。

## 1.4. シェアサイクルサービスのAPI標準化による課題解決

シェアモビリティの利便性と持続可能性を高めるため、ポート融通などを協調領域と定義し、リソースの相互利用に必要な標準システム連携インターフェースと売上配分等の事業モデルを開発します。これらを標準仕様として提供することで、事業者間のポート共有の仕組みを確立し、利用者がサービスを横断的に活用できる環境を実現します。

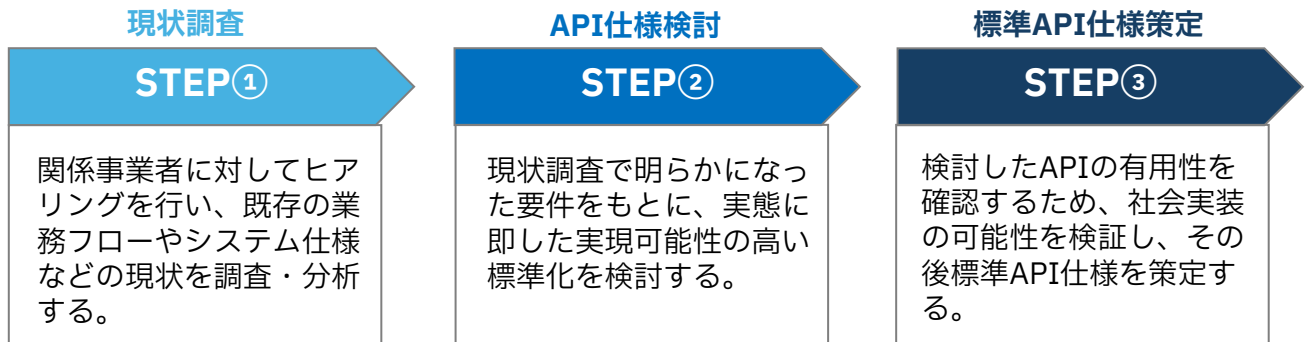
### 課題解決のアプローチ

- 用地確保やサービス品質はビジネスにおける競争領域としつつ、事業者のイニシアティブによる用地融通やアプリ連携のための技術仕様を協調領域として定義し、シェアモビリティサービスの利便性と持続可能性の向上を図る。
- このため、事業者各社が所有するリソース（ポート及びモビリティ）を事業者間で共有・相互利用するための標準的なシステムインターフェースを標準API仕様として開発する。
- 標準API仕様を運用するための売上のシェアを行うロジックや、メンテナンス等の運用オペレーションの情報を共有する業務手順を標準業務モデルとして定義する。
- 業務モデル及び標準API仕様を提供することで、複数事業者が相互にポート共有する仕組みを標準化し、利用者が複数サービスを横断的に活用できる環境を実現する。



## 1.5. API標準化までのプロセス

標準化は以下の3段階で推進します。まず「現状調査」で関係各社の既存業務やシステム仕様を把握します。次に、その仕様を基に標準的な業務モデルとAPI仕様を検討します。最後に有用性の検証を通して、コスト削減効果や技術的価値、社会実装の可能性などを踏まえた、標準API仕様を策定します。



### 現状調査対象企業

本プロジェクトにおいて、ヒアリングを実施した企業と選定理由は以下のとおりです。

#	業界	企業名	選定理由
1	シェアサイクルサービス	OpenStreet株式会社	業界シェア上位であるため
2		株式会社ドコモ・バイクシェア	業界シェア上位であるため
3		Ecobike株式会社	複数の地域でシェアサイクル事業を展開しているため
4		チャリチャリ株式会社	複数の地域でシェアサイクル事業を展開しているため
5		株式会社LUUP	業界シェア上位であるため

## 現状調査及び標準化の対象

本プロジェクトにおける標準化の対象は、業務の根幹となる「業務モデル」およびシステム連携に関わる「API仕様」としてしています。それぞれの領域における詳細な調査結果を踏まえ、実効性のある標準ドキュメントの作成を進めます。

### 業務モデル

#	調査項目名	主要論点	調査手法	調査アウトプット	獲得されるナレッジ
1	業務モデル	ステーション（ポート）を運用するにあたって、どのような作業が発生するか？	ステークホルダーヒアリング	標準業務フロー・標準業務一覧	ポート共同利用に関する技術的なフィジビリティスタディ（業務モデルの整理）
2		貸出・返却判定の方式は？		標準業務フロー・標準業務一覧 API仕様	
3		ポート共有を行う際に追加で必要となる作業は？		標準業務フロー・標準業務一覧	
4	アプリ及びバックエンド	ポートを検索する際に必要な情報は？	ステークホルダーヒアリング ドキュメントリサーチ	標準業務フロー・標準業務一覧 API仕様	ポート共同利用に関する技術的なフィジビリティスタディ（アプリ及びバックエンドの仕様整理）
5	情報共有方法	共有するポート情報の項目と共有方法は？		標準業務フロー・標準業務一覧	ポート共同利用に関する技術的なフィジビリティスタディ（情報共有方法の整理）
6	オペレーション分担方法	ポートを管理するためのオペレーションはどのようなものか？	ステークホルダーヒアリング	標準業務フロー・標準業務一覧	ポート共同利用に関する技術的なフィジビリティスタディ（オペレーション分担方法の整理）
7	按分・清算方法	按分・清算をするために必要な情報とシステムはどのようなものか？		標準業務フロー・標準業務一覧 API仕様	ポート共同利用に関する技術的なフィジビリティスタディ（按分・清算方法の整理）

### API仕様

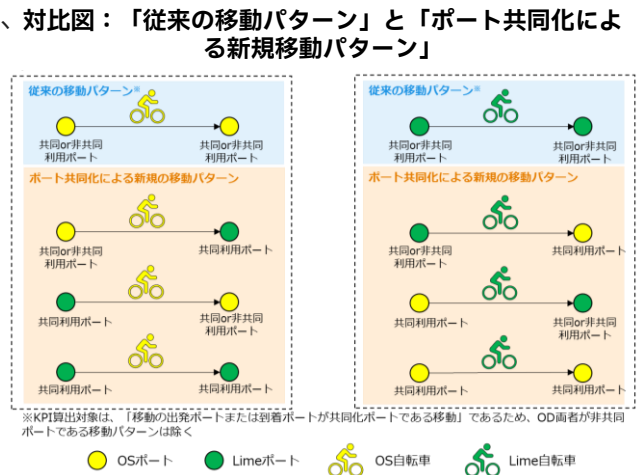
#	調査項目名	主要論点	調査手法	調査アウトプット	獲得されるナレッジ
1	ポート・モビリティ情報のAPI仕様	アプリにおけるポート情報を取得するための項目およびAPI仕様（リクエスト方式・リクエストパラメータ）は？	ステークホルダーヒアリング	API仕様	ポート共同利用に関する技術的なフィジビリティスタディ（API仕様の整理）
2		アプリにおける予約（モビリティ&返却ポート）情報を取得するための項目およびAPI仕様（リクエスト方式・リクエストパラメータ）は？			ポート共同利用に関する技術的なフィジビリティスタディ（API仕様の整理）
3	決済情報・利用ログのシェアサイクル	事業者間精算に必要な決済情報を取得するための項目およびAPI仕様（リクエスト方式・リクエストパラメータ）は？			ポート共同利用に関する技術的なフィジビリティスタディ（API仕様の整理）

## 1.6. API標準化がもたらす便益

これまでシェアサイクル事業者のアプリごとに分断されていたポートネットワークを、標準APIを介して相互接続します。これにより、事業者の垣根を越えたポート共有（相互貸出・返却）が可能となり、限られた都市空間の有効活用と、ユーザーにとっての移動体験の向上を実現します。

### 実証内容と仮説

- ポート共有に伴い移動の選択肢が増えることで、シェアサイクルサービスの利用が促進される。総ODペア数、総貸出・返却回数が実証前月比：1.0倍以上になる。
- 複数事業者による公共空間での協調運用を支える標準業務モデルが受容され、行政財産・公有地の有効活用に貢献する。ヒアリング先の8割以上の自治体から賛同を得る。
- ポート共有によって、利用者の利便性が向上する。アンケートを通じて、NPSスコア：0以上を確認する。
- 実用に耐えうる標準APIを開発できる。運用時にポート情報の共有エラーが発生しない（0件）。

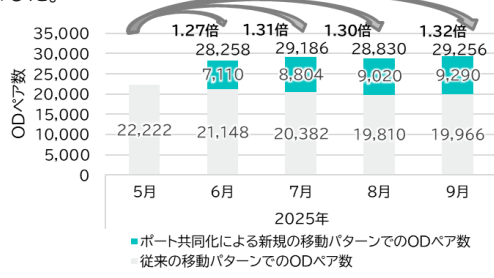


### 実証結果と便益

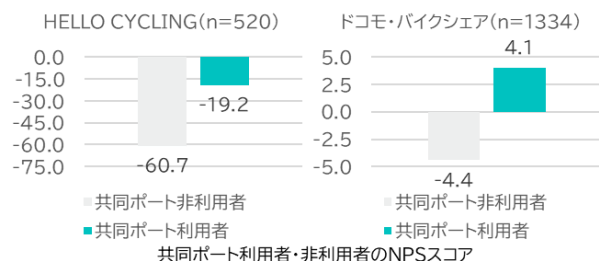
- ポート共有によって、事業者をまたいだポート利用が可能となり、総ODペア数は約1.3倍、総貸出・返却回数は約1.1倍の利用回数(※)の増加を確認しました。ポートの柔軟な活用により、サービス提供範囲の実質的な拡張効果が得られました。
- ヒアリングを通じて、ポート共有を導入されていない全て（10割）の自治体からポート共有への関心が示され、公有地の効率的活用や都市回遊性向上への効果が評価されました。複数事業者による公有地でのポート共有が、現実的な選択肢となることが確認されました。
- 利用者へのアンケート(※)を通じて、ポート共有の利用者は非利用者と比較して高いNPSスコアを記録しました。HELLO CYCLINGでは、共有ポート利用者のNPSスコアは-19.2(41.5向上)で目標を未達となりましたが、ドコモ・バイクシェアではNPSが4.1(8.5向上)を記録し目標を達成しました。
- 実証において、標準APIを利用したデータ連携が適切に機能し、エラーが発生しないことを確認しました。

※ポート共有後の利用回数の変化およびアンケート結果については、横浜市においてドコモ・バイクシェアとHELLO CYCLINGが実施したポート共同化の実証データを用いています。アンケートはドコモ・バイクシェアで2025年10月20日から2025年11月2日、HELLO CYCLINGは2025年11月17日から2025年11月30日の期間で実施しました。

利用者視点での共同ポート利用体験やポート共有の価値が共通であることから、ポート共有の効果検証を補完するデータとして活用しました。



ポート共同化前後における総ODペア数の推移(5月:共同化前、6月以降:共同化後)



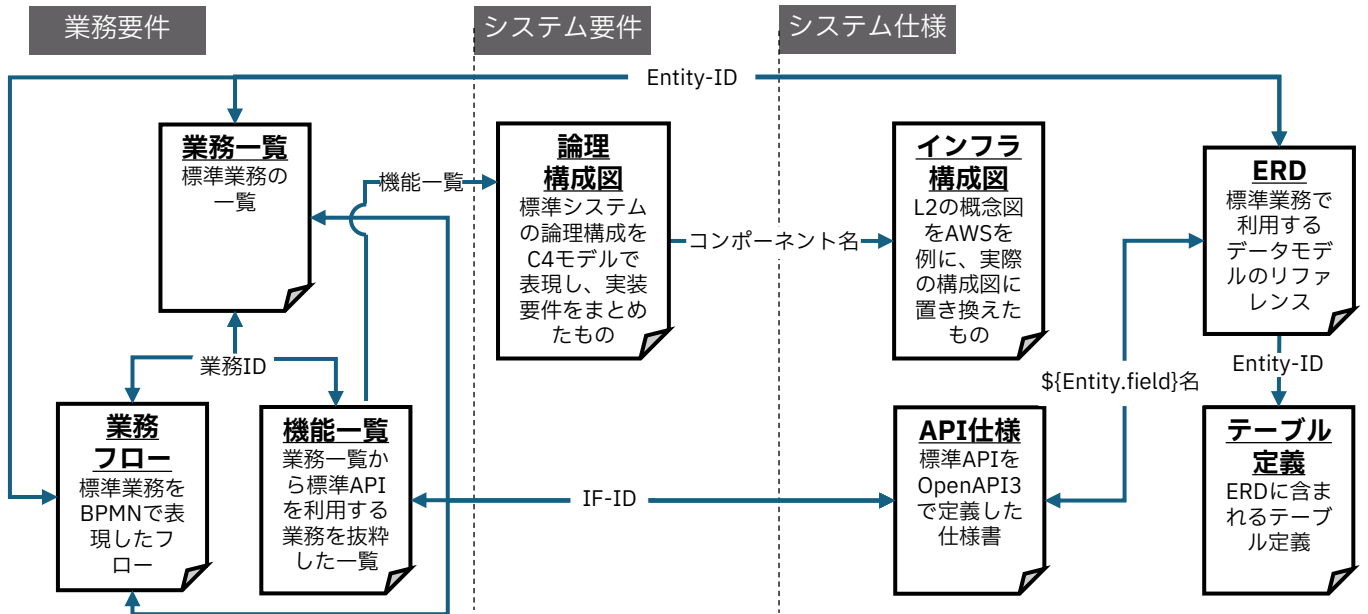
## 1.7. 標準ドキュメントの種類

標準ドキュメントでは、「業務要件覧」、「システム要件」、「システム仕様」の3種類で定義しています。

まず、システムが実現すべき「業務要件」を定義するのが「業務フロー/業務一覧/機能一覧」です。これを具現化する「システム要件」は、システムの全体像を示す「論理構成図」で表現されます。システム要件をさらに詳細化した「システム仕様」では、具体的なリファレンス実装を示す「インフラ構成図」と、システム間の連携インターフェースを示す「API」、データ構造やエンティティ間の依存関係を示す「ERD」で構成されます。

また各ドキュメントは、業務ID、IF-ID、Entity-IDなどの名称で機能ごとに採番され、各ドキュメント間で紐づけを行っております。

### 標準ドキュメントの種類



名称	内容	ファイル形式		想定読者
		参照用	Rawデータ	
シェアサイクルポート共有API標準仕様書_ガイダンス	標準化の意図、ビジネス上の意味など、仕様の背景や文脈（コンテキスト）の解説する。「標準ドキュメント」を読むための「前提知識」を提供する。	PDF	PPTX	
シェアサイクルポート共有API標準仕様書_業務一覧	業務フローで扱う業務内容をリスト化し説明する。IDにより他資料との紐づきを管理する	PDF	XLSX	交通事業者 自治体担当者 開発担当者
シェアサイクルポート共有API標準仕様書_業務フロー	抽象度の高い事業機能から、現場の作業手順までを体系化したもの。APIを定義するにあたり、システムが実現すべき業務の流れを説明する。	PDF	drawio	
シェアサイクルポート共有API標準仕様書_機能一覧	業務一覧から標準APIを利用する業務を抜粋した一覧	PDF	XLSX	
シェアサイクルポート共有API標準仕様書_論理構成図	APIを実装するための、システムの全体像を概念的に説明する。	PDF	drawio	開発担当者
シェアサイクルポート共有API標準仕様書_インフラ構成図	論理構成図のシステム部分を実際の実装例を踏まえて詳細化して説明する。	PDF	drawio	
シェアサイクルポート共有API標準仕様書_API	システム間の連携インターフェース仕様を説明する。この仕様を推奨として活用促進する。	HTML	yaml	
シェアサイクルポート共有API標準仕様書_ERD	APIが保持すべきデータの構造を定義する。	PDF	mermaid	
シェアサイクルポート共有API標準仕様書_テーブル定義	ERDで定義されたデータを実際のデータベースに落とし込むための仕様。	PDF	markdown	

※ドキュメントのダウンロードURL(<https://www.mlit.go.jp/commmons/document/002/>)



## 2. 標準ドキュメントの読み方

---

- 2.1. システムアーキテクチャの読み方
- 2.2. APIの読み方
- 2.3. ERDの読み方
- 2.4. 業務フロー/業務一覧の読み方



## 2.1. システムアーキテクチャの読み方

システムアーキテクチャでは、システムの論理構成およびインフラ実装の参照モデルを定義しています。論理構成には「C4モデル」を採用し、全体像から詳細への階層的な可視化を通じて関係者間の共通理解を形成します。また、インフラ構成ではAWS Lambda等を活用したサーバレス構成をリファレンスとして例示し、設計指針を示します。

### 論理構成図の説明

「Level 1 (System Context)」はビジネス視点で各プレイヤーとシステムの連携関係を示し、「Level 2 (Container)」はそのシステム内部を機能単位(コンテナ)に分解した技術視点の図です。

#### 階層の意味




Level 1 (System Context)

- 視点: 誰が (アクター)、どのシステムと連携するか。
- 目的: プロジェクトの境界線と、外部システムとの関係性を把握する。







Level 2 (Container)

- 視点: システムの中身がどのような技術要素 (アプリ、API、DB) で構成されているか。
- 目的: ソフトウェアの責務分担とデータの保存場所を把握する。

#### 記号の意味

	<p>人型 (Person):</p> <ul style="list-style-type: none"><li>➤ システムを利用するユーザー、または事業者 (操作の主体)。</li></ul>
	<p>四角 (Software System / Container):</p> <ul style="list-style-type: none"><li>➤ Level 1では: 1つの大きなシステム全体。</li><li>➤ Level 2では: その中にある「Webアプリ」「APIサーバー」「モバイルアプリ」などの実行単位。</li></ul>
	<p>円筒 (Database):</p> <ul style="list-style-type: none"><li>➤ データが保存される場所。</li></ul>

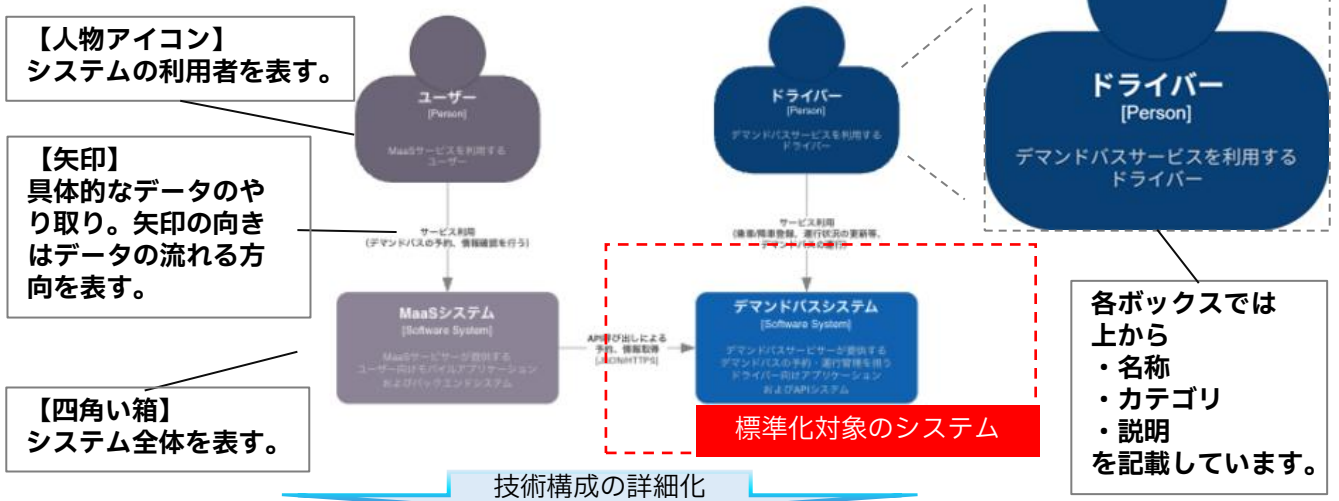
#### 色の意味

	-----➤ 人物(IF標準化対象システム利用)
	-----➤ ソフトウェアシステム(IF標準化対象)
	-----➤ コンテナ(IF標準化対象)
	-----➤ コンテナ(IF標準化対象外)
	-----➤ 人物(IF標準化対象外システム利用)
	-----➤ ソフトウェアシステム(IF標準化対象外)

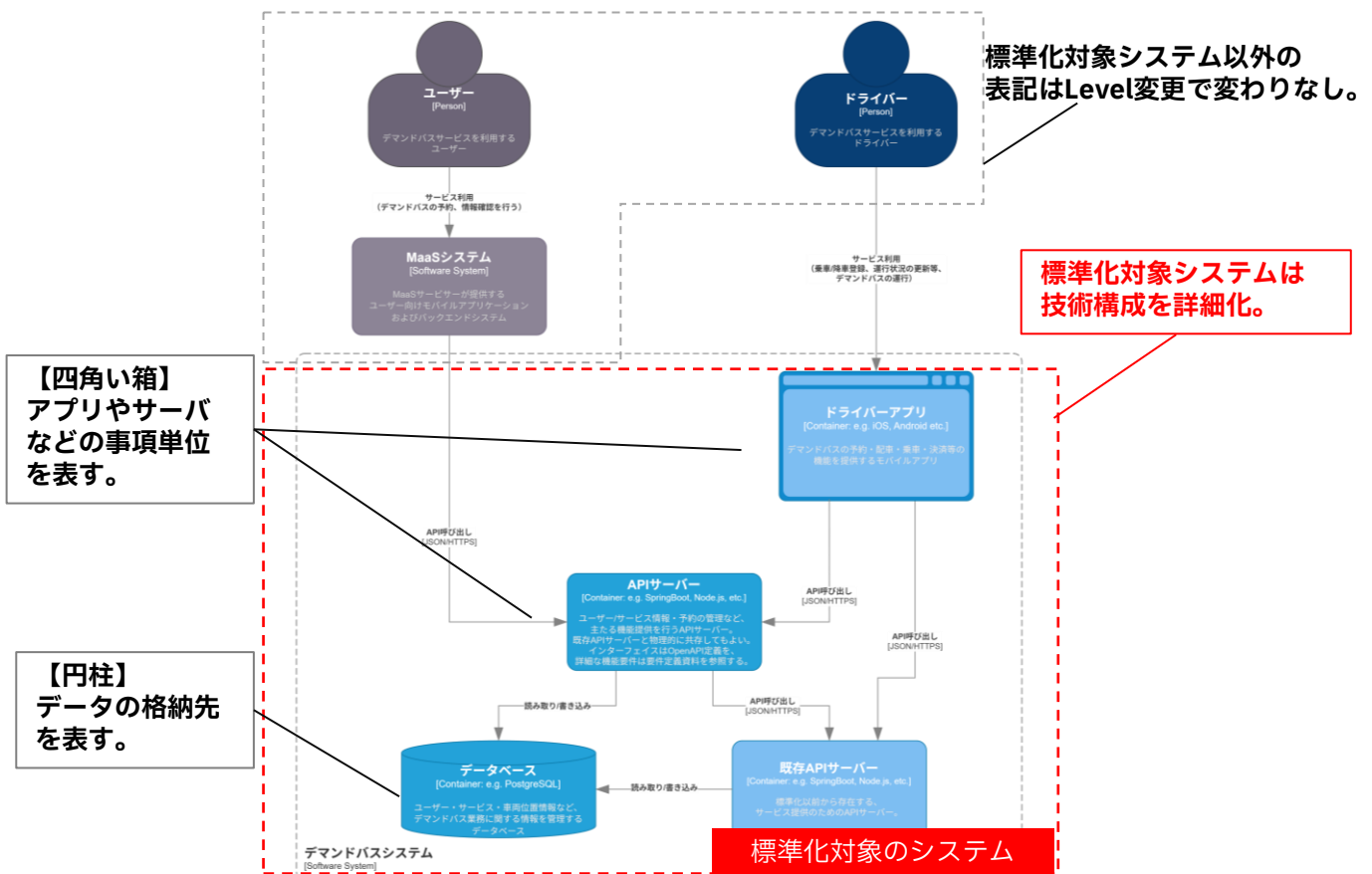
## 論理構成図の読み方

本図は標準システムアーキテクチャの階層構造を示します。上段（Level1）は利用者（ユーザー・ドライバー）とシステムの関わりや、外部システム（MaaS等）との連携関係を表します。下段（Level2）は標準化対象システム内部をアプリ・API・DB等の機能単位に分解し、具体的なデータの流れや実装の参照モデルを定義しています。

### Level 1 (System Context) の例



### Level 2 (Container) の例



## インフラ構成図の説明


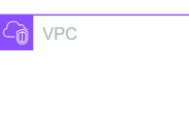

インフラ構成図は関連システムのクラウド構成の実装例を可視化したものです。本書では、広く一般に普及しているクラウドコンピューティングサービスであるAWSを実装例として用いていますが、利用するサービスを限定するものではありません。

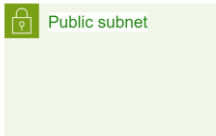
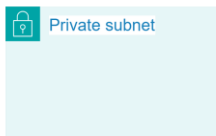
インフラ構成図は、利用者や管理画面からの操作が、セキュリティ層とAPIを経由し、サーバーレス処理とデータベースで処理・保存される流れを示しています。また、インターネット接続可能な領域と閉域を区分けし、さらに専用線接続などで安全性を担保したネットワーク設計全体を俯瞰できます。

### システム構成要素

	実装例として扱う技術要素
 <p><b>セキュリティ</b></p> <p>ユーザーの認証・認可や通信の暗号化を行い、不正アクセスや脅威からシステムを保護する機能。</p>	  <p>AWS WAF    AWS Network Firewall</p>
 <p><b>コンピュータ</b></p> <p>アプリやビジネスロジックを実際に動作させ、データを処理するための計算リソース（実行基盤）。</p>	  <p>Amazon API Gateway    AWS Lambda</p>
 <p><b>データベース</b></p> <p>業務データやシステムの状態を保存し、効率的に検索・更新できるように管理する格納庫。</p>	 <p>Amazon Aurora</p>
 <p><b>ゲートウェイ</b></p> <p>外部からのリクエストを一元的に受け付け、適切なコンピュータリソースへ通信を中継・制御するシステムの玄関口。</p>	    <p>Internet Gateway    NAT Gateway    Direct Connect Gateway    Virtual Private Gateway</p>

### 枠組みの説明

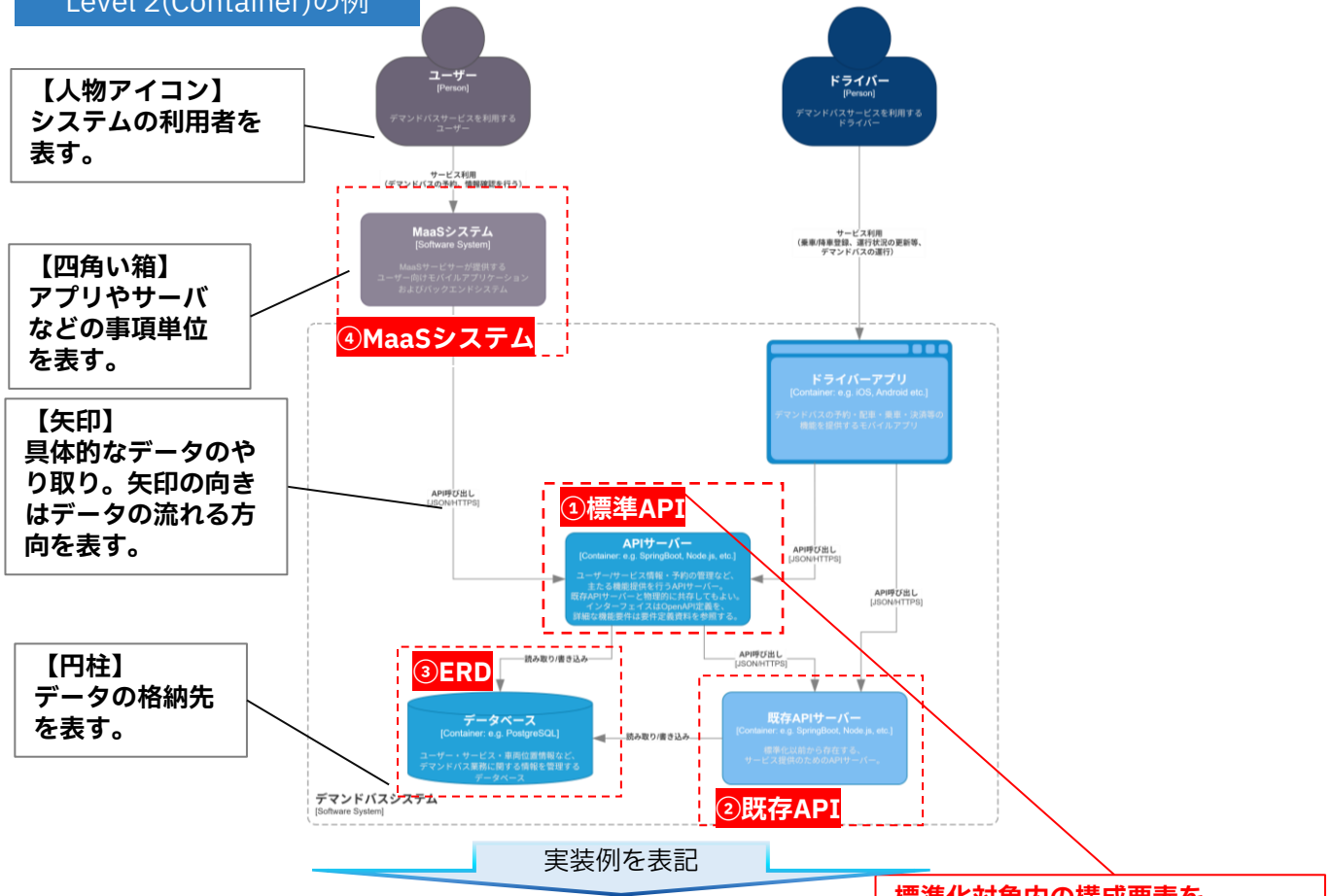
	<p>外枠(システム/アカウント): 「MaaSシステム」「認証システム」など、システムごとの管理境界を表します。</p>
	<p>VPC(紫の枠): Virtual Private Cloud。AWS上に構築された他から独立した仮想ネットワーク空間です。</p>
	<p>リージョン(緑の点枠): データセンターが集まる物理的な地域です。</p>

	<p>サブネット(色付き四角): ネットワークを用途別に区切った部屋です。</p> <ul style="list-style-type: none"> <li>➤ Public: インターネットへの出入り口がある区画 (Webサーバーやセキュリティ機器など)。</li> </ul>
	<ul style="list-style-type: none"> <li>➤ Private: インターネットから直接アクセスできない安全な区画 (データベースや処理実行部など)。</li> </ul>

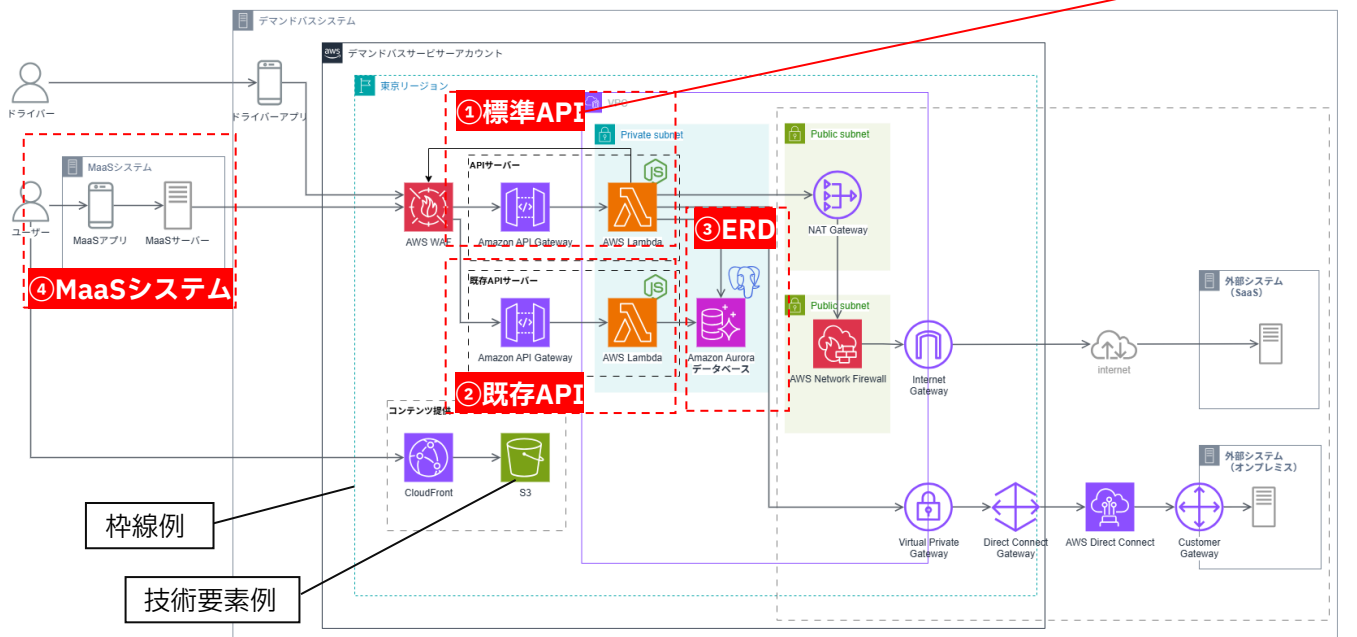
## インフラ構成図の読み方

システムの解像度を「論理」から「実装」へと高めるプロセスに着目した図です。前述の論理構成図(Level2)を基礎に、具体的な実装例を用いて詳細化しています。

### Level 2(Container)の例



### インフラ構成図の例

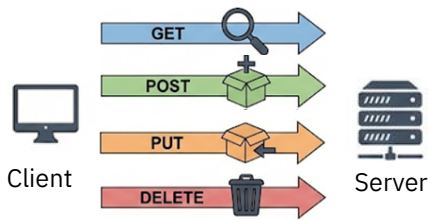


## 2.2. APIの読み方

API仕様は、RESTful APIのインターフェース定義標準であるOpenAPI 3.0によって記述されています。実装者は本定義に基づき、リソースへのアクセスパス、必須パラメータ、バリデーションルール、およびHTTPステータスコードによるエラーハンドリング仕様を読み解く必要があります。

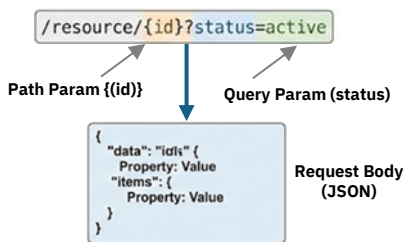
### 仕様定義の構造

#### Paths & Operations



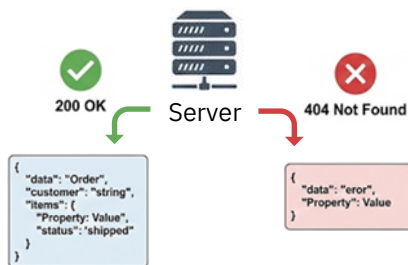
- Paths : サーバー上の「どのデータ」を操作するかを指定するURI(Uniform Resource Identifier)です。
- Operations : 実行する操作タイプです。
  - GET: リソースの取得 (副作用なし)。クエリパラメータでのフィルタリングが主。
  - POST: リソースの新規作成。リクエストボディに作成データを含めます。
  - PUT: リソースの置換・更新。ID指定でリソースの状態を変更します。
  - DELETE: リソースの削除。

#### Parameters & Request Body



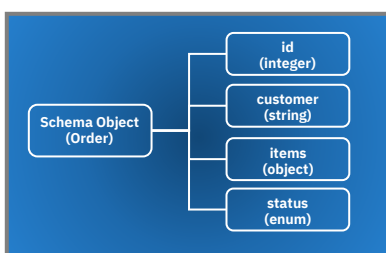
- Parameters (in): パラメータの格納場所を指定します。
  - path: URIの一部 (例: `/resource/{id}`)。リソースの特定に使用。required: trueが必須。
  - query: URIの末尾 (例: `?status=active`)。検索条件やソート指定に使用。
  - header: HTTPヘッダー (認証トークン等)。
- Request Body: POST/PUT時に送信するペイロードです。
  - content: メディアタイプを指定 (主に `application/json`)。
  - required: ボディ自体の必須有無。

#### Responses



- HTTP Status Codes: 処理結果をコードで分類します。
  - 200 OK / 201 Created: 正常終了。content内に返却データスキーマが定義されます。
  - 400 Bad Request: クライアント側の入力・形式エラー。
  - 401 Unauthorized: 認証失敗・未認証。
  - 404 Not Found: 指定リソースが存在しない。
  - 500 Internal Server Error: サーバー内部エラー。

#### Components & Schemas



- Data Types:
  - type: データ型 (integer, string, object..)
  - format: 型のフォーマット詳細。
  - enum: 列挙型。指定可能な固定値のリスト。
  - nullable: null値を許容するかどうか。
  - required: オブジェクト内で必須となるプロパティ名のリスト。

## APIの読み方

APIを利用する際に参照する仕様書の構成を説明します。左メニューの目次から、中央のパラメータ詳細、右側の実行サンプルまで、どこにどのような情報が記載されているか、画面の見方をポイントを記載しています。

The screenshot shows the API documentation for the endpoint 'ユーザーの規約同意を登録'. The page is divided into several sections, each highlighted with a red dashed box and a numbered callout:

- 1**: Search bar and navigation menu.
- 2**: API title 'ユーザーの規約同意を登録'.
- 3**: API summary text explaining the endpoint's purpose and requirements.
- 4**: Path parameters section showing 'passenger\_id' as a required string.
- 5**: Request body schema section showing 'agreements' as a required array of objects.
- 6**: Responses section listing status codes: 200 OK, 400 Bad Request, and 500 Internal Server Error.
- 7**: HTTP method and URL path section showing 'POST /passengers/{passenger\_id}/agreements'.
- 8**: Request samples section showing a JSON payload for the request.
- 9**: Response samples section showing JSON responses for different status codes (200, 400, 500).

番号	エリア名称	説明
①	目次	APIのリソース（データの種類）やエンドポイントの一覧が表示されています。ここから閲覧したいAPIを選択したり、キーワードで検索したりするためのサイドバーです。
②	APIタイトル	選択しているAPIの機能名（例：「ユーザーの規約同意を登録」）が表示されています。
③	API概要説明	このAPIが具体的にどのような処理を行うか、どのような前提条件（制約事項）があるかといった仕様の詳細が記述されています。
④	パスパラメータ	URLの一部として指定する変数（例：passenger_id）の定義です。必須項目かどうかや、パラメータの説明が書かれています。
⑤	リクエストボディ定義	データ登録や更新時に送るデータの中身（スキーマ）の定義です。データの型（String, Arrayなど）や必須有無が記載されています。
⑥	レスポンス定義	このAPIが返すHTTPステータスコード（200, 400, 500など）の一覧です。クリックすると詳細が開く形式になっていることが多いです。
⑦	HTTPメソッドとパス	実際にリクエストを送る際のHTTPメソッド（GET, POST, PUT, DELETEなど）と、エンドポイントのURLパスが表示されています。
⑧	リクエストサンプル	リクエストを送信する際の具体的なデータの記述例（Payload）です。開発者がコピー＆ペーストして試せるようになっています。
⑨	レスポンスサンプル	処理結果として返ってくるデータの具体例です。ステータスコード（200や400など）ごとの返却イメージを確認できます。

## 2.3. ERDの読み方

ERDは、システム開発やデータベース設計において、データの構造と関係性を可視化する図面です。その中で標準的に使われる「IE記法」は、線の端につく記号（鳥の足や丸印など）の組み合わせで、「1対1」「1対多」といったデータの関係（カーディナリティ）を表現します。

### テーブル定義とIE記法

#### テーブル定義

① m_ticket_routing_rule ・ チケットルーティングマスタ			
varchar	rule_id	PK	ルールID
varchar	maas_system_id	FK	MaaSシステムID
varchar	maas_ticket_id		MaaS商品ID
varchar	description		振り分け説明

四角い箱は1つのデータ管理単位(テーブル)を表します。左から順に以下の要素で構成されています。

- ① テーブル名(m\_ticket\_routing\_rule)
  - データベース上の物理テーブル名（英字）と、その内容を表す論理名（日本語）が併記されています。
- ② データ型 (varchar 等)
  - データの種類とサイズ。
- ③ 物理名 (user\_id 等)
  - システム（データベース）上で実際に使われるアルファベットの列名。
- ④ キー (PK, FK)
  - PK (Primary Key / 主キー)：その行を特定するための唯一のID。重複しない。
  - FK (Foreign Key / 外部キー)：他のテーブルと紐付くためのID。
- ⑤ 論理名 (ユーザーID 等)
  - 人間が理解しやすいように付けられた日本語の項目名。

#### IE記法

##### 基本表記

—○	0
—+	1
—<	多

- 基本となる3つの表記  
以下の3つの基本形状の組み合わせで成り立っています。  
丸 (○)：0 (ゼロ / 存在しない可能性がある)  
縦棒 (|)：1 (イチ / 単一)  
鳥の足 (<)：多 (タ / 複数)
- オプションを含んだ表記  
オプションを含んだ表記では、以下の意味を有します。

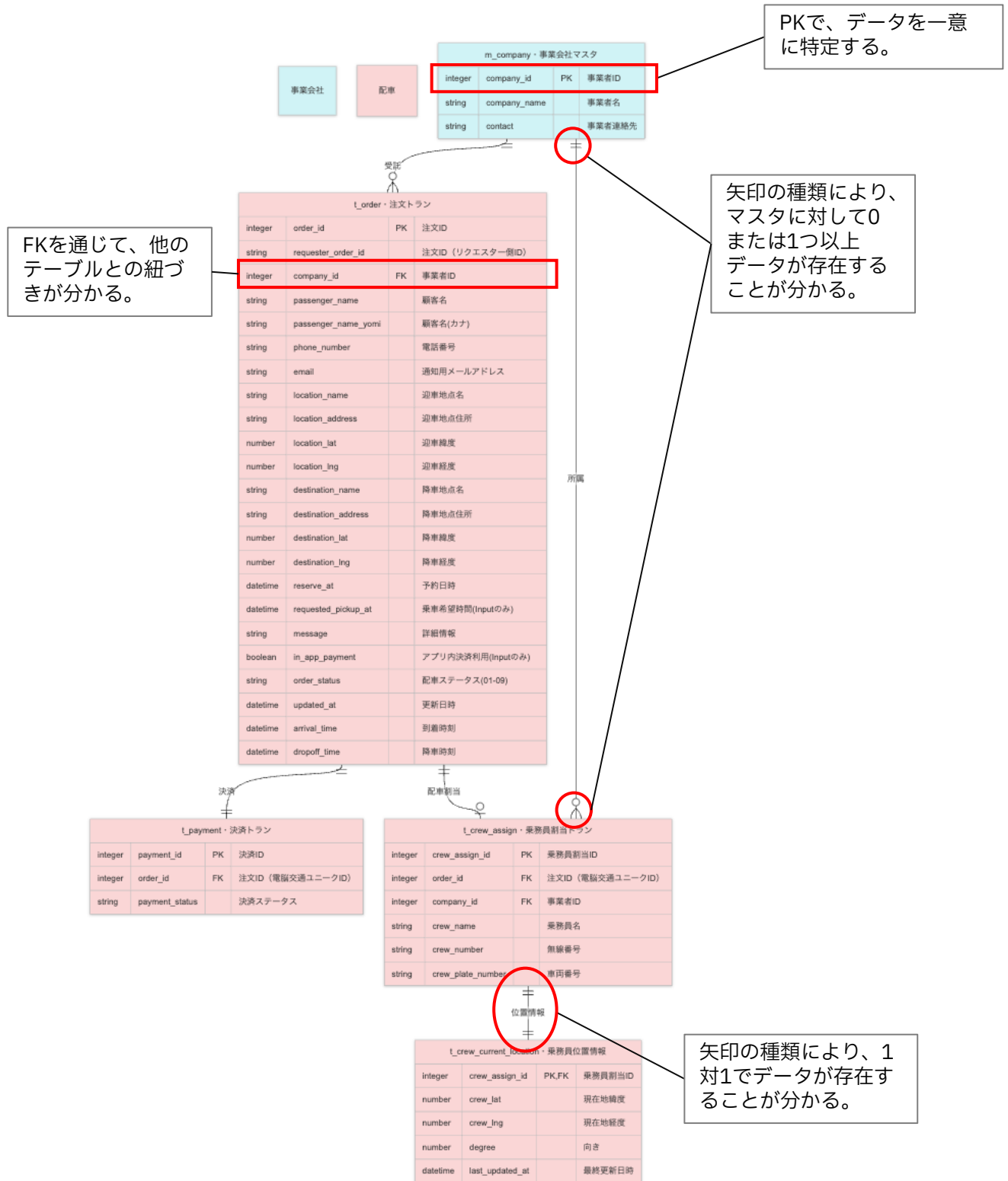
##### オプション表記

—<	1:1以上
—○<	1:0以上
—○+	1:0 or 1
—<	1:多
—	1:1

名称	意味・解説
1:1以上	「少なくとも1つは必ず必要で、複数あってもよい」 (例：注文には、必ず1つ以上の商品明細が必要)
1:0以上	「なくてもよいが、複数あるかもしれない」 (例：会員は、注文履歴が0件かもしれないし、多数あるかもしれない)
1:0 or 1	「なくてもよいが、あるとしても1つだけ」 (例：社員に対して、社用車は0台か、割り当てられても1台)
1:多	「多数」 ※これは少し簡略化された表記で、通常は上の「0以上」か「1以上」を明確にして使います。
1:1	「必ず1つだけ存在する」 (例：注文明細には、必ず1つの商品情報が紐づく)

## ERDの読み方

ERDはシステムが扱うデータのつながりを表す設計図です。四角い箱（テーブル）は注文や事業者ごとのデータの棚であり、中に管理項目が定義されています。箱を結ぶ線はデータ同士の関係性（リレーション）を示し、線の端の記号で「1対1」や「1対多」といった結びつきのルールを表現しています。中心となる箱から線を辿ることで、業務における情報の流れや依存関係を読み解くことができます。



## 2.4. 業務フロー/業務一覧の読み方

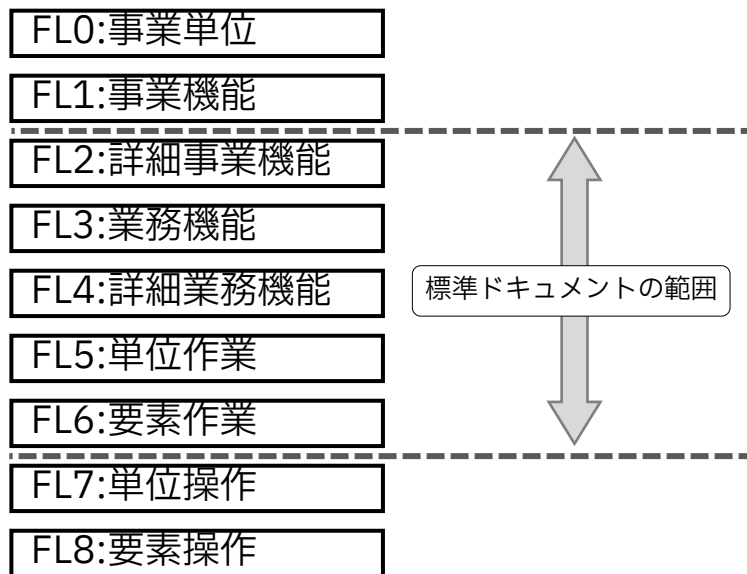
業務フロー/業務一覧は、標準API仕様に基づくシステム実装を運用するための業務モデルの指針です。業務をファンクションレイヤー（以下、FLとする）定義に従って「事業」から「要素作業」まで階層化し、標準的な粒度で整理します。その上で、BPMN 2.0表記法に基づきシステムと人の動きを可視化します。これにより業務要件とAPIの整合性を確保し、システムを運用するための具体的な業務手順を理解することができます。

### ファンクションレイヤーによる業務の階層化

業務を階層化して整理する理由は、読み手の役割によって必要な「情報の解像度」が異なるためです。

例えば、全体像を把握したい管理者層には「大まかな業務の流れ」があれば十分ですが、システムを実装する開発者には「具体的な処理手順」が必要です。

本ドキュメントでは、双方の視点をカバーし、ビジネスの全体理解から実際のシステム設計までスムーズにつなげるために、適切な粒度（FL2～FL6）に分けて定義しています。



### FLによる業務分解例

機能階層	具体例	想定利用者	利用目的
FL0	自動車保険事業	経営層	グループ全体の事業管理
FL1	個人向け販売	事業責任者	バリューチェーンの構築
FL2	ディーラー経由販売	部門マネージャー	ビジネスモデル・チャネル設計
FL3	見積り依頼～契約	プロセスオーナー	業務サイクルの管理・KPI設定
FL4	見積り依頼・価格提示	チームリーダー	部署間の連携・進捗管理
FL5	見積り依頼受付	実務担当者	自身のタスク・責任範囲の確認
FL6	依頼書開封・チェック	業務改善担当	ツール選定・工数（時間）分析
FL7	依頼書ファイル開く	RPA開発者	自動化手順の設計
FL8	ファイルメニュークリック	RPA開発者	ロボットの具体的な動作指定

## 業務一覧の読み方

業務一覧は、業務全体を大きな塊（FL2）から具体的な作業手順（FL6）へと段階的に分解し、整理したものです。「左から右へ」視線を動かすことで、業務の解像度が高まる構造になっています。また、APIを利用する業務については、APIのIDを割り振り、APIの資料と対応付けて確認できるようになっています。

FLごとにIDと名称を記載。

No	FL0		FL2		FL3		FL4		FL5		FL6	
	ID	事業単位	ID	詳細事業機能	ID	業務機能	ID	詳細業務機能	ID	単位作業	ID	要素作業
1	M13	QRチケット事業	M13-FL2-010	商品企画								
2	M13	QRチケット事業	M13-FL2-010	商品企画	M13-FL3-010	商品企画						
3	M13	QRチケット事業	M13-FL2-010	商品企画	M13-FL3-010	商品企画	M13-FL4-010	商品企画立案				
4	M13	QRチケット事業	M13-FL2-010	商品企画	M13-FL3-010	商品企画	M13-FL4-010	商品企画立案	M13-FL5-010	商品企画検討		
5	M13	QRチケット事業	M13-FL2-010	商品企画	M13-FL3-010	商品企画	M13-FL4-010	商品企画立案	M13-FL5-010	商品企画検討	M13-FL6-010	商品企画検討
6	M13	QRチケット事業	M13-FL2-010	商品企画	M13-FL3-010	商品企画	M13-FL4-010	商品企画立案	M13-FL5-020	商品企画承認		
7	M13	QRチケット事業	M13-FL2-010	商品企画	M13-FL3-010	商品企画	M13-FL4-010	商品企画立案	M13-FL5-020	商品企画承認	M13-FL6-020	商品企画承認
8	M13	QRチケット事業	M13-FL2-010	商品企画	M13-FL3-010	商品企画	M13-FL4-020	商品企画確認				
9	M13	QRチケット事業	M13-FL2-010	商品企画	M13-FL3-010	商品企画	M13-FL4-020	商品企画確認	M13-FL5-030	商品企画確認		
10	M13	QRチケット事業	M13-FL2-010	商品企画	M13-FL3-010	商品企画	M13-FL4-020	商品企画確認	M13-FL5-030	商品企画確認	M13-FL6-030	商品企画確認
11	M13	QRチケット事業	M13-FL2-010	商品企画	M13-FL3-010	商品企画	M13-FL4-020	商品企画確認	M13-FL5-040	商品企画書受領		
12	M13	QRチケット事業	M13-FL2-010	商品企画	M13-FL3-010	商品企画	M13-FL4-020	商品企画確認	M13-FL5-040	商品企画書受領	M13-FL6-040	商品企画書受領

FL4以下の業務についてアクターを表現。

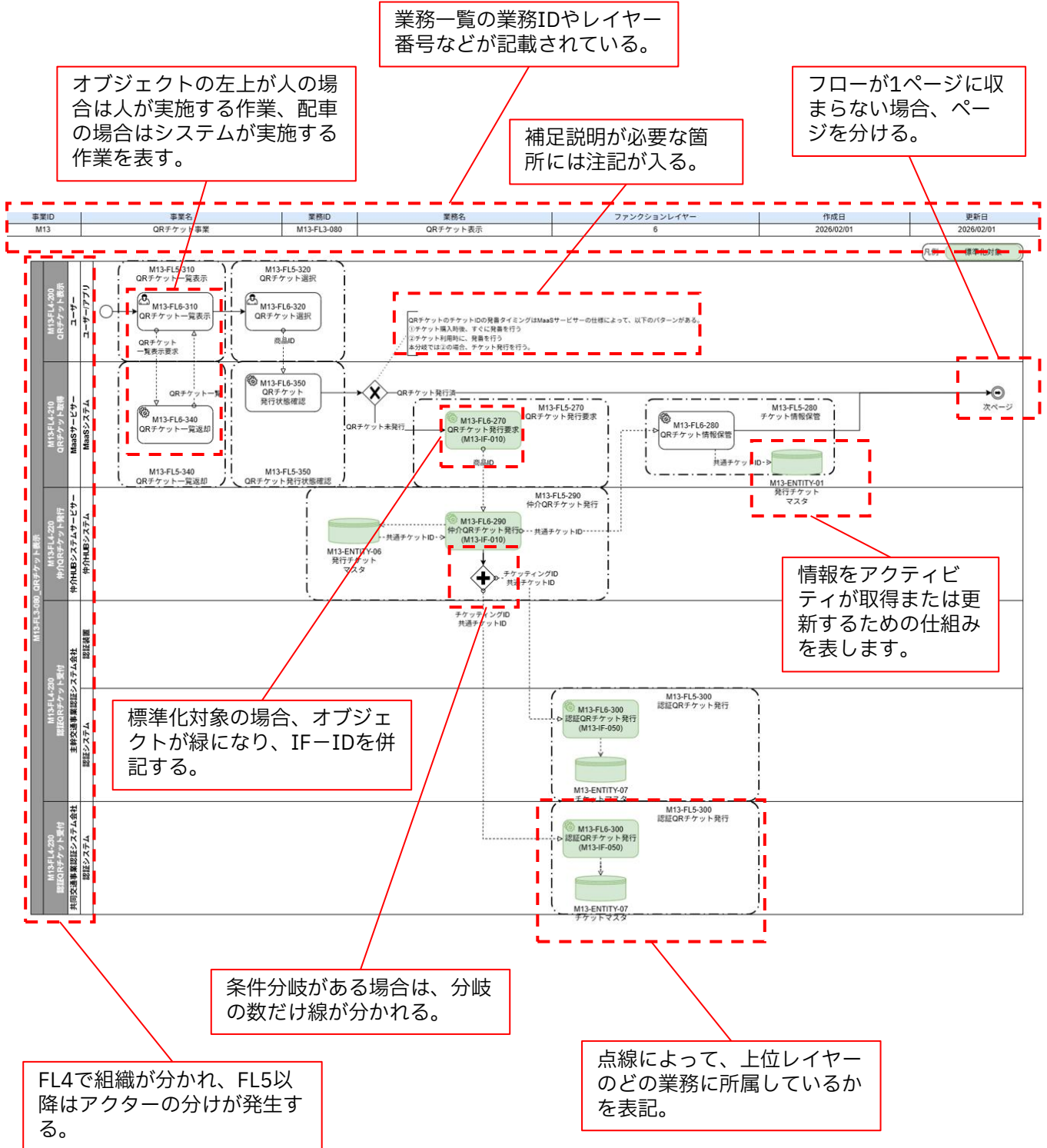
各業務の詳細を説明。

アクター	システム	API		目的・内容
		ID	名称	
				交通事業者、MaaSサービス、認証システム間で共通のQRチケットのサービスを提供するにあたり、商品を決めるための業務。 共通のQRチケットを利用する事業者間で、商品情報(商品名、販売金額など)を取り決める業務プロセス。
主管交通事業者				主管交通事業者が、商品企画を検討し、共同交通事業者と同意を行う。 主管交通事業者の規格事が商品企画の検討を行い、共同交通事業者へ確認をしてもらう。 主管交通事業者の規格事が商品企画の検討を行い、共同交通事業者へ確認をしてもらう。
主管交通事業者				主管交通事業者の承認者が、担当者から受領した商品企画の承認を行う。承認した企画書は共同交通事業者へ送信する。 主管交通事業者の承認者が、担当者から受領した商品企画の承認を行う。承認した企画書は共同交通事業者へ送信する。
主管交通事業者				共同交通事業者が、主管交通事業者から受領した商品企画の確認を行う。 共同交通事業者の担当者が、主管交通事業者から受け取った商品企画の確認を行い、合意を行う。 共同交通事業者の担当者が、主管交通事業者から受け取った商品企画の確認を行い、合意を行う。
主管交通事業者				共同交通事業者の担当者が、主管交通事業者から企画書を受領する。 共同交通事業者の担当者が、主管交通事業者から企画書を受領する。

APIを扱う業務について、IDと名称を記載。



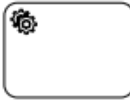


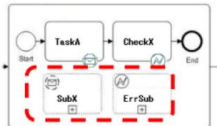

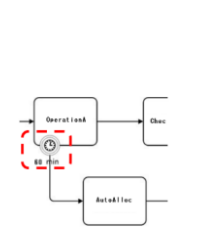


## 業務フローの読み方

業務フローは、業務一覧で細分化した業務を、階層ごとの流れとしてまとめた資料です。視線を左から右へ動かすことで、大枠の「詳細事業機能（FL2）」から、詳細な「要素作業（FL6）」へと段階的に具体化される構造になっています。



## BPMN 2.0 に基づく業務フロー記述ルール

本標準ドキュメントでは、「業務一覧」で整理したFL構造を、BPMN2.0のルールに従って図式化します。本標準ドキュメントで使用するBPMN2.0のオブジェクトは以下の種類があります。

No	分類	図形名称	説明	図形
1	アクティビティ	ユーザータスク (User Task)	ユーザータスクは、BPMNエンジンと連携してユーザーが実行する業務処理を表す。	
2	アクティビティ	緑のユーザータスク (Service Task)	緑のユーザータスクは、サービスタスクの中でも本プロジェクトの標準化対象のものを表す。	
3	アクティビティ	サービスタスク (Service Task)	サービスタスクは、Webサービスのような人間の介入がないアプリケーションによって実行される業務処理を表す。	
4	アクティビティ	緑のサービスタスク (Service Task)	緑のサービスタスクは、サービスタスクの中でも本プロジェクトの標準化対象のものを表す。	
5	アクティビティ・ マーカ	サブプロセス (Sub-Process)	サブプロセスは、親ビジネスプロセスに組み込まれた下位のビジネスプロセスを表す。	
6	アクティビティ・ その他	イベント・サブプロセス (Event Sub-Process)	イベント・サブプロセスはプロセスの枠線を点線で表す(※図中の点線で囲った部分)。 イベント・サブプロセスは、ビジネスプロセス内に配置され、そのビジネスプロセスで発生したイベントをトリガとして実行されるサブプロセスである。	
7	イベント	イベント(イベントタイプなし) (None Event) (Start Event / End Event)	<p>■ Start 名前が示すとおり、開始イベント (Start Event) は、特定のプロセスまたはコレオグラフィーがどこから始まるかを示します。</p> <p>■ End 名前が示すとおり、終了イベント (End Event) は、プロセスまたはコレオグラフィーがどこで終わるかを示します。</p>	
8	イベント	境界イベント (Boundary Event)	<p>タイマー・トリガの境界イベントの例(※図中の点線で囲った部分)。</p> <p>境界イベントは、タスク、サブプロセス及びコールアクティビティ内でメッセージ、エラー、タイマーなどのイベントが発生した場合に、境界イベントに接続するフローにトークンが流れる。</p> <p>■ 中断イベントの場合は、タスク、サブプロセス及びコールアクティビティの実行を中断し、トークンは境界イベントに接続するフローに流れる。</p> <p>■ 非中断イベントの場合は、タスク、サブプロセス及びコールアクティビティの実行を中断せず、後続のフローが継続されると共に、トークンは境界イベントに接続するフローに流れる。</p>	
9	イベント	エラー (Error Event)	<p>エスカレーション・イベント・サブプロセスは、ビジネス・アクティビティが実行上の制約 (時間ベースの締め切りなど) を満たさない場合に、その完了を迅速化するための手段を実行します。</p> <p>エスカレーション開始イベントは、インライン・イベント・サブプロセスをトリガーするためのみに許可されます。</p>	
10	イベント	リンク (Link Event)	エラーイベントは、アクティビティでエラーが発生した場合に処理を中断し、エラー処理に接続するイベントを表す。	

No	分類	図形名称	説明	図形
11	イベント	強制終了 (Terminate Event)	強制終了イベントは、ビジネスプロセスの即時停止を表す。同一ビジネスプロセス内の全ての実行中タスクを強制終了し、当該ビジネスプロセスをただちに終了させる。ただし、サブプロセスで使用されている場合、当該サブプロセスを終了し、親ビジネスプロセスに戻る。	
12	ゲートウェイ	排他ゲートウェイ (Exclusive Gateway)	排他ゲートウェイは、フローが排他的に分岐すること又は分岐したフローが結合することを表す。実行可能モデルにおける排他ゲートウェイは、分岐条件情報の設定条件で評価しフローの分岐を行う。 (1)、(2)のいずれかを使用するかは、製品で対応している図形を使用し、両方に対応している場合は(2)を使用する。	
13	ゲートウェイ	並列ゲートウェイ (Parallel Gateway)	並列ゲートウェイは、フローが並列に分岐しその後のフローで並列処理されていたフローが同期的に結合することを表す。	
14	接続オブジェクト	シーケンスフロー (Sequence Flow)	シーケンスフローは、タスク、イベント、ゲートウェイ間の処理の流れを表す。	
15	接続オブジェクト	方向性の関連、 管理データへの関係 (Directional Association, Directed Data Association)	データの関連付けは、タスクとデータの関連性、タスクと管理データへの関係を図示する。	
16	データ	データオブジェクト (Data Object)	データオブジェクトは、アクティビティの入出力要素として表す。 データオブジェクトは、実行モデルの処理フローに影響しない。	
17	データ	データストア (Data Store)	データストアは、個別DB、共有DBを表す。 データストアは、実行モデルの処理フローに影響しない。	
18	データ	緑のデータストア (Data Store)	緑のデータストアは、サービスタスクの中でも本プロジェクトの標準化対象のものを表す。	
19	スイムレーン	プール (Pool)	プールは、一連のビジネスプロセスを記述する範囲を表す。プールには、一意なビジネスプロセス識別子を指定する。	
20	スイムレーン	レーン (Lanes)	レーンは、タスクを実行するユーザーや部署の範囲を表す。	
21	その他	グループ (Group)	グループは、要素のセットを強調表示する。 制約や規則を課すものではない。	
22	その他	テキスト注釈 (Text Annotation)	テキスト注釈は、プロセス又はその要素に関する補足情報(コメント)を表す。ダイアグラム上の任意の場所に配置することができ、どの要素にもアタッチすることができる。	



## 3. 標準ドキュメントの解説

---

- 3.1. 標準化のスコープと概要
- 3.2. システムアーキテクチャの解説
- 3.3. APIの解説
- 3.4. ERDの解説
- 3.5. 業務フロー/業務一覧の解説(一部抜粋)



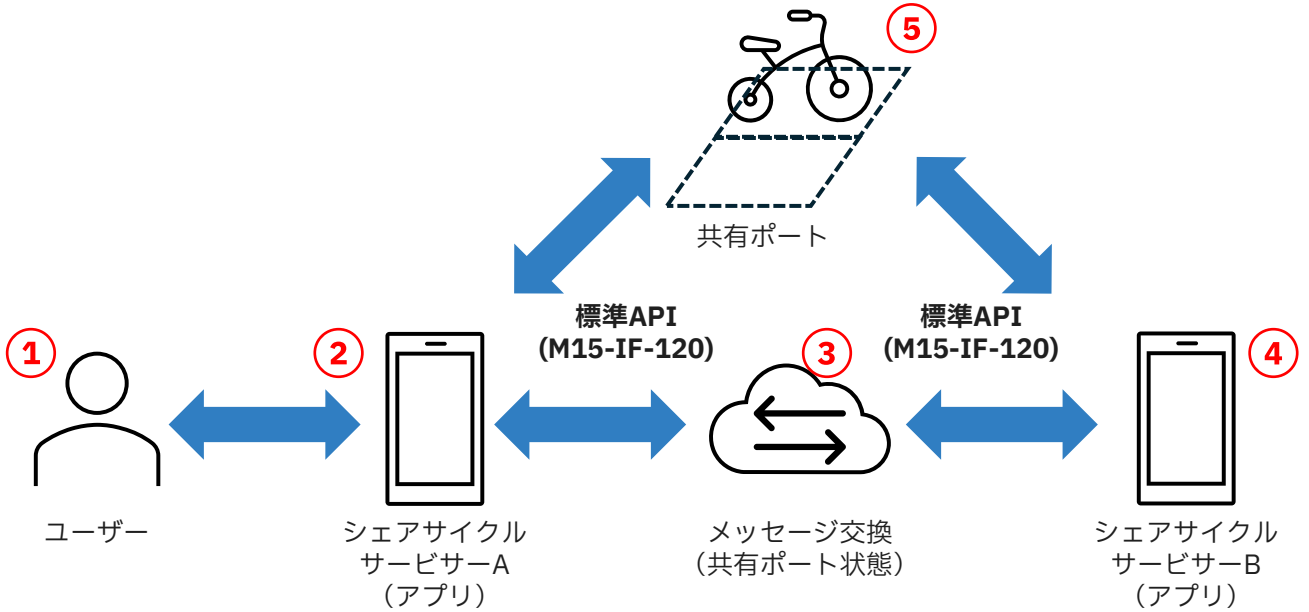
### 3.1 標準化の範囲と概要

本章で解説する標準化の範囲と概要を解説します。

#### 標準化の範囲

本プロジェクトでは、異なるシェアサイクル事業者が保有するポートを相互に利用可能にするための標準API仕様を策定することを目的としています。

下図は、標準化の対象となるシステム範囲と、関与する各プレイヤー（アクター）の関係性を示したものです。



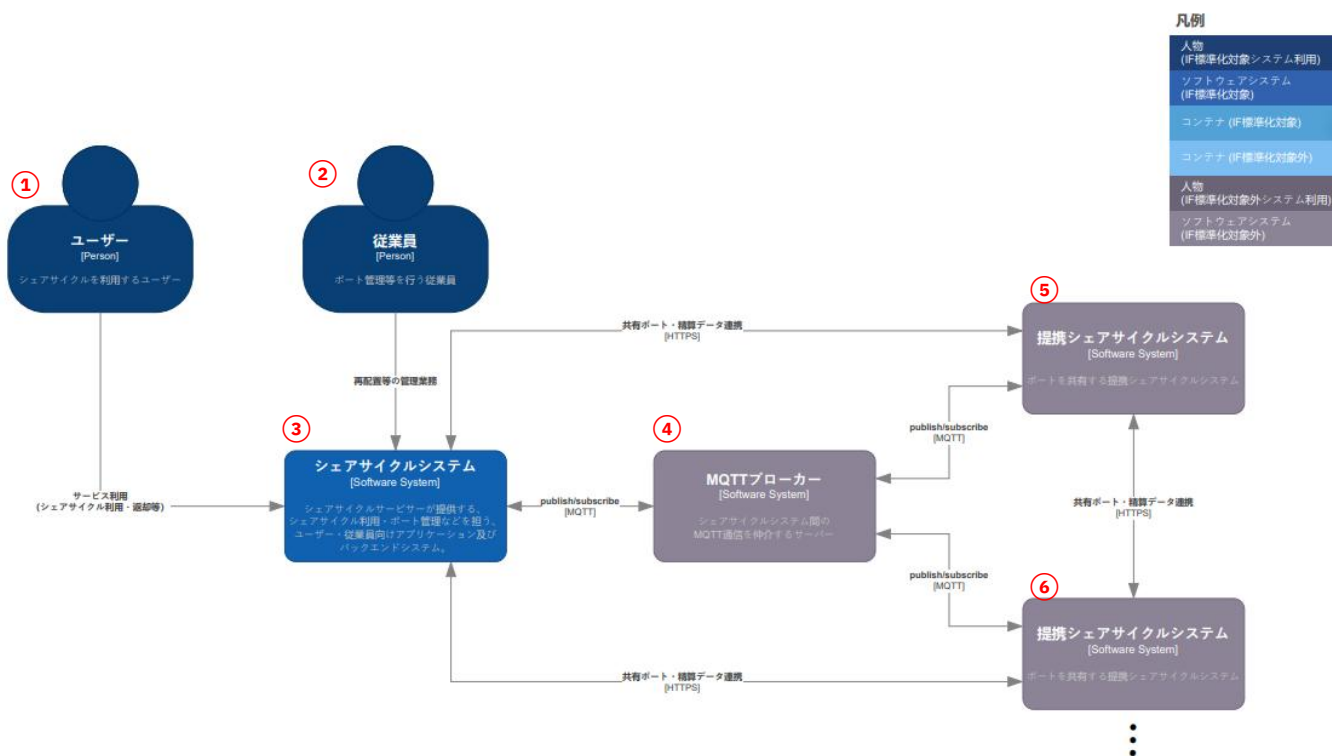
番号	アクター	役割
①	ユーザー	シェアサイクルサービスの利用者。 専用アプリ（シェアサイクルサービスA）を利用して、ポートの検索、自転車の予約、貸出・返却の手続きを行います。
②	シェアサイクルサービスA	ユーザーが直接契約・利用している事業者システム。 自社のユーザーに対し、標準APIを通じて他社（サービスB）のポート情報や予約機能を提供し、サービス利用を管理します。 シェアサイクル事業者Aは、状況次第でシェアサイクル事業者Bと同じ役割を持ちます。
③	メッセージ交換	事業者間のデータ連携を仲介する基盤。 標準API（MQTT等）を用いて、各事業者のポート満空情報（貸出・返却可能数）をリアルタイムに同期・共有する役割を担います。
④	シェアサイクルサービスB (アプリ)	ポートを共有（提供）する連携先事業者システム。 自社が管理するポート情報を標準API経由で公開し、他社（サービスA）ユーザーによる自転車の貸出・返却を受け入れます。 シェアサイクル事業者Bは、状況次第でシェアサイクル事業者Aと同じ役割を持ちます。
⑤	共有ポート	相互利用の対象となる物理的なサイクルポート。

## 3.2. システムアーキテクチャの解説

システムアーキテクチャのドキュメントを掲載します。

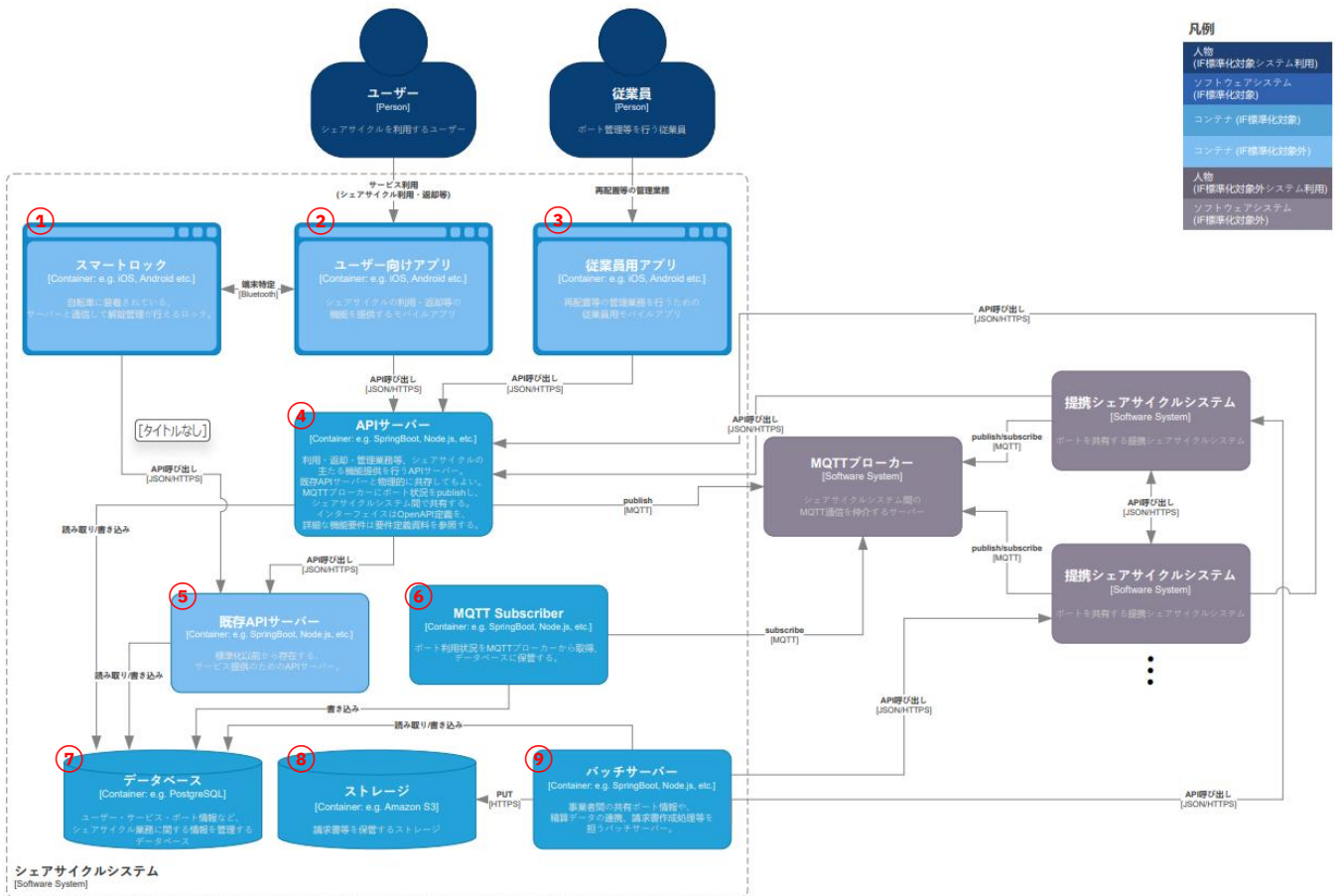
### 論理構成図

#### Level1: System Context



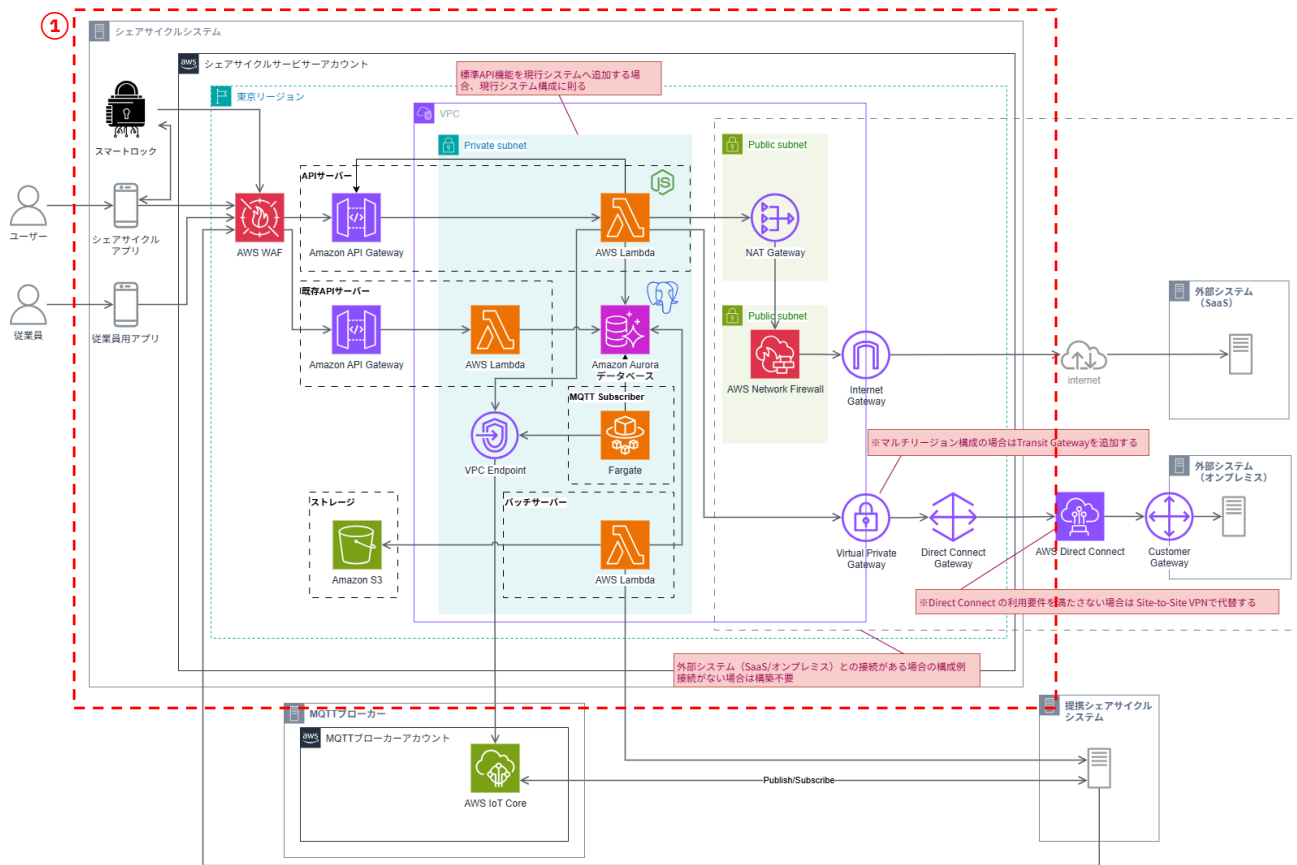
番号	名称	カテゴリ	説明
①	ユーザー	Person	シェアサイクルを利用するエンドユーザー。サービスの利用（貸出・返却等）をシステム経由で行う。
②	従業員	Person	ポートの管理や車両の再配置など、運営に関わる管理業務を行う担当者。
③	シェアサイクルシステム	Software System	本プロジェクトの核となるシステム。ユーザー・従業員向けアプリおよびバックエンド機能を担う。
④	MQTTブローカー	Software System	シェアサイクルシステム間におけるMQTT通信（Publish/Subscribe）を仲介するサーバー。
⑤	提携シェアサイクルシステム	Software System	自システム（③）と連携し、ポートの共有や精算データの連携を行う外部のシステム。
⑥	提携シェアサイクルシステム	Software System	⑤と同様、システム間連携を行う別の提携先システム。

## Level2: Container



番号	名称	所属システム	説明
①	スマートロック	シェアサイクルシステム	自転車に装着されている、サーバーと通信して解錠管理を行うデバイス。
②	ユーザー向けアプリ	シェアサイクルシステム	シェアサイクルの利用・返却等の機能を提供するモバイルアプリ。
③	従業員用アプリ	シェアサイクルシステム	再配置等の管理業務を行うための従業員用モバイルアプリ。
④	APIサーバー	シェアサイクルシステム	利用・返却・管理業務等、シェアサイクルの主たる機能提供を行うAPIサーバー。
⑤	既存APIサーバー	シェアサイクルシステム	標準化以前から存在する、サービス提供のための既存APIサーバー。
⑥	MQTT Subscriber	シェアサイクルシステム	ポート利用状況をMQTTブローカーから取得し、データベースに保管するコンポーネント。
⑦	データベース	シェアサイクルシステム	ユーザー、サービス、ポート、車両等に関する情報を管理するリレーショナルデータベース。
⑧	ストレージ	シェアサイクルシステム	請求書等のファイルを保管するオブジェクトストレージ。
⑨	バッチサーバー	シェアサイクルシステム	事業者間の共有ポート情報や、精算データの連携、請求書作成処理等を担うサーバー。

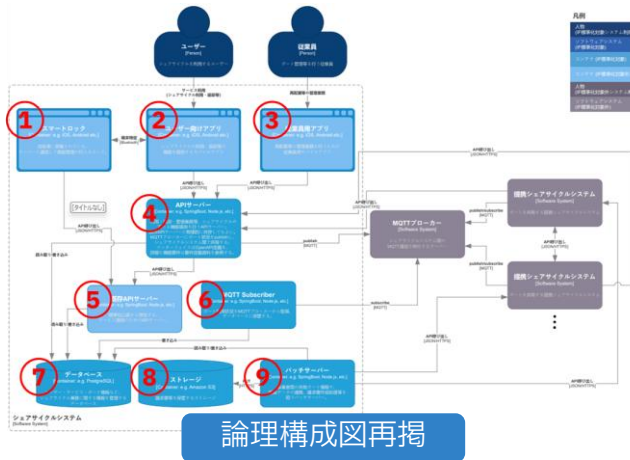
# インフラ構成図(実装例)



番号	システム領域名	詳細説明
④	シェアサイクルシステム	<ul style="list-style-type: none"> <li>サーバーレス構成の採用: API実行環境に <b>Lambda</b>、MQTT受信に <b>Fargate</b> を利用することで、管理負荷を抑えつつ負荷に応じたスケーリングを実現しています。</li> <li>セキュアなネットワーク設計: データベース (Aurora) やロジック (Lambda) を <b>Private Subnet</b> 内に配置し、<b>AWS Network Firewall</b> や <b>WAF</b> を通じて外部の脅威から保護しています。</li> <li>ハイブリッド・マルチクラウド連携: 外部のSaaSやオンプレミスシステムとは <b>Internet Gateway</b> または <b>Direct Connect</b> を通じて連携し、事業者間のデータ交換 (MQTT含む) が可能な構成となっています。</li> </ul>

### 3.3. APIの解説

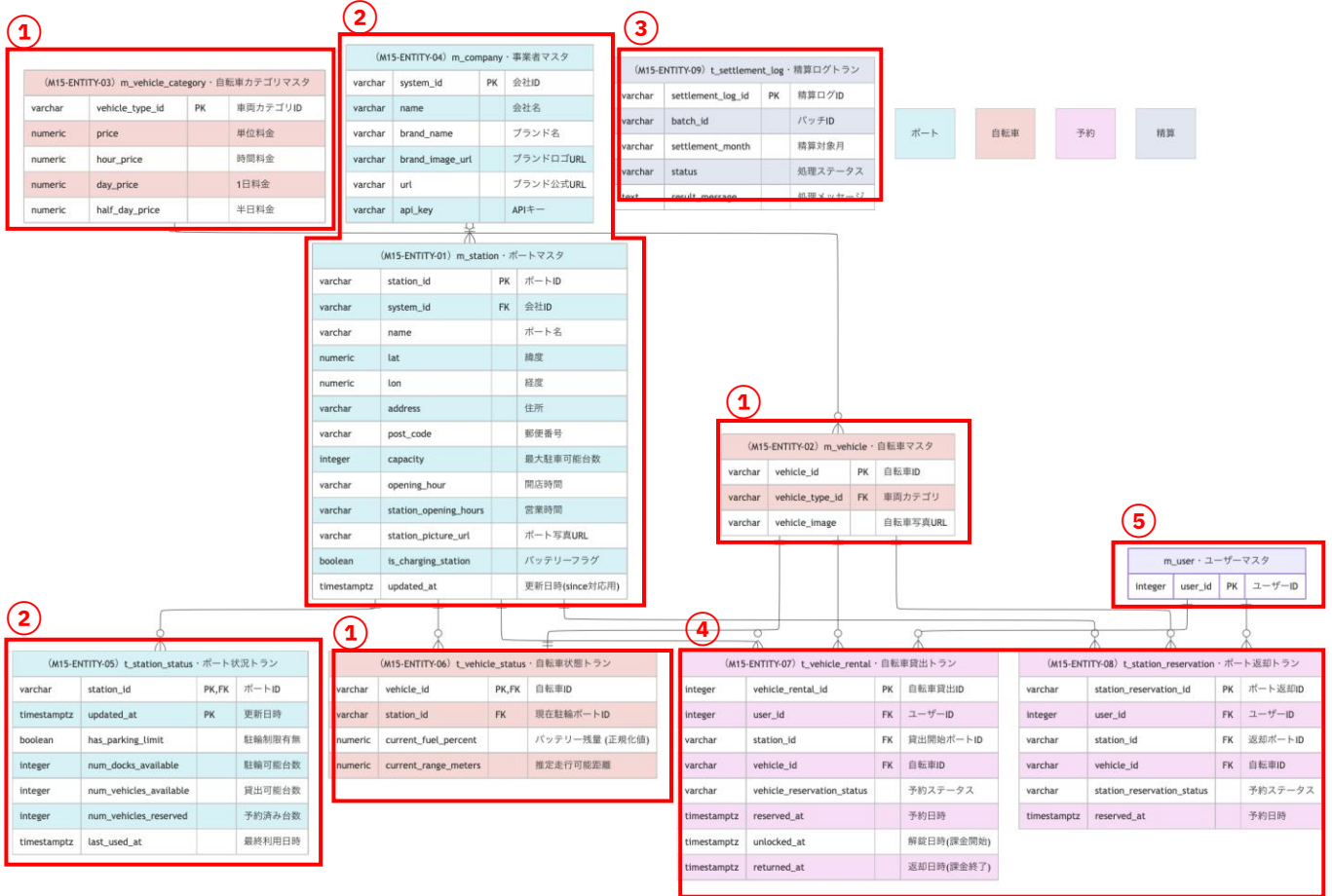
APIの一覧を掲載します。



APIサー バー	API名	API使用場所	機能説明	目的
シェアサ イクルシ ステム	(M15-IF-010) 共有ポート管理	⑨ → ⑤	共有ポート情報を取得する。	<b>ポート情報の相互利用</b> 他事業者のポート情報を自システムへ取り込み、ユーザーが横断的にポートを利用できるようにするため。
	(M15-IF-020) 共有ポート管理(静的)	⑨ → ⑤	共有ポート情報を取得する。	<b>マスタデータの同期</b> 住所や名称など更新頻度の低いポート基本情報を一括取得し、DBのマスタ情報を最新化するため。
	(M15-IF-030) ポート一覧確認(動的)	② → ④	ポートの一覧情報を取得する。	<b>リアルタイム状況の可視化</b> ユーザーアプリ上の地図等に、周辺ポートの最新の貸出可能台数をリアルタイムに表示するため。
	(M15-IF-040) ポート詳細確認	② → ④	指定されたポートIDの詳細情報を取得する。	<b>ポート情報の詳細提供</b> 特定のポートにおける利用可能サービスや営業時間など、詳細な属性情報をユーザーに提示するため。
	(M15-IF-050) 予約登録	② → ④	自転車の予約を登録する。ユーザーの予約上限数を超えている場合や予約しようとした自転車が、直前に他のユーザーに予約されてしまった場合に422を返す。	<b>車両在庫の確保</b> 特定の車両を一定時間キープし、他のユーザーによる重複予約や飛び込み利用を防止するため。
	(M15-IF-060) 予約キャンセル	② → ④	予約をキャンセルする。既にレンタルが開始されている注文をキャンセルしようとした場合に422を返す。	<b>予約資産の開放</b> 利用しなくなった予約を速やかに取り消し、他のユーザーが再度予約・利用できる状態に戻すため。
	(M15-IF-070) 返却予約登録	② → ④	返却予約を登録する。ポートが満車で、予約枠が確保できない場合に422を返す。	<b>返却場所の確約</b> 目的地ポートでの返却枠を事前に確保し、到着時の満車による返却不能トラブルを回避するため。
	(M15-IF-080) 返却予約キャンセル	② → ④	返却予約をキャンセルする。	<b>返却枠の二次利用促進</b> 不要になった返却予約を解放し、他の返却希望ユーザーがその枠を利用できるようにするため。
	(M15-IF-090) 解錠処理	② → ①	Bluetoothで検知したMACアドレスをもとに自転車を解錠する。車両の近くにいない場合に422を返す。	<b>車両利用の開始</b> 予約した車両とアプリを持つユーザーの近接を確認し、物理的なロックを解錠して利用を開始させるため。
	(M15-IF-100) 返却処理	② → ④	自転車の返却処理を行います。ポートの近くにいない場合や、ポートが営業時間外である場合に422を返す。	<b>利用実績の確定</b> 適切な場所・時間での返却を確認してレンタルを終了させ、課金対象となる利用時間を確定させるため。
	(M15-IF-110) 精算データ受信	⑨ ← 外部	共同利用ポート利用履歴から精算対象のデータを他事業者から受信する	<b>事業者間精算の実行</b> 他事業者の車両が自社ポートを利用した実績（またはその逆）を取り込み、最終的な対価の支払い・請求を行うため。

### 3.4. ERD

ERDのドキュメントを掲載します。



番号	カテゴリ	主なテーブル (エンティティ)	内容・役割の説明
①	自転車・マスタ	<ul style="list-style-type: none"> <li>自転車カテゴリマスタ</li> <li>自転車マスタ</li> <li>ポートマスタ</li> <li>自転車状態トラン</li> </ul>	<b>車両および拠点の管理</b> 自転車の車種や料金設定、および貸出拠点となるポートの静的情報を管理する。また、各車両の現在の駐輪場所やバッテリー残量などの動的状態も保持する。
②	ポート・状況管理	<ul style="list-style-type: none"> <li>事業者マスタ</li> <li>ポート状況トラン</li> </ul>	<b>拠点稼働状況の可視化</b> サービスを提供する事業者の情報と、各ポートにおける現在の貸出可能台数や予約済み台数といったリアルタイムの混雑状況を管理する。
③	精算管理	<ul style="list-style-type: none"> <li>精算ログトラン</li> </ul>	<b>利用料金の確定と記録</b> バッチ処理等によって算出された精算対象月ごとの料金データや、処理ステータスを管理し、事業者間精算やユーザーへの請求根拠として記録する。
④	利用・予約トラン	<ul style="list-style-type: none"> <li>自転車貸出トラン</li> <li>ポート返却トラン</li> </ul>	<b>レンタルサイクルの実行管理</b> ユーザーによる車両の予約から貸出（解錠）、返却までのサイクルを管理する。いつ、誰が、どの車両を、どのポートで利用したかという実績を時系列で保持する。
⑤	ユーザー管理	<ul style="list-style-type: none"> <li>ユーザーマスタ</li> </ul>	<b>利用者情報の特定</b> サービスを利用するユーザーの一意的なIDを管理し、貸出・予約・精算といった各トランザクションデータと紐付けるための起点となる。

### 3.5. 業務フロー/業務一覧

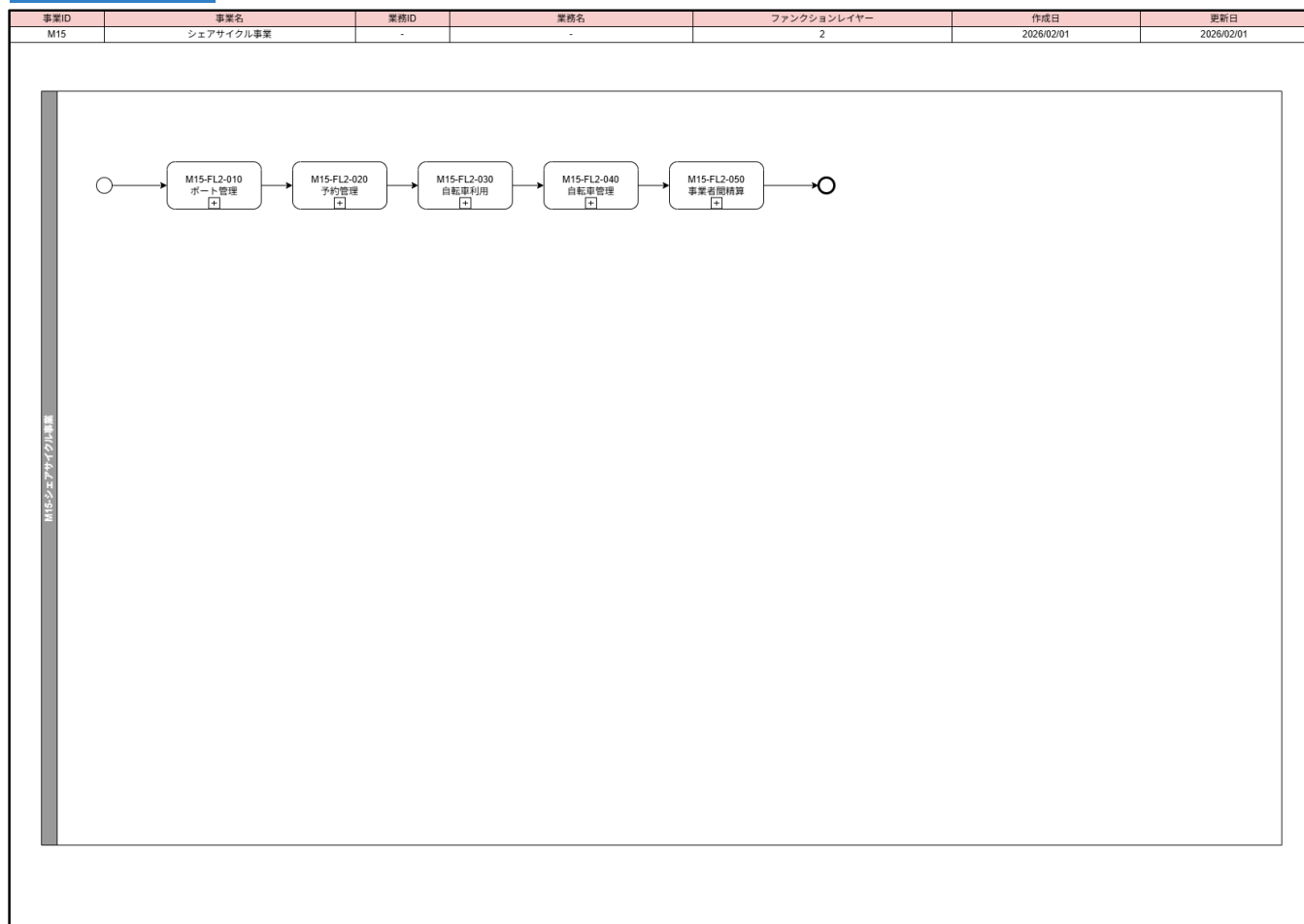
業務フロー/一覧のドキュメント(重要箇所)を掲載します。  
FL4以降では、シェアサイクルの利用予約・利用・返却予約・返却の業務を抜粋して説明します。

#### FL(ファンクションレイヤー)2

FL2フローは、業務全体の概略を最大粒度で把握することを目的とします。

まず、事業者間で、共有するポートの情報を管理する「ポート管理」を行います。次に、ユーザーによる自転車の予約の管理を行う「予約管理」を経て、実際の「自転車利用」へと移ります。利用終了後は、自転車の再配置やバッテリー交換を管理する「自転車管理」が実施され、最終的に共有ポートの利用実績に基づいた金銭の授受を行う「事業者間精算」が行われます。

#### 業務フロー



#### 業務一覧

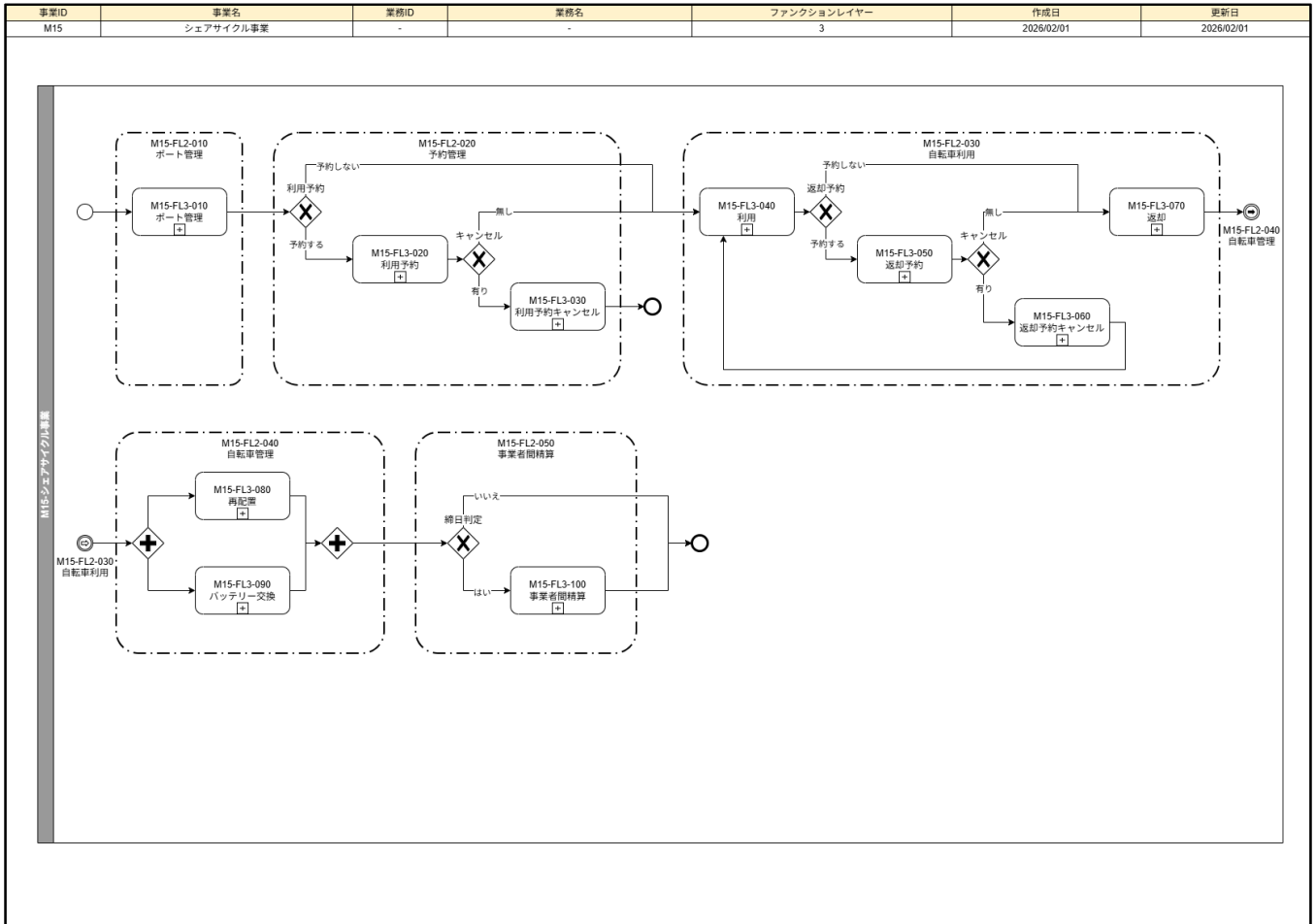
ID	詳細事業機能	内容
M15-FL2-010	ポート管理	シェアサイクルサービス間で共有ポート情報を相互に連携・共有するための業務。
M15-FL2-020	予約管理	自転車の利用予約、及びその予約キャンセルを行う業務。
M15-FL2-030	自転車利用	自転車を利用(解錠から返却まで)する一連の業務。
M15-FL2-040	自転車管理	自転車の物理的な管理に関する業務。
M15-FL2-050	事業者間精算	異なる事業者間でのポート相互利用などに関する精算業務。

## FL(ファンクションレイヤー)3

FL3フローは、FL2を主要業務群へ分解し、各業務の実行順序を把握することを目的とします。

事業者間で共有ポートの連携後、ユーザーが自転車の予約または直接利用を開始します。予約時はキャンセルも可能です。利用後は返却予約を経て、あるいは直接返却を行います。その後、事業者が自転車の再配置やバッテリー交換を行います。最終的に精算締日に共有ポートの利用実績を基に事業者間精算を行います。

### 業務フロー



### 業務一覧

ID	業務機能	内容
M15-FL3-010	ポート管理	シェアサイクルサービサー間で共有ポート情報を相互に連携・共有するための業務。
M15-FL3-020	利用予約	自転車の利用予約をする一連の業務。
M15-FL3-030	利用予約キャンセル	自転車の利用予約をキャンセルする一連の業務。
M15-FL3-040	利用	自転車を利用する一連の業務。
M15-FL3-050	返却予約	自転車の返却先ポートを事前に予約する一連の業務。
M15-FL3-060	返却予約キャンセル	シェアサイクルの返却予約をキャンセルする一連の業務。
M15-FL3-070	返却	ユーザーが自転車を返却する一連の業務。
M15-FL3-080	再配置	シェアサイクルサービサーAが、自社と共有ポート間の自社自転車台数の偏りを是正するために再配置を行う業務。 (事業者間で共有ポートの空き状況はオペレーションでカバーする想定)
M15-FL3-090	バッテリー交換	シェアサイクルサービサーAが、電動アシスト自転車のバッテリー交換やメンテナンスを行う業務。
M15-FL3-100	事業者間精算	シェアサイクル事業者AとBの間で、相互利用実績に基づいた精算データを作成し、送受信する業務。

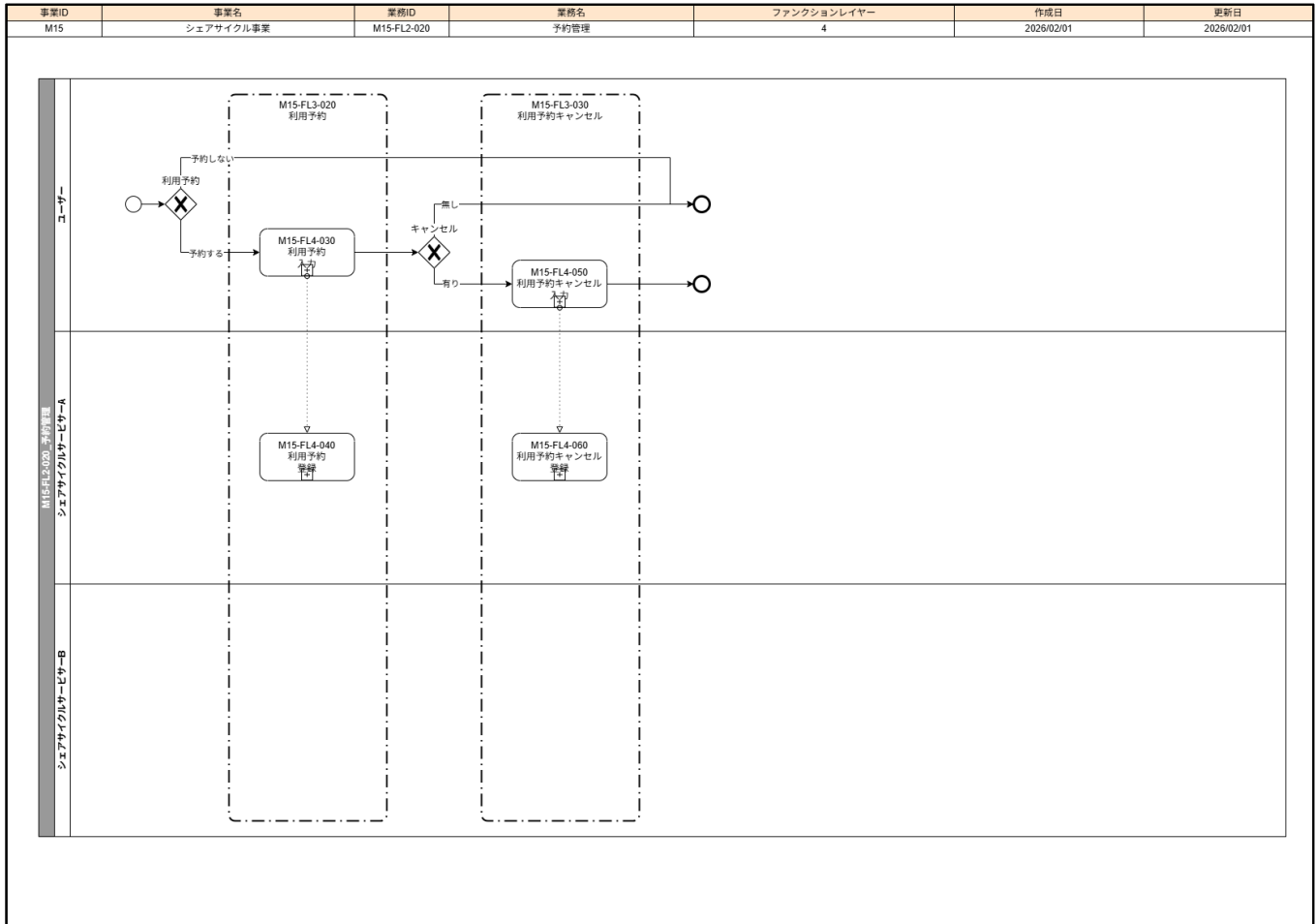
## FL(ファンクションレイヤー)4

FL4フローは、FL3のタスクを組織間の受け渡し単位に分解し、各組織の役割を把握します。

ユーザーの利用予約入力を起点に、シェアサイクルサービサー側で予約登録が実行されます。ユーザーが利用予約キャンセルを行った場合は、シェアサイクルサービサー側でキャンセル登録が処理されます。また、自転車の予約を行わずに、直接自転車の利用を開始することも可能です。なお、サービサーAとBの役割は状況次第で反転することも考えられます。

### 業務フロー

### ー利用予約・利用の業務を抜粋ー



### 業務一覧

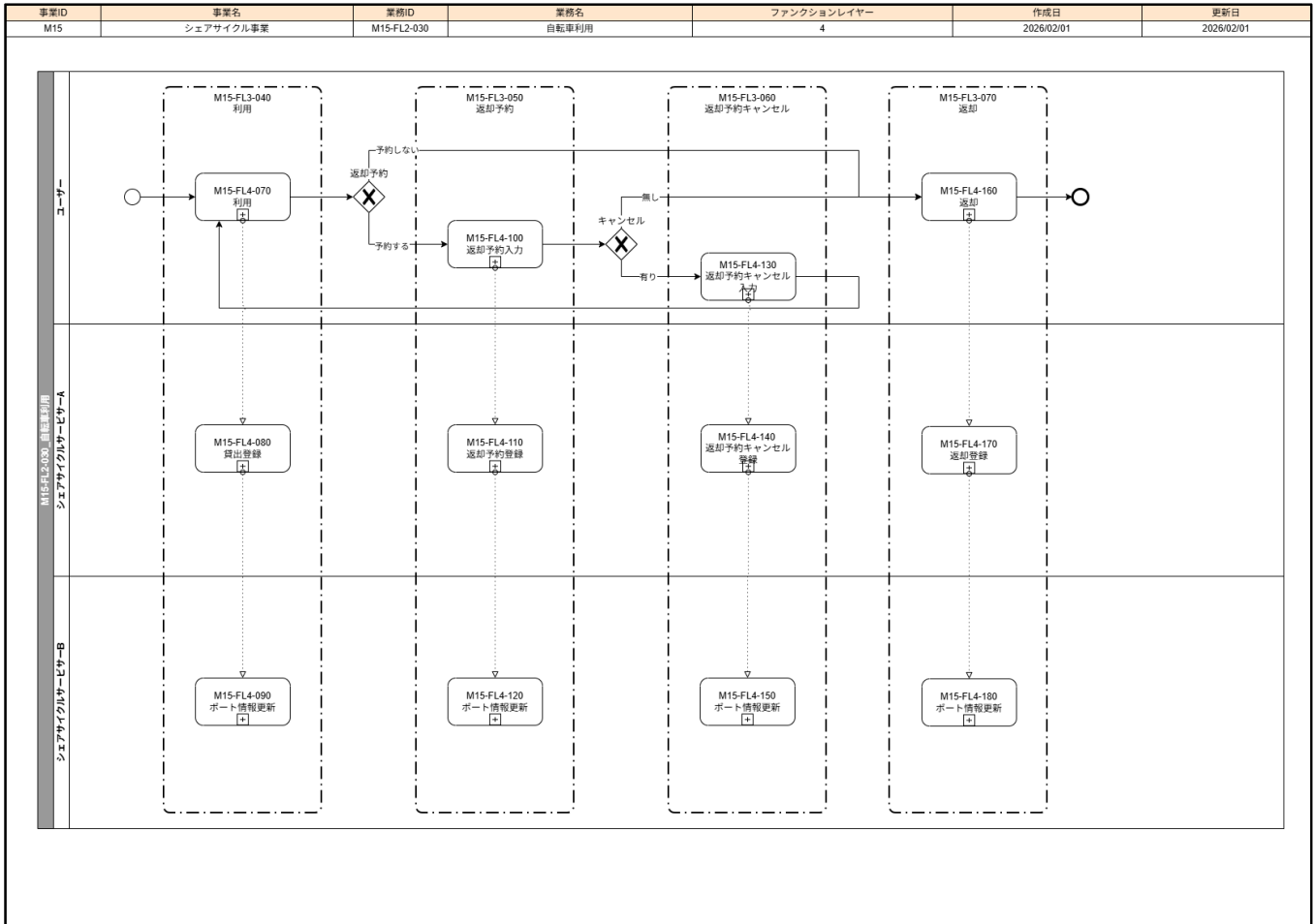
ID	詳細業務機能	内容
M15-FL4-030	利用予約入力	ユーザーが、ポーター一覧から利用するポートと自転車を選択し、シェアサイクルサービサー-Aへ予約を行う業務。
M15-FL4-040	利用予約登録	シェアサイクルサービサー-Aが、ユーザーの予約依頼を受け付け、予約登録を行う業務。
M15-FL4-050	利用予約キャンセル入力	ユーザーが、予約履歴を照会し、利用予約のキャンセルを行う。
M15-FL4-060	利用予約キャンセル登録	シェアサイクルサービサー-Aが、ユーザーの利用予約キャンセル入力に応じて、キャンセルを登録する業務。

## FL(ファンクションレイヤー)4

ユーザーによる利用、返却予約、返却の各操作を起点に、シェアサイクルサービサー側で貸出・返却・キャンセル等の登録処理と、ポートの共有先事業者へポート情報の更新の連携を行います。自転車の返却自体は予約を必須とせず、直接返却を行うことも可能です。なお、サービサーAとBの役割は状況次第で反転することも考えられます。

### 業務フロー

### —返却予約・返却の業務を抜粋—



### 業務一覧

ID	詳細業務機能	内容
M15-FL4-070	利用	ユーザーが自転車の解錠を申請し、乗車するまでの一連の業務。
M15-FL4-080	貸出登録	シェアサイクルサービサーAが、ユーザーの解錠リクエストを受けて解錠を行い、利用開始登録を行う業務。
M15-FL4-090	ポート情報更新	シェアサイクルシステムBが、連携された情報に基づきポート情報を更新する。
M15-FL4-100	返却予約入力	ユーザーが、アプリを通じて返却可能なポートを選択し、返却予約を行う一連の操作。
M15-FL4-110	返却予約登録	シェアサイクルサービサーAが、ユーザーからの返却予約入力に応じて、予約を登録する業務。
M15-FL4-120	ポート情報更新	シェアサイクルシステムBが、連携された情報に基づきポート情報を更新する。
M15-FL4-130	返却予約キャンセル入力	ユーザーが、予約履歴を確認し、返却予約のキャンセルを行う。
M15-FL4-140	返却予約キャンセル登録	シェアサイクルシステムAが、ユーザーのキャンセル依頼を受け付け、登録処理を行う一連の業務。
M15-FL4-150	ポート情報更新	シェアサイクルシステムBが、連携された情報に基づきポート情報を更新する。
M15-FL4-160	返却	ユーザーが自転車を手動で施錠し、アプリで返却リクエストを行うまでの一連の操作。
M15-FL4-170	返却登録	シェアサイクルシステムAが、ユーザーの返却リクエストを受け付け、料金計算や決済処理を行い、返却を完了させる一連の業務。 シェアサイクルシステムAはユーザーが操作しているアプリケーションに関するシステムを指す。
M15-FL4-180	ポート情報更新	シェアサイクルシステムBが、連携された情報に基づきポート情報を更新する。

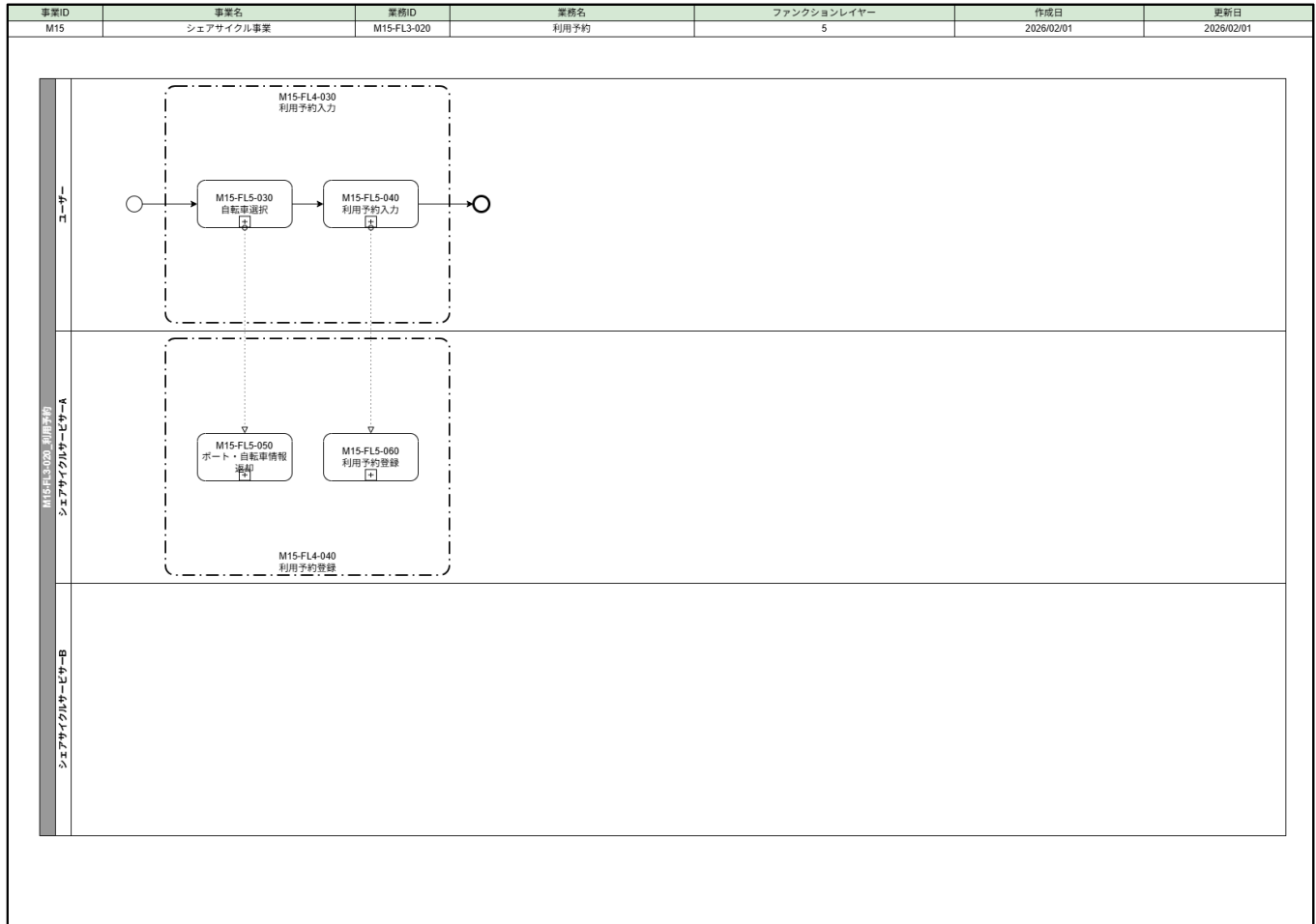
## FL(ファンクションレイヤー)5

FL5フローは、組織内のアクターの具体的な業務内容の把握を目的としたものです。

ユーザーが、シェアサイクルアプリからマップ上のポートから利用したい自転車を選択し、シェアサイクルサービサーはユーザーの選択したポート・自転車情報を返却します。続いてユーザーが利用予約入力を行い、シェアサイクルサービサー側で利用予約登録がされることで自転車の予約が完了する流れとなります。

### 業務フロー

### ー利用予約・利用の業務を抜粋ー



### 業務一覧

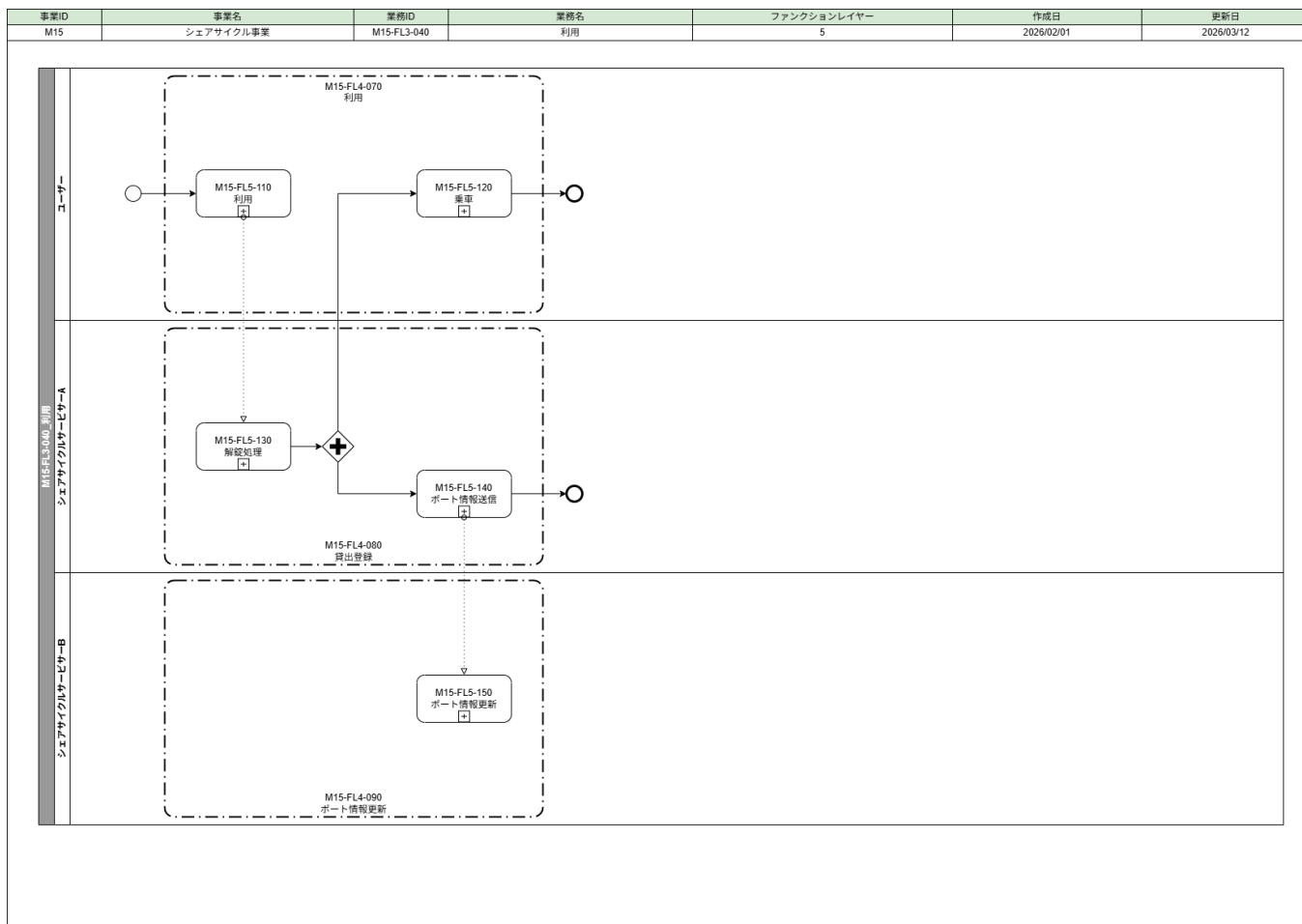
ID	単位作業	内容
M15-FL5-030	自転車選択	ユーザーが利用するポートと自転車を選択する業務。
M15-FL5-040	利用予約入力	ユーザーが利用する自転車の予約を行う業務。
M15-FL5-050	ポート・自転車情報返却	シェアサイクルサービサーAがポートと自転車情報を返却する業務。
M15-FL5-060	利用予約登録	シェアサイクルサービサーAが、ユーザーからの予約リクエストに応じて、利用予約の登録を行う業務。

## FL(ファンクションレイヤー)5

ユーザーの利用開始を起点に、シェアサイクルサービサー側で利用登録が実行されます。登録完了後、ユーザーの乗車プロセスと並行してシステム側ではポート共有先事業者へポート情報の更新の連携を行います。これにより、車両の貸出状態とポートの最新状況が他事業者と同期され、自転車の利用プロセスが完結する流れとなります。

### 業務フロー

### ー利用予約・利用の業務を抜粋ー



### 業務一覧

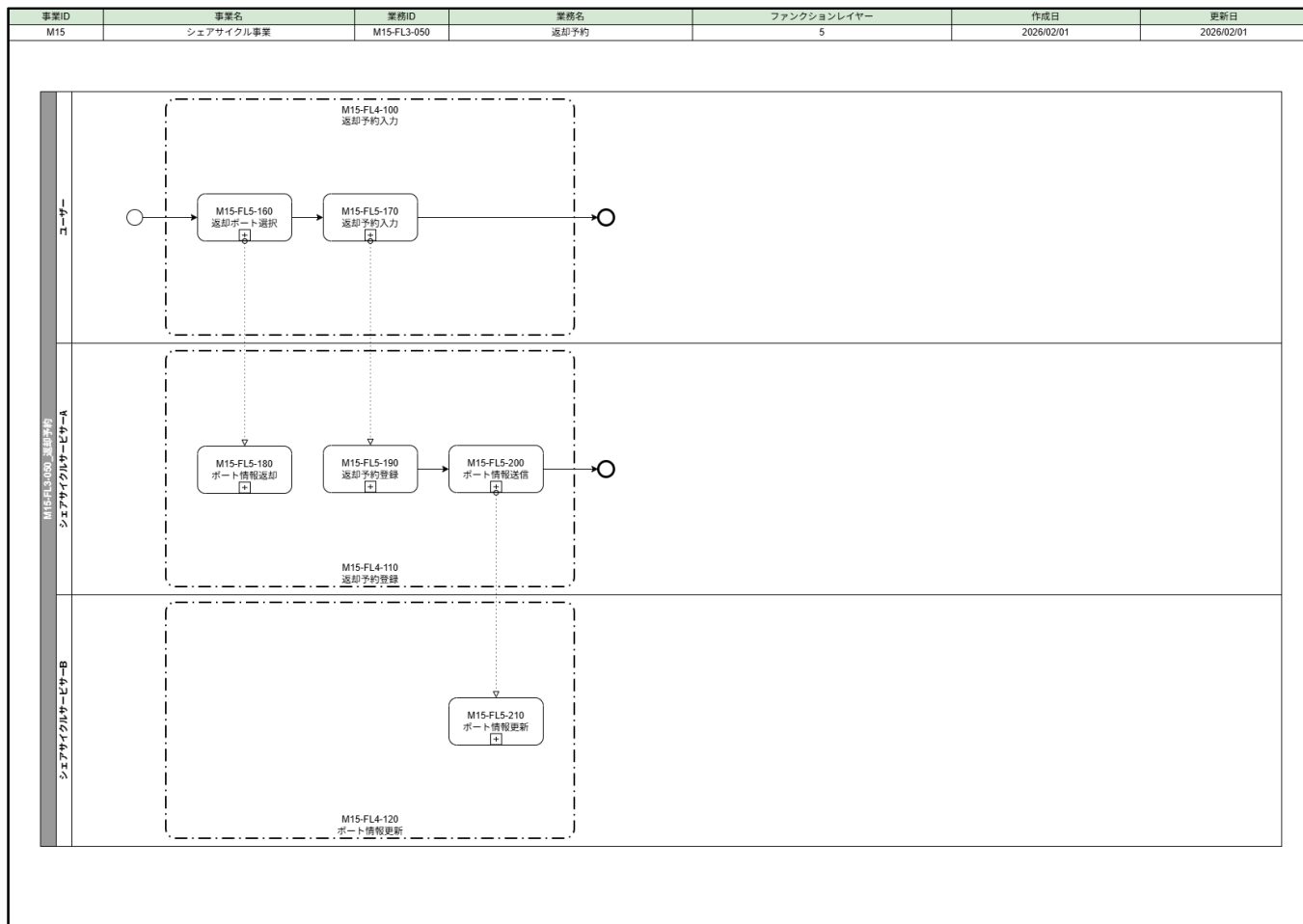
ID	単位作業	内容
M15-FL5-110	利用	ユーザーがアプリを通じて、自転車の解錠を依頼する。
M15-FL5-120	乗車	ユーザーが、スマートロックが物理的に解錠されたことを確認する。システム側で解錠処理が完了後、スマートロックが解錠される。
M15-FL5-130	解錠処理	シェアサイクルサービサーAが、ユーザーからの利用に応じて、スマートロックの解錠を行う。
M15-FL5-140	ポート情報送信	自転車の貸出によりポートの返却可能台数が変動するため、変動情報を他シェアサイクルサービサーへ連携する。
M15-FL5-150	ポート情報更新	シェアサイクルサービサーBが、シェアサイクルサービサーAからポートの返却可能台数の変動情報を取得し、シェアサイクルサービサーBの共有ポート情報の更新を行う。

## FL(ファンクションレイヤー)5

ユーザーが、シェアサイクルアプリからマップ上の返却先ポートを選択し、シェアサイクルサービサーが対象のポート情報を返却します。続いてユーザーが返却予約入力を行うと、シェアサイクルサービサー側で返却予約登録および他事業者へポート情報送信が実行されます。これによりシステム内のポート情報が更新され、返却予約プロセスが完了する流れとなります。

### 業務フロー

### —返却予約・返却の業務を抜粋—



### 業務一覧

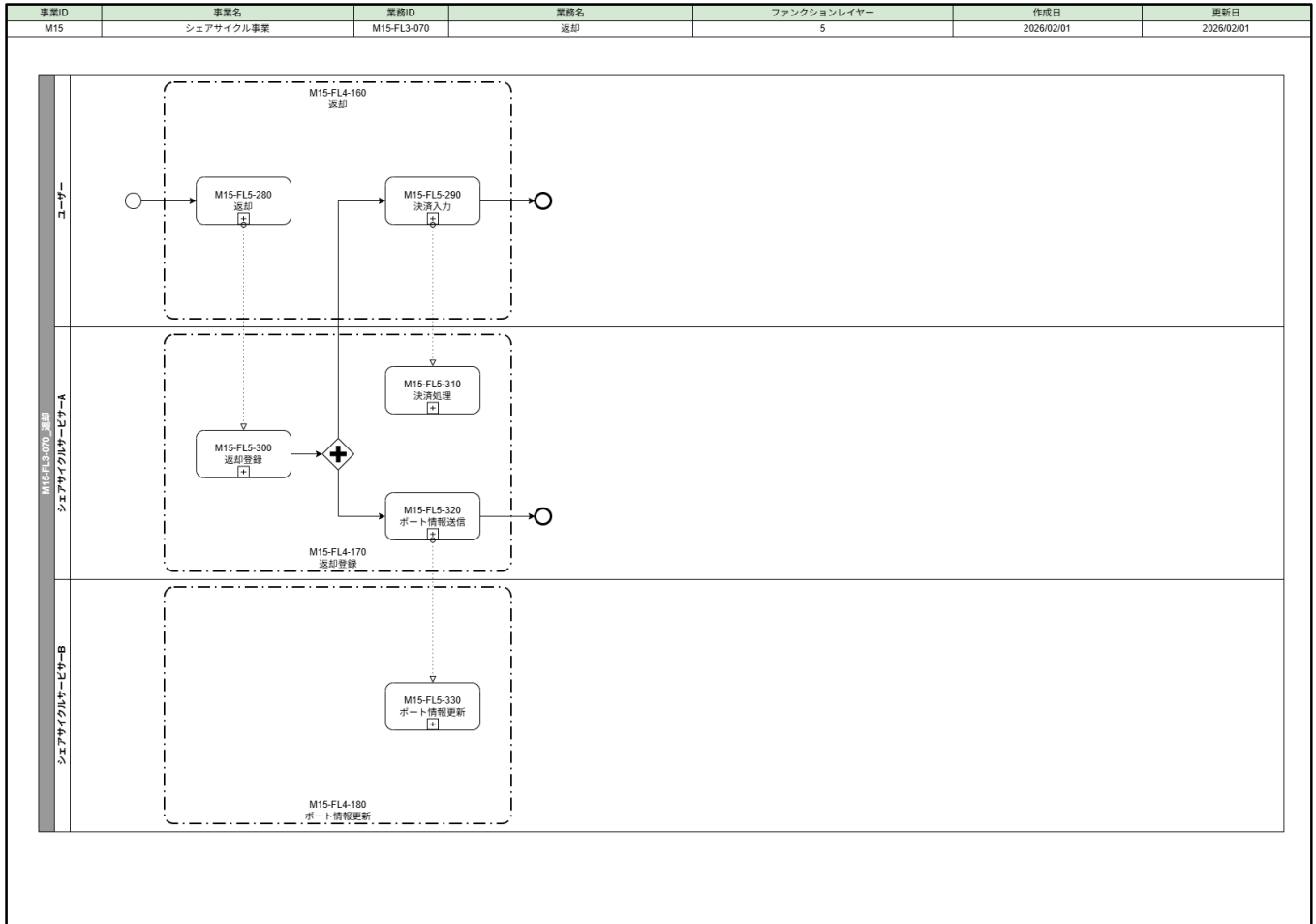
ID	単位作業	内容
M15-FL5-160	返却ポート選択	ユーザーが、アプリを通じて返却可能なポートを選択し、返却予約を行う一連の操作。
M15-FL5-170	返却予約入力	ユーザーが、返却ポートの予約依頼を行う。
M15-FL5-180	ポート情報返却	シェアサイクルサービサーAが、ユーザーにポートと自転車情報を返却する業務。
M15-FL5-190	返却予約登録	シェアサイクルサービサーAが、ユーザーから返却予約入力に応じて、予約登録を行い、ポート情報のDBを更新する。
M15-FL5-200	ポート情報送信	自転車の返却予約によりポートの返却可能台数が変動する為、変動情報を他事業者へ連携する。
M15-FL5-210	ポート情報更新	シェアサイクルサービサーBが、シェアサイクルサービサーAからポートの返却可能台数の変動情報を取得し、共有ポート情報の更新を行う。

## FL(ファンクションレイヤー)5

ユーザーの返却操作を起点に、シェアサイクルサービス側で返却登録が実行されます。登録完了後、決済入力とポート情報送信が並行して行われ、システム側で決済処理とポート情報の更新がそれぞれ実施されます。これにより、共有ポートの最新状況が事業者間で共有され、一連の返却プロセスが完了します。

### 業務フロー

### 一返却予約・返却の業務を抜粋



### 業務一覧

ID	単位作業	内容
M15-FL5-280	返却	ユーザーが自転車を手動で施錠し、アプリで返却リクエストを行うまでの一連の操作
M15-FL5-290	決済入力	ユーザーが請求書を確認し、アプリを通じて利用料金の決済を実施する。リクエストパラメータに貸出IDとユーザーIDを指定し、システムへリクエストする。システムでの決済処理完了を持って一連の自転車利用が完了する。
M15-FL5-300	返却登録	ユーザーから返却リクエストを受け、自転車の返却処理を実行する。リクエストパラメータとして貸出IDとユーザーIDを指定し、DBに格納されているポート情報を更新する。
M15-FL5-310	決済処理	シェアサイクルサービスAがユーザーの自転車利用実績から利用料金を算出し、決済処理を行う業務。
M15-FL5-320	ポート情報送信	自転車の返却処理を実施したタイミングで、最新のポート情報をシェアサイクルシステムBへ送信する。MQTTを利用してシステム間連携を実施する。シェアサイクルシステムBはデータ連携先のシステムを指す。
M15-FL5-330	ポート情報更新	シェアサイクルサービスBが、シェアサイクルサービスAからポートの返却可能台数の変動情報を取得し、シェアサイクルサービスBの共有ポート情報の更新を行う。

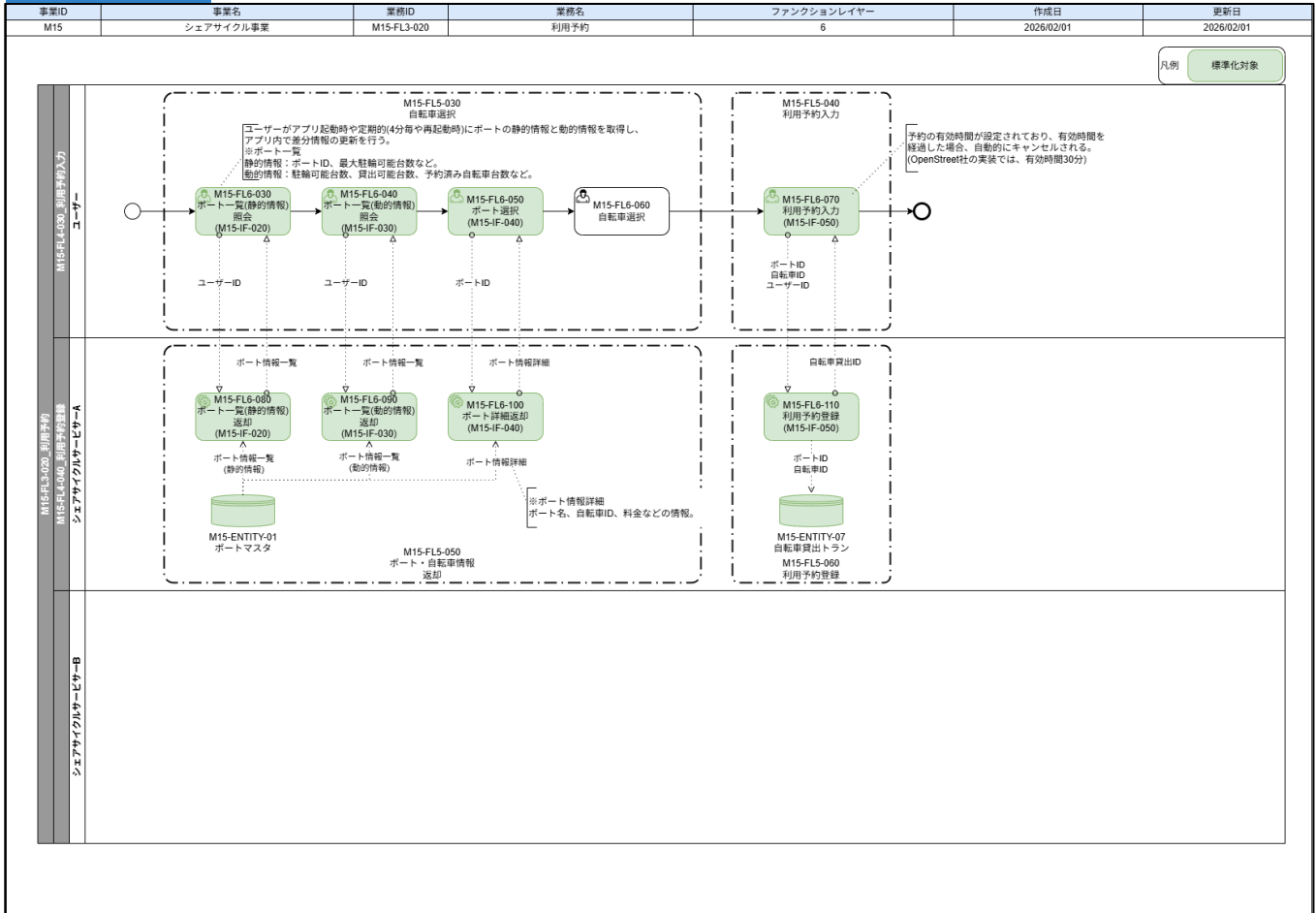
## FL(ファンクションレイヤー)6

FL6フローは、手順レベルまで業務を分解し、アクター間の情報伝達の把握を目的とします。

ユーザーによるポート照会を起点に、ポート一覧の静的・動的情報が返却されます。ポート選択後、詳細情報のリクエストに対し、詳細データが応答されます。自転車選択を経てユーザーが利用予約入力を行うと、APIを介してシェアサイクルサービサーへ予約登録リクエストが送信され、貸出IDを含むレスポンスが返却されることで予約が完了します。

### 業務フロー

### —利用予約・利用の業務を抜粋—



### 業務一覧

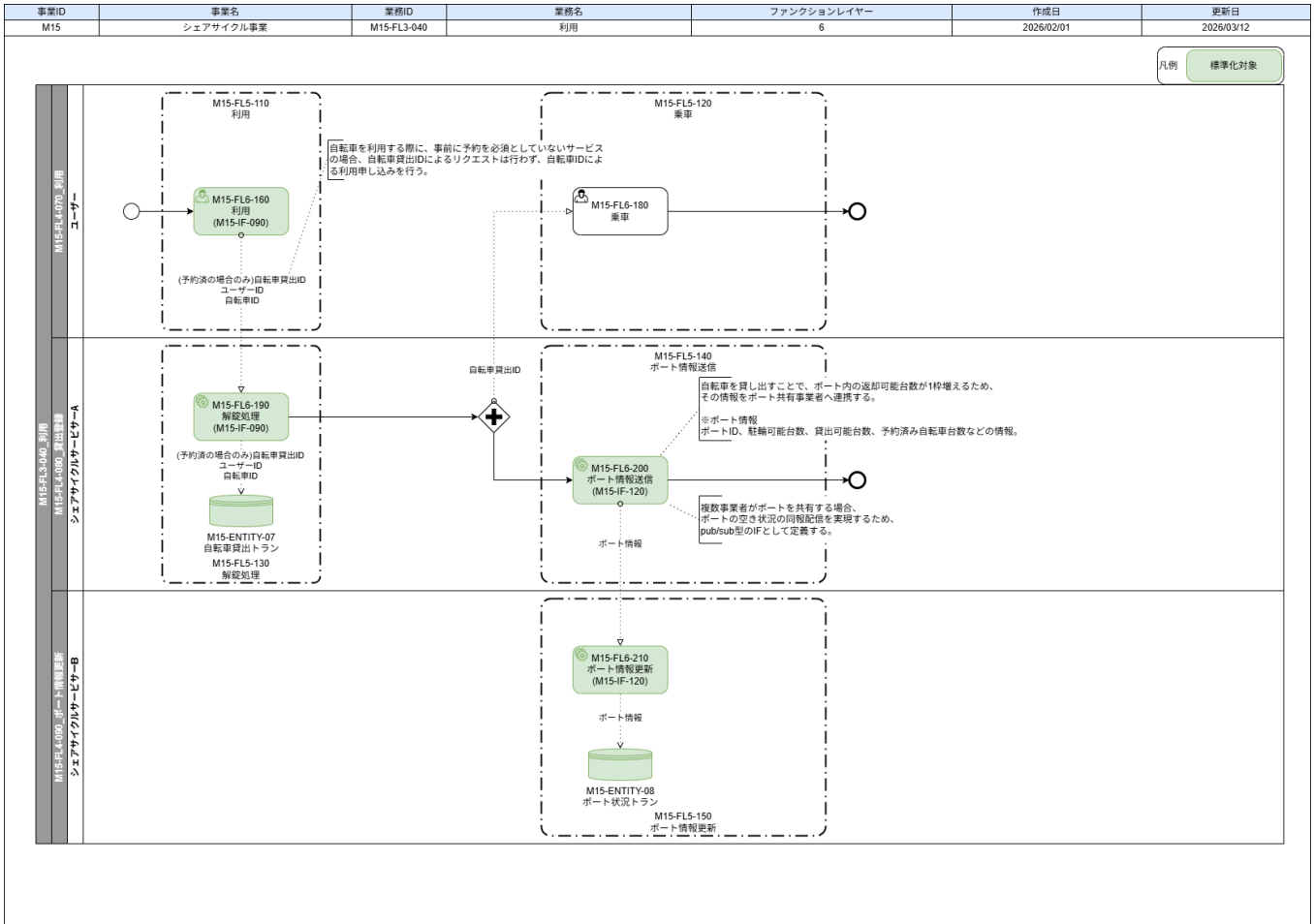
ID	要素作業	内容
M15-FL6-030	ポート一覧(静的情報)照会	ユーザーがシェアサイクルアプリを起動し、ポート情報一覧の静的情報(ポートID、最大駐輪可能台数など)を取得する。
M15-FL6-040	ポート一覧(動的情報)照会	ユーザーがシェアサイクルアプリを起動し、ポート情報一覧の動的情報(駐輪可能台数、貸出可能台数など)を取得する。
M15-FL6-050	ポート選択	ユーザーが、マップ上のポート一覧から利用したいポートを選択する。選択したポートに紐づいているポートIDをシステムへリクエストし、ポート詳細情報および自転車情報を取得する。
M15-FL6-060	自転車選択	ユーザーが、選択したポート内の利用可能な自転車を選択する。
M15-FL6-070	利用予約入力	ユーザーが、利用する自転車の予約を確定する。ポートID、自転車ID、ユーザーIDを送信し、システムでの予約依頼する。予約確定後、シェアサイクルサービサーAから貸出予約IDを取得する。
M15-FL6-080	ポート一覧(静的情報)返却	シェアサイクルサービサーAが、ユーザーからのリクエストに応じて、ポート情報一覧の静的情報(ポートID、最大駐輪可能台数など)を返却する。
M15-FL6-090	ポート一覧(動的情報)返却	シェアサイクルサービサーAが、ユーザーからのリクエストに応じて、ポート情報一覧の動的情報(駐輪可能台数、貸出可能台数など)を返却する。
M15-FL6-100	ポート詳細返却	シェアサイクルサービサーAが、ユーザーが指定したポートIDのリクエストに応じて、ポートの詳細情報(自転車情報など)を返却する。
M15-FL6-110	利用予約登録	シェアサイクルサービサーAが、ユーザーが指定した自転車IDのリクエストに応じて、自転車の利用予約の登録を行う。登録完了後、貸出予約IDを発行し、ユーザーへ返却する。

## FL(ファンクションレイヤー)6

ユーザーの利用操作を起点に、シェアサイクルサービサーが利用登録を行い、自転車の解錠を行います。自転車を貸し出すことで、共有ポート内の返却可能台数が1枠増えるため、ポートの更新情報を、APIを通じてポート共有事業者へ連携します。

### 業務フロー

### —利用予約・利用の業務を抜粋—



### 業務一覧

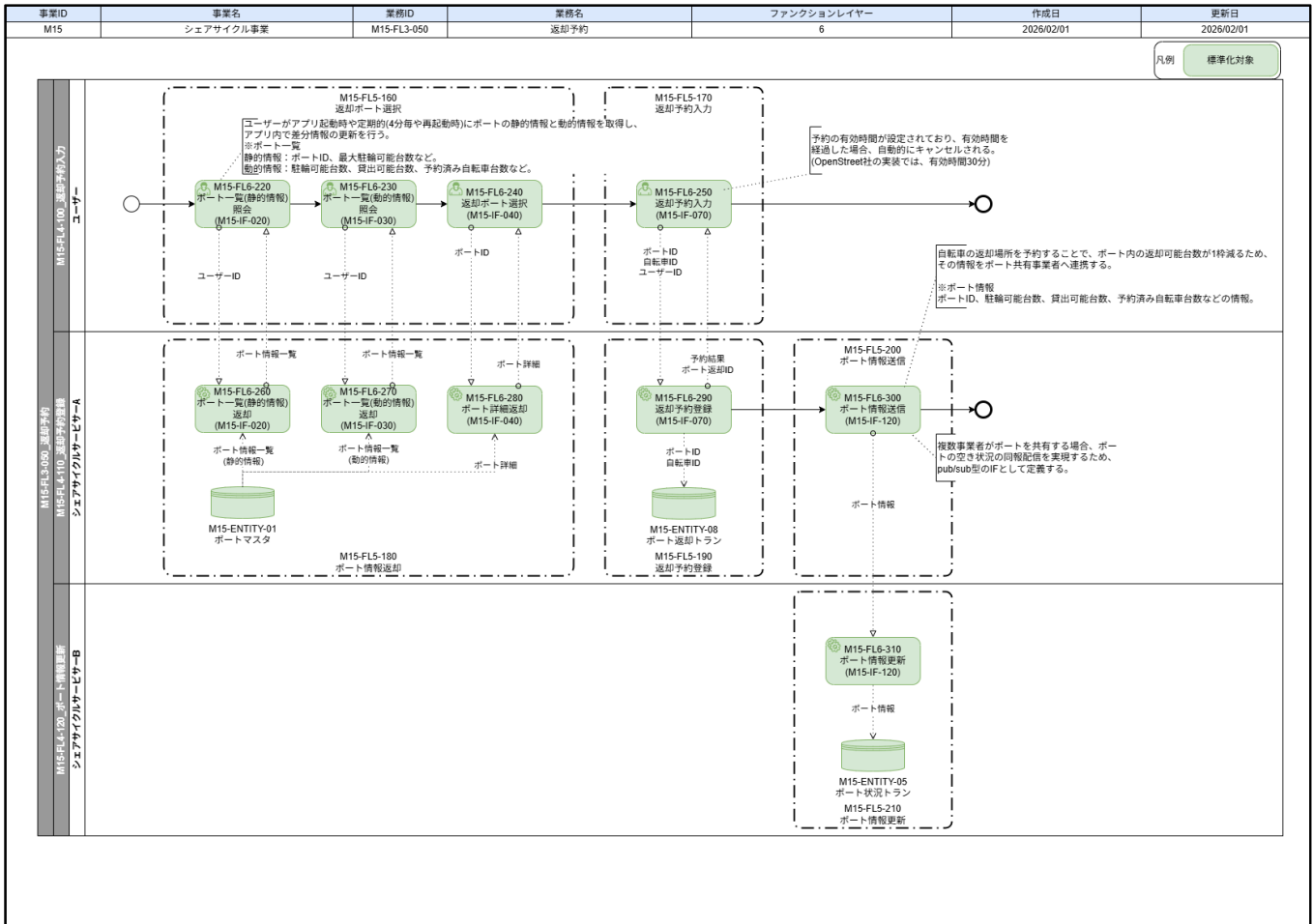
ID	要素作業	内容
M15-FL6-160	利用	ユーザーがアプリを通じて、自転車の解錠を依頼する。貸出予約ID(※事前に利用予約をしていた場合)とユーザーIDをシェアサイクルサービサーへ送信する。
M15-FL6-180	乗車	ユーザーが、解錠された自転車に乗って利用を開始する。
M15-FL6-190	解錠処理	シェアサイクルサービサーAが、ユーザーからの利用に応じて、スマートロックの解錠を行う。解錠に伴いDBにポート情報の更新を行う。
M15-FL6-200	ポート情報送信	自転車の貸出によりポートの返却可能台数が変動するため、変動情報を他シェアサイクルサービサーへ連携する。
M15-FL6-210	ポート情報更新	シェアサイクルサービサーBが、シェアサイクルサービサーAからポートの返却可能台数の変動情報を取得し、シェアサイクルサービサーBの共有ポート情報の更新を行う。

## FL(ファンクションレイヤー)6

ユーザーが、シェアサイクルアプリからポート一覧を確認し、返却先ポートを選択します。その後、返却先ポートの予約登録リクエストを行うとシェアサイクルサービサーが返却予約登録を行い、それに伴い、共有ポート内の返却可能台数が1枠減るため、API経由でポート共有事業者へ連携を行い、返却予約の一連の業務が完了します。

### 業務フロー

### 一返却予約・返却の業務を抜粋



### 業務一覧

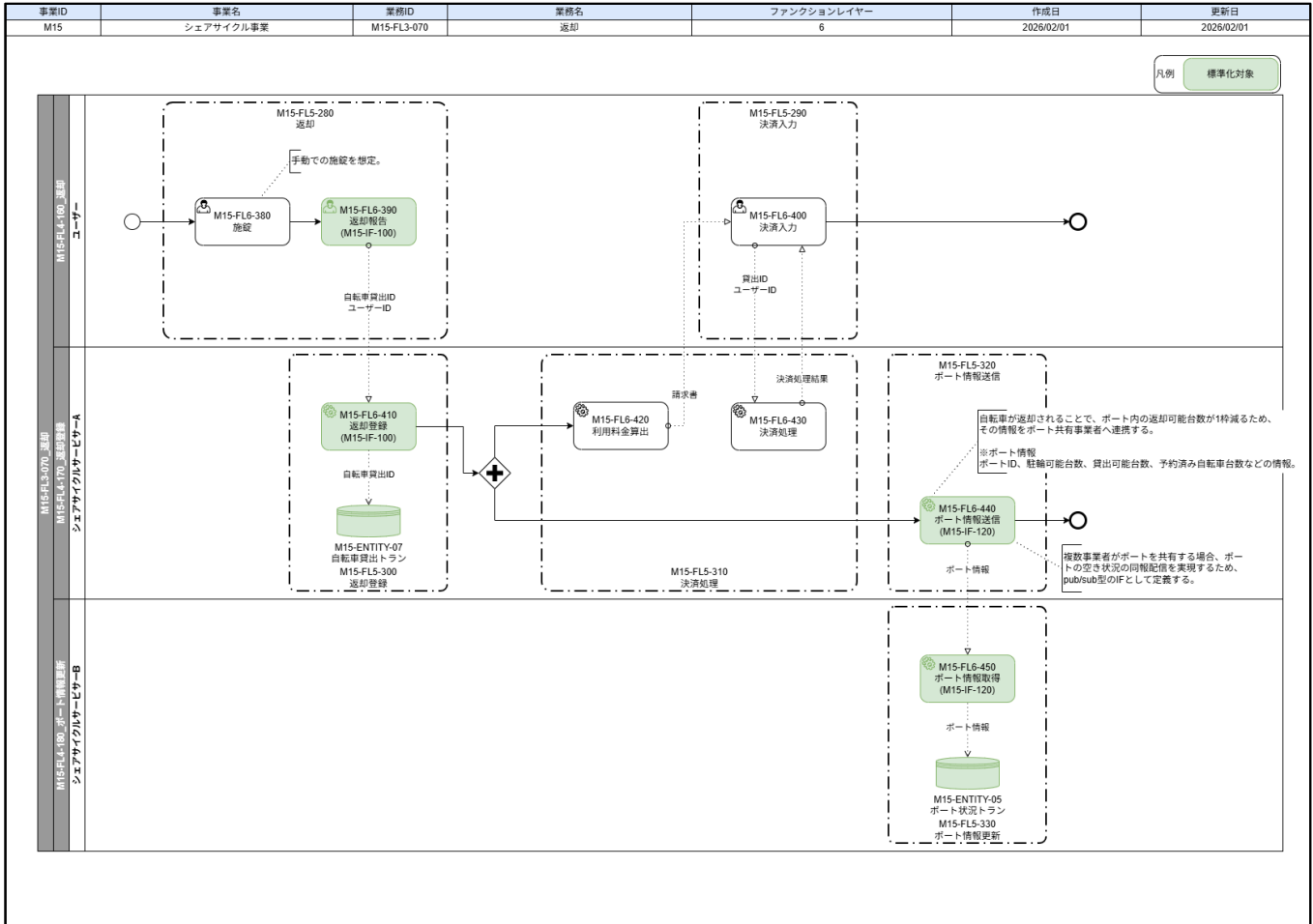
ID	要素作業	内容
M15-FL6-220	ポート一覧(静的情報)照会	ユーザーがシェアサイクルアプリを起動し、ポート情報一覧の静的情報(ポートID、最大駐輪可能台数など)を取得する。
M15-FL6-230	ポート一覧(動的情報)照会	ユーザーがシェアサイクルアプリを起動し、ポート情報一覧の動的情報(駐輪可能台数、貸出可能台数など)を取得する。
M15-FL6-240	返却ポート選択	ユーザーが、ポート一覧から利用したいポートを選択する。
M15-FL6-250	返却予約入力	ユーザーが、返却ポートの予約申請を行う。ポートID、貸出ID、ユーザーIDを送信し、システムでの予約を確定させる。 予約確定後は予約結果と返却予約IDを取得する。
M15-FL6-260	ポート一覧(静的情報)返却	シェアサイクルサービサーAが、ユーザーからのリクエストに応じて、ポート情報一覧の静的情報(ポートID、最大駐輪可能台数など)を返却する。
M15-FL6-270	ポート一覧(動的情報)返却	シェアサイクルサービサーAが、ユーザーからのリクエストに応じて、ポート情報一覧の動的情報(駐輪可能台数、貸出可能台数など)を返却する。
M15-FL6-280	ポート詳細返却	シェアサイクルサービサーAが、ユーザーが指定したポートIDのリクエストに応じて、ポートの詳細情報(自転車情報など)を返却する。
M15-FL6-290	返却予約登録	シェアサイクルサービサーAが、ユーザーから返却予約入力に応じて、予約登録を行い、ポート情報のDBを更新する。 リクエストパラメータとして、ポートID、貸出ID、ユーザーIDを受け取り、その内容をDBに格納する。 登録完了後、返却予約IDを発行し、予約結果と一緒に返却する。
M15-FL6-300	ポート情報送信	自転車の返却予約によりポートの返却可能台数が変動する為、変動情報を他事業者へ連携する。
M15-FL6-310	ポート情報更新	シェアサイクルサービサーBが、シェアサイクルサービサーAからポートの返却可能台数の変動情報取得し、の共有ポート情報の更新を行う。

## FL(ファンクションレイヤー)6

ユーザーによる施錠と返却報告を起点に、APIを介してサービサーへ返却登録リクエストが送信されます。これを受け、シェアサイクルシステム側では利用料金の算出と決済処理が実行されます。同時に、共有ポート内の返却可能台数が1枠減るため、APIを通じてポート共有事業者へポート状況の連携を行い、一連の返却業務が完了します。

### 業務フロー

### 一返却予約・返却の業務を抜粋



### 業務一覧

ID	要素作業	内容
M15-FL6-380	施錠	ユーザーが、自転車の鍵を手動で施錠する。 (アプリ操作ではなく物理での作業を想定)
M15-FL6-390	返却報告	ユーザーがアプリを通じて、返却リクエストする。 リクエストパラメータに貸出IDとユーザーIDを指定し、システムへリクエストする。
M15-FL6-400	決済入力	ユーザーが請求書を確認し、アプリを通じて利用料金の決済を実施する。 リクエストパラメータに貸出IDとユーザーIDを指定し、システムへリクエストする。システムでの決済処理完了を持って一連の自転車利用が完了する。
M15-FL6-410	返却登録	ユーザーから返却リクエストを受け、自転車の返却処理を実行する。 リクエストパラメータとして貸出IDとユーザーIDを指定し、DBに格納されているポート情報を更新する。
M15-FL6-420	利用料金算出	利用時間、ポート情報などに基づき利用料金を算出する。 ユーザーには請求書を返却する。
M15-FL6-430	決済処理	ユーザーから決済リクエストを受け付けシステムで、決済処理を実行する。 リクエストパラメータに設定される貸出IDとユーザーIDに基づき決済を実施する。処理完了後、決済処理の結果を返却する。
M15-FL6-440	ポート情報送信	自転車の返却処理を実施したタイミングで、最新のポート情報をシェアサイクルシステムBへ送信する。 MQTTを利用してシステム間連携を実施する。シェアサイクルシステムBはデータ連携先のシステムを指す。
M15-FL6-450	ポート情報取得	シェアサイクルサービサーBが、シェアサイクルサービサーAからポートの返却可能台数の変動情報を取得し、シェアサイクルサービサーBの共有ポート情報の更新を行う。

### 3.6. ポート共有で得た収益を互いに清算する

ポート共有で得た収益をお互いに清算する考え方を示します。

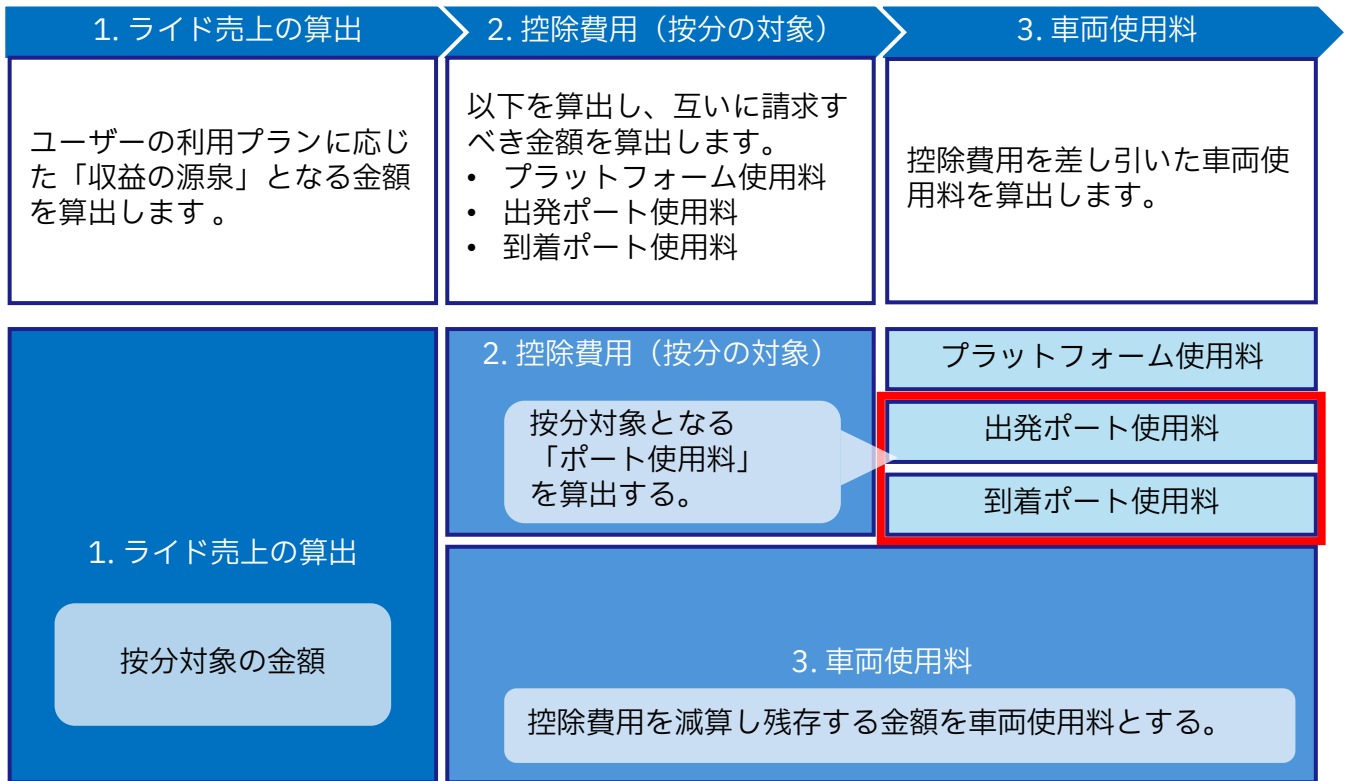
#### 按分・清算に関する用語の定義

互いに得た収益の按分・清算に関する用語の定義は以下のとおりとします。

No	用語	説明
1	ライド	本件車両の1回の貸出から返却までの利用単位をいう。
2	ライド売上	ユーザーに発生する1「ライド」の本件サービス利用料をいう。
3	プラットフォーム使用料	ユーザーが車両を予約する際に利用したアプリの事業者に応じて、「ライド売上」のうちその事業者に帰属する部分をいう。
4	プラットフォーム使用料按分率	「ライド売上」に対する「プラットフォーム使用料」の比率をいう。
5	車両使用料	「ライド売上」のうち、当該ライドにかかる車両を提供する一方当事者に帰属すべき使用料をいう。
6	車両使用料按分率	「ライド売上」に対する「車両使用料」の比率をいう。
7	出発ポート	ユーザーが本件車両の貸出を受けるために使用したポートをいう。
8	出発ポート使用料	「ライド売上」のうち、「出発ポート」を管理する一方当事者に帰属すべき使用料をいう。
9	出発ポート使用料按分率	「ライド売上」に対する「出発ポート使用料」の比率をいう。
10	到着ポート	ユーザーが本件車両を返却するために使用したポートをいう。
11	到着ポート使用料	ライド売上のうち、「着ポート」を管理する一方当事者に帰属すべき使用料をいう。
12	到着ポート使用料按分率	「ライド売上」に対する「到着ポート使用料」の比率をいう。
13	ポート使用料	「出発ポート売上」及び「到着ポート売上」の合計をいう。
14	ポート使用料按分率	「出発ポート売上按分率」及び「到着ポート売上按分率」の合計をいう。

## 収益按分の考え方

ポート資料料金の按分の考え方は以下のとおりとなります。



## 按分のパターン

ポート共有で発生するパターンは以下のとおりとなります。

プラットフォーム	出発ポート	到着ポート	車両
A社アプリ	A社ポート	A社ポート	A社車両
A社アプリ	A社ポート	B社ポート	A社車両
A社アプリ	B社ポート	B社ポート	A社車両

A社アプリからはA社車両のみを借りることができます。  
B社への支払いは、A社車両がB社ポートを利用した場合のみとなります。

## 1. ライド売上の算出

ユーザーの利用プランに応じた「収益の源泉」となる金額。

### 都度払いプラン：

$$1\text{ライドの利用料(税込)} \times (1 - \text{決済処理手数料率})$$

例) 利用料(160円(税込))  $\times$  (1 - 決済処理手数料(5%)) = 152円

### 定額課金プラン：

$$\{(\text{定額料金(税込)} \div \text{ライド回数}) + \text{延長料金(税込)}\} \times (1 - \text{決済処理手数料率})$$

決済処理手数料率は事業者間で協議の基で設定する。

決済処理手数料率を変更する場合、変更を行う当事者は他方当事者に対して事前の書面通知を行い、決済処理手数料率の変更理由を示す資料を提供しなければならない。

## 2. 控除費用（按分の対象）

### ライド売上から差し引かれる各事業者の取り分

#### プラットフォーム使用料

ユーザーが車両を予約する際に利用したアプリの事業者に帰属する金額。

#### プラットフォーム使用料：

$$\text{ライド売上} \times \text{プラットフォーム使用料按分率}$$

※ 1円未満は切り上げるものとする。

例) ライド売上(152円)  $\times$  プラットフォーム使用料按分率(10%) = 16円

プラットフォーム使用料按分率は、事業者間で協議の基で設定する。

プラットフォーム使用料は、提携プロダクトを利用したライドにのみ適用される。

#### ポート使用料の算出

ポート使用料は、以下の計算式を用いて算出する。

#### 出発ポート使用料：

請求の対象

$$\text{出発ポート使用料(税込)} = \text{ライド売上} \times \text{出発ポート按分率}$$

※ 1円未満は切り上げるものとする。

例) ライド売上(152円)  $\times$  出発ポート配分率(15%) = 23円

#### 到着ポート使用料：

請求の対象

$$\text{到着ポート使用料(税込)} = \text{ライド売上} \times \text{到着ポート按分率}$$

※ 1円未満は切り上げるものとする。

例) ライド売上(152円)  $\times$  到着ポート配分率(15%) = 23円

出発ポート按分率及び到着ポート按分率は、事業者間で協議の基で設定する。

出発ポート使用料および到着ポート使用料は、それぞれの出発ポート及び到着ポートの管理者に分配される。

### 3. 車両使用料

車両使用料は、以下の計算式を用いて算出する。

#### 車両使用料：

車両使用料（税込）＝ライド売上－プラットフォーム使用料－ポート使用料  
※ 1円未満は切り上げるものとする。

例) ライド売上(152円)－プラットフォーム使用料(16円)－ポート使用料(23円＋23円)＝90円

車両使用料は、当該車両を管理する当事者に帰属する。

本事業においては、予約を行うアプリと車両を管理する事業者が同一となるため、車両使用料はアプリを管理する事業者側に計上される。

### 算出した売上の支払いルール（案）

算出した売上の支払いについては、以下のとおり行うものとする。

#### データ提供（月末＋10営業日以内）

利用料を受領した当事者は、他方当事者が請求書を発行するために必要なデータを提供する。

#### 請求書発行（データ受領＋5営業日以内）

データを受領した当事者は、内容を確認の上、請求書を発行する。

#### 支払実行（請求書受領＋30日以内）

請求書の受領日から30日以内に、指定銀行口座へ電信送金にて支払う。

#### その他規定

**消費税**：支払金額には消費税および地方消費税を明細付きで含める。

**手数料**：銀行振込手数料は支払当事者の負担とする。

**紛争解決**：請求金額またはデータに不一致がある場合、両当事者は、請求または請求書の受領日から2か月以内に、相互に解決するか、または独立した第三者機関を関与させて解決することに合意する。

## シェアサイクルサービス間の連携を実装する際の留意点

— 日本の制度・都市環境に即した運用要件への対応 —

- 本プロジェクトでは、国内事業者と海外で展開実績を持つシェアサイクル事業者との連携実証を通じ、API仕様の整合だけでは解決できない運用・非機能面の要件が明確になった。

### 1. 日本の事情に即したシェアサイクル運用上の前提

- 公有地ポートを前提とした厳格なキャパシティ管理  
公有地利用においては、ポート満車や車両溢れを発生させないことが強く求められる
- リアルタイム性を重視した運用  
都市部では利用状況が短時間で変化するため、即時性の高い情報更新が不可欠
- 利用者トラブルが行政対応・事業継続に直結  
返却不可や放置車両は、苦情や指導につながる可能性がある

### 2. 実証で明らかになった主な留意点（非機能要件）

#### (1) リアルタイム同期要件に対する前提の違い

日本の公有地ポート運用では車両溢れを確実に防ぐ必要があり、予約・利用開始・返却などの状態を1秒未満で同期する前提で設計することが求められる。

#### (2) イベント即時通知を前提とした連携の必要性

リアルタイム性を成立させるには、状態変化を検知した瞬間に相手へ伝えるイベント駆動連携（Webhook等）が実務上の要件となる。

#### (3) 自動終了・強制返却といった運用ルールへの対応

連携では通常操作だけでなく、120分経過の自動終了や15分アイドリングによる強制返却など、相手システム側で発生する強制イベントまで整合させる必要がある。

これらはAPI仕様というよりも制度・現場運用と密接に結びついた制御要件当初のAPI対応スコープには含まれておらず追加的な仕様整理と工数確保が必要となった

### • 4. 課題の整理（原因の捉え方）

本件の本質は技術的に不可能というより、日本の運用条件（公有地ポートでの溢れ防止、秒単位同期、強制イベント取り扱い）を前提とした要件共有が初期に不足した点にある。標準APIへの“対応”を機能実装に限定して捉えたことで、非機能・運用要件の差分が後工程で顕在化した。

### • 5. 今後に向けた示唆

海外展開実績のある事業者と連携する際は、API仕様提示に加え、日本の事情に即した非機能・運用要件を契約・計画段階で明文化し、リアルタイム同期水準、イベント通知方式、強制終了イベントの扱いを合意しておくべきである。



**COMmmONS**

by MLIT

シェアサイクルポート共有API標準仕様書 ガイダンス

発行日：2026年2月

発行元：国土交通省 総合政策局 公共交通政策部門 モビリティサービス推進課

受託者：パシフィックコンサルタンツ株式会社

OpenStreet株式会社