

報告書別添

# 本実証におけるロボットサービス運行基盤 の仕様解説

2024年3月

# 目次

章	節	項
1. はじめに	1.1 目的	4
	1.2 適用範囲	5
2. ロボットサービス運行基盤の仕様概要	2.1 共通データ基盤（空間情報管理）	7
	2.2 ロボット管理ミドルウェア（データ連携・蓄積・分析）	12
	2.3 ロボット管理ミドルウェア（運行管理・交通管制）	15
	2.4 構成要素間のインターフェース仕様	18
	2.5 データモデル仕様	22
	2.6 API や共通ライブラリのソースコード等	23
3. ロボットサービス運行基盤の運用概要	3.1 ロボットサービス運行基盤を用いた分散協調運行の概要	28
	3.2 走行環境地図・走行ルールの共有	30
	3.3 一時的進入禁止領域の共有	33
	3.4 位置情報の共有	34
	3.5 経路占有情報の共有	36
	3.6 屋内外横断走行時の地図乗り換え	40
	3.7 分散協調運行に関する課題・今後の展望	41

# 1. はじめに

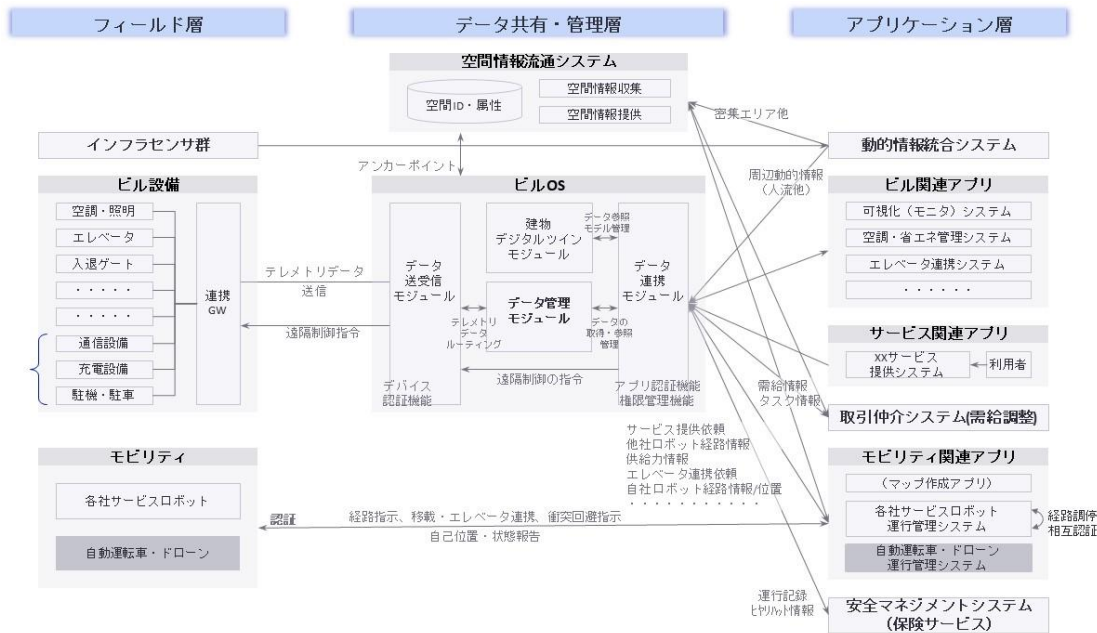
1.1 目的

1.2 適用範囲

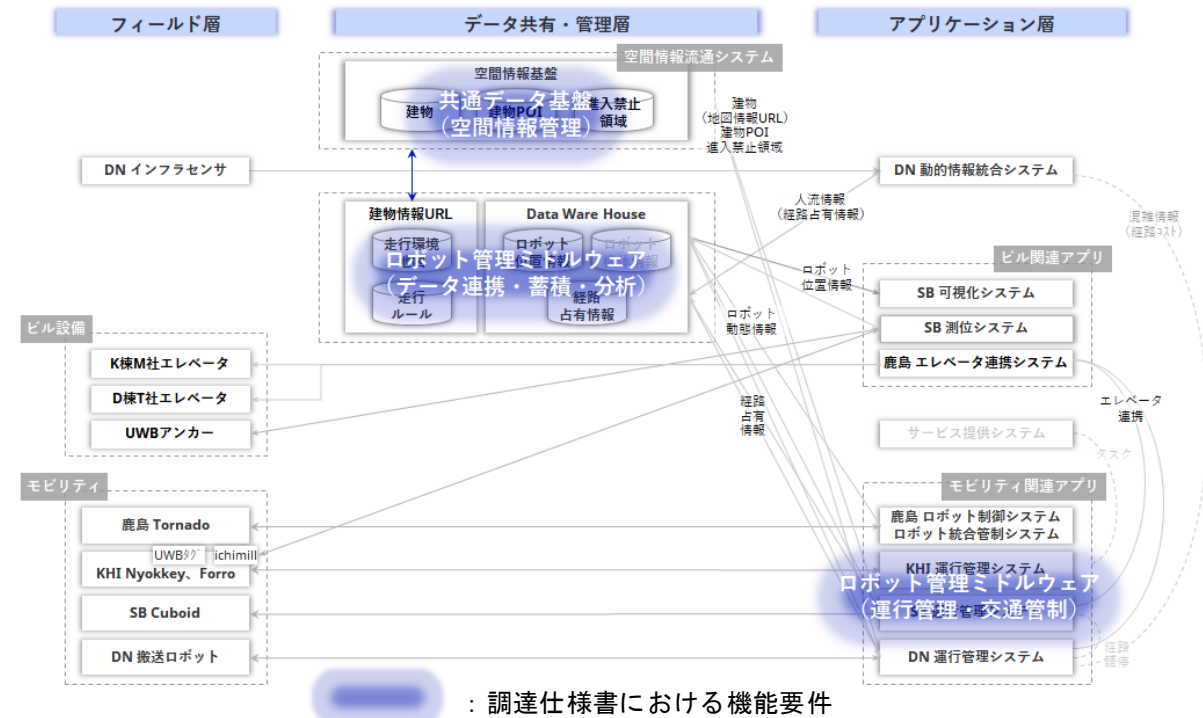
# 1.1 目的

報告書記載の実証環境のうち、調達仕様書の機能要件である「共通データ基盤（空間情報管理）」  
 「ロボット管理ミドルウェア（データ連携・蓄積・分析）」「ロボット管理ミドルウェア（運行管理・  
 交通管制）」の機能要素に関して、技術仕様を解説する

## アーキテクチャ仮説



## 実証実験環境の全体概要



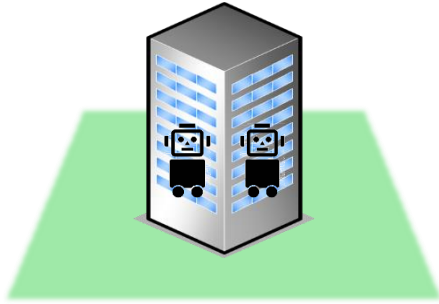
左記の仮説をベースに、既に存在/保有する仕組みを活用しつつ、  
 空間情報基盤・ロボット管理ミドルウェアなど一部暫定機能を用意した環境を構築

凡例 鹿島：鹿島建設  
 KHI：川崎重工業  
 DN：デンソー  
 SB：ソフトバンク

## 1.2 適用範囲

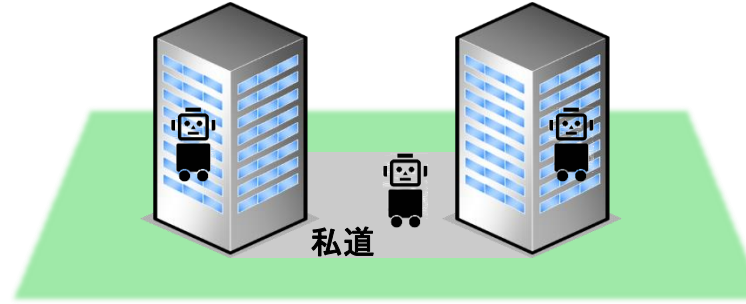
同一管理エリア内の、単一建物あるいは屋外（私道）を跨いだ建物間における複数ベンダー、複数ロボットの分散協調運行をスコープとする

### 建物内運用

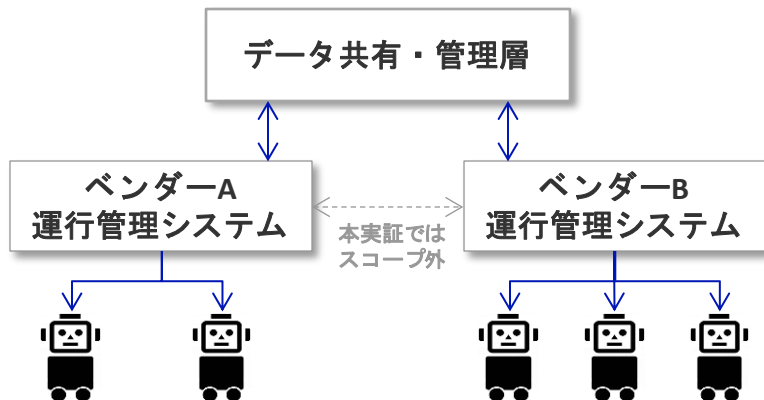


- 移動は建物内のみ
- 複数ベンダーの複数ロボットが建物内で運行

### 建物内⇄私有地運用



- 複数ベンダーの複数ロボットが、複数の建物内で運行
- 屋内から屋外への移動あり
- 屋内、屋外とも管理者は同一（屋外は私道）



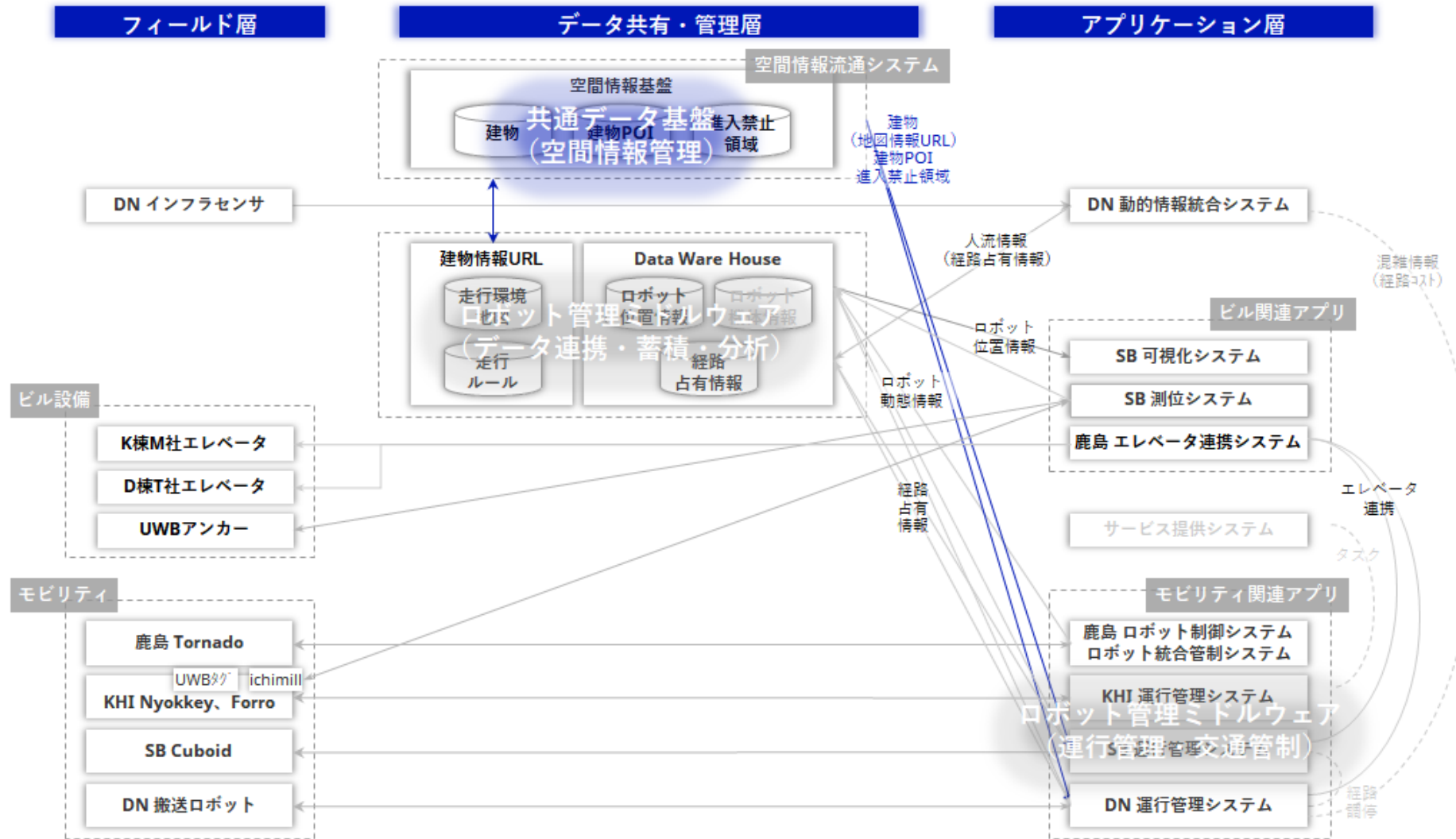
- ✓ 複数ベンダーの運行管理システムがデータ共有・管理層を通して、地図情報、走行ルール、リソース占有情報等を取得して、協調運行するシステムを想定
- ✓ 建物の出入口・エレベータホール・駐機場等における、ロボットの混雑・お見合いの回避をスコープとする
- ✓ 複数ベンダーの運行管理システム同士による調停は、スコープ外とする
- ✓ ロボットの種類としては、搬送ロボットを想定
- ✓ ロボットの同時稼働台数は、最大5台程度を想定

## 2. ロボットサービス運行基盤の仕様概要

- 2.1 共通データ基盤（空間情報管理）
- 2.2 ロボット管理ミドルウェア（データ連携・蓄積・分析）
- 2.3 ロボット管理ミドルウェア（運行管理・交通管制）
- 2.4 構成要素間のインターフェース仕様
- 2.5 データモデル仕様
- 2.6 APIや共通ライブラリのソースコード等

# 2.1 共通データ基盤（空間情報管理）

本実証実験環境における「共通データ基盤（空間情報管理）」の部分を示す



## 2.1 共通データ基盤（空間情報管理）

ロボットサービス運行基盤における共通データ基盤のうち、空間IDを利用した情報共有について記載する

### ■空間情報管理の目的

サービスロボットに対するデータ共有のため、空間IDを用いた共通データ基盤による空間情報管理を行う。

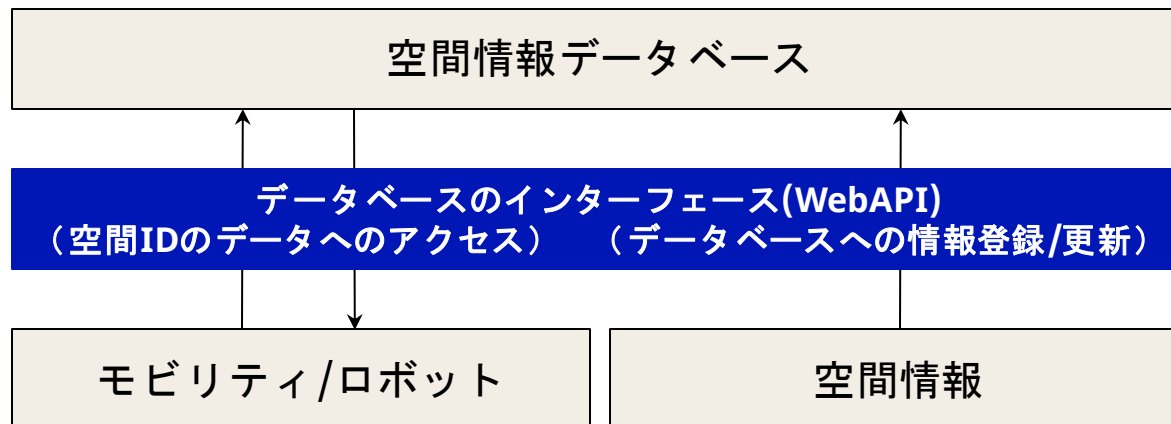
サービスロボット運行の特性として、屋内及び屋外を横断的に走行するという点が挙げられる。

現状、屋内空間に関してはロボットが作成する走行地図ごとに定義されるローカル座標系が用いられることが一般的である一方、屋外空間についてはGNSS測位などを元にした絶対座標系を用いられることが多く、屋内外で一貫した空間指定の概念として空間IDの利用を想定したものである。

### ■共通データ基盤の概略

共通データ基盤は、①空間情報を格納・管理するデータベースと②ロボット等クライアントがデータアクセスするため及び空間情報データを格納、更新するためのインターフェースから構成される。

なお本資料で取り扱う空間IDの技術仕様については、「[4次元時空間情報基盤アーキテクチャガイドライン（β版）](#)」[1]に定められたものを採用する。



[1]: 4次元時空間情報基盤アーキテクチャガイドライン（β版）  
2023年4月28日  
経済産業省/国土交通省/国土地理院/NEDO/IPA  
[4次元時空間情報基盤アーキテクチャガイドライン（β版）\(ipa.go.jp\)](#)  
(2024年3月1日アクセス)



## 2.1 共通データ基盤（空間情報管理）

### ① 空間情報データベース

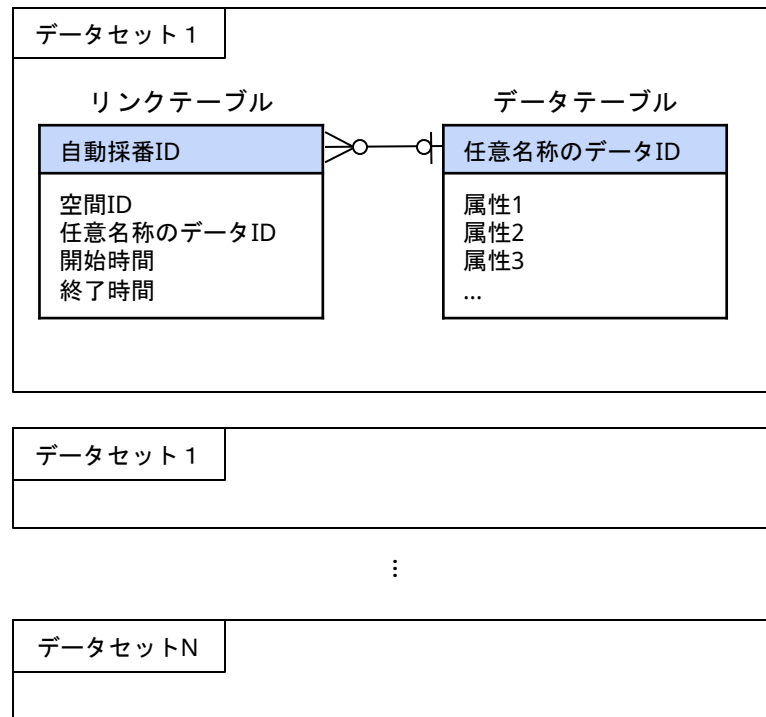
空間情報のデータベースはテーブルの基本構成として、リンクテーブル、データテーブルの2つのテーブルで1つのデータセットを構成し、1つのデータセットに対して1つのメタデータ情報をメタデータテーブルに持つ構造になる。

リンクテーブルは、空間IDにデータテーブルのデータを紐付ける役割のテーブルとなる。

データテーブルは任意名称のデータIDを主キーとして持ち、データの属性、制約を自由に定義できるテーブルとして具体的な情報格納を行うことのできるテーブルとなる。

メタデータテーブル

データセットID
データの名称 概要 主題分類 空間IDへの登録者 キーワード 登録日付 データの利用制限 データの範囲（水平方向） 高さ情報の有無 データの範囲（鉛直方向） 高さの基準 空間ID登録ズームレベル 全データ範囲を包含する空間ID 全データの所在 ソースデータ名 ソースデータの作成者 ソースデータの所在 ソースデータ形式 ソースデータの座標系 ソースデータの精度 ソースデータの収集期間 データの範囲（時間要素）



リンクテーブルのテーブル定義

カラム名	項目名	型	必須	主キー	外部キー	ユニーク	備考
id	自動採番のID	Bigserial	○	○			
voxel_id	空間ID	Varchar	○			○	
(任意名称)	任意名称のデータID	Textか整数値	○		○	○	
start_datetime	開始日時	Timestamp	○			○	・日本標準時 ・省略した場合、「-infinity」値
end_datetime	終了日時	Timestamp	○			○	・日本標準時 ・省略した場合、「-infinity」値

データテーブルのテーブル定義

カラム名	項目名	型	必須	主キー	外部キー	ユニーク	備考
(任意名称)	任意名称のデータID	Textか整数値	○	○			
...	...	...					
...	...	...					

## 2.1 共通データ基盤（空間情報管理）

### ②データベースのインターフェース（WebAPI）

データベースに登録されたデータを配信するためのインターフェース及び、データベースへの情報登録/更新/削除を行うためのインターフェースとしてWebAPIを想定する。

本件では、「デジタルツイン構築に関する調査研究」[1]にて構築したWebAPIをその実例として掲載する。

本WebAPI経由で各テーブルへの問合せ/アクセスが可能、アクセスにはBasic認証での認証を必要とする。

ただし、本WebAPI経由でのメタデータの登録・更新・削除は行えない。

また、本WebAPIで提供されるメソッドは以下の通りである。

No.	APIメソッド名	概要	備考
1	query	指定した条件で検索し、一致する情報を一覧で取得する	
2	adds	新規にデータテーブル情報を登録し、指定した空間IDとデータIDからリンクテーブル情報を登録する	
3	updates	指定した空間ID、建物IDに紐づくデータテーブル情報を更新する	
4	deletes	指定した空間ID、建物IDに紐づくリンクテーブル情報、データテーブル情報を削除する	
5	get_metadata	該当データセットのメタデータ情報取得する	各データセットのURLで提供
6	list_metadata	指定した条件でメタデータを検索し、一致するメタデータ情報を取得する	メタデータ取得用のURLでのみ提供

リクエストへの応答はJsonをgzip圧縮した形式で返送する

[1]令和4年度「デジタルツイン構築に関する調査研究」

- [デジタルツイン構築に関する調査研究 調査報告書](#)
- [デジタルツイン構築に関する調査研究 3次元空間IDデータ標準製品仕様書](#)
- [デジタルツイン構築に関する調査研究 3次元空間IDデータ標準作業手順書](#)

# 2.1 共通データ基盤（空間情報管理）

## 実証実験にて準備したデータセット例

共通データ基盤にて、本実証のために準備をしたデータセット例を示す

### A. 建物検索用

リンクテーブル

ID
空間ID
建物ID
開始時間
終了時間

データテーブル

建物ID
建物名称
建物情報URL
建物種別
建物内ルール種別

ズームレベル：22

#### データセット利用概要

1. 自己位置、あるいは対象位置に対応する空間IDから検索対象の建物IDを取得
2. 取得した建物IDから建物情報を取得

#### 本実証での利用

本実証では2. で取得した建物情報のうち、特に建物情報URLとして取得したデータから地図情報をDLして利用する動作を検証した

### B. 建物POI検索用

リンクテーブル

ID
空間ID
建物POIID
開始時間
終了時間

データテーブル

建物POIID
建物ID
フロアID
建物POI種別
建物POI情報URL
管理者連絡先(電話)
管理者連絡先(メール)

ズームレベル：26

#### データセット利用概要

1. 自己位置、あるいは対象位置に対応する空間IDから検索対象の建物POI IDを取得
2. 取得した建物POI IDから建物POI情報を取得

#### 過去実証での利用（本実証では利用なし）

過去実証において、目的地・経路地の指定や、POIと紐付けた一時的な進入禁止/利用禁止などの動的（準動的）な情報のロボットへの配信用途の検証を行った

### C. 進入禁止領域検索用

リンクテーブル

ID
空間ID
進入禁止領域ID
開始時間
終了時間

データテーブル

進入禁止領域ID
進入禁止種別
モビリティ高さ上限
モビリティ幅上限
モビリティ速度上限
モビリティ速度下限

ズームレベル：26

#### データセット利用概要

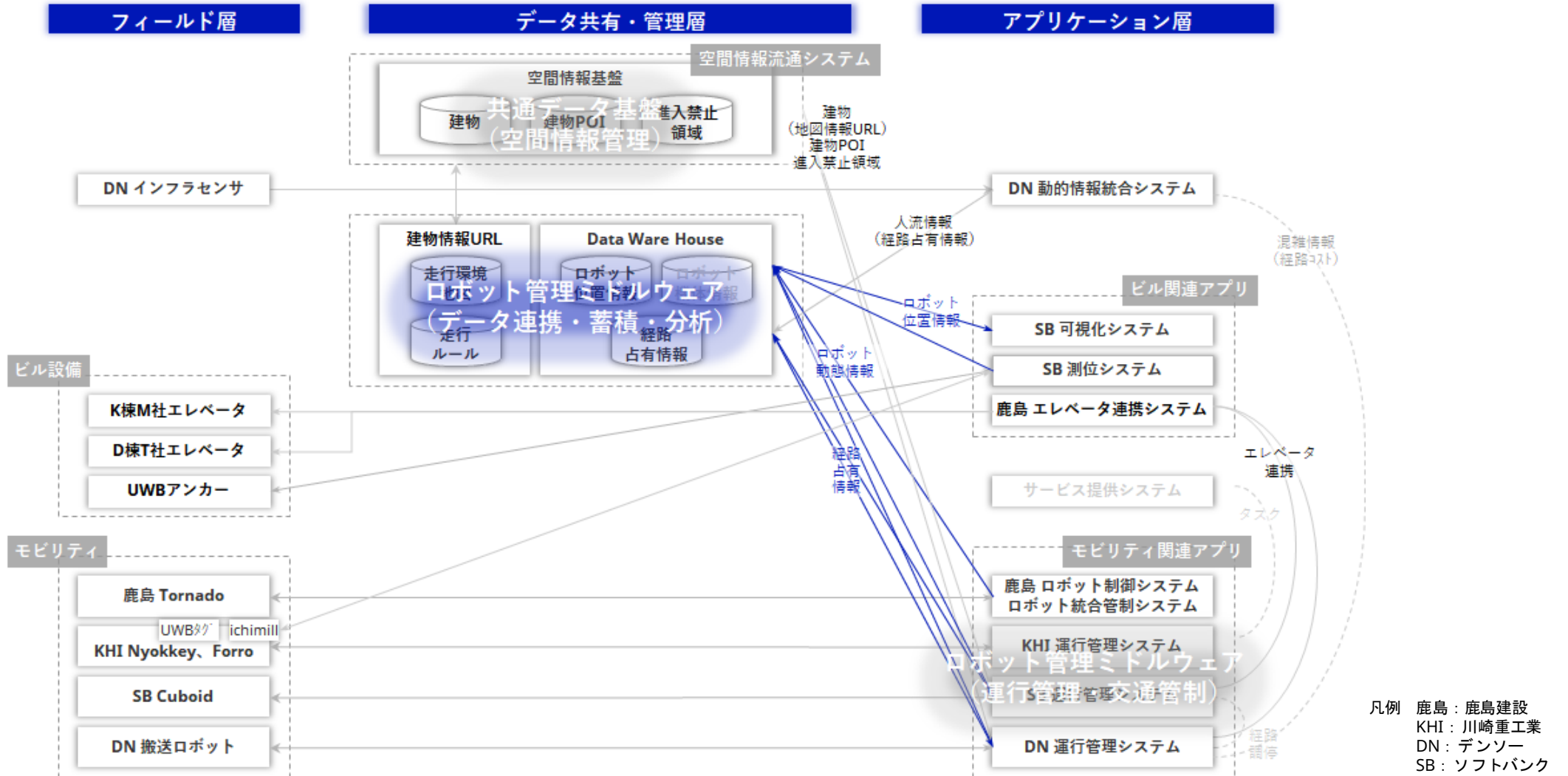
1. 対象位置に対応する空間IDから検索対象の進入禁止領域IDを取得
2. 取得した進入禁止領域IDから進入禁止領域情報を取得

#### 本実証での利用

本実証では本データを準動的な共通データとして取り扱い、リンクテーブルより一括取得した空間IDをロボット用の地図に追加反映することで状況に合わせた規制情報の配信を行った

## 2.2 ロボット管理ミドルウェア（データ連携・蓄積・分析）

本実証実験環境における「ロボット管理ミドルウェア（データ連携・蓄積・分析）」の部分を示す



## 2.2 ロボット管理ミドルウェア（データ連携・蓄積・分析）

本項では、本実証で活用した各社ロボットからのデータを収集し、監視や分析に利用するミドルウェア(以下ロボット管理ミドルウェア)の機能概要について、連携/蓄積/分析という側面から説明する。

### ▼データ連携

本実証では、下記の仕様でのデータ連携を実施した。各仕様の選択については、参加各社との調整の上、開発の難易度等を考慮した結果となっている。

表題	仕様概要	補足情報
通信プロトコル	MQTT v3.1.1	本実証ではQoS0にて実装
通信フォーマット	別紙参照(位置姿勢情報/機器情報)	-
通信頻度	連携ロボット、システムによって調整可能(本実証では10fpsと1fpsで送信)	ロボット等クライアントの用途や、連携必要性からミドルウェアからの頻度指定は非効率と判断
サーバ概要	クラウドにはAzureを利用 MQTTブローカはEvent Gridを採用 データパイプラインはStream Analytics Functionを採用	同クラウドにおけるIoT Hubの場合、送信情報のトピック名称を全て固有にする必要があり、管理ミドルウェアとして不適格と判断
認証	証明書とID/PWを払い出す形式を採用	開発期間等の制約から、最低限のセキュリティ認証を採用

### ▼データ蓄積

上記方式で連携したデータを、下記の形式で蓄積した。

表題	仕様概要	補足情報
サーバ概要	連携I/Fと同じくAzureを利用 データストアはData Lake Storageを採用	本実証で求められるビッグデータの容量/検索性から、CosmosDBと比較し、採用
保存形式	受信した生データを、文字列として"{データ種別}/{デバイスID}/YYYYMMDD/HH/{ファイル名}.json"形式で保存	-
取得形式	指定した期間とデバイスのデータを抽出するスクリプトを用い、テキストで出力	-

## 2.2 ロボット管理ミドルウェア（データ連携・蓄積・分析）

本項では、本実証で活用した各社ロボットからのデータを収集し、監視や分析に利用するミドルウェア(以下ロボット管理ミドルウェア)の機能概要について、連携/蓄積/分析という側面から説明する。

### ▼データ分析

本実証では求められる分析手法が、各社ごとに異なる、また実証という特性上、種々の分析を適応させる必要があることから、ミドルウェア側に分析機能を設けていない。本項目では、“管理/監視”という目的に主眼を置いた際の可視化時の仕様と、分析を実施する場合の想定仕様について述べる。

表題	仕様概要	補足情報
サーバ概要	クラウドにはAzureを利用 可視化アプリの配信はStatic Web Appを採用 データの可視化アプリへの繋ぎこみはWeb PubSubを採用	本実証で求められるパフォーマンスから、各要素を選択
可視化アプリ概要	ベースマップはGoogle Maps APIを採用 ロボット位置情報等の可視化にはDeck.glを採用し、ベースマップにオーバレイする形式で表現	本実証では各社緯度経度でのデータ連携が容易に可能であったこと、エリアが建物を跨ぐことから、一般的なマップAPIを選択 多数のロボットから10fpsという高頻度での位置情報可視化が求められるため、ベースマップとロボット可視化のライブラリを分割
分析仕様	Stream Analyticsから、他システムに影響を与えず、データを分岐することが可能 分岐先で必要な分析機能に連携することで、各種分析に活用が可能と想定	-

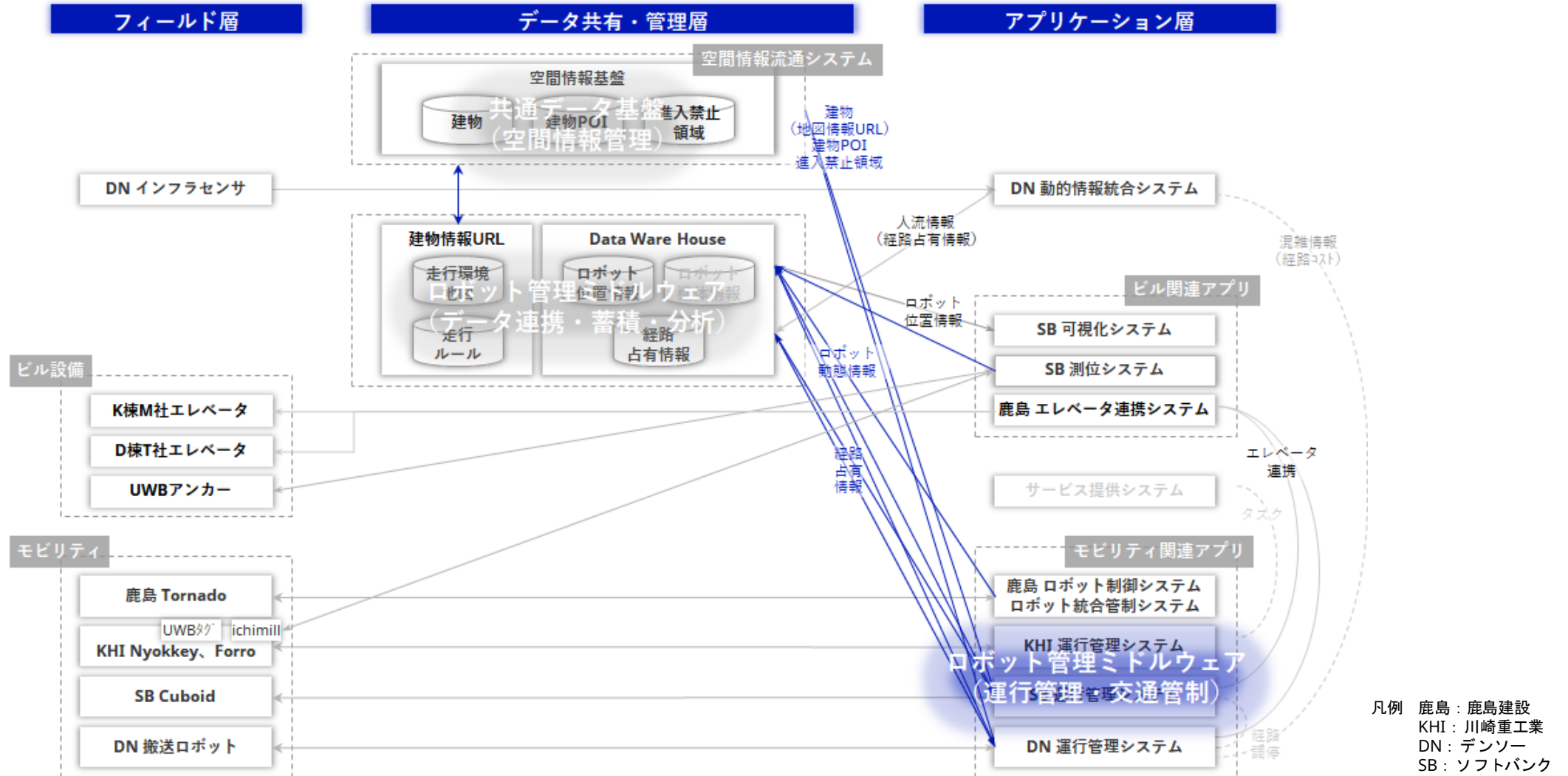
### ▼リソース占有情報管理API

占有情報については、上記仕組みとは独立して存在しているため、その仕様について概要を述べる。

表題	仕様概要	補足情報
サーバ概要	クラウドにはAzureを利用 一般的なREST APIを採用	占有データの保管はRedisを採用
提供API	占有、占有解除といった、個別APIに加え、全解除、全件取得といったマスタAPIを提供	-

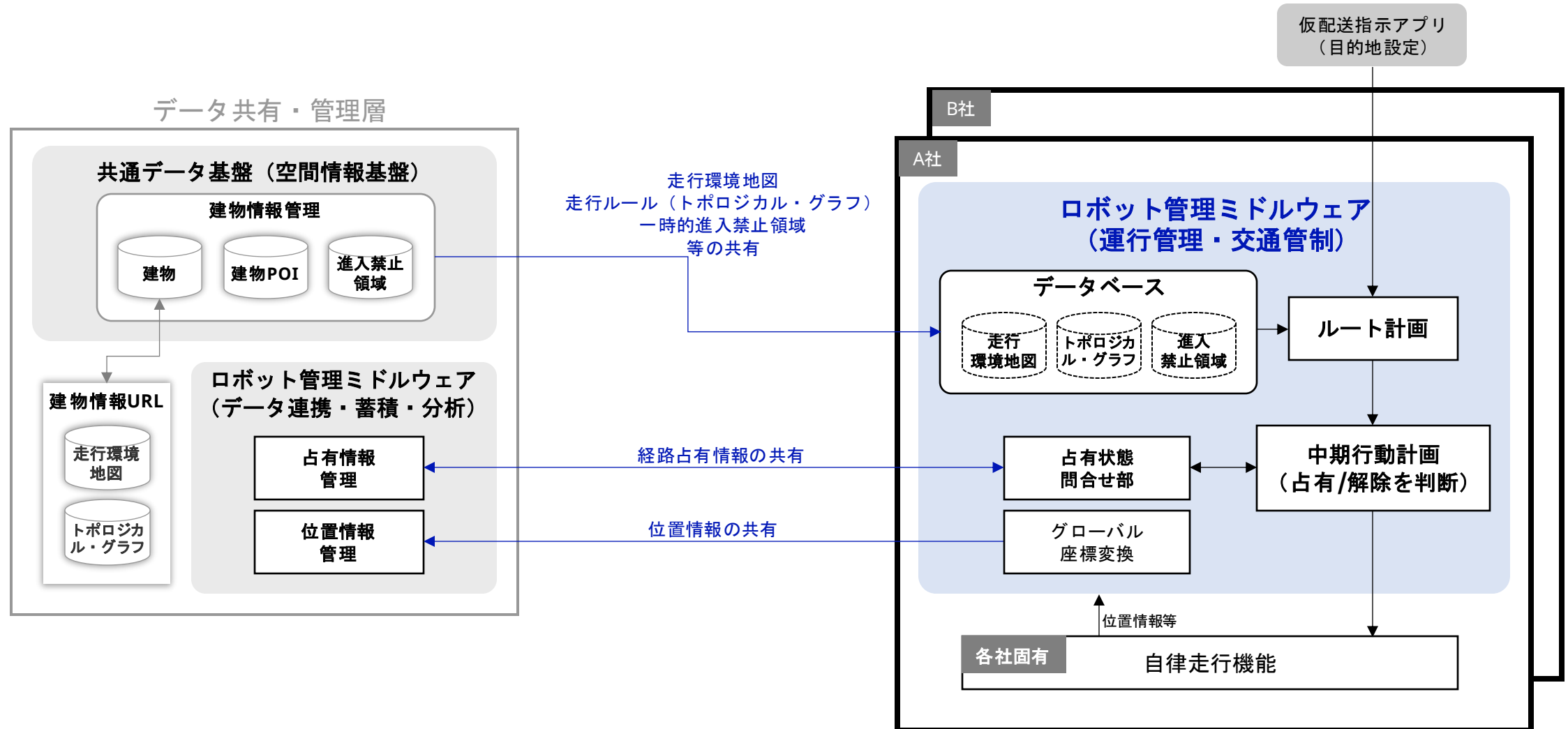
## 2.3 ロボット管理ミドルウェア（運行管理・交通管制）

本実証実験環境における「ロボット管理ミドルウェア（運行管理・交通管制）」の部分を示す



## 2.3 ロボット管理ミドルウェア（運行管理・交通管制）

ロボット管理ミドルウェア（運行管理・交通管制）に関するシステム概要を示す





## 2.3 ロボット管理ミドルウェア（運行管理・交通管制）

ロボット管理ミドルウェア（運行管理・交通制御）の各構成要素の概要を以下に示す

構成要素	概要
データベース	共通データ基盤（空間情報基盤）を介して取得した走行環境地図や走行ルール（トポロジカル・グラフ、一時的進入禁止領域）を格納して「ルート計画」部へ送信する。
ルート計画	データベースに格納されたトポロジカル・グラフと指示された目的地に基づいて大局的な経路生成を行なう。
占有状態問合せ部	リソース占有情報管理APIにより、占有情報管理サーバへ経路占有／解除の申請や占有状態の確認を行なう。
中期行動計画	トポロジカル・グラフ及びロボットの現在位置に基づいて、調停対象場所の占有／解除を判断する。調停対象場所が占有できない場合は、手前で待機する。

## 2.4 構成要素間のインターフェース仕様

本実証で利用した「ロボット管理ミドルウェア（運行管理・交通管制）」と「ロボット管理ミドルウェア（データ連携・蓄積・分析）」間のインターフェースの概要について述べる

No.	IF名称	IF概要	入力 (FROM)	出力 (to)	接続方式		送信タイミング		項目例
					受渡方式	データ形式	タイミング	頻度	
001	Location Adapter	各ロボットから、位置及び姿勢情報を送信する。 本実証においては、位置情報は緯度経度とし、姿勢情報は四元数で表現された方位(真東0°)と定義した。	ロボット本体 各社運行システム 位置測位API	DWH	MQTT	JSON	リアルタイム/定期送信	1-10fps (機器依存)	位置情報(緯度経度高度) 位置情報(空間ID) 姿勢情報(四元数) フロア名称
002	Device Information Adapter	各ロボットから、機体情報を送信する。 本実証では、右記の項目のルール化や、各社で取得可能な情報の整理/意味の整合性などのすり合わせに時間を要し、最低限必要な枠組みの定義に留まった。	ロボット本体 各社運行システム 位置測位API	DWH	MQTT	JSON	リアルタイム/定期送信	1-10fps (機器依存)	バッテリー残量(使用はロボット各社による) 機体ステータス(未使用) タスクステータス(未使用)
003	Request Occupation	占有したい空間IDを指定することで、既に他デバイスに占有されている/自身で占有している/占有されていないため、占有完了のステータスを返却する	ロボット本体 各社運行システム	DWH	API	HTTP (POST)	別紙参照	-	-
004	Release Occupation	既に占有している空間IDに対して、占有を解除する。 自身で占有している空間IDに対してのみ占有が解除可能	ロボット本体 各社運行システム	DWH	API	HTTP (POST)	別紙参照	-	-
005	Clear All Occupation	自身で占有している全ての空間IDを解除する サーバ側で長期占有を監視し、強制的に占有を解除する仕組みを要しておらず、占有を解除しなければ他デバイスが占有不可となってしまうため、オペレーション完了時には全解除を推奨する	ロボット本体 各社運行システム	DWH	API	HTTP (POST)	別紙参照	-	-
006	Get All Occupation	デバイスに抛らず、全ての空間IDの占有状況を取得する	人流情報管理システム	DWH	API	HTTP (GET)	-	-	-

## 2.4 構成要素間のインターフェース仕様

位置情報共有のための「Location Adapter」インターフェースの事例を示す

Location Adapter		Topic Name : hic/dwh/location-adapter		
Properties	論理名	型	データ例	備考
device	デバイス	object	-	複数デバイス対応を考慮した構造体
device_id	デバイスID	string	"sb_cuboid_01"	デバイス固有の文字列
location_timestamp	位置情報タイムスタンプ	number	1694515065522	位置情報を取得したUNIX時間(ミリ秒)
position	位置	object	-	デバイスに紐づく位置情報
position_type	座標情報タイプ	number	1	座標情報の出力タイプ 1: 緯度経度高度 2: 空間ID
latitude	緯度(°/十進数表記)	number	35.68948234	(position_type = 1 の場合に有効)
longitude	経度(°/十進数表記)	number	139.6929585	(position_type = 1 の場合に有効)
altitude	高度(m)	number	45.32	(position_type = 1 の場合に有効)
voxel_id	空間ID	string	"26/36/59607689/26431901"	(position_type = 2 の場合に有効)
orientation	姿勢	object	-	デバイスに紐づく姿勢情報 右手系、x軸正が真東/y軸正が真北/z軸正が真上、 回転方向は右ネジとする
orientation_x	四元数 x 値	number	0.5086478	取得不可の場合、0.0ではなく Noneを入力
orientation_y	四元数 y 値	number	0.4351648	取得不可の場合、0.0ではなく Noneを入力
orientation_z	四元数 z 値	number	0.7426805	取得不可の場合、0.0ではなく Noneを入力
orientation_w	四元数 w 値	number	-0.018295	取得不可の場合、0.0ではなく Noneを入力
floor_name	フロア名称	string	"M2"	任意の文字列

## 2.4 構成要素間のインターフェース仕様

経路占有情報共有のための「Request Occupation」「Release Occupation」「Clear All Occupation」インターフェースの事例を示す

Request Occupation		ロボットからノード(空間ID)の占有を申請		
Properties	論理名	型	データ例	備考
ts	UNIX TIME	int	1694515065522	UNIX時間(ミリ秒)
device_id	デバイスID	string	"sb_cuboid_01"	デバイス固有の数値
spatial_id_list	占有申請する空間IDリスト	array	["26/20/59606770/26456523", "26/20/59606779/26456520"]	

Release Occupation		ロボットからノード(空間ID)の占有解除を申請		
Properties	論理名	型	データ例	備考
ts	UNIX TIME	int	1694515065522	UNIX時間(ミリ秒)
device_id	デバイスID	string	"sb_cuboid_01"	デバイス固有の数値
spatial_id_list	占有解除申請する空間IDリスト	array	["26/20/59606770/26456523", "26/20/59606779/26456520"]	

Clear All Occupation		該当するデバイスIDを持つロボットの全てのノード占有を解除		
Properties	論理名	型	データ例	備考
ts	UNIX TIME	int	1694515065522	UNIX時間(ミリ秒)
device_id	デバイスID	string	"sb_cuboid_01"	デバイス固有の数値

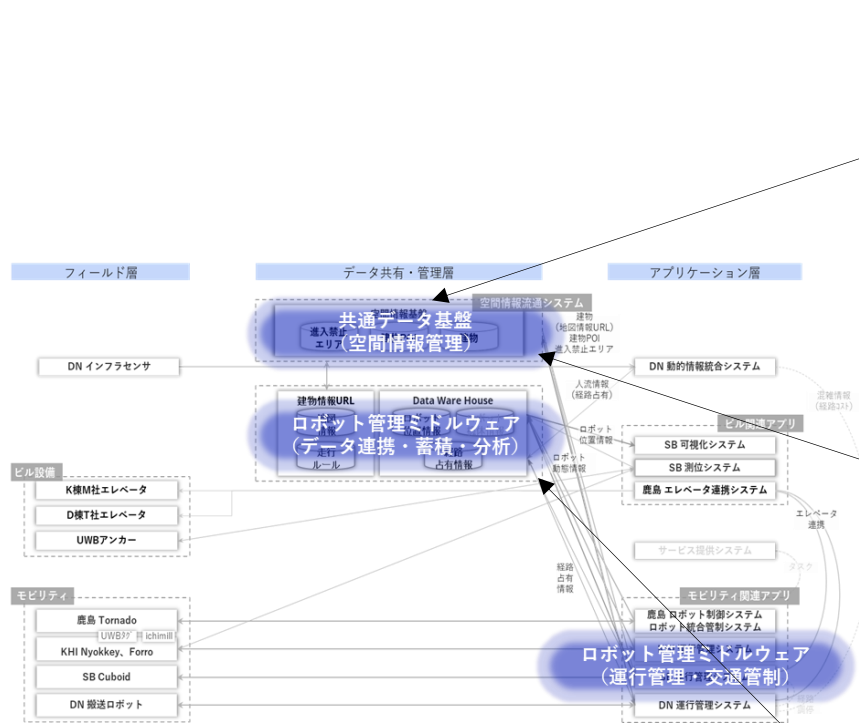
## 2.4 構成要素間のインターフェース仕様

インターフェース設計において発生した課題及び対応案（今後の展望）について、システム観点を中心に考察を述べる

No.	課題	概要	対応案（今後の展望）
001	時刻同期について	各社ロボットは言い換えると個別のサーバであり、それぞれが異なる時刻を持っている。 経路計画の精緻化や、タスクの連携など、時刻の厳密性が求められるミッションを分散協調するためには時刻同期が必要になるシーンが発生すると考えられる。	① 5G/Wi-fi等に拠る時刻同期専用の仕組みの導入 ② GPS等に拠るパブリックな時刻同期の仕組みの導入 参考: Wi-Fi
002	各種情報のフォーマット化について	本検証で活用した各種データにおいて、それぞれ定義が無ければ解釈に相違が出る可能性を内包している。下記に例示する。 ① 位置情報 緯度経度の座標系(WGS84/19系 etc.)、緯度経度の表現方法(十進数 or 時分秒 etc.) 高度の算出方法(海拔/ジオイド高さ etc.) ② 姿勢/方位情報 どこを基準とした方位(真北0度/磁北0度/真東0度 etc...)、四元数の系(右手系/左手系、回転方向) ③ 各種ステータス等 走行中/停止中の定義、タスクに対する管理手法の定義	① 情報フォーマットの策定 - 機器内部で活用するローカルフォーマットと、連携のために活用されるグローバルフォーマットの定義 - 導入障壁を上げる可能性が懸念されるため、必須項目/推奨項目など、情報への優先順位の付与
003	より高度かつ安定した占有について	本実証で活用した占有APIでは、先勝ち方式を採用しているため、ロボットの位置に拠らず占有が発生し、スタックする事象が確認された。 システムだけで吸収することは困難であると想定されるが、右記にいくつかの解決方法を示唆する。	① ロボット及び遂行タスクの優先順位の定義 ② チケット等による占有予約の仕組みの導入 ③ ロボットの位置等による占有妥当性のサーバ側での判断
004	より汎用的なプロトコルについて	本実証では、ROS内部のトピックがMQTTであること、データ種別に応じて送信頻度や重要度合いが異なることなどを勘案し、MQTTの採用に至った。 しかし、データを送信するだけでなく、双方向通信が必要になるシーンが将来的には想定される。 この場合、安全かつより高いパフォーマンスが求められると考えられ、将来的にはプロトコルの見直しが求められる可能性がある。	① 緊急時(衝突回避、緊急停止 etc)を想定した双方向通信の必要性の議論及び、求められるパフォーマンスの策定と検証
005	より多様なデータ連携について	本実証では、実証すべき要素が決まっているため、優先順位に伴って実装を行った。 しかし、将来的にはさらに多くのデータを連携することで、よりよい分散協調を目指す必要があると考えられる。	下記項目の連携に向けた検討 ① 映像情報(カメラ搭載機器) 連携プロトコルの選定 セキュリティに関する考察 ② センサ情報(各種センサ搭載機器) センサの多様性(温度/CO2濃度 etc.)に耐えうる汎用的なフォーマットの定義

# 2.5 データモデル仕様

本実証で利用したデータモデルの標準仕様の概要を示す



## データモデル

静的データ		
<b>建物情報</b>	<b>建物POI情報</b>	
空間ID 建物ID 建物名称 建物情報URL 建物タイプ	空間ID 建物POI ID 建物ID 建物POI名称 管理者連絡先	
準動的データ		
<b>進入禁止領域情報</b>		
空間ID 進入禁止種別ID モビリティ高さ上限 モビリティ幅上限 モビリティ速度上限		
動的データ		
<b>位置情報</b>	<b>機体情報</b>	<b>占有情報</b>
デバイスID 測位時刻 位置情報 姿勢情報 滞在フロア情報	デバイスID 計測時刻 バッテリー残量 機体ステータス 作業ステータス	空間ID 占有時刻 デバイスID

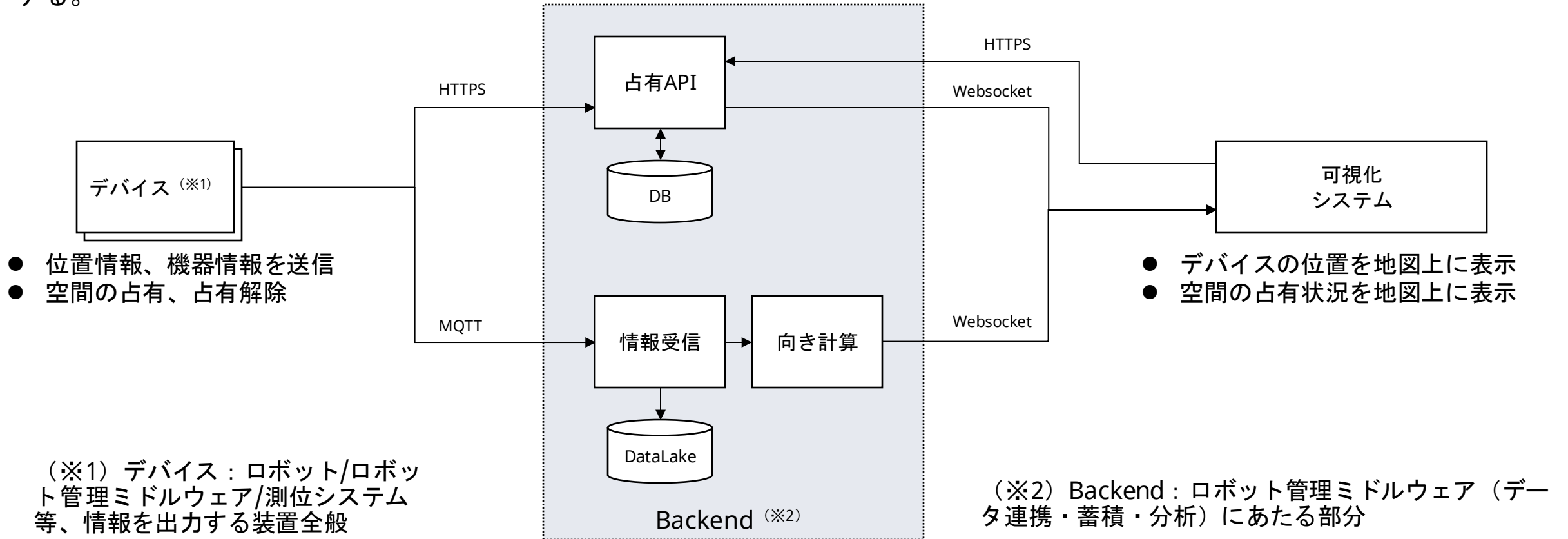
## データ項目リスト

エンティティ	概要	項目例
静的データ	原則更新が不要な、建物等のデータ	<ul style="list-style-type: none"> <li>建物/設備情報</li> <li>機体情報</li> </ul>
準動的データ	一定の期間で変化する、イベント等のデータ	<ul style="list-style-type: none"> <li>工事/イベント等による規制情報</li> <li>点群データ</li> <li>バス等運行情報 etc.</li> </ul>
動的データ	ほぼリアルタイムで更新されるデータ	<ul style="list-style-type: none"> <li>ロボット位置情報</li> <li>インフラセンサ情報</li> <li>天候/災害情報</li> <li>交通情報 etc.</li> </ul>

## 2.6 API や共通ライブラリのソースコード等

### ①システム概念図

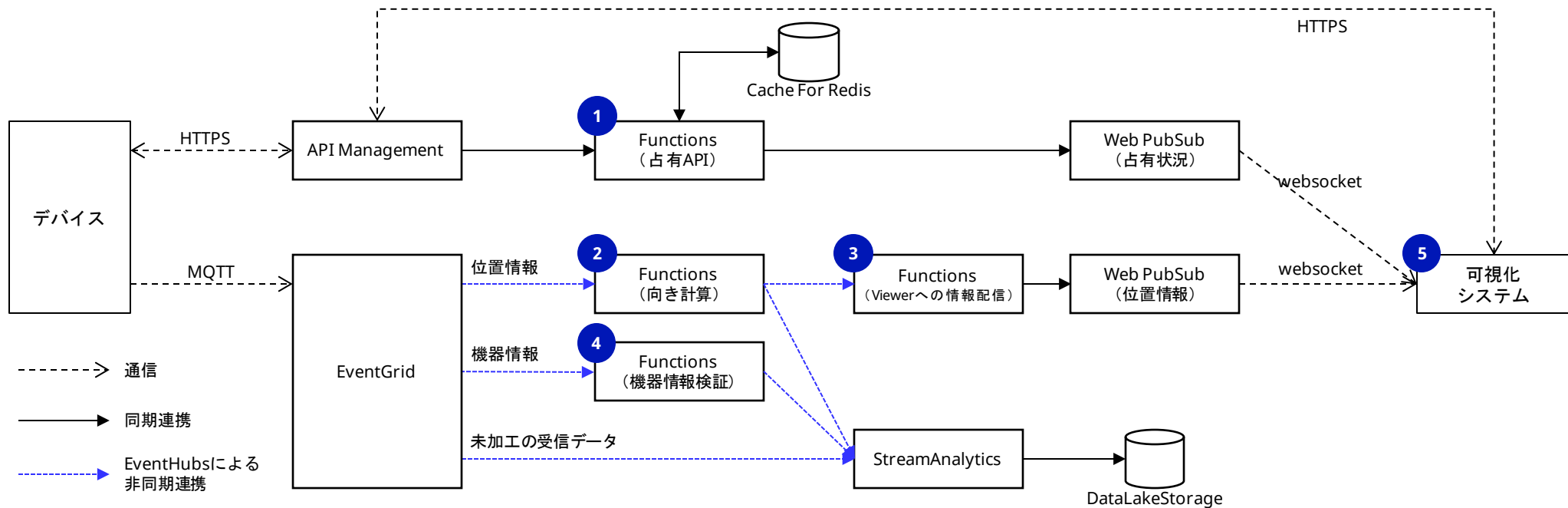
本システムはデバイス（※1）、Backend（※2）、可視化システムの3つで構成される。各デバイスが送信した情報をBackendが受け取り、向き計算等、必要な処理を施した上で可視化システムにリアルタイムに配信、地図上に各デバイスの位置を表示する。受信した情報はDataLake内に保存する。またBackendは空間の占有状態の管理を行う。占有状況は可視化システムにリアルタイム配信し、地図上に占有状況を表示する。



## 2.6 API や共通ライブラリのソースコード等

### ②クラウドのリソース構成

BackendはAzure上に構築している。MQTT受信などにはマネージドサービスを利用。  
リソース間のデータ連携にはEventHubsを使用し、低Latencyな非同期連携を実現している。



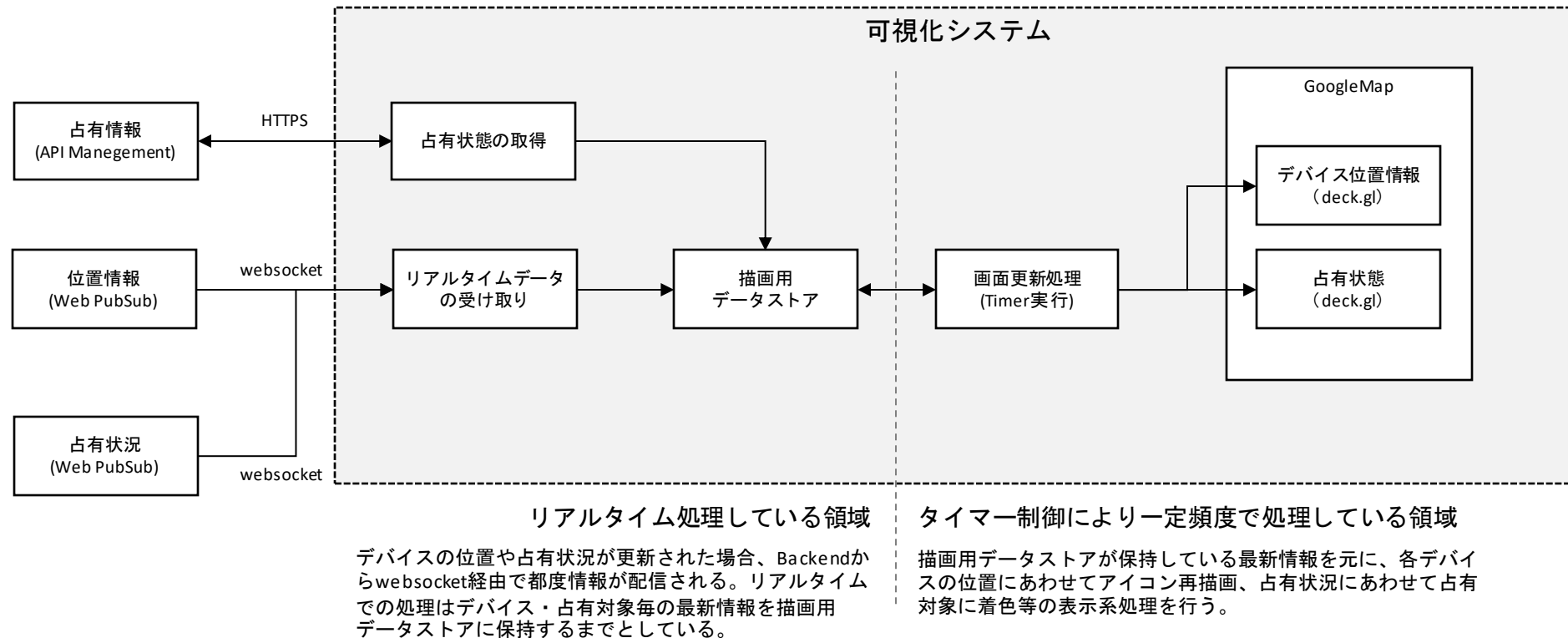


## 2.6 API や共通ライブラリのソースコード等

### ③可視化システムの内部構成

地図表示にはGoogle Maps APIを利用している。レイヤー描画にはdeck.glを採用。

位置情報、占有状況はリアルタイムに配信されるが、表示内容の更新はタイマー制御により一定間隔で行う。これによりデバイス数が増加した場合でも処理量を一定に保つ。



## 2.6 API や共通ライブラリのソースコード等

### ④参照ソースコードファイル名一覧

#	対象機能	ソースコード概要	ソースコードファイル名
1	占有API	各ロボットから送信された各種占有リクエストを処理する	ソースコード一式(Backend)/occupationApi/src/RequestOccupation.cs
			ソースコード一式(Backend)/occupationApi/src/ReleaseOccupation.cs
			ソースコード一式(Backend)/occupationApi/src/ClearAllOccupation.cs
			ソースコード一式(Backend)/occupationApi/src/GetOccupationStatus.cs
2	向き計算	受信した位置情報の検証、及び、向き計算を行う	ソースコード一式(Backend)/func/locationTrigger.cs
3	可視化システムへの情報配信	向き計算後の位置情報を可視化システムに配信する	ソースコード一式(Backend)/func/PublishToApp.cs
4	機器情報検証	受信した機器情報の検証を行う	ソースコード一式(Backend)/func/DeviceInfoTrigger.cs
5	可視化システム	受信した位置/向き情報及び占有情報を可視化する	ソースコード一式(可視化システム)/src 配下参照

# 3. ロボットサービス運行基盤の運用概要

- 3.1 ロボットサービス運行基盤を用いた分散協調運行の概要
- 3.2 走行環境地図・走行ルールの共有
- 3.3 一時的進入禁止領域の共有
- 3.4 位置情報の共有
- 3.5 経路占有情報の共有
- 3.6 屋内外横断走行時の地図乗り換え
- 3.7 分散協調運行に関する課題・今後の展望

# 3.1 ロボットサービス運行基盤を用いた分散協調運行の概要

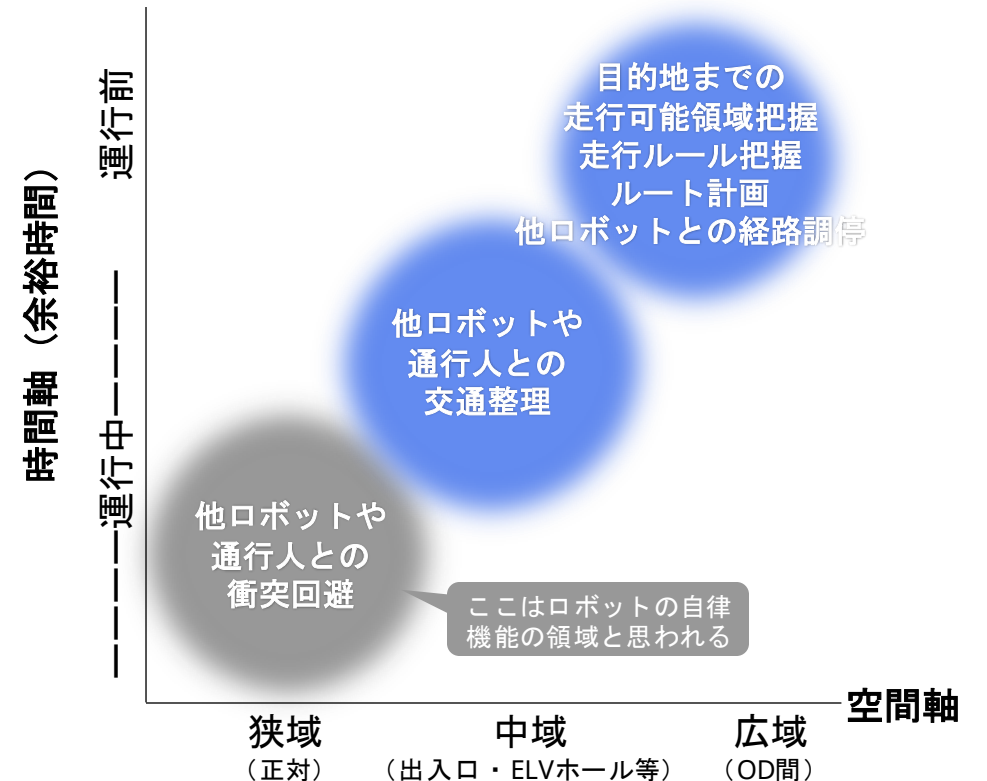
「共通データ基盤（空間情報管理）」「ロボット管理ミドルウェア（データ連携・蓄積・分析）」を通じて共有されるデータを用いた、複数ベンダー・複数ロボットによる分散協調運行について述べる

## 分散協調運行における共通データ

		走行する空間
運行前	静的	走行環境地図（走行可能領域） （階段・エスカレータ等の恒久的進入禁止領域を含む） 走行ルール（トポロジカル・グラフ） （双方向/一方向、速度、勾配、段差、重要/高さ制限等） 建物内外結節点 （トポロジカル・グラフのノードに乗換用属性定義）
	準動的	一時的進入禁止領域（空間ID群） 他ロボット経路情報 人流情報（混雑領域）
運行中	動的	他ロボット経路占有情報 各ロボット位置情報/ステータス 交通参加者の位置情報

これらの共通データを共有することでの、  
 運行前から運行中にかけての分散協調運行のシナリオ例を次頁に示す。

## 運行全般の時空間において 協調が必要と思われる領域



# 3.1 ロボットサービス運行基盤を用いた分散協調運行の概要

## 共通データを用いた分散協調運行のシナリオ例

### 長期（運行前）：

- ロボット運行事業者が目的地が含まれる施設を空間IDを通じて把握し、当該施設の「走行環境地図（走行可能領域）」「走行ルール（トポロジカル・マップ）」を取得し、目的地までのルートを計画する
- イベント等による一時的な進入禁止領域があれば、それらの空間ID群を取得し、その領域を避けるルートを計画する
- 計画したルートをロボットに指示する

### 中期（運行中）：

- 経路前方に十分なスペースがない場合は、以下のような時間的な調停を図る
    - エレベータホールにおいてエレベータ待ちのロボットで先がつかまっていたら手前で待機する
    - すれ違いが困難な出入口においては片方が手前で待機する
- 本実証実験においては、当該箇所にトポロジカル・グラフのノードを配置し、先にそのノードを占有したほうが進行し、占有できなかったほうは手前で待機する「先勝ち方式」を検証した。（ロボット運行事業者間での交渉ではなく、押しボタン式信号のボタンを押して他ロボットが通行するのを待つイメージ）
- 建物を跨いだ運行においては、共有されたトポロジカル・グラフの、建物出入口属性が付与されたノードの場所において、走行に用いる地図を乗り換える

### 短期（運行中）：

- 他の交通参加者や他ロボットと正対した場合は、ロボットの自律機能により衝突回避する
- 相手が他ロボットの場合、お互いの軌道計画（位置・方向）を共有することで、事前に減速あるいはすれ違い可能な軌道への変更が可能と思われるが、本実証では位置情報を「ロボット管理ミドルウェア（データ連携・蓄積・分析）」にあげるまでであり、それを共有することによる協調までは未検証

## 3.2 走行環境地図・走行ルールの共有

サービスロボットが使用する地図情報としては、一般に以下の3種類がある[1]。

[1]出典：「運行管理機能インターフェース仕様書Version 1.0」2023年6月  
RRIロボットイノベーションWG ソフトウェアアーキテクチャ調査検討委員会

- 自己位置推定用地図：スキャンマッチング等の自己位置推定において基準となるラスタ地図
- 走行環境地図：走行可能領域を把握するためのラスタ地図（ROSではpgm形式で作成される）
- 運行管理地図：走行ルールや経路コスト情報を含む経路計画用のベクター地図（ネットワーク地図と呼ばれる）

走行可能領域把握や自己位置推定のためのラスタ地図に関しては、サービスロボットにおける周辺環境認識や自己位置推定のためのセンサの種類や設定位置、および処理アルゴリズムが様々であることから、それらの特徴に合わせて生成される走行環境地図や自己位置推定用地図の統一的な管理・共有は困難である。

しかしサービスロボットが走行する建物内外の地物情報に対して、複数のサービスロボット運行事業者が個々に計測・地図生成する方法では、走行する場ごとに計測が必要となるため、場ごとのロボットのプラグ&プレイのような運用はコストがかかるとされる。そこで各サービスロボットの特徴に合わせる前のベースとなる地図を共通とする取組みが過去に検討されている[2]。

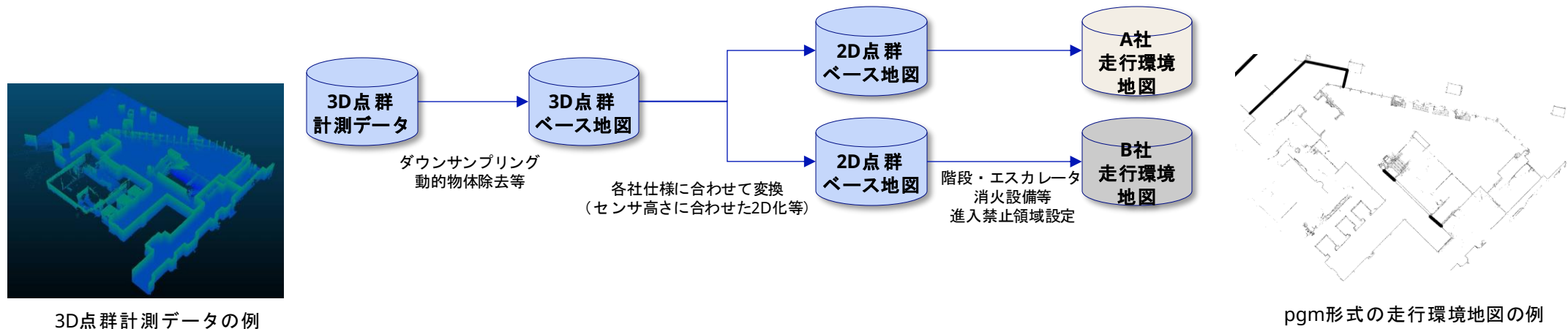
[2]【NTT西日本】異種ロボット間の地図シェアリングの実現に向けた共同研究を開始 -労働力不足を解消する多様なロボットと共生する社会の実現へ- (2024年3月1日アクセス)

「歩行空間の3次元地図ワーキンググループ」

第1回（令和5年6月21日）資料5「3次元点群データ活用要件、取得・統合方法」[001618437.pdf \(mlit.go.jp\)](#) (2024年3月1日アクセス)

第2回（令和5年10月5日）資料2「3次元点群データの区分案について」[001635166.pdf \(mlit.go.jp\)](#) (2024年3月1日アクセス)

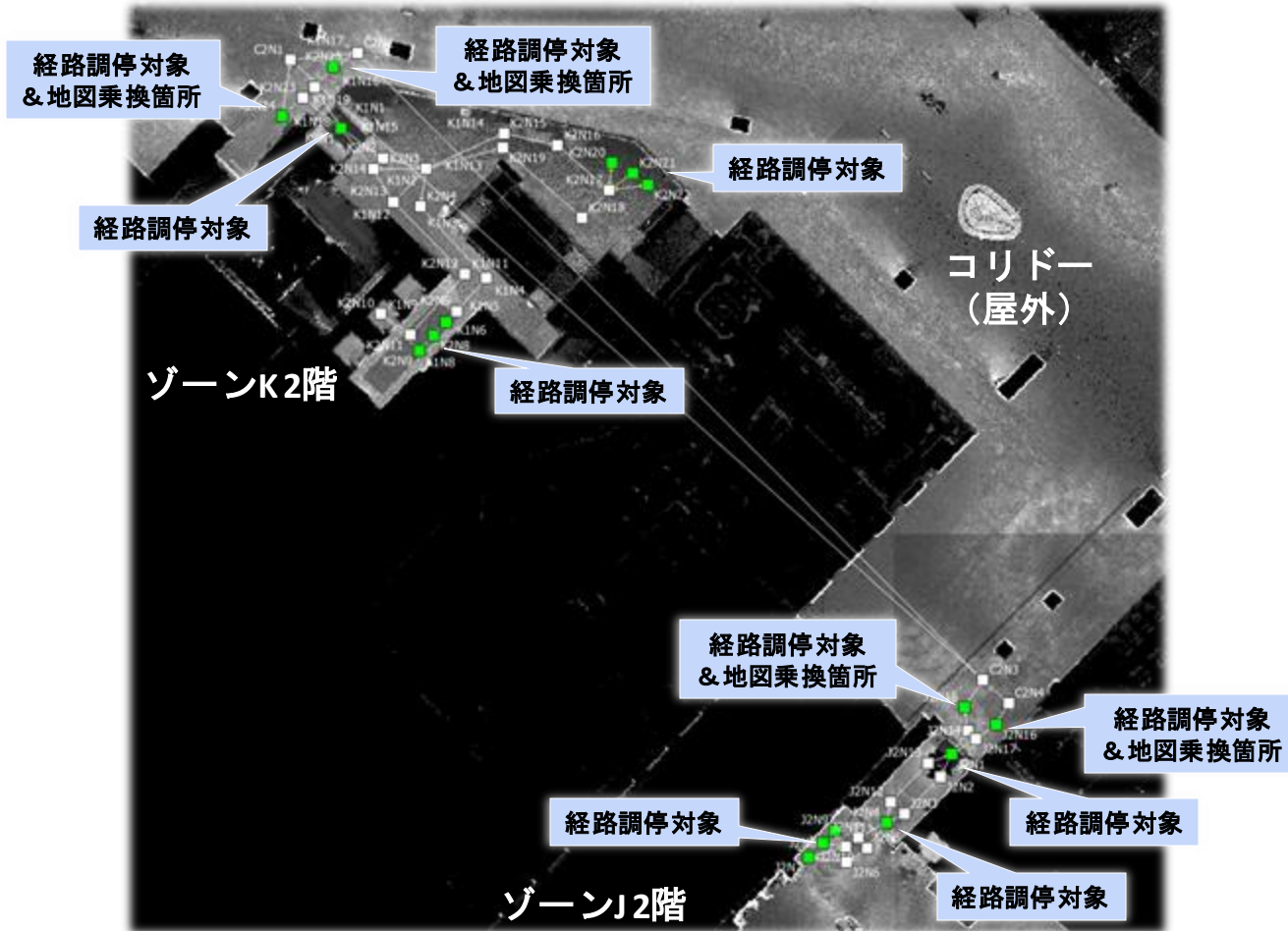
そこで本実証実験のユースケースC、D（複数ベンダー・複数ロボットによる搬送）においては、計測装置（GNSS付LiDARハンディスキャナ）を用いてベースとなる3D点群データを計測し、それを各社の仕様に合わせて走行環境地図に変換することとしたが、本実証実験の場所では床面が並行かつ壁面が垂直等の条件により、走行環境地図を2社で共有した。



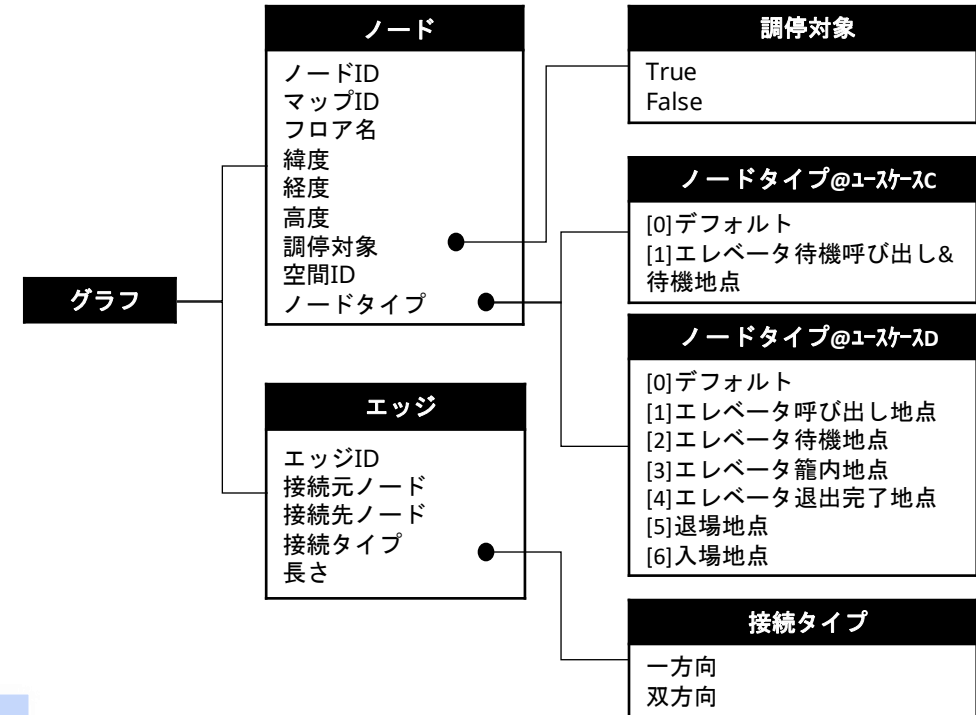
## 3.2 走行環境地図・走行ルールの共有

本実証では走行ルールを示す運行管理地図としてトポロジカル・グラフを使用したためその例を示す

ユースケースCのトポロジカル・グラフ例



データモデル



トポロジカル・グラフでは、走行ルール（本実証では1方向/双方向ルールのみ）に加え、経路占有の調停に用いる属性や、建物間移動での地図乗り換えに用いる属性をノードに付与した。

## 3.2 走行環境地図・走行ルールの共有

### ■地図情報（走行可能領域把握や自己位置推定のための地図）の共有に関して

- 走行可能領域把握や自己位置推定のための地図（2D/3D点群地図）の共有は、ロボットの特徴（センサの種類・搭載位置や自律移動ロジック等）や、走行環境によって要件が異なるため、難しいと思われる。
- ただし走行環境をバリアフリーの屋内に限定する場合は段差を考慮する必要がないため、壁面の形状・特徴点を抽出できるのであれば走行可能領域を示す走行環境地図の共有可能性はある。
- 走行可能領域については、階段やエスカレータ等の物理的にロボットが走行できない領域の他に、消火・防火施設などもあるため施設管理者の承認を受ける必要がある。
- 一時的な進入禁止領域については、走行環境地図を更新するのではなく、進入禁止領域を表す空間ボクセルを設定することを想定して検証した。
- ベースとなる3D点群データを共有することにより、各社ロボットのスキャンマッチングによる自己位置推定結果の相対的な誤差を抑えるメリットがある。また経済的には各社毎で地図作成するコストの削減も考えられるが、実際の点群地図作成はロボット運行の知見をもった者が適切と思われる。
- ベースとなる3D点群データの共有に際しては、自己位置推定や行動計画等の走行安全性に影響する情報であるため、品質を担保する仕組みが必要である。
- ベースとなる3D点群データをロボットベンダー毎に作成する場合であっても、複数のロボットが位置情報を共有して、荷物の受け渡し等の連携サービスをする場合においては、双方の位置精度を確保することが必要であり、そのためには双方のロボットが自己位置推定に使用する地図の位置精度を合わせておく必要がある。そのためには、建物内に共通の基準点を複数設けて、その基準点と地図を合わせこむこと等の事前作業が必要と思われる。（特に歪みの補正が必要と想定される）

### ■走行ルール（ネットワーク地図）の共有に関して

- ノード・エッジで構成されるトポロジカル・グラフ形式の地図に走行ルールを設定し、このトポロジカル・グラフ地図自体を共有することで、走行ルールを共有可能とする方法を検証した。
- エッジに対しては、双方向/一方向、左側通行/右側通行（双方向時）、最高速度、縦断勾配、段差、重量/高さ制限、進入可能台数等のルールが設定可能である。
- ノードに対しては、進入可否制御対象/非対象等のルールが設定可能であり、本実証実験においては「出入口」「エレベータホール」「待機場所」等のボトルネックと思われる箇所のノードを進入可否制御対象とした。
- これらのルールは建物の構造や人の流れによって変わるため、施設（建物）毎にルールがあると想定される。よって施設（建物）毎にルール（ネットワーク地図）を取得する必要がある。またロボットの種類（清掃・警備・配送等）により適用が異なる場合は、ロボットの種別毎に取得することも想定される。
- これらのルールの設定者は施設管理者を想定するが、実際のトポロジカル・マップ作成はロボット運行の知見をもった者が適切と思われる。

### ■共通の課題

- 走行環境地図・走行ルールの共有（受信）は、目的地が含まれる建物情報を検索し、建物情報に含まれる地図情報格納先URLよりダウンロードすることを想定し、検証した。
- 走行環境地図・走行ルールの更新頻度については、施設（建物）毎で異なると想定されるため、今後検討が必要であると考えられる。

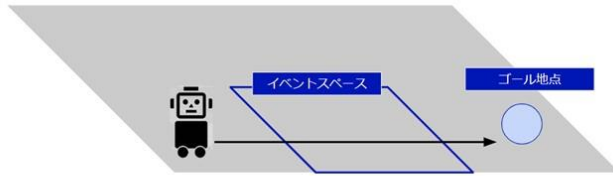


### 3.3 一時的進入禁止領域の共有

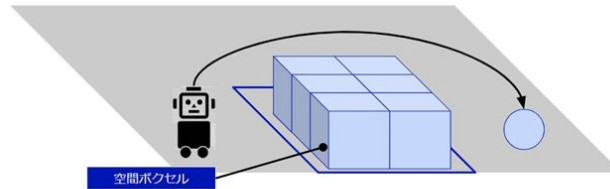
一時的進入禁止領域情報の共有のため、空間情報基盤の構成要素として「進入禁止情報検索用データベース」を用意し、領域情報を空間IDで表し格納する

施設内で一定期間障害物が配置されていたり、イベント等により空間が占有されているようなケースにおいて、施設管理者がそれらの領域へのロボットの進入を防ぐために一時的（準動的）な進入禁止領域を設定したい状況が考えられる。本実証実験においては、一時的な進入禁止領域の共有方法として、「共通データ基盤（空間情報管理）」を介した空間IDによる一時的進入禁止領域情報の共有を実施した。

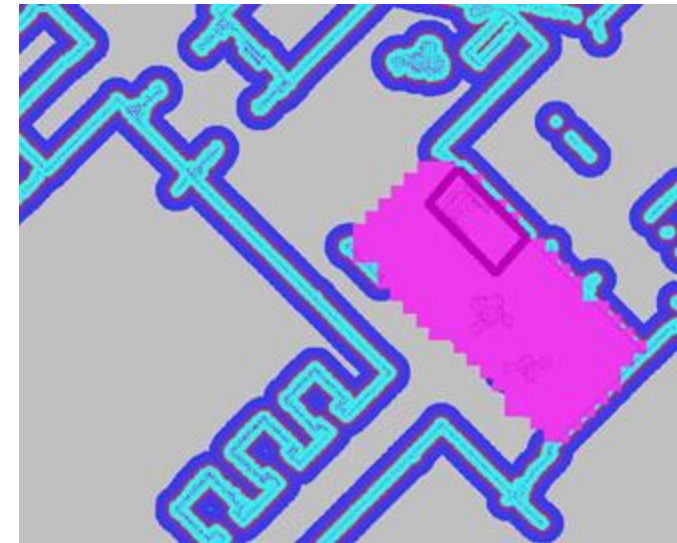
一時的進入禁止領域情報の共有を実現するため、データベースとして「進入禁止情報検索用データベース」を事前に用意した。一時的に進入禁止指定としたい施設内の領域をズームレベル26のボクセル群で表し、それらの空間IDをデータベースに格納した。データベースは禁止領域の期間や禁止対象モビリティの種類などの指定なども可能な設計となっているが、今回の実証においては指定されたボクセル全てを一律で進入禁止対象領域とした。また、実際の領域としてはボクセル底辺を構成する4つの頂点で囲われたエリアを進入禁止領域とした。このデータベースより空間ID情報を取得し、走行マップの進入禁止領域並びにルート設計に反映した。



モビリティは通常事前に与えられた壁や障害物の情報を元にゴール地点までの最短経路を設計する



事前に領域を覆う（囲む）ような空間ボクセルのIDをデータベースに格納  
モビリティはこれを取得、展開し、迂回する経路でゴール地点まで走行



ユースケースDにおいて空間ID（レベル26のボクセル）で指定した一時的進入禁止領域（ピンク塗りつぶし領域）

## 3.4 位置情報の共有

位置情報共有の用途としては監視や経路調停が考えられるが、本実証実験では全ユースケースにおいて、まず「複数ベンダーの複数ロボットの位置情報が一元管理できること」を検証した。

ロボットの位置情報は、以下の2種類を検証

- ①各ロボットの自己位置推定結果 →ユースケースA、C、D
- ②ロボットに後付けした測位システム（屋外はGNSS、屋内はUWB） →ユースケースB

位置情報の集約方法について

- ①については、各ロボットの運行を管理している運行管理システムがビルOS（DWH）に送信（MQTT）
- ②については、測位システムがビルOS（DWH）に送信

位置情報のデータ定義について

今回は屋内外横断運行もあるため、建物内相対座標ではなく、緯度経度（グローバル座標）とした。

緯度経度から算出可能な空間IDのフィールドも用意したが、位置情報を把握する用途（必要とする空間分解能）により使い分けが可能である。

位置情報以外には、向き情報を送信した。機体ステータスやタスクステータスのフィールドを設けたが内容は未定義。

（詳細はインターフェース仕様書を参照）

位置情報の共有（他ロボットの位置情報の把握）については未検証だが、用途により位置精度や取得頻度の要件が異なると思われる。

複数ベンダー・複数ロボットの位置精度を担保するためには、自己位置推定あるいは測位システムの精度を確保する必要がある。

スキャンマッチングによる自己位置推定の場合は、マッチング先となる地図の精度確保が必要である。

ロボットからロボットへの受け渡しのようなユースケースにおいては、直接ロボット間で位置情報を共有する方法も考えられる。

ユースケースBの実証実験においては、一方のロボットにARマーカを貼付し、もう一方のロボットがARマーカを認識して受け渡しの位置を調整した。

## 3.4 位置情報の共有

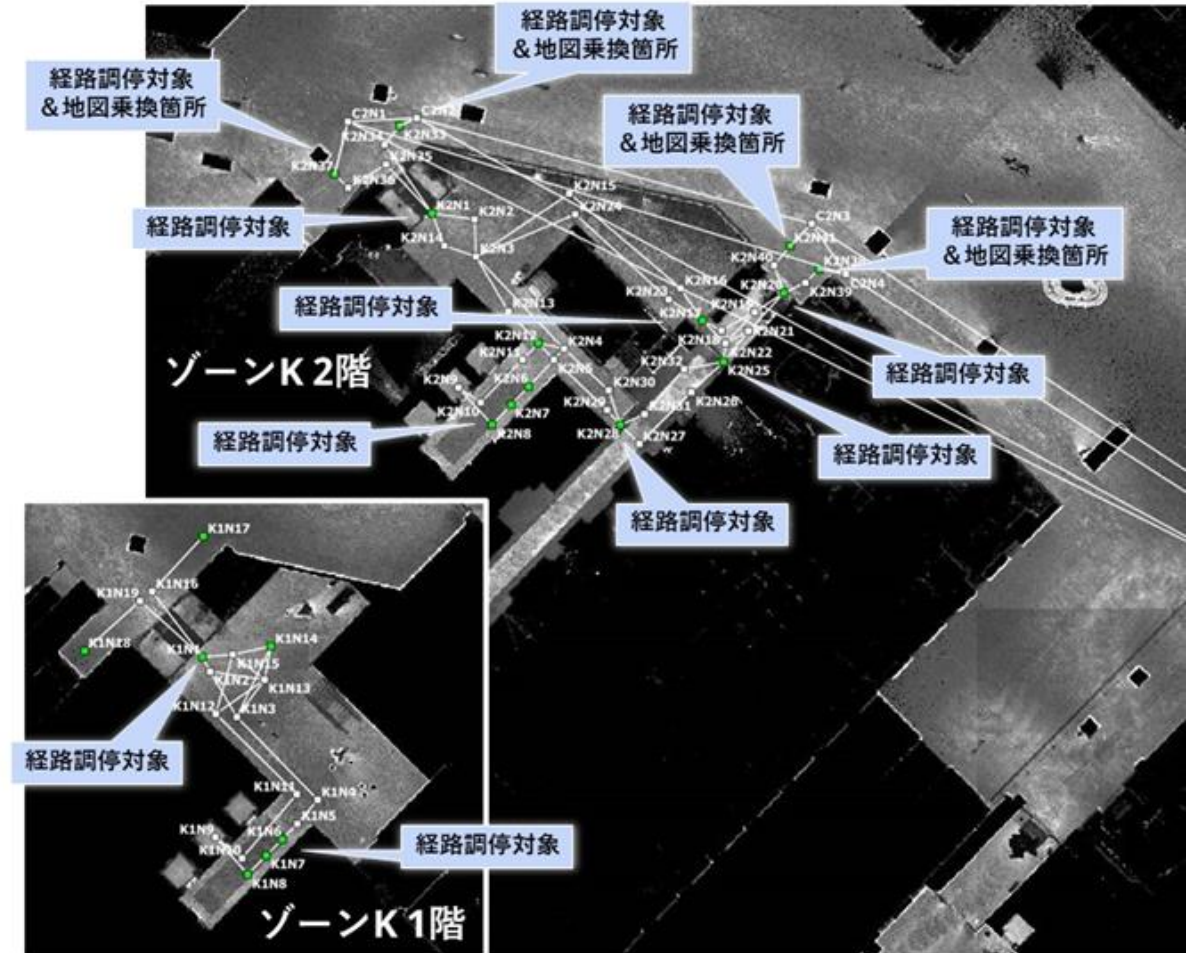
- 本実証では、ロボットの持つ特性や各種条件から、2種類の方法で運行基盤への位置情報の共有を実施した
- 将来的に、種々のロボットが種々のタスクを実行する世界観において、この2種類のオプションをどのような基準で選択し、運用の全体像を構築するかは、大きな議題になると考えられる

比較項目	ロボット/ロボットシステムからの位置情報共有	外付けデバイス（屋外：RTK-GNSS 屋内:UWB)を用いた位置情報共有
対応機器	ロボット等、通信可能かつ計算基盤を持つ機器	・センチメートル級で位置情報演算が可能な外付け機器
導入コスト	対応機器内部の改修、必要な情報の送信機構の開発 開発コストがかかる分、密結合が可能	・デバイス（屋内/屋外測位、通信）購入費用、利用料 ・測位環境構築コスト（設置+測量）※屋内デバイスのみ ・運行基盤への接続API開発
連携可能データ	実証時:位置姿勢情報、機器情報 他情報(画像/映像、センサデータ等)もインターフェース開発次第で連携可能	実証時：位置情報（緯度/経度） 屋内環境はローカル座標から緯度経度への変換が必要 位置情報以外の他情報の連携(画像/映像、センサデータ等)は不可
連携頻度	送信機器側で自由に設定可能	約1fps固定
連携速度 (レイテンシ)	500ms程度 (通信環境に依存)	2000ms程度 (通信環境に依存)
実現可能な機能	<ul style="list-style-type: none"> <li>・位置情報可視化による運行管理 <ul style="list-style-type: none"> <li>↳リアルタイムの位置情報表示トラッキング</li> <li>↳過去経路（移動軌跡）の表示</li> </ul> </li> <li>・機体のステータス監視によるレスキュー</li> <li>・衝突等の統合監視</li> </ul>	<ul style="list-style-type: none"> <li>・位置情報可視化による運行管理 <ul style="list-style-type: none"> <li>↳リアルタイムの位置情報表示トラッキング</li> <li>↳過去経路（移動軌跡）の表示</li> </ul> </li> </ul>
運行基盤からの データ送信	実装次第可能(地震等、緊急データ)	実装不可（外付けデバイスからの情報連携は原則発信のみ）

# 3.5 経路占有情報の共有

ロボットサービス運行基盤の運用における経路占有情報の共有について述べる

- 渋滞やお見合いが想定される狭い空間リソースの表現形式としては、ボクセル/ポリゴン/トポロジカル・グラフ形式等があるが、本実証では走行ルールの共有に用いたトポロジカル・グラフのノードを占有/解除する方法を試行
- トポロジカル・グラフのノード（右図■）の占有/解除（進入可否）状態を共有することで、出入口でのすれ違い、エレベータ前待機等を調停する
- 先にノードを占有したほうが進行し、占有できなかったほうは手前で待機する
- ロボット運行管理システム間の交渉ではなく、押しボタン式信号のボタンを押して他ロボットが通行するのを待つ。



トポロジカル・グラフの例

# 3.5 経路占有情報の共有

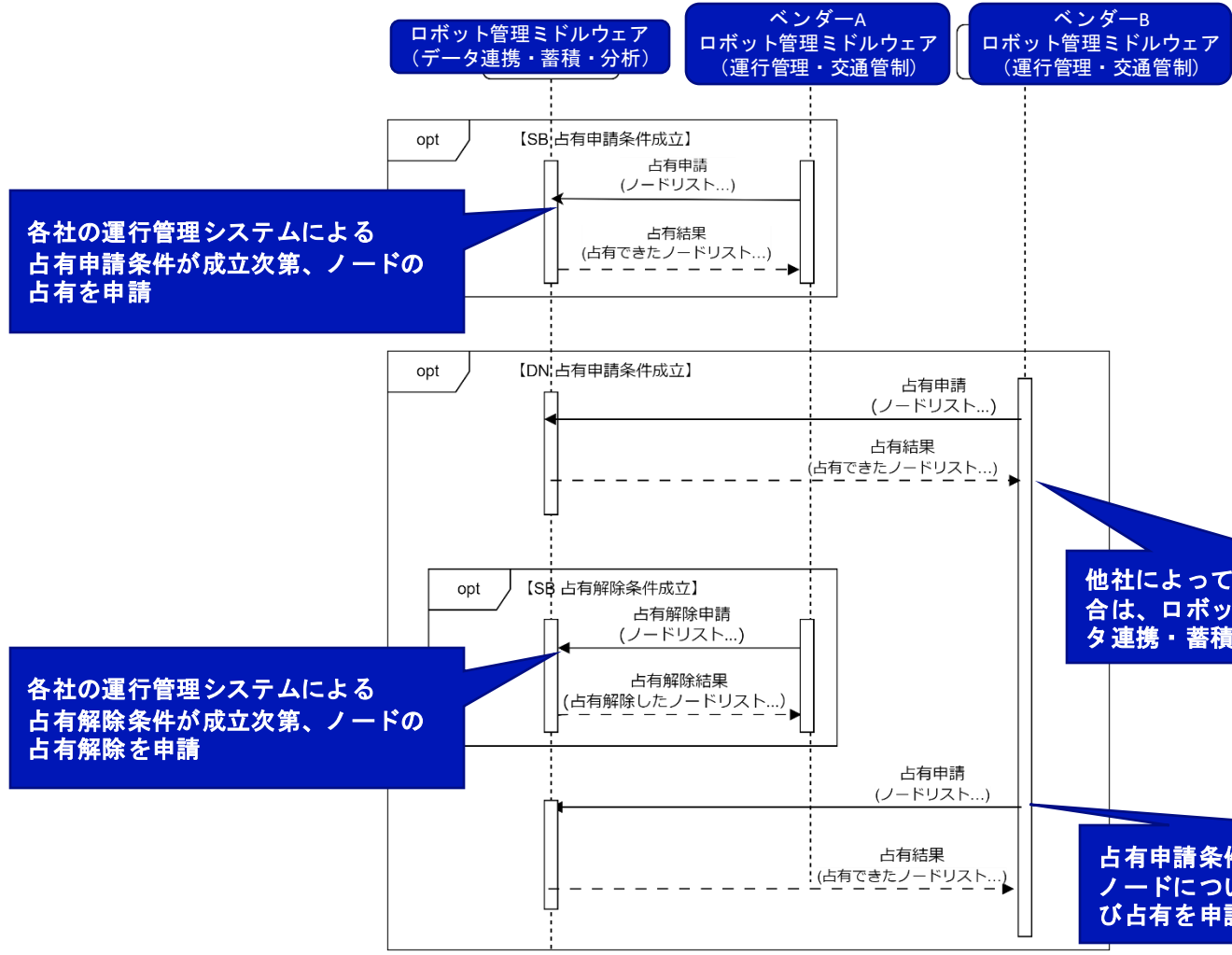
## ロボットサービス運行基盤の運用における経路占有情報の共有について述べる

本実証では、RRIの運行管理地図（トポロジカル・グラフ）を複数ベンダーで共有し、その地図上のノードの占有/解除（進入可否）状態を共有することで、出入口でのすれ違い、エレベータ前待機を調停（本実証では、管制が不要な「先勝ち方式」を試行）

分類	項目	概要	補足
前提条件	経路占有の管理・調停	占有条件/ルールの設定については施設管理者が設定することを想定	本実証ではデータ共有・管理層側での調停が不要な先勝ち方式を試行
	経路調停のタイミング	調停対象ノードへの接近時	複数ベンダーのロボット間で適切に経路調停を行なうためには、ノード占有タイミング等のルールを合わせておく必要がある。
	経路調停の対象	出入口やエレベータ待機場所等の複数台通行で課題となる場所を調停箇所と設定	通路が限られる建物内で目的地までの経路を予約すると、他のロボットが通行できなくなるため、出入口やエレベータ待機場所等の複数台通行で課題となる場所に限って予約する。
調停ノードの占有／解除フロー (詳細は次項参照)	占有申請	各社の運行管理システムによる占有申請条件が成立次第、ノードの占有を申請	本実証では「調停ノードの1つ手前のノードとの距離が所定以内に接近※」を条件とした。 ※調停ノードが占有できなかった場合に、余裕を持って一つ手前のノードで減速/停止できるように定めた。
	占有／待機	占有したロボットのみノードを通行可 (先に占有されていたら、他ロボットの占有が解除されるまで手前で待機)	占有申請条件が成立しているが占有できないノードについては、各社の運行管理側より再び占有を申請
	占有解除	各社の運行管理システムによる占有解除条件が成立次第、ノードの占有解除を申請	本実証では「調停ノード一つ先のノードに到着/経由」を条件とした。
その他	データ連携していないロボットへの対応	未検討（ルール設定/監視の役責問題）	本実証ではデータ共有・管理層と連携しないロボットは対象外とした。
	時間的分割による調停	未検討	時間的に分割して予約する方法も考えられるが、建物内は人流やエレベータ待ちにより通過時間の事前把握が困難と想定される。（「無人航空機（ドローン）と同様の航路登録およびそれに従った航路調停の仕組み」の適用は困難と推察。）

# 3.5 経路占有情報の共有

RRI「運行管理機能インターフェース仕様書 Version 1.0」の「航空管制型調停」を参考に、管制が不要な「先勝ち方式」を検討。調停ノードの占有/解除フローを示す



本実証において複数ベンダー間で共有した占有申請/解除条件

	条件
占有申請	調停ノード一つ手前のノードとの距離が所定以内に接近※
占有解除	調停ノード一つ先のノードに到着/経由

※調停ノードが占有できなかった場合に、余裕を持って一つ手前のノードで減速/停止できると定めた

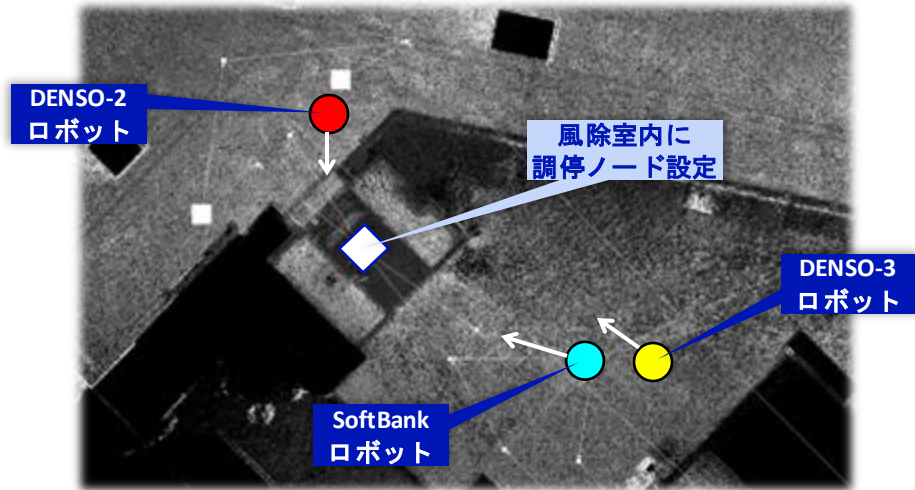
他社によってノードが占有されている場合は、ロボット管理ミドルウェア（データ連携・蓄積・分析）より空を返す

押しボタン式信号のボタンを押して他ロボットの通行を待つ方式

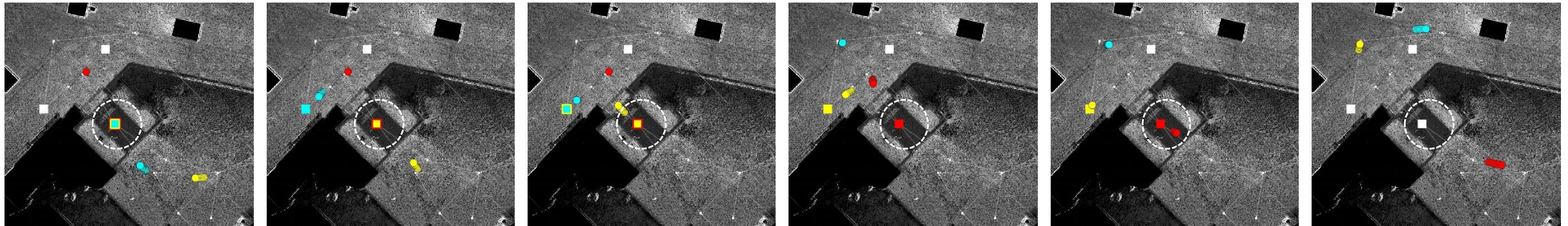
占有申請条件が成立しているが占有できないノードについては、各社の運行管理側より再び占有を申請

# 3.5 経路占有情報の共有

ユースケースCにおける、ゾーンK 2階北西の出入口でのすれ違い時の協調運行の事例を示す



- : SoftBankロボット
- : DENSO-2ロボット
- : DENSO-3ロボット
- : SoftBankロボット占有ノード
- : DENSO-2ロボット占有ノード
- : DENSO-3ロボット占有ノード



①ノードはSBロボットが占有しており、DN-2ロボットとDN-3ロボットが占有申請している。占有できるまでは手間のノードで待機

②SBロボットが通過して占有を解除した後、DN-3ロボットが占有し、DN-2ロボットは占有申請を継続

③DN-3ロボットがノードを通過

④DN-3ロボットが通過して占有を解除した後、DN-2ロボットが占有

⑤DN-2ロボットがノードを通過

⑥DN-2ロボットが通過したのちに占有を解除





# 3.7 分散協調制御に関する課題・今後の展望

本実証で適用した分散協調運行の概要及び課題と考察（今後の展望）のサマリを示す

項目	概要	課題	考察（今後の展望）
ノード占有／解除フロー	<ul style="list-style-type: none"> <li>本実証ではデータ共有・管理層側での調停が不要な先勝ち方式を試行</li> <li>調停ノードへの接近時（本実証での条件「調停ノードの1つ手前のノードとの距離が所定以内に接近」）にノード占有を申請</li> <li>占有したロボットのみノードを通行可。先に占有されていた場合、他ロボットの占有が解除されるまで手前で待機</li> <li>調停ノード通過後に、占有解除条件が成立次第（本実証での条件「調停ノード一つ先のノードに到着/経由」）、ノードの占有解除を申請</li> </ul>	<ul style="list-style-type: none"> <li>本実証の事前確認段階でノード占有申請／解除条件を複数ベンダー間で合わせずに実施した場合、ノード占有条件が異なるため、後方車の追越しが発生</li> <li>経路占有の先勝ち方式について、複数台のロボットからの占有リクエストをFIFOで出し入れするキューイングに対応していないため、リクエストタイミングによっては、後方車の追越しが発生</li> </ul>	<ul style="list-style-type: none"> <li>複数ベンダーのロボット間で適切に経路調停を行なうためには、ノード占有タイミング等のルールを合わせておく必要がある</li> <li>課題に対しては、台数制限した上で、キューイング対応する等が考えられる</li> <li>ノード占有／解除の判定は様々な方式が考えられるが、責任分界や監視等の観点も考慮して決定する必要がある</li> </ul>
トポロジカル・グラフの粒度や付加ルール	<ul style="list-style-type: none"> <li>本実証では最小限のルールを設定したものを複数ベンダーで共有（付与したルールは「双方向／一方向」のみ）</li> <li>出入口やエレベータ待機場所等の複数台通行で課題となる場所を調停ノードと設定</li> </ul>	<ul style="list-style-type: none"> <li>ロボット台数が増加すると、円滑に運行するためにトポロジカル・グラフのノード・エッジの設定が複雑化</li> </ul>	<ul style="list-style-type: none"> <li>走行ルールの追加（左寄り通行等）によりトポロジカル・グラフの複雑化を抑えることは可能と考えられる（ロボット側仕様の高高度化とトレードオフ）</li> <li>中期的調停のみではロボット台数が増加すると破綻してしまうため、複数のロボット運行管理システム間で行なう運行前ルート計画時の長期的調停も必要と考えられる</li> </ul>
建物出入口	<ul style="list-style-type: none"> <li>地図切替を考慮して、建物出入口には退出側と入場側を分けて調停ノードを設置</li> </ul>	<ul style="list-style-type: none"> <li>建物出入口にある風除室では、狭いエリアに複数方向からロボットが進入⇒各ロボットが互いに走行を阻害し、ノードに到達できずにスタックが発生</li> </ul>	<ul style="list-style-type: none"> <li>課題に対しては、エリア（例ゾーンK1F）毎に台数制限を設けて対処することが有効と思われる</li> <li>局所的に密度が高くなるエリアにおける集中制御の要否の検討が必要</li> </ul>
エレベータ	<ul style="list-style-type: none"> <li>エレベータ呼出し待機位置及び手前の待機位置を設けて、全てトポロジカル・グラフの調停ノードとして設定</li> <li>エレベータ内にノードを設けて、エレベータ呼出し待機位置の占有を解除</li> </ul>	<ul style="list-style-type: none"> <li>本実証で運行した最大ロボット台数（4台）では問題なく運行することができた。但し、同じ空間にそれ以上の多数のロボットが密集する場合、待機場所が団子状態になり、歩行者や他のロボットの通行の妨げになってしまうことが考えられる</li> </ul>	<ul style="list-style-type: none"> <li>台数制限や局所的に密度が高くなるエリアにおける集中制御の要否の検討が必要</li> </ul>
建物間移動	<ul style="list-style-type: none"> <li>本実証では、建物外はトポロジカル・グラフの管理外としているため、単純に建物の出入口ノード同士を接続</li> </ul>	<ul style="list-style-type: none"> <li>オープンエリアにおいても、狭いエリアではロボットが密集し、周辺歩行者の通行の妨げや、ロボットのスタックが発生する可能性がある</li> </ul>	<ul style="list-style-type: none"> <li>私道・公道（オープンスペース）においてもルール/占有状態の共有が必要と思われる</li> </ul>

# デジタル庁

Digital Agency