２０２０年度 データ連携基盤技術報告書

スーパーシティのデータ連携基盤に関する調査業務

２０２１年３月

第１版

内閣府

委託先　日本電気株式会社

株式会社日立製作所

アクセンチュア株式会社

一般社団法人データ流通推進協議会

目次

1. はじめに 1-1
1-1. 本報告書の位置付け 1-1
1-2. 参考文献 1-1
1-3. 別紙一覧 1-1
1-4. スーパーシティとしての目指すべき姿 1-3
1-5. データ連携基盤に関する調査の目的 1-7
1-5-1. データ連携基盤の目指すべき姿 1-7
1-5-2. データ連携基盤に関する調査の目的 1-7
2. 調査プロセス 2-1
2-1. データ連携基盤の全体像の明確化 2-1
2-1-1. データ連携基盤の全体像を明確化するための手法 2-1
2-2. （ア）データ仲介（ブローカー）機能 2-3
2-2-1. ブローカーを検討するための手法 2-3
2-2-2. ブローカーの調査 2-7
2-2-3. ブローカーの評価及び改善方法 2-9
2-3. （イ）-① API 共通ルール/標準仕様 2-15
2-3-1. 調査方針 2-15
2-3-2. API 設計/開発・公開・運用プロセス 2-16
2-3-3. API 利用規約テンプレート 2-16
2-3-4. API 標準仕様案 2-17
2-4. （イ）-② API カタログ/開発者ポータル 2-18
2-4-1. API カタログ、及び開発者ポータルの仕様案策定におけるポイント 2-18
2-4-2. API カタログの仕様案策定 2-19
2-4-3. 開発者ポータルの仕様案策定 2-20
2-4-4. 検討プロセス 2-21
2-5. （ウ）データモデルの方針 2-23
2-5-1. 想定課題 2-23
2-5-2. 調査の前提条件 2-23
2-5-3. 調査の進め方 2-24
2-6. （エ）先端的サービスとデータ連携基盤の接続検証 2-27
2-6-1. 先進技術の調査と選定 2-27
2-6-2. 先端的サービスを支える先進技術とデータ連携基盤の接続検証 2-28
2-7. （オ）その他データ分散型のデータ連携基盤を運用する上で必要となる事項 2-31
2-7-1. データ連携基盤における運用面での課題認識 2-31
2-7-2. 課題抽出の観点及び調査方針 2-33
3. 調査結果：データ連携基盤の全体像 3-1
3-1. 先端的サービスの例 3-1
3-2. データ連携基盤の全体像 3-2
3-2-1. Society 5.0 リファレンスアーキテクチャとの関係性 3-2
3-2-2. データ連携基盤に必要となる構成要素の導出（仮説設定） 3-4
3-2-3. 本報告書におけるデータ連携基盤の全体像 3-13

4. 調査結果（ア）データ仲介機能 4-15
4-1. ブローカー調査 4-15
4-1-1. ユースケース調査 4-15
4-1-2. リファレンスモデル・製品事例調査 4-19
4-1-3. 考察 4-22
4-1-4. ブローカー調査より抽出されたブローカーに求められる要件 4-31
4-2. ブローカー評価 4-33
4-2-1. 評価対象のブローカー選定 4-33
4-2-2. ブローカー評価結果 4-40
4-2-3. 考察 4-67
4-2-4. 評価結果から抽出されたブローカー要件 4-74
4-3. ブローカー要件（案） 4-76
4-3-1. 機能要件 4-78
4-3-2. 非機能要件 4-80
4-3-3. ブローカーにて取り扱うデータ種別 4-81
4-3-4. 導入時の方針 4-82
5. 調査結果（イ）①API 共通ルール/標準仕様 5-1
5-1. API 設計/開発・公開・運用プロセス 5-1
5-1-1. はじめに 5-1
5-1-2. 設計/開発プロセス 5-1
5-1-3. 公開プロセス 5-11
5-1-4. 運用プロセス 5-12
5-2. API 利用規約テンプレート 5-17
5-2-1. はじめに 5-17
5-2-2. 利用規約の各項目と解説 5-19
5-2-3. ユースケース事例と関連法令 5-29
5-3. API 標準仕様案 5-34
5-3-1. はじめに 5-34
5-3-2. 基本となる考え方 5-34
5-3-3. API リクエストの標準 5-38
5-3-4. API レスポンスの標準 5-44
5-3-5. 認証方式の標準 5-50
5-3-6. システム構成 5-51
5-3-7. API 標準仕様の今後について 5-52
6. 調査結果（イ）②API カタログ/開発者ポータル 6-1
6-1. 概要・目的 6-1
6-2. 仕様検討 6-3
6-2-1. 基礎調査によるモデルケースの選定 6-3
6-2-2. 基礎調査に基づく仕様検討 6-6
6-3. ヒアリングに基づく仕様の精査 6-22
6-3-1. ヒアリングの考え方（目的、ヒアリング先、ヒアリング項目） 6-22
6-3-2. ヒアリング結果 6-23
6-3-3. 仕様案へのフィードバック 6-25

6-4. 仕様検討結果 6-26
6-4-1. 仕様検討結果 6-26
6-4-2. 将来的に発生し得る課題と想定対処法 6-33
7. 調査結果（ウ）データモデルの方針 7-1
7-1. この文書について 7-1
7-1-1. 位置付け 7-1
7-1-2. 想定読者 7-1
7-1-3. 想定する本書の利用場面 7-1
7-1-4. アーキテクチャや API との関係性 7-2
7-1-5. 対象範囲 7-2
7-1-6. 用語定義 7-2
7-2. データモデルの役割 7-8
7-2-1. 一般論 7-8
7-2-2. ユースケースから見たデータモデル 7-8
7-2-3. データモデルと相互運用性 7-10
7-2-4. スーパーシティにおけるデータモデルの役割 7-12
7-2-5. 都市 OS とデータモデル 7-13
7-3. 推奨データモデル概要 7-15
7-3-1. 検討の背景と目的 7-15
7-3-2. 推奨データモデルの整理原則 7-15
7-3-3. 具体的な要素 7-16
7-3-4. データモデルの検討対象 7-17
7-3-5. 推奨データモデル検討の実施内容 7-19
7-4. 基本データの共通記述方法 7-25
7-4-1. 文字 7-25
7-4-2. 外国語表記・ピクトグラム 7-25
7-4-3. 日時 7-25
7-4-4. 緯度、経度 7-26
7-4-5. 住所等 7-26
7-4-6. 連絡先 7-27
7-4-7. センサーデバイス 7-28
7-5. コード体系・ID 体系 7-31
7-5-1. 地理空間に関するコード 7-31
7-5-2. 時間、時期に関するコード 7-31
7-5-3. 状態に関するコード 7-33
7-6. 推奨データモデル 7-34
7-6-1. 土地 7-34
7-6-2. 建物 7-37
7-6-3. 施設 7-39
7-6-4. 出入口 7-59
7-6-5. 設備 7-60
7-6-6. 道路 7-67
7-6-7. その他の地物 7-68

7-6-8. 交通 7-69
7-6-9. イベント 7-75
7-6-10. センサーデータ 7-79
7-6-11. 建物内、地下街 7-84
7-6-12. 地下埋設物 7-84
7-6-13. 移動オブジェクト 7-85
7-6-14. 自然 7-87
7-6-15. 緊急情報 7-88
7-6-16. 行政情報 7-88
7-7. データカタログのデータモデル定義 7-89
7-7-1. 情報の管理主体 7-90
7-7-2. データセットの基本属性 7-91
7-7-3. データセットの概要 7-91
7-7-4. データの品質 7-92
7-7-5. データセットの利用条件 7-93
7-8. スーパーシティにおけるデータモデル検討手順（データモデルの適用） 7-95
7-8-1. 想定されるスーパーシティサービスのユースケースシナリオ 7-95
7-8-2. スーパーシティサービスにおけるデータセット及びデータモデルを整理するまでの手順 7-95
7-8-3. スーパーシティサービスにおけるデータセット定義及び推奨データモデルの適用 7-96
7-9. 共通的に使われる地理空間データの既存標準規格に関する考慮 7-99
7-9-1. 既存標準規格の調査 7-99
7-9-2. 既存標準規格の調査結果から見るデータセット定義の考察 7-99
7-10. スマートシティ構築における知見を踏まえたデータモデル検証 7-100
7-10-1. 個々のデータモデルに関する検証結果 7-100
7-10-2. データモデルの方針に関する評価 7-100
8. 調査結果（エ）先端的サービスとデータ連携基盤の接続検証 8-1
8-1. 概要・目的 8-1
8-1-1. 先進技術の選定プロセス 8-1
8-1-2. 先進技術の選定 8-4
8-1-3. 検証対象となる「先端的サービスを支える先進技術」 8-5
8-1-4. 調査結果 8-7
8-2. 先端的サービスを支える先進技術とデータ連携基盤の接続検証 8-16
8-2-1. 接続検証のポイント 8-16
8-2-2. 接続検証について 8-17
8-3. まとめ 8-39
9. 調査結果（オ）基盤を長期運用する上で必要となる事項 9-1
9-1. データ連携基盤を長期的に運用するための課題整理 9-1
9-1-1. ヒアリング方針とヒアリング対象 9-1
9-1-2. 自治体・団体におけるデータ連携基盤の長期運用課題 9-7
9-1-3. 課題を解決するための仕組み 9-21
9-2. 陳腐化しない運用の仕組み 9-24
9-2-1. 調査方針と調査対象 9-24

9-2-2. データ連携基盤が公開する API やデータの鮮度を維持する仕組み 9-25
9-2-3. 時代の変化に応じて柔軟にシステムを拡張する仕組み 9-32
9-3. 安心してデータを扱える仕組み..... 9-33
9-3-1. 透明性を重視したパーソナルデータ管理の検討 9-34
9-3-2. 継続的なパーソナルデータ品質管理の検討 9-37
9-4. まとめ 9-39
10. まとめ 10-1

1. はじめに

1-1. 本報告書の位置付け

本報告書は、各指定区域でのデータ連携基盤の構築に際し、令和 2 年 9 月に公開された、スーパーシティ／スマートシティの相互運用性の確保等に関する検討会の最終報告書をうけ、データ連携基盤として、具備すべき機能、考慮すべき事項についての調査結果を示したものであり、各区域の企画、実装、運営を行うそれぞれの参加者が参照することを想定している。

1-2. 参考文献

本書における主な参考文献を表 1-2-1 に示す。

表 1-2-1　主な参考文献

No.	文書名
1	「スーパーシティ構想」について （内閣府） https://www.kantei.go.jp/jp/singi/tiiki/kokusentoc/supercity/openlabo/supercitycontents.html
2	スーパーシティ／スマートシティの相互運用性の確保等に関する検討会　最終報告書 （スーパーシティ／スマートシティの相互運用性の確保等に関する検討会） https://www.kantei.go.jp/jp/singi/tiiki/kokusentoc/supercity/supercity_kentoukai.html
3	スマートシティリファレンスアーキテクチャ　ホワイトペーパー （戦略的イノベーション創造プログラム（SIP）第 2 期／ビッグデータ・AI を活用した サイバー空間基盤技術のアーキテクチャ構築ならびに実証研究事業　スマートシティ分野アーキテクチャ構築） https://www8.cao.go.jp/cstp/stmain/20200318siparchitecture.html
4	パーソナルデータリファレンスアーキテクチャ （戦略的イノベーション創造プログラム（SIP）第 2 期／ビッグデータ・AI を活用した サイバー空間基盤技術のアーキテクチャ構築ならびに実証研究事業　パーソナルデータ分野アーキテクチャ構築） https://www8.cao.go.jp/cstp/stmain/20200318siparchitecture.html

1-3. 別紙一覧

本書における別紙の一覧を表 1-3-1 に示す。

表 1-3-1　別紙一覧

No.	タイトル	説明
ア-1	調査結果詳細	（ア）で実施した「ユースケース調査」及び「リファレンスモデル・個別製品調査」の調査結果の詳細情報
イ②-1	想定利用シーン一覧	API カタログ・開発者ポータルそれぞれについて、整理したアクターごとの利用シーンをまとめた一覧
イ②-2	機能一覧案	既存事例の基礎調査に加え、アクター検討・利用シーン検討を通じて整理した、API カタログ・開発者ポータルが具備すべきと想定される機能の一覧

No.	タイトル	説明
イ②-3	ヒアリング結果 （シビックテック推進団体）	本テーマの検討内容に関して、利用者・開発者の観点からシビックテック推進団体に対してヒアリングした内容と回答の一覧
イ②-4	ヒアリング結果 （運用事業者）	本テーマの検討内容に関して、運用者の視点から既存事例の運用事業者に対してヒアリングした内容と回答の一覧
イ②-5	機能一覧案（再評価後）	「別紙イ②-2 機能一覧案」について、ヒアリングの結果を基に内容・区分を見直した機能の一覧
ウ-1	ユースケース集	「3-1(2)先端的サービスイメージ」「7-8 スーパーシティにおけるデータモデル検討手順（データモデルの適用」にて検討対象とするユースケースの詳細をまとめた資料
ウ-2	データ定義	別紙ウ-1 で定義される各ユースケースで用いるデータを定義した資料
ウ-3	地理空間情報の既存企画調査及びデータモデル調査報告書	地理空間情報の既存規格及びデータモデルの調査結果、及び考慮すべき事項を整理した報告書。「7-6 推奨データモデル」「7-8 スーパーシティにおけるデータモデル検討手順（データモデルの適用」の参考として用いた資料
エ-1	先端的サービスを支える 技術一覧	先端的サービスを支える先進技術として推奨する先進技術について調査結果を整理した一覧
オ-1	仮説課題及び ヒアリング結果一覧	データ連携基盤の長期運用課題に関する仮説課題、及び自治体等へのヒアリング実施結果一覧
オ-2	ヒアリングシート	データ連携基盤の長期運用課題に関する自治体等へのヒアリング実施時に使用したヒアリングシート
オ-3	安心してデータを扱える 仕組みの検討	安心してデータを扱えるための仕組み（透明性を重視したパーソナルデータ管理の検討、継続的なパーソナルデータ品質管理の検討）に関する詳細検討資料

1-4. スーパーシティとしての目指すべき姿

(1)国内外のスマートシティの現状（背景）

(a)都市課題の複雑化

経済発展に伴って都市化が進行すると、渋滞等の交通問題、大気汚染等の環境問題、人が密集することによる犯罪発生の懸念等、様々な問題が発生する。これらの問題は様々な分野の問題が複雑に絡み合っているパターンが多いことから、単一の分野に対応する施策だけでは問題解決に至らないケースも発生している。

また、都市化が進行する一方で地方の過疎化が進行しているケースも往々にしてある。例えば、地方経済の衰退や交通難民、インフラの老朽化等が連鎖的に発生している例がある。加えて、都市化が進行し成熟した後は、高齢化や人口減少、それに伴う労働力不足が発生し、地方と同じく課題の悪循環が発生する。

これらの都市課題をデジタルの活用によって解決しようとする取り組みの 1 つがスマートシティであり、これまでも国内及び諸外国で進行している。その状況を次項(b)(c)に示す。

(b)世界のスマートシティの状況

欧州をはじめとした諸外国には、スマートシティへの取り組みが先行している国がある。例えば住民がスマートシティの取り組みに参画しやすいようにユーザインターフェースの 1 つとしてスマートフォンのアプリケーションを活用している例や、画期的なサービスを生み出すためにイノベーション創発に着眼しスタートアップや地元企業の参画を拡大するような例がある。このようなスマートシティの取り組みの中心には、データ連携基盤が位置しているケースが多い。

(c)我が国のスマートシティの状況

我が国においては、2016 年に第 5 期科学技術基本計画において「Society 5.0」が策定された。これは、先端のデジタル技術を活用して我が国が直面する諸課題を解決し、人間中心の超スマート社会を実現し、かつ経済的再生を成し遂げようというものである。Society 5.0 の実現は、我が国のみならず世界の様々な課題の解決にも通じ、国連の SDGs（Sustainable Development Goals：持続可能な開発目標）の達成にも貢献するものである。この Society 5.0 の実現に向けて、現在様々な地域においてデジタル活用により都市課題を解決する挑戦が進んでいる。この挑戦は、地域ごと及び分野ごとの課題解決に向けて実証事業を中心に進展しつつある一方で、推進上の課題も顕在化し始めている。それらの課題について次項(2)に詳しく示す。

(2)スマートシティ推進上の課題

(a)複雑な都市課題の解決に向けて

前述した複雑化している都市課題の解決に向けては、以下 3 点のポイントを踏まえた全体最適化が必須である。

1 点目は複数地域に跨る広域での全体最適化である。例えば、道路や河川は 1 つの地域に閉じるものではなく多くが隣接の地域に繋がっている。通勤・通学者も隣接の地域等に自由に往来している。これを防災分野に当てはめてみれば、隣接地域の河川の氾濫情報やそれに伴う道路の通行可否等の交通情報が的確な避難誘導を促進し、隣接地域に往来する住民や通勤・通学者等への当該情報の提供が逃げ遅れの防止に大いに繋がる。したがって、地域に閉じずに地域間を跨ぐような情報の連携が必須となる。

2 点目は複数領域に及ぶ情報連携の観点である。例えば同じく防災分野において、特に有事の際には要介護者への配慮や訪日外国人への情報提供をはじめ、医療・福祉・交通・物流・インフラ等、異

なる組織が保有する情報の連携が必須である。

3 点目はステークホルダとの合意形成の観点である。課題解決に向けてある施策を打ち出せば、その地域の住民や事業者等、個々の利害関係が地域全体の課題解決の推進を遅滞させるケースもある。細部に注目すれば個々に利害関係が発生してもマクロ的には地域の発展に繋がるような着地点を見出し、各ステークホルダで合意を形成していくことが必須である。

また、全体最適化を進める一方で個別最適化の視点も忘れてはならない。スマートシティの目的は都市課題の解決にあり、複雑化した課題の解決には上記 3 点の全体最適化が必須であるが、都市の主役は住民であり、住民目線で課題解決するためには究極的には個々の住民に寄り添った対応が必要となる。住民には、老若男女、外国人、障害者・健常者、学生・就労者、単身・家族等、様々な立場があり、様々な暮らしや経済活動がある。こうした個々の住民ごとに個別最適化されたサービスを提供していけることが求められる。

(b)デジタル化に加えて戦略・政策、組織等の運用に関する施策も重要

データ連携基盤がデジタル化の中心にあることによって、スマートシティ化そしてデジタル活用が推進されていることから、課題解決に向けた一定の効果は得られているものの、その効果を最大化していくためにはさらに以下 2 点の施策が必要である。

1 点目は、都市の DX（Digital Transformation: デジタルトランスフォーメーション）に向けて、都市経営のリーダーを担う人材とそれを支えるアーキテクト人材の存在である。なぜなら複雑化した都市課題の解決は容易ではなく、解決のために実行する施策は成果に結びつくまでに時間を要するケースがほとんどだからである。そうした中で、将来目指すべきビジョンを示しながら、施策の継続や方向修正の判断を行うためには強力なリーダーシップを持ったリーダーが必要である。また、リーダーを支え、リーダーの志を実現するために、戦略・戦術レベルに落とし込んで施策を具体的に推進していくためのアーキテクト人材の存在も必要である。

2 点目は、地域の住民や事業者がまちづくりに参画し、自ら行動変容するよう後押しする仕組みの構築が重要である。特に住民は受益者である一方で、まちとまちづくりに欠かせない構成員であるため、その意識醸成が必要である。また、このようにまちづくりに参画するステークホルダが増えるとそこには個々の利害関係が生まれるため、これらを解決するためには協議会等による意識醸成や合意形成の場を設置していく必要がある。

(c)パーソナルデータの安全な活用

地域課題の解決のためには、(a)で述べたとおり究極的には個々の住民に寄り添った対応が必要であり、国や自治体が提供するまちなかに設置した IoT 機器から収集したデータや地理データ等のオープンデータに加えて、住民が保有するパーソナルデータ（例えばウェアラブルデバイスから取得したバイタルデータ等）の活用が必要となる。前述した防災分野における「要介護者への配慮」もパーソナルデータ活用の例である。パーソナルデータ活用に際しては、個人情報保護を考慮しながらそのデータの価値を有効活用するため、オプトイン・オプトアウト、セキュリティ担保への対応が必須となる。一部海外で先行する監視社会を前提としたスマートシティ、具体的にはオプトイン・オプトアウトが十分に機能していない状況でのパーソナルデータ活用ではなく、住民（個人）が主体のパーソナルデータ活用、すなわち利用するデータの発生源が住民である場合は、住民への説明を確実に実施することや、住民の意思により利用や削除ができることが必要となる。加えて、データの所在は民間企業や住民にあることを前提に活用の在り方を検討する必要がある。

他方、日本では個人を保護する規制が情報活用の阻害要因となっている場合がある。例えば、各地域の病院やクリニックが共通の症例について情報共有したい場合でも、地域ごとに定めている条例に

よって個人情報ルールと必要な手続きが異なるため、情報をやり取りするためには手間やコストがかかる場合がある、といった課題がある。

(3)目指すべきスーパーシティの姿

スーパーシティは、様々な最先端技術を実装した都市を示す。情報通信技術を活用して都市のスマート化を目指すという意味では、スマートシティと目指す方向性は同じではあるが、現状のスマートシティに関する取り組みの多くは、エネルギーや交通など、個別の分野に特化した技術の実証に留まっているという課題がある。

(a)スーパーシティの実現に必要となる要素

スマートシティ推進上の課題を踏まえた上で、住民目線で都市課題を解決するため、目指すべきスーパーシティの姿として以下の要素が必要となる。すなわち、住民の参画、ビジョンと目標の設定、必要な時に必要なデータを領域を跨いで連携・共有すること、及び個人情報保護の観点を踏まえた上でのパーソナルデータの活用である。加えて、新しい取り組みにスピード感を持ってチャレンジするためには、特区による規制緩和も必要となる。それらについて以下に詳しく示す。

(b)住民が参画し住民目線で創る 2030 年頃のまるごと未来都市

スーパーシティの取り組みは、都市課題を解決し、住民の豊かで希望ある暮らしを実現することが原点となる。その実現を支えるためには、スーパーシティ構想に基づく地域社会の柔軟で持続可能な仕組みを確立する必要がある。これには、住民と様々なステークホルダが同じビジョン（2030 年頃のまるごと未来都市）を共有しながらバックキャストで必要なサービスを創り出し、これから起こる様々な環境変化に適応しながらサービスを持続させていくエコシステムの構築が必要である。加えて、住民は単なるサービスの受益者ではなくデータを生成する大事な構成員でもあることから、オープン（参画の間口を広げること）であることはスーパーシティ構想にとって非常に重要な概念である。オープンの意味合いでは、前述の住民の参画だけでなく各ステークホルダに対してオープンであることが重要だが、課題解決に向けてサービスを提供する事業者等の参入しやすさもスーパーシティ構想の実現に向けては重要な要素である。

(c)ビジョン、目標の設定

まちづくり、特にスーパーシティにおいては、未来のまちがどのようになっているべきか、市民の行動がどのようになっているべきかといった「ビジョン」を定め、そのビジョンをステークホルダ間で共有し、その上で施策をバックキャストで検討することが重要である。また、ビジョン実現に向けての達成度は、間接的な定量的評価ではなく、可能な限り直接的な定量的評価、すなわち住民の課題解決に直結していてそれがどの程度進捗しているのかや、住民の行動変容を起こしているのかが明確となる指標とすることが重要である。

以上のビジョン策定及び共有、明確な指標及び施策の評価によって、より実効性のある PDCA サイクル（Plan：計画⇒Do：施策の実行⇒Check：達成度の評価⇒Action：改善に向けた行動）を回すことが可能となる。具体的には、必要に応じて施策の継続や別の施策への転換等の Action をより明確に行うことができるため、ビジョンの実現に大きく寄与するものである。

(d)必要な時に必要なデータの領域を跨いで連携・共有し利活用することの必要性

複雑化している都市課題はその原因が複数分野に跨っているケースが多いことから、その課題の解決に向けては、複数分野を跨いだ施策（サービス等）が有効であり、そのためには複数分野（特に、

住民の生活に繋がり住民の生活をより良くする医療や教育といった領域を含む）を連携できるデータ連携基盤が必須である。データ連携基盤は、スーパーシティの実現にあたり、各エリアで構築されるインフラであり、このデータ連携基盤とサービスに協調領域、競争領域を内包することで、そのポテンシャルを高めることができる。協調すべき領域は、ベンダロックせず誰でも使えるオープンなデータ連携基盤とオープン API である。競争すべき領域は、住民目線の課題解決のためのサービス開発、及びより高度な課題解決を目指すためのデータ連携基盤としての独自機能である。競争によってより良いサービスやデータ連携基盤の独自機能を事業者等が実現すると考えられる。これに加えて、住民をはじめとした多くのステークホルダが関わることでイノベーティブな新たなサービスが生み出され相乗効果が生まれる。

このサービスを一過性のサービスで終わらせることないように、持続可能な状態となるようなキャッシュフローの仕組みも含めてエコシステムの構築が必要である。また都市間を跨いで共通のサービスを展開する場合等においては、サービスそのものが相互にデータの互換性を保ち相互運用性を担保することが重要である。これにより、後から共通サービスを導入する都市でもより早くコストも抑えることが可能であり、それが全国的な普及促進に繋がる。

また、より個々の住民に寄り添ったサービスを展開し課題解決に資するためにはパーソナルデータの活用が重要となる。パーソナルデータを取り扱う上では、プライバシーインパクトを考慮した分散方式を採用し、適切なセキュリティ対策を施した上で、令和 2 年 6 月 12 日に公布された「個人情報の保護に関する法律等の一部を改正する法律」にも準拠した、オプトイン・オプトアウトの考え方を導入することが必要となる。

(e)特区による規制緩和

スーパーシティは、高度成長期に培われた個々の制度・規制を新たな時代に合わせて大胆に見直す挑戦となる。だが、先進的な取り組みは法律の壁に阻まれる場合が多く、さらに複数の法律が関係する場合があるため、スピーディーな取り組みの推進に大きく影響する。例えば、ドローンや自動車の自動運転の実証実験を行う際には、電波利用等に関して関係省庁等の許可等を個々に受ける必要があるが、先進的な取り組みであればあるほど各関係者との調整に時間を要する場合がある。このことからも、特例措置による規制改革が求められている。特例措置によってまずは区画を区切って各法の許可を得ながら効果測定と想定外の問題が発生しないかを検証し、社会課題の解決を図る生活実装実験を行うことが必要である。さらに実験から実装に向けては、強力なリーダーシップと戦略遂行のためのアーキテクト人材、そして多様なステークホルダの参画、とりわけ住民のまちづくりの参画が重要である。

1-5. データ連携基盤に関する調査の目的

1-5-1.　データ連携基盤の目指すべき姿

複数分野に跨る未来の生活サービスを前倒しで実現するスーパーシティにおいては、様々な主体とともに、異なるサービス間相互の相乗効果を住民目線で追求することができるような多様性が重要となる。そのためには、それぞれのサービスがどのような仕様・技術を用いる場合であっても、必要な時に必要なデータの相互連携・共有を可能とするオープンなデータ連携基盤が必要となる。

図　1-5-1-1 に示すとおり、スーパーシティにおけるデータ連携基盤は、API[1]を介し、様々な主体から提供されるデータ（行政・住民データ、地理空間データ、企業保有データ、個人データ等）を集約及び適切な形式に変換し、配信する機能を提供するものである。データ連携基盤の API は、誰もがデータ連携基盤に接続でき、かつサービス間データ連携により複数分野を跨った便益を提供可能とするために、国が整備する API カタログ上で公開する。

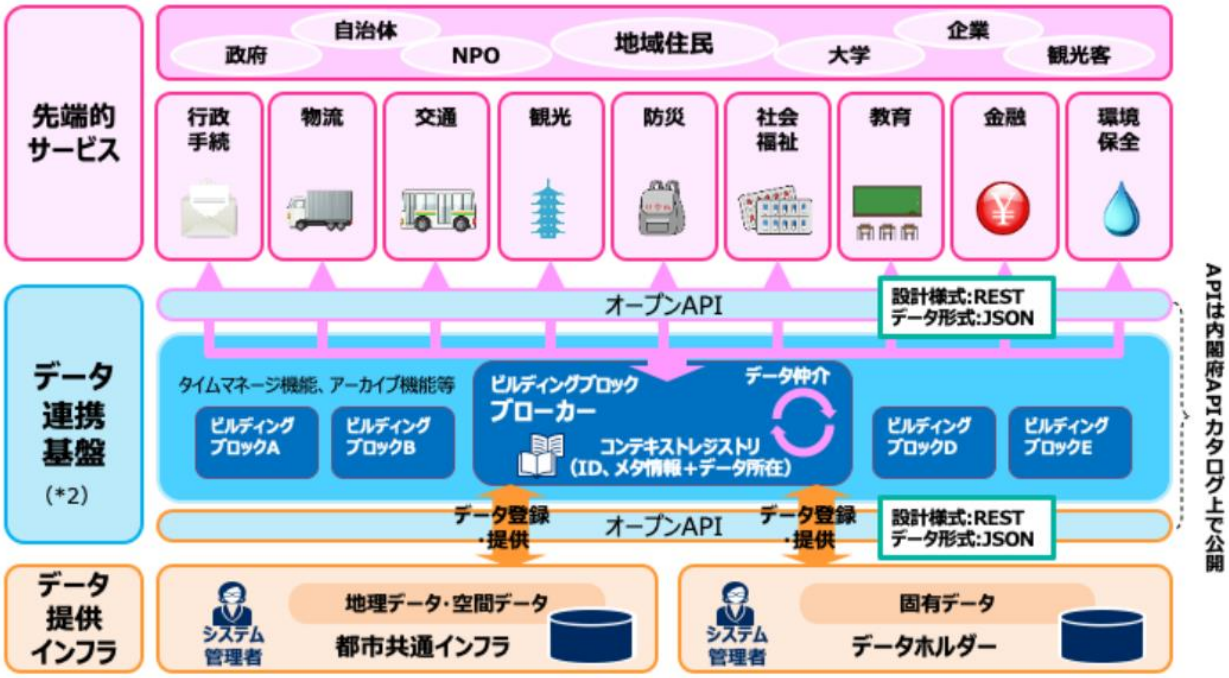


図 1-5-1-1　スーパーシティ構想におけるデータ連携基盤

1-5-2.　データ連携基盤に関する調査の目的

データ連携基盤に関する調査業務においては、上述した「データ連携基盤の目指すべき姿」に基づき、データ連携基盤の整備に向けた最適なシステムの調査を進める。

上述のとおり、スーパーシティは様々なデータを分野横断的に収集・整理して提供する「データ連携基盤」を軸に、住民福祉や利便の向上を図る都市である。日本国政府においては、スーパーシティの実現に向け、データ連携基盤の整備、先端的サービスの構築支援、各エリアにおける事業計画作成・協議会開催支援等、多面的な支援体制の整備を進めている（図 1-5-2-1）。

[1] API（Application Programming Interface）：あるサービスやアプリケーションにおいて、その機能や管理するデータなどを他のサービスやアプリケーションから呼び出して利用するための接続仕様等。

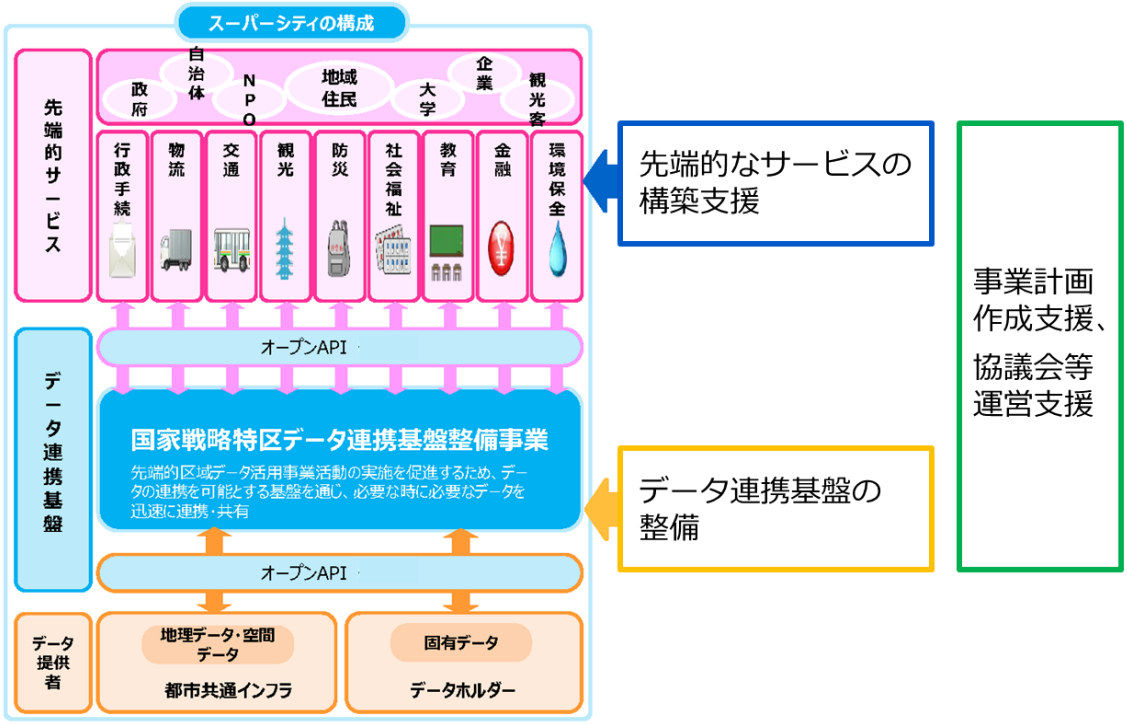


図 1-5-2-1　スーパーシティ構想の推進に係る施策

このうち、データ連携基盤に関しては、スーパーシティに選定された各エリアにおいてそれぞれ整備するものであるが、スーパーシティに選定された複数の都市間連携を図ることに加えて、スーパーシティのエリア外でも汎用的に使用されていくためには、各エリアが別々に設計をするのではなく、共通的な仕様等を決定していく必要がある。データ連携基盤に関する調査業務は、データ連携基盤の共通的な仕様等を調査・検討することで、各エリアにおけるデータ連携基盤の効率的かつ効果的な構築・運営を支援することを目的に実施されるものである（図 1-5-2-2）。

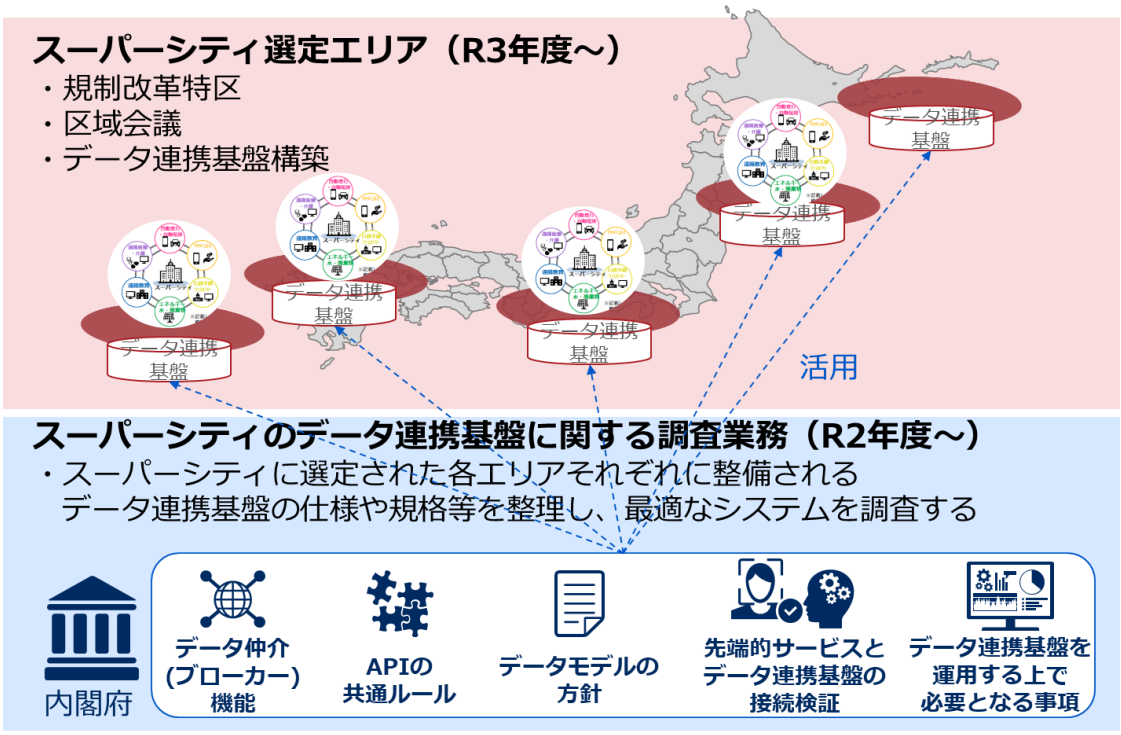


図 1-5-2-2　本報告書の位置付け

データ連携基盤には、相互運用性確保・共通仕様案策定等スーパーシティの普及促進を図る協調領域と、事業者独自の技術やアイデアにより付加価値提供を図る多様性を許容した競争領域とが含まれるが、本報告書においては協調領域を中心に述べる。本報告書の全体像を図 1-5-2-3 に示す。

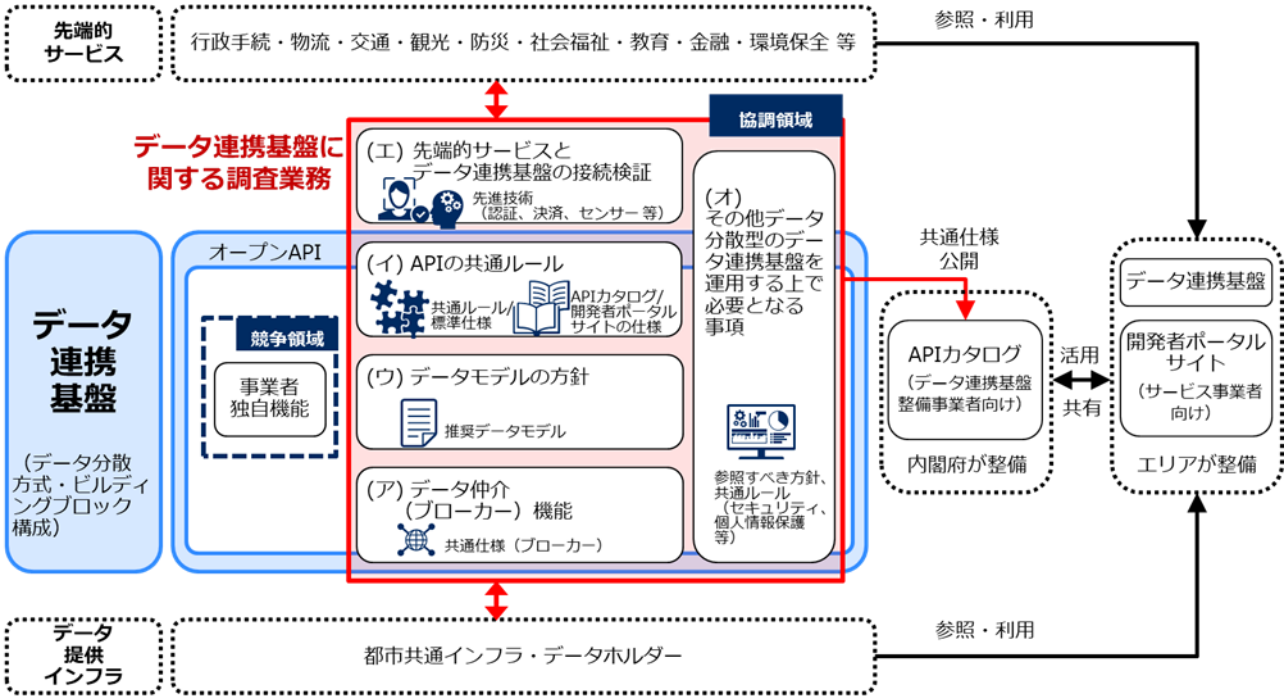


図 1-5-2-3　本調査業務の全体像

本報告書では、サービス間・都市間・分野間等の相互運用性確保を実現する上で中核となる「(ア) データ仲介（ブローカー）機能」、「(イ) API の共通ルール」、「(ウ) データモデルの方針」について調査を進め、調査結果を基に検討した仕様案について述べる。また、(イ) API の共通ルールについては、API の設計、公開に際する共通ルールを調査する「(イ) -①API 共通ルール/標準仕様」と、API 公開サイト仕様を調査する「(イ) -②API カタログ/開発者ポータル」に章分解して整理する。

「(エ) 先端的サービスとデータ連携基盤の接続検証」では、ニーズ、汎用性、将来性・拡張性を踏まえ、データ連携基盤と親和性の高い技術について述べる。

「(オ) その他データ分散型のデータ連携基盤を運用する上で必要となる事項」では、運用面（ルール）及びシステム面（技術）の両方の視点で長期運用に必要な事項について述べる。

上記の調査結果を踏まえて、「データ連携基盤の全体像」として調査結果をまとめる。

以下に、本書における各調査項目に対応する章と、各章に対する想定読者を示す。

表 1-5-2-1　各調査項目の対応章と想定読者

調査項目	調査プロセス	調査結果	データ連携基盤推進主体者 (※1)	データ連携基盤構築事業者 (※2)	サービス構築事業者 (※3)
データ連携基盤の全体像	2-1 章	3 章	●	●	
(ア) データ仲介（ブローカー機能）	2-2 章	4 章		●	
(イ) ①API 共通ルール/標準仕様	2-3 章	5 章		●	●
(イ)②API カタログ/開発者ポータル	2-4 章	6 章		●	
(ウ) データモデルの方針	2-5 章	7 章	●	●	●
(エ) 先端的サービスとデータ連携基盤の接続検証	2-6 章	8 章			●
(オ) 基盤を長期運用する上で必要となる事項	2-7 章	9 章	●	●	

●: 読者として想定している範囲

※1：スーパーシティ、データ連携基盤の導入を推進する団体を指す。本書では地方自治体の職員等を想定して記載。

※2：データ連携基盤を構築する団体を指す。本書では IT ベンダ等の技術者を想定して記載。

※3：データ連携基盤を活用したサービスを構築する団体を指す。本書では技術者を想定して記載。

2. 調査プロセス

2-1. データ連携基盤の全体像の明確化

2-1-1.　データ連携基盤の全体像を明確化するための手法

(1)背景

2018 年に内閣府よりスーパーシティ構想が打ち出されている。ここでは、問題解決型思考のアプローチならびにビッグデータの分野横断的な活用及び国家戦略特区を活用した規制改革を用いた技術実装といった考え方や制度の活用により、第四次産業革命を体現する最先端技術の都市への実装を目指すことが掲げられている。未来の生活を前倒しして実現するスーパーシティでは、様々な生活サービスを展開する複数のプレイヤーの協業が不可欠である。住民の抱える各種の社会課題・ニーズに対して、AI やビッグデータ等、最先端の技術を活用した複数の生活サービスで住民の暮らしを支えるためには、様々なプレイヤーが協調と競争を重ね合わせる中で生み出される創発的なサービスを基に、利用者の意見を参考にして改善及び改良を重ねていくエコシステムの構築が重要となる。

エコシステムの構築には、官民が提供する様々な暮らしのサービスが、健全に競争しつつも、サービス間相互で、必要な時に、必要なデータを連携・共有できるよう、オープンに相互接続できるデータ連携基盤が必要となるが、これまでデータ連携基盤全体の具体的なスコープや要件は明らかになっておらず、データ連携基盤を構築する自治体又は地域ではステークホルダ間で認識に齟齬が生じるおそれがある。本報告書ではデータ連携基盤全体像を明らかにし、そのスコープの明確化を行う。

(2)データ連携基盤の全体像を明確化するプロセス

データ連携基盤の全体像を明確化するにあたり、内閣府地方創生推進事務局より公表されている「『スーパーシティ』構想について」に掲げられた「スーパーシティ」構想イメージ（A 市～D 市）及び「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会」、「スマートシティリファレンスアーキテクチャ」等の参考文献を基に、先端的なサービスの例を挙げた上で、データ連携基盤に関する全体像の仮説を設定する。次に、この仮説の検証を実施する。本報告書での他の実施事項（ア）～（エ）の検討結果より、データ連携基盤全体のスコープや必要とされる要件を明確にする。これらを踏まえて、上記の仮説を見直し、データ連携基盤の全体像を確定させるというプロセスで進める。

仮説を設定するにあたっては、「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」に記載されているサービス及び相互運用性に関する以下の観点を踏まえ検討をすることとする。

- サービス

データ連携基盤は、異なるサービス間の相乗効果を追求するため、データの連携を柔軟かつ、効果的に進めることを目的に実装される必要がある。各サービスが、多種多様な仕様及び技術を用いる場合であっても、容易に相互にデータを連携及び共有できることが必要となる。住民が移動することにより、サービスも移動する。住民は 1 つのサービスに依存することなく、同様あるいは類似のサービスを生活に取り入れていくことが望ましい。

- 相互運用性

「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」では、都市間、都市プラットフォーム間、サービス間、そして将来の自分に対してデータを相互利用し、サービスを継続して利用できることを相互運用性として定義し、原則として以下 3 点が挙げられている。

①住民が抱える課題を解決し、便益がもたらされているかどうかを重視する

②データ連携では、相乗効果の追求を徹底する

③データの管理に透明性を持つ

2-2. （ア）データ仲介（ブローカー）機能

前項で述べたとおり、複数分野に跨る未来の生活サービスを前倒しで実現するスーパーシティにおいては、必要な時に必要なデータの相互連携・共有を可能とするデータ連携基盤が必要となる。ブローカーは様々な主体が提供するデータを集約し、適切な処理を経た上で公開する機能を提供するものであり、データ連携基盤の中核をなす機能である。汎用的かつ共通的なブローカー仕様を協調領域として整備することは、サービス間・都市間・分野間等の相互運用性を確保し、スーパーシティの普及拡大を図る上で重要である。

ブローカーにおけるデータ仲介方式は、大きく分け、ブローカー内にデータを蓄積して一元管理する「データ蓄積方式」と、ブローカー内にデータを蓄積せずデータが必要となる際に都度データ提供者へのアクセスを行う「データ分散方式」とに分類できる。スーパーシティにおいてはプライバシーインパクトが懸念されるデータや個人情報等をブローカー内に集積してしまう可能性を加味し、原則としてデータ分散方式を利用することが求められる。その一方、正確性やリアルタイム性等の性能改善に寄与すると考えられる静的データ等については、データを蓄積することも有効と考えられる。

以上に鑑み、データ分散方式を前提としつつ、分野/都市横断的なデータ分析との親和性、汎用性、既存システムとの接続性、コスト、スケーラビリティ等の観点も加味して、ブローカーとして持つべき機能や蓄積すべき静的データを調査し、これらを実現可能な技術・実装について、性能評価を含めて比較評価を行う。

以下、ブローカーを検討するための手法、ブローカーの調査、ならびにブローカーの評価及び改善方法について説明する。

2-2-1. ブローカーを検討するための手法

汎用的かつ共通的なブローカー仕様の整備に向け、先端的サービスのユースケース、各種団体が公開するリファレンスモデルや製品事例を調査し、ユースケースと技術動向の両者を踏まえてブローカーが備えるべき要件を整理する。図 2-2-1-1 に検討の全体プロセスを示す。

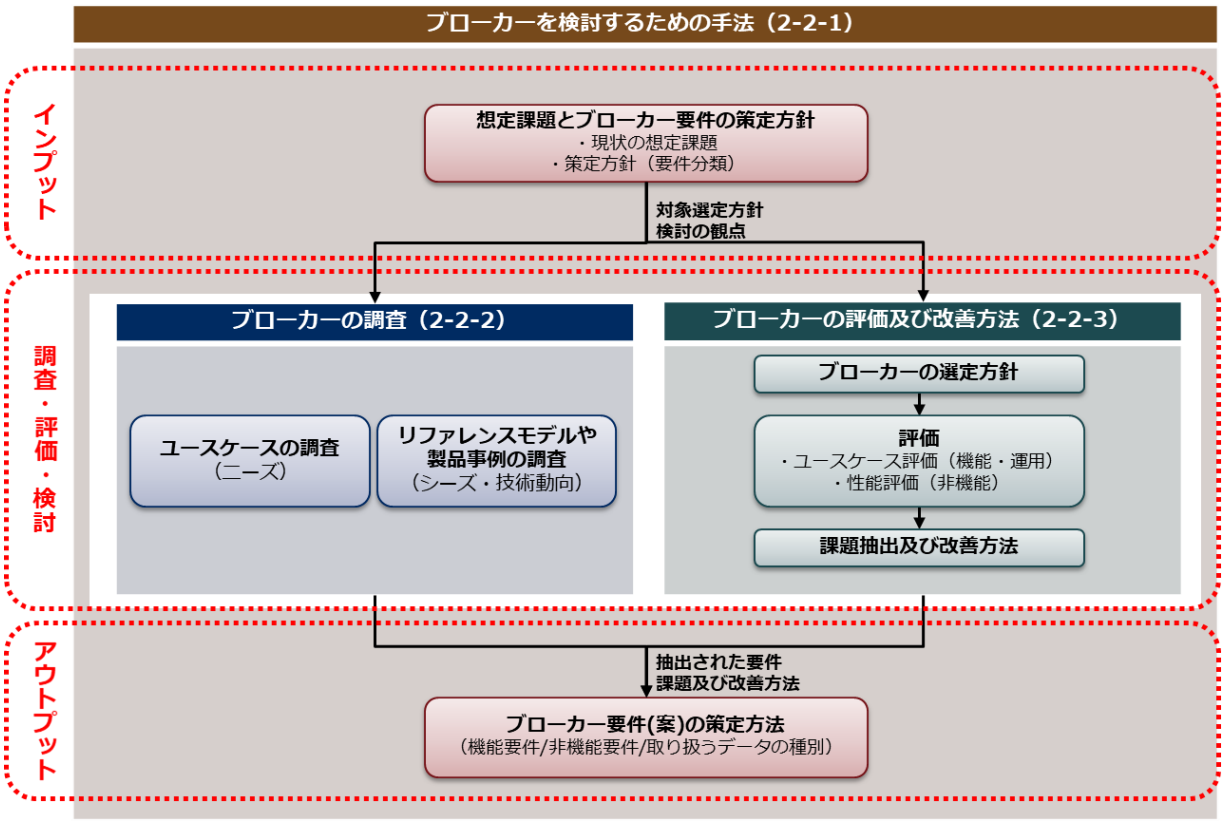


図 2-2-1-1　ブローカー機能の検討プロセス

以下、想定課題とブローカー要件の策定方針、ブローカー要件（案）の策定方法について説明する。なお、ブローカーの調査については 2-2-2、ブローカーの評価評価及び改善方法については 2-2-3 で説明する。

(1)想定課題とブローカー要件の策定方針

(a)現状の想定課題

ブローカー要件（案）を検討する上で考慮すべき現状の想定課題と、これらへの検討方針案を表 2-2-1-1 に示す。

表 2-2-1-1　想定課題及び検討方針案

No.	課題	説明	検討方針案
1	データ分析	データを一元的に蓄積しないデータ分散型のシステムであっても、接続するシステム全体を 1 つのビッグデータとみなし AI 分析等を活用するためにはどういった方法があるか。	各種サービスに対して物理的なデータの所在を隠蔽し、透過的なデータアクセスを実現する方法を検討する。
2	汎用性	データ分散型を前提にしつつ、国際的にも汎用性が高いデータ連携基盤を作るにはどのようなモデルとすればよいか。	先端的サービスのユースケースや各種団体が公開するリファレンスモデル及び製品事例を調査し、ユースケースと技術動向の両面から汎用的かつ共通的なブローカー要件案を検討する。

No.	課題	説明	検討方針案
3	蓄積すべきデータ	データ連携に係る接続の正確性とスピードを重視するため、データ連携基盤に最低限蓄積すべき静的データ（地図情報等）はどういったものがあるか。	先端的サービスのユースケースを基に、使用されるデータの種別、データサイズ、更新頻度等を整理し、ブローカーにて蓄積すべきデータを具体化する。
4	既存システムとの連携	交通、行政等の既存の稼働中のシステムに接続する場合、これらのシステム及びデータ連携基盤そのものに大きな影響や負担を与えることなく正確にデータ連携するには、どういった方法があるか。	既存システムへの影響を最小化し容易に接続可能な方式について、連携先システム・データ連携基盤・ブローカー間の適切な機能分担も含めて検討する。また、利用者と他システムとの仲介という観点で、利用者への通知や、連携先システムへの機能呼び出し等メッセージの仲介をイベントドリブンで実施できるかを検討する。後述の「ユースケース評価」にて既存システムとの接続性を検証し、課題を抽出する。さらに、課題に対する解決策を検討する。
5	コスト	運用性・保守性等を加味した上で、ランニングコストを低減させるためにはどういった方法があるか。	運用保守における属人化/特定ベンダ依存を極力排除した、ブローカーの運用保守作業の効率化手法を検討する。後述の「ユースケース評価」にて検証し、課題を抽出する。さらに、課題に対する解決策を検討する。
6	スケーラビリティ	都市間連携・分野間連携の連携数が将来的に増加した場合においても、リアルタイム性等のブローカー性能を維持可能なように拡張性を保つにはどのような方法があるか。	利用状況に応じたリソース拡張が柔軟に行えるよう、仮想化技術の活用について検討する。後述の「性能評価」にてスケーラビリティを検証し、課題を抽出する。さらに、課題に対する解決策を検討する。
7	パーソナルデータの利活用	地域課題を解決するために、プライバシーリスクを軽減しつつ必要最小限のデータ連携のもとで最大限の価値を発揮するためにはどのような方法があるか。	プライバシーインパクトが懸念されるデータや個人情報等をデータ連携基盤にて扱うための手法を検討する。

(b)策定方針

調査や評価を行うにあたり、ブローカーの要件について、機能や非機能を分類する。前述のとおり、ブローカーとは、「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」によると、様々な主体が提供するデータを集約し、適切な処理を経た上で公開する仕組みである、と定義されている。このように、ブローカーの本質的な役割は、データ提供者から提供されるデータをデータ利用者へ公開すること、つまり、データを仲介することである。「スマートシティリファレンスアーキテクチャ」では、データ仲介という機能ブロックで整理されている。「スマートシティリファレンスアーキテクチャ」におけるデータ仲介には、個別機能として、「データ分散機能」、「データ蓄積機能」、「イベント処理機能」の3つが挙げられている。これを基にブローカーの機能要件を抽出する。ただし、ブローカーには、この3つの機能以外にも必要な機能が抽出される可能性があるため、その他機能も分類に追加する。例えば、想定課題の「既存システムとの連携」を解決するためには、データ連携基盤で扱うことができるようデータ変換を行う必要がある。また、想定課題の「コスト」に対する運用面を考慮すると、利用時に必要な運用機能等が必要となる。

表 2-2-1-2　ブローカーの機能分類

項番	機能分類	説明
1	データ分散	分野や都市に分散するデータに対し、データを仲介（登録・参照・更新・削除）できること。
2	データ蓄積	データ連携基盤内で管理するデータに対し、データを仲介（登録・参照・更新・削除）できること。
3	イベント処理	データ連携基盤が仲介するデータに対し、事前に定義されたシナリオに従いリアルタイムに処理を実施できること。 （条件に合致したデータ発生の通知、接続先システムの機能の呼び出しなど）
4	その他	上記以外。（データ変換、API 仕様、運用機能、ユーザビリティ等）

非機能については、「スマートシティリファレンスアーキテクチャ」にて、IPA/ISEC（独立行政法人情報処理推進機構 セキュリティセンター）が公開する「非機能要求グレード[2]」を活用する、と記載されている。「非機能要求グレード」によると、非機能要求は「可用性」、「性能・拡張性」、「運用・保守性」、「移行性」、「セキュリティ」、「システム環境・エコロジー」の 6 つに分類される。これを基にブローカーの非機能要件を抽出する。

表 2-2-1-3　非機能要求グレード

項番	非機能要求	説明
1	可用性	システムサービスを継続的に利用可能とするための要求
2	性能・拡張性	システムの性能、及び将来のシステム拡張に関する要求
3	運用・保守性	システムの運用と保守のサービスに関する要求
4	移行性	現行システム資産の移行に関する共有
5	セキュリティ	情報システムの安全性の確保に関する要求
6	システム環境・エコロジー	システムの設置環境やエコロジーに関する要求

要件案の策定は、想定課題の解決策を基に要件を抽出する。以下に想定課題に対して、対応する要件分類を整理した（表 2-2-1-4）。機能については、4 つの分類を観点として抽出する。非機能については、6 つの分類のうち、想定課題に対応する「運用・保守性」、「性能・拡張性」、「セキュリティ」の 3 つ分類を観点として抽出する。これを基に、調査及び評価を行う。

[2] https://www.ipa.go.jp/sec/softwareengineering/std/ent03-b.html

表 2-2-1-4　想定課題と要件の対応

想定課題		対応要件分類									
		機能				非機能					
No.	課題	データ分散	データ蓄積	イベント処理	その他	可用性	性能・拡張性	運用・保守性	移行性	セキュリティ	システム環境・エコロジー
1	データ分析	○		○	○						
2	汎用性	○	○	○	○						
3	蓄積すべきデータ	○	○								
4	既存システムとの連携			○	○						
5	コスト				○			○			
6	スケーラビリティ						○				
7	パーソナルデータの利活用									○	

(2)調査ブローカー要件（案）の策定方法

後述の調査及び評価の結果を基に、ブローカーの役割や具備すべき機能要件・非機能要件を整理し、汎用性の高い共通的なブローカー要件（案）を策定する。また、調査及び評価の結果から考察した内容についても、フィードバックし、ブローカー要件（案）へと反映する。

また、ブローカーにて取り扱うデータの種別を整理し、各種データがデータ分散方式又はデータ蓄積方式での管理が適しているかをまとめる。

2-2-2.　ブローカーの調査

上記想定課題の精査と解決策の具体化を進めてブローカーが備えるべき機能を抽出するために、先端的サービスのユースケース、また標準化団体が方針や机上検討結果を公開しているリファレンスモデルや製品事例を調査する。

内閣府及び政策統括官（科学技術・イノベーション担当）が設置する技術検討会と連携し、同検討会での助言を反映しながら調査を実施する。

(1)ユースケースの調査

汎用性の高いブローカー仕様案の検討、蓄積すべきデータの検討等に活用するために、表 2-2-1-1 の想定課題観点から、先端的サービスのユースケースを調査する。具体的には、国外のスマートシティ/スーパーシティの先端的なユースケース事例を 5 事例選定し、詳細調査を行う。以下、調査観点を表 2-2-2-1 に示す。

表 2-2-2-1　ユースケースの調査における調査観点

No.	課題	調査観点	説明
1	データ分析	(ｱ)地域課題	どのような地域課題を解決するか
		(ｲ)ユースケース概要	どのようなユースケース概要やシステム概要か
		(ｳ)先進技術	どのような先進技術を使用するか
2	汎用性	(ｴ)汎用性	汎用的な仕様であるか
3	蓄積すべきデータ	(ｵ)連携データ	どの分野の誰の何のデータを連携するか
		(ｶ)データ属性	どのようなデータ種別、サイズ、更新頻度が想定されるか
		(ｷ)サービス連携要件	サービス連携時、どの程度の正確性・リアルタイム性・ユーザ数が想定されるか
4	既存システムとの連携	(ｸ)既存システムとの連携	既存システムと連携しているか
5	コスト	調査対象外（2-2-2(2)リファレンスモデルや製品事例の調査 にて検討する）	
6	スケーラビリティ	調査対象外（2-2-3 ブローカーの評価及び改善方法にて検討する）	
7	パーソナルデータの利活用	(ｹ)パーソナルデータを取り扱う事例	パーソナルデータを取り扱う事例はあるか

(2)リファレンスモデルや製品事例の調査

ブローカーのあるべきアーキテクチャ・技術仕様を検討するためのインプットとするため、各種団体が公開するリファレンスモデルや製品事例について国内外の取り組みを調査する。

リファレンスモデルの調査においては、データ連携基盤・ブローカーに関して、国内外の業界団体・標準化団体・政府等が公開している標準仕様やホワイトペーパー等を調査した。

製品事例の調査においては、データ連携基盤・ブローカーに関して、国内外の製品事例・技術事例を調査する。

いずれの調査も、調査対象は、都市間連携及び分野間連携が可能であること、北米・欧州・アジアのそれぞれの地域における代表的事例であること、等の観点を総合的に評価した上で選定し、両者合わせて 5 事例を調査する。

上記調査対象について、上記想定課題の観点から調査事例の特徴を整理・分析する。調査観点を表 2-2-2-2 に示す。

表 2-2-2-2　リファレンスモデルや製品事例の調査における調査観点

No.	課題	調査観点	説明
1	データ分析	(ｱ)概要	製品の概要
		(ｲ)複数データソースへの透過的なアクセス方式	どのように複数のデータソースへのアクセスをするのか
2	汎用性	(ｳ)汎用性	認証 API、データマネジメント API、データモデル、標準規格への適合性
3	蓄積すべきデータ	(ｴ)取り扱うデータ	ブローカーに蓄積するデータ 蓄積目的及び蓄積方式
4	既存システムとの連携	(ｵ)既存システムとの連携	既存システムとの連携方式 データ変換方式
5	コスト	(ｶ)サポート体制	サポート体制・開発者コミュニティ体制等運用機能
6	スケーラビリティ	調査対象外（2-2-3 ブローカーの評価及び改善方法にて検討する）	
7	パーソナルデータの利活用	(ｷ)プライバシー対策・セキュリティ対策	ブローカーにおけるプライバシー対策・セキュリティ対策はどうなっているか

2-2-3.　ブローカーの評価及び改善方法

ブローカーの選定は、既存の製品/OSS[3]を活用する。これにより、既存のアセット・知見を有効活用して効率良く評価を実施する。

ブローカーの選定方針について次節に記載する。

(1)選定方針

選定方針は主に下記 3 点とし、選定数は 3 とする。

(a)異なるブローカー技術が用いられていること

ブローカーの基本機能であるデータ仲介として異なる技術が採用されているブローカーを選定することで、それぞれの技術の特徴、ユースケースへの適性が検証できることが期待できる。

(b)ブローカーとして想定される機能を具備していること

表 2-2-1-2 で示した機能を具備していること。さらに具体的に絞り込むために下記を条件とした。

- データ仲介機能のデータ分散方式を実現可能であること
  （スマートシティリファレンスアーキテクチャ（7.1.1.2 データ流通（ながれる））で定義）
- REST API 形式が利用可能であること
  （スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書（3.4 データモデル及び API に関する情報の公開方法）で定義）
- 様々なデータ種別を取り扱えるために柔軟なデータ表現が可能なこと
  （スマートシティリファレンスアーキテクチャ（7.1.1.2 データ流通（ながれる））で例示）

(c)運用性が高いこと

[3] OSS（Open Source Software）：利用者の目的を問わずソースコードを使用、調査、再利用、修正、拡張、再配布が可能なソフトウェアの総称。

本調査事業の成果を次年度以降に多様な主体（データ連携基盤事業者）が活用できることを想定し、運用性（導入/保守のしやすさ、又はそれらを補助する機能を具備）を具備していること。

(2)評価

本項では、選定したブローカーに対する評価内容と、評価結果からの課題抽出及び改善方法の内容について記載する。

今回の評価内容としては、スーパーシティ選定エリアでの実装を想定した「ユースケース評価」と、データ仲介のパターン別の「性能評価」を行い、ユースケース（機能・運用）と性能の両面から課題を抽出する。また、表 2-2-1-1 で示した現状の想定課題を基に今回の評価観点を表 2-2-3-1 にまとめる。

表 2-2-3-1　想定課題に対する評価観点

No.	課題	課題概要	評価観点
1	データ分析	データを一元的に蓄積しないデータ分散型のシステムであっても、接続するシステム全体を1つのビッグデータとみなしAI分析等を活用するためにはどういった方法があるか。	• データ統合管理（所在隠蔽）に必要な機能の抽出/評価 • 物理的にデータが分散されている状態で共通のインターフェースでアクセスが可能か • データアクセスに必要なデータの参照、検索機能の抽出/評価 ⇒ユースケース評価にて実施
2	汎用性	データ分散型を前提にしつつ、国際的にも汎用性が高いデータ連携基盤を作るにはどのようなモデルとすればよいか。	• 汎用的かつ共通的な仕様（データモデル、API 等）の抽出/評価 • ⇒上記観点で評価作業から抽出することは困難と考えるため、4-1 ブローカー調査内での実施とし、評価対象外とする
3	蓄積すべきデータ	データ連携に係る接続の正確性とスピードを重視するため、データ連携基盤に最低限蓄積すべき静的データ(地図情報等)はどういったものがあるか。	• データ連携基盤内で蓄積すべきと判断されるデータを検討することを目的とし、ブローカーで取り扱い可能なデータを抽出する（データの種類、形式、更新頻度等を考慮） ⇒ユースケース評価にて実施
4	既存システムとの連携	交通、行政等の既存の稼働中のシステムに接続する場合、これらのシステム及びデータ連携基盤そのものに大きな影響や負担を与えることなく正確にデータ連携するには、どういった方法があるか。	• データ連携管理に必要な機能の抽出/評価 • 想定される既存システムとの連携方法抽出/評価（既存システムへの低負荷） • データ連携に係る接続の正確性に関わる機能の抽出/評価 ⇒ユースケース評価にて実施
5	コスト	運用性･保守性等を加味した上で、ランニングコストを低減させるためにはどういった方法があるか。	• 想定される運用作業に必要な機能の抽出/評価 • 運用・保守に必要な公開情報、サポート情報の抽出/評価 ⇒ユースケース評価にて実施
6	スケーラビリティ	都市間連携・分野間連携の連携数が将来的に増加した場合においても、リアルタイム性等のブローカー性能を維持可能なように拡張性を保つにはどのような方法があるか。	• 想定される仲介（連携）パターンを整理し、パターンごとの傾向を抽出した上でどのように拡張可能かを評価 ⇒性能評価にて実施
7	パーソナルデータの利活用	地域課題を解決するために、プライバシーリスクを軽減しつつ必要最小限のデータ連携のもとで最大限の価値を発揮するためにはどのような方法があるか。	• パーソナルデータを扱う上でのセキュリティ機能を抽出/評価 ⇒ユースケース評価にて実施

次項以降で、ユースケース評価、性能評価、課題抽出及び改善方法について説明する。

(a)ユースケース評価

ユースケース評価では、スーパーシティにおける先端的サービスのユースケースで想定される連携

サービス/システム及びデータの洗い出しを行い、その上で、表 2-2-3-1 で記載した評価観点から評価項目を抽出した。評価は複数の異なるブローカーに対して実施することで各課題に対して異なるアプローチでの解決策を抽出することが期待できる。

表 2-2-1-4 の要件分類に対して、表 2-2-3-1 の評価観点で抽出した評価内容を表 2-2-3-2 に示す。抽出された評価内容について各ブローカー製品での評価を実施し、ブローカーとして具備すべき機能についての検討を行う。

表 2-2-3-2　ユースケース評価内容

要件分類		評価内容
機能	データ分散	• データ利用者はデータの所在を意識することなく透過的にアクセスすることが可能であること
		• 分散されたデータ提供者に対して、ブローカーを介することで共通のインターフェースでアクセス可能であること
	データ蓄積	• 様々な種類、形式のデータについて取り扱い可能であること （評価、調査結果を基にブローカーで蓄積すべきデータの検討を行う）
	イベント処理	• 既存システムでデータ更新があった場合にブローカーを通して利用者側に通知が送信されること（イベント通知）
		• 既存システムが提供するサービスに対して既存システム側に負担をかけず仲介できること（サービス呼び出し）
	その他	• データ提供者のデータに対して様々な方法でのデータ参照(検索)が可能であること
		• ブローカーとデータ提供者の間でデータの変換処理が可能であること
		• データ連携に係る接続の正確性を担保する機能を具備していること (再送処理、データアクセスログ等)
非機能	運用・保守性	• 蓄積されたデータの管理や分散されたデータとの仲介に必要な管理機能を具備していること
		• ブローカーについて公開ドキュメントやサポート体制などが整っていること
	性能・拡張性	• 性能向上に係る拡張性を具備していること
		• 可搬性を具備していること(実行環境に依存しないこと)
	セキュリティ	• パーソナルデータを扱うためのセキュリティ機能を具備していること （オプトイン機能、データの匿名化等）

(b)性能評価

データ連携基盤では、リアルタイム性やプライバシーインパクトを考慮してデータ分散方式が推奨されている。しかしながら、データ分散方式においては、データ蓄積方式と比較し、データの分散化に伴う性能オーバーヘッドが懸念される。複数のブローカー方式について性能評価を実施し、実際にどの程度のデータ仲介が実現可能かを評価する。

性能評価では想定課題の「スケーラビリティ」について実施する。性能評価については下記の 2 軸の観点で評価を実施する。この 2 軸の評価を行うことで、性能向上に係る拡張性と、連携する分野数や都市数に応じてどのようなブローカー構成にすればよいかの検討が可能となる。

- ブローカーのスケーラビリティ評価

ブローカーがスケールアップ/アウトした際にどの程度まで性能が向上するかを検証する。

スケールアップ評価：
ブローカーサーバ 1 台に対してベースとなるサーバスペックから CPU とメモリを変動させて性能測定を行う。

スケールアウト評価：
ブローカーサーバスペックを固定し、ブローカーサーバの構成台数を変動させて性能測定を行う。

- データ仲介パターンによる性能評価

以下、図 2-2-3-1、及び表 2-2-3-3 に示すデータ仲介パターンについての性能評価を実施する。

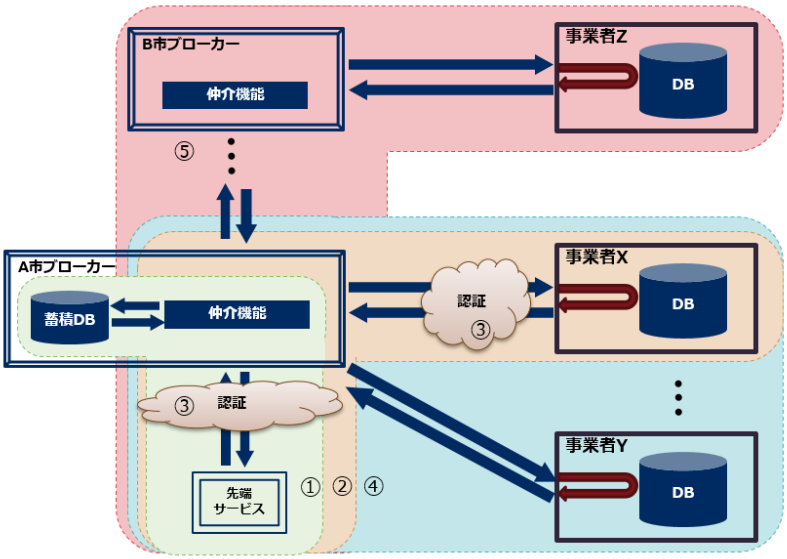


図 2-2-3-1　データ仲介パターンイメージ

表 2-2-3-3　データ仲介パターンにおける評価ポイント

項番	データ仲介パターン	評価ポイント
①	データ蓄積方式 (ブローカー内の蓄積 DB に蓄積)	• 蓄積方式の性能測定
②	データ分散方式 (単一ブローカーによる単一分野連携)	• 分散方式の最小構成の性能測定(限界性能)

③	データ分散方式 （単一ブローカーによる単一分野連携） ※認証処理あり	• 認証処理のオーバーヘッドによるブローカーへの • 影響検証
④	データ分散方式 （単一ブローカーによる複数分野連携）	• 仲介対象の分野数増によるブローカーへの影響検証 ※処理負荷の全体メッセージリクエスト数を固定し、分野数を 1 分野、2 分野、5 分野、10 分野と変動させてブローカーへの影響を検証する
⑤	データ分散方式 （複数ブローカーによる単一分野連携）	• 仲介のホップ数増によるブローカーへの影響検証 ※処理負荷の全体メッセージリクエスト数を固定し、ブローカーの経由数を 1 段、2 段、5 段と変動させてブローカーへの影響を検証する

(3)課題抽出及び改善方法

ユースケース評価、性能評価によって抽出された各課題に対し、改善方法を検討する。改善方法は、ブローカー機能の改善と、ブローカー以外の他機能又は運用による対応の両観点から検討する。

抽出された課題に対して複数観点から改善方法を検討して整理する。課題に対する対処方法は、各スーパーシティエリアの方針により異なることが想定される。多角的な改善方法を整理することで、各エリアの方針に合った改善方法を採用することができる。

2-3. （イ）-① API 共通ルール/標準仕様

2-3-1. 調査方針

(1)調査の目的

スーパーシティにおいてデータ連携基盤や先端的サービスが整備される中で、多数の API が開発・提供されることが想定される。これらの API はオープン API として、多数のサービスから利用され、相互に接続・連携できることが求められる。これまでに検討・整備されている「戦略的イノベーション創造プログラム（SIP）第 2 期/ビッグデータ・AI を活用したサイバー空間基盤技術のアーキテクチャ構築ならびに実証研究事業」で記載されたスマートシティリファレンスアーキテクチャや「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」にはオープン API でデータ連携基盤とサービスが連携することが記述されているが、API の共通ルールや標準仕様までは規定されていない。そのため、全国自治体のスーパーシティ/スマートシティの取り組みにおいては、API がエリアの独自ルール・仕様で設計/開発・公開・運用され、エリア内外での相互運用性・互換性が向上しないリスクがあると想定される。

そこで、本報告書では API の共通ルールとして、API 運用者向けの設計/開発・公開・運用プロセスと API 利用者向けの利用規約（テンプレート）、及び API 設計・開発者向けの標準仕様案を策定し、エリア内外で一定の統制を取ることを目標とする。これにより、図 2-3-1-1 に示すとおり、API の相互運用性や互換性を向上し、先端的サービスや API の再利用や共用を目指す。

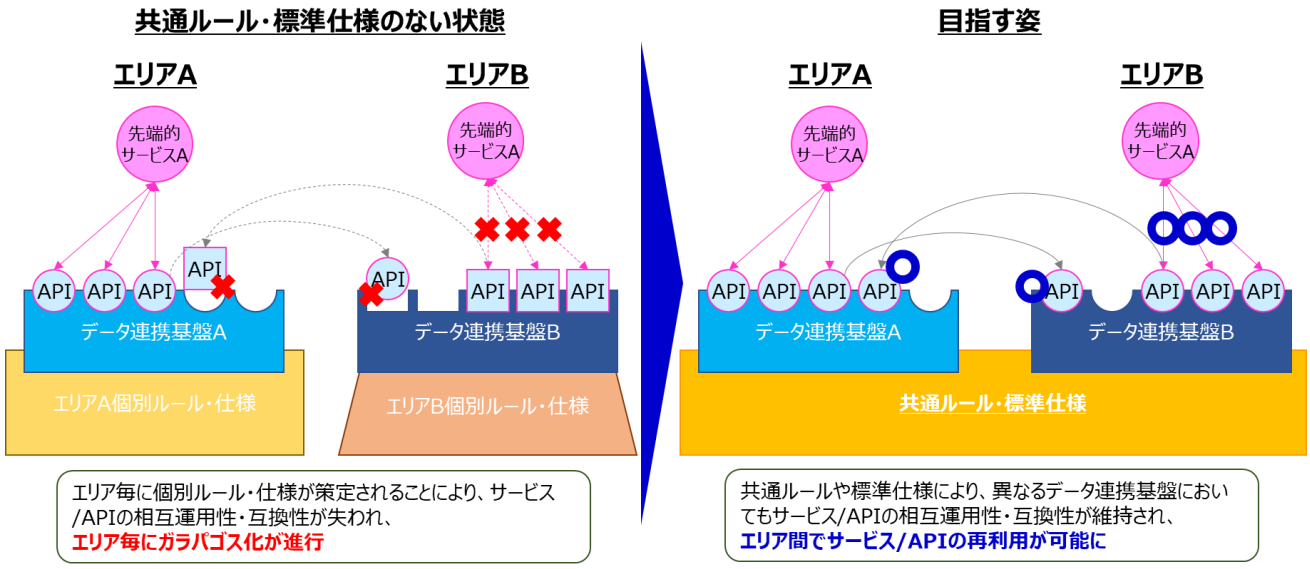


図 2-3-1-1　API の共通ルールや標準仕様の有無による相違イメージ

(2)調査の対象

「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」の「追補：Society 5.0 定義レイヤとの対応」によると、Society 5.0 におけるデータ連携層は、スマートシティリファレンスアーキテクチャでいうデータ連携層と機能層に対応する。本報告書では、これらデータ連携層と機能層を対象とし、認証及びデータマネジメントに関する API を検討範囲として優先的に定めることとした。

(3)調査の方法

API の共通ルールや標準仕様検討のアプローチとして、図 2-3-1-2 の流れで調査を実施した。

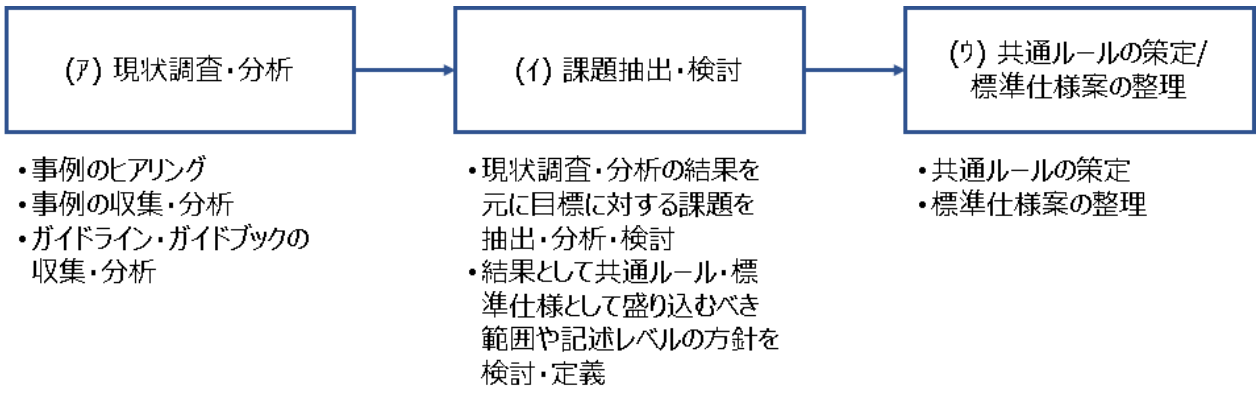


図 2-3-1-2　検討アプローチ

2-3-2.　API 設計/開発・公開・運用プロセス

(1)現状調査・分析

API を整備するプロセスを、設計/開発・公開・運用に分解し、それぞれ調査を進めた。具体的には、API に関する文献として内閣官房情報通信技術(IT)総合戦略室が発行する API 導入実践ガイドブックや API テクニカルガイドブックを参照した。また、API の設計/開発・公開・運用実績がある、表 2-3-2-1 に示す団体・民間事業者へ対して、現状のプロセスやプロセスごとの取り組み、指針、基準等に関するヒアリングを実施した。机上での調査結果とヒアリング結果を基に、API に関する現状の分析を行った。

表 2-3-2-1　ヒアリング先

団体/企業名
日本気象協会
G 空間情報センター
Code for Japan

(2)課題抽出・検討

ヒアリング結果と、「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」の「3.2 API の開放性設計指針」を基に、本調査業務の目標や目指す姿に照らして課題を抽出・検討し、プロセスとして盛り込む範囲を明確化した。

(3)共通ルール策定

設計/開発、公開、運用ごとに必要なプロセスフローを定義し、各プロセスの望ましい在り方を定義した。

2-3-3.　API 利用規約テンプレート

(1)現状調査・分析

政府標準利用規約（第 2.0 版）と国・自治体及び民間事業者にて提供されている API の利用規約を収集し、範囲や記述レベルを確認した。収集対象の利用規約は表 2-3-3-1 のとおり。また、スマートシティリファレンスアーキテクチャホワイトペーパーの第 4 章「スマートシティルール」及びスーパーシテ

ィの構想イメージとして公開されているA市～D市の事例を参考に、データ連携基盤が担う先端的区域データ活用事業活動のうち、APIを扱うオンラインサービスに関連する取り組みと関連法令、想定される特例措置例についても調査、分析を行った。

表 2-3-3-1　収集対象の利用規約

利用規約名	発行組織
AWDS(AIZUWAKAMATSU Developer’s Site) 利用規約	会津地域スマートシティ推進協議会
Chatwork API 利用規約	Chatwork株式会社
freee API利用規約	freee株式会社
Google APIs Terms of Service	Google法人
J-STAGE WebAPI 利用規約	国立研究開発法人科学技術振興機構
LINE公式アカウントAPI利用規約	LINE株式会社
Rakuten Rapid API 利用規約	楽天株式会社(R Software Inc)
RESUS-API利用規約	内閣府 地方創生推進室 経済産業省 地域経済産業調査室
RUNDA API利用規約	アウル株式会社
Sansan API利用規約	Sansan株式会社
Slack API サービス利用規約	Slack法人
ヌーラボAPI利用規約	株式会社ヌーラボ
マイナポータルAPI利用規約	内閣府大臣官房番号制度担当室
メディカル革命 byGMO API利用規約	GMO医療予約技術研究所株式会社
官民連携データプラットフォーム 規約（素案）	東京都戦略政策情報推進本部戦略事業部総務課
銀行法に基づくAPI利用規約の条文例	一般社団法人全国銀行協会
東京公共交通オープンデータチャレンジ API 利用許諾規約	公共交通オープンデータ協議会

(2)課題抽出・検討

収集した利用規約を本調査業務の目標や目指す姿に照らし、課題を抽出、検討し、利用規約テンプレートとして盛り込む範囲や記述レベルを明確化した。

(3)共通ルール策定

収集した事例を参考に明確化した範囲や記述レベルで利用規約テンプレートを策定した。

2-3-4.　API標準仕様案

(1)現状調査・分析

「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書（3.4 データモデル及びAPIに関する情報の公開方法)」で定義された設計様式（REST）及びデータ形式（JSON）を利用するAPIを主な調査対象とし、APIの認証方式としては、スマートシティリファレンスアーキテクチャで定義されたOAuth2.0やOIDC（OpenID Connect）の仕様をベースとして調査・分析を行った。

内閣官房情報通信技術(IT)総合戦略室が発行するAPI導入実践ガイドブックやAPIテクニカルガイドブックを参照した。また、インターネット上に広く公開されているAPIの標準仕様についても収集・分析を行った。収集対象のAPI標準仕様は表 2-3-4-1のとおり。

表 2-3-4-1　収集対象の標準仕様

標準仕様名	発行組織
18F API Standards	アメリカ政府
API Design Guide(GOV.AU)	オーストラリア政府
API technical and data standards	イギリス政府
Application programming interfaces (APIs)(GOV.UK)	イギリス政府
HMCTS Reform Programme RESTful API Standards	イギリス政府
Service Standard	イギリス政府
SP 800-63-3 Digital Identity Guidelines	アメリカ政府
White House Web API Standards	アメリカ政府
Working with open standards	イギリス政府

(2)課題抽出・検討

先端的サービスが利用するためのデータ連携基盤のオープン API を想定し、課題を抽出、検討し、標準仕様案として盛り込む内容を明確化した。

(3)標準仕様案の整理

本調査業務を通じて調査された（ア）データ仲介（ブローカー）機能、（イ）API の共通ルール、（ウ）データモデルの方針で検討されるデータ連携仕様案の検討結果も踏まえ、内閣官房情報通信技術(IT)総合戦略室が発行する API 導入実践ガイドブックや API テクニカルガイドブックに従い、標準仕様案を整理した。

2-4.（イ）-② API カタログ/開発者ポータル

2-4-1.　API カタログ、及び開発者ポータルの仕様案策定におけるポイント

本項では 2-3 で記述される API の共通ルールや標準仕様等について、中央主導で各エリアに対して情報を公開するための API カタログ及び各エリアで構築される予定の開発者ポータルに関して、必要と想定される機能や仕様を取りまとめるための観点とプロセスについて記載する。

API カタログは、API の共通ルールや標準仕様等について中央主導で各エリアに対し情報を公開することが主な役割である。一方、各エリアで独自の API を公開したり、国が定めた API 共通ルールを参照して作成された API を各エリアで公開する場合、各エリアにおいても個別に API を公開する機能が必要となる。そのため、本調査業務における API カタログの仕様検討においては、中央から各エリアの API を提供する観点と、各エリアにおいて独自で API の詳細情報を公開する観点それぞれについて実施する。

一方、開発者ポータルは、データ連携基盤を実装する各エリア単位で実装される。各エリアにおいて開発者ポータルを実装するにあたり、標準的に具備すべき機能の仕様を画一化し、各エリアに共有することで、重複投資の削減やスーパーシティにおいて重要な要素である相互運用性を担保することができると考えられる。

ここで、本項における検討範囲を図 2-4-1-1 に示す。

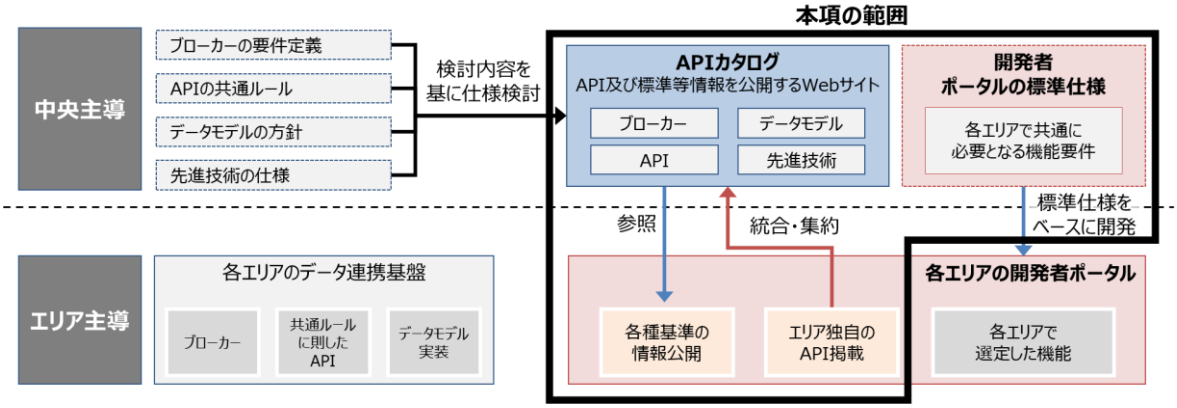


図 2-4-1-1　API カタログと開発者ポータルの本項における検討範囲

また、本項における検討のプロセスを図 2-4-1-2 に示す。なお、検討プロセスについての詳細は 2-4-4 にて記載する。

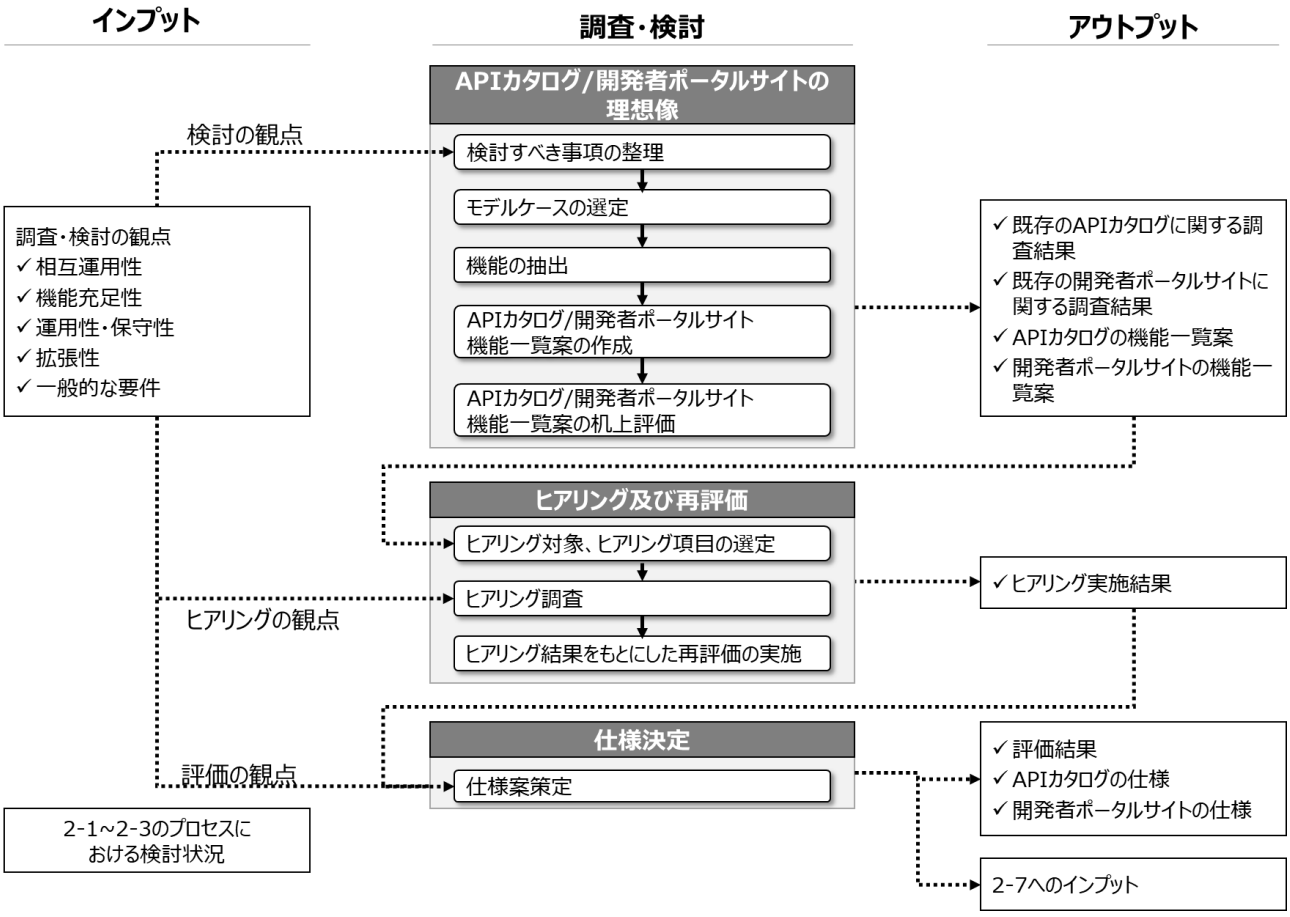


図 2-4-1-2　API カタログ/開発者ポータルにおける仕様案の検討プロセス

2-4-2.　API カタログの仕様案策定

本調査業務において検討される API の共通ルールや標準データモデル、ブローカーの仕様等を公開する場として、幅広い利用者に利活用頂くことが想定される API カタログの仕様案について、以下の観点

を踏まえ、利用者・運用者・開発事業者それぞれの立場から検討する。

(1)国と各エリアとの相互運用性の確保（相互運用性）

スーパーシティ構想において重要な観点である相互運用性を実現するために、2 つの側面から機能検討が必要となる。1 つ目が API カタログにおいては中央から各エリアへ情報を展開すること、そして 2 つ目が各エリアから中央へ情報を統合・収集することである。

(2)利用者のニーズを考慮した機能の充足性（機能充足性）

本調査業務における API カタログ独自の役割として、データ連携基盤の仕様や開発者ポータルの仕様、及び各エリアでの実装状況等の情報を提供することが想定される。このことから、自治体や各分野のサービス事業者等、利用者が多様になることが想定されるため、利用主体ごとの特色を考慮したニーズに幅広く対応し、利便性を高められるような観点での仕様（コンテンツ・デザイン案等含む）検討が必要となる。

(3)継続利用のための運用事業者のニーズを考慮した運用性・保守性（運用性・保守性）

運用・保守のしやすさの観点から API カタログを継続的に運用でき、広く普及させるための機能の検討が必要である。また、各エリアでの実装を想定した場合、規模の大小によらず導入・運用できるような仕組みを検討する必要がある。

(4)運営主体・運営方法の可用性（可用性）

継続的に運営を続けるにあたって必要となる運営主体の在り方や運営パターンの考慮が必要である。本観点については、2-7 における検討項目として記載する。

(5)将来の運用を想定した機能拡張の容易さ（拡張性）

特に技術革新が速く、多岐にわたる分野が関連するスーパーシティ構想においては、将来的に発生し得る変更に耐えられる設計である必要がある。また、機能追加や変更が必要となった際も、コア部分に手を入れずに拡張できるような変更の容易さを備えていることが必要となる。

(6)API カタログの一般的な要件

スーパーシティ構想における API カタログとして特に検討が必要となる要件以外にも、一般的に実装されている機能の考慮、デザインや運用方法等を参考にし、利用者目線で違和感のない仕様に落とし込む必要がある。

2-4-3. 開発者ポータルの仕様案策定

2-4-1 に記載の前提を踏まえ、本調査業務における開発者ポータルの仕様案策定において検討が必要と想定される観点を以下に示す。

(1)各エリア間の相互運用性の確保（相互運用性）

各エリアの開発者ポータル間で相互にデータを連携することで、全国的なナレッジの共有やエリアを跨ったシームレスな利用を実現できると考えられる。そこで、連携させるデータについてエリア間で共通の理解を醸成するために、開発者ポータルの仕様は中央で整備することが望ましい。一方で、標準仕様の範囲が過剰に広い場合、各エリアの独自性を反映することが難しくなるため、標準仕様の範囲についても併せて検討が必要となる。

(2)利用者のニーズを考慮した機能の充足性（機能充足性）

本調査業務における開発者ポータルの利用者である先端的サービス事業者は、スーパーシティにおいては複数の分野に跨ってサービスを展開していることが想定される。各利用主体の特徴を考慮したニーズに対応できる機能（コンテンツとその提供形態）の洗い出しが重要である。

(3)継続利用のための運用性・保守性（運用性・保守性）

導入事業の主体となる自治体は、団体によって規模も解決すべき社会課題の分野範囲・規模も異なる。そこで、各エリアの規模に応じて、開発者ポータルの継続的な保守・運用の実現や導入の障壁を下げ、広く利活用されるために具備すべき機能についても検討する必要がある。

(4)システム障害に強く、安心して利活用できる仕組みで継続的に利用・運用することができる実現性（可用性）

スーパーシティにおける開発者ポータルは、様々な開発者や事業者がアクセスするサイトであり、先端的サービスの開発において重要な情報提供の場であるため、高い可用性が確保される必要がある。これらの運用に関する観点については、2-7 における検討項目として記載する。

(5)将来的な利活用を考慮した機能拡張の容易さ（拡張性）

技術革新が短期間で行われるスーパーシティの特徴に鑑み、開発者ポータルもそれらに柔軟に対応可能な構成とする必要がある。システム連携部分の標準化や、追加部分を疎結合な仕様とすること等を通じて、拡張性の向上や障害発生時の影響範囲の限定、及び各エリアにおける独自性を反映させた機能実装を支援する必要がある。

(6)開発者ポータルの一般的な要件

上記で挙げた事項以外にも、スーパーシティの開発者ポータルにおいて共通的に検討すべき事項として RASIS の考慮や利便性、デザイン等が挙げられる。

2-4-4. 検討プロセス

2-4-2 及び 2-4-3 に記載の検討観点を踏まえ、API カタログ・開発者ポータルの仕様案策定における検討プロセスを以下に示す。

(1)検討すべき事項の整理

検討すべき観点について改めて整理する。

(2)モデルケースの選定

本事業における API カタログ・開発者ポータルに類する既存事例について調査を実施し、特徴的・典型的な事例の課題とそれに対応した事例をモデルケースとして選定する。モデルケースの選定に際しては、自治体・中央省庁への事業参画における知見を活用するとともに、民間事業者や標準化団体、金融分野の先行事例等、複数の事例をインプットとしながら検討を実施する。

(3)機能要件の抽出

選定したモデルケースより得られた機能要件及び、机上調査（類似事例のノウハウ適用、文献・Web からの情報収集）により、API カタログの仕様案策定に向けて必要となるニーズ検討を踏まえて、機能要件

を見直す。なお、机上調査実施時に参照する文献は、スマートシティリファレンスアーキテクチャ[4]やスーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書[5]等を予定している。

(4)API カタログ・開発者ポータルの機能一覧案の作成

API カタログ・開発者ポータルに必要な機能要件に関する機能一覧案を作成する。

(5)機能一覧案の机上評価

API カタログ・開発者ポータルの想定アクターを整理し、それを基に利用シーン及び画面イメージを検討する。機能一覧案について、各アクターの利用シーンや運用の流れを想定した机上評価を行うことで、実際に発生し得る課題の洗い出しを行う。洗い出した課題については仕様案にフィードバックを行う。

(6)ヒアリング及び再評価

作成した機能一覧案に関して、既存の API 公開ウェブサイトを構築・運営している事業者や、構築主体の例として想定されるシビックテック推進団体に対し、利用者ニーズや構築・運用する上で課題となり得る事項についてヒアリングを行う。ヒアリング実施後、運用フロー及び画面イメージを用いて、ヒアリングに即した機能が満たされているか、利用者・運営者にとっての利便性が具体的にイメージできる仕様となっているか等、再度評価する。

(7)仕様案の整理

再評価した機能一覧案を基に、(1)～(6)で検討した API カタログ及び開発者ポータルの仕様（画面イメージ、機能、利用シーン)、ヒアリング結果、データ仲介（ブローカー）機能、API の共通ルール、及びデータモデルの方針で得られた見解を踏まえ仕様案としてまとめる。

[4] スマートシティリファレンスアーキテクチャ

[5] スーパーシティ／スマートシティの相互運用性の確保等に関する検討会　最終報告書

2-5. （ウ）データモデルの方針

2-5-1. 想定課題

データ連携基盤は、スーパーシティに選定された各エリアにおいてそれぞれ整備するものであるが、スーパーシティに選定された複数の都市間連携を図ることに加えて、スーパーシティのエリア外でも汎用的に使用されていくためには、各エリアが別々に設計をするのではなく、共通的な仕様等を決定していく必要がある。データ層においては、様々な事業者が持つデータそれぞれが持つ意味を明確化し、異なる主体間で互換性の高いモデルを構築することが重要である。海外では、EU の SynchroniCity 等、都市間での相互運用性を意識した取り組みが進んでいる。一方、国内でもスマートシティの取り組みは進められているが、データモデルにまで踏み込んで相互運用性の確保を考慮した取り組みの事例は数少ないと認識しており、ここが課題であると考える。

既にオープンな標準が存在する場合は優先的に活用するが、API を構築してデータを配布するにあたり、やむなく新たなデータモデルを策定する場合は、広くそのモデルを普及させるためにデータモデルを公開すべきである。新たに策定したデータモデルは、利用者によるフィードバック等、モデルを見直すプロセスが重要であると考えており、データモデル公開における推奨プロセスの定義も重要な課題であると考えている。

2-5-2. 調査の前提条件

本調査業務では、スーパーシティ間及びスーパーシティ外でも汎用的に使用されていくデータモデルの仕様、規格等の整理を目的に、データモデル標準化についての方針を提示する。そのため、検討対象とするデータモデルには、「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」（本項では、以下「最終報告書」と記す）より、以下の要件を想定する。

(a)対象とする領域
(ｱ)最小限のデータで最大の価値を生むこと
(ｲ)既存のデータを多目的に活用

(b)データモデルとしての条件
(ｱ)アクセスログ等により利用状況が測定できること
(ｲ)実在データの品質検証やツール等を用いたクレンジングに活用できること
(ｳ)既存のデータモデルが存在する場合には、可能な限りそのデータモデルを使用すること

(c)ユーザからのフィードバックと改善のプロセス
(ｱ)データモデル、データの改善要望への対応窓口と、反映していく責任
(ｲ)普及啓発

最終報告書によると、「スマートシティにおける連携レイヤー」のうち、戦略・政策層～ビジネス層は「施策及びプロジェクトとしての評価」、機能～データ連携層は「システムとしての評価」と、評価レイヤーは分離すると記載されている。しかしながら、施策及びプロジェクトとしての評価軸を持たずに定義されたデータモデルやデータ項目定義は用途やビジネス価値が不明瞭で、自治体等が公開しても使われず、使われない結果更新が滞り、さらに利用価値が下がるといった負の循環を招く危険がある[6]。こうした現状を踏まえ、スーパーシティデータ連携基盤としては、

[6] 例えば、オープンデータの取り組みにおける課題や問題点のトップは「オープンデータの効果・メリット・ニーズが不明確」である。また、政府 CIO ポータル「推奨データセット」14 種のうち、AED 設置

上位評価レイヤーの目的に合致するデータモデルを対象に調査を行うべきと考える。

2-5-3. 調査の進め方

2-5-1 及び 2-5-2 を踏まえ、以下のとおり調査手順を記載する。

なお、本調査業務の企画競争説明書にも記載があるとおり、内閣府及び政策統括官が設置する技術検討会（以下、内閣府技術検討会）等と連携し、同検討会の提言を基に、進め方については柔軟に対応する想定である。

(1)既存データモデルの調査

(a)調査項目の検討

データモデルの記述レベルには様々なものがある。実装を可能にするサンプルレコードまで記述するモデル、クラス図による表現などもあるが、現時点での整理では、データ項目名とその項目説明というシンプルな項目を対象とする。複雑な設計によってデータ項目がばらばらになることを避け、調査結果は今後更新される前提で、まずはスーパーシティデータ連携基盤の基本となるデータモデルが共有されるための調査である。

(b)調査対象の選定

国内外の主要スマートシティで共通的に使われているデータモデル、国内でニーズの高いデータモデルを対象に調査対象を選定する。具体的には以下(ｱ)(ｲ)を調査する。

(ｱ)有識者から提示されるデータモデル

内閣府技術検討会で取りまとめる「スーパーシティのデータモデルについて」が対象とするデータモデル。

(ｲ)ユースケースシナリオを起点としたデータモデル

スーパーシティ/スマートシティ上で展開され得るサービスのユースケースを想定し、「戦略的イノベーション創造プログラム（SIP）第 2 期/ビッグデータ・AI を活用したサイバー空間基盤技術/パーソナルデータアーキテクチャ構築」事業の成果であるユースケースシナリオテンプレートを用いてユースケースシナリオを整理する。本ユースケースは「2-5-3(4)ユースケースの検討」にて作成されるものをベースとし、ユースケースシナリオに基づき導かれる、サービスにおいて必要と考えられるデータモデル。

(c)調査

(b)調査対象の選定を対象に、(a)調査項目の検討で検討した項目について調査を実施する。

箇所一覧は自治体の 27.7%が公開しているが、「公衆トイレ一覧」は 6.5%、「消防水利施設一覧」は 6.8%に留まる（いずれも政府 CIO ポータル「地方公共団体へのオープンデータの取り組みに関するアンケート結果」より）。

(2)データモデルの方針検討

(1)既存データモデルの調査の結果を踏まえ、スーパーシティデータ連携基盤に求められるデータモデルの方針を検討し報告する。報告には、最終報告書の記載内容及び DTA がこれまで培ってきた知見を踏まえ、以下の要素を含むことを想定する。

(a)デジタル 3 原則実現を適える原則[7]への対応

1)アクセシビリティ

2)相互運用性

(b)データモデルのライフサイクルを考慮した基本方針

1)設計

2)データ及びデータモデルの公開

3)効果検証

4)改定・拡充を含む運用

(c)持続可能な体制

1)データセットに対する社会合意（ルール・規約）

2)持続的改善を前提とした提供方法、体制、プロセス

(d)推奨データモデル　　分野間で活用可能な既存データモデル

(3)データモデル方針の検証

スマートシティの共通設計図かつ運用マニュアルであるアーキテクチャ「都市 OS」開発の知見を踏まえ、データモデル方針の検討結果について検証を行い、実効性を確認する。

(4)ユースケースの検討

スーパーシティやスマートシティ関係者がデータモデルの設計や活用に取り組みやすいよう、(1)既存データモデルの調査、(2)データモデルの方針検討を踏まえ、スーパーシティ/スマートシティ上で展開されるサービスユースケースをまとめる。

(a)本ステップの目的

本ステップでは(1)～(3)の調査結果等を基に、スーパーシティ/スマートシティ上で展開され得るサービス・ユースケースを想定し、データモデルとして成立するかを検証する。本ステップで検証する範囲としてはスマートシティリファレンスアーキテクチャで示されたデータモデル要素（語彙体系、データ項目、データ構造）の範囲において、ユースケース上必要となる適切なデータ項目（地理空間データ含む）とそのモデリングを対象に、データ連携の実現性検証を机上で実施し、報告書にまとめるものとする。

(ｱ)達成目標

スーパーシティ/スマートシティサービス・ユースケースが成立し得るデータモデル例を報告する。対象分野は内閣府『第 4 回「スーパーシティ」構想の実現に向けた有識者懇談会』における

[7] Accessibility、Interoperability は ISA Base Registry で Once Only の原動力として掲げられている。

『「スーパーシティ」構想の実現に向けた関係府省の取組と連携（案)』にて議論された、スーパーシティ構成領域である「移動」「物流」「支払い」「行政」「健康・医療・介護」「教育」「エネルギー・水」「環境・ゴミ」「インフラ・防災」「防犯・安全」「都市 OS」の中から対象領域を複数選定する。データセットを選定する際に、各領域で共通的に使うものを対象とし、他分野に横展開できること選定基準とする。この中で、ベースレジストリ候補を含む地理空間データの推奨データセットを検討する。対象領域選定後に定めるユースケースを基に検証する内容については、本調査業務体制内での会議体や内閣府及び政策統括官が設置する技術検討会等と連携を経て、得られた助言を基に検証を進めることとする。

(b)プロセス

本ステップは、(1)～(3)の調査結果を基に、次の計画内容を進めるものとする。

(ｱ)対象領域の選定

「移動」「物流」「支払い」「行政」「健康・医療・介護」「教育」「エネルギー・水」「環境・ゴミ」「インフラ・防災」「防犯・安全」「都市 OS」の中から複数選定する。選定基準としては、スーパーシティ/スマートシティサービスを展開する上で、地理・空間データ（ベースレジストリ候補含む）等共通で使われる複数のデータを扱う領域とする。

(ｲ)サービス・ユースケースの選定

社会的背景よりスーパーシティ/スマートシティサービスとして検討すべきもの、また『「スーパーシティ」構想の実現に向けた関係府省の取組と連携（案)』にて検討されたスーパーシティ構成領域ごとに到達すべき姿を考慮した上で、必要となり得るユースケースを検討していく。

(ｳ)データ項目選定

上記ユースケースに必要となるデータ項目を選定していく。また標準規格として整理されているデータについては調査の上最大限活用し、適切な採用を行っていく（例えば「健康・医療・介護」業界であれば多数存在する日本及び国際標準規格の中からデファクトとして普及している規格を考慮し適宜採用する)。特に選定候補の 1 つである地理・空間データ（ベースレジストリ候補含む）においては、現時点で調査対象の標準規格及び団体は以下を想定する。

・標準規格：ISO/JIS、CityGML、LandXML、IFC、GeoJson、他
・標準化団体：OGC(Open Geospatial Consortium)、bSI(building Smart International)

(ｴ)データモデリング

(ｱ)～(ｳ)の内容を基にデータ項目分野ごとに技術者を選出し、検討会を開催し、スマートシティにおけるデータモデリングの在り方・手法を検討していく。

2-6.（エ）先端的サービスとデータ連携基盤の接続検証

2-6-1.　先進技術の調査と選定

(1)先進技術の調査・選定における観点

「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」にも記載されているとおり、スーパーシティの相互運用性を担保するためには、スーパーシティにおける施策及びシステムが、各層において適切に連携することが重要である。特にシステムの観点から、機能層・データ層・データ連携層が相互に連携することが相互運用性を保つために不可欠である。そのため、機能層に位置する先端的サービスを支える先進技術（個人認証、決済、センサー等）として、共通に推奨すべきものとしては、データ層・データ連携層に位置するデータ連携基盤と親和性の高い技術を選定することが重要である。

本プロセスでは、データ連携基盤の構築において、将来的に各エリア共通で必要となり得る先端的サービスを支える先進技術について以下 3 つの観点で調査・選定する。

(a)先端的サービスを支える先進技術としてのニーズが高いか。（ニーズ）

先端的サービスを支える先進技術としてニーズが高いか、また個別サービスに特化した内容ではなく複数のサービス間で共通に利用され得る技術であるか調査・検討する必要がある。具体的には、公募要領にも記載されている個人認証、決済、センサー等の他に、スマートシティリファレンスアーキテクチャにて定義されているパーソナライズ機能、コンテンツ管理機能、地域ポイント管理機能、パーソナルデータの公開範囲を定めるオプトイン機能、及び可視化分析ダッシュボード機能等が挙げられる。

(b)各種サービス分野との相互運用性や相乗効果が期待できるか。（相互運用性）

先端的サービス、又は他の先進技術との親和性が高い技術であるか、また、スーパーシティ構想における先端的サービスは複数のサービス分野（行政、金融、物流等）に跨って提供されることが想定されるため、それらを支える先進技術としても各種サービス分野に相乗効果をもたらすことが期待されるか調査・検討する必要がある。例えば、認証技術の場合、OAuth2.0 と OIDC を併用（OIDC の仕様に基づき OAuth の認可要求を行う）しているか等、準拠していることが望ましい標準規格とどの程度適合している技術であるか確認する必要がある。

(c)社会動向を考慮した将来性があるか。（将来性・拡張性）

新型コロナウイルス感染症の流行を受けたニューノーマルへの対応等、地域が直面し得る社会課題の解決を継続的に対応する先端的サービスに不可欠な先進技術であるか検討する必要がある。例えば、非接触での操作や認証を実現する技術、リモートでのサービス提供の可否等の考慮が重要と想定する。

(2)先端的サービスを支える先進技術の調査・選定の進め方

各エリアへ共通に推奨すべき先進技術の調査・選定プロセスを以下に示す。

(a)先端的サービスを支える先進技術の調査

調査方法は受託事業者が有する先端的サービスの知見の適用、及び文献・Web を通じた机上調査を想定している。既存の先端的サービスについて調査を行い、既存の先端的サービスから共通的な機能を抽出する。また、現状の社会課題やニーズの状況を基に、将来的に存在し得る先端的サービスを想定し、それらに共通的に必要となる機能を先進技術のニーズとして列挙する。

(b)先端的サービスを支える先進技術の検討

調査結果によって得られた先進技術について、調査・選定の観点（①先端的サービスを支える先進技術としてのニーズがあるか、②相互運用性や相乗効果をもたらすかどうか、③社会動向を考慮した将来性があるか）に照らして検討する。

(c)推奨すべき先進技術の決定

上記で検討した先進技術及びそれらの有用性についてまとめ、最終的に推奨すべき先進技術の決定を行う。

2-6-2.　先端的サービスを支える先進技術とデータ連携基盤の接続検証

(1)考え方

本項では、先端的サービスを支える先進技術と各エリアで構築されるデータ連携基盤の接続検証のポイントと進め方について記載する。本項で検証を想定している範囲を以下の図 2-6-2-1 に示す。

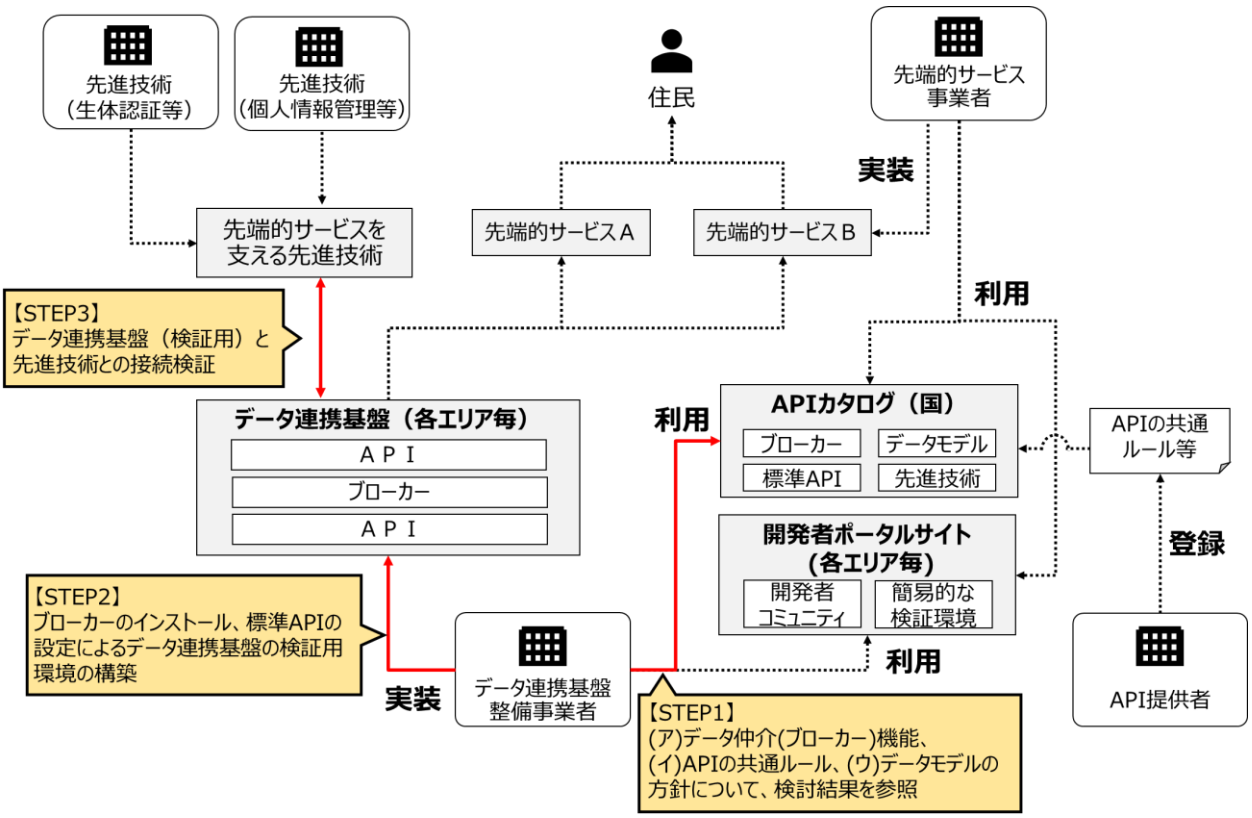


図 2-6-2-1　接続検証の範囲

(2)接続検証のポイント

2-6-1 で選定した共通に推奨すべき技術に関して、特に「先端的サービスを支える先進技術において相互運用性や相乗効果をもたらすか」という観点について、本調査業務を通じて調査された（ア）データ仲介（ブローカー）機能、（イ）API の共通ルール、（ウ）データモデルの方針の検討内容と、先端的サービスとの親和性及び相互運用性を検証するプロセスが必要である。本プロセスでは、先端的サービスを支える先進技術とデータ連携基盤との接続等の検証として、以下 2 つの観点に基づき、机上による評価と実施体制内 PoC 環境を用いた接続検証を行い、先進技術との親和性の検証や相乗効果の検証、及び技術的

課題の洗い出しを行う。

(a)データ連携層である API の共通ルールに則しているか

先端的サービスを支える先進技術が、（イ）で検討する API の共通ルールに則しているかを検証する。この際、先進技術が API の共通ルールに則していない場合、連携させるために解決すべき課題を標準 API と先端的サービスの視点で整理する。

(b)データ層に位置するデータモデル及びデータ仲介（ブローカー）機能の仕様に則しているか

先端的サービスを支える先進技術が、（ウ）で検討されるデータモデルを活用できるか、及び（ア）で検討されるデータ仲介（ブローカー）機能と連携したデータ流通が可能か、ブローカーがサポートする API 形式と適合しているかという観点で親和性を検証する。

机上評価では、データ連携基盤及び他サービスとの相互運用性や、複数サービスの同時利活用を想定した評価項目についての検討を想定している。その中でも本調査業務のベースとなるデータ連携基盤との接続に関する部分の検証は、（ア）～（ウ）の検討内容を踏まえたデータ連携基盤の実施体制内 PoC 環境を構築し、対象を絞った上で詳細な検証を行うことを想定している。

本プロセスにて想定する机上評価及び実環境を用いた接続検証による評価と、2-6-1 における共通的に推奨すべき技術の選定との関係性を以下に示す。

表 2-6-2-1　机上評価及び実環境を用いた接続検証による検討評価項目

分類	検討評価項目	検討・評価実施タイミング		
		共通に推奨すべき技術の選定	机上評価	実環境を用いた実証評価
スーパーシティに活用される技術・ニーズ	先端的サービスを支える先進技術としてのニーズがあるか	●		
	各種サービス分野との相互運用性や相乗効果が期待できるか	●		
	社会動向を考慮した将来性・拡張性があるか	●		
相互運用性を考慮した接続・連携が可能か	データ連携基盤との相互運用性に資する標準的な仕様に則しているか		●	
	データ連携層である API の共通ルールとの親和性		●	●
	データ層であるデータモデル・データ仲介（ブローカー）機能との親和性		●	●
複数サービスを同時に利活用した場合に発生し得る課題の観点	複数サービスの間での連携時の接続容易性や技術的課題		●	
	複数サービスを同時に連携する際の動作おける課題		●	
	複数サービスを同時に連携する際のデータの一貫性に関する課題		●	

(3)接続検証のプロセス

先端的サービスを支える先進技術とデータ連携基盤の接続検証におけるプロセスを以下に示す。

(a)検証対象となる先端的サービスを支える先進技術の選定

2-6-1 にて選定した先進技術の中から、ニーズ・相互運用性・将来的な利活用可能性の観点で特に今回優先順位が高いと想定される技術を評価（机上及び実環境を用いた接続検証）の対象として選定する。また、実環境を用いた接続検証の対象選定にあたっては本調査業務の期間内で接続検証を効率的に推進するために、本調査業務の受託事業者が既に提供可能なサービスを優先する。

(b)机上評価の実施

(a)にて選定した机上評価の対象となる先端的サービスを支える先進技術について、表 2-6-2-1 に挙げる観点を用いて、実環境を用いた接続検証に先立つ接続性の観点での机上評価を実施する。

(c)データ連携基盤（検証用）の構築

本調査業務の 4 章において検討した仕様・基準及びブローカーのプロトタイプをベースに、実施体制内 PoC 環境にデータ連携基盤（検証用）を作成する。具体的にはブローカー、またデータ連携に必要な API を動作させるために必要な OSS 等のインストール、設定作業等を行う。

(d)接続検証対象の先端的サービスを支える先進技術からデータ連携基盤（検証用）への接続設定

接続検証対象の先進技術（本調査業務の受託事業者各社の既存サービス提供環境）からデータ連携基盤（検証用）への接続設定を行う。なお、接続検証対象の先進技術とデータ連携基盤（検証用）を接続する際に簡易的なスクリプト等が必要な場合は、この準備についても本プロセスにて実施する。

(e)接続検証対象の先端的サービスを支える先進技術からデータ連携基盤（検証用）への接続テスト

接続検証対象の先進技術とデータ連携基盤（検証用）の接続テストを行う。

(f)接続検証対象の先進技術からデータ連携基盤（検証用）への接続テストの結果評価

2-7. （オ）その他データ分散型のデータ連携基盤を運用する上で必要となる事項

2-7-1. データ連携基盤における運用面での課題認識

(1)課題認識

データ連携基盤を構築し、運用を開始しても、扱うデータや搭載されるサービスが陳腐化すれば次第に利用されなくなる。社会環境は常に変化を続けるため、データ連携基盤の陳腐化が始まることを認識し、常にニーズに合わせて変化を続けていける仕組みが必要である。具体的には、静的なオープンデータのみならず各分野において鮮度のあるデータを扱えること、搭載されるサービスが競争領域の中で常に進化できること、そしてデータ連携基盤を含むスマートシティ全体が時代の変化を察知して追従できることが望まれる。

さらに、陳腐化に対する検討だけでなく、安心してデータを扱える仕組みの考慮も必要である。住民のQoL（Quality of Life）を向上するため、あるいは個人の行動変容を促すためにも、データ連携基盤を用いた個人情報の活用は重要である。その一方で、個人のプライバシーは保護されなくてはならない。しかし、各個人の個人情報に対する捉え方は多様である上、場面や状況によっても異なる。そのため、個人情報の保護に関する法律に従ってデータを活用した場合でも、場面や状況によって、非難・批判が発生し、収集できない事例が過去に発生している。非難・批判が起きた場合は、データ連携基盤を用いて個人情報を活用することができなくなる可能性が大いにある。長期的にデータ連携基盤を用いた個人情報の活用を続けるためにはオプトイン・オプトアウト及びデータのセキュリティ担保について場面や状況に合わせた適切な対応が必要となる。

最後に、スーパーシティに関わるステークホルダ間の温度差や利害関係が異なることによる検討の停滞が発生する可能性も考慮する必要がある。データ連携基盤を長期的に運用する際に、改修、拡張、あるいは運用方法の変更等が発生する場合がある。その際、当該基盤の目的、スコープ等が不明確な場合、ステークホルダ間の利害関係等から検討が紛糾する可能性がある。時間が経過して関わる利害関係者が変動するにつれ、初期の暗黙知が崩れていくため、変更すべきでない事項を形式化しておく必要がある。

(2)課題認識から導出した検討事項

データ連携基盤を長期運用するためには、データ連携基盤の目的、スコープ等を定めた上で、ニーズの変化に柔軟に対応する仕組みや安心してデータを扱える仕組みを運用面（ルール）及びシステム面（技術）の両方の視点で検討する必要がある。そこで、本調査業務体制の参画事業者が有する知見を基に、優先度の高い観点として「陳腐化しない運用の仕組み」、「安心してデータを扱える仕組み」を取り上げ本調査業務にて考察する。

(a)陳腐化しない運用の仕組み

(ｱ)データ連携基盤が公開するAPIやデータの鮮度を維持する仕組み

幅広いエリアにおいてデータ連携基盤・API カタログ・開発者ポータルサイトの積極的な導入と継続的な運用を実現するためには、データ連携基盤が公開するAPIやデータといったコンテンツを常に最新化するための仕組みの実現、さらにその仕組みを継続的に運用していくための運営環境（エコシステム）の実現が求められる。以下に、それらを実現するために必要と想定される検討事項を記載する。

1) データ連携基盤・APIカタログ・開発者ポータルサイトの運営に関する検討

継続的な運営を実現するために、運営主体についての検討が必要と想定する。運営主体の検討にあたっては、本調査業務の推進に係る責任分担の定義（説明責任等）や公平性の観点についてもあらかじめ検討が必要と想定する。

2) 認知度向上・普及促進に向けた施策案の検討（宣伝・拡販）

継続的利用・運用を実現するための大前提として、まずはデータ連携基盤やAPIカタログ・開発者ポータルサイトの存在及びその利用価値について、広範囲にわかりやすく宣伝する施策が重要と想定する。そこで、顧客分析のフレームワークを用いて、サービスの認知から継続的利用に至るまでのプロセスを踏まえ、積極的な利活用に至るまでに課題が生じ得る箇所と対応策の整理が必要と想定する。

3) データ連携基盤・APIカタログ・開発者ポータルサイトの利活用向上に向けた施策案の検討

継続的な利活用の実現のために、データ連携基盤として実装されている技術や機能、APIカタログや開発者ポータルサイトとしてのコンテンツは、常に最新技術動向や利用者のニーズを反映したものである必要がある。そこで、評価システムのような利用者からのフィードバック機能を活用し、最新ニーズへ対応することで、コンテンツの継続的な最新化に資する仕組みの検討が必要と想定する。また、利用規約等における制約の度合い（商用利用へのハードルが高い、データ活用時の制限が多い等は利活用の妨げとなる）についても併せて検討が必要である。

(ｲ)時代の変化に応じて柔軟にシステムを拡張する仕組み

短時間で技術革新が進む現代のシステムでは、その恩恵を受けるために機能ごとに更新できる構成となっている必要がある。ビルディングブロックの概念に沿ってシステムを実装し、ブロック間での情報のやり取りをAPIとして統一することにより、他のビルディングブロックや、住民に提供されるサービス自体には影響を与えることなく一部のブロックのみ更新できるように設計することが望ましい。ビルディングブロック構成は技術革新に対応するだけではなく、各地の事情に応じた実装が可能となり、サービス停止、ハッキング等による障害の範囲を限定することができる等の利点もある。また、各ビルディングブロックが独立していることで、新しいビルディングブロックを追加し、システム全体の機能をアップグレードすることも可能となる。

(b)安心してデータを扱える仕組み

(ｱ)個人データのオプトイン管理や適切な運用ルール

住民が安心してデータ連携基盤を利用するためには、「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」の相互運用性に関する基本原則にあるとおり、データの管理に透明性を持つ必要がある。データ連携基盤の定義に記載しているとおり、スマートシティ及びスーパーシティにおいて流通するデータの多くは、住民である各個人に由来するものである。つまり、データ主体は住民であり、その利用に対しては各個人がその権利を行使できるようになっていることが重要である。個人を由来として発生する情報については、その個人がデータの流通をコントロール可能な状態にしておくべきで、該当データがどのように扱われているかを適切に開示することができる必要がある。データの管理においては、適切な同意のもと、個人のデータがどのように利用されているかが確認できるようになっていること、そして必要に応じてその利用について適切なオプトアウト手段が提供されている必要がある。

(ｲ)各種連携データの品質を確保するための対策

サービス品質を維持・改善するためには、データ品質管理（品質低下防止と継続的な品質改善）を行うことが重要である。品質低下防止対策としては、サービスを提供するために必要十分なデータ項目を含むデータモデルであるか、収集されたデータ項目値が定められた精度となっている

か、などについて考慮する必要がある。継続的な品質改善対策としては、定性的・定量的評価に基づくデータモデルの改善を継続的に行うことについて考慮する必要がある。

2-7-2.　課題抽出の観点及び調査方針

(1)課題抽出の観点

2-7-1 にて記載した課題認識については、データ連携基盤を長期運用する上で必要、かつ優先度の高いものを検討項目として定めたため、仮説の段階である。そのため、検討の具体性と網羅性を担保する必要があり、以下の(2)に記載するとおり、他に優先すべき検討項目がないかを判断する必要がある。

(2)調査方針

調査方針は、(a)～(c)の手順にて実施予定である。

(a)スーパーシティの取り組みを検討している自治体及び民間事業者を選定（5 か所程度）し、実現するデータ連携基盤を長期運用する上で課題となる事項を当該自治体に対してヒアリングを実施する。なお、ヒアリング対象となる自治体については、世界経済フォーラム（英：World Economic Forum）と相談し、スマートシティ推進中又は検討中の自治体を対象とする。また、ヒアリング内容については、「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」に記載された「スマートシティにおける連携レイヤー」の実施事項に準ずる形で実施する。

(b)(a)の結果から、データ連携基盤の長期運用に係る課題を導出し、まとめる。まとめ方の想定として、「スマートシティにおける連携レイヤー」を用いて、各自治体における課題を以下のように可視化する。また、本調査事業のスコープの想定を赤枠で示す。

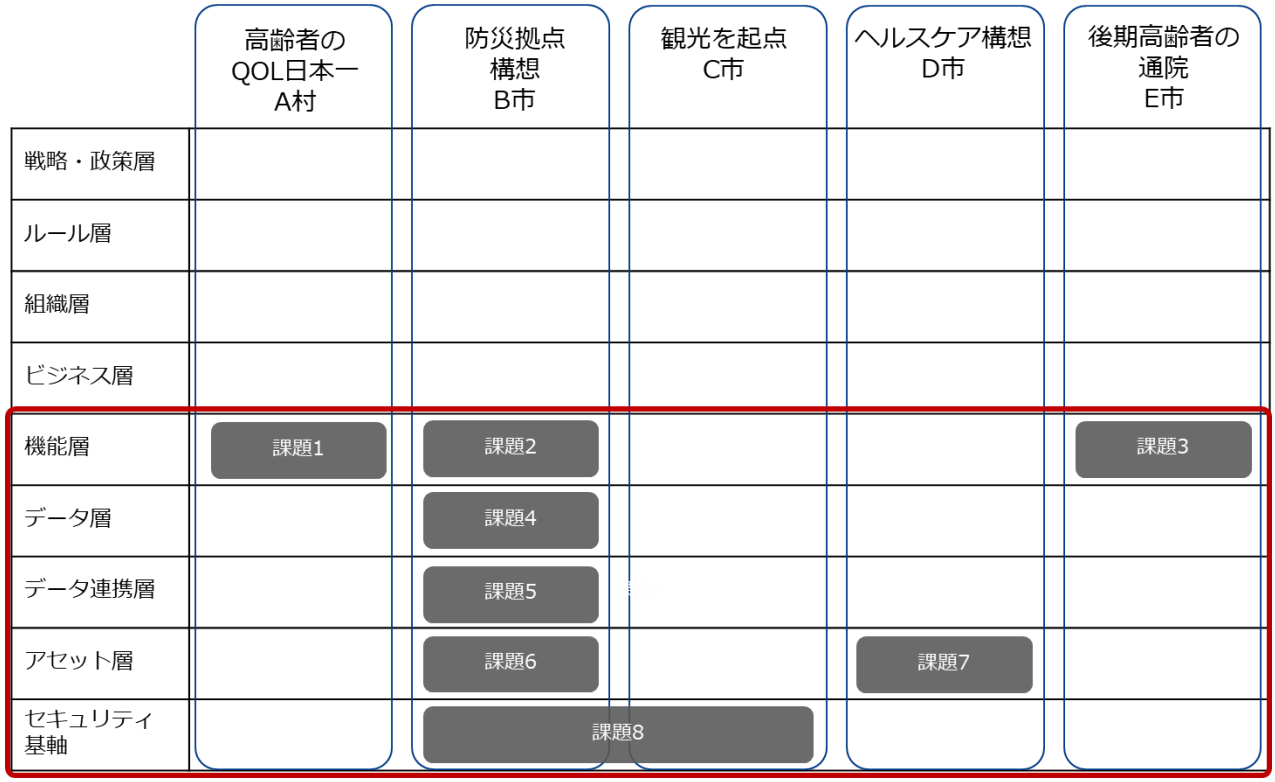


図 2-7-2-1　「スマートシティにおける連携レイヤー」と自治体ごとの課題

(c)各自治体に共通して課題となる特に重要な事項に関して、内閣府又は有識者と協議の上、解決するための仕組み（運用面、システム面）に関する案を提示する。

(3)課題抽出の例

課題抽出の際には、例えば下記のマトリックス図を活用し、自治体にヒアリングしたユースケースをベースとし、整理する。以下は令和 2 年 10 月付で内閣府地方創生推進事務局より公表されている「『スーパーシティ』構想について」にある「観光を起点とする B 市のスーパーシティ構想」を取り上げて検討した課題抽出の例（抜粋）である。

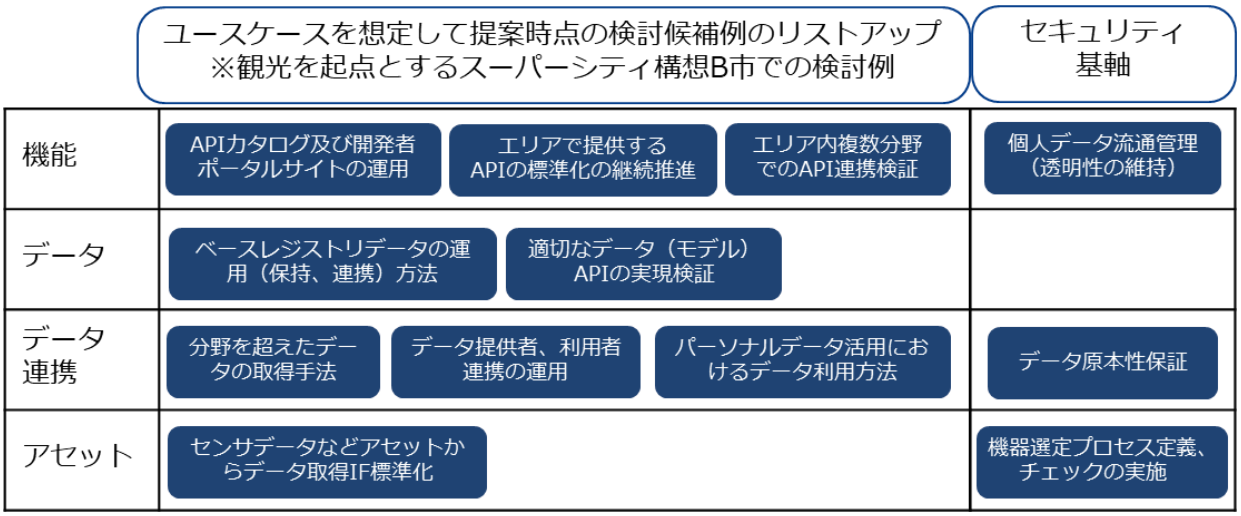


図 2-7-2-2　課題抽出例（マトリックス図）

(a)機能層（エリア内複数分野での、API 連携検証）

スーパーシティの運用についてはエリア内の複数分野（もしくは複数エリア）でのサービス間においての API による機能連携の実現がユースケース上で実現されることが必要であり、実際に API を用いた連携がうまく進むか検証が必要となる。

(b)データ層（ベースレジストリデータの運用（保持、連携取得）方法の検証）

地理、空間情報等、重要かつそのデータサイズが膨大であるデータ等については、データ連携基盤の中でどのような保持方法がなされるべきかの検討が必要である。

(c)データ連携層（分野を超えたデータの取得方法の検証）

エリアのサービスの実現においてデータ連携基盤として、必要なデータの取得を支援する機能の実現が必要である（データの検索、取得）。

(d)アセット層（センサーデータ等アセットからデータ取得インターフェース標準化）

アセットとしてデータ連携基盤が取得すべきデータについては、センサー（位置情報、水位情報）をはじめとして多様なデバイスからのデータを取得可能とするためにインターフェースの標準化等の検討が必要である。

(e)セキュリティ基軸

(ｱ)機能層

長期運用する上では、各個人が自身のデータ流通を管理・メンテできるようにし、個人の意思をもってコントロールさせることで透明性の長期維持が達成される形になっているものが必要である。

(ｲ)データ連携層

データ原本性保証は長期にわたって安心安全に運用するために必要である。セキュリティバイデザインに基づき実装されたセキュリティ対策の維持メンテを長期にわたり実施していくことが重要である。

(ｳ)アセット層

機器選定プロセスの定期的な見直しや、選定された機器が選定時と同等のセキュリティの担保が機能しているか、定期的にチェックできる仕組みが重要である。

3. 調査結果：データ連携基盤の全体像

3-1. 先端的サービスの例

(1)先端的サービスの目的

スーパーシティは住民のために設計されるものであり、Society5.0 で実現する社会コンセプトのように、一人一人が快適で活躍できる社会の実現が目指される。住民が参画し、住民目線で 2030 年頃に実現される地域の未来社会を支えるサービスが先端的サービスであり、AI やビッグデータなどの先進技術を活用し、「移動」「物流」「支払」「行政」「健康」「医療介護」「教育」「観光」「エネルギー水」「インフラ防災」「防犯安全」「都市 OS 運営」等複数分野で提供されるものである。

(2)先端的サービスイメージ

ここでは提供されるサービス例として、内閣府地方創生推進事務局『「スーパーシティ」構想について』[8]にて記載されている、『参考２．「スーパーシティ」構想イメージ』の事例等をモデルに、本調査事業にて地域課題の深堀を行い、課題解決に資するユースケースの作成を実施した。ユースケースでは、単一自治体エリアで実装されるサービスとして、「高齢者通院サポートサービス」「高齢者避難支援サービス」「ものづくりツーリズムサービス」、また複数自治体エリアを跨って実装されるサービスとして「シニア向け観光サポートサービス」の仮説を立てている。作成にあたっては、スーパーシティ先端的サービスとして求められる複数分野で提供される要件だけでなく、分野を横断することやエリアを横断するサービス要件として記載し、後述するデータ連携基盤に必要となる要件が検討できるユースケースとしている。

サービスイメージ例として「高齢者避難支援サービス」を図示する。その他「高齢者通院サポートサービス」「ものづくりツーリズムサービス」「シニア向け観光サポートサービス」等詳細については【別紙ウ-1 ユースケース集】を参照すること。

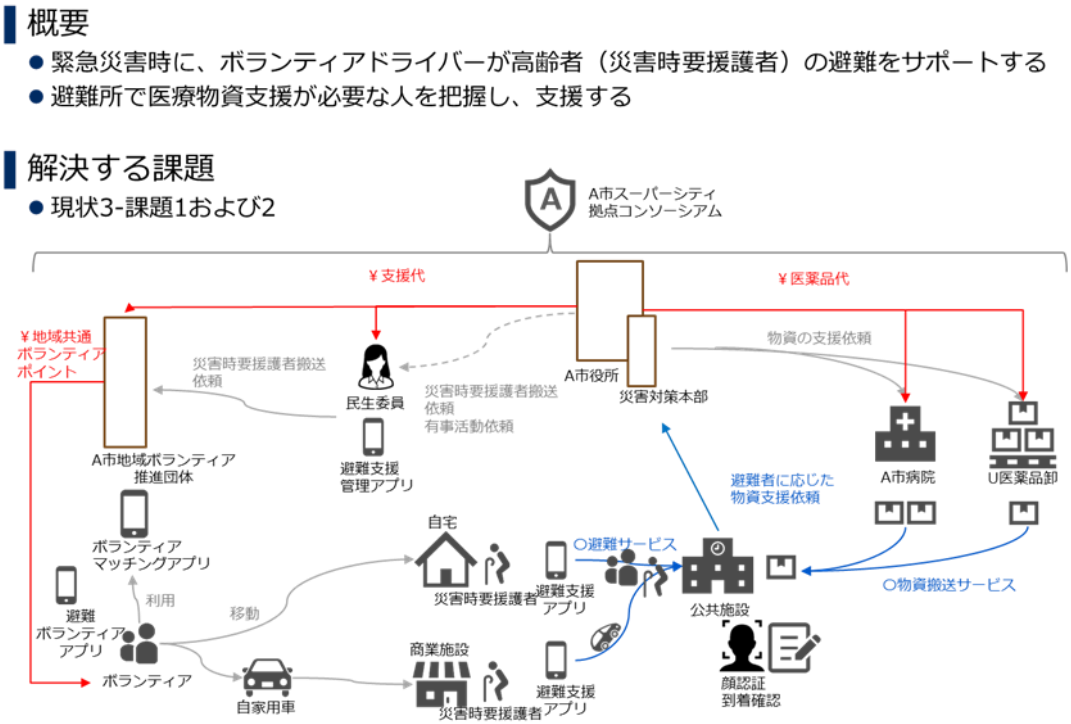


図 3-1-1　高齢者避難支援サービスのサービスシナリオ

[8] 引用：https://www.kantei.go.jp/jp/singi/tiiki/kokusentoc/supercity/supercity.pdf

3-2. データ連携基盤の全体像

3-2-1.　Society 5.0 リファレンスアーキテクチャとの関係性

(1) スーパーシティとスマートシティリファレンスアーキテクチャとの関係性

データ連携基盤のスコープを各ステークホルダが共通認識するため、スマートシティリファレンスアーキテクチャをスーパーシティにあてはめ、データ連携基盤と先端的サービスの位置付けを整理した。スマートシティリファレンスアーキテクチャから見るスーパーシティの構想を図 3-2-1-1 に示す。

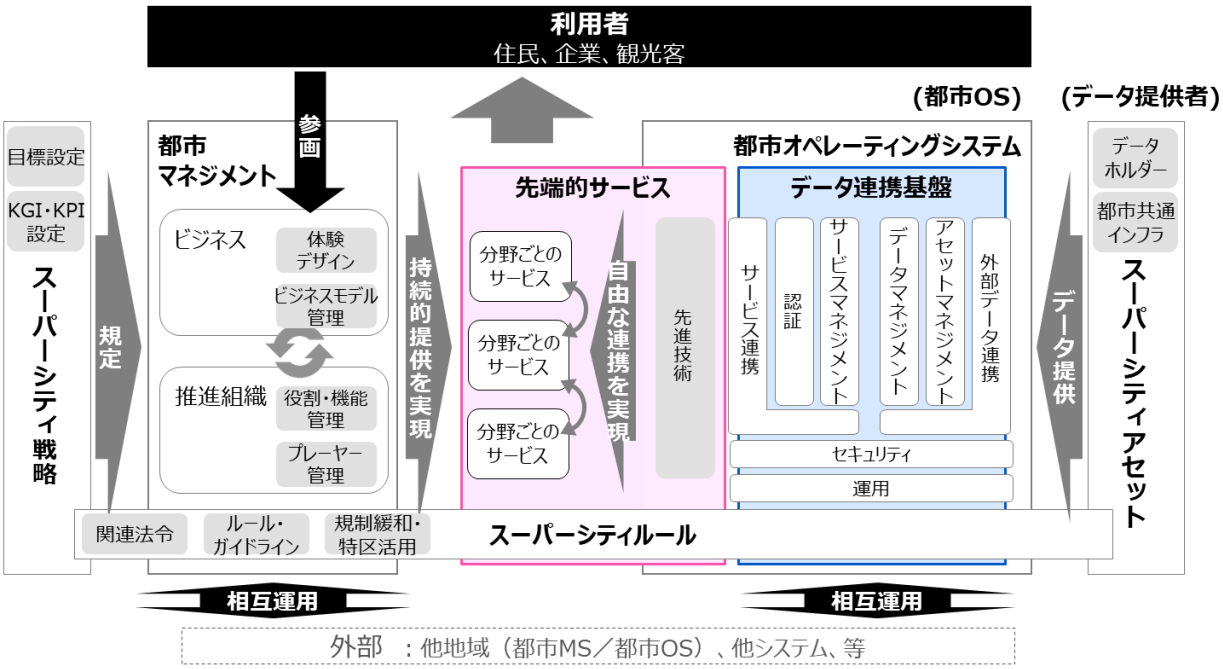


図 3-2-1-1　スーパーシティとスマートシティリファレンスアーキテクチャとの関係性

データ連携基盤は各地域に整備され、相互運用性の確保を目的とした最低限の機能を持ち、各地域に分散されたデータの集積や配信、分野ごとのサービスの連携を担うことで、分野間/都市間における自由な連携を実現し、先端的サービスの整備に寄与するものである。また、データ連携基盤及び先端的サービスを整備する際には、スマートシティリファレンスアーキテクチャにおける都市 OS[9]を設計図として参照することで、成果の再利用や分野間/都市間の相互運用を実現しやすくすることとなる。

データ連携基盤を中核に、先進技術を活用した多様な機能が組み合わさることで、住民目線の課題解決のための先端的サービスの整備、及びより高度な課題解決を目指し、スマートシティ/スーパーシティを推進する。これに加えて、都市マネジメントによる持続的な提供と、住民をはじめとした多くのステークホルダが関わることでイノベーティブな新たなサービスが生み出され相乗効果が生まれる。

[9] 都市 OS：スマートシティ実現のために、スマートシティを実現しようとする地域が共通的に活用する機能が集約され、スマートシティで導入する様々な分野のサービスの導入を容易にさせることを実現する IT システムの総称。

(2)データ連携基盤と都市 OS の関係性

データ連携基盤は、スマートシティリファレンスアーキテクチャにおける都市 OS を参照し整備されることとなるが、データ連携基盤と都市 OS の関係性について、Society 5.0 リファレンスアーキテクチャを基に検討された「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」と「スマートシティリファレンスアーキテクチャ」を比較して整理した。

「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」と「スマートシティリファレンスアーキテクチャ」では、データ連携に資する基盤について表 3-2-1-1 のとおり、それぞれ記載されている。

表 3-2-1-1　データ連携基盤と都市 OS

参考文献	名称	説明
スーパーシティ/スマートシティの相互運用性の確保等に関する検討会 最終報告書	データ連携基盤	ビルディングブロック方式を用いて構成され、公開された API を通じてデータの集積や配信を行う機能。 データ連携基盤は、スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書における、データ連携層・データ層に該当する。
スマートシティリファレンスアーキテクチャ	都市 OS	スマートシティを実現しようとする地域が共通的に活用する機能が集約され、スマートシティで導入する様々な分野のサービスの導入を容易にさせることを実現する IT システムの総称。 都市 OS は、スマートシティリファレンスアーキテクチャにおける、機能層・データ層・データ連携層に該当する。

各参考文献を基に、スーパーシティにおけるデータ連携基盤のレイヤーを、図 3-2-1-2 に整理した。

各参考文献ではそれぞれのアーキテクチャ視点が異なるため一部レイヤーが交差している。例えば、「相互運用性の確保等に関する検討会」における機能層は、個々のサービスやサービス群について言及しているが、スマートシティリファレンスアーキテクチャにおける機能層は、サービスと連携するために必要な機能（API や認証等）について言及しているといった点である。

データ連携基盤は、「相互運用性の確保等に関する検討会」における、データ層、データ連携層が中心となるが、データや API を公開するにあたり、スマートシティリファレンスアーキテクチャにおける機能層も一部必要と考える。

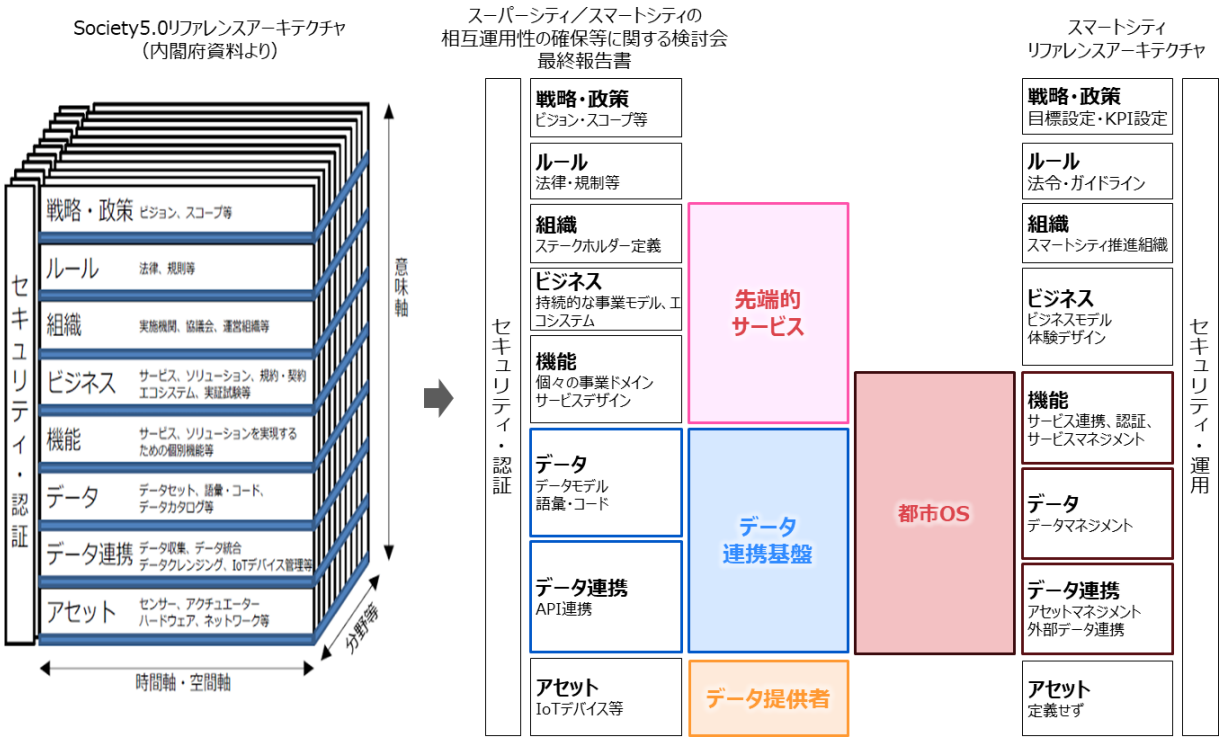


図 3-2-1-2　Society 5.0 リファレンスアーキテクチャとの関係性

3-2-2.　データ連携基盤に必要となる構成要素の導出（仮説設定）

データ連携基盤に必要となる構成要素の導出にあたり、データ連携方式を整理した上で、データ連携基盤に求められる要件と、それを満たす機能について、「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」に記載されているサービス及び相互運用性の観点を中心に整理する。各観点で導出した構成要素については、本報告書（ア）～（エ）の調査結果を踏まえ、3-2-3 にて本報告書におけるデータ連携基盤の全体像を示す。

(1)データ連携方式

データ連携基盤におけるデータ連携方式を、「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」に記載されているデータ仲介の分類（データ蓄積方式、データ分散方式）から整理した。

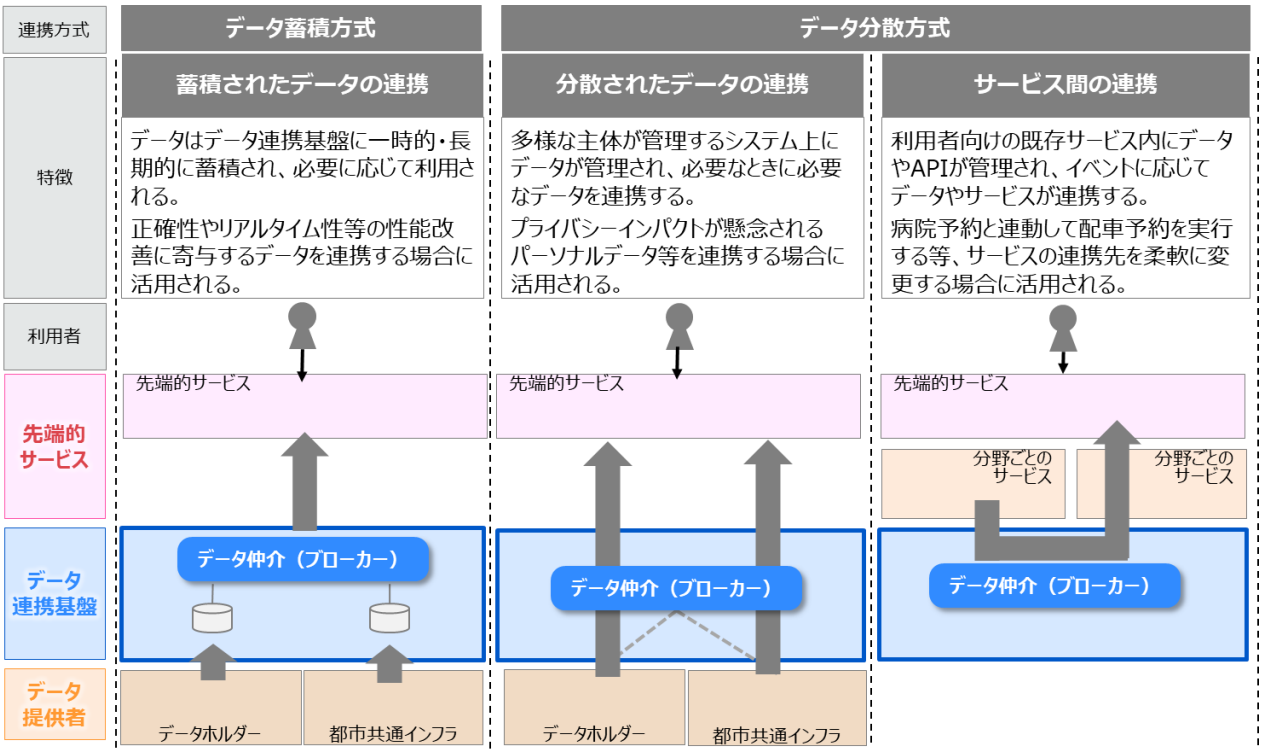


図 3-2-2-1　データ連携方式

図 3-2-2-1 で整理した各連携方式を実現するためにデータ連携基盤に必要となる事項について、相互運用性の確保及び先端的サービスの観点から導出する。導出過程を次項(2)、(3)、(4)にて示す。

(2)相互運用性の確保に必要となる事項

スーパーシティにおけるデータ連携基盤は、公開された API を通じてデータの集積や配信を行う機能を提供するものである。データ連携基盤の API は、誰もがデータ連携基盤に接続でき、かつサービス間データ連携により複数分野を跨った便益を提供可能とするために、開発者ポータル（API カタログ）上で公開することで相互運用性を確保する必要がある。

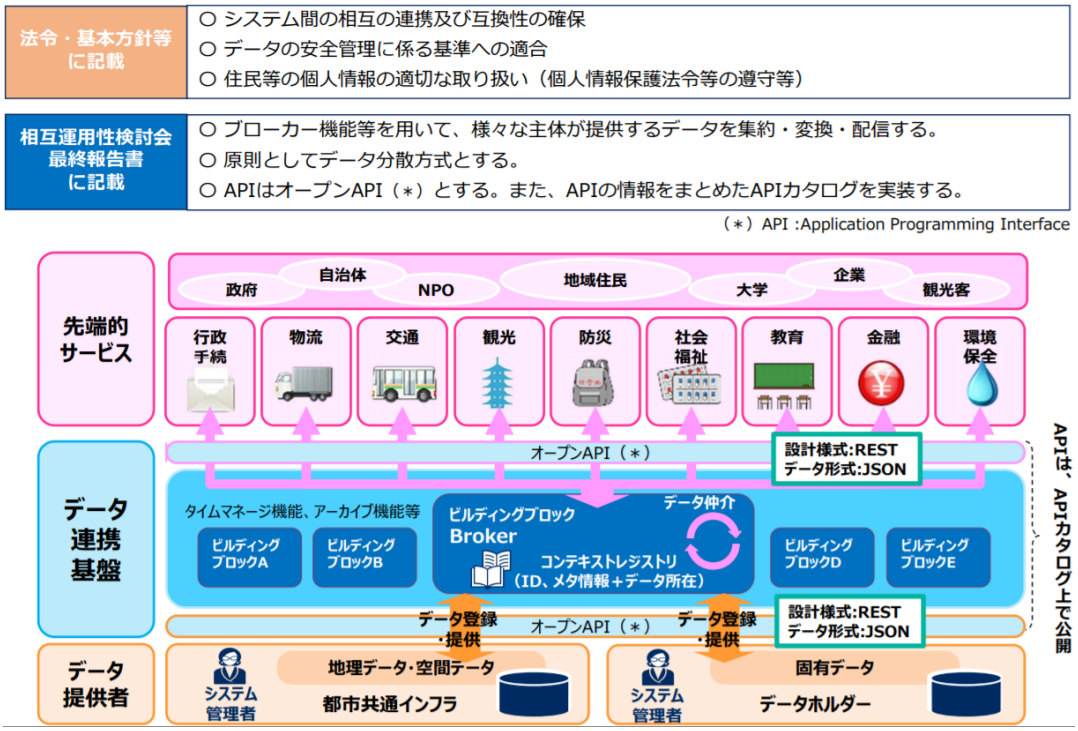


図 3-2-2-2　スーパーシティ構想におけるデータ連携基盤について

「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」に記載されている相互運用性を確保に必要となる事項から、データ連携基盤に必要となる機能は以下と考える。本機能を実現することで、データ連携基盤を利用するサービス提供者やデータ提供者が望めば、データ連携基盤に接続できる状態になる。

表 3-2-2-1　相互運用性の確保に必要となる機能（案）

相互運用性確保に必要となる事項	機能
• ブローカー機能等を用いて、様々な主体が提供するデータを集約・変換・配信する • 原則としてデータ分散方式とする	データ仲介（ブローカー）
• API はオープン API とする	オープン API
• API の情報をまとめた API カタログを実装する	開発者ポータルサイト（API カタログ）

表 3-2-2-1 に示した機能を活用することで、図 3-2-2-1　データ連携方式に示す蓄積されたデータの連携及び分散されたデータの連携が可能となる。具体的な例を説明する。

(a) 蓄積されたデータの連携

蓄積されたデータ連携の例を図 3-2-2-3 に示す。データ連携基盤はオープン API・データ仲介（ブローカー）を通じ、アセットからのデータ登録を受付け、一時的・長期的なデータ管理と、利用者へのデータ配信を行う。

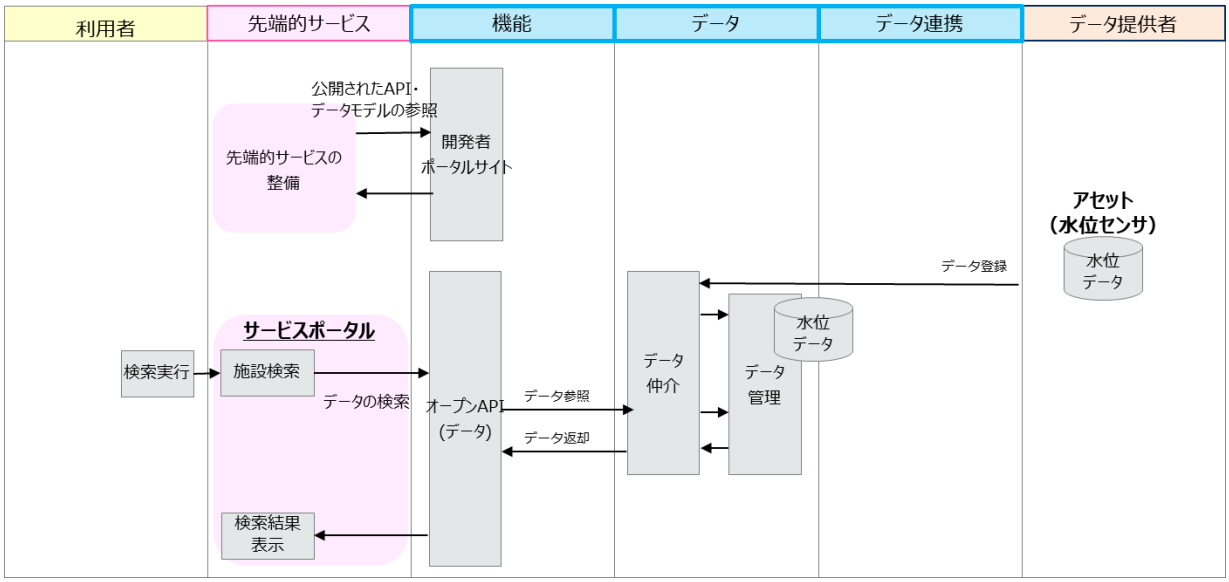


図 3-2-2-3　蓄積されたデータの連携

(b)分散されたデータの連携

分散されたデータ連携の例を図 3-2-2-4 に示す。データ連携基盤はオープン API・データ仲介（ブローカー）を通じ、連携対象となる他システム上のデータを参照し、利用者へのデータ配信を行う。他システムが提供する API やデータモデルが共通的な仕様に則していない場合は、データ変換によりその差異を吸収する必要がある。

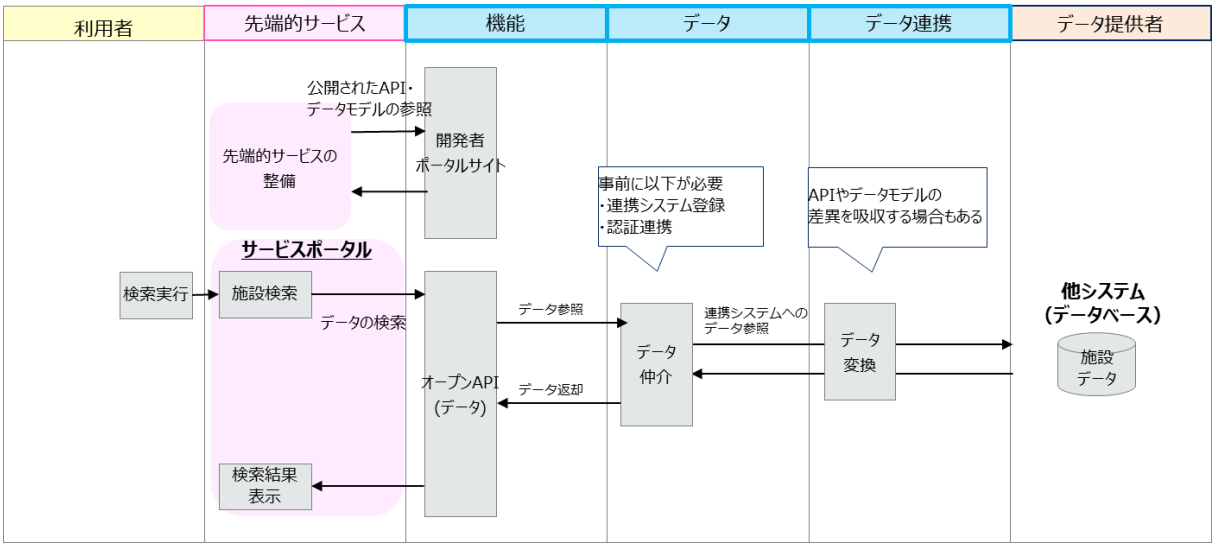


図 3-2-2-4　相互運用性の確保に必要となる機能

(3)単一自治体エリアで実装されるサービスに必要となる事項

前述した先端的サービスの例における単一エリアのユースケースを実現する方法を検討する。単一自治体エリアで実装されるサービスの例にある、後期高齢者の通院対策を図る A 市の構想を抽出し、サービス利用時に必要なデータ連携の要件と、データ連携基盤が提供すべき機能を整理した。

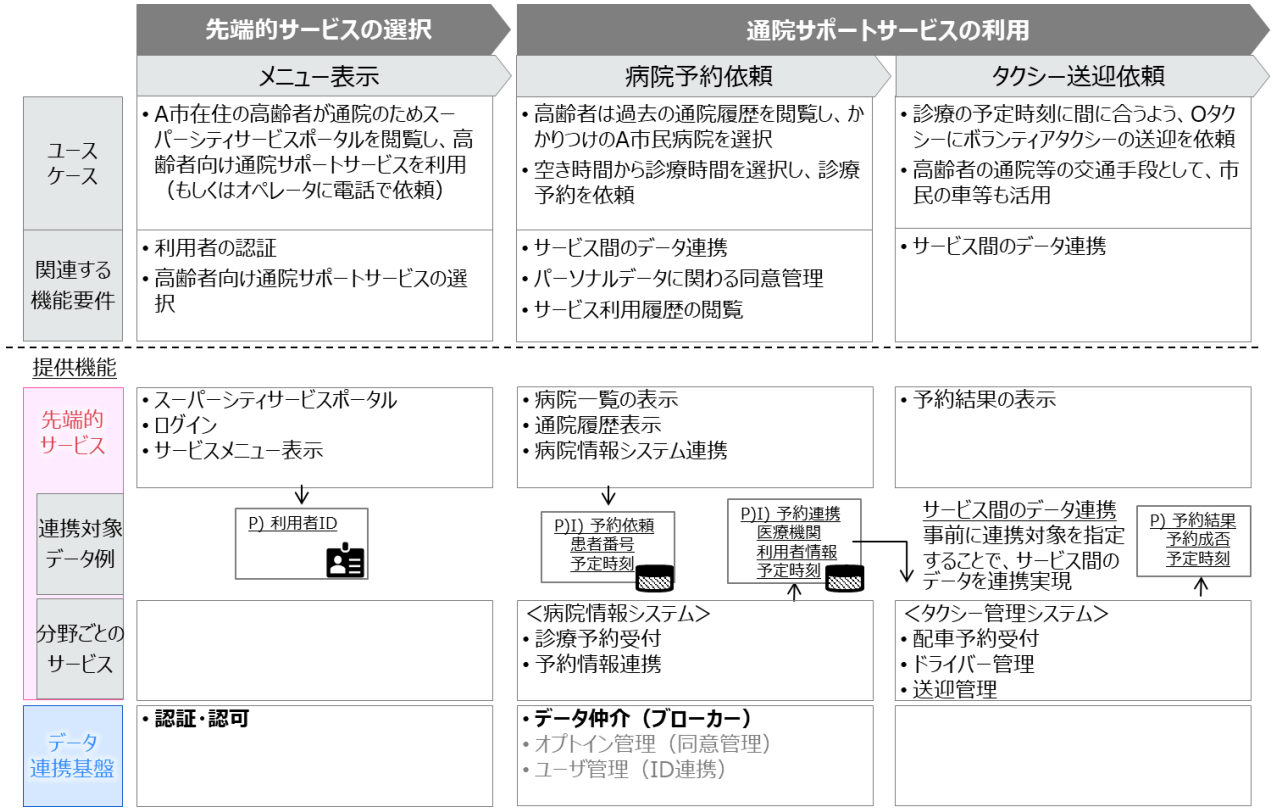


図 3-2-2-5　後期高齢者の通院対策を図る A 市の構想（サービス利用時）

図 3-2-2-5 で整理した結果より、単一自治体エリアで実装されるサービスを実現するために関連する機能要件と、データ連携基盤に必要となる機能は以下と考える。

表 3-2-2-2　単一自治体エリアで実装されるサービスに必要となる機能（案）

関連する機能要件	説明	機能
• 利用者の認証	連携する複数のサービスに対する利用者の認証を連携する	認証・認可
• サービス間の連携	複数のサービス間のデータやサービスを連携する	データ仲介（ブローカー）
• パーソナルデータの同意管理	利用者が同意の上で、第三者にパーソナルデータの公開範囲を指定する	オプトイン管理（同意管理）

本機能を実現することで、図 3-2-2-1　データ連携方式に示すサービス間の連携、及び分散されたデータの連携（特にパーソナルデータの連携）が可能となる。

表 3-2-2-2 の機能要件の中で、調査プロセスに示したとおり特にサービス及び相互運用性の観点を踏まえ、「サービス間の連携」と、連携するデータの種類がパーソナルデータとなった場合に必要となる「分散されたデータの連携（パーソナルデータの連携）」について具体的な例を説明する。

(a)サービス間の連携

サービス間のデータ連携の例と、必要となる機能の仮説を図 3-2-2-6 に示す。利用者は病院情報システムによる通院予約を契機に、タクシー管理システムでの配車予約が連動し、サービス間を連携する先端的サービスを提供することを想定する。このようなサービス間の連携によるユースケースでは、各種サービスを呼び出すためのデータ仲介（ブローカー）が重要になる。

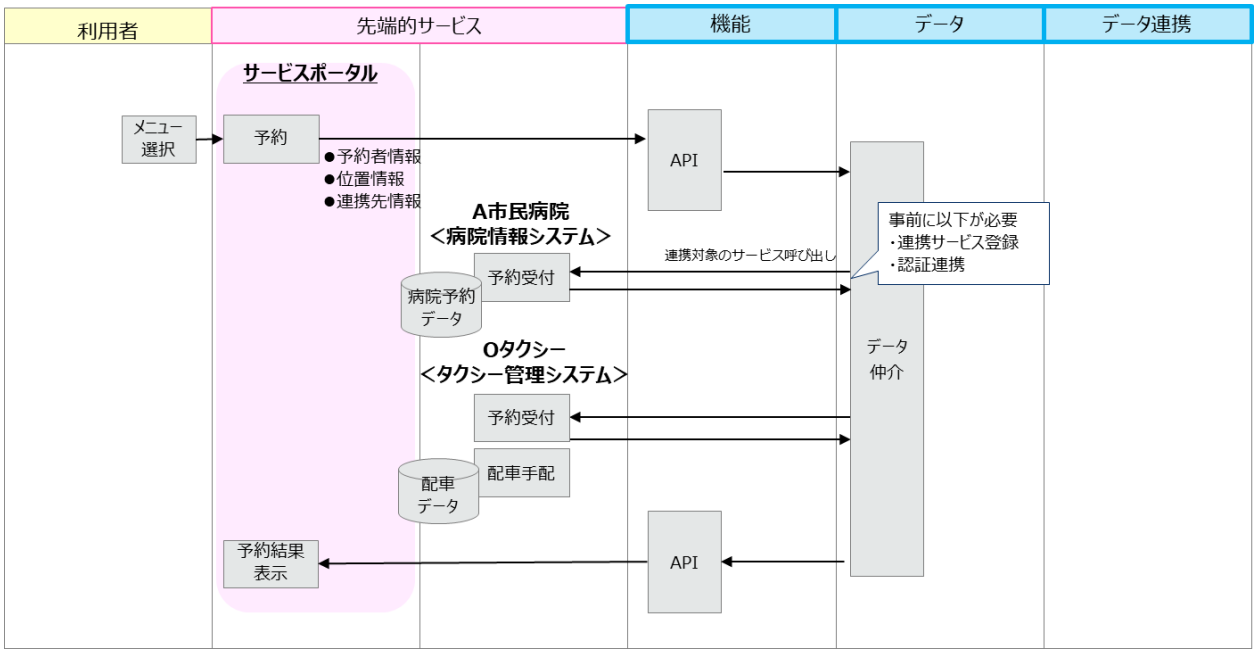


図 3-2-2-6　サービス間の連携に必要と想定される機能

(b)分散されたデータの連携（パーソナルデータの連携）

パーソナルデータの連携には、分散されたデータの連携に加え、利用者の同意の上で必要な時に必要なデータを相互連携・共有する必要があるため、オプトイン管理（同意管理）が重要になる。また、各連携サービスが保持する利用者 ID は異なるため、各サービス間の利用者 ID の名寄せのためのユーザ管理（ID 連携）や、利用者が提供したデータがどのように利用されているかを利用者本人が確認することでデータの管理に透明性を持つための利用履歴管理等が必要になると想定される。

各機能の詳細については本報告書では調査対象外ではあるが、今後検討が必要となる。

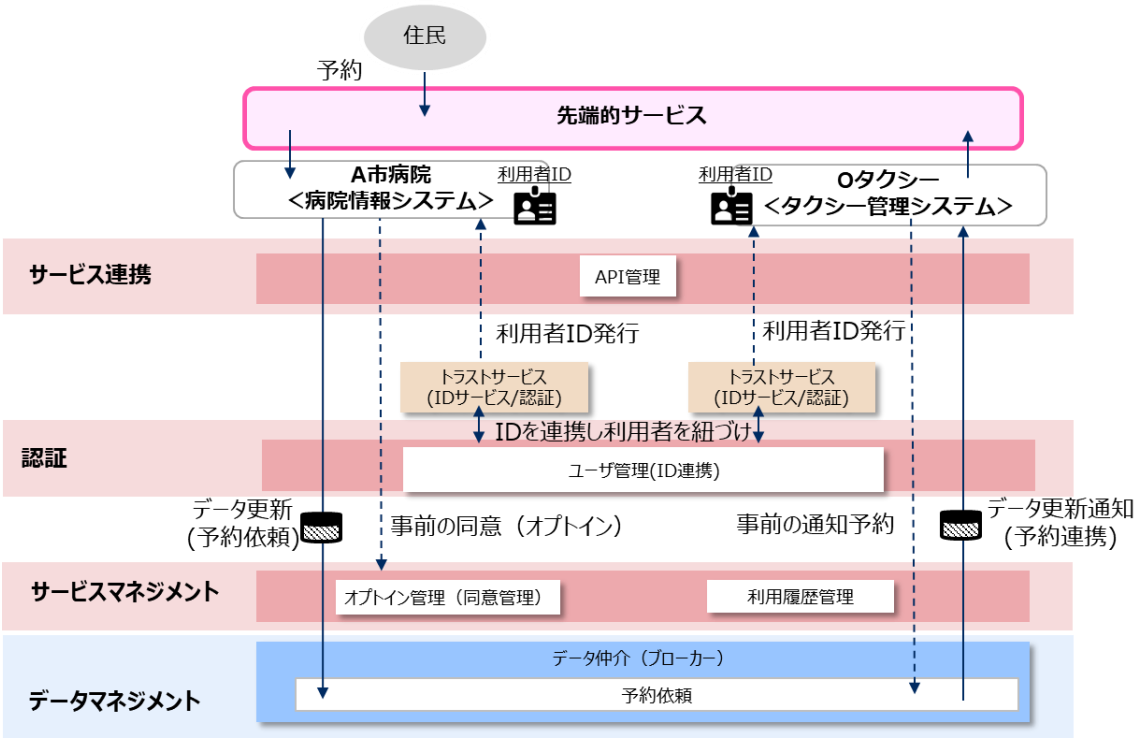


図 3-2-2-7　パーソナルデータの連携に必要と想定される機能（概要）

(4)複数自治体エリアを跨って実装されるサービスに必要となる事項

前述した先端的サービスの例における複数エリアのユースケースを実現する方法を検討する。複数自治体エリアで実装されるサービスの例にある、4-1.［旅行］A 市住民（高齢者）が B 市へ日帰り旅行を抽出し、サービス利用時に必要なデータ連携の要件と、データ連携基盤が提供すべき機能を整理した。

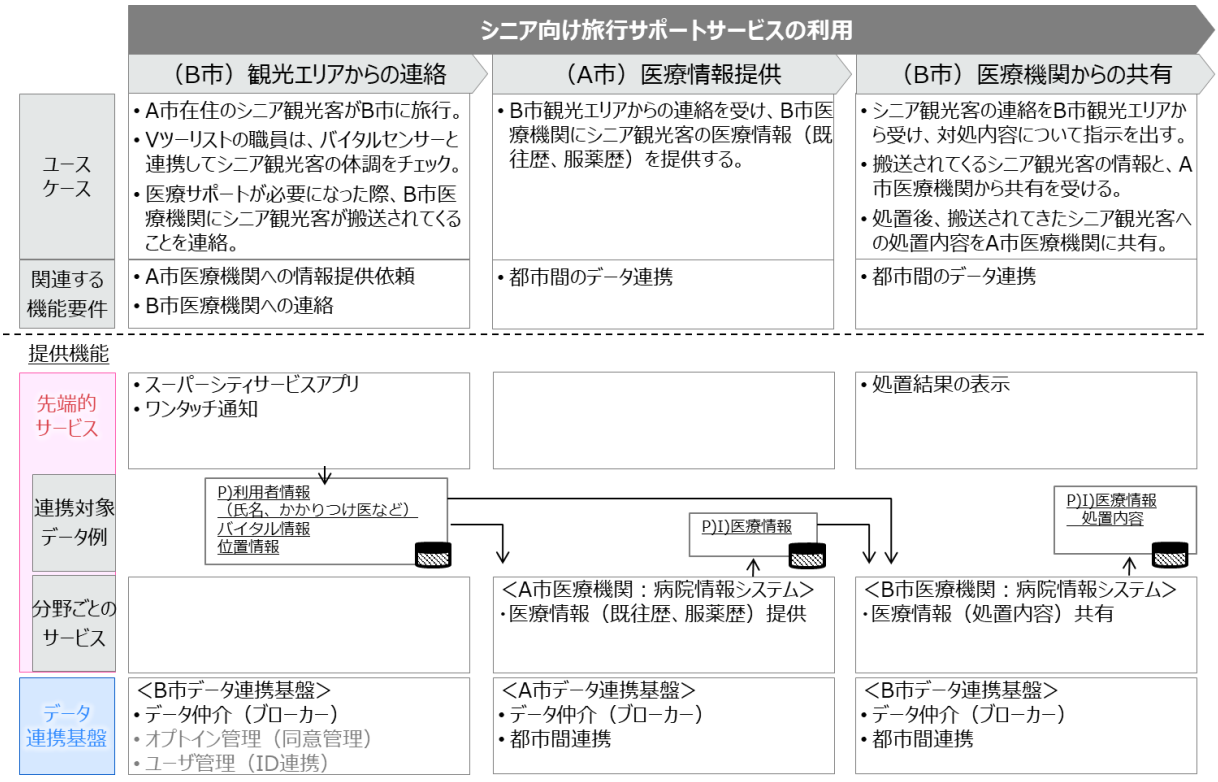
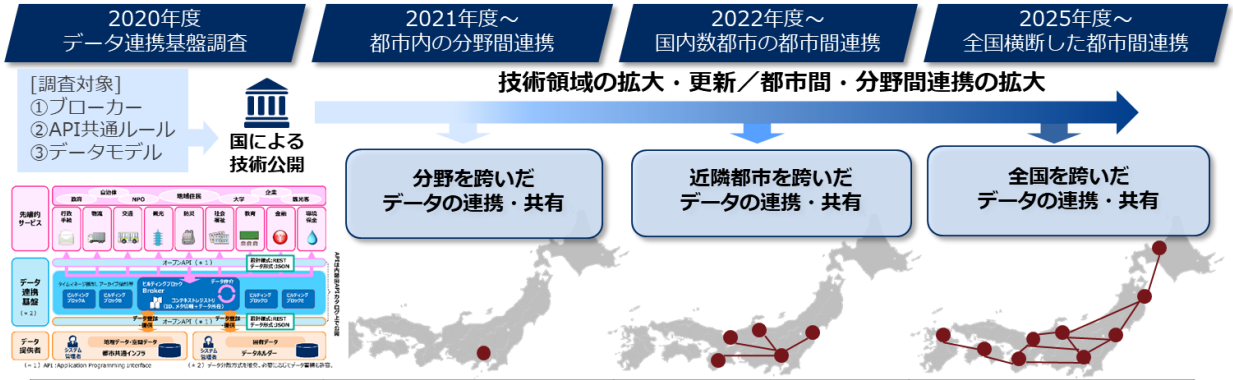


図 3-2-2-8　シニア向け旅行サポートサービス（サービス利用時）

本ユースケースでは、A 市と B 市の医療機関が、B 市のデータ連携基盤に直接つながるパターンや、各都市のデータ連携基盤を通じて連携するパターン等、多様な接続方法により利用者の医療情報が仲介されることを想定する。その際、各都市のデータ連携基盤にて、データ仲介(ブローカー)、API 共通ルール、推奨データモデルが共通的に具備されることで、複数自治体エリアを跨って実装されるサービスにおいても、都市間のサービス連携やデータ連携が容易になると考える。本報告書では、相互運用性の確保を目的とし、データ仲介（ブローカー）の要件案や API 共通ルールの標準仕様案、推奨データモデルについて記載しており、それらを今後活用する地域が増えることにより、分野間・都市間の連携が促進することに期待する。

一方、連携協定等を結んだ都市同士（1:1、1:N）の連携は可能となるが、将来的に全国の都市が相互に連携し、全国を跨いだ都市間（N:N）のデータ連携・共有を実現するためには、より高度な技術要素が必要になると考える。都市間・分野間連携の拡大イメージ（図 3-2-2-9）とそのために必要となる今後の課題について示す。



データ連携基盤の導入地域数		5	100	1000
都市間連携地域数		0	5	100
ブローカー	分野間連携	●	●	●
	都市間連携		●	●
	都市間の連携方式	1：1	1：N	N：N
API共通ルール	REST+JSON	●	●	●
	厳格なAPI共通ルール		●	●
データモデル		推奨データセット	分野拡大	分野拡大

図 3-2-2-9　都市間・分野間連携の拡大イメージ

(a)都市間連携における今後の課題

都市間連携における今後の課題として、全国を跨いだデータの連携・共有のために、特に各都市のデータ所在管理、都市間の厳格な API 仕様の管理、ユニークな ID 管理が必要になると考える。

各機能の詳細については本報告書では記載対象外ではあるが、今後検討が必要となる事項のため、執筆時点の仮説について記載する。

(ｱ)各都市のデータ所在管理

分散型データ連携を実現する場合、データのアクセス先（所在）の管理が必要となるが、連携対象となる都市が増大した場合、その対象となるデータのアクセス先の肥大化や、アクセス先の

更新に関わる運用が煩雑になる可能性がある。そのため、全国を跨いだ都市全体のデータ所在を管理・制御する仕組みや、直接連携していない都市同士を論理的に連携する仕組みが必要になると考える。

(ｲ)都市間の共通ルール

API 共通ルールを相互に実装することで、都市間が望めば互いに接続できる状態になるが、データ連携基盤を導入する都市数が増大した場合、コストや構築時期の関係により API 共通ルールを順守できないことも考えられる。中央主導で指針となる API 仕様のサンプルの提示や、API の実装を容易にする開発ツールを提供、データモデルの継続的な整備等、今回策定した API 共通ルールを各エリアで採用するための仕組みが必要となると考える。

(ｳ)ユニークな ID 管理

スマートシティリファレンスアーキテクチャでは、異なる都市間でデータを相互運用するためには、データに対しグローバルでユニークな ID 管理をする必要があると言及されている。一方、スーパーシティにおいては、データの ID 管理に加えて、データの分散管理を前提とした都市間連携の実現には、利用者、組織、サービス、アセット(デバイス)についてもユニークな ID で管理する必要がある。これにより様々な連携対象からデータ、サービス、個人の特定や、利用者や組織が地域をまたいだ場合でもそれぞれ関連付けすることが可能となる。

3-2-3.　本報告書におけるデータ連携基盤の全体像

各地域で整備されるデータ連携基盤においては、相互運用性の確保の事項で示したデータ仲介（ブローカー）、オープン API、開発者ポータルサイトを最低限必要な機能として整備する必要があると考える。その他の機能は、各地域が解決する課題に応じた先端的サービスのユースケースに従い、ビルディングブロック方式で必要となる機能を整備する必要がある。スマートシティリファレンスアーキテクチャにおける都市 OS の機能を参考とし、データ連携基盤を整備する際の全体像と必要となる構成要素を図　3-2-3-1 に示す。

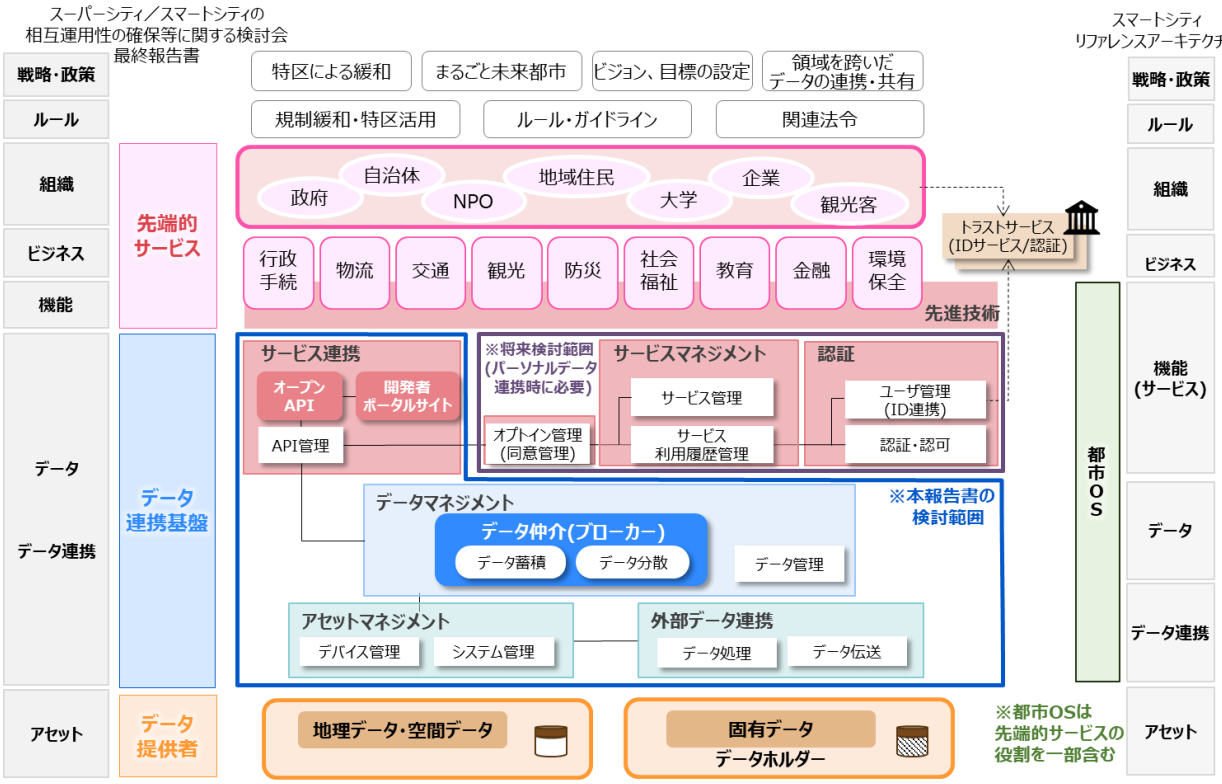


図 3-2-3-1　データ連携基盤の全体像

データ連携基盤に必要となる機能の一覧について表　3-2-3-1 に示す。各機能ブロックに含まれる個別機能の要件については、スマートシティリファレンスアーキテクチャを参照頂きたい。

表 3-2-3-1　データ連携基盤の機能一覧（案）

機能群	機能ブロック	説明	必須	推奨
サービス連携	オープン API	データ連携基盤が提供する API を、他の主体が利用するために接続仕様を外部公開する	○	
	開発者ポータルサイト	利用者向けに、API やデータの検索・仕様の開示が可能なカタログ機能、及び、API を評価可能なコンソール機能等を提供する	○	
	API 管理	データ連携基盤が提供・連携する各種機能を API として管理・公開する		○
	オプトイン管理（同意管	利用者が同意の上で、サービス提供者やデータ提供		○

機能群	機能ブロック	説明	必須	推奨
	理）	者に、パーソナルデータの公開範囲を指定する		
サービス マネジメント	サービス管理	都市 OS と連携するサービスを管理・公開する		○
	利用履歴管理	利用者の同意の上で、利用者に紐づくサービスやデータの連携状況を管理する		○
認証	認証・認可（認証連携）	連携する複数のサービスに対する利用者の認証を一元的に管理し、認証連携を実現する		○
	ユーザ管理（ID 連携）	データ連携基盤の管理する利用者を一元的に管理する		○
データ マネジメント	データ仲介（ブローカー）	データ連携基盤内外に点在するデータの所在情報を管理することで、データアクセスを仲介する	○	
	データ管理	サービス連携や、外部データ連携を通じて収集したデータを、分類や形式を問わず保存・蓄積する		○
アセット マネジメント	デバイス管理	データ連携基盤に接続する IoT デバイス等の状態を管理・監視し、システム管理者がデバイスの接続異常等を検出可能にする		○
	システム管理	データ連携基盤に接続する他システムの認証情報、接続情報、契約情報等の他システムに接続するために必要な情報を管理する		○
外部データ連携	データ処理	スマートシティアセットや他システムから収集したデータの差異を吸収し、標準データモデル等へ変換する		○
	データ伝送	スマートシティアセットや他システムと接続する際に、接続プロトコルの差異を吸収する		○

本報告書では、データ連携基盤を整備するにあたり、必須機能となる「データ仲介（ブローカー）」、「オープン API」、「開発者ポータルサイト」、について、それぞれ、4 調査結果（ア）データ仲介機能、5 調査結果（イ）①API 共通ルール/標準仕様、6 調査結果（イ）②API カタログ/開発者ポータルに仕様案を提示している。

また、パーソナルデータの連携に必要となる事項（オプトイン管理、ユーザ管理等）や、都市間連携のために必要となる事項（各都市のデータ所在管理、都市間の共通ルール、ユニークな ID 管理等）については、相互運用性の確保のため、今後検討が必要になると考える。

4. 調査結果（ア）データ仲介機能

4-1. ブローカー調査

汎用的かつ共通的なブローカー仕様の整備に向け、先端的サービスのユースケース、各種団体が公開するリファレンスモデルや製品事例等を調査し、ユースケースと技術動向の両者を踏まえてブローカーが備えるべき機能を整理した。

本調査の中で、ユースケースや、リファレンスモデル・製品事例が取り扱うデータについては、スマートシティリファレンスアーキテクチャの表 7.1.2 にて定義されているデータ種別ごとに整理を行う。以下、本調査で取り扱うデータ種別の説明を示す。

- メタデータ

データ本体を効率的に管理したり検索したりするために、データ本体についてのデータモデル（データ項目や形式）や属性情報等を記述した付帯データ。

- 静的データ

更新頻度が比較的少なく、長期間保存、参照されるデータ。データ形式としてはテキスト形式やバイナリデータ等が挙げられる。

- 動的データ

更新頻度が高く、リアルタイムに生成される時間軸に連続したデータ。データ形式としてはテキスト形式や動画（ストリーミング）データ等が挙げられる。

- 地理空間データ

空間上の特定の地点又は区域の位置に関する情報（位置情報）を持つデータ。別紙ウ-3 の地理空間情報の既存企画調査及び既存データモデル調査報告書にて、地理空間情報の定義をしているが、本調査では、地図や 3 次元建物情報、3 次元土木構築物などデータ形式がベクトル形式やラスター形式で表すものを地理空間データと定義する。それ以外のテキスト形式やバイナリデータとして表現されるものは静的データとして検討する。

- パーソナルデータ

個人情報や、個人情報との境界が曖昧なものを含み、個人と関係性が見出される広範囲のデータ。スマートシティリファレンスアーキテクチャでは加工された情報等個人と関係性が見出される広範囲のデータと定義されているが、本調査では個人情報を含むものをパーソナルデータと定義する。

4-1-1. ユースケース調査

ユースケース調査では、データ連携基盤を利用するサービス側からの視点で、先端的な技術の活用、都市への普及、データ利活用の促進等の観点で、幅広くユースケースを調査し、汎用性の高いブローカー仕様案の検討、蓄積すべきデータの検討等に活用する。

(1)調査対象

調査にあたり、先端的なユースケースのうち、複数のデータを組み合わせて実現、かつ都市への具体的な適用事例が見られる 5 事例を選定し詳細調査を行った。

表 4-1-1-1 に調査対象のユースケースと、その理由を記載する。

表 4-1-1-1　ユースケース調査対象と選定理由

No.	ユースケース名	国・地域	選定理由
1	リスボン・インテリジェント・マネジメント・プラットフォーム	ポルトガル リスボン市	既存システムやセンサーで収集した複数分野のデータをプラットフォームで統合管理し、そのデータを組み合わせて住民サービスを提供するなど、市民の生活向上に向けて取り組み、都市への普及が進んでいる先端的な事例のため。
2	oneTRANSPORT	イギリス	IoT データや交通サービスの複数データをマーケットプレイスにて公開し、全国的にアクセスできるようにしたことで、データ利活用を促進している先端的な事例のため。
3	CityPulse	欧州	既存のシステムやセンサーから収集した複数分野のデータ、関連する SNS 上の投稿内容のデータを集約し、公開することで、データ利活用を促進している先端的な事例のため。
4	杭州 ET City Brain	中国　杭州市	既存システムやセンサーで収集した複数分野のデータを統合し、分析することにより、現状を把握し、信号の制御を行うなど、先端的な技術を活用した事例のため。
5	Neighborhoodscout	米国全土	不動産や教育など複数分野のデータを集約し、基盤内に具備する分析機能での分析した結果を犯罪リスクの予測など異分野で活用したりしている、先端的な技術を活用した事例のため。

(2)調査結果

各事例の調査結果を表 2-2-2-1 にて整理した 9 つの調査観点に沿って、表 4-1-1-2 に記載する。

(ｱ) 地域課題（どのような地域課題を解決するか）

(ｲ) ユースケース概要（どのようなユースケース概要やシステム概要か）

(ｳ) 先進技術（どのような先進技術を使っているか）

(ｴ) 汎用性（汎用的な仕様であるか）

(ｵ) 連携データ（どの分野の誰の何のデータを連携するか）

(ｶ) データ属性（どのようなデータ種別、サイズ、更新頻度が想定されるか）

(ｷ) サービス連携要件（サービス連携時、どの程度の正確性・リアルタイム性・ユーザ数が想定されるか）

(ｸ) 既存システムとの連携（既存システムと連携しているか）

(ｹ) パーソナルデータを取り扱う事例（パーソナルデータを取り扱う事例はあるか）

表 4-1-1-2　ユースケース調査結果の概要

事例 No.	1	2	3	4	5
ユースケース名	リスボン・インテリジェント・マネジメント・プラットフォーム	oneTRANSPORT	CityPulse	杭州 ET City Brain	Neighborhoodscout
(ｱ)地域課題	市民の生活の質	交通	交通	交通	不動産、移住
(ｲ)ユー	モバイルアプリに	交通状況の改善に	交通渋滞緩和のた	交通問題解決に向	移住検討の際、条件

事例 No.	1	2	3	4	5
スケース概要	よりデータを市民公開。複数分野のデータを統合オペレーションセンターで管理。	向け、駐車場データや交通情報により、駐交通整理や最適経路誘導アプリを提供。	めに複数のデータを組み合わせ、ユーザへの経路案内・駐車場案内のアプリを提供。	け、交通カメラ等の複数データの分析により混雑状況を把握し、信号の最適化	に合ったエリア検索。 公立校ランキングや各地域の不動産価値トレンド予測を提供。
(ｳ)先進技術	過去の交通量データと気象やイベントのデータとの相関関係を分析し、交通渋滞予測	交通カメラによってナンバープレート自動認識し、交通量を解析	SNS の生データを認識、解釈し、相互運用性を持った形に変換	監視カメラの映像データをリアルタイムに画像認識する顔認証システムを利用	データ作成の際、データを組み合わせ、独自アルゴリズムで分析
(ｴ)汎用性	オープンかつ標準化された認証方式や API、データモデルを採用	標準規格である OneM2M 仕様のアーキテクチャの採用	開発者向けに API をオープンソースとして公開	エリアのみの利用想定のため汎用性は目的でない	基盤内の分析結果のみを提供するため、汎用性は検討していない様子
(ｵ)連携データ	施設の利用統計情報 公共バス運行情報 監視カメラの映像データ 河川水位センサーからの水位情報 など	駐車場データ 自動車交通量データ 交通量監視カメラ情報 大気環境データ など	自動車交通量データ 駐車場利用状況データ 大気環境データ SNS 上の投稿データ など	人流データ 公共バス位置情報 交差点カメラや監視カメラの映像データ 地図データ など	国勢調査データ FHFA 住宅価格指数 統一テスト結果 地方警察機関の犯罪データ など
(ｶ)データ属性	メタデータ、静的データ、動的データ（更新頻度は 10 分～）、パーソナルデータを取り扱う。	メタデータ、静的データ、動的データ（更新頻度は 1 分～1 日単位）を取り扱う。	メタデータ、静的データ、動的データ（更新頻度は 1 分～数分間）、パーソナルデータを取り扱う。	メタデータ、静的データ、動的データ（更新頻度は数秒～と推察）、地理空間データ、パーソナルデータを取り扱う。	メタデータ、静的データ、地理空間データ、パーソナルデータを取り扱う。リアルタイムデータの活用はなし
(ｷ)サービス連携要件	市民向けにアプリを公開。現時点の情報を数秒以内（推察）に取得	市民向けにアプリ公開。現時点や現在位置から必要情報を数秒以内（推察）に取得	市民向けにアプリ公開。現時点や現在位置から必要情報を数秒以内（推察）に取得	アプリ未公開だが、アプリ側で異常検知時、必要情報を数秒以内（推察）に取得し分析	Web サイトを公開。複数条件により必要情報を数秒以内（推察）に取得
(ｸ)既存システムとの連携	既存システムからのデータを変換し基盤に統合したり、システムから基盤へ通知。 アプリケーション	• 既存システムのデータを oneM2M やファイル形式で取得。 • アプリケーション側からシステム	• 既存システムからのデータを基盤にて分析。 • アプリケーション側からシステムのデータを参	• 既存システムから取得したデータを基盤に蓄積。 • アプリケーションから、既存システムを制御	• 既存システムからのデータをデータ連携基盤に蓄積し、分析。

事例 No.	1	2	3	4	5
	側からシステムのデータを参照	のデータを参照	照。		
(ケ)パーソナルデータを取り扱う事例	基盤へ匿名化された形で蓄積しているが、サービス側で位置情報と個人を結び付け、管理している	そのような事例はなし	基盤にて匿名化をした形に変換し利用するため、そのような事例はなし	基盤に蓄積した個人情報を複数のサービスにて利用	基盤にて匿名化をした形に変換し利用するため、そのような事例はなし

ユースケース調査結果の詳細については、「別紙：ア-1_調査結果詳細」1.（ア）ユースケース調査結果詳細 を参照。

4-1-2. リファレンスモデル・製品事例調査

リファレンスモデル・製品事例調査では、データ連携基盤自体の観点で、類似する事例からデータ連携基盤やブローカーの目指すべき姿を検討するために、各種団体が公開するリファレンスモデルや製品事例について国内外の取り組みを調査した。

(1)調査対象

調査対象としては、はじめに国際的に相互運用を指向し、グローバルコミュニティで仕様を議論する標準プラットフォーム化団体のリファレンスモデルや製品事例が挙げられる。これらはフレームワークの色彩が強いものの、多面的な機能の実現に向けた検討を含んでいる。また、民間企業が独自に開発・供給している産業系・情報通信系のプラットフォームは、データ仲介を含む、より大きなフレームワークとして製品を提供している製品事例があるので、これらについても調査を行う。

上述より、データ仲介機能と思われる IoT プラットフォームを以下のようにタイプ分けを行い、各事例を整理した。

① 標準プラットフォーム
- 相互運用性や政府・自治体のニーズの観点から、国際的コミュニティにおいて要件が策定され、設計開発されたデジュール標準的な IoT プラットフォーム。

② 情報通信系プラットフォーム
- IT ベンダや通信事業者が IoT 製品を組み合わせて産業用ソリューションとして製品化したもののうち、行政や自治体をターゲットとした IoT プラットフォーム。

③ 産業系プラットフォーム
- ビル、鉄道、電気ガス、その他プラントや設備機器のメーカーが顧客サービスの一環として開発した、クラウドベースで都市全体への適用も行うことが可能な IoT プラットフォーム。

このうち、直近 3 年（2018～2020 年）で新しい機能や具体的な都市への適用事例が見られ、かつデータ仲介機能に該当する技術情報が開示されているもの 5 種に絞ってリファレンスモデル・製品事例調査を行った。以下、表 4-1-2-1 に、調査対象のリファレンスモデル・製品事例と、その理由を記載する。

表 4-1-2-1　リファレンスモデル・製品事例の調査対象と選定理由

No.	分類	リファレンスモデル・製品名	公表団体名/企業名	選定理由
1	①	FIWARE	FI-PPP（欧州）	分野横断データ利活用を実現するオープンプラットフォームの製品事例で、分散型のアーキテクチャを採用している点がスーパーシティにおけるデータ連携基盤の構想と合致しているため。
2	①	X-Road	エストニア政府	電子国家エストニアで開発された分散型のデータ連携プラットフォームであり、同国の電子行政サービス等において社会実装され、活用されているため。
3	①	IES City	NIST（米国国立標準技術研究所）	大規模かつ複雑なシステムのマネジメント性を考慮するために必要な相互運用性の確保に向けた要求事項をまとめたリファレンスモデルの事例で、データ連携基盤において備えるべき機能として参考になるため。

No.	分類	リファレンスモデル・製品名	公表団体名/企業名	選定理由
4	②	Kinetic for City	Cisco	都市課題の解決に使われる IoT 機器やデータの収集や加工に特化した基盤の製品事例で、スマートシティ向けに開発され、実際に導入までされているため。
5	③	MindSphere	Siemens	民間企業が独自に顧客サービスの一環として開発したデータプラットフォームの製品事例で、クラウドベースで、都市全体への適用も行うことが可能なため

(2)調査結果

各事例の調査結果を表 2-2-2-2 にて整理した 7 つの調査観点に沿って、表 4-1-2-2 に記載する。

(ｱ)概要（製品の概要）

(ｲ)複数データソースへの透過的なアクセス方式（どのような方法でアクセスを実現しているか）

(ｳ)汎用性（認証 API、データマネジメント API、データモデル、標準規格への適合性）

(ｴ)取り扱うデータ（ブローカーに蓄積するデータ、蓄積目的及び蓄積方式）

(ｵ)既存システムとの連携（既存システムとの連携方式、データ変換方式）

(ｶ)サポート体制（サポート体制・開発者コミュニティ体制等、運用機能）

(ｷ)プライバシー対策・セキュリティ対策（ブローカーにおけるプライバシー対策・セキュリティ対策はどうなっているか）

表 4-1-2-2　リファレンスモデル・製品事例の調査結果　概要

No.	1	2	3	4	5
リファレンスモデル・製品名	FIWARE	X-Road	IES City	Kinetic for City	MindSphere
(ｱ)概要	EU の次世代インターネット官民連携プログラム(FI-PPP)で開発・実装された基盤ソフトウェア。	エストニアで開発され、同国の電子行政サービス等に活用されているデータ連携基盤。	NIST が中心となり、相互運用可能なスマートシティの要件を示したフレームワーク。	Cisco がスマートシティ向けに開発・提供するクラウドベースの IoT プラットフォーム。	Siemens 社が顧客サービスの一環として開発・提供するクラウドベースの IoT プラットフォーム。
(ｲ)複数データソースへの透過的なアクセス方式	• データの所在管理により要求を所在元へ転送することで分散したデータへ透過的なアクセスが可能。 • Pub/Sub 機能に	データ利用者とデータ提供者間のデータ連携をセキュリティサーバを介して当事者間で直接連携する。	データ融合の概念に基づく同期/非同期アクセスの技術的論点を整理しており、パターンの1つとしてイベントドリブン方式(Pub/Sub)を示す。	サウスバウンド[10]側にて、データ転送の仕組みとしてイベントドリブン方式 (Pub/Sub) を提供。	Siemens 専用の通信機能やカスタムアプリケーションを経由して、複数のデータソースへのアクセスを実現。

[10] サウスバウンド（Southbound）：コントロール層と物理ネットワークハードウェア（デバイスなど）間の接続

No.	1	2	3	4	5
	より、FIWARE 内外のシステムへ更新内容を転送可能。				
(ウ)汎用性	標準API、標準的なデータモデルや認証方式、ビルディングブロック構造による汎用性への対応。	データ連携にはSOAP及びRESTを提供。セキュリティサーバ間通信は TLS によって相互認証を行う。	外部との連携は、標準規格の考慮を促す。データモデルに関する仕様はないがモデリングの重要性を指摘。	標準 API と REST API に対応、標準データモデルを定義し、公開。	外部 API としては標準的なものに対応、標準的な認証方式を利用。
(エ)取り扱うデータ	データ同士を関連付けるコンテキスト情報として、メタデータ、静的データ、動的データ、パーソナルデータを取り扱う。	X-Road 自体には特定のデータタイプを扱う機能はない。	取り決めは特になく、アプリケーションでの利用例として、分野ごとにデータ利用例を紹介。	機器や設備からのデータが主な対象でメタデータ、静的データ、動的データを取り扱う。パーソナルデータ活用の想定はなし。	データ種別やサイズの制約はない。メタデータ、静的データ、動的データの取り扱い可能。パーソナルデータは匿名化が必要。
(オ)既存システムとの連携	標準API に加え、プロトコル変換等の既存システムとの連携機能が定義。	SOAP、REST サービスを提供。セキュリティサーバを介して既存システムと連携可能。	ノースバウンド[11]側はREST、SOAP などが推奨。サウスバウンド側はイベントドリブン方式、データ交換機能と Pub/Sub 機能モデルを推奨。	外部連携はREST が提示。アプリケーションは外部パートナー経由で提供される様子。	同期型通信、メッセージキューイングを使った非同期通信などの複数の連携機能を提供。
(カ)サポート体制	OSS として公開されており、ライセンスに従い無償で利用可能。	MIT ライセンスのOSS として公開。ベンダカスタムされた製品も提供されている。	フレームワークの取り決めなので、スコープ外。	SaaS パッケージで販売され、有料。またアプリ保守には運用が必要。	無料で開始可能。初期投資を少なくし、運用コストに転嫁する課金方式を採用。
(キ)プライバシー対策・セキュリティ対策	パーソナルデータの利用は本人同意を得るプロセス等が必要。本人同意等の個人情報取扱に特化した機能は存在なし。	セキュリティサーバを介した通信によって通信の暗号化とタイムスタンプ付与が行われる。	匿名化の必要性について言及。データ保護、プライバシーについての考慮点の整理。	センサーやデバイス等からのデータを前提としており、パーソナルデータを取り扱う場合独自機能はなし。	パーソナルデータは匿名化して利用する。また、顧客要請に基づき、削除可能。

リファレンスモデル・製品事例調査の結果詳細については、「別紙：ア-1_調査結果詳細」2.（ア）リファレンスモデル・製品事例調査結果詳細 を参照。

[11] ノースバウンド（Northbound）：コントロール層とアプリケーション間の接続

4-1-3. 考察

(1)調査結果の考察

ユースケース（3-1、4-1-1)、リファレンスモデル・製品事例（4-1-2）の調査結果を基に、ブローカーとしての想定課題に対する解決策を検討する。

ユースケース：ユースケースを実装するために必要な機能・非機能という観点で解決策を検討

リファレンスモデル・製品事例：分野間データ連携の実現方式という観点で解決策を検討

はじめに、本調査ではデータ連携において、データ連携基盤からデータ利用者側を Northbound、データ連携基盤からデータ提供者側を Southbound と分けてまとめられていることより、データ連携基盤におけるデータ連携の目的が「データ利活用」と「データ収集」の 2 側面があり、要件が異なることがわかった。ブローカーにおいても、求められる機能が異なることより、データ連携目的を分けて整理する。「データ利活用」を目的とする場面とは、データ利用者が必要なタイミングでデータを取得するケースである。「データ収集」を目的とする場面とは、データ利用者がデータ発生時にデータを取得するケースである。

ユースケース調査において、「データ利活用」の事例として oneTRANSPORT と Neighborhoodscout、「データ収集」の事例としてリスボン・インテリジェント・マネジメント・プラットフォームと CityPulse と杭州 ET City Brain に分けられる。例えば、「データ利活用」の事例である oneTRANSPORT では、イベント開催時の渋滞など交通の問題を解決するために、マーケットプレイス上に地方交通局や自治体が提供する駐車場利用状況・バス位置情報などのデータを公開し、これらのデータを利用するアプリケーションで公共交通機関の利用を促進していた。また、「データ収集」の事例であるリスボン・インテリジェント・マネジメント・プラットフォームでは、既存システムや民間事業者のサービス等の 40 以上のシステム、また市が保有する総数 5,000 以上のセンサーや、監視カメラ 200 台から随時データを取得しており、このように多くのデータ中から、河川水位センサーは閾値を超えたタイミングでアラートを出して、防災活動に活用していた。

リファレンスモデル・製品事例調査の API 仕様において、FIWARE と IES City と Kinetic for City と MindSphere では、データ利活用を目的とする API と、センサーなどからデータ収集を目的する API を分けて整理していることがわかった。例えば、IES City では、ノースバウンド（Northbound）側とサウスバウンド（Southbound）側では異なる利用者（開発者等）向けに異なるインターフェース群が定められるという状況が想定されていた。なお、X-Road では、データ利用者がデータを取得する際に、システム間をセキュアに繋ぎ、接続先システムから提供される API でデータを連携することが特徴であった。この調査結果から、X-Road が「データ利活用」に特化した基盤であると言える。

下記に、2-2-1 で挙げた「1.データ分析」から「7.パーソナルデータの利活用」までの想定課題の枠組みに沿ってブローカーとしての解決策を記載する。

表 4-1-3-1　調査結果からの想定課題解決策

No.	課題	説明	調査結果からの想定課題の解決策
1	データ分析	データを一元的に蓄積しないデータ分散型のシステムであっても、接続するシステム全体を 1 つのビッグデータとみなし AI 分析等を活用するためにはどういった方法があるか。	• データ利活用 データ利用者からの参照リクエストをデータ提供者へ転送する(リクエスト転送）方法が必要。 • データ収集 データ提供者が送信したデータを必要としているデータ利用者へ通知する方法（Pub/Sub）が必要。
2	汎用性	データ分散型を前提にしつつ、国際的にも汎用性が高いデータ連携基盤を作るにはどのようなモデルとすればよいか。	• データ利活用 REST などの標準的な API の利用を推奨。 • データ収集 既存アセットとの連携を考慮し､多様な接続方式をサポートすることを推奨。 • データ利活用/データ収集 標準的なデータモデルや認証方式の利用を推奨
3	蓄積すべきデータ	データ連携に係る接続の正確性とスピードを重視するため、データ連携基盤に最低限蓄積すべき静的データ（地図情報等）はどういったものがあるか。	調査では取り扱うデータに関してデータ種別ごとに調査 • メタデータ ブローカーではなく､データ連携基盤のデータカタログなどの機能にて管理すべき。 • 静的データ ブローカーではテキスト形式のデータ取り扱い可能。 • データ連携基盤としては、テキスト形式、地理空間データ、バイナリデータを取り扱えるべき。 • 動的データ ブローカーではテキスト形式のデータ取り扱い可能。 • データ連携基盤としては、テキスト形式、動画データを取り扱えるべき。 • パーソナルデータ データ連携基盤として取り扱えるべきであるが､ブローカーに管理機能等を具備する必要はない。取り扱う際は、リスクアセスメント及びデータのセキュリティ対策が必要。
4	既存システムとの連携	交通、行政等の既存の稼働中のシステムに接続する場合、これらのシステム及びデータ連携基盤そのものに大きな影響や負担を与えることなく正確にデータ連携するには、どういった方法があるか。	• データ利活用 データ連携基盤と接続するシステムに対して、標準的な API を提供し、メッセージ（処理・手続き）を要求する方法が必要。 • データ収集 データ発生時に利用者へ通知するイベントドリブン方式が必要。 • データ利活用/データ収集 システムとデータ連携基盤のインターフェース相互変換が必要。
5	コスト	運用性・保守性等を加味した上で、ランニングコストを低減させるためにはどういった方法があるか。	ライセンス費用が無償、カスタマイズの自由度、情報の検索性などの点で OSS 利用によるコストメリットがある。ただし OSS は無保証であるため、社会実装として安定運用を実現するためには保守体制の整備又はベンダサポート等の考慮が必要。

No.	課題	説明	調査結果からの想定課題の解決策
6	スケーラビリティ	都市間連携・分野間連携の連携数が将来的に増加した場合においても、リアルタイム性等のブローカー性能を維持可能なように拡張性を保つにはどのような方法があるか。	調査結果からの解決策はなし。 4-2 にて検討する。
7	パーソナルデータの利活用	地域課題を解決するために、プライバシーリスクを軽減しつつ必要最小限のデータ連携のもとで最大限の価値を発揮するためにはどのような方法があるか。	パーソナルデータを取り扱う場合は、データ連携基盤にて、本人同意管理や匿名化等の機能が必要。また信頼されたサービス間のみでのセキュアなデータ交換するなど、セキュリティ対策を実施した上で利用が必要。

下記、調査結果からの想定課題解決策の詳細を示す。

(a)データ分析

「データ利活用」において、あるサービスを実現するために、ブローカーを介して、外部サービスからデータを取得する機能が必要であることが判明した。

- ユースケース調査では、異なるシステムやサービスなど物理的に分散されているデータを取得して、別のサービスでデータを利活用する事例があった。例えば、oneTRANSPORT の事例では、交通に関するシステムやサービスからのデータをマーケットプレイス上にて公開し、データ提供者は複数のデータを組み合わせ、アプリケーションにて利用している。また、Neighborhoodscout は、不動産や教育などの複数分野のデータをデータ連携基盤に蓄積し、エリアの不動産価値や学校の質などの意味のある形に分析し、Web サイトにて利用者へ提供している。
- リファレンスモデル・製品事例調査では、データ利用者からのデータ要求に対して、データ提供者からのデータ返却を同期処理で行う事例があった。例えば、FIWARE の場合は、データの所在情報をあらかじめ登録しておくことで要求を所在元へ転送する機能があり、様々なデータへ同一インターフェースでアクセスすることができる。ブローカーにて、データ所在情報を管理することで、データ利用者に対してデータの所在を隠蔽することができる。

上述より、データ連携基盤の外部サービスが保持するデータから必要なデータを必要なタイミングで取得する機能が考えられる。これを実現するために、ブローカーとしては、データ利用者からの参照リクエストをデータ提供者へ転送する（リクエスト転送）機能が必要である。

「データ収集」において、センサーなどのデバイスから大量に発生するデータに対し、あるサービスが必要とするデータをブローカーを介して抽出する機能が必要であることが判明した。

- ユースケース調査では、大量のセンサーからの情報から異常の検知や履歴の蓄積を行う事例があった。例えば、リスボン・インテリジェント・マネジメント・プラットフォームでは、河川水位センサーから随時データを取得しており、閾値を超えたタイミングでアラートを発している。
- リファレンスモデル・製品事例調査では、受け取ったデータを他のシステムやデータ連携基盤内の別機能へ送信・通知を非同期処理で行う事例があった。例えば、IES City では相互連携において想定すべき一般的なデータ連携パターンのモデルとして、データの受信登録（Subscribe）をすることでデータ利用者へ更新を通知（Publish）するイベントドリブン方式

(Pub/Sub)が含まれている。また、Cisco Kinetic for City でも同様のパターンを提供している。

上述より、必要とする相手へデータの更新を通知する機能が必要と考えられる。これを実現するために、ブローカーとしては、データ提供者が送信したデータを必要としているデータ利用者へ通知する（Pub/Sub）機能が必要である。

(b)汎用性

「データ利活用」における API としては、調査では標準的な API を利用する事例が複数あった。

- リファレンスモデル・製品事例調査では、例えば FIWARE では標準規格である NGSI を利用しているが、これは REST API ベースとしている。また X-Road は SOAP や REST をプロトコルと定義している。

上述より、データを扱う際の利便性を考慮し、標準ルールに沿った API（REST 等）の提供が必要と考えられる。

「データ収集」における API としては、調査では大量の多種多様なセンサーからデータを取得する事例があった。

- ユースケース調査では、例えば、リスボン・インテリジェント・マネジメント・プラットフォームでは 5,000 以上の様々なセンサーから随時データを取得しており、このような要件に耐え得る必要があると考えられる。
- リファレンスモデル・製品事例調査では、多様な接続方式をサポートする事例が複数あった。例えば、Cisco Kinetic for City ではモジュール内でデータキャプチャ・変換や前処理や転送ルール適用、確実なデータ配信、履歴データ管理等を行う機能を持っており、これにより、多様で分散したデバイスからデータを取得することができる。

上述より、各種アセットとの接続性や高トラフィックのリクエストを処理する仕組みの提供が必要と考えられる。

また、標準的なデータモデルや認証方式を利用している事例があった。

- データモデルに関しては、リファレンスモデル・製品事例調査において、IES City では相互運用性や標準を意識したモデリングを行うことの重要性が指摘されている。また、Cisco Kinetic for City では対象とする領域において、API として定義されているいくつかの標準データモデルを提供している。
- 認証方式に関しては、リファレンスモデル・製品事例調査において、FIWARE や MindSphere では標準的な認証・認可機能である OAuth2.0 を採用している。

上述より、オープンで標準的なデータモデルや標準的な認証方式の利用を推奨する。ただし、ブローカー機能ではなく、データ連携基盤全体での検討が必要である。

(c)蓄積すべきデータ

データ蓄積に関しては、表 2-2-1-2、表 2-2-1-3 にて定義したとおり、データ連携基盤にて取り扱えるデータや、ブローカーにて仲介するデータという観点で調査を行った。また、調査を行うにあたり、4-1 にて定義したデータ種別ごとに調査結果を整理した。

- メタデータ

データセットごとに管理されている事例が複数あった。

- ユースケース調査では、どの事例もデータセットごとにメタデータを取り扱い、例えばoneTRANSPORTは、サービスポータルにて登録や管理ができる。
- リファレンスモデル・製品事例調査では、メタデータの管理機能がある事例があり、例えばX-Roadの場合は、セントラルサーバにて利用者の情報やセキュリティサーバの管理などの管理を行っている。

上述より、データ連携基盤にてメタデータを管理する機能は必要であるが、ブローカーで仲介するのではなく、データカタログなどの機能にて管理すべきと考える。

● 静的データ

テキスト形式については、どの事例でもデータの取得・参照はAPI経由で可能であった。バイナリデータについては、取り扱うことができるよう整備されている事例があった。

- 例えば、リファレンスモデル・製品事例調査のCisco Kinetic for Cityにおけるカメラ画像などが挙げられる。

上述より、データ連携基盤にて静的データは取り扱えるべきであると考える。重ねて、ブローカーにてテキスト形式での取り扱いは必須である。またバイナリデータも同様に、ブローカーではなく、バイナリデータ管理などの機能にて管理すべきと考える。

● 動的データ

タイムスタンプを付与したテキスト形式の取り扱いが可能な事例が複数あった。

- ユースケース調査において、例えば杭州 ET City Brainの事例ではタイムスタンプを付与した交通量データなどを扱っており、更新頻度は数秒～と推察する。
- リファレンスモデル・製品事例調査においても、MindSphereの事例では、APIを経由して、HTTPベースで、標準データとして取り込むことができる。動画（ストリーム）データを取り扱う事例も複数あり、例えばユースケース調査のリスボン・インテリジェント・マネジメント・プラットフォームや杭州 ET City Brainでは監視カメラの映像データを取り扱っている。またリファレンスモデル・製品事例調査のCisco Kinetic for Cityの事例では、ネットワークなどの関連ソリューションが提供されている。

上述より、データ連携基盤にて動的データは取り扱えるべきであると考える。重ねて、ブローカーにて、テキスト形式でのデータ取り扱いは必須である。動画（ストリーム）データの取り扱いに関しては、ブローカーではなく、VMSなどの機能にて管理すべきと考える。

● 地理空間データ

地理空間データについては、調査よりブローカーに地図情報を配信する機能はないが、地理空間上に描写できる事例が複数あった。

- ユースケース調査において、例えば、リスボン・インテリジェント・マネジメント・プラットフォームでは市民へ向けてもモバイルアプリにて地図を公開している。
- リファレンスモデル・製品事例調査においても、例えばMindSphereではデータを可視化する機能は存在しないが、パートナーのソリューションと連携することで表現可能である。

上述より、ブローカーではなく、GISなどの機能にて管理すべきと考える。

また地理空間データには、データ形式がベクトル形式やラスター形式で表すものと大きく2つに分かれるが、どちらもバイナリデータで表現される。よってデータ種別としては、静的データに含めて検討する。

● パーソナルデータ

調査から、パーソナルデータを扱う事例と扱わない事例の 2 つに大きく分かれたが、地域課題の解決のために住民に寄り添ったサービスなどを提供するためにはパーソナルデータを取り扱うべきであると考える。

- パーソナルデータを取り扱う事例は、ユースケース調査の City Pulse や Neighborhoodscout であり、データ連携基盤にて加工して、匿名化後に利用者へ提供する。また、杭州 ET City Brain ではオンプレミスのクラウド上に構築し公開していない。リファレンスモデル・製品事例調査からは、例えば IES City では匿名化の必要性について言及されている。
- 「スマートシティリファレンスアーキテクチャ」ではパーソナルデータの取り扱いに関するルールに従い、プライバシー保護と高度なセキュリティ対策が必要であると示されており、取り扱う場合はリスクアセスメント及びデータのセキュリティ対策が必要と考える。

上述より、データ連携基盤にてパーソナルデータの取り扱いは必須で、リスクアセスメント及びデータのセキュリティ対策が必要であると考える。ただし、ブローカーで管理機能などを取り扱う必要はないと考える。

以上、データ種別ごとにデータ連携基盤やブローカーにて取り扱うべきかの観点で整理をしたが、下記にてデータを蓄積するかの観点に関して考察する。

調査から、河川水位データ等のように継続的に観測することに価値があるデータを取り扱う事例があった。

- 例えば、ユースケース調査のリスボン・インテリジェント・マネジメント・プラットフォームでは、河川水位センサーから定期的に水位データを取得し、ダッシュボードで河川の水位状況等を表示している。

上述より、継続的に観測することに価値があるデータの取り扱いが考えられる。また。これらのデータは、様々なデータ利用者が参照することを考慮すると、データ連携基盤に蓄積する方が良い場合が考えられる。

ただし、これはデータをデータ連携基盤内に蓄積したいかどうかという観点によって決まるとし、データ連携基盤を導入するエリアや地域での考え方や方針によって異なるものと考える。

(d)既存システムとの連携

下記 3 つの機能が考えられる。

「データ利活用」において、サービス利用を想定した API を提供する事例があった。

- ユースケース調査では、データ利用者側から、接続するシステムに対して、データ参照などの処理を要求する事例があった。例えば、oneTRANSPORT ではアプリケーション側から、周辺の交通情報検索があった際、API を介して、地方当局からの交通量データ、自治体が管理している駐車場の利用状況データなどの既存システムからのデータを参照していると推察する。
- リファレンスモデル・製品事例調査では、既存システムとの連携の際の標準 API が提供されている事例があった。例えば IES City の場合、データ融合における異種システムからのデータ

収集に関連する標準としてSOAP、また他システムとはRESTといった標準的なプロトコルの採用を前提としている。

上述より、データ連携基盤と接続するシステムやサービスへの機能を呼び出すために、メッセージ（処理・手続き）を要求する機能が考えられる。

「データ収集」においては、既存システムとの連携の場合も、必要とする相手へデータの更新を通知する機能が必要と考えられる。これを実現するために、ブローカーとしては、データ提供者が送信したデータを必要としているデータ利用者へ通知する（Pub/Sub）機能が必要である。

また、「データ利活用」では様々なサービスのデータに対して、「データ収集」ではセンサーなど多様なデバイスから発生するデータに対して、ブローカーを介してデータを仲介する必要がある。

- ユースケース調査では、「データ利活用」のため、異なる主体が管理する様々なデータからデータを取得し、別のサービスで利用する事例があった。例えば、oneTRANSPORTでは、サービスにてデータを利用する際、マーケットプレイスを介してデータ提供者からデータを取得している。
- リファレンスモデル・製品事例調査では、「データ収集」におけるプロトコル変換や接続のためのコンポーネント等、受け取ったデータを他のシステムやデータ連携基盤内の別機能利用できる仕組みを用意してある事例があった。例えば、FIWAREはスマートシティを含む様々な分野でのデータモデルの相互運用ができるように取得したデータをFIWAREデータモデルに変換する機能を保有している。

上述より、「データ利活用」及び「データ収集」において、様々な主体から提供されるデータをデータ連携基盤で扱うことのできるデータモデルに変換する機能が必要と考えられる。

(e)コスト

リファレンスモデル・製品事例調査において、OSSをカスタマイズなどをして提供するものと、有償でサービスのパッケージを提供するものとの2つに分類される。FIWAREコンポーネントは全てOSSとして公開しており、X-Roadは政府が開発したソフトウェアをOSSとして公開している。それに対して、Cisco Kinetic for CityとMindSphereは有償のライセンスが提供されている。

一般的に、OSSを利用することで、ライセンス費用が無償、カスタマイズの自由度、情報の検索性などコストメリットがある。ただし、OSSは無保証であるため、社会実装として安定運用を実現するためには保守体制の整備又はベンダサポート等の考慮が必要となる。

上述より、OSSを利用する場合、サポート体制が整備されていること、又は開発者コミュニティが活性化されていることで、運用保守の業務に支障を来さないことが必要と考えらえる。緊急時のサポートや脆弱性の対応などを受けるためのサポート体制維持は削減できないコストだと考えられる。

(f)スケーラビリティ

調査対象外のため、検討なし。

(g)パーソナルデータの利活用

データ連携基盤にて、パーソナルデータを加工して匿名化したり、同意管理している事例があった。

- ユースケース調査では、パーソナルデータをデータ連携基盤にて分析し、匿名化した上で利用している。例えば、City Pulseはパーソナルデータに分類されるSNS上の生データを扱っているが、データ連携基盤にて分析し、匿名化した上で、データ提供者側へ提供している。

- リファレンスモデル・製品事例調査において、ブローカー製品には同意管理・匿名化等のパーソナルデータ活用に関する機能を具備している事例はなかった。

上述より、パーソナルデータを利活用する際は、データ連携基盤にて、同意管理、匿名化等の機能が必要である。

また、X-Road はセキュアなデータ交換を実現するために信頼されたサービス間で直接データ連携を行っていることがわかった。

上述より、クローズドなデータ連携基盤を利用する場合は、信頼されたサービス間でのセキュアデータ交換を実現することが必要である。

(2)調査結果からの課題の抽出と改善方針

ブローカーの要件を抽出するために、表 4-1-3-1 の調査結果からの改善策からブローカーで持つべき機能か、データ連携基盤で持つべき機能かの整理を表 4-1-3-2 にて示す。

表 4-1-3-2　想定課題の解決策からの機能分類

No.	課題	調査結果からの想定課題の解決策（再掲）	ブローカー	データ連携基盤
1	データ分析	• データ利活用 データ利用者からの参照リクエストをデータ提供者へ転送する（リクエスト転送）方法が必要。	◎	△
		• データ収集 データ提供者が送信したデータを必要としているデータ利用者へ通知する方法（Pub/Sub）が必要。	◎	△
2	汎用性	• データ利活用 REST などの標準的な API の利用を推奨。	△	◎
		• データ収集 既存アセットとの連携を考慮し、多様な接続方式をサポートすることを推奨。	△	◎
		• データ利活用/データ収集 • 標準的なデータモデルや認証方式の利用を推奨。	△	◎
3	蓄積すべきデータ	• メタデータ	△	◎
		• 静的データ	◎ テキストのみ	○ ※1
		• 動的データ	◎ テキストのみ	○ ※2
		• パーソナルデータ	○	◎
4	既存システムとの連	• データ利活用 • データ連携基盤と接続するシステムに対して、標準的な API を提供し、メッセージ（処理・手続き）を要求する方法が必要。	◎	△

No.	課題	調査結果からの想定課題の解決策（再掲）	ブローカー	データ連携基盤
	携	• データ収集 • データ発生時に利用者へ通知するイベントドリブン方式が必要。	◎	△
		• データ利活用/データ収集 システムとデータ連携基盤のインターフェース相互変換が必要。	○	◎
5	コスト	ライセンス費用が無償、カスタマイズの自由度、情報の検索性などの点で OSS 利用によるコストメリットがある。ただし OSS は無保証であるため、社会実装として安定運用を実現するためには保守体制の整備又はベンダサポート等の考慮が必要。	-	◎
6	スケーラビリティ	調査結果からの解決策はなし。 4-3 にて検討する。	-	-
7	パーソナルデータの利活用	パーソナルデータを取り扱う場合は、データ連携基盤にて、本人同意管理や匿名化等の機能が必要。	△	◎
		信頼されたサービス間のみでのセキュアなデータ交換するなど、セキュリティ対策を実施した上で利用が必要。	○	○

※1　地理空間データ、バイナリデータの取り扱い
※2　動画（ストリーム）データの取り扱い

【凡例】
◎：必要（Must）
○：推奨（Should）
△：可能（Could）
- ：調査結果から判断不可

調査によって抽出された課題を以下に示す。

表 4-1-3-3　調査から抽出された課題及び改善方針

No.	課題	改善方針
1	データを蓄積すべきかどうかは、データ参照時のレスポンスや、データのサイズなどの観点で決まるため、調査からは判断できない。	4-2 のブローカー評価にて検討する。
2	継続的に観測することで価値があるデータについて、様々なデータ利用者から参照できるように敢えてデータ連携基盤内で蓄積したいケースがある。	データをデータ連携基盤内に蓄積するかは「ブローカーに蓄積する価値があるか」で、これはデータ連携基盤を導入するエリアや地域ごとに異なる。よって、エリアや地域ごとの判断とする。
3	パーソナルデータの取り扱いに特化した機能がない。	データ連携基盤にて、同意管理や匿名化する機能が必要であるが、ブローカーとしてはテキスト形式の仲介が必要である。
4	地理空間データの取り扱いに特化した機能がない。	データ連携基盤にて GIS などの機能にて管理すべきであり、ブローカーとしてはテキストやバイナリデータ形式の仲介が必要である。

No.	課題	改善方針
5	メタデータの取り扱いに特化した機能がない。	データ連携基盤として、データカタログなどの機能にて管理すべきであるが、ブローカーとしてはテキスト形式の仲介が必要である。
6	データモデルや、ID 管理・認証はブローカーにて定義するものではない。	標準的なデータモデルや、標準規格へ適合した方式（OAuth 等）を採用するべきであるが、ブローカーだけでなく、データ連携基盤全体に関わるため、3 章で検討する。

4-1-4. ブローカー調査より抽出されたブローカーに求められる要件

ユースケース（3-1、4-1-1）、リファレンスモデル・製品事例（4-1-2）の調査結果を基に、ブローカーに求められる要件を抽出する。

(1)機能要件

ブローカーの機能要件について、表 2-2-1-2 で挙げた下記 4 つの分類で整理する。

表 4-1-4-1　ブローカーの機能分類（再掲）

項番	機能分類	説明
1	データ分散	分野や都市に分散するデータに対し、データを仲介（登録・参照・更新・削除）できること。
2	データ蓄積	データ連携基盤内で管理するデータに対し、データを仲介（登録・参照・更新・削除）できること。
3	イベント処理	データ連携基盤が仲介するデータに対し、事前に定義されたシナリオに従いリアルタイムに処理を実施できること。 （条件に合致したデータ発生の通知、接続先システムの機能の呼び出しなど）
4	その他	上記以外。（データ変換、API 仕様、運用機能、ユーザビリティ等）

(a)データ分散

「データ利活用」において、データ利用者からの参照リクエストをデータ提供者へ転送する（リクエスト転送）機能が必要である。この機能により、データ連携基盤の外部サービスが保持するデータから必要なデータを必要なタイミングで取得できる。

(b)データ蓄積

評価（4-2）にて検討するため、調査では対象外とした。調査では、データ種別ごとにデータ連携基盤での取り扱いやブローカーにてデータ仲介するべきかという観点で整理した。

(c)イベント処理

「データ収集」において、データ提供者が送信したデータを必要としているデータ利用者へ通知する（Pub/Sub）機能が必要である。この機能により、必要とする相手へデータの更新を通知することができる。

「データ利活用」において、データ連携基盤と接続するシステムやサービスの機能を呼び出す（サ

ービス呼び出し）機能が必要である。この機能により、データ連携基盤の外部サービスにおける処理を実行し結果を返却することができる。

(d)その他

「データ利活用」における API としては、データを扱う際の利便性を考慮し、標準ルールに沿った API（REST 等）の提供が必要と考えられる。

「データ収集」における API としては、各種アセットとの接続性や高トラフィックのリクエストを処理する仕組みの提供が必要と考えられる。

「データ利活用」及び「データ収集」において、様々な主体から提供されるデータをデータ連携基盤で扱うことのできるデータモデルに変換する（データ変換）機能が必要と考えられる。

(2)非機能要件

ブローカーの非機能について、表 2-2-1-3 で挙げた非機能要求グレードに分類し、整理する。

ただし、性能・拡張性については、調査の観点から対象外としたため、評価結果から要件を抽出する。

表 4-1-4-2　非機能要求グレード（再掲※一部修正）

項番	非機能要求	説明
1	性能・拡張性	システムの性能、及び将来のシステム拡張に関する要求
2	運用・保守性	システムの運用と保守のサービスに関する要求
3	セキュリティ	情報システムの安全性の確保に関する要求

(a)運用・保守性

サポート体制が整備されていること。OSS を利用する場合、開発者コミュニティが活性化されていることで、運用保守の業務に支障を来さないことが必要である。緊急時のサポートや脆弱性の対応などを受けるためのサポート体制維持は削減できない。

(b)性能・拡張性

評価（4-2）にて検討するため、調査では対象外とした。

(c)セキュリティ

ブローカーとしての要件は抽出されなかった。パーソナルデータを取り扱う際、同意管理・匿名化等のプライバシー保護に関する機能が必要である。ただし、データ連携基盤内のブローカーではない別の機能にて、実現することが妥当である。

(3)ブローカーにて取り扱うデータ種別

調査では、データ連携基盤で取り扱うデータをデータ種別ごとに整理する。

- メタデータ

  ブローカーでは取り扱わない。データ連携基盤におけるデータカタログなどの機能にて、メタデータを管理する。

- 静的データ

  テキストデータを取り扱うことができるべきである。

地理空間データをバイナリデータとして取り扱うことが可能である。地理空間データとしての取り扱いについては、データ連携基盤における GIS などの機能にて管理する。

- 動的データ

  テキストデータを取り扱うことができるべきである。

  動画（ストリーム）データをバイナリデータとして取り扱うことが可能である。動画（ストリーム）データとしての取り扱いについては、データ連携基盤における VMS などの機能にて管理する。

- パーソナルデータ

  パーソナルデータを取り扱うことができるべきである。

4-2. ブローカー評価

本項では、ブローカー要件を抽出するため、ブローカーの評価を実施する。

まず 2-2-3 の方針に従い評価を行うブローカーを選定し、選定された各ブローカーに対し実機での評価（ユースケース評価、性能評価）を実施する。その評価結果を基に、検出した課題や判明した特徴に対する考察を行い、新たなブローカー要件（機能要件、非機能要件）を抽出する。

4-2-1. 評価対象のブローカー選定

(1)選定結果

本項では、一般的なブローカー製品を洗い出し、2-2-3(1)の方針に従い評価に利用するブローカーを選定する。

まず、異なるブローカー技術に関してはブローカーの基本機能であるデータ仲介における「データ分散方式」にフォーカスし、3 パターンに分類した。各パターンの詳細を表 4-2-1-1 に示す。

表 4-2-1-1 データ分散方式のパターン

パターン	方式	説明
パターン A	同期方式 (Pull 型)	データ利用者のリクエストの即座に返信する(同期方式)特徴を持つ。 データ提供者の所在を隠蔽して透過的アクセスできるものや、直接アクセスするものなど製品によって実装の違いがある。
パターン B	非同期方式 (Pull 型)	一般的に市場に存在するブローカー製品の主流な方式(Pub/Sub 型モデル)。データ利用者とデータ提供者のリクエスト/返信のタイミングが異なる(非同期方式)特徴を持つ。 データ利用者のタイミングでデータを取得する。
パターン C	非同期方式 (Push 型)	パターン B と同様に、データ利用者とデータ提供者のリクエスト/返信のタイミングが異なる(非同期方式)特徴を持つ。 データ提供者のタイミングでデータ利用者へ通知を行う。

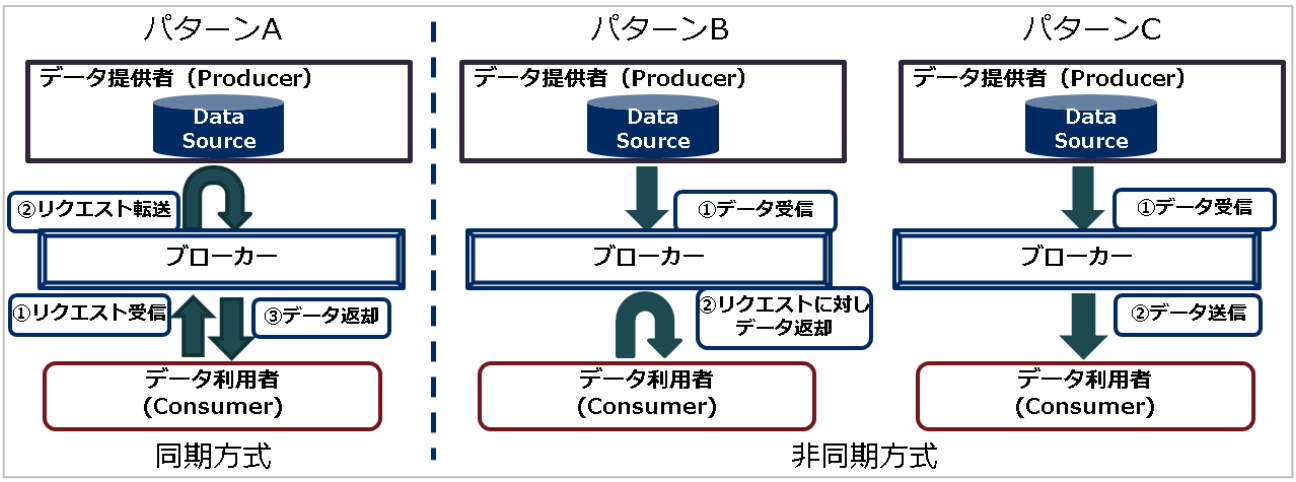


図 4-2-1-1　データ分散方式のパターン図

上記結果に加え 2-2-3(1)の選定方針に従いブローカー製品の中から選定する。
表 4-2-1-2 に一般的なブローカー製品の一覧及び選定方針に対する比較結果を示す。

表 4-2-1-2　代表的なブローカー製品比較

製品/OSS	異なるブローカー技術			想定される機能			運用性	その他特記事項
	パターン			データ分散方式	REST API	柔軟なデータ表現	運用性	
	A	B	C					
FIWARE Orion Context Broker	○	×	○	○	○	○	○	OSS
Scorpio Broker	○	×	○	○	○	○	○	OSS
Apache Kafka	×	○	×	○	△ ※2	○	○	OSS
Mosquitto	×	×	○	○	×	○	○	OSS
RabbitMQ	×	×	○	○	△ ※2	○	○	OSS
Solace PubSub+	×	△ ※1	○	○	△ ※2	○	○	Solace 社製品
WebSphere Message Broker	×	×	○	○	△ ※2	○	○	IBM 社製品

Amazon SQS	×	○	×	○	△ ※2	○	○	Amazon.com 社製品
Amazon SNS	×	×	○	○	△ ※2	○	○	Amazon.com 社製品
X-Road	○	×	×	○	△ ※2	△ ※3	○	OSS

※1 一部対応
※2 標準は非 REST だが、一部機能は REST-API をサポート
※3 WSDL へ都度定義する必要あり

【凡例】
○：対応可能
△：対応可能だが一部制約あり
×：対応不可

上記結果から、評価対象のブローカーを選定する。

まず、今回の調査事業の目的はブローカー技術の比較評価及び課題抽出/改善方針検討であるため、ある程度自由に試行錯誤可能なものが望ましいと考える。そのため評価には OSS として仕様が公開されているものから選定する（ブローカーとして OSS が必ずしも好ましいというわけではない)。

またその他選定方針を考慮し、下記 5 種類が有力候補として挙がった。

- FIWARE Orion Context Broker（以下「FIWARE Orion」と記す）
- Scorpio Broker
- Apache Kafka
- RabbitMQ
- X-Road

上記 5 製品の中で、ブローカー技術の比較のため、仲介パターン A～パターン C を網羅できるようパターンごとにブローカー製品 1 種類を選定することとした。

まずパターン B に対応可能な「Apache Kafka」を選定した。続けて、パターン C としては対応可能な「FIWARE Orion」、「Scorpio Broker」、「RabbitMQ」が挙がったが、REST API への対応可能の観点から「FIWARE Orion」、「Scorpio Broker」が有力となった。「FIWARE Orion」と「Scorpio Broker」はいずれも FIWARE 製品であるが、リファレンスとして将来的に公開することを想定し、実績を考慮して「FIWARE Orion」を選定した。最後にパターン A としては「X-Road」を選定した。

選定した評価対象のブローカーと選定理由を表 4-2-1-3 に示す。

表 4-2-1-3　評価対象のブローカー

製品/OSS 名	選定理由
FIWARE Orion	• データ仲介方式の**パターン A** と**パターン C** として評価可能 • 想定される機能（データ分散方式、REST API、柔軟なデータ表現）を具備 • データブローカーとして国外/国外のスマートシティ案件で多数実績あり
Apache Kafka	• データ仲介方式の**パターン B** として評価可能 • 想定される機能（データ分散方式、REST API、柔軟なデータ表現）を具備 • メッセージブローカーとして、LinkedIn、Twitter 等の様々な大規模サービスで実績あり
X-Road	• データ仲介方式の**パターン A** として評価可能 • 想定される機能（データ分散方式、REST API、柔軟なデータ表現）を具備

製品/OSS 名	選定理由
	• 電子行政分野（エストニア）で実績のあるデータ交換プラットフォームであり、国内でも活用検討が始まっている

(2)製品概要

以下、選定した各製品の概要

(a)FIWARE Orion

FIWARE Orion は FIWARE のコア機能であるコンテキストブローカーに位置する製品である。

主な特長としてはデータ蓄積とデータ分散双方の機能を具備する点が挙げられる。提供される接続インターフェースとして NGSI というオープン API で実装され、オープンソースとして公開されていることから誰にでも利用可能な製品となっている。また、欧州を中心として日本国内でも多数の導入実績を持つブローカーである。

以下に FIWARE Orion のデータフロー図、及び機能を記載する。

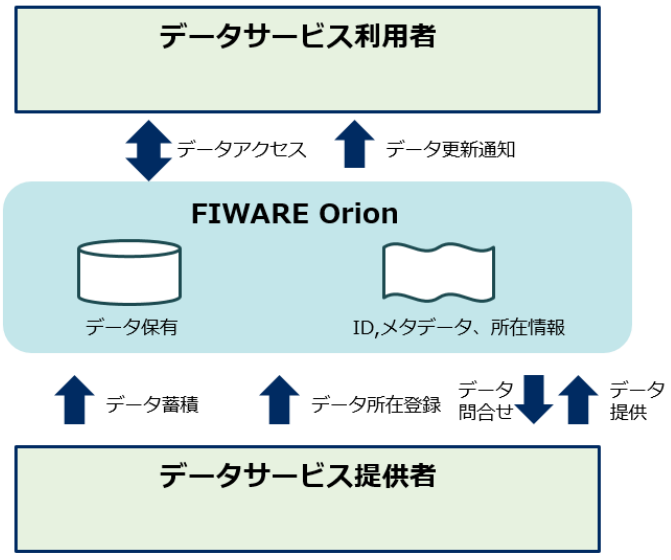


図 4-2-1-2　FIWARE Orion データフロー図

表 4-2-1-4　FIWARE Orion 主要機能一覧

機能名	概要
データ蓄積機能	FIWARE Orion 内にデータを蓄積する機能。蓄積されたデータに対しては RestAPI でアクセスすることが可能。
データ所在管理機能	データそのものではなくデータ提供者が持つデータのアクセス先を管理する機能。データ利用者は蓄積データと同様のアクセス方法でアクセスすることが可能となり、FIWARE Orion は登録された所在情報に従いデータ利用者の代わりにデータ提供者へデータの問合せを行う。（データ仲介）
データ更新通知機能	データ更新が行われた場合にあらかじめ登録されている通知先へデータ更新通知を送信する機能。

(b)Apache Kafka

Apache Kafka は、分散メッセージキュー機能を有するオープンソースソフトウェアである。メッセージキュー機能とは、一時的なデータ保持を行うことで、システム間のデータ受け渡しを仲介する機能である。Apache Kafka はメッセージキューを複数サーバに分散して保持する機能を有し、データ受け渡しの信頼性の向上やデータ送受信性能の向上を実現している。

以下の図に、メッセージキュー機能の概念図を示す。

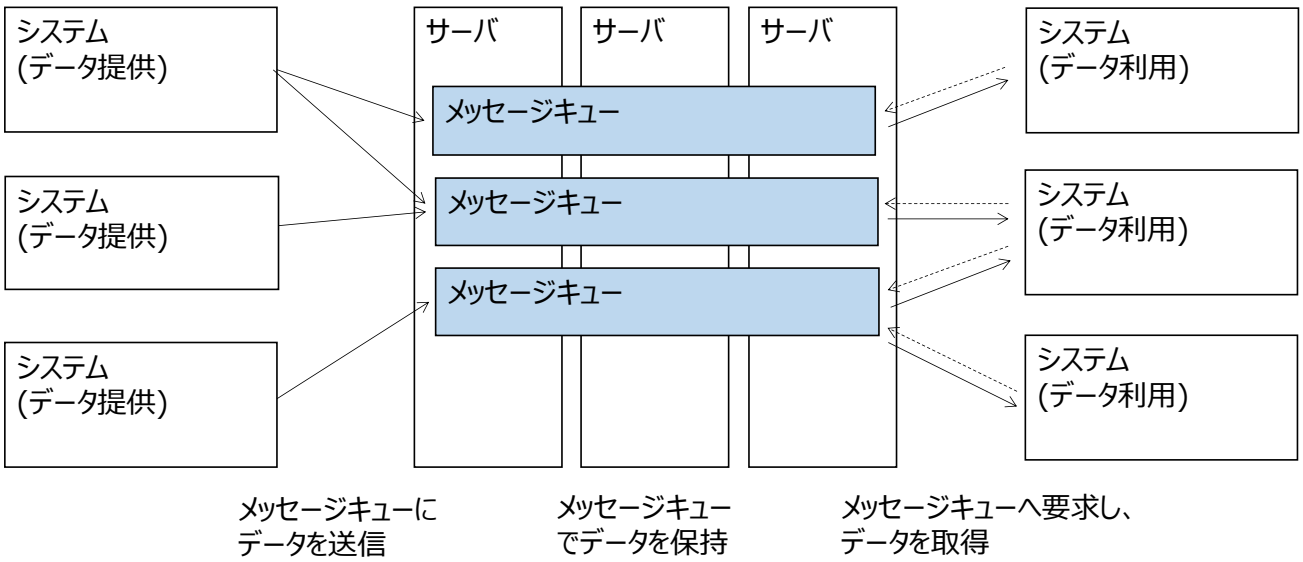


図 4-2-1-3　メッセージキュー機能の概念図

Apache Kafka は民間企業での活用事例が多く、Apache Kafka 自体の開発者コミュニティはもちろんのこと、機能拡張ソフトウェアの開発者コミュニティも盛んに活動している。サードパーティ製ソフトウェアも多数公開されている。Apache Kafka の機能拡張ソフトウェアの 1 つに、Apache Kafka が REST API により json 形式のデータ送受信を行う機能を付加するオープンソースソフトウェアである、Confluent REST Proxy が存在する。

今回は Apache Kafka と Confluent REST Proxy を組み合わせて以下の動作を実現することで、非同期方式（Pull 型）のブローカー評価環境の構築を行い、評価する。

(1)データ提供者が任意のタイミングでメッセージキューに対して連携データを送信する。

(2)メッセージキューにおいて、連携データを一時的に保持する。

(3)データ利用者が任意のタイミングでメッセージキューに対して要求を行い、連携データを取得する。

(c)X-Road[12] [13]

X-Road は、サービス指向アーキテクチャ（SOA）を採用した、データ利用者と提供者の情報システム間のメッセージバスとして動作する、データ交換インフラストラクチャである。X-Road の主要なインスタンス構成について、図 4-2-1-4 に示す。

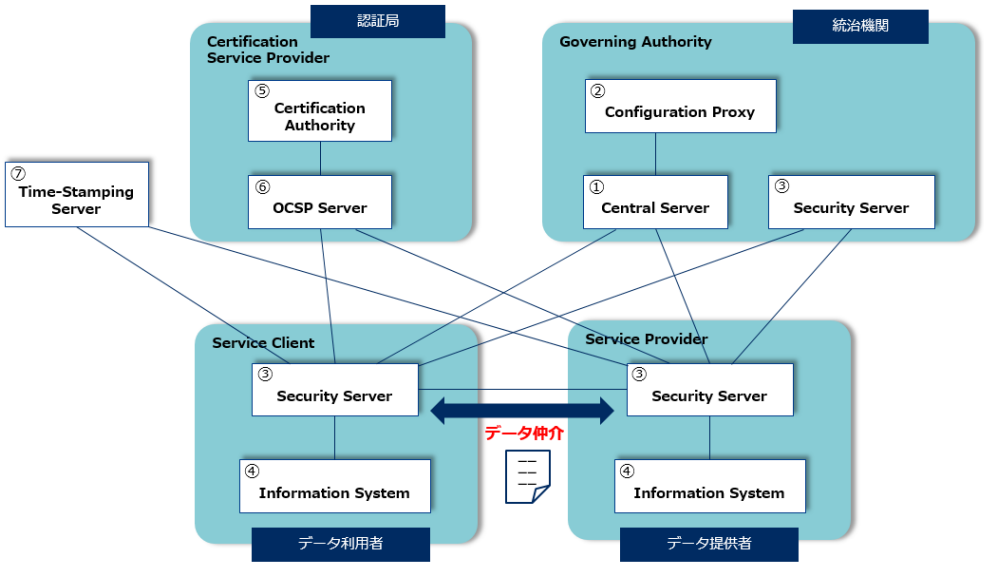


図 4-2-1-4　X-Road のアーキテクチャ（インスタンス構成）

表 4-2-1-5　X-Road のインスタンス

No.	インスタンス名	役割・機能
①	Central Server	X-Road 利用者、及びセキュリティサーバ情報の管理を行う。 また、信頼できる認証局やタイムスタンプ機関のリストなど、X-Road インスタンスのセキュリティポリシーを管理する。
②	Configuration Proxy	セキュリティサーバの情報を構成ファイルとして作成し、各セキュリティサーバに公開する。
③	Security Server	データ利用者とデータ提供者間のサービス呼び出しとサービス応答を仲介する。 情報システムに対して、SOAP 及び REST 用のメッセージプロトコルを提供し、セキュリティサーバ間通信においてセキュリティ面をカプセル化する(※)。 ※ 署名と認証のためのキーの管理、デジタル署名付きのメッセージの証明値の作成、TSL によるセキュリティで保護されたチャンネルを介したメッセージ通信、タイムスタンプ、ロギングなどの機能を有する。また、X-Road は、異なるタイプのメッセージ及びサービス間の自動変換は提供しない。
④	Information System	データ利用者、及びデータ提供者が運用する情報システム。 X-Road を利用するにあたって、SOAP 又は REST サービスを実装する必要がある。

[12] 参考資料：https://x-road.global/architecture
[13] 参考資料：https://github.com/nordic-institute/X-Road/tree/develop/doc/Architecture

No.	インスタンス名	役割・機能
⑤	Certification Authority	セキュリティサーバに認証用証明書を発行する。また、X-Road 利用者組織の情報システムに署名用証明書を発行する。 これによってセキュリティサーバ間の接続を保護する。
⑥	OCSP Server	クライアントから発行された証明書の有効性の問合せに対して、認証局の証明書失効リストを参照して当該証明書の状態を確認し、有効/失効/不明のいずれかの応答を返す。
⑦	Time-Stamping Server	特定時点のデータ項目の存在を証明するタイムスタンプを発行する。 タイムスタンプはセキュリティサーバによってログに記録される。

X-Road におけるデータ連携は、データ利用者とデータ提供者が運用するセキュリティサーバを介して当事者間で直接、同期処理によってデータ連携が行われる（図 4-2-1-5）。

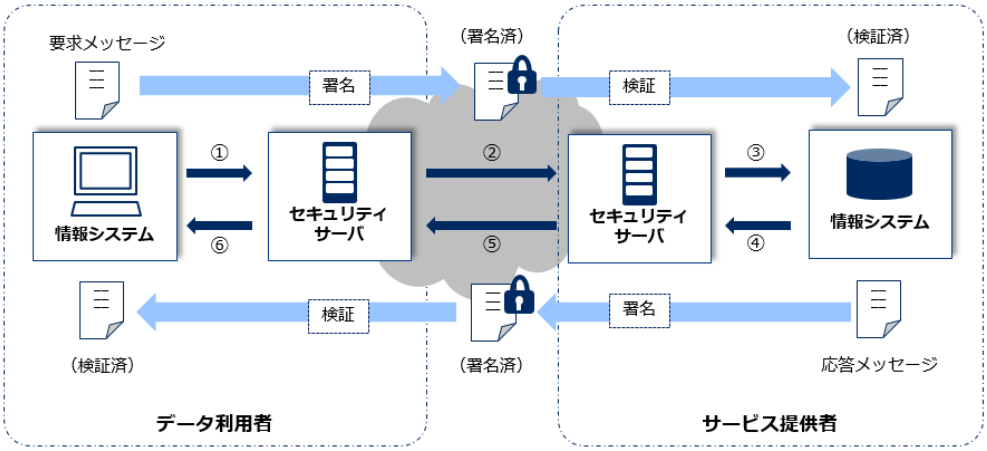


図 4-2-1-5　X-Road のデータ連携概要

X-Road において、データ利用者がデータ提供者に対してデータ要求をする場合、データ利用者の情報システムは組織内のセキュリティサーバに対して要求メッセージを送信する（図 4-2-1-5 の①）。要求メッセージを受けたセキュリティサーバは、相手先のデータ提供者のセキュリティサーバへ要求メッセージを転送する（図 4-2-1-5 の②）。要求メッセージを受信したデータ提供者側のセキュリティサーバは、組織内の情報システムに要求メッセージを送信し（図 4-2-1-5 の③）、情報システムは受信した要求メッセージの内容に基づいて処理を行う。データ提供者の情報システムは、データ要求に対する処理を実行後、処理結果を応答メッセージとして返却する。応答メッセージについては要求メッセージと逆の経路を辿り、要求時と同様に当事者のセキュリティサーバを介して応答メッセージを返却することでデータ連携処理が完了する（図 4-2-1-5 の④～⑥）。

X-Road において、データ利用者がデータ提供者に対する要求メッセージを作成する場合、セキュリティサーバは REST、及び SOAP サービスを提供する。データ利用者は、データ提供者が提供するインターフェース仕様に基づいて、メッセージの送信元、送信先の ID 情報やデータの検索条件を含めた要求メッセージを作成し送信する。また、セキュリティサーバ間通信は、HTTPS に基づいて TLS で暗号化される（図 4-2-1-5 の②と⑤）。通信の完全性において、送受信される全てのメッセージに対して電子署名とログ、タイムスタンプが付与されるため、改ざんやなりすまし、否認が困難である。

4-2-2.　ブローカー評価結果

本項では各ブローカーの評価結果について記載する。

(1)システム構成

下記に評価に利用した各ブローカー製品のシステム構成を示す。

(a)FIWARE Orion

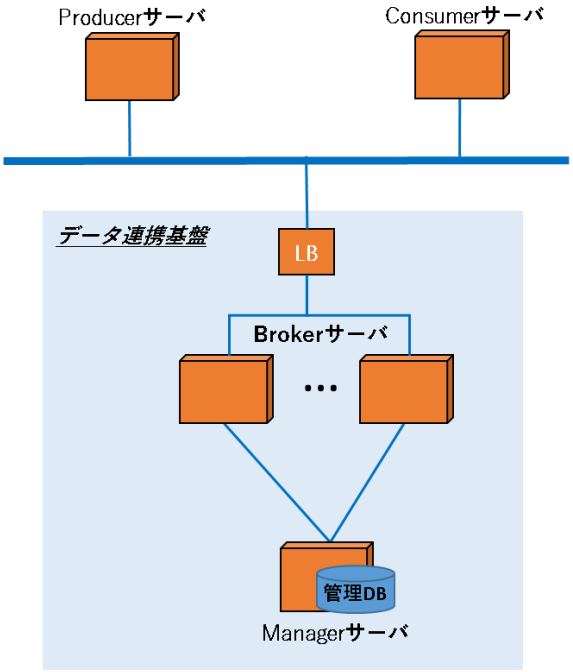


図 4-2-2-1　FIWARE Orion のシステム構成

システム構成の詳細は以下の表に記載する。

表 4-2-2-1　FIWARE Orion を構成するサーバ種別

No	サーバ種別	説明
1	Broker	Manager に保持されているコンテキスト情報を基に、データ仲介を行うサーバ。各コンテキストに Producer の所在情報(URL)を紐付けることによりデータ分散型を実現している。
2	Producer	Broker のコンテキストに対し、メッセージ送信を行うサーバ。本サーバに対して REST API を発行しメッセージ送信を行うことができる。
3	Consumer	Broker からのコンテキストに対し、メッセージ通知を行うサーバ。本サーバに対して REST API を発行しメッセージ取得要求を行うことができる。
4	Manager	コンテキスト情報（メタデータ）、所在管理、サブスクリプション情報を保持するサーバ。管理 DB として MongoDB を有する。

今回のシステム構成の特徴は以下となる。

- 特徴 1：Producer と Consumer の間のデータ連携を Broker を経由して実現する。Producer/Consumer と Broker 間に LB(LoadBalancer)を配置することで、Broker のスケールアウトを実現する。
- 特徴 2：Producer/Broker、Broker/Consumer 間のデータ送受信は、REST プロトコルにより実施する。
- 特徴 3：Consumer は任意のタイミングでデータ提供のリクエストを実施でき、リクエストに対する応答を待ち受ける。Producer の所在情報は Manager により管理されており、Consumer 自身は Producer の所在を意識することなくデータアクセスが可能となる。(データ仲介パターン A)
- 特徴 4：Producer のデータ更新のタイミングで Manager にあらかじめ登録されている Consumer に通知を行うことも可能である。(データ仲介パターン C)
- 特徴 5：Consumer の取得要求に応じたデータ連携（検索条件に従ったデータ提供など）が可能である。

(b)Apache Kafka

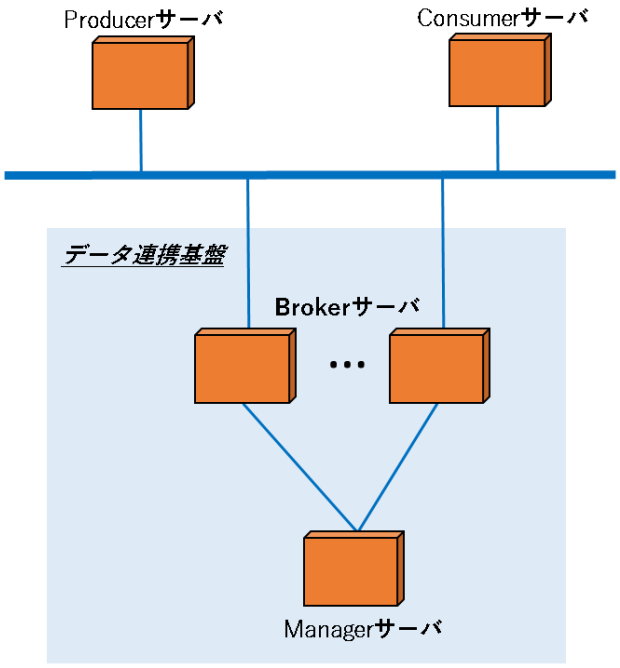


図 4-2-2-2　Apache Kafka のシステム構成

システム構成の詳細は以下の表に記載する。

表 4-2-2-2　Apache Kafka を構成するサーバ種別

No	サーバ種別	説明
1	Broker	メッセージキュー（Apache Kafka では topic と呼ぶ）を保持し、データ仲介を行うサーバ。連携するデータ項目単位で topic を作成することを想定し、連携するデータ分野の個数分の topic を作成し、評価を実施した。 Broker を複数台構成とした場合には、Producer 及び Consumer から Broker へのアクセスは自動的に負荷分散処理が行われる。そのため負荷分散装置等は不要である。
2	Producer	Broker の topic に対し、メッセージ送信を行うサーバ。本サーバの REST API を経由し、メッセージ送信を行うことができる。
3	Consumer	Broker の topic に対し、メッセージ取得要求を行うサーバ。本サーバの REST API を経由し、メッセージ取得要求を行うことができる。
4	Manager	Apache Kafka 環境の構成情報などを保持するサーバ。内部的に Apache Zookeeper が動作している。

性能や可用性の観点から、Producer、Consumer、Manager のサーバ種別ごとに複数台構成とすることが望ましい。しかし、今回のブローカー評価構成ではブローカー機能・性能の評価を目的としていたため、1 台構成とした。

今回のシステム構成の特徴は以下となる。

- 特徴 1：Producer と Consumer の間のデータ連携をメッセージキューを経由して実現する。Producer と Consumer が個別に連携インターフェースを持つ必要はないため、N：M のデータ連携が効率的に実現できる。
- 特徴 2：Producer からメッセージキューへのデータ送信、及び Consumer によるメッセージキューからのデータ取得は、REST API により json 形式のデータが利用可能である。これにより、推奨 API 仕様、及び、推奨データモデルに適応する。
- 特徴 3：Producer は任意のタイミングでデータ提供を実施でき、Broker からのデータ要求アクセスによる性能影響などを考慮する必要がない。Consumer も任意のタイミングで Broker からデータを取得できる。このため、疎なデータ連携が実現できる。
- 特徴 4：連携データをメッセージキューに一時的に保持することにより、Producer が提供したデータを複数の Consumer により繰り返し利用することができる。デバイスからのセンサーデータ仲介など、データを保持する機能を持たない Producer から、複数の Consumer へのデータ仲介が可能である。
- 特徴 5：Consumer の取得要求に応じたデータ連携（検索条件に従ったデータ提供など）は実現できない。
- 特徴 6：メッセージキューのデータは定期的に削除することができるものの、データを Broker に保持させない動作はできない。匿名性の高いデータの取り扱いには注意が必要と考えられる。

(c)X-Road

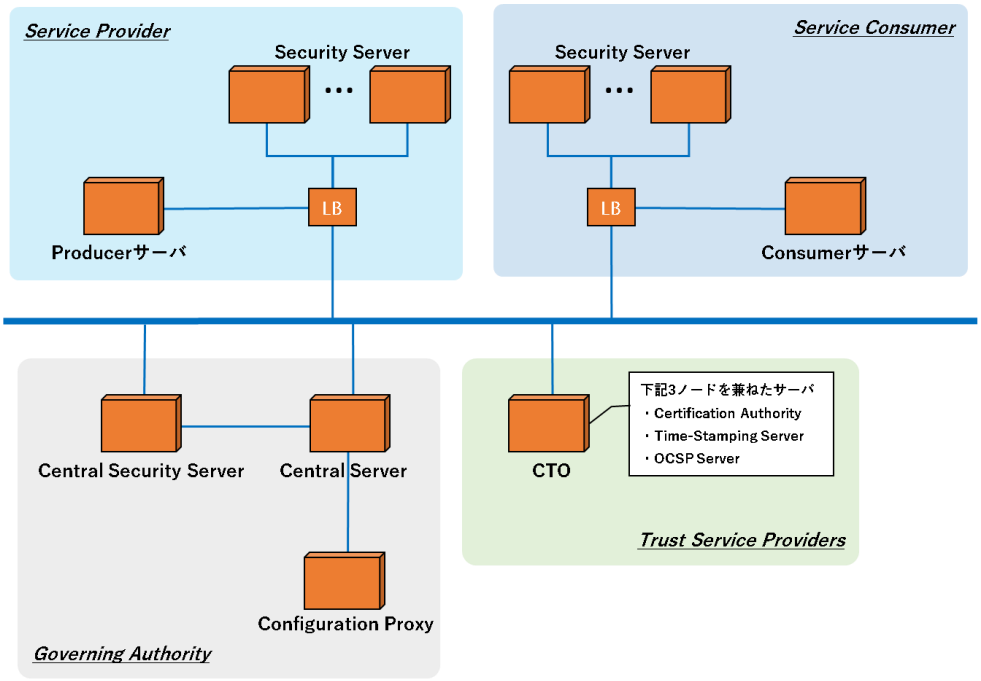


図 4-2-2-3　X-Road のシステム構成

システム構成の詳細は以下の表に記載する。

表 4-2-2-3　X-Road を構成するサーバ種別

No	サーバ種別	説明
1	Security Server	Consumer からの要求を受け取り Producer へリクエストを転送し、結果を Consumer へ返却するサーバ。Consumer、Producer それぞれに Security Server(※)があり Security Server 間の接続は高セキュアな通信となっている。 ※Service Provider 側 Security Server を「Service Security Server」、Service Consumer 側 Security Server を「Client Security Server」と呼ぶ。
2	Producer	Service Security Server からメッセージ取得要求を受け取り、結果を返却するサーバ。 本サーバに対して REST API を発行し要求を行うことができる。 X-Road では「Service Information System」と呼ばれる。
3	Consumer	Client Security Server に対し、メッセージ取得要求を行うサーバ。Security Server に対して REST API を発行しメッセージ取得要求を行うことができる。 X-Road では「Client Information System」と呼ばれる。
4	Central Server	X-Road メンバ、及び Security Server 情報の管理を行うサーバ。 また、信頼できる Certification Authority や Time-Stamping Server のリストなど、X-Road インスタンスのセキュリティポリシーを管理する。
5	Configuration Proxy	Security Server の情報を構成ファイルとして作成し、各 Security Server に公開するサーバ。

No	サーバ種別		説明
6	Central Security Server		Security Server から Central Server への管理サービス呼び出しを仲介するサーバ。
7	CTO	Certification Authority	Security Server に認証用証明書を発行するサーバ。また、Service Consumer に署名用証明書を発行する。これによって Security Server 間の接続を保護する。
		Time-Stamping Server	特定時点のデータ項目の存在を証明するタイムスタンプを発行するサーバ。
		OCSP Server	クライアントから発行された証明書の有効性の問合せに対して、認証局の証明書失効リストを参照して当該証明書の状態を確認し、有効/失効/不明のいずれかの応答を返すサーバ。

今回のシステム構成の特徴は以下となる。

- 特徴 1：Producer と Consumer の間のデータ連携を Security Server を経由して実現する。Security Server と他ノード（Producer や Consumer 等）間に LB(LoadBalancer)を配置することで、Security Server のスケールアウトを実現する。
- 特徴 2：Producer/Consumer と Security Server 間のデータ送受信は REST プロトコルにより実施する。
- 特徴 3：Security Server 同士の通信は必ず TSL による認証、暗号化が行われる。そのため、セキュアな通信が可能である。
- 特徴 4：データ提供側（Producer ならびに Service Security Server）の所在情報は Central Server により管理されており、Consumer は Central Server からデータ提供側の所在情報を受け取り、任意のタイミングでデータ提供のリクエストを実施する。同じデータ仲介パターン A の「FIWARE Orion」とは異なり、「X-Road」の Consumer はデータ提供者側に直接アクセスするため、データの所在情報を隠蔽することは不可である。
- 特徴 5：Certification Authority、Time-Stamping Server、OCSP Server は今回の評価向けに構築する。便宜的にこれら 3 ノードを 1 サーバに集約したものを「CTO」サーバと名付ける（3 ノードの頭文字を取った名称)。

(2)製品別評価結果

以下に、評価を実施した 3 製品の評価結果についてまとめる。

(a)ユースケース評価

- 2-2-3 にて整理したユースケース評価を評価内容（表 2-2-3-2）に従って評価結果を記載する。
- 評価結果に対する考察は、表 2-2-3-1 で示した現状の想定課題に基づく評価観点を軸に考察する。

(b)性能評価

- 2-2-3 にて整理したブローカーのスケーラビリティ評価とデータ仲介パターンによる性能評価における評価結果を記載する。

(ｱ)スケーラビリティ評価

ブローカーサーバ単体のサーバスペックを変動させる「スケールアップ評価」とブローカーの構成台数を変動させる「スケールアウト評価」を行う。

1)スケールアップ評価：

ブローカーサーバ 1 台に対して、最小構成のベースとなるサーバスペックを CPU：2 コア/メモリ：8GB とし、CPU 及びメモリをスケールアップさせて性能測定を行う。

サーバスペック変動条件： 2 コア/8GB → 4 コア/16GB → 8 コア/32GB

2)スケールアウト評価：

ブローカーサーバ 1 台のスペックを固定（CPU：2 コア/メモリ：8GB）し、ブローカーサーバの構成台数を変動させて性能測定を行う。

サーバ台数変動条件：1 台 → 2 台 → 4 台

(ｲ)データ仲介パターン評価

表 2-2-3-3 に示すデータ仲介パターンについての性能評価を実施する。

- 性能測定では、上記の評価内容に対してスループット、及び TAT(Turn Around Time)を測定し、測定結果から評価内容に対する影響を考察する。ただし、今回の評価では、データ量による定量値によって選定した 3 製品の優劣を評価するのではなく、各評価において、条件を変動した際の処理結果の傾向に着目して評価を行うこととする。

※ スループット（メッセージ/s）：秒間あたりに処理できるメッセージ量

TAT（ms）：1 メッセージあたりの処理時間（ミリ秒）。10 分間繰り返した際の平均 TAT。

(c)FIWARE Orion

(ｱ)ユースケース評価

FIWARE Orion のユースケース評価結果について以下に記載する。

表 4-2-2-4　ユースケース評価結果概要(FIWARE Orion)

【評価結果凡例】

○： 製品単体の機能により評価内容を満たすことができる。

△： 製品単体の機能により条件付きで評価内容を満たすことができる。

－： 製品単体では提供しない機能のため、該当要件を満たすには個別製品と組み合わせて実現する必要がある。

要件分類		評価内容	評価結果
機能	データ分散	データ利用者はデータの所在を意識することなく透過的にアクセスすることが可能であること	評価結果：○ 参照/更新要求を別のコンポーネントや外部システムへ仲介する機能を具備しており、本機能により分散方式のデータ連携を実現している。
		分散されたデータ提供者に対して、ブローカーを介することで共通のインターフェースでアクセス可能であること	評価結果：○ 分散されたデータについて NGSI という共通のインターフェースでアクセスが可能。
	データ蓄積	様々な種類、形式のデータについて取り扱い可能であること (評価、調査結果を基にブローカーで蓄積すべきデータの検討を行う)	評価結果：△ データ仲介機能によりブローカー内に用意した蓄積 DB にデータ蓄積することは可能だが、取り扱い可能なデータに制限がある。 取り扱い可能なデータについては他のブローカー製品と合わせて 4-2-3(1)(c)で示す。
	イベント処理	既存システムでデータ更新があった場合にブローカーを通して利用者側に通知が送信されること（イベント通知）	評価結果：○ サブスクリプション機能を利用してデータの更新通知を行うことが可能。
		既存システムが提供するサービスに対して既存システム側に負担をかけず仲介できること（サービス呼び出し）	評価結果：－ サービス呼び出しを行うことはブローカー単体の機能では実現不可。 アダプタ機能等と組み合わせて実現する必要がある。
	その他	データ提供者のデータに対して様々な方法でのデータ参照(検索)が可能であること	評価結果：○ データへのアクセスに検索フィルタリングをかけることが可能。
		ブローカーとデータ提供者の間でデータの変換処理が可能であること	評価結果：－ 変換処理については製品単体の機能では実現不可。 他コンポーネントと組み合わせて実現する必要がある。
		データ連携に係る接続の正確性を担保する機能を具備していること(再送処理、データアク	評価結果：○ ログによるトレーサビリティ機能を具備して

要件分類		評価内容	評価結果
		セスログ等)	いる。 再送機能を具備することで信頼性の高い接続を実現している。
非機能	運用・ 保守性	蓄積されたデータの管理や分散されたデータとの仲介に必要な管理機能を具備していること	評価結果：－ 管理用のAPIは具備されているが管理機能は製品として保有していない。 データ連携基盤の運用には別途管理機能を準備する必要がある。
		ブローカーについて公開ドキュメントやサポート体制などが整っていること	評価結果：○ 事例も多く、公開ドキュメントやサポート体制(コミュニティ)も充実している。
	性能・ 拡張性	性能向上に係る拡張性を具備していること	評価結果：○ FIWARE Orion を構成する各コンポーネント(Producer、Consumer、Broker)の単位ごとにスケールアウトが可能であり、システム拡張が容易である。
		可搬性を具備していること(実行環境に依存しないこと)	評価結果：○ クラウドサービス上での構築、及び、オンプレサーバでの構築の両方に対応可能である。
	セキュリティ	パーソナルデータを扱うためのセキュリティ機能を具備していること（オプトイン機能、データの匿名化等)	評価結果：－ オープンデータを扱うことを前提としている製品であるためパーソナルデータを扱うためのセキュリティ機能は具備されていない。

1)考察・課題

FIWARE Orionのユースケース評価結果に対して以下に考察を記載する。

表 4-2-2-5　ユースケース評価結果考察(FIWARE Orion)

No.	課題	課題概要	考察及び課題
1	データ分析	データを一元的に蓄積しないデータ分散型のシステムであっても、接続するシステム全体を1つのビッグデータとみなしAI分析等を活用するためにはどういった方法があるか。	• データ提供者の持つデータの ID と所在情報(URL)を登録し紐付けることで複数のデータ提供者の持つデータを一元的に管理することが可能となる。 • データへのアクセスについて、検索フィルタリングをかけることが可能であるが、条件付検索など分析に必要と思われる機能は十分とは言えないと考える。
3	蓄積すべきデータ	データ連携に係る接続の正確性とスピードを重視するため、データ連携基盤に最低限蓄積すべき静的データ(地図情報等)はどういったものがあるか。	• json 形式のデータを取り扱い可能であることから、属性付きデータのデータ連携が実現できている。バイナリデータはデータのエンコードなどの連携前処理を行うことで取り扱い可能である。 どのようなデータを蓄積すべきかについては 4-2-

No.	課題	課題概要	考察及び課題
			3(1)(c)で検討する。
4	既存システムとの連携	交通、行政等の既存の稼働中のシステムに接続する場合、これらのシステム及びデータ連携基盤そのものに大きな影響や負担を与えることなく正確にデータ連携するには、どういった方法があるか。	• データ提供者（既存システム）に NGSI サーバ機能（データの json 化及び REST 形式での API 提供）を具備する必要がある。 • データ利用者側では、ブローカーの REST API によりデータ取得し、json 形式の取得データをシステムに取り込む機能追加が必要となる。 • 今回の評価では確認できなかったが、ブローカーへの不正データ混入やデータ不正利用を防ぐために、データ提供者、データ利用者の認証やアクセス制御の機能追加が必要と考える。
5	コスト	運用性・保守性等を加味した上で、ランニングコストを低減させるためにはどういった方法があるか。	• 所在情報の管理などの運用作業を実施することが必要であるが、運用作業を容易にする機能は具備していない。 • ドキュメントは多数公開されているものの、不定期のソフトウェア更新運用や、バグに対する修正版の提供など OSS 一般に共通する運用・保守の課題がある。対策としては、OSS 有識者による体制構築などが考えられるが、エリアごとに整備するとコスト面のデメリットが大きい。
7	パーソナルデータの利活用	地域課題を解決するために、プライバシーリスクを軽減しつつ必要最小限のデータ連携のもとで最大限の価値を発揮するためにはどのような方法があるか。	• データ連携基盤に必須と考えられるオプトイン/オプトアウト処理の他、暗号化処理、匿名化処理などのセキュリティ機能については、今回の評価では確認できなかった。 ただし、上記のようなセキュリティ機能は、取り扱うデータにより実装要件が大きく異なってくると考えられる。このため、ブローカーとは別機能ブロックとして実装することが適当と考える。

(ｲ)性能評価

FIWARE Orion の性能評価結果について以下に記載する。

前提条件：1 メッセージのデータサイズは以下。

- パターン A：

  送信（consumer からの送信）： 0.15KB

  受信（consumer での受信）： 7.9KB

- パターン C：

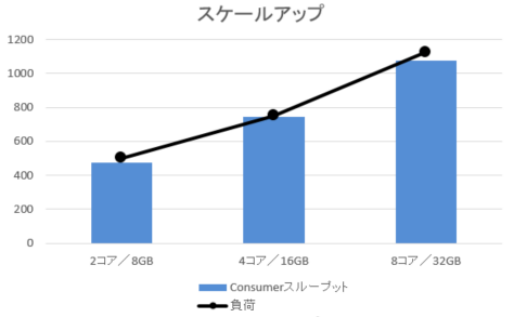
  送信（producer からの送信）： 0.4KB

  受信（consumer での受信）： 0.3KB

1)スケーラビリティ評価

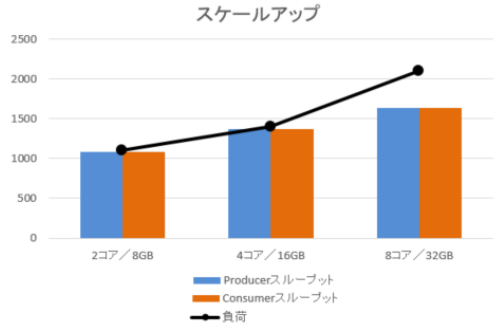
- スケールアップ（パターン A）

Broker ｻｰﾊﾞ条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
2 コア/8GB	Consumer:473.3	17.8
4 コア/16GB	Consumer:743.2	10.1
8 コア/32GB	Consumer:1077.4	7.0



- スケールアップ（パターン C）

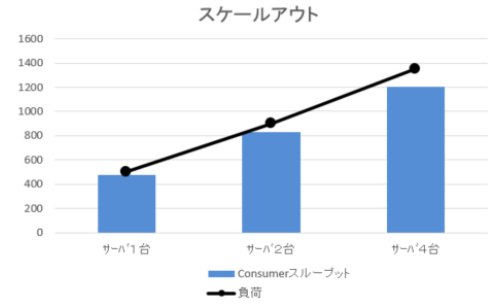
Broker ｻｰﾊﾞ条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
2 コア/8GB	Producer:1082.7	5.6
	Consumer:1082.7	
4 コア/16GB	Producer:1373.7	4.6
	Consumer:1373.7	
8 コア/32GB	Producer:1633.7	5.6
	Consumer:1633.7	



- スケールアウト（パターン A）

Broker ｻｰﾊﾞ条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
1 台	Consumer:473.3	17.8
2 台	Consumer:829.6	9.7
4 台	Consumer:1206.6	7.5

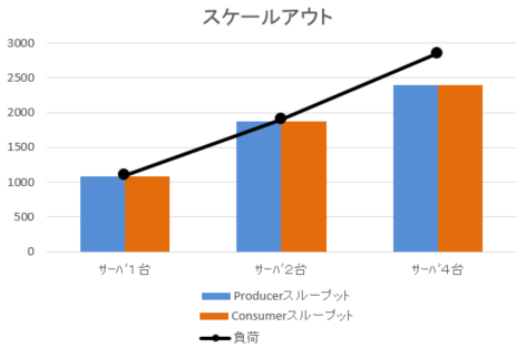
※ 1 台あたりのスペック：2 コア/8GB



- スケールアウト（パターン C）

Broker ｻｰﾊﾞ条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
1 台	Producer:1082.7	5.6
	Consumer:1082.7	
2 台	Consumer:1878.2	3.6
	Consumer:1878.2	
4 台	Consumer:2396.3	2.64
	Consumer:2396.3	

※ 1 台あたりのスペック：2 コア/8GB



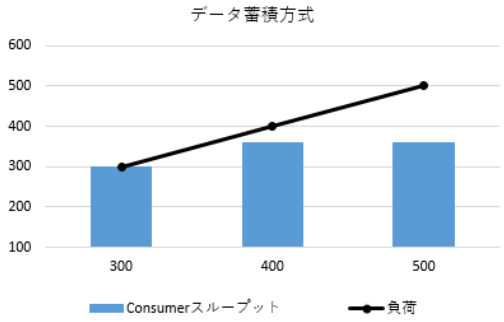
2)データ仲介パターン評価

データ仲介パターン評価の前提条件： Broker サーバスペック　1 台、2 コア/8GB

- データ蓄積方式

パターン A：

処理負荷（ﾒｯｾｰｼﾞ/s）	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
300	Consumer:300.8	22.1
400	Consumer:360.5	26.1
500	Consumer:357.7	27.6



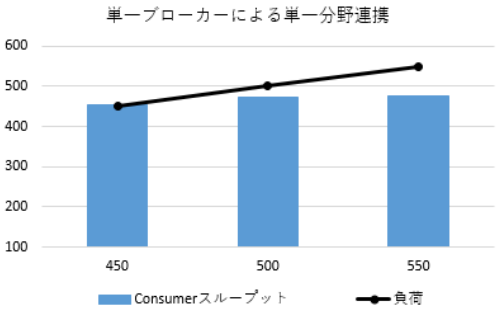
パターン C：

データ蓄積方式のデータ仲介は、パターン C では対応できないため、評価実施対象外とした。

- データ分散方式（単一ブローカーによる単一分野連携）

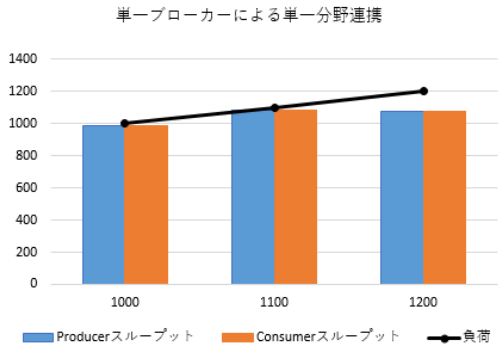
パターン A：

処理負荷（ﾒｯｾｰｼﾞ/s）	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
450	Consumer:453.0	10.3
500	Consumer:473.3	17.8
550	Consumer:477.0	19.5



パターン C：

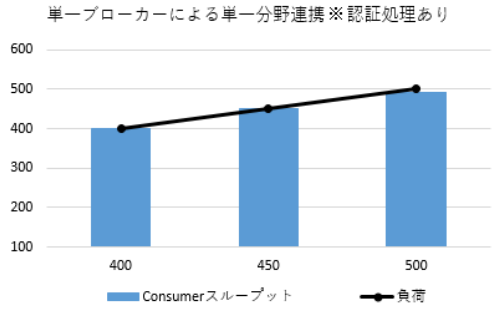
処理負荷（ﾒｯｾｰｼﾞ/s）	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
1000	Producer:986.5	4.0
	Consumer:986.5	
1100	Producer:1082.7	5.6
	Consumer:1082.7	
1200	Producer:1074.0	8.9
	Consumer:1074.0	



- データ分散方式（単一ブローカーによる単一分野連携）※認証処理あり

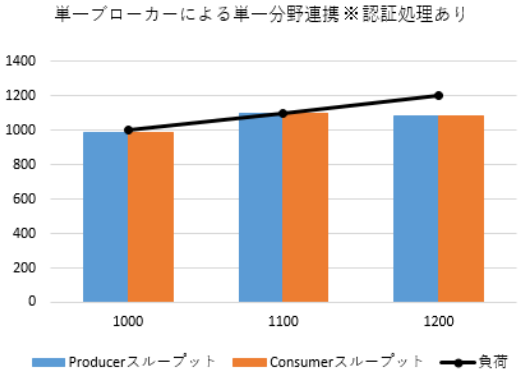
パターン A：

処理負荷（ﾒｯｾｰｼﾞ/s）	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
400	Consumer:399.0	9.4
450	Consumer:452.0	11.2
500	Consumer:491.4	15.1



パターンＣ：

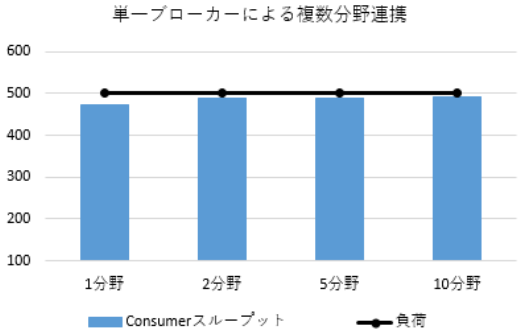
処理負荷（ﾒｯｾｰｼﾞ/s）	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
1000	Producer:989.3	3.7
	Consumer:989.3	
1100	Producer:1098.8	4.0
	Consumer:1098.8	
1200	Producer:1080.0	9.0
	Consumer:1080.0	



● データ分散方式（単一ブローカーによる複数分野連携）

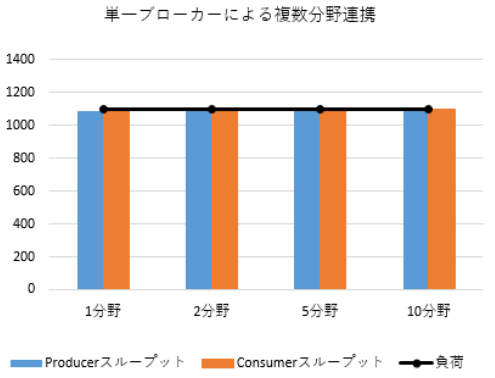
パターンＡ：※処理負荷を 500 メッセージ/s に固定

測定条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
分野数：1 分野	Consumer:473.3	17.8
分野数：2 分野	Consumer:489.1	15.0
分野数：5 分野	Consumer:489.7	14.7
分野数：10 分野	Consumer:491.1	13.8



パターンＣ：※処理負荷を 1100 メッセージ/s に固定

測定条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
分野数：1 分野	Producer:1082.7	5.6
	Consumer:1082.7	
分野数：2 分野	Producer:1102.7	4.3
	Consumer:1102.7	
分野数：5 分野	Producer:1103.3	5.1
	Consumer:1103.3	
分野数：10 分野	Producer:1094.9	5.4
	Consumer:1094.9	

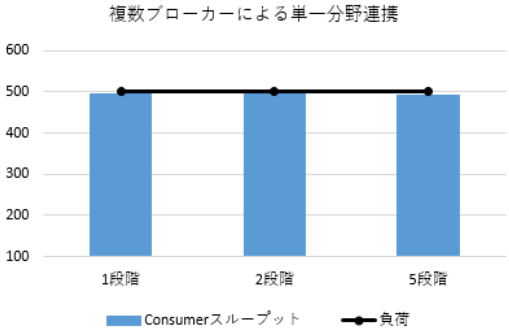


● データ分散方式（複数ブローカーによる単一分野連携）

パターンＡ：※処理負荷を 500 メッセージ/s に固定

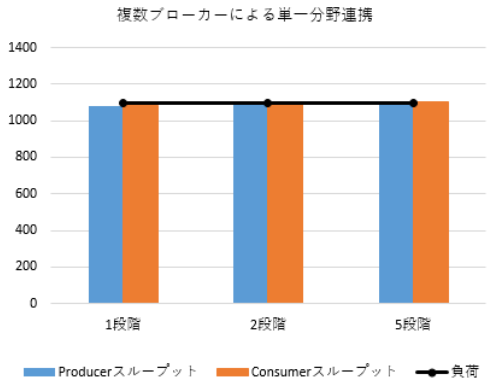
測定条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
ブローカー数：1 段	Consumer:494.5	4.2
ブローカー数：2 段	Consumer:495.5	5.8
ブローカー数：5 段	Consumer:492.6	10.9

※ Consumer からブローカーへのリクエストを
"/v2/registrations/"(id 指定)とした場合



パターンC：※処理負荷を1100メッセージ/sに固定

測定条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
ブローカー数：1段	Producer:1082.7	5.6
	Consumer:1082.7	
ブローカー数：2段	Producer:1098.9	3.5
	Consumer:1098.9	
ブローカー数：5段	Producer:1101.5	3.3
	Consumer:1101.5	



3)考察・課題

スケーラビリティ評価において、スループット増加にはスケールアップよりスケールアウトが効果的であることが確認できた。パターンA、パターンCともに、サーバ台数の変化に対し（1台、2台、4台）ほぼ比例したスループット増加が確認できた。スケールアップよりスケールアウトが効果的となった理由としては、1台の性能を倍にするよりも台数を倍にした方が処理の競合が少なくなり、スループットが向上したものと考えられる。

データ仲介パターン評価結果においては以下のとおり考察する。

データ蓄積方式については、データ分散方式(単一ブローカーによる単一分野連携)と比べてスループットが約24%低下した。これは、今回の評価ではデータ蓄積方式についてNGSIをSQLに変換するテストアプリを開発して組み込んでおり、その分のオーバーヘッドが影響していると考えられる。

今回の評価において、認証処理ありのデータ分散方式(単一ブローカーによる単一分野連携)については、認証処理を行わない場合と比べてスループットの低下は見られなかった。今回の評価では、トークン認証(X-Auth-Token ヘッダ)を追加したことによるブローカーへの影響を検証したが、この程度の負荷ではオーバーヘッドになり得なかったと考える。

データ分散方式(単一ブローカーによる複数分野連携)については、パターンAの場合、分野数を増加させてもスループットの低下は見られなかった。今回の評価では、トータルのデータ量は変えずに受け取る側（Producer）のサーバスペックも同一の条件で分野数を増加させたため、Producerの処理が分野数で分散され、分野数が多くなるごとにスループットがやや向上しているように見受けられた。このことより、ブローカーへの影響としては、単純な分野数の増加よりも全体のデータ量や各Producerのサーバスペック、ネットワーク環境などが重要になると考える。

また、パターンCにおいても分野数の増加によるスループットの低下は見られなかった。

データ分散方式(複数ブローカーによる単一分野連携)については、パターン Aにおいて、Consumerからブローカーへのリクエストを"/v2/registrations/"(id指定)とした場合、ブローカーの段階数を増加させてもスループットの低下は見られなかった。ただし、段階数が増えたことにより、TATが伸びていることを確認した。

(d)Apache Kafka

(ｱ)ユースケース評価

Apache Kafka のユースケース評価結果について以下に記載する。

表 4-2-2-6　ユースケース評価結果概要(Apache Kafka)

【評価結果凡例】

○： 製品単体の機能により評価内容を満たすことができる。

△： 製品単体の機能により条件付きで評価内容を満たすことができる。

－： 製品単体では提供しない機能のため、該当要件を満たすには個別製品と組み合わせて実現する必要がある。

機能分類		評価内容	評価結果
機能	データ分散	データ利用者はデータの所在を意識することなく透過的にアクセスすることが可能であること	評価結果：○ Broker サーバが保持するメッセージキュー(topic)によりデータ提供者とデータ利用者の紐付けを行うことでデータ利用者に対してデータ提供者を隠ぺいしたデータ連携を実現する。 データ提供者：データ利用者が 1：1、N:1、1：M、N:M のいずれのケースでもデータ連携が可能。
		分散されたデータ提供者に対して、ブローカーを介することで共通のインターフェースでアクセス可能であること	評価結果：○ メッセージキューへのデータ送信、及びデータ取得は、REST API により json 形式のデータが利用可能。
	データ蓄積	様々な種類、形式のデータについて取り扱い可能であること (評価、調査結果を基にブローカーで蓄積すべきデータの検討を行う)	評価結果：－ データ蓄積方式のデータ仲介については、非同期方式（Pull 型）では対応できない。 取り扱い可能なデータについては他のブローカー製品と合わせて 4-2-3(1)(c)で示す。
	イベント処理	既存システムでデータ更新があった場合にブローカーを通して利用者側に通知が送信されること（イベント通知）	評価結果：－ 非同期方式（Pull 型）のブローカーではイベント通知機能を実現できない。
		既存システムが提供するサービスに対して既存システム側に負担をかけず仲介できること（サービス呼び出し）	評価結果：－ 非同期方式（Pull 型）のブローカーではサービス呼び出し機能は実現できない。
	その他	データ提供者のデータに対して様々な方法でのデータ参照(検索)が可能であること	評価結果：－ データ提供者がブローカーに提供したデータ全件をデータ利用者が利用する方法に限られる。
		ブローカーとデータ提供者の間でデータの変換処理が可能であること	評価結果：－ ソフトウェア単体ではデータ変換機能を持たないため、別ソフトウェアで実装する必要がある。

機能分類		評価内容	評価結果
		データ連携に係る接続の正確性を担保する機能を具備していること(再送処理、データアクセスログ等)	評価結果：○ Apache Kafka のメッセージ送信保証機能により、データ連携の正確性を向上できる。 REST API に対するアクセスログを取得できる。
非機能	運用・保守性	蓄積されたデータの管理や分散されたデータとの仲介に必要な管理機能を具備していること	評価結果：－ CLI による topic の追加削除など、運用機能を持っているが、GUI による運用機能や蓄積データ管理機能は持っていない。
		ブローカーについて公開ドキュメントやサポート体制などが整っていること	評価結果：○ コミュニティ活動が活発であり、ソフトウェアのドキュメント(英語)が公開されている。 有償サポートサービスの提供ベンダが存在し、安定的なサポートを受けることが可能である一方、民間企業での利用実績が多数存在し、特定の企業、組織に依存しないコミュニティが形成されているソフトウェアである。
	性能・拡張性	性能向上に係る拡張性を具備していること	評価結果：○ Kafka を構成する各コンポーネント(Producer、Consumer、Broker)の単位ごとにスケールアウトが可能であり、システム拡張が容易である。
		可搬性を具備していること(実行環境に依存しないこと)	評価結果：○ クラウドサービス上での構築、及び、オンプレサーバでの構築の両方に対応可能である。
	セキュリティ	パーソナルデータを扱うためのセキュリティ機能を具備していること（オプトイン機能、データの匿名化等）	評価結果：－ ソフトウェア単体ではパーソナルデータを扱うためのセキュリティ機能を持たないため、別ソフトウェアで実装する必要がある。

1)考察・課題

Apache Kafka のユースケース評価結果に対して以下に考察を記載する。

表 4-2-2-7　ユースケース評価結果考察(Apache Kafka)

No.	課題	課題概要	評価観点
1	データ分析	データを一元的に蓄積しないデータ分散型のシステムであっても、接続するシステム全体を１つのビッグデータとみなしAI 分析等を活用するためにはどういった方法があるか。	• 複数のデータ提供者から提供されたデータを topic 情報の指定により利用できることから、共通の IF でのデータアクセス、データ所在の隠蔽を実現する方法を提供している。ただし、topic に提供されたデータ全件を利用者に連携する動作となるため、システム全体を１つのデータとして扱う動作とはいえない。
3	蓄積すべき	データ連携に係る接続の正確性とスピード	• json 形式のデータを取り扱い可能であることから、属

No.	課題	課題概要	評価観点
	データ	を重視するため、データ連携基盤に最低限蓄積すべき静的データ(地図情報等)はどういったものがあるか。	性付きデータのデータ連携が実現できている。バイナリデータはデータのエンコードなどの連携前処理を行うことで取り扱い可能である。 どのようなデータを蓄積すべきかについては 4-2-3(1)(c)で検討する。
4	既存システムとの連携	交通、行政等の既存の稼働中のシステムに接続する場合、これらのシステム及びデータ連携基盤そのものに大きな影響や負担を与えることなく正確にデータ連携するには、どういった方法があるか。	• データ提供者のシステムにおいては、提供データを json 形式で出力し、ブローカーの REST API によりデータ送信する機能追加が必要となる。 • データ利用者のシステムにおいては、ブローカーの REST API によりデータ取得し、json 形式の取得データをシステムに取り込む機能追加が必要となる。 • 非同期方式（Pull 型）の特徴として、データ提供者、データ利用者ともに任意のタイミングでデータ連携を実行すればよいため、データ連携によるシステム負荷に対応しやすい。 • 今回の評価では確認できなかったが、ブローカーへの不正データ混入やデータ不正利用を防ぐために、データ提供者、データ利用者の認証やアクセス制御の機能追加が必要と考える。データ利用者の認証やアクセス制御はオプトイン/オプトアウト処理の前提でもある。
5	コスト	運用性・保守性等を加味した上で、ランニングコストを低減させるためにはどういった方法があるか。	• topic の管理などの運用作業を実施することが必要であるが、運用作業を容易にする機能は持っていない。 • ドキュメントは多数公開されているものの、不定期のソフトウェア更新運用や、バグに対する修正版の提供など OSS 一般に共通する運用・保守の課題がある。対策としては、ベンダサポートの活用や OSS 有識者体制の確保などが考えられるが、エリアごとに本対策を実施するとコスト面のデメリットが大きい。
7	パーソナルデータの利活用	地域課題を解決するために、プライバシーリスクを軽減しつつ必要最小限のデータ連携のもとで最大限の価値を発揮するためにはどのような方法があるか。	• データ連携基盤に必須と考えられるオプトイン/オプトアウト処理の他、暗号化処理、匿名化処理などのセキュリティ機能については、今回の評価では確認できなかった。 • ただし、上記のようなセキュリティ機能は、取り扱うデータにより実装要件が大きく異なってくると考えられる。このため、ブローカーとは別機能ブロックとして実装することが適当と考える。

(ｲ)性能評価

Apache Kafka の性能評価結果について以下に記載する。

前提条件：1 メッセージのデータサイズは以下。

パターン B：

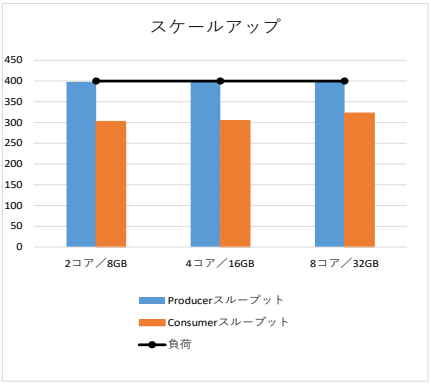
送信（producer からの送信）： 5.3KB

受信（consumer での受信）： 5.3KB

1)スケーラビリティ評価

- スケールアップ

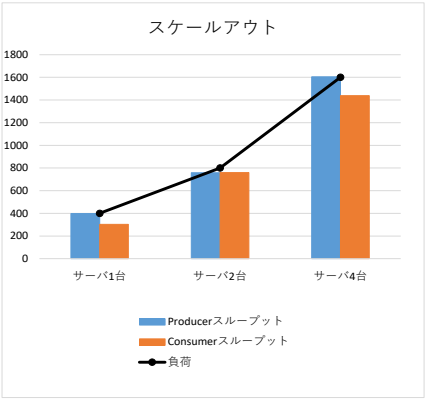
Broker ｻｰﾊﾞ条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）
2 コア/8GB	Producer:398.1
	Consumer:302.7
4 コア/16GB	Producer:398.2
	Consumer:304.7
8 コア/32GB	Producer:398.2
	Consumer:322.8



- スケールアウト

Broker ｻｰﾊﾞ条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）
1 台	Producer:398.1
	Consumer:302.7
2 台	Producer:758.8
	Consumer:760.1
4 台	Producer:1604.0
	Consumer:1438.5

※ 1 台あたりのスペック：2 コア/8GB



2)データ仲介パターン評価

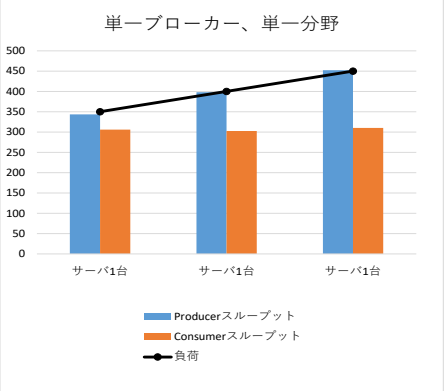
データ仲介パターン評価の前提条件： Broker サーバスペック 1 台、2 コア/8GB

- データ蓄積方式

データ蓄積方式のデータ仲介については、非同期方式（Pull 型）では対応できないため、評価実施対象外とした。

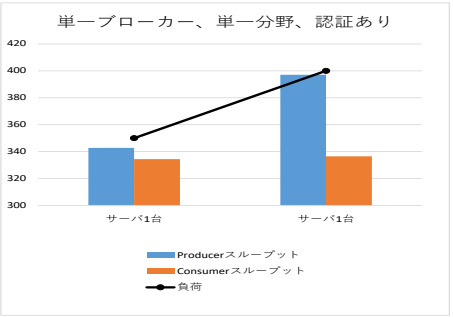
- データ分散方式（単一ブローカーによる単一分野連携）

処理負荷（ﾒｯｾｰｼﾞ/s）	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）
350	Producer:343.6
	Consumer:305.9
400	Producer:398.1
	Consumer:302.7
450	Producer:452.2
	Consumer:310.1



- データ分散方式（単一ブローカーによる単一分野連携　※認証処理あり）

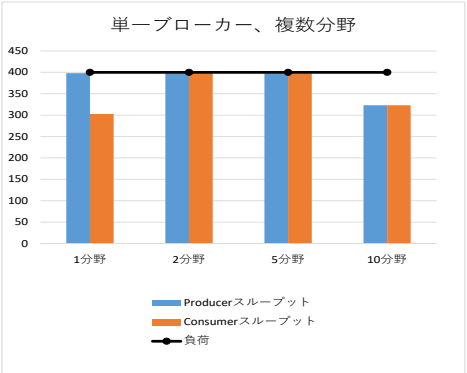
処理負荷（ﾒｯｾｰｼﾞ/s）	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）
350	Producer:342.7
	Consumer:334.4
400	Producer:397.1
	Consumer:336.5



- データ分散方式（単一ブローカーによる複数分野連携）

※ 処理負荷は 400 メッセージ/s を Producer に送信で固定

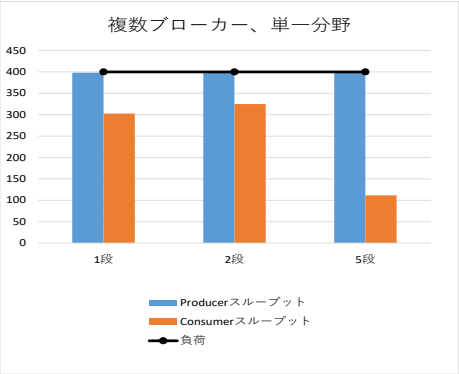
測定条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）
分野数：1 分野	Producer:398.1
	Consumer:302.7
分野数：2 分野	Producer:397.8
	Consumer:397.8
分野数：5 分野	Producer:397.4
	Consumer:397.4
分野数：10 分野	Producer:323.0
	Consumer:323.0



- データ分散方式（複数ブローカーによる単一分野連携）

※ 処理負荷は 400 メッセージ/s を Producer に送信で固定

測定条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）
ブローカー数：1 段	Producer:398.1
	Consumer:302.7
ブローカー数：2 段	Producer:398.1
	Consumer:325.0
ブローカー数：5 段	Producer:399.8
	Consumer:111.6



3)考察・課題

スケーラビリティ評価において、スループット増加にはスケールアップよりスケールアウトが効果的であることが確認できた。さらに、サーバ台数の変化に対し（1 台、2 台、4 台）ほぼ比例したスループット増加が確認できた。この結果は、スケールアウトによる性能改善の効率の良いシステムであることを示している。一方、Broker サーバのスケールアップにより Consumer スループットの増加はほとんどなかった。これは、Consumer が 1topic から 1 リクエストで取得できるメッセージ数に上限があったことが原因と考えられる。topic を構成する partition 数設定の変更により、スループットが増大すると見込まれる。

データ仲介パターン評価においては、単一ブローカーによる 10 分野連携において Producer スループットが減少した。Producer サーバ、Broker サーバのリソース使用率には大きな変化は見られなかったことから、ソフトウェア内部動作に起因する可能性が高い。データ連携の分野数に応じて、Broker サーバのスケールアウトが必要であると考えられる。

また、5 段のブローカーによる単一分野連携において、Consumer スループットが減少したが、これは 2 段～5 段目のブローカー処理を同一の Kafka 環境で実行する今回の疑似環境構成に起因するものと評価している。非同期方式（Pull 型）のブローカー処理は、2 段のブローカー処理を直列につないだ処理となることから、ブローカーの段数による性能影響は発生しにくい処理方式である。

その他のデータ仲介パターンの差異によるスループットの変化はみられなかった。

定量的な性能評価として、今回の評価において、Broker サーバを 4 台構成とした場合には毎秒 1,400 メッセージのスループットが確認できた。

このスループットは、データ提供者として、ランダムなタイミングでデータ提供するデバイスを想定すると、以下のデータ量に相当する（実機環境ではデータ送信タイミングの偏りが発生する可能性があり、データ処理件数は 2～3 倍の余裕を持って設計する必要がある）。

- 84,000 台のデバイスが 1 分間に 1 回の頻度でデータ送信
- 840,000 台のデバイスが 10 分間に 1 回の頻度でデータ送信
- 5,000,000 台のデバイスが 1 時間に 1 回の頻度でデータ送信

上記のように、実機環境におけるスループット目標はデータ提供間隔を考慮して定めることになる。実機環境におけるデータ連携では、データの特性（誤差や提供経路レイテンシーなど）により、有効なデータ提供間隔の下限が存在するため、データ提供間隔を小さくしすぎるとコ

スト面のデメリットが大きい。またデータ提供間隔に依存してデータ利用のリアルタイム性が変化することから、ブローカーが提供するデータがどの程度のリアルタイム性を持つデータであるかをデータ利用者に対し明らかにする必要がある。

(e)X-Road

(ｱ)ユースケース評価

X-Road のユースケース評価結果について以下に記載する。

表 4-2-2-8　ユースケース評価結果概要(X-Road)

【評価結果凡例】

○： 製品単体の機能により評価内容を満たすことができる。

△： 製品単体の機能により条件付きで評価内容を満たすことができる。

－： 製品単体では提供しない機能のため、該当要件を満たすには個別製品と組み合わせて実現する必要がある。

機能分類		評価内容	評価結果
機能	データ分散	データ利用者はデータの所在を意識することなく透過的にアクセスすることが可能であること	評価結果：－ データ提供者ごとにセキュリティサーバが異なるため、データ利用者はデータの所在を把握し、指定する必要がある。
		分散されたデータ提供者に対して、ブローカーを介することで共通のインターフェースでアクセス可能であること	評価結果：○ セキュリティサーバを介して SOAP 又は REST を利用してデータ交換が可能。 ※今回は REST にて評価を実施。
	データ蓄積	様々な種類、形式のデータについて取り扱い可能であること (評価、調査結果を基にブローカーで蓄積すべきデータの検討を行う)	評価結果：－ 製品のアーキテクチャとして、ブローカー内のデータ蓄積は提供しない。 取り扱い可能なデータについては他のブローカー製品と合わせて 4-2-3(1)(c)で示す。
	イベント処理	既存システムでデータ更新があった場合にブローカーを通して利用者側に通知が送信されること（イベント通知）	評価結果：－ 製品自体は、データ利用者から送信されたリクエストに対して応答を返すのみであるため、イベント通知機能は製品単体では実現できない。
		既存システムが提供するサービスに対して既存システム側に負担をかけず仲介できること（サービス呼び出し）	評価結果：－ サービス呼び出しを行うことは製品単体では実現できない。
	その他	データ提供者のデータに対して様々な方法でのデータ参照(検索)が可能であること	評価結果：－ 製品自体にはデータの検索や変更、削除の機能は持たないため、製品単体では実現できない。
		ブローカーとデータ提供者の間でデータの変換処理が可能であること	評価結果：－ データ変換処理については製品単体の機能では実現不可。 情報システム側で考慮する必要がある。
		データ連携に係る接続の正確性を担保する機能を具備していること(再送処理、データアクセスログ等)	評価結果：○ ログによるトレーサビリティ機能を具備している。 全てのアクセス情報は信頼されたタイムスタンプサーバによる時刻情報が付与されること

機能分類		評価内容	評価結果
			によって、いつ誰がアクセスしたかが記録される。 ただし、データの再送機能はない。
非機能	運用・ 保守性	蓄積されたデータの管理や分散されたデータとの仲介に必要な管理機能を具備していること	評価結果：○ セントラルサーバとセキュリティサーバに管理 WebUI を具備している。 セキュリティサーバの情報登録は管理 WebUI を用いて実施可能。
		ブローカーについて公開ドキュメントやサポート体制などが整っていること	評価結果：○ ソフトウェアは MIT ライセンスの OSS として公開されている。
	性能・ 拡張性	性能向上に係る拡張性を具備していること	評価結果：○ セキュリティサーバの前段にロードバランサを配置することで高可用性と高パフォーマンスを目的としたスケーラビリティが可能。
		可搬性を具備していること(実行環境に依存しないこと)	評価結果：○ クラウドサービス上での構築、及び、オンプレサーバでの構築の両方に対応可能である。
	セキュリティ	パーソナルデータを扱うためのセキュリティ機能を具備していること（オプトイン機能、データの匿名化等）	評価結果：－ 製品自体にはパーソナルデータに対するオプトイン機能やデータの匿名化機能は存在しないため、データ提供者側で考慮する必要がある。

1)考察・課題

X-Road のユースケース評価結果に対して以下に考察を記載する。

表 4-2-2-9　ユースケース評価結果考察(X-Road)

No.	課題	課題概要	考察
1	データ分析	データを一元的に蓄積しないデータ分散型のシステムであっても、接続するシステム全体を 1 つのビッグデータとみなし AI 分析等を活用するためにはどういった方法があるか。	• X-Road の製品自体は、データ利用者とデータ提供者のデータ交換をセキュリティサーバを介して行うことによって、標準化された、セキュアな伝送経路を提供することをコンセプトとした製品と言える。X-Road は、それ自体にデータの検索や更新の機能は存在しない。具体的に実現するためには、データ提供者側に Web アプリケーション等を構築し、HTTP リクエストの GET パラメータや POST パラメータにて、検索条件や更新条件のキーを指定し、Web アプリケーション側で対応する必要がある。 • X-Road はセキュリティサーバを介して当事者間でデータ連携を行うため、データ利用者は、データの所在を把

No.	課題	課題概要	考察
			握した上でそれぞれのデータ提供者を指定してデータ要求を行う必要がある。X-Road は接続するシステム全体を 1 つのビッグデータとして扱うことはできない。
3	蓄積すべきデータ	データ連携に係る接続の正確性とスピードを重視するため、データ連携基盤に最低限蓄積すべき静的データ（地図情報等）はどういったものがあるか。	• X-Road の製品自体に特定のデータファイルを専門的に扱う機能は存在せず、テキストはテキストとして、バイナリはバイナリとして伝送することができるのみである。 どのようなデータを蓄積すべきかについては 4-2-3(1)(c)で検討する。
4	既存システムとの連携	交通、行政等の既存の稼働中のシステムに接続する場合、これらのシステム及びデータ連携基盤そのものに大きな影響や負担を与えることなく正確にデータ連携するには、どういった方法があるか。	• X-Road のセキュリティサーバは、SOAP 及び REST と提供することによってデータ連携メッセージを標準化しているため、X-Road に既存システムを接続するためには、SOAP もしくは REST サービスを実装する必要がある。データ連携のセキュリティサーバ間通信に関しては、署名と認証、暗号化、ロギングなどをセキュリティサーバが透過的に行うため、既存システムとしては、それらの機能を実装する必要がない。 • X-Road は、データ利用者から送信されたリクエストに対して応答を返す機能しか存在しないため、データ提供者側からのイベント通知機能等は存在しない。リアルタイム性を持ったデータ提供者側のデータ更新内容を知るためにはデータ利用者側からの定期的なポーリングが必要となるが、データ提供者側のリソース負荷を考慮すると妥当ではないと考える。
5	コスト	運用性・保守性等を加味した上で、ランニングコストを低減させるためにはどういった方法があるか。	• セントラルサーバとセキュリティサーバには管理 WebUI が具備されており、セキュリティサーバの情報登録は管理 WebUI を用いて実施できるように運用性が考慮されている。 • X-Road は MIT ライセンスの OSS として公開されているため、不定期のソフトウェア更新運用や、バグに対する修正版の提供など OSS 一般に共通する運用・保守の課題がある。対策としては、OSS 有識者体制の確保や、ソフトウェアベンダが X-Road をベースに開発した製品を利用し有償サポート等を受けることが考えられるが、スーパーシティごとに本対策を実施するとコスト面のデメリットが大きい。
7	パーソナルデータの利活用	地域課題を解決するために、プライバシーリスクを軽減しつつ必要最小限のデータ連携のもとで最大限の価値を発揮するためにはどのような方法があるか。	• セキュリティ観点について、データ連携を行うセキュリティサーバ間通信は信頼された認証機関による電子署名と認証、暗号化が行われるが、X-Road の製品自体にはパーソナルデータに対するオプトイン機能やデータの匿名化機能は存在しない。 • ただし、上記のようなセキュリティ機能は、取り扱うデ

No.	課題	課題概要	考察
			ータにより実装要件が大きく異なってくると考えられる。このため、ブローカーとは別機能ブロックとして実装することが適当と考える。

(ｲ)性能評価

X-Road の性能評価結果について以下に記載する

前提条件：1 メッセージのデータサイズは以下。

パターン A：

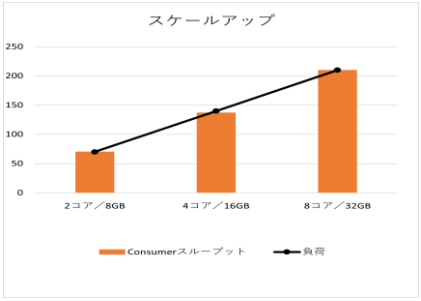
送信（consumer からの送信）： 0.3KB

受信（consumer での受信）： 6.1KB

1)スケーラビリティ評価

● スケールアップ

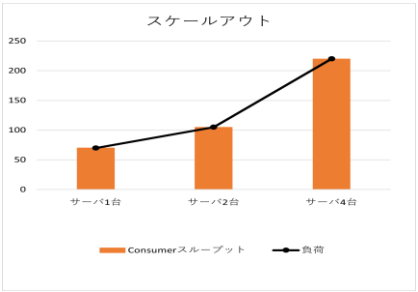
Security Server 条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
2 コア/8GB	70.0	75.6
4 コア/16GB	139.6	69.5
8 コア/32GB	209.8	152.5



● スケールアウト

Security Server 条件	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
1 台	70.0	75.6
2 台	104.7	123.1
4 台	219.7	256.6

※ 1 台あたりのスペック：2 コア/8GB



2)データ仲介パターン評価

データ仲介パターン評価の前提条件： Broker サーバスペック 1 台、2 コア/8GB

● データ蓄積方式

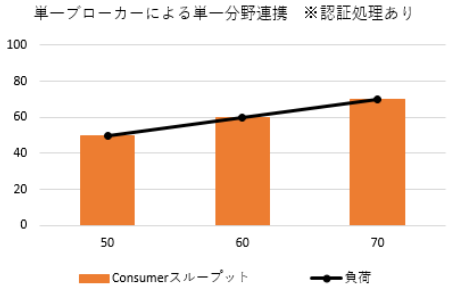
X-Road にはデータ蓄積方式に分類される機能が存在していないため、評価実施対象外とした。

● データ分散方式(単一ブローカーによる単一分野連携)

X-Road はデータ取得者とデータ提供者をセキュアに連携させることを目的に開発されており、伝送経路の途中には必ず X-Road の機能としての認証が行われる。そのため、認証なしのデータ仲介パターンは存在していないため、評価対象外とした。

- データ分散方式（単一ブローカーによる単一分野連携　※認証処理あり）

処理負荷（ﾒｯｾｰｼﾞ/s）	ｽﾙｰﾌﾟｯﾄ（ﾒｯｾｰｼﾞ/s）	TAT（ms）
50	50.0	47.6
60	60.0	56.7
70	70.0	75.6



- データ分散方式（単一ブローカーによる複数分野連携）

X-Road には、単一ブローカーへの 1 つのリクエストで複数のデータ提供者から情報取得を行うような機能は存在していないため、評価対象外とした。

- データ分散方式（複数ブローカーによる単一分野連携）

X-Road には、多段階連携を実現することができる機能は存在していないため、評価対象外とした。

3)考察・課題

スケーラビリティ評価において、スループット増加の効果はスケールアップ/スケールアウトのどちらでもほぼ同じで、性能/サーバ台数に比例してスループットが増加していくことが確認できた。ただし、スループットが頭打ちになって以降、スケールアップの方が早く効率が悪くなっていく点も確認できている。

※スケールアウト時（サーバ 4 台）は毎秒 280 メッセージに対してスループットは毎秒 200 メッセージだが、スケールアップ時（8 コア/8GB）は同条件でスループットは毎秒 180 メッセージ程度になった。

データ仲介パターン評価については、X-Road の基本的な機能が「データ取得者とデータ提供者をセキュアに連携させること」を目的に設計されているため、データ取得者とデータ提供者の 1 対 1 の安全な連携機能しか提供されておらず、認証処理ありのデータ分散方式（単一ブローカーによる単一分野連携）しか評価できていない。また、基本的に内容はスケーラビリティ評価の結果と同一である。

定量的な性能評価としては、今回の評価において、ブローカーサーバを 4 台構成（各サーバ 2 コア/8GB）とした場合には毎秒 220 メッセージのスループットが確認できた。このスループットは、データ提供者として、ランダムなタイミングでデータ提供するデバイスを想定すると、以下のデータ量に相当する（実機環境ではデータ送信タイミングの偏りが発生する可能性があり、データ処理件数は 2～3 倍の余裕を持って設計する必要がある）。

- 13,200 台のデバイスが 1 分間に 1 回の頻度でデータ送信
- 132,000 台のデバイスが 10 分間に 1 回の頻度でデータ送信
- 792,000 台のデバイスが 1 時間に 1 回の頻度でデータ送信

上記のように、スループットの効率という観点では、X-Road はあまり良い結果が出ていない。評価の結果から、スループットが頭打ちになった際に性能に影響を与えている要素は、ClientSecurityServer と ServiceSecurityServer の CPU 使用率と DiskWrite 頻度と想定されるが、これは X-Road 自体の認証処理とその履歴情報のデータベース等への DiskWrite が原因と考えられる。また、CPU/DiskWrite ともに、常時数値が高いわけではなく、極端に高くなるタイミングが見受けられるため、内部的な処理が並列で動作していると想定される。そのため、リクエストに対して適切なキュー処理を挟むことができれば負荷軽減の可能性があると考える。また、実機環境におけるデータ連携では、さらにデータ特性も考慮に入れる必要があるが、大規模なネットワーク環境構築を行う場合、CPU/Disk 性能も考慮に入れる必要がある。

4-2-3. 考察

(1)評価結果考察

下記に評価結果に基づきブローカーとして想定課題に関する総括を記載する。

(a)ユースケース評価

表 4-2-3-1 ユースケース評価結果考察

No.	課題	課題概要/検証方針	考察及び課題
1	データ分析	データを一元的に蓄積しないデータ分散型のシステムであっても、接続するシステム全体を1つのビッグデータとみなしAI分析等を活用するためにはどういった方法があるか。 【検証方針】各種サービスに対して物理的なデータの所在を隠蔽し、統合的なデータアクセスを実現する方法を検証する。	データ提供者の持つデータの ID と所在情報(URL)を登録し紐付けることで複数のデータ提供者の持つデータを一元的に管理することが可能となる。ただし、ブローカーが保持する情報は原則として所在情報のみであり、ブローカー自体がデータの条件等による検索を実施することは難しい。条件付き検索等の機能はデータ提供者側の API 仕様に依存する。
3	蓄積すべきデータ	データ連携に係る接続の正確性とスピードを重視するため、データ連携基盤に最低限蓄積すべき静的データ(地図情報等)はどういったものがあるか。 【検証方針】ブローカー製品でどのようなデータが取り扱うことができるかを検証する。	全てのブローカー製品がテキスト形式に加えバイナリ形式のデータ仲介が可能であった。ただしバイナリデータをテキスト形式へエンコードする手法が主であり、データサイズが大きい場合は NW 負荷やクライアント負荷が高くなることが懸念される。サイズが大きい場合はストリーム配信が可能なオンラインストレージ等への蓄積が望ましいと考えられる。詳細については(c)参照。
4	既存システムとの連携	交通、行政等の既存の稼働中のシステムに接続する場合、これらのシステム及びデータ連携基盤そのものに大きな影響や負担を与えることなく正確にデータ連携するには、どういった方法があるか。 【検証方針】連携先システム・ブローカー間の適切な機能分担を検証する。また、利用者と他システムとの仲介という観点で、利用者への通知や、連携先システムへの機能呼び出し等メッセージの仲介をイベントドリブンで実施できるかを検討する。	ブローカー製品自体には既存システムのインターフェースに合わせてプロトコルやデータ形式を変換する機能を具備していないケースが主である。ETL 等を併用することでインターフェース変換機能を実装するのが望ましい。
5	コスト	運用性・保守性等を加味した上で、ランニングコストを低減させるためにはどういった方法があるか。 【検証方針】運用保守における属人化/特定ベンダ依存を極力排除した、ブローカーの運用保守作業の効率化を可能とする機能/構成となっていることを検証する。	今回検証した全てのブローカー製品がパブリッククラウド上に構築可能なことが確認できた。パブリッククラウドの機能を活用することで、運用性・保守性を向上しつつランニングコストを効率化させることができる。 一方、ブローカー製品自体の管理機能が UI で提供されるケースは限定的であり、API 提供が主流である。運用性・保守性を考慮すると UI で管理可能なダッシュボード等を実装するのが望ましい。

No.	課題	課題概要/検証方針	考察及び課題
6	スケーラビリティ	都市間連携・分野間連携の連携数が将来的に増加した場合においても、リアルタイム性等のブローカー性能を維持可能なように拡張性を保つにはどのような方法があるか。 【検証方針】スケールアップ/アウトによる性能向上傾向確認及び、連携する都市数・分野数、他機能が性能に与える影響範囲の検証。	スケールアップ/アウトすることで性能が向上することが確認できたが、ブローカーのように数多くアクセスを分散して受け渡しする場合はスケールアウトの方がボトルネックが発生しにくく相性も良い。 詳細は性能評価の結果として 4-2-3(1)(b)参照。
7	パーソナルデータの利活用	地域課題を解決するために、プライバシーリスクを軽減しつつ必要最小限のデータ連携のもとで最大限の価値を発揮するためにはどのような方法があるか。 【検証方針】ブローカー製品がどのようにプライバシーリスクを軽減することが可能かの検証。	ブローカー製品はデータ（メッセージ）を仲介する機能に特化しており、オプトインや匿名化のようなパーソナルデータに特化した機能は具備しておらず、別機能との連携が必須である。 X-Road は事例からも機微なデータを取り扱うことが想定されているが、高度なセキュリティ技術に加え接続サービスを認可制にすることで制限された範囲でのデータ相互利用を実現している。

(b)性能評価

(ｱ)スケーラビリティ評価

ブローカーのスケールアップとスケールアウトに関して、結論としては、どのブローカー技術においてもスケールアウトの方が効率的であると考える。

FIWARE Orion では、スケールアップとスケールアウトにて、どちらも一定のスループットの性能向上が確認できたが、スケールアウトの場合、サーバ台数の増加に対しほぼ比例したスループット増加したため、スケールアウトの方が効率的であることが確認できた。Apache Kafka については、今回の評価でのスケールアップでは Consumer が 1topic から 1 リクエストで取得できるメッセージ数の上限によりスループットは変わらなかったが、スケールアウトではサーバ台数の増加に対しほぼ比例したスループット増加が確認できた。X-Road においては、スループット増加の効果はスケールアップ/スケールアウトのどちらでもほぼ同様で、性能/サーバ台数に比例してスループットが増加していくことが確認できたが、スループットが頭打ちになって以降、スケールアップの方が早く効率が悪くなっていく点が確認できた。

今回の性能測定では、ブローカーへのリソース影響として CPU、メモリの使用率、DiskI/O を確認したが、それらがボトルネックになっていることは確認できなかった。サーバのスケールアップに関しては、拡張できるリソースが CPU やメモリに限られ、その上限についてもサーバ環境に依存する。一方、スケールアウトについては、台数を増加することによりブローカー1 台に対する処理の競合が少なくなり、スループットの向上が見込める。また、ブローカー台数を増やすことで可用性強化（冗長化）にも効果的である。

(ｲ)データ仲介パターン評価

今回の評価では、認証処理追加によるオーバーヘッドのボトルネックよりブローカー性能が先

に来たため、認証処理を要因としたスループットの低下は確認できなかったが、認証やデータ変換の処理内容によっては全体の性能に影響する可能性もあるため、データ仲介の間に機能追加を行う場合はその影響を考慮する必要があると考える。

また、今回の評価では、データ仲介における、連携分野数の影響、及び都市間連携を想定した多段連携数による影響を評価した。連携分野数における評価については、全体のリクエスト数（データ量）は変えずに単純な分野数の増加によるブローカーへの影響を評価したが、スーパーシティで想定する連携分野数（5 分野）では問題にはならなかった。多段連携数による影響については、同期側のブローカーにおいて、段階数が増えたことにより、スループットの変化はなかったが、TAT が伸びていることを確認した。

データ仲介パターンによる各ブローカー技術への影響を以下にまとめる。

表 4-2-3-2　ブローカー技術ごとの仲介パターンによる影響

機能	パターン A （同期方式 Pull 型）	パターン B （非同期方式 Pull 型）	パターン C （非同期方式 Push 型）
周辺機能による影響	製品単体の機能では実現不可であるデータ変換等を他コンポーネントと組み合わせて実現する場合は影響を考慮する必要がある。	製品単体の機能では実現不可であるデータ変換等を他コンポーネントと組み合わせて実現する場合は影響を考慮する必要がある。	製品単体の機能では実現不可であるデータ変換等を他コンポーネントと組み合わせて実現する場合は影響を考慮する必要がある。
連携分野数による影響	スーパーシティで想定する連携分野数（5 分野）では問題にはならない。	スーパーシティで想定する連携分野数（5 分野）では問題にはならない。	スーパーシティで想定する連携分野数（5 分野）では問題にはならない。
多段連携数による影響	段階数が増えたことにより、スループットの変化はなかったが、TAT が伸びていることを確認した。	非同期方式 Pull 型のブローカー処理は、ブローカーの段数による性能影響は発生しにくい処理方式である。	非同期方式 Push 型のブローカー処理は、ブローカーの段数による性能影響は発生しにくい処理方式である。

(c)取り扱い可能なデータ

本項では評価結果を基に各製品がどのようなデータを取り扱うことができるか整理する。

対象データは 4-1 で定義したデータとする。

表 4-2-3-3　取り扱い可能なデータ

	メタデータ	データ本体					
静的/動的	－	静的			動的		パーソナル
データ形式	－	テキスト	地理空間	バイナリ	テキスト	動画 (ストリーム)	
FIWARE Orion	△(※4)	○	×(※1)	○(※2)	○	×	△(※3)
Apache Kafka	△(※4)	○	×(※1)	○(※2)	○	×	△(※3)
X-Road	△(※4)	○	×(※1)	○(※2)	○	×	△(※3)

※1 地理情報を配信する機能は具備していない。
　テキスト情報への位置情報の付加、バイナリデータとしての仲介は可能。
※2 製品ごとにサイズ制限がある
※3 取り扱いはデータ形式に依存する。パーソナルデータ用に特別な機能は具備されていない
※4 取り扱いはデータ形式に依存する。ブローカー自体はデータ本体を扱うものであるためメタデータ向けの特別な機能は具備されていない

(d)考察

仲介についてはデータ形式に関わらずブローカーで実現可能である。ただし、地理空間データなどデータ利用者側に必要な機能（配信や可視化）を具備している必要がある場合はブローカーで仲介しないこと望ましいデータも存在するため、そのあたりも加味した上でブローカーに蓄積すべきデータを検討すべきである。また、大きなサイズのデータの仲介については仲介自体に負荷がかかることに加えてブローカー全体の負荷となるためオンラインストレージへの蓄積なども検討すべきである。

(2)ブローカー技術の違いによる特徴

評価結果から得られたブローカー技術の違いによる特徴を記載する。
各ブローカー技術と製品の対応については、4-2-1 で述べたとおり。

- 同期方式　：Pull 型：X-Road 及び FIWARE Orion
- 非同期方式：Pull 型：Apache Kafka
- 非同期方式：Push 型：FIWARE Orion

本項では評価結果による各ブローカー技術について、機能/非機能面での違いを表 4-2-3-4、各ブローカー技術の特性について表 4-2-3-5 にまとめる。
特性については実際の利用を想定し、データ提供時、利用時への影響などを加味し以下 6 つの観点でまとめる。

- 主な特徴
- 性能/スケーラビリティ
- データ参照

- 到達保証
- データ発生からの到達時間
- クライアント負荷

表 4-2-3-4　ブローカー比較評価結果

項番	比較項目		FIWARE Orion	Apache Kafka	X-Road	考察
1	アーキテクチャ	方式	同期方式：Pull 型 非同期方式：Push 型	非同期方式：Pull 型	同期方式：Pull 型	表 4-2-3-5 参照
2	機能	データ分散	○	○	○	アーキテクチャが異なるが、データ分散方式に対応。
3		データ蓄積	○	△ 要ポーリング	× 分散方式が前提。	Orion のみイベントドリブン方式でデータを通知する機能を持つため、蓄積/イベント処理に対応可能。 Kafka はブローカーに溜めたキューを随時監視する機能を実装することで蓄積 DB への仲介やイベント処理に対応可能。
4		イベント処理	○	△ 要ポーリング	× イベント処理の仕組みがない。	
5	非機能 拡張性 セキュリティ	性能	○	○	×	Orion, Kafka と比較し、X-Road は性能が低い結果となった。X-Road の性能が低い理由は高度なセキュリティが備わっていることが要因と考えられる。
6		拡張性	○	◎	○	全ての製品がスケールアップ/スケールアウトにより性能拡張が可能。 スケールアップと比較し、スケールアウトの方が性能の伸びが良い。 Kafka は様々な大規模サービスで実績があり、大規模なデータ処理が可能。
7		セキュリティ	○	○	◎	Orion, Kafka は OAuth 等の認証方式を利用しセキュリティ確保が可能。 X-Road はディジタル署名、タイムスタンプ、ロギングなどによる高度なセキュリティ機能を有する。

表 4-2-3-5　ブローカー技術ごとの特性

項番	比較項目	同期方式：Pull 型	非同期方式：Pull 型	非同期方式：Push 型
1	主な特徴	利用者側が必要なタイミングで各システムへ要求を送信する方式。 同期方式のため、要求に対する応答を返却できる点が特徴。（データ検索、到達保証など）	Pub/Sub型のメッセージ仲介方式。 Push 型との違いは、利用者側が必要なタイミングで取得できるため、負荷平準化などが可能な点が特徴。	Pub/Sub型のメッセージ仲介方式。 Pull 型との違いは、データ発生時に取得することができるため、リアルタイム性が特徴。
2	性能/ スケーラビリティ	△ 非同期方式よりは性能が劣る。	○	○
3	データ参照	○	△ 購買手続きしたデータのみ。	△ 購買手続きしたデータのみ。
4	到達保証	○	△	△ 到達保証ができない。
5	データ発生からの 到達時間	△ データ更新検出にポーリング要。 ※利用者へ負担がかかるため非推奨 。	△ データ更新検出にポーリング要。	○
6	クライアント負荷	△ メッセージ量が多い場合に提供者側の負荷増。	○	△ メッセージ量が多い場合に利用者側の負荷増。

(3)抽出した課題に対する改善方針

ブローカー評価によって抽出された課題を以下に示す。

表 4-2-3-6　抽出された課題及び改善方針

No.	概要	改善方針
1	既存システム/アセットに合わせたインターフェース変換をする機能がない	ETL 等の機能を組み合わせることでインターフェース変換機能を実装し、既存システムへの影響を最小化する。
2	パーソナルデータの取り扱いに特化した機能がない	オプトイン等の機能を持つ製品と組み合わせることで、本人同意に基づく分散型データ連携を実現する。
3	連携対象分野数増加による性能低下	スケールアップ又はスケールアウトによりブローカの性能を拡張する。スケールアップには限界があるため、スケールアウトによる拡張を考慮した構成にするのが望ましい。
4	大容量データの連携	オンラインストレージ等にデータを格納し、データ格納先 URL 等をブローカーで仲介するのが望ましい。
5	運用管理機能の不足	ブローカーが提供する API を利用し、運用管理機能を実装する。

4-2-4. 評価結果から抽出されたブローカー要件

ユースケース評価、性能評価の評価結果を基に、ブローカーに求められる要件を抽出する。

(1)機能要件

ブローカーの機能について、2-2-1(1)で挙げた下記 4 つの分類で整理を行い、評価結果より得られたブローカーの機能要件を記載する。

(a)データ分散

まず、データの分散管理には、データ利用者からのリクエストがあった場合に分散されたデータ提供者への仲介をするというブローカーにおける基本的な機能が必要となる。その上で、利用者がデータの提供方式を意識せずに共通のインターフェースで利用できることが望ましい。

表 4-2-1-1 で示したとおり、データの分散方式にはいくつかのパターンがあり、扱うデータ数、データ利用者側/提供者の負荷等実際に利用するユースケースを加味して選択、もしくはいくつかのパターンを組み合わせることを検討する必要がある。

(b)データ蓄積

ブローカーとして、テキスト/バイナリ等形式を問わず仲介できる機能は具備することが望ましい。ただし、蓄積先としてブローカーを選択する場合はプライバシーインパクトへの配慮に加えてファイルサイズ過多によるブローカー性能の劣化を考慮する必要がある。また、データ特有の配信方法が必要なもの（動画、地理空間等）は専用機能で蓄積するのが望ましい。

(c)イベント処理

データ更新があった場合、リアルタイムに利用者へ通知するという機能は具備されていることが望ましい。これにより「必要な時に必要なデータ」を連携することが可能となる。

また、データそのものだけではなく手続き、サービス機能の呼び出しを仲介できることでサービス間連携を実現できると考えられるが、実現するためにはサービス側とブローカーのインターフェースの差異を吸収する仕組みが必要となる。サービス側のインターフェースが REST 等の一般的 Web インターフェースである場合はブローカーが実装することが推奨されるが、特殊なインターフェース実装や導入されるデータ連携基盤の方針によっては他の機能コンポーネントで実装する必要があると考える。

(d)その他（データ変換、API 仕様、業務機能、ユーザビリティ等）

(a)～(c)の実現を補完するための機能として分散されたデータ提供者のインターフェース、データ形式差異を吸収するデータ変換機能やデータ利用者がよりデータを扱いやすくなるための検索機能が望ましいと考える。これらによりデータ提供者やデータ利用者のデータ連携基盤への接続をある程度簡易にすることが可能になると考える。

(2)非機能要件

(a)運用性・保守性

データ連携基盤の継続的な発展には運用性が 1 つのポイントとなってくる。そのためにはブローカーの運用性への支援機能の具備、具体的にはデータ仲介の管理機能（データ連携の紐付け、連携ログの管理）等が必要であると考える。また、それらが属人的にならず継続的に改善していく仕組み（ドキュメントや改善事例の公開）が構築されていることが望ましい。

(b)性能・拡張性

徐々に連携分野数や都市数が多くなっていく性質があるデータ連携基盤において、性能の拡張性は必須となる。そのため状況に応じてスケールアップ/アウトできる仕組みが具備されていることが望ましい。

また、独自の要件に合わせて容易に他製品等との組み合わせが選択できるビルディングブロック方式が採用されていることが望ましい。

(c)セキュリティ

ブローカー単体として既存製品/技術で具備されているものは少ないが、データ連携を行う上でセキュリティ機能は必須となる。そのため、オプトインや匿名化などパーソナルデータを扱う機能や認証/認可などのアクセス制限を行う機能との連携が必須となる。

4-3. ブローカー要件（案）

本項では、ブローカーの機能要件及び非機能要件の策定、ブローカーにて取り扱うデータ種別の整理、エリアへの導入する際の方針について述べる。

ブローカー要件（案）として、ブローカー調査（4-1)、ブローカー評価（4-2）にて抽出した要件について、機能要件と非機能要件をまとめる。また、調査や評価における考察や、抽出された課題の改善方法についても反映する。ブローカー調査では、想定課題から調査観点を整理し、データ連携基盤に関する調査をユースケース（ニーズ）調査とリファレンスモデル・製品事例（シーズ）調査を実施した。ブローカー評価では、ユースケース（機能観点）及び性能（性能観点）の評価を実施した。

表 4-3-1　課題と調査及び評価の結果

No.	課題（調査観点）	検討方針	調査及び評価の結果
1	データ分析	各種サービスに対して物理的なデータの所在を隠蔽し、透過的なデータアクセスを実現する方法を検討する。	• スター型のデータ分散方式（FIWARE、Kafka 等）では、所在を隠蔽した透過的なアクセスを実現できる。一方、メッシュ型のデータ分散方式（X-Road 等）では、利用者側で所在を意識する必要がある。 透過的なアクセスという観点ではスター型のデータ分散方式を推奨する。 • 連携方式の観点ではリクエスト転送型と Pub/Sub 型があり、ノースバウンド（Northbound）ではリクエスト転送型が、サウスバウンド（Southbound）では Pub/Sub 型が適している。
2	汎用性	先端的サービスのユースケースや各種団体が公開するリファレンスモデル及び製品事例を調査し、ユースケースと技術動向の両面から汎用的かつ共通的なブローカー仕様案を検討する。	• ノースバウンド（Northbound）のインターフェースは REST 等の標準 API が主流である。 • サウスバウンド（Southbound）のインターフェースは多様な接続方式をサポートするケースが多く、既存アセットとの連携が考慮されている。
3	蓄積すべきデータ	先端的サービスのユースケースを基に、使用されるデータの種別、データサイズ、更新頻度等を整理し、ブローカーにて蓄積すべきデータを具体化する。	静的データ： 分散型が望ましいが、頻繁な変更が発生しないため蓄積型でも支障はない。 動的データ： 頻繁な変更が発生するため分散型が望ましいが、データ連携基盤がマスタデータとなる場合は蓄積型でも支障はない。 パーソナルデータ： プライバシーインパクトを考慮し、分散型が望ましい。 バイナリデータ： サイズや形式等でブローカーでの仲介に限界があり、オンラインストレージ等への蓄積が望ましい。
4	既存システムとの連携	既存システムへの影響を最小化し容易に接続可能な方式について、連携先システム・データ連携	• 既存システムとデータ連携基盤のインターフェース相互変換が必要であるが、本機能をブローカー製品が具備しているケースは少ない。ETL ツール等と連携す

No.	課題（調査観点）	検討方針	調査及び評価の結果
		基盤･ブローカー間の適切な機能分担も含めて検討する。また、利用者と他システムとの仲介という観点で、利用者への通知や、連携先システムへの機能呼び出し等メッセージの仲介をイベントドリブンで実施できるかを検討する。	ることでインターフェース変換機能を実現することを推奨する。 • データ利用者側から参照要求を出すケースに加え、データ発生時に利用者側へ通知するイベントドリブン方式のデータ連携機能が必要となるケースがある。
5	コスト	運用保守における属人化/特定ベンダ依存を極力排除した、ブローカーの運用保守作業の効率化手法を検討する。	• 分散型データ連携を実現する場合、データのアクセス先（所在）の管理が必要となるが、各ブローカー製品は管理機能を API で提供するケースが主であるが、運用保守性を考慮すると UI での管理機能があるのが望ましい。 • ライセンス費用が無償、カスタマイズの自由度、情報の検索性などの点で OSS 利用によるコストメリットがある。ただし OSS は無保証であるため、社会実装として安定運用を実現するためには保守体制の整備又はベンダサポート等の考慮が必要。
6	スケーラビリティ	都市間連携･分野間連携の連携数が将来的に増加した場合においても、リアルタイム性等のブローカー性能を維持可能なように拡張性を保ち、利用状況に応じたリソース拡張が柔軟に行えるよう、仮想化技術の活用について検討する。	• スケールアップと比較し、スケールアウトは線形かつ性能頭打ちが発生しにくくスケーラビリティが高い。冗長化による可用性向上も可能となるため、スケールアウト方式を推奨する。 • 接続する分野数などが多くなるにつれ TAT、スループットともに性能は劣化するが、スループットはスケールアウト方式により性能拡張が可能であるため、都市内でのデータ連携や複数都市（数都市）のデータ連携には問題ない（TAT は徐々に劣化する）。 • 都市間連携に関しては、全国の自治体を繋ぐような大規模な連携を実現する場合は別途都市間連携のアーキテクチャ検討が必要。
7	パーソナルデータの利活用	プライバシーインパクトが懸念されるデータや個人情報等をデータ連携基盤が集積しないように、原則的に、データ分散方式でのデータ管理を前提とする。	• ブローカー製品には同意管理等のパーソナルデータ活用に関する機能は具備しておらず、別途検討が必要 • 機微な情報に関してはオープンなデータ連携基盤ではなく、信頼されたシステム間でのみデータを共有するクローズなデータ連携基盤を併用する方式もある（X-Road 方式）。

ブローカーとしての機能面の拡張性という観点で、ビルディングブロックのアーキテクチャが望ましい。エリアへの導入時に、全ての要件を満たすブローカーを用意することは困難である。また、実装レベルが大きく変わる機能（セキュリティ、データ蓄積、データ変換等）について、エリアごとに適切なものを取捨選択できるべきである。そのためには、他機能と組み合わせできるようにビルディングブロックというアーキテクチャが良い。

4-3-1.　機能要件

ブローカーとは、様々な主体が提供するデータを集約し、適切な処理を経た上で公開する仕組みである。データ連携の目的として、「データ利活用」と「データ収集」の 2 種類があり、サービス側の用途に応じて使い分ける。それぞれ求められる機能が異なるため、各機能についてデータ連携目的を分けて整理した。

①データ利活用

データ利用者が適切な対象から必要なデータを必要なタイミングで取得する（同期方式 Pull 型）。例えば、「交通サービスから現在位置周辺にいるタクシーの情報を取得する」という場合や「交通サービスへタクシーの予約を要求する」という場合に使用する。

②データ収集

データ発生時にデータ提供者がデータを送信する（非同期方式 Push 型）。例えば、「交通サービスで予約したタクシーが現在位置付近に到着したら通知する」という場合に使用する。

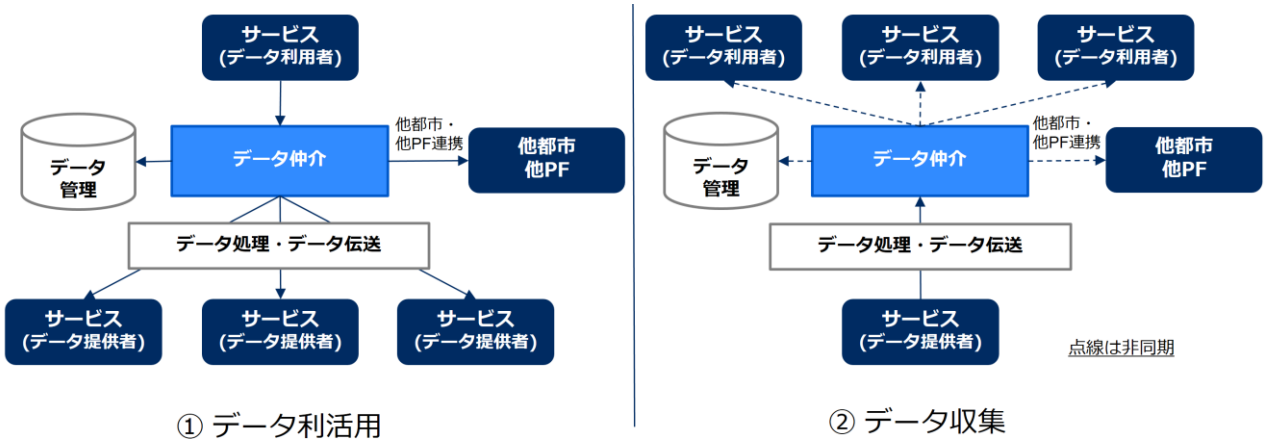


図 4-3-1-1　ブローカー機能概要図

データ利活用の要件には、データ参照・サービス呼び出し・API 仕様・データ変換がある。データ参照では、データ利用者がブローカーへデータ参照を要求すると、ブローカーは外部サービスにて保持しているデータを取得し、データ利用者へ返却する。ブローカーにて、データの所在管理が可能な場合、データ利用者は対象データを保持しているサービスを特定する必要がなく、透過的なアクセスが実現できる。また、データの保持先が外部サービスではなく、データ連携基盤内に蓄積したデータであっても、データ参照が実現できることが望ましい。サービス呼び出しでは、データ利用者がブローカーへあるサービスの機能呼び出しを要求すると、ブローカーは該当サービスの機能を実行し、データ利用者へ実行結果を返却する。データ参照とサービス呼び出しにおいて、データ利用者がブローカーの API を実行する。そのため、データ利活用の利便性が高い API を提供する必要がある（API 仕様）。なお、外部サービスとの接続時には、対象のサービスのインターフェースに合わせたデータ変換が必要な場合がある（データ変換）。

データ収集の要件には、データ更新・API 仕様がある。データ更新では、データ発生時にデータ提供者がブローカーへデータを送信すると、ブローカーは必要としているサービスへ該当データを送信する。データ提供時にリアルタイムに加工・分析・変換等の処理が必要な場合がある。また、データの送信先が外部サービスではなく、データ連携基盤内のデータストア機能等への蓄積であっても、データ更新が実現できることが望ましい。データ発生源となるセンサーなどがデータ更新の API を実行する想定のため、様々な接続方式への対応が求められる（API 仕様）。

上記のとおり、抽出した要件を機能一覧にまとめた（表 4-3-1-1)。分類は、表 2-2-1-2 に基づいて、整理した。必須/推奨については、ブローカーが具備しなければデータ連携基盤の原則であるデータ分散方式でのデータ利活用を実現できない機能を必須、他サービス等の連携で代替可能と考えられる機能を推奨とした。

表 4-3-1-1　機能一覧

データ連携目的	要件名	分類	説明	必須	推奨
データ利活用	データ参照	データ分散	データ参照の要求を受付け、外部サービスが保持するデータを返却可能なこと	○	
			データ利用者に対してデータの所在を隠蔽することができること		○
		データ蓄積	データ参照の要求を受付け、データストア機能に蓄積されたデータを返却可能なこと		○
	サービス呼び出し	イベント処理	サービス呼び出しの要求を受付け、外部サービスの処理を実行し結果を返却可能なこと（例：交通サービスでタクシーを予約）		○
	API 仕様	API 仕様	データ利活用の利便性を考慮し、標準ルールに沿った API（REST 等）を提供可能なこと	○	
	データ変換	データ変換	外部サービスへの接続時、接続先サービスのインターフェースに合わせたデータ変換が可能なこと		○
データ収集	データ更新	イベント処理	データ提供者からデータを受付け、必要なサービスへデータを送信できること		○
			データ送信時、リアルタイムにデータの分析・変換・加工処理等が可能なこと		○
		データ蓄積	データ提供者からデータを受付け、データストア機能に蓄積可能なこと		○
	API 仕様	API 仕様	多種多様なアセットからのデータ収集を想定し、標準 API に限らず様々な接続方式に対応可能なこと（MQTT 等）		○

必須：本機能がないとデータ分散方式でのデータ利活用が実現できないもの（コア機能）
推奨：あるのが望ましいが、他サービス等との連携で代替可能なもの

4-3-2. 非機能要件

想定課題を基に、運用・保守性、性能・拡張性、セキュリティという観点で、ブローカーの非機能要件を整理した。

表 4-3-2-1 非機能一覧

分類	要件名	概要	必須	推奨
運用・保守性	サポート体制	サポート体制が整備されていること 社会実装として安定運用を実現するためには保守体制の整備又はベンダサポート等の考慮が必要	○	
	運用管理 I/F	UI での運用管理機能があること		○
	情報開示（ドキュメント整備）	保守作業の効率化のため、構築・運用・利用に関する情報が入手可能であること		○
性能・拡張性	性能	• 5 分野以上の先端的サービス間のデータ連携ができること • 2 段以上のデータ仲介ができること （接続する他都市のブローカーを経由して他都市の先端的サービスへアクセスする想定）		○
	リソース拡張	利用状況に応じた柔軟なリソース拡張が行えること （スケールアップ・スケールアウトなど） ※スケールアウトの実現を推奨	○	

セキュリティについて、ブローカーの要件としては定義しない。パーソナルデータを取り扱う際、同意管理・匿名化等のプライバシー保護に関する機能が必要である。ただし、データ連携基盤内のブローカーではない別の機能にて実現することが妥当である。

4-3-3. ブローカーにて取り扱うデータ種別

ブローカーにて取り扱うデータの種別を整理し、各種データがデータ分散方式又はデータ蓄積方式のどちらでの管理が適しているかを整理した。データ管理主体がデータを保持すべきという考え方で、データ種別問わず原則不可とした。ブローカー調査から、データ連携基盤において取り扱うデータのデータ種別を抽出し、ブローカー評価から、ブローカーにおいて取り扱うデータのデータ種別を選択した。

表 4-3-3-1 ブローカーにて取り扱うデータ種別

	メタデータ	データ本体					
静的/動的	—	静的			動的		パーソナル
データ形式	—	テキスト	地理空間	バイナリ	テキスト	動画（ストリーム）	
代表的なデータ例	所有者情報	避難所情報	地形図	観光地写真	水位センサー情報	河川監視画像	行動履歴
データ分散方式	対象外（※1）	必須	対象外（※2）	必須	必須	対象外（※3）	必須
データ蓄積方式	対象外（※1）	原則不可（※4）	対象外（※2）	原則不可（※4）	原則不可（※4）	対象外（※3）	原則不可（※4）

必須：スーパーシティのブローカー機能として取り扱い必須

原則不可：原則として取り扱い不可ではあるが、特定の条件下でのみ取り扱い可であるもの

対象外：ブローカー機能の対象外

（※1）：データカタログ等の専用機能で管理されるため、ブローカー機能の対象外

（※2）：GIS 等の専用機能で管理されるため、ブローカー機能の対象外

バイナリデータとしてデータファイル自体を取り扱うことは可能

（※3）：VMS 等の専用機能で管理されるため、ブローカー機能の対象外

（※4）：データ分散方式を原則とするが、データ連携基盤運営者が保持するデータであればデータ蓄積方式も可能とする

データ分散方式とデータ蓄積方式はトレードオフの関係になっているため、上記分析を参考に導入するエリアによって判断する。データ連携基盤は、データ分散方式のデータ管理を原則としている。また、データ蓄積方式は、蓄積したデータを管理するコストやデータ漏洩のリスクなどが増大するというデメリットもある。しかし、データ分散方式に比べデータ蓄積方式の方がレスポンスが早い、該当エリアにおいて蓄積する価値があるデータなどのメリットも考えられる。

4-3-4.　導入時の方針

あるエリアにデータ連携基盤を構築するにあたり、ブローカーを導入する際の方針について整理する。上述の要件を全て満たしたブローカーを用意することができれば良いが、現実的には製品仕様上の制約や導入又は運用の費用との兼ね合い上、困難である。ケース（用途）別の機能を整理することで、導入時のブローカーの実装すべき機能を取捨選択できるようにすることが目的である。

(1) ケースの例

データ連携基盤又はブローカーが導入される際に、解決すべき課題によって、必要な機能は異なる。以下のような 3 つのケースにおいて、必要な機能を明確にする。

ケース 1：ブローカーを経由して各サービスからデータを取得する
ケース 2：ブローカーを経由して各サービスへデータを送信する
ケース 3：データ連携基盤にデータを蓄積する

(a) ケース 1

ケース 1 は、ブローカーを経由して各サービスからデータを取得する事例である（データ利活用）。

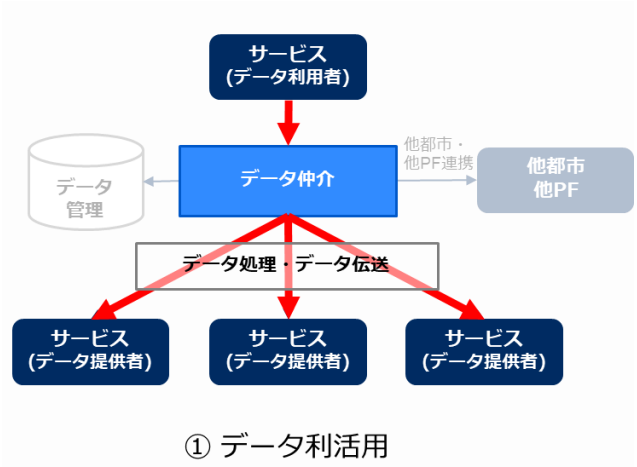


図 4-3-4-1　ケース 1 の機能概要図

具体的な事例として、ユースケースとブローカー製品を用いて説明する。

ユースケースの事例としては、3-1 で挙げられている『参考２．「スーパーシティ」構想イメージ』[14]が該当する。例えば、後期高齢者の通院対策の場合、データ利用者は「高齢者通院サポートサービス」であり、データ提供者は病院やタクシー事業者にあたる。ブローカーによって、データ利用者から、配車可能なタクシーの検索（データ参照）や、病院の通院予約（サービス呼び出し）など適切なデータ提供者に対してデータの連携を行うことができる。

ブローカー製品の事例としては、4-1-2 で挙げられている「X-Road」が該当する。データ利用者からのデータ要求に対して、データ提供者がデータを返却する。X-Road は、このケースをセキュアな通信で実現していることが特徴の製品である。

[14] https://www.kantei.go.jp/jp/singi/tiiki/kokusentoc/supercity/supercity.pdf

(b) ケース 2

ケース 2 は、ブローカーを経由して各サービスからデータを取得する事例である（データ収集）。

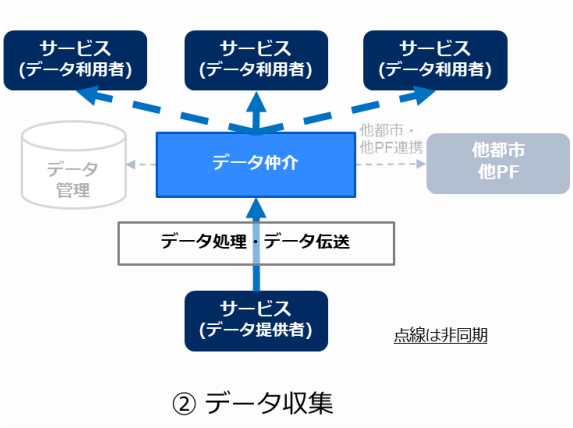


図 4-3-4-2　ケース 2 の機能概要図

具体的な事例として、ユースケースとブローカー製品を用いて説明する。

ユースケースの事例としては、4-1-1 で挙げられている「リスボン・インテリジェント・マネジメント・プラットフォーム」及び「杭州 ET City Brain」が該当する。どちらのユースケースにおいても、データ利用者は警察、救急などのシステムであり、データ提供者は監視カメラなどのセンサーにあたる。データ提供者から定期的に送信されるデータから、緊急事態が発生したデータのみをデータ利用者へ、通知（データ更新）を行う。

ブローカー製品の事例としては、4-1-2 で挙げられている「FIWARE」が該当する。上記の「リスボン・インテリジェント・マネジメント・プラットフォーム」は、FIWARE を活用し、実現されている。

(c) ケース 3

ケース 3 は、データ連携基盤にデータを蓄積する事例である。

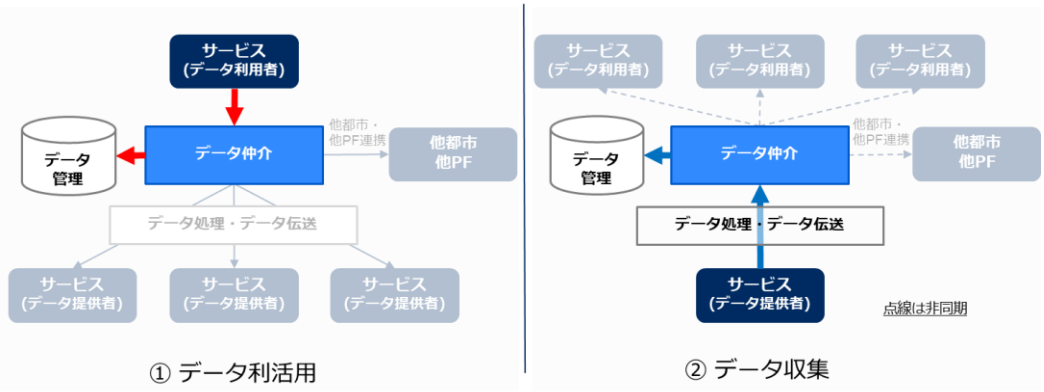


図 4-3-4-3　ケース 3 の機能概要図

具体的な事例として、ユースケースとブローカー製品を用いて説明する。

ユースケースの事例としては、4-1-1 で挙げられている「Neighborhoodscout」が該当し、ブローカー製品の事例としては、4-1-2 で挙げられている「MindSphere」が該当する。どちらの事例も、デー

タ収集として、データ提供者であるセンサーからのデータをデータ連携基盤内に蓄積し、データ利活用として、蓄積データからデータ利用者の要求に即したデータを返却する。蓄積データに対して、独自の分析技術で付加価値を与える等の機能を活用する事例もある。

(2)用途別の必要機能

ケースによって、必要となるブローカーの機能を抽出する。

表 4-3-4-1　用途別機能要件

データ連携目的	要件名	分類	説明	ケース別要件（○：対象、―：対象外）		
				ケース 1	ケース 2	ケース 3
データ利活用	データ参照	データ分散	データ参照の要求を受付け、外部サービスが保持するデータを返却可能なこと	○	―	―
			データ利用者に対してデータの所在を隠蔽することができること	○	―	―
		データ蓄積	データ参照の要求を受付け、データストア機能に蓄積されたデータを返却可能なこと	―	―	○
	サービス呼び出し	イベント処理	サービス呼び出しの要求を受付け、外部サービスの処理を実行し結果を返却可能なこと（例：交通サービスでタクシーを予約）	○	―	―
	API 仕様	API 仕様	データ利活用の利便性を考慮し、標準ルールに沿った API（REST 等）を提供可能なこと	○	―	○
	データ変換	データ変換	外部サービスへの接続時、接続先サービスのインターフェースに合わせたデータ変換が可能なこと	○	―	―
データ収集	データ更新	イベント処理	データ提供者からデータを受付け、必要なサービスへデータを送信できること	―	○	―
			データ送信時、リアルタイムにデータの分析・変換・加工処理等が可能なこと	―	○	―
		データ蓄積	データ提供者からデータを受付け、データストア機能に蓄積可能なこと	―	―	○
	API 仕様	API 仕様	多種多様なアセットからのデータ収集を想定し、標準 API に限らず様々な接続方式に対応可能なこと（MQTT 等）	―	○	○

導入するエリアによって、解決すべき課題や導入後の利用方針によって、適した組み合わせでブローカーを実装することが求められる。上記以外にも、同意管理・匿名化等のプライバシー保護に関する機能や、データを蓄積する目的や手段などは、必要可否を含めて、導入時に検討する必要がある。このように、ブローカーとしての機能面の拡張性という意味で、他機能と組み合わせられるようにビルディングブロックのアーキテクチャが望ましい。

5. 調査結果（イ）①API 共通ルール/標準仕様

5-1. API 設計/開発・公開・運用プロセス

5-1-1.　はじめに

本節では、スーパーシティに具備される API の設計/開発・公開・運用のプロセスについて、それぞれ望ましい在り方を示す。スーパーシティでは、様々な API が多数のサービスに利用され、サービスやシステムが API で相互に接続・連携するような形態が想定されるため、API の再利用性と互換性の確保が重要となる。今後スーパーシティの整備に向けて、多数の API が新規に開発・運用されることを想定し、設計/開発の段階からその望ましいプロセスを示すことで、API の品質の均質化と安定した API の提供を推進することが本節の目的である。なお、本節で述べる内容は、データ連携基盤に関する API の開発と運用を行う運営主体や事業者（API を提供する者）を対象読者とする。提供された API を利用して先端的サービスを提供する事業者（API を利用する者）は、本節で取り扱うドキュメントを参照してそれぞれ開発を進めることを想定する。

API の設計/開発・公開・運用プロセスは、次のよう分解される。各プロセスにおける実施事項の詳細を後述する。

1. 設計/開発プロセス（「5-1-2 設計/開発プロセス」で説明）
   (1) 計画：既存 API の有無を調査し、関係者と連携しながら開発方針を策定する。
   (2) 要件定義：サービス機能に合わせて必要となる API 機能を定義する。
   (3) 設計：OpenAPI 仕様に準拠したインターフェースとデータ項目を設計する。
   (4) 開発：策定した仕様に準拠するインターフェースとビジネスロジックを開発する。
   (5) テスト：サービス向けの公開を前提とした観点で成果物をテストする。
2. 公開プロセス（「5-1-3 公開プロセス」で説明）
   (1) API の公開と廃止：開発した API の提供を開始又は終了する。
   (2) ドキュメントの公開：OpenAPI 仕様ドキュメントと利用規約を公開する。
3. 運用プロセス（「5-1-4 運用プロセス」で説明）
   (1) 監視とモニタリング：API ログ、稼働状況、性能、セキュリティの観点で監視する。
   (2) 障害対応：障害の検知、原因調査、復旧対応を行う。
   (3) サポート対応：API 利用者からの Q&A 対応や、フィードバックを受けて改善を検討する。

以降では、スーパーシティにおいて望ましい API 設計/開発・公開・運用プロセス及びその考え方について、1. API の設計/開発プロセス、2. API の公開、3. API の運用にプロセスをさらに詳細に分解し、それぞれを説明する。

5-1-2.　設計/開発プロセス

(1)計画

(a)既存 API の調査

サービスの求める機能を提供している API が既に存在する場合、新規に API を開発・提供する必要はない。そのため、API の開発を検討する前に、サービスが求める API の有無を事前に調査することが望ましい。他の地域や団体が提供する既存の API は、内閣府が整備を予定しているスーパーシティ API カタログや、民間 API の情報提供サイトを参照することで調査が可能である。

(b)ステークホルダとの連携

スーパーシティに関する API の整備は、地域内のサービス提供者、API 提供者、API 運用者、データ提供者をはじめとして、他の地域と連携する場合はその接続性や互換性などの相互運用性を考慮すべき事業者など、多数のステークホルダが存在する点で一般的な API の整備と異なる。そのため、これら多数の関係者がいる中でスムーズに API を整備するため、日常的に連携自治体間でサービスやデータの連携に関するコミュニケーションをとり実現したい目的を共有・具体化したり、既存 API の開発や運用の実績を取り入れることで、相互運用性の担保をより容易に実現することが重要となる。

(c)開発手法の選択

(ア)ウォーターフォール型開発とアジャイル開発

近年のシステム開発は、要件定義、設計、開発、テストと一連の開発工程を進めていくウォーターフォール型の開発に加え、短期間で細かく機能をリリースしながら開発を繰り返していくアジャイル型の開発手法も広く採用され始めている。一般的に、ウォーターフォール型開発では開発計画や予算見積が容易である一方で、後続作業の中で要件・仕様に変更が発生した場合、手戻りが大きくなる可能性がある。アジャイル開発はスプリントの単位で一定柔軟に開発を進めることができるが、要件の詰め込みと開発チームの生産性のバランスによっては期間内に目標とする成果物が完成しないなどのリスクがある。

スーパーシティにおける API の開発においては、ウォーターフォールとアジャイルのどちらの手法を選択してもよい。選択にあたっては、実際に API 開発に携わる開発者の意向や開発の規模、そして発注を伴う際にはその調達発注に必要となる契約や手続きとの兼ね合いを考慮し、効率的な開発手法を選択するべきである。なお、自治体では単年度で予算を策定することが基本となっており、先に記述した要件と開発生産性のバランスにより目標を達成できない可能性のあるアジャイル開発を選択しづらい背景もある。アジャイル開発を選択する際にはサービスの開発に合わせて API の設計・開発を短サイクルで実施することになるが、一般的に API はサービス開発の前提となる。そのため、API 開発作業とサービス開発作業が先行後続の直列関係となることで、全体の工期が伸び得る。そこで（イ）API 開発のアプローチで解説する内容を採用することで工期を短縮できる可能性がある。また、アジャイル開発の契約については、情報処理推進機構(IPA)より公開されている契約書の雛形[15]に従って進めることが望ましい。

なお、自治体では単年度で予算を策定することが基本となっており、組成した開発チームの開発速度によっては、目標を達成できない可能性のあるアジャイル開発を選択しづらい背景もある。アジャイル開発を選択する際には、サービスの開発に合わせて API の設計・開発を短サイクルで実施する。一般的に API はサービス開発の前提となるため、先行して開発する必要があるが、作業の先行後続関係となることで、全体の工期が伸び得る。そこで（イ）API 開発のアプローチで解説する内容で工期を短縮できる可能性がある。また、アジャイル開発の契約については、情報処理推進機構(IPA)より公開されている契約書の雛形[16]に従って進めることが望ましい。

[15] 情報処理推進機構(IPA)　アジャイル開発版「情報システム・モデル取引・契約書」
https://www.ipa.go.jp/ikc/reports/20200331_1.html

[16] 情報処理推進機構(IPA)　アジャイル開発版「情報システム・モデル取引・契約書」
https://www.ipa.go.jp/ikc/reports/20200331_1.html

(ｲ)API 開発のアプローチ

- コードファーストによる開発
  プログラムフレームワークの中に OpenAPI 仕様の自動生成機能を組み込むことで、開発を先行して進めると自動的に OpenAPI 仕様が作成される。
- コントラクトファーストによる開発
  プログラム開発に先立って OpenAPI 仕様を作成し、その仕様ファイルからプログラムコードやテストを自動生成し、機能の開発を進めていく。

どちらの方法でも、開発する API のインターフェース仕様を記述する OpenAPI 仕様準拠のドキュメント整備を前提とした開発手法であり、API をカタログ化して再利用を促すスーパーシティの思想に合致する方法である。これまではコードファーストのアプローチが主流だったが、複数の開発者が並行して実装を進めると、独自実装が生まれ統制を取ることが難しいなどの問題から、現在ではコントラクトファーストでの開発手法が採用される事例もある。

また、コントラクトファーストによる開発では API の仕様が早期に定まるため、API を利用するサービスの開発者に対し、OpenAPI 仕様から自動生成したモック API を提供することができる。モック API は具体的な機能を提供せず、サービスから API を呼び出すインターフェースのテストに利用することができ、API 完成前にサービスとの連携開発に着手することができる。サービスの開発期間に制約がある場合は、このような方法をとることで全体の工期短縮を目指すこともできる。

(d)API の整備をサポートするツールの導入検討

次のようなツールを活用することで API の開発期間を短縮できる場合がある。以下に具体的なツール例（2021/3/1 調査時期時点）を示すが、API を整備するプラットフォームや開発フレームワーク等の制約がある場合があるため、導入を検討する際は、API の設計、開発、公開、運用などの各フェーズにおけるフローを具体化し、ツールの適用範囲を検討した上で導入することが望ましい。

- API 設計ツール（例：Apicurio Studio、Swagger Editor など）
  GUI ベースで OpenAPI 仕様などの標準フォーマットに合わせて API 仕様を記述することができるツール。
- モック API 生成ツールとテストツール（例：Microcks、prism、Postman など）
  テスト用のモック API の生成や API の実行を簡単に行うことができるツール。
- サービス開発フレームワーク（例: Django REST framework、Jersey、Lumen など）
  API 開発で必要となる共通的な機能（コード）の集まり。
- デプロイメントツール（例：Maven、Capistrano、gradle など）
  API プログラムのソースコードを実行可能な形式に変換し、API の稼働サーバに配置・展開するためのツール。
- インフラ管理ツール（例：Terraform、Ansible、Chef など）
  API の稼働する基盤（インフラ）の構築や変更作業を自動化するためのツール。
- 仮想・コンテナ環境管理ツール（例：KVM、Kubernetes、OpenShift など）
  API の稼働するサーバやアプリケーションの実行環境を構築、管理するためのツール。
- CI/CD プラットフォーム（例：Jenkins、GitLab CI、GitHub Actions など）
  API 開発で必要となる作業（変更管理、テスト実行、リリース 等）の管理や自動化をするため

のツール。

- API ゲートウェイ（例：Kong、AWS API Gateway、Azure API Management など）
  API 利用者からのアクセス（通信）を集約して受付け、個々の API が動作するサーバへ中継するためのツール。API に共通で求められる機能（セキュリティ、トラフィック制御・分析、ロギング 等）が提供される。

(2)要件定義

(a)機能要件の検討

スーパーシティにおける API 利用者は、サービスを開発・提供する事業者である。そのため、様々な先端的サービスで共通的に必要とされる機能を API として提供することが望ましい。データの提供や連携にあたっては、データの公開範囲に基づき決定する。API 利用者の権限やデータ所有者の本人同意（オプトイン）に基づく必要最少権限でのデータ連携を実現する認証・認可方式への対応が必要となる。

機能要件の検討は、API として提供する機能（データ種、データ操作方法など）を整理することになるが、API 利用者視点で検討することが重要である。API 利用者である先端的サービスが、どのようなデータ種を、どのような単位で、どのように利用したいのかを整理し、API の提供単位や必要機能を明確にして一覧化を行う。API ごとに返却するデータ種・データ操作方法だけでなく、データ粒度に関しても合わせて検討することが重要である。

例えば､施設情報の検索機能について考える場合､検索結果に施設 ID の一覧のみが 20 件返却され、施設名や利用用途などの情報が含まれない場合に施設一覧画面を生成するためには、1 回の検索 API に加えて 20 回の施設情報取得 API を呼び出す必要がある。API の呼び出し回数が増えると、Web ページを生成する時間が長くなってしまう。そのため、検索結果には条件に合致する施設 ID に加えて、施設名や利用用途などの施設情報も含めて返却することが望ましい。

(b)非機能要件の検討

API に関する非機能要件については、情報処理推進機構(IPA)が公開している非機能要求グレードを基に検討する。非機能要求グレードにおける社会的影響の程度は、整備する API を利用するサービスによって異なるが、API カタログで公開されることを考慮して非機能要件の程度を検討する。特にスーパーシティで整備する API では、その機能や取り扱うデータの特性によって、公開範囲を検討する必要がある。

「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」では、API の開放性について 5 つの段階と 2 つの類型が定められている。同報告書では、図中「Public」「Acquaintance」の 2 つを狭義のオープン API[17]、これらに「Member」「Partner」を加えた 4 つを広義のオープン API[18]と定めている。スーパーシティで整備される API は、可能な限り狭義のオープン API として公開し､API を利用するサービスに対して適切な開放性とアクセス権を定める｡この検討には、API を介して提供されるデータの所有者やプライバシー、あるいは API を利用するサービスがもたらす影響の大きさ（人命に関わる場合など）を加味して進める必要がある。

[17] 狭義のオープン API：厳格な要件や契約などを必要とせず、外部から誰でもアクセス可能な API のこと。

[18] 広義のオープン API：特定の団体が外部に公開する API のこと。アクセスする際に契約行為などが交わされるケースも含む。

図 5-1-2-1　オープン API の類型

（出典：「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」）

広義のオープン API の公開範囲を設定するために、アクセストークンを利用する方法がよく採用される。アクセストークンの利用により、アクセス制御やログ取得など、API を利用するサービスに対して細やかな制御が可能となる。一方で、トークンの配布と利用が前提となるため、API 利用の敷居がやや高まる。API 公開範囲の設計を行う際には、API を利用するサービス事業者の接続コストについても考慮することが望ましい。

(3)設計

(a)インターフェースとデータモデルの設計

(ｱ)OpenAPI 仕様の作成

API の要件が明確になったら、API が提供するインターフェースと取り扱うデータ種を記述する API 仕様を作成する。ここ数年で行われた標準化の試みにより、2021 年現在、OpenAPI 仕様が API のインターフェース仕様を記述する業界標準となっている。また、API 設計者、開発者、API を利用するサービス提供者が OpenAPI 仕様を共有して開発を進めることで、API の仕様についての統一的な理解を基に開発できるため、API 仕様の記述には OpenAPI 仕様を採用する。

OpenAPI 仕様には、一般に表 5-1-2-1 の情報が記載される。

表 5-1-2-1　OpenAPI 仕様に記載する主要な情報

記載項目	内容
基本情報	API の名称、API が提供する機能の概要、API 利用時の留意事項（実行回数制限など）、利用規約又はそのリンク先情報、API のバージョンなどを記載する。
サーバ情報	API サーバの URL とその説明（開発用、本番用など）を記載する。
認証方式	HTTP 基本認証、API キー、OAuth 2.0、OpenID Connect など、API を利用するために必要な

	認証方式に関する説明を記載する。
API 機能に関する情報	API サーバが提供する機能ごとに、URL パスと HTTP メソッド（GET、POST、PUT、PATCH、DELETE など）の組で記載する。各 API 機能には、URL パラメータ、HTTP ヘッダ、送受信するデータモデル、API 実行結果を示すステータスコード（200、201、403、500 など）を合わせて記載する。
データ種・構造	API で取り扱うデータ種・構造を記載する。データ種を構成する属性値や属性値が必須か任意か、及びその説明を合わせて提示する。

OpenAPI 仕様は API を利用するサービス開発者に参照されるため、サービス開発者が利用しやすいものであることが望ましい。また、整備した API は API カタログに登録され、多数のサービスが利用する可能性がある。そのため、後述の「5-3 API 標準仕様案」を参照し、業界標準のデータモデルやアーキテクチャを採用するよう留意する。

(ｲ)OpenAPI 仕様に関する支援ツールの利用

OpenAPI 仕様は JSON もしくは YAML のフォーマットで記述され、特定の言語やフレームワークに依存しない。そのため、多数のサードパーティ製ツールが存在しており、様々な言語やフレームワークで活用できる。

以下に OpenAPI 仕様を基にして自動生成できるものを例示する。

- サーバプログラムの雛形

  OpenAPI 仕様で定義された API の URL パスやメソッドに合わせたサーバプログラムの雛形を生成できる。ビジネスロジックを追加実装することで、API サーバの開発を省力化できる。
- API を呼び出すクライアントプログラムの雛形

  OpenAPI 仕様に定義された API 機能を呼び出すためのクライアントプログラムの雛形を生成できる。ビジネスロジックを追加実装することで、API クライアントプログラムの開発を省力化できる。
- モック API サーバ

  OpenAPI 仕様に合わせた URL パスやメソッドでリクエストを待ち受け、ダミーのレスポンスを返却する HTTP サーバを生成できる。API を利用するプログラムや、API のテストケースの作成に利用できる。
- API ドキュメント

  JSON や YAML で記述された OpenAPI 仕様ファイルから、人間に読みやすいフォーマットのドキュメントを生成できる。ブラウザから API のテスト実行を行える Web ドキュメントを生成するものもある。
- インターフェースのテストプログラム

  開発した API サーバプログラムが OpenAPI 仕様に準拠していることを確認するためのテストケースと、そのテストを実行するプログラムを生成できる。

例えば、OpenAPI 仕様から API ドキュメントを自動生成するツールを利用することで、API が提供する機能と使い方の例が記述された人間が読みやすい API ドキュメントを、統一されたフォーマットで生成できる。その結果、API 開発者は API のドキュメントを個別に作成する必要がなくなり、API ドキュメントの品質が均質化される。

(b)セキュリティの設計

(ｱ)資格情報による認証・認可

先に述べた API の公開範囲を実現するためには、API を呼び出す資格の有無を確認する機能が必要となる。一般的に、API 呼び出しに対する認証・認可を設定する仕組みには HTTP Basic 認証、OAuth 2.0、OpenID Connect などが存在する。スーパーシティで整備する API では、柔軟な認証・認可の仕組みを提供することを目的として、後述の「5-3 API 標準仕様案」に基づき OpenID Connect を採用する。

OpenID Connect では、認証・認可のための資格情報は API サーバとは別に用意される認証プロバイダに格納されており、API と連携して資格情報を確認する。スーパーシティでは多数のサービスが開発され、それらが利用する API も多岐にわたることが想定されるため、個別のサービスに対する API 認証・認可設定の運用は煩雑なものとなる可能性があるが、設定が一元的に管理されていれば、ある API の認証・認可の方法や範囲が変更された場合に、影響を受けるサービスを特定することが可能となる。

(ｲ)一般的な攻撃に対する対策

API を公開すると、様々な地域から脆弱性をつくことを目的とした攻撃リクエストが発行されることが想定される。API の呼び出し元が特定の IP アドレスや地域に限定可能な場合は、IP アドレス範囲を指定して許可/拒否することができる。また、特定のデバイスやユーザ単位で時間あたりの呼び出し回数を制限することで、API へのリクエスト負荷を緩和することができる。提供する API の想定利用方法に合わせて、このような対策を複合して実施することで、一般的な攻撃に対する耐性を高めることができる。

その他、Open Web Application Security Project (OWASP)では、Web アプリケーションのセキュリティを向上させる上で役立つ資料や情報が公開されている。特に、OWASP TOP 10[19]と呼ばれるレポートや OWASP Cheat Sheet Series[20]と呼ばれるチートシートでは、流行しているセキュリティリスクや攻撃に関する情報や対策がわかりやすくまとめられており、API の設計を進める上で参照することが望ましい。

(ｳ)オリジン間リソース共有(CORS)の設定

一般的に、API リクエストの発行元が Web ブラウザになる場合、セキュリティ上の理由から JavaScript 等のスクリプトによる呼び出しは Web サイトと同じドメインに制限されている。サービスの設計上、Web ブラウザから直接 API を呼び出す場合は、オリジン間リソース共有(CORS)の設定を有効化する必要がある。この設定は API 側で実施する必要があるため、API を利用するサービス側にこのような要件がある場合に考慮しておく必要がある。

(c)モック API サーバによる OpenAPI 仕様のテスト

(ｱ)OpenAPI 仕様を基にしたモック API サーバの生成

API の開発とサービスの開発を並行して実施する必要がある場合、OpenAPI 仕様に準拠した動作をするモック API サーバを用意し、サービスの開発に役立てることができる。モック API サーバは、OpenAPI 仕様に記述されたインターフェースを備えるダミーの API サーバプログラムで、API サーバの内部ロジックが実装されていない段階でも、API を呼び出すプログラムの開発を進

[19] https://owasp.org/www-project-top-ten

[20] https://cheatsheetseries.owasp.org

めることができる。また、モック API サーバに対する API 呼び出し方法がサービス開発側の想定と異なる場合、API 本体の実装に入る前にフィードバックし、仕様を修正することで、手戻りの発生を防ぐこと可能である。モック API サーバは OpenAPI 仕様を基に手動で実装することもできるが、OpenAPI 仕様を入力としてモック API サーバを自動生成するツールも公開されている。このような自動生成ツールを利用したモックの生成とインターフェースのテストを、API の内部ロジックを実装する前の段階で実施することが望ましい。

(ｲ)モック API サーバを介した API 仕様のテスト

サービスがまだ開発されていない段階であっても、API を呼び出す順序やデータの想定があれば、Postman 等の API 呼び出しツールや curl 等のコマンドを利用することで、モック API サーバに対するリクエストを発行できる。発行するリクエストと、期待するレスポンスの組をあらかじめ定義しておけば、サービスが API を呼び出す順序や取り扱うデータの処理方法などを事前に想定でき、サービス開発を円滑に進めることができる。

また、モック API サーバが API ゲートウェイを介してアクセスできる状態にあれば、API 機能の開発ができた段階で切り替えることができる。API ゲートウェイは多数ある API を管理し、各 API で利用される共通機能を一元的に提供するサーバである。API ゲートウェイで特定の URL に対する API の呼び出し先を切り替えることで、モック API サーバの API を呼び出していたサービスの設定を変更することなく、必要機能が実装された API に切り替えることができる。

(ｳ)モック API サーバと CI/CD ツールを利用したサービス開発

モック API サーバが用意されていれば、API 機能が未実装の状態でもサービスの開発を進めることができる。サービス開発が継続的インテグレーション/継続的デリバリ(CI/CD)を支援するツールと連携して進められている場合、CI/CD のテストフェーズでサービスが正しく API を呼び出し、そのデータを処理しているかを確認することができる。API 機能が実装された後、モック API サーバから切り替えられれば、実動作する API を利用したテストも実施することができる。

(4)開発

(a)OpenAPI 仕様からの雛形プログラム生成

(ｱ)プログラミング言語とフレームワークの選定

API 開発では、多数のプログラミング言語で開発フレームワークやライブラリが用意されており、これらを活用することが短期間、低コスト、高品質な API を整備する上で重要となる。言語やフレームワークの選定は、技術の潮流、プロジェクトメンバの得意/不得意、開発者の数、フレームワークやライブラリの利用実績や開発継続性などの要素が判断材料となる。独自のフレームワークを新しく開発することは開発・維持にかかるコストが高く、セキュリティ上のリスクに早期に対処できない場合があるため、相当な理由がない限りは避けるべきである。

(ｲ)雛形プログラムの生成

OpenAPI 仕様は API のインターフェース仕様を記述するものであるため、API が提供する機能とは分離して考えることができる。このため、API の開発は、OpenAPI 仕様に準拠したインターフェースのみが用意された雛形プログラムをベースとして進め、機能（ビジネスロジック）実装を追加する方式で開発することが望ましい。後述する「5-3 API 標準仕様案」でも推奨される REST API に準拠した設計とする場合、OpenAPI ジェネレータを利用することで、約 40 種類のプログラ

ミング言語とフレームワークを選択し、雛形プログラムを生成することができる。

(b)ビジネスロジックの実装

実装アプローチやベストプラクティスは、使用するプログラミング言語やフレームワークによって異なるが、一般的に望ましい開発方式を以下に挙げる。API のビジネスロジックを実装する際は、これらの開発方針に準拠することが望ましい。

- Git 等のソースコード管理システムを採用し、ソースコードの版管理を実施すること。
- 機能、バグフィックス、リリース等の単位でブランチを作成して管理するためのルールを定めること。また、ブランチをマージする際はレビューを実施すること。
- ビジネスロジック実装時にユニットテストを含め、作成したクラスやメソッド、プロパティ、モジュール等が意図したとおりに動作することを確認すること。テストフレームワークが利用できる場合は開発コストを抑えることができる。
- テストの実施を CI/CD のフローに組み込むこと。

なお、API 開発が進む中で OpenAPI 仕様が変更・更新された場合、雛形プログラムを再生成し、それに合わせてビジネスロジックを追加･削除することが求められる。このような状況が生じた場合は、雛形プログラムの更新部分を差分管理ツール等を利用して可視化し把握することで、API 開発における抜けや漏れを防止することができる。

(5)テスト

(a)テスト計画

API のテストは、ユニットテスト、内部結合テスト、外部結合テスト、システムテスト、受け入れテストなどの一般的なテストフェーズで実施できる。また、OpenAPI 仕様に記述されるインターフェースに関するテスト、実装されたビジネスロジックのテスト、及び性能やセキュリティなどの非機能に関するテストにそれぞれ観点を分解できる。テスト計画では、API のテストをこれらのフェーズと観点で整理し、テスト実行の自動化の可否や、テストケースを生成するツールの利用を含めて検討する。スーパーシティでは API カタログや開発者ポータルを通して API が公開され、様々なサービスから利用されることが想定されるため、以下では各テストフェーズで検討すべき観点について述べる。

(b)OpenAPI 仕様の準拠性の観点

OpenAPI 仕様を基に生成した雛形プログラムを基にビジネスロジックの開発を進めた場合でも、開発途中で意図せず雛形が書き換えられたり、ビジネスロジック内部でデータフォーマットを決定するケースがあり、OpenAPI 仕様に準拠しない API になることがある。そのため、OpenAPI 仕様のとおり API が開発されていることを確認する。

OpenAPI 仕様を基に手動でテストケースを作成することもできるが、以下のようなツールを利用することでテストケースの実行を自動化し、省力化できる。

- Dredd(https://dredd.org/en/latest/index.html)
  OpenAPI 仕様の検証をするための、言語に依存しないコマンドラインツール。OpenAPI 仕様の定義ファイルを読み込んで API のテストを行うことができる。
- Swagger Inspector(https://swagger.io/tools/swagger-inspector/)
  Web ブラウザから API のテストを行うことができるツール。公開されている OpenAPI 仕様の定

義ファイルを読み込ませることもできる。

- ReadyAPI(https://smartbear.com/product/ready-api/overview/)
  APIテストツール。機能テスト、セキュリティテスト、パフォーマンステスト、仮想API（モック生成）の機能が提供される。
- Postman(https://www.postman.com/)
  APIテストツール。API開発のためのコラボレーションプラットフォーム。APIクライアント、自動テスト、モックサーバ、APIドキュメント生成、APIモニタリングなど一連の機能が提供される。

(c)API機能の観点

APIのビジネスロジック部分に関するテストを実施する。ビジネスロジックはAPIごとに異なるため、テスト内容については個別検討となる。開発言語やフレームワークによって、テストケースの生成など省力化するツールが提供されている場合があるため、活用を検討する。

ユニットテストでは、APIの内部実装を基に条件分岐を網羅するテストケースを作成し、メソッドやクラスなど内部ロジックにおける最小単位でテストを行う（ホワイトボックステスト)。条件分岐やエラー判定など、ソフトウェアの各ユニットが意図したとおりに動作することを検証する。

内部結合テストでは、開発するAPIが別の既存APIを呼び出す場合や、ブローカー機能を利用する場合など、データ連携基盤事業者内で公開するために必要となる要素との接続性を確認する。開発者ポータルで閲覧できるAPIドキュメントや、データ連携基盤事業者のAPI開発に関するマニュアルを参照し、各要素に負荷をかけていないか等の観点も合わせてテストを進める。

外部結合テストでは、サービス開発者と連携し、サービスが正しくAPIを呼び出し、その結果の利用に問題がないかを確認する。具体的には、APIを呼び出すURL、メソッド、パラメータ、ヘッダ、データ形式などを確認し、サービスがAPIを正しく呼び出せているかを確認する。また、APIから返却されるステータスコード、ヘッダ、データ形式をサービスが正しく取り扱えているかを確認する。

受け入れテストでは、ユーザがサービスをどのように利用するかのシナリオ（ユーザストーリーと呼ばれる）を基に、サービスとAPIが接続され、ユーザに想定したサービス価値を提供できているかを確認する。

(d)非機能の観点

情報処理推進機構(IPA)の非機能要求グレードを参考にテストケースを作成する。特に、多数のサービスが利用するAPIはリクエスト負荷による性能劣化がサービス提供に大きな影響を与える場合があるため、事前に要求される性能をテストしておく必要がある。また、個人情報等やプライバシーに関する情報を取り扱うAPIでは、インターフェース部分だけでなくビジネスロジック部分でもセキュリティを考慮した実装にする必要がある。

API性能要件のテストは、Apache Benchmark、Fortio、JMeter等のツールを利用することでAPIリクエストを自動生成し、APIに負荷をかけることができる。高負荷下にあるAPIのCPUやメモリ、ネットワークの使用率を観測したり、ログの出力内容を確認したりすることで、ボトルネックを調査できる。ボトルネック箇所が特定できたら、設計やビジネスロジックの修正や、APIサーバの並列稼働数の拡張などを行うことにより改善を図る。

APIの認証・認可は、API標準仕様にあるとおりOpenID Connectで対処できる。一方、APIのセキュリティのテストは、一般的なセキュリティ診断ツールの利用や、ソースコードの静的解析ツールやLintツールを利用したソースコードのチェックが有効である。また、OWASPが発行するレポートを確

認し、開発したAPIの設計や実装を確認することで、APIのセキュリティを向上させることができる。また、専門のセキュリティテストベンダからペネトレーションテストを実施することも検討する。

5-1-3. 公開プロセス

(1)APIの公開と廃止

(a)APIのデプロイとCI/CDの活用

一般に、開発したプログラムとその実行に必要なライブラリを配置し、プログラムを実行してAPI利用者からのリクエストを待ち受け、APIによるデータ提供を実行できる状態にすることをデプロイと呼ぶ。デプロイ先の環境は、開発環境、検証環境、本番環境の3種類が用意されることが多く、開発環境や検証環境で充分なテストを実施した後、本番環境にデプロイする。デプロイ作業は、開発したAPIサーバプログラムがどのように公開されるか（物理サーバ/仮想サーバ、プロセス/コンテナ、IaaS/PaaSなど）によって異なるが、CI/CDツールを利用して一連の作業を自動化することで、作業を省力化できる。

ソースコード管理システムにソースコードが登録されたタイミングで、CI/CDツールへその旨を通知する設定を利用できる。CI/CDツールは、APIサーバプログラムの更新を検知すると開発内容が記録されているリポジトリを取得し、あらかじめ指定された手順でプログラムをテストし、その後ビルドして実行可能なバイナリを作成する。この際、必要に応じてインストール用のパッケージを作成し、デプロイが実施される。具体的なデプロイ手順は、デプロイ先のプラットフォームにより異なるため、プラットフォームに合わせて整備する必要がある。

APIの利用に認証や認可が必要な場合は、APIゲートウェイ等のAPI管理機能を操作し、アクセス制御の設定を行う。API利用者に対して認証キーを発行したり、OpenID Connect等の認証・認可サーバと連携した設定を行ったりする方法がとられる。また、アクセス元地域によってアクセス可否を切り替えたい場合などもこの段階で設定を行う。

(b)APIのバージョン管理

既に古いバージョンのAPIがデプロイされている場合、新しいバージョンに更新することで古いAPIは利用できなくなる。API機能のバグ修正や性能向上を目的する更新は、OpenAPI仕様が共通であるため、APIを利用する既存プログラムは修正不要である。しかし、APIの機能追加や削除に伴いOpenAPI仕様が変更される場合、APIを利用する既存プログラムに対する修正が必要となる場合があるため、一定の移行期間を設けることが一般的である。そのため、後述の「5-3 API標準仕様案」に記載されているように、APIのURLにはバージョンを含めた形で公開することが望ましい。

- https://api.example-city.jp/v1/...
- https://api.example-city.jp/v2/...

また、デプロイ後に問題が発生した場合に備えて、以前のバージョンにロールバックできるようにしておくことが望ましい。デプロイ後のテストに漏れがある場合など、新しいバージョンのAPIを提供開始し、しばらくしてからAPI利用者からの問合せにより不具合が明らかになる場合もある。APIのデプロイ方法によってはプラットフォーム側でロールバック機能が用意されているものもあるが、そうでない場合は手動又は半自動で対応する必要がある。

(c)APIの廃止

APIの提供を終了する場合、廃止対象となるバージョンのAPI利用者が新しいAPIに移行するよう

一定の期間を設けることが望ましい。通常、API は一度公開されたら事前の通知や移行期間なく廃止されることはなく、一定期間は非推奨(Deprecated)というステータスを経て廃止(Obsolete)される流れとなる。API 利用者が登録されておりメールアドレスが識別されている場合には、API を非推奨とした段階で移行を促す連絡を実施する。API 利用者は、廃止されるまでの間に API を利用しているプログラムを修正するなどの対応が求められる。

(2) ドキュメントの公開

(a)API ドキュメントの公開

開発した API がデプロイされ、利用可能な状態になったとしても、公開された事実と API の利用方法が開示されていなければ API は利用されない。また、API の利用方法が記載されたドキュメントに充分な情報がなかったり、説明が難解では、API の利用は促進されない。そのため、API を本番環境にデプロイしたら、API の利用方法をわかりやく記述したドキュメントを併せて公開する必要がある。また、単一地域に多数の API が公開される場合、ドキュメントのフォーマットが統一され、必要な情報が網羅的に記載されていることが求められる。OpenAPI 仕様が作成されていれば、ドキュメントを自動生成するツールを利用することもできる。

API ドキュメントには、ブラウザから API を実行できる Web UI を備えたものもある。API の実行時に指定するパラメータやデータモデルの例ともに提示されていれば、API 利用者に具体的な API 利用方法をイメージさせやすい。また、API を利用するサービス開発を想定した、API を呼び出すプログラム例（スニペットと呼ばれる）を合わせて提供することで、スムーズでバグの少ないサービス開発を支援できる。

(b)API 利用規約の公開

API を提供するにあたり、どのような条件で利用を許諾するかは、利用規約という形で事前に明示し、API 利用するサービス開発者と合意する必要がある。利用規約に記載すべき内容は、「5-2 API 利用規約テンプレート」で述べる。

5-1-4. 運用プロセス

(1)監視とモニタリング

(a)API ログの収集と分析

API の運用が始まると、想定外のエラーやシステムダウン、パフォーマンスの劣化など様々な問題が発生する可能性がある。そのため、システム全体の監視を行い、問題の発生が予想される場合には迅速に対応する必要がある。定期的な監視とモニタリングを継続し、API の安定提供に役立てることにより、API 利用者の利便性の維持・改善を図ることができる。

API を監視・分析するためには、以下の 3 つが主な要素となる。

- ログ　　：API 処理内容やその結果などが記載されたデータ
- トレースデータ　：API 内部ロジックで発生した特定のイベントに関するデータ詳細
- メトリクス　　：リソース使用率や応答時間などの測定データ

複数の API を組み合わせて利用するシステムでは、関連する API 全体の稼働状況を一元的に確認できることが重要となるため、上記のログ、トレースデータ、メトリクスを計測・収集し、併せて分析、可視化するための機能を配置することが望ましい。

次のようなツールやサービスを活用することで、API の監視とモニタリングが容易になる。例示しているツールは、API を整備するプラットフォーム全体の運用を考慮した上で検討する必要がある。

- ログ分析（Elastic Stack、Splunk など）
  サーバや NW 機器、アプリケーションが出力する様々なログデータを収集、蓄積し、検索及び分析を行うためのツール。
- 監視（Prometheus、Zabbix など）
  サーバや NW 機器、アプリケーションの稼働状況の確認、収集を行い、条件に応じてアラート通知や後続処理を実行するためのツール。
- パフォーマンス管理（New Relic、Datadog、AppDynamics など）
  システムやアプリケーションの応答時間や負荷などのパフォーマンス状況を計測し、性能を可視化し管理するためのツール。
- データ可視化（Grafana、Kibana など）
  蓄積されたデータの集計、整形、加工、分析を行い、グラフ描画やダッシュボード生成などデータの可視化を行うためのツール。
- トレーシング（Zipkin、Jaeger など）
  1 つの処理要求に対して実行される複数の処理の流れや依存関係を追跡し、関連しあった処理の状況を一連のフローとして確認するためのツール。

(b)稼働状況と性能のモニタリング

API の運用にあたって、提供する API が利用可能な状態にあるかを監視する必要がある。API の監視は、単純な API の応答有無（死活監視）だけでなく、API アクセス数、成功/エラー比率、平均応答時間、さらには地理的関連データ（例えば、API 呼び出し元地域）など、様々な情報を可視化、モニタリングできることが望ましい。それにより、個々の API がいつ、誰に、どこから利用され、どのような結果だったかを包括的に把握することが可能になり、API の利用状況や処理結果などの情報を基に、各 API の機能や性能の改善検討を進めるための材料となり得る。

API を直接公開するのではなく、複数の API をまとめて管理する API ゲートウェイを介して公開する場合、API に関するメトリクス（API アクセス数、成功/エラー比率、平均応答時間 等）を取得する機能が提供されている場合がある。先に述べたログ分析や可視化ツールと組み合わせることで、API のモニタリングや一元管理のための仕組みを構築する労力を省力化できる。

性能のモニタリングには APM(Application Performance Management)を活用する方法が考えられる。APM は、アプリケーションやシステムのパフォーマンスを横断的して監視し、可視化、モニタリングするツールである。APM を活用することは、パフォーマンスの問題を検出し、どこがボトルネックになっているかを調査するためにも有用である。

(c)セキュリティ監視

スーパーシティにおける API の公開については、総務省『スマートシティセキュリティガイドライン』[21]を参照し、適切なセキュリティ監視を行う必要がある。特に API 観点では、WAF(Web Application Firewall)と呼ばれるファイアフォール機能を用いたセキュリティ対策が重要になる。

セキュリティ監視の手段としては、SIEM(Security Information and Event Management)を活用することが考えられる。SIEM はセキュリティに関する情報やイベントを管理し、自動的に分析するソリューションである。SIEM では、NW 機器やセキュリティソフトなどからログを収集し、収集した大量のログをリアルタイムに相関分析することで、システムを危険にさらす可能性のある潜在的なセキュ

[21] https://www.soumu.go.jp/main_content/000710778.pdf

リティ上の事象を早期に発見することが可能である。

(2)障害対応

(a)障害検知

障害をいち早く解消し復旧させるためには、迅速な障害検知が重要になる。障害検知は、API の停止又は著しい性能劣化など、正常な API の提供ができない状態になったことを API 提供者に知らせ、迅速に対処することが目的である。API 利用者側からの問合せなどにより、障害を検知することも想定されるが、API 提供者側で検知できる内容はシステム的に自動で検知することが望ましい。

障害として判断すべき状態は、次の例が考えられる。

- API を利用できない（アクセスできない、応答がない など）
- API の動作が不安定（応答時間が長い、成功する場合と失敗する場合がある など）
- API が仕様どおりに動かない（応答内容が仕様書と異なる、特定条件下で失敗する など）

API 提供者側にて異常な状態を迅速に検知するには、「5-1-4(1)監視とモニタリング」に記述した監視結果に応じて、運用担当者に通知する仕組みが必要になり、障害原因の大まかな切り分けが行える内容を通知できることが望ましい。そのため、監視結果がどのような場合に、誰に、どのような通知を行うかなど、通知条件や通知先、通知内容を事前に定義しておく必要がある。

障害対応は、一般的に、発生する障害のレベルにより必要な対応が異なる。そのため、API 利用者への影響度や重要度に応じて障害レベルを定義し、障害検知の通知に障害レベルを含めることで、後の対応を円滑化できる。

障害を検知したら、以下の項目について検討し、原因調査を進める。

- 障害対応の担当者と役割
- 連絡先と連絡手段
- 想定される障害とその対応手順
- 復旧の完了条件
- 目標復旧時間
- API 利用者への通知要否、タイミング

(b)原因調査

原因調査は、検知した障害内容について原因を調査し、発生した障害の復旧手段の候補を特定できる状態にすることが目的である。障害を引き起こす要因は様々なことが考えられる。あらかじめ障害対応手順を定めておくことで、スムーズな原因調査と復旧作業につなげることができる。

障害原因には、次の例が考えられる。

- API アクセスの急増
- API 利用者による API 仕様を逸脱する利用（意図しない巨大データの送信など）
- API が稼働する基盤（インフラ）や連携システム/サービスの問題
- 提供する API 自体の不具合
- システムメンテナンス作業などでの人的ミス

原因調査にあたっては、障害が発生した時間帯の状況を確認することが重要になる。障害検知により通知・連絡された内容から原因調査を実施することになる。障害が発生した時間帯のメンテナンス作業の有無を確認し、システム的な問題が原因なのか、人的ミスなどが原因なのかを切り分ける必要がある。

システム的な問題の場合、API の各種ログ、API が稼働する基盤（インフラ）のログが原因調査の起点となる。API 利用者側からの問合せによる障害の原因調査については、事象の再現確認も実施することも重要になる。障害対応手順化できていない障害が発生した際には、エンジニアにエスカレーションを行い、障害原因の切り分けた原因調査、特定を実施する。

(c)復旧作業

復旧作業とは、「5-1-4(2)(b)原因調査」により特定した障害原因を取り除き、再度 API を利用可能な状態にする作業を指す。障害対応手順をあらかじめ定めておくことが望ましい。

障害対応手順に沿った対応にて障害が解消しない場合には、エンジニアにエスカレーションを行い、障害原因の切り分けや原因調査をして、特定後、復旧作業を実施する。復旧対応完了後には、障害が起きた際の事象や対応内容の整理、分析を行い、障害対応手順の更新やナレッジ共有などの取り組みを行うことで、類似障害発生時の効率化や再発防止に繋げることができる。

障害発生から復旧までのダウンタイム（API が利用できない状態）を最小化するには、一連の処理を可能な範囲で自動化することが望ましい。例えば、アクセス数の急増などにより CPU やメモリ、ディスクなどのシステムリソースが不足している場合には自動で拡張する、特定のサービスやプログラムがダウンしている場合に自動で再起動を行う、などの仕組みを取り入れることが考えられる。

(3)サポート対応

(a)サポート窓口による対応

API の利用が進むと、API 利用者からの質問や要望、不具合の連絡など様々なフィードバックが発生する。そのため、API の運営主体は自地域で運用する開発者ポータル等において、メールや問合せフォーム、コメント可能な Issue 管理（課題管理）機能の提供など、API 利用者からの直接的なフィードバックを受ける窓口を設ける必要がある。

フィードバックへの迅速な対応は、API 利用者の満足度の維持・向上につながる。得られたフィードバックは組織内部で記録し、重要度に応じて優先度をつけるなどの管理を行った上で、実装などの対応が必要かどうかを検討し対応を行う。フィードバック内容は公開することが望ましいが、セキュリティや個人情報の開示に関するフィードバックなど、即時の公開が困難な内容に関しては、適宜公開を行うかどうかを検討する。また、類似の内容が複数回質問される際には、その内容を FAQ として公開するなどの運用を検討する。

(b)開発者コミュニティによる対応

API は先端的サービスの開発者に利用され、先端的サービスとして住民に届くことによって価値を生み出す。API の運用を円滑にし、API の直接的な利用者である開発者との良好な関係を保ち続けるためには、デベロッパーリレーションが重要となる。デベロッパーリレーションとは、開発者との相互コミュニケーションを通じて、継続的かつ良好な関係性を築くためのマーケティング手法を指す。デベロッパーリレーションの取り組みの1つとして開発者コミュニティの拡大､活性化が挙げられる。

開発者コミュニティとは、API を利用している特定の開発会社や開発者個人により形成されるコミュニティを指すが、デベロッパーリレーションでは、今後 API を利用する可能性がある開発者も視野に含めた活動を行い、開発者コミュニティ自体を拡大、活性化することが重要となる。開発者との良好な関係を構築するにあたっては、オンライン上のフィードバックに加え、ミートアップやカンファレンスといったイベント開催なども活用し、開発者コミュニティに対して API 活用のアイデアや事例などの情報提供を行い、開発者の開発作業を容易にすることが必要となる。デベロッパーリレーションは API の運用を行う上で、フィードバックの獲得だけではなく、API の仕様変更や廃止、今後の予

定に関する告知を行う際にも有用となる。

(c)フィードバックの収集・対応

組織の内外から多くのフィードバック得られることは、提供する API の潜在的な問題や改善点、新たな API のアイデアの発見といった効果も期待できる。受動的なフィードバックの受付けだけではなく、API 利用者や API を利用したサービスの利用者に対して能動的にヒアリングやアンケートを行うことで、取り組みへのエンゲージメントを高め、API がより利用されやすい地盤を形成できる。

また、フィードバックへの対応を継続して実施することにより、フィードバックの報告者には、「フィードバックが受け入れられる」、「対話が可能である」という実感が生まれ、より良いフィードバックを呼び起こすことが期待できる。フィードバックへの対応による対話のサイクルを回し、API 利用者とのコミュニティを作り出すことは、地域に根づいた継続的な開発を行うにあたって大きな無形資産となる。

API が提供された後の API のパフォーマンスについても、API の利用者からフィードバックを収集することは、API を継続的に改善していくために重要である。したがって、API 利用者からのフィードバックを収集するプロセスを容易にするために、地域に整備される開発者ポータルにフィードバック共有用の機能を追加する等の方法で、API の利用者が個々の API を直接評価・コメントできることが望ましい。

5-2. API 利用規約テンプレート

ここでは、データ連携基盤整備事業者と、データ連携基盤整備事業者が提供する API を利用して先端的サービスを開発・提供する事業者及び別のデータ連携基盤整備事業者との間で取り交わされる API 利用規約のテンプレートを提供し、各条文サンプルを解説する。また、データ連携基盤整備事業が規定される国家戦略特別区域法に関連し、国家戦略特別区域会議では「先端的区域データ活用事業活動を実施する主体が国家戦略特別区域において新たな規制の特例措置の適用を受けて先端的区域データ活用事業を実施し又はその実施を促進する必要があると認める時は、内閣総理大臣に対し、当該新たな規制の特例措置の整備を求めることができる」としているため、データ連携基盤が担う先端的区域データ活用事業活動のうち、API を扱うオンラインサービスに関連する取り組みと関連法令、想定される特例措置例についても記載する。そのためここで想定する読者は API の提供により先端的区域データ活用事業活動を支援するデータ連携基盤整備事業者である。

なお、API 利用規約のテンプレートについては、表 2-3-3-1 に記載の国・自治体及び民間における API 利用規約の条文を参考に整理した。

5-2-1. はじめに

本 API 利用規約テンプレートが対象とする API と、API を提供するデータ連携基盤整備事業者、API を利用する先端的サービス事業者やデータ連携基盤整備事業者との関係を下図に示す。

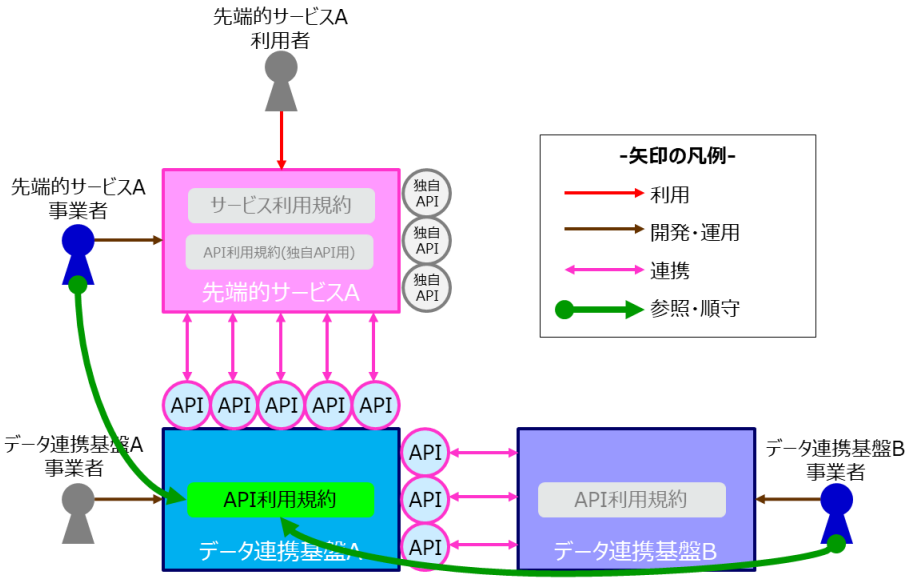


図 5-2-1-1 API 利用規約テンプレートの位置付け

上図のとおり、本 API 利用規約テンプレートは、データ連携基盤整備事業者から、データ連携基盤が提供する API を利用して先端的サービスを開発する先端的サービス事業者や、API を利用して新たな API を開発する別のデータ連携基盤整備事業者に提供される API 利用規約を対象としたものであり、先端的サービス事業者が開発する先端的サービスの利用者に向けた利用規約や、先端的サービス事業者が独自に提供する API の利用規約については本書の対象外とする。また、本 API 利用規約テンプレートは、特定の単一 API に適用されるものでなく、複数の API に対して一括して適用されることを想定している。

「5-2-2 利用規約の各項目と解説」では API 利用規約の各項目の条文例を四角枠の中に記載し、続け

て解説を行う。特に第 1 条の四角枠内の条文例では、API を提供するデータ連携基盤整備事業者を「*API 提供者名*」、データ連携基盤整備事業者が提供する API のことを「*API サービス名*」と略して記載しているため、本 API 利用規約テンプレートを採用するデータ連携基盤整備事業者の正式名称及び正式な API サービス名に置き換えて利用頂きたい。

一方、関連法令に関しては、前述の先端的サービスのユースケースのうち API を扱うオンラインサービスである下表に挙げた 3 つの取り組みについて、その関連法とサービスを実施する上での特例措置例を「5-2-3 ユースケース事例と関連法令」に記載する。

表 5-2-1-1　先端的サービスに関連するオンラインサービス例

	オンラインサービス	取り組み
1	地域共通ボランティア ポイント（金融）	地方自治体等の組織がボランティア活動によって貯まるポイントを地域電子通貨として発行する。他の行政サービスでの支払いやボランティア以外の地域貢献活動などと広く連携することで利用シーンが広がることが期待される。
2	オンライン診療、 オンライン服薬指導、 混合診療（医療）	運動や食事データ等のライフログや医療データを連携することで、健康～未病～治療のサイクルをシームレスに繋ぐヘルスケアプラットフォームを構築する。また、あらゆる運動データをウェアラブルデバイスから収集し、健康状態と突合し、パーソナライズ化された運動メニューの推奨や発病リスクのアラート、健康 e ラーニングの配信など、健康～未病に関わる取り組みを健康アプリを通じて促進する。発病した際は、自覚症状が出る前に AI により受診勧奨の通知が行われると同時に、ライフログや健康診断データ等が電子カルテに統合され、最適なオンライン診療とオンライン服薬指導を可能にする。治療後は、再発予防のためのパーソナライズ化された最適な取り組み（運動・食事等）をレコメンドし、健康維持を促進する。
3	オンデマンド型 e ラーニング授業 （教育）	平成 29･30 年改訂学習指導要領にもあるように「主体的・対話的で深い学び」の視点に立った授業改善を行うことで、学校教育における質の高い学びを実現し、学習内容を深く理解し、資質・能力を身に付け、生涯にわたって能動的に学び続けるようにするアクティブラーニングと呼ばれる学習法が重要視されている[22]。それを踏まえて、学校の授業に多様なデジタルコンテンツを駆使したアクティブラーニングを採用し、様々な分野の専門家に免許状を有する教員と同等に授業を担当してもらえるようにする。また、オンデマンド型の e ラーニング形式の授業を組み合わせた完全オンライン遠隔教育を実施する高等学校を実現し、遠隔授業で全単位取得可能にする。

なお、行政手続きも API を扱うオンラインサービスの 1 つになり得るが、本書では詳細は記述しない。2021 年現在、デジタル庁設立に向けた活発な取り組みがなされ、デジタル手続法により行政のあらゆるサービスをデジタルで完結させることを目指している。

[22] 「平成 29･30 年改訂 学習指導要領、解説等」の「主体的・対話的で深い学びの実現（「アクティブ・ラーニング」の視点からの授業改善）について（イメージ）」https://www.mext.go.jp/content/1421692_8.pdf

5-2-2. 利用規約の各項目と解説

本項では、本 API 利用規約テンプレートとしての利用規約の各項目の条文例とその解説を記載する。ただし、ここで挙げた条文例や解説は、利用規約の効率的な作成に資するための参考資料として提示しているものであり、その使用等について一切の責任を負わないこと、実際の案件にあたっては、必要に応じて弁護士の確認を経るなどして、各データ連携基盤事業者の責任で適切な利用規約を定めるべきことにつき、留意いただきたい。

第 1 条（目的）

本 API 利用規約（以下「本利用規約」という。）は、*API 提供者名*（以下「API 提供者」という。）が提供する下記のアプリケーション・プログラミング・インタフェース（以下「本 API」という。）の利用に際し、API 利用者が遵守すべき利用条件を定めることを目的とする。本利用規約に同意いただけない場合、本 API を利用することはできない。

記

本 API の名称：*API サービス名*

解説：

本条は、本利用規約が、API を利用する際に遵守すべき利用条件を定めることを目的として策定されたものであることを明らかにし、API 提供者（データ連携基盤整備事業者）と API 利用者（先端的サービス事業者等）との間に適用される規約であることを示すものである。

第 2 条（定義）

本利用規約において使用する用語は次の各号の意味で使用する。

(1) 「API 提供者」とは、*API 提供者名* をいう。
(2) 「API 利用者」とは、本利用規約に同意の上、本 API を利用する者（国、地方公共団体を含む。）をいう。
(3) 「API 認証情報」とは、API 利用者を特定するために API 提供者が API 利用者に対し提供する認証情報をいう。
(4) 「API 利用アプリケーション」とは、API 利用者が本 API を組み込んで開発するアプリケーションやプログラム等をいう。
(5) 「API 提供情報」とは、本 API を経由して API 利用アプリケーションに提供される情報をいう。
(6) 「エンドユーザー」とは、API 利用アプリケーションを利用するユーザーをいう。
(7) 「本契約」とは、本利用規約を内容として API 提供者と API 利用者との間で成立する、本 API の利用に係る契約をいう。

解説：

本条は、本利用規約において繰り返し登場する用語をまとめて定義するものである。わかりやすさの観点から、データ連携基盤整備事業者や先端的サービス事業者等の言葉を用いることなく、「API 提供者」や「API 利用者」等の一般的な用語によることとしている。

(1)項は、「API 提供者」を定義するものであり、本 API を提供するデータ連携基盤整備事業者の名称を記載する。地方自治体がデータ連携基盤整備事業者となる場合には、当該地方自治体を記載する。(2)項は、「API 利用者」を定義するものである。国や地方公共団体も本利用規約に基づいて本 API を利用する

ことがあり得ることを想定し、かっこ書きを付すこととしている。(7)項は、「本契約」を定義するものである。本 API の利用にあたっては、登録申請画面において本利用規約が適用される旨をあらかじめ表示しておくこと等により、本利用規約をその内容に組み込んだ契約が API 提供者と各 API 利用者との間で個別に成立するようにする必要があり、そのようにして成立した契約を本契約と呼ぶこととしている。

第 3 条（登録）

(1) 本 API の利用を希望する者は、本利用規約に同意し真正かつ正確な所定の情報（以下「登録事項」という。）を提供することにより、API 提供者に対し、本 API の利用登録を申請することができる。

(2) API 提供者は、前項の申請に対する登録の可否を判断し登録を認める場合には、その旨を申請者に通知するものとする。

(3) 前項の通知をもって申請者と API 提供者との間で本契約が成立し、申請者は API 利用者として本 API を利用することができるようになる。

(4) API 利用者の名称、所在地その他の登録事項に変更が生じた場合、当該 API 利用者は直ちにその旨を API 提供者に届け出るものとする。

解説：

本条は、本 API の利用にあたって利用登録を必要とする場合に、その登録の手続について定めるものである（本 API の利用にあたり登録が不要な場合には本条を設けることを要しない）。

(1)項は、本 API の利用にあたっては、API 提供者に対する利用登録の申請が必要であること、申請にあたっては本利用規約に同意するとともに登録事項を提供しなければならないこと、及び、提供する登録事項は真正かつ正確な情報でなければならないことを、明らかにしたものである。登録事項として提供してもらう情報の内容については、「所定の情報」としており、本利用規約とは別に定める建付けとしている。登録事項に変更が生じた場合には、(4)項により、API 提供者に対する変更の届出を義務付けている。

(2)項は、登録を認める場合には API 提供者が申請者に通知すべき旨を定めるものである。データ連携基盤整備事業における API の場合であっても、契約自由の原則により、いかなる相手と API 利用に係る契約を締結するかは基本的には API 提供者の自由である。したがって、例えば、申請者におけるセキュリティが十分に担保されていない、反社会的勢力に該当すると認められる等の理由によって登録を認めないことは可能であるが、他方で、国家戦略特別区域法の趣旨や事業の内容・規模によっては公的インフラとしての性質を帯びることから、合理的な理由なく不当に差別的な取り扱いをすることのないよう留意して運用すべきである。なお、例えば、事前に接続試験を実施し合格することを登録の要件とする場合には、その旨を定めることも考えられる。

第 4 条（API 認証情報の発行及び管理）

(1) API 提供者は、登録の完了した API 利用者に対し、API 認証情報を発行する。

(2) API 利用者は、API 提供者より発行された API 認証情報を自己の費用と責任において厳重に管理するものとし、これを第三者に利用させたり、貸与、譲渡、売買その他一切の処分をしてはならないものとする。

(3) API 提供者は、(1)により発行された API 認証情報を利用した本 API の利用を API 利用者本人による本 API の利用とみなし、API 利用者は API 認証情報を利用して本 API を利用した者の行為に基づくすべての責任を負うものとする。API 利用者による API 認証情報の管理不十分、使用上の過誤、第三者による使用等による損害の責任は API 利用者が負うものとし、API 提供者は一切責任を負わないものとする。

(4) API 利用者は、API 認証情報が第三者により不正に利用されたこと又はその可能性を認知した場

合、直ちにAPI提供者に通知するとともに、API認証情報を再発行するための手続を執る等API提供者の求める措置を講ずるものとする。

解説：

本条は、登録が完了したAPI利用者に対して個別のAPI認証情報（第2条(3)）を発行する場合に、その手続及びAPI利用者における管理の在り方について定めるものである（本APIの利用にあたり登録が不要な場合には本条を設けることを要しない）。

API認証情報が利用されて本APIへの接続が行われる場合、API提供者の側において、その背景や経緯等の事情を知ることはできず、API利用者本人が正当な権限の下に利用しているか否かを都度判断することは難しい。そこで、(2)項は、API認証情報の重要性に鑑み、第三者がこれを利用することのないよう、API利用者に厳重な管理を義務付けている。その上で、(3)項は、API認証情報を利用した本APIの利用については、API利用者本人による本APIの利用とみなし、API利用者が全ての責任を負う旨、API認証情報の管理不十分、使用上の過誤、第三者による使用等による損害の責任はAPI利用者が負う旨を定めている。

(4)項は、API認証情報の不正利用等があった場合に、API提供者がセキュリティの改善・向上等のための対策を講じることができるようにするための規定である。

第5条（本APIの提供条件）

(1) API提供者は、本利用規約及び本利用規約に付属する規約の定めに従うことを条件として、API利用者に本APIを非独占的に使用させることを許諾する。

(2) 本APIの仕様は別途定めるAPI仕様書のとおりとする。API提供者はAPI利用者に事前の通知をすることなく本APIの全部又は一部の仕様をいつでも変更することができるものとする。

(3) API提供者は、本APIの負荷状況に応じて本APIの利用の制限をかけることができるものとする。

(4) API利用者は、自らの費用と責任において、ハードウェア、ソフトウェアその他の機器、インターネット接続回線等、本APIの利用に必要な環境（以下、「利用環境」という。）の一切を整備するものとする。

(5) API利用者は、自己の利用環境に応じて、コンピューターウィルスの感染の防止、不正アクセス及び情報漏洩の防止等のセキュリティ対策を自らの費用と責任において講ずるものとする。

(6) API提供者は、本APIに係るコンピューターシステムの定期的な点検又は保守作業を行う場合、その他運用や技術上、本APIの停止が必要であると判断した場合には、API利用者に事前に通知の上、本APIの利用の全部又は一部を停止することができるものとする。

(7) 前項の規定にかかわらず、API提供者は、以下のいずれかに該当する場合には、API利用者に事前に通知することなく、本APIの利用の全部又は一部を停止することができるものとする。

（ア） 本APIに係るコンピューターシステムの緊急の点検又は保守作業を行う場合

（イ） 本APIに係るコンピューターシステム、通信回線等が事故により停止した場合

（ウ） 火災、停電、事故、天災地変などの不可抗力により本APIの提供ができなくなった場合

（エ） その他、運用や技術上、API提供者が緊急の停止が必要であると判断した場合

解説：

本条は、本APIをどのような条件・仕様のもとで提供するかについて定めるものである。

(1)項は、複数のAPI利用者に本APIを提供することが想定されることを前提に、API利用者に対して非独占的に本APIを使用することを認めることとしている。

(2)項は、本APIがオープンAPIであることを前提に、その仕様は開発者ポータル等において別途定め

るAPI仕様書のとおりである旨を明らかにするとともに、必要なアップデートを行うことができるよう、API提供者がいつでも仕様を変更することができる旨を定めている。この点、API利用者に重大かつ広範な影響を及ぼす可能性が高い変更を行う場合には、API 利用者に対して事前に通知することが望ましいが、他方で、API利用者に及ぼす影響の程度は変更内容によって様々であり、また、セキュリティの改善・向上等のために直ちに変更を行わなければならない事態もあり得ることから、本利用規約では、事前の通知をすることなく変更できるとの文例を採用している。

(3)項は、本APIへの接続状況等に照らして過大な負荷が生じている場合には、API提供者の判断により、本APIの利用を制限することができる旨を定めている。

(4)項は、本APIを利用するために必要となるAPI利用者側の設備等については、API利用者の費用と責任で行うべき旨を確認的に定めている。

(5)項は、API 利用者の利用環境におけるセキュリティ対策についても、API 利用者の費用と責任で行うべき旨を定めている。なお、API利用者におけるセキュリティ対策の実施を担保するため、API利用者のセキュリティに係る管理体制や機能等を報告するためのチェックリストを策定し、その提出を義務付けることも考えられる。

(6)及び(7)項は、本 API の利用の全部又は一部の停止措置をとる場合について定めている。定期的な点検又は保守作業に伴って本APIの利用を停止する場合には、事前にAPI利用者に対する通知を要する。これに対して、緊急の点検又は保守作業を行う必要があるなど、直ちに停止措置をとる必要性が高い場合には、API利用者に対する事前の通知は不要としている。

第6条（料金）

(1) API利用者は、API提供者が別途定めるところに従い、本APIの利用に必要な料金をAPI提供者が指定する期日までにAPI提供者が指定する方法で支払うものとする。支払いに要する費用はAPI利用者の負担とする。支払いの事実が確認できない場合、本APIの利用の全部又は一部を停止することがある。

(2) API利用者は、前項の料金を支払期日までに支払わない場合、支払期日の翌日から支払済みに至るまで、年●%の割合で遅延損害金を支払うものとする。

解説：

本条は、API利用者による本APIの利用を有償とする場合に、その料金について定めるものである。料金の詳細については、本利用規約とは別に定めることとしている。また、振込手数料等の支払いに要する費用については、API利用者の負担であることを確認している。

第7条（API利用者の義務・責任）

(1) API利用者は、自己の費用及び責任においてAPI利用アプリケーション及びこれに関連するサービスをエンドユーザーに提供するものとする。

(2) API利用者は、API利用者が提供するAPI利用アプリケーション及びこれに関連するサービスはAPI利用者が開発し運営するものでありAPI利用者がその責任を負う旨を、エンドユーザーに明示するものとする。また、エンドユーザーとの間で予想されるトラブル等についてエンドユーザーが著しく不利とならないよう、API利用者及びエンドユーザーの責任範囲を適切に定め、エンドユーザーが理解できるように明示するものとする。

(3) API利用者は、API利用アプリケーション及びこれに関連するサービスの提供にあたり、個人情報の保護に関する法律、特定商取引に関する法律、不当景品類及び不当表示防止法、消費者契約法その他の法令を遵守するものとする。

(4) API利用者は、本APIの利用又はAPI利用アプリケーション及びこれに関連するサービスの提供

に関し、不正アクセス等が行われた場合、不正アクセス等による情報の流出・漏洩・改竄等もしくは財産の処分が判明した場合、又はそれらの具体的な可能性を認識した場合、直ちに API 提供者に報告するものとする。

(5) API 利用者は、前項の場合、速やかに実施可能な対策を講じた上で、API 提供者と協力して原因の究明及び対策を行うものとする。API 提供者は、十分な対策が講じられるまでの間、当該 API 利用者の本 API の利用を制限又は停止することができるものとする。API 利用者は、不正アクセス等の発生時に原因の調査等を行うことができるよう必要なアクセスログの記録及び保存を行うものとする。

解説：

本条は、API 利用者が、API 利用アプリケーション（第 2 条(4)）をエンドユーザ（第 2 条(6)）に提供してサービスを行うにあたって、API 提供者との関係で負う義務と責任について定めるものである。本 API を利用して実施される個々の先端的サービスは API 利用者が提供するものであり、API 利用者の責任においてエンドユーザ向けの利用規約の策定等が行われるべきものであるが、そのサービスがエンドユーザに著しく不利な内容でないか、法令を遵守して実施されているか、不正アクセス等によって API 利用者やエンドユーザに損害が生じないか等については、本 API を提供する API 提供者にとっても重要な関心事であることから、本利用規約中でも API 提供者の義務及び責任という形で一定の規律を及ぼすこととしたものである。

(1)項は、API 利用アプリケーション及びこれに関連するサービスのエンドユーザへの提供は、API 利用者の費用と責任において行われるべきことを確認的に定めている。

(2)項は、API 利用者に対し、API 利用アプリケーション及びこれに関連するサービスについては API 利用者が責任を負うものである旨をエンドユーザ向けに明示すること、及び、エンドユーザとの間で予想されるトラブル等について API 利用者及びエンドユーザの責任範囲を適切に定めてわかりやすく明示することを義務付けている。

(3)項は、API 利用者に対し、API 利用アプリケーション及びこれに関連するサービスを開発し運営するにあたって諸法令を遵守すべきことを義務付けている。

(4)及び(5)項は、API 利用者による本 API の利用又は API 利用アプリケーション及びこれに関連するサービスの提供に関し、不正アクセス等やそれに伴う情報漏洩・財産の処分等が生じた場合に、API 利用者において、その原因を究明し、セキュリティ改善・向上等のための対策を講じるべき旨を定めるものである。すなわち、API 利用者は、不正アクセス等やそれに伴う情報漏洩・財産の処分等が判明した場合のみならず、それらの具体的な可能性を認識した段階で、直ちに API 提供者に報告しなければならないものとしている。その上で、API 利用者は、API 提供者と協力して原因の究明及び対策を行わなければならないものとしている。なお、そのような場合には、API 提供者において、当該 API 利用者による本 API の利用を制限又は停止することができることについても明らかにしている。

第 8 条（禁止事項）

API 利用者は、本 API の利用又は API 利用アプリケーション及びこれに関連するサービスに関し、自ら又は第三者を利用して、次の各号に該当する行為を行ってはならないものとする。

(1) 国民の安全に脅威を与える行為

(2) API 提供者もしくは第三者に不利益もしくは損害を与える行為、又はそのおそれのある行為

(3) 本 API の運用や第三者による利用を妨害する行為、又はそのおそれのある行為

(4) 短時間における大量のアクセスその他本 API の運用に支障を与える行為、又はそのおそれのある行為

(5) 以下の内容を含むサイトやアプリケーション上又はアプリケーション内で、本 API 又は本 API 提

供情報を利用する行為
(ア)公序良俗・法令等などに違反し、又はそのおそれのある表現
(イ)API 提供者又は第三者の財産権、知的財産権、信用、名誉、プライバシー権、肖像権その他の権利を侵害しているもの
(ウ)コンピューターウィルス等のソフトウェア、ハードウェア又は通信機器等の適切な動作を妨害、破壊もしくは制限し、又はそのおそれのあるもの
(エ)その他 API 提供者が不適当と判断するもの

解説：

本条は、API 利用者が行ってはならない行為について定めるものである。

(1)項は、本 API が公的インフラとしての性質を有することに鑑み、国民の安全に脅威を与える行為を禁止している。

(3)項は、本 API の運用を積極的に妨害する行為を、(4)項は、結果として本 API の運用に支障を与える行為を、それぞれ禁止している。

(5)項は、本 API が個々の先端的サービスにおいてどのように利用されるのかについては、API 提供者にとっても重要な関心事であることから、本 API のレピュテーションを貶めるおそれのある内容を含むサイト・アプリケーションや、第三者の権利を侵害するおそれのある内容を含むサイト・アプリケーション等において、本 API 又は本 API 提供情報（第 2 条(5)項）を利用する行為等を禁止している。

第 9 条（利用解除）

API 提供者は、API 利用者が本利用規約の各条項に違反した場合、その原因に係る当該 API 利用者の帰責性の有無にかかわらず、事前の通告なしに、本 API の利用を全部もしくは一部を停止し又は本契約を解除することができるものとする。

解説：

本条は、API 提供者が本利用規約に違反した場合における、利用の停止又は解除について定めるものである。ここにいう「停止」とは、一時的に本 API の全部又は一部を利用できないようにすることを意味している。「解除」とは、本契約上の地位を失わせ、永続的に本 API を利用できないようにすることを意味している。API 利用者が本利用規約に違反した場合には、債務不履行となり、民法の原則によっても履行を催告した上で解除をすることができるが（民法第 541 条）、本条では、本 API が多数の API 利用者やエンドユーザに影響を及ぼすものであり、公的インフラとしての性質を有するものであることに鑑み、無催告で即時に利用をやめさせることができるようにする一方、解除だけではなく停止措置をとることもできるようにすることにより、違反の内容に応じて違反の解消後に本 API を再び利用できる手段も認めることとしている。なお、必要に応じて、停止措置を取り消して本 API を再び利用できるようにするための条件や、その際の手続等について定めることも考えられる。

第 10 条（権利の帰属・利用）

(1) 本 API に係る知的財産権（著作権、特許権、商標権その他の財産権を含む。以下同様。）は、API 提供者又は当該権利を有する第三者に帰属するものとし、API 利用者は、本規約等に明示的に規定される場合を除き、いかなる権利も取得しないものとする。
(2) API 利用者は、API 利用アプリケーション及びこれに関連するサービスを提供するにあたり、API 提供者所定の方法によってのみ、API 提供者の商標及び商号等を使用することができるものとする。

解説：

本条は、本 API に関する知的財産権の帰属等を定めるものである。

(1)項は、本 API に係る知的財産権は、本 API を開発した API 提供者又はライセンスを受けている権利者に帰属するものであり、API 利用者はそうした知的財産権を取得するものではないことを明らかにしている。

(2)項は、API 利用者が、API 提供者所定の方法によって、API 提供者の商標や商号を使用することを認めるものである。例えば、API 利用アプリケーションの説明の中で、本 API のロゴを表示するといったことが考えられる。

第 11 条（免責）

(1) API 提供者は、本 API について次の各号につき、いかなる保証も行うものではない。さらに、API 利用者が API 提供者から直接又は間接に本 API に関する情報を得た場合であっても、API 提供者は API 利用者に対し本利用規約において規定されている内容を超えていかなる保証も行うものではない。

(ア) 本 API の利用に起因して利用環境に不具合や障害が生じないこと

(イ) 本 API 又は API 提供情報の正確性、信頼性、完全性、適法性、非侵害性、有効性、目的適合性

(ウ) 本 API を提供するためのシステムにエラー、バグ、不具合、中断その他の瑕疵又はセキュリティ上の欠陥が存在しないこと

(エ) API 利用者に適用のある法令、業界団体の内部規則等への適合性

(2) API 提供者は、天災、労働紛争、停電、通信インフラの故障、公共サービスの停止、自然現象、暴動、政府の行為、テロ、戦争その他の不可抗力により API 利用者に生じた損害について責任を負わない。

解説：

本条は、API 提供者の免責について定めるものである。

(1)項は、API 提供者は、本 API に関し、その利用に起因して API 利用者の利用環境に不具合や障害が生じないことや、本 API 又は API 提供情報（第 2 条(5)）の正確性、信頼性、完全性等について、保証するものではないことを明らかにしている。

(2)項は、API 提供者は、不可抗力による損害については責任を負わないことを明らかにしている。

第 12 条（補償/賠償）

(1) API 利用者は、本利用規約等の違反により API 提供者又は第三者（エンドユーザーを含む）に損害が発生した場合、当該損害の全額（弁護士費用を含みます）を賠償するものとする。

(2) API 利用者は、API 利用アプリケーション及びこれに関連するサービスの提供に関連して、第三者（エンドユーザーを含む）との間で生じたクレーム、請求その他の紛争等については、自らの費用と責任において解決するものとする。

(3) API 利用者は、API 利用アプリケーション及びこれに関連するサービスの提供に関してエンドユーザーに損害が生じた時は、速やかにその原因を究明し、API 利用アプリケーションの利用規約等に基づき賠償又は補償が不要となる場合を除き、当該サービスの利用者に生じた損害を賠償又は補償するものとする。

(4) API 提供者は、API 提供者の故意又は重大な過失による場合を除き、いかなる場合にも、本 API の利用に関して API 利用者に生じた損害について損害賠償責任を負わないものとする。

(5) API 提供者が損害賠償責任を負う場合であっても、賠償すべき損害の範囲は、API 利用者に現実に発生した直接かつ通常生じる範囲内の損害に限るものとし、逸失利益及び特別の事情から生じた損害については責任を負わないものとする。また、その賠償額は、本契約に基づき API 利用者

が API 提供者に対して支払った別途定める月額利用料金の（*API 提供者側で規定した月数*）分を上限とする。

解説：

本条は、本 API の利用に伴う賠償責任の範囲について定めるものである。(1)項～(3)項は API 利用者の責任について、(4)項及び(5)項は API 提供者の責任について、それぞれ定めている。

(1)項は、本利用規約の違反があった場合の API 利用者の賠償責任を定めるものである。

(2)項は、API 利用アプリケーション及びこれに関連するサービスの提供（開発・運営）に関連する第三者との紛争等については、API 利用者の費用と責任において解決すべきことを確認したものである。

(3)項は、API 利用アプリケーション及びこれに関連するサービスの提供に関してエンドユーザに損害が生じた時は、速やかに原因を究明するとともに、API 利用者とエンドユーザとの間の利用規約や契約に定められた範囲で、API 利用者がエンドユーザに賠償又は補償すべき旨を定めるものである。

(4)項は、API 提供者が本 API の利用に関して API 利用者に対して損害賠償義務を負う場合を、API 提供者に故意又は重大な過失があった場合に限定するものである。

(5)項は、API 提供者が故意又は重大な過失があり損害賠償責任を負う場合であっても、その賠償の範囲は、現実に発生した直接かつ通常生じる範囲内の損害に限られ、いわゆる逸失利益や特別損害については損害賠償責任を負わない旨を定めるものである。なお、「また、…」以下は、本 API の提供が有償（従量料金・月額課金）である場合に、API 提供者の損害賠償責任の限度額を、特定の月数分の利用料金相当額に限定する場合の文例である。

第 13 条（反社会的勢力の排除）

(1) API 利用者及び API 提供者は、現在、暴力団、暴力団員、暴力団員でなくなった時から 5 年を経過しない者、暴力団準構成員、暴力団関係企業、総会屋等、社会運動等標ぼうゴロ、特殊知能暴力集団等、その他これらに準ずる者（以下総称して「反社会的勢力」という。）のいずれにも該当しないこと、及び次の各号のいずれにも該当しないことを表明し、かつ将来にわたっても該当しないことを確約する。

(ア)反社会的勢力が経営を支配していると認められる関係を有すること

(イ)反社会的勢力が経営に実質的に関与していると認められる関係を有すること

(ウ)自己、自社もしくは第三者の不正の利益を図る目的又は第三者に損害を加える目的をもってするなど、不当に反社会的勢力を利用していると認められる関係を有すること

(エ)反社会的勢力に対して資金等を提供し、又は便宜を供与するなどの関与をしていると認められる関係を有すること

(オ)役員又は経営に実質的に関与している者が反社会的勢力と社会的に非難されるべき関係を有すること

(2) API 利用者及び API 提供者は、自ら又は第三者を利用して、次の各号の一にでも該当する行為を行わないことを確約する。

(ア)暴力的な要求行為

(イ)法的責任を超えた不当な要求行為

(ウ)取引に関して、脅迫的な言動をし、又は暴力を用いる行為

(エ)風説を流布し、偽計を用い、又は威力を用いて、相手方の信用を棄損し、又は相手方の業務を妨害する行為

(オ)反社会的勢力の活動を助長し又はその運営に資する行為

(カ)その他前各号に準ずる行為

(3) API 利用者及び API 提供者は、相手方が本条に違反した場合には、何らの催告をすることなく本

契約を解除することができる。
(4) API 利用者及び API 提供者は、前項に基づく解除により相手方に損害が生じた場合であっても、当該損害の賠償義務を負わず、当該解除に起因して自己に生じた損害については、相手方に対し損害賠償請求することができる。

解説：

本条は、いわゆる反社会的勢力の排除条項を定めるものである。

相手方が反社会的勢力に該当したり、暴力的な要求行為を行った等の場合には、無催告で本契約を解除することを可能としている。また、仮に当該解除に伴って相手方に損害が生じても、その賠償義務は負わないし、逆に自らに損害が生じた場合には、相手方に対して賠償を請求できるものとしている。

第 14 条（秘密保持）

API 利用者は、本 API サービスに関連して API 提供者が API 利用者に対して秘密に扱うことを指定して開示した情報について、API 提供者の事前の書面による承諾がある場合を除き、開示目的以外に利用せず、また、第三者に開示しないものとする。

解説：

本条は、API 利用者の秘密保持義務を定めるものである。

オープン API を前提に、比較的簡潔な文例としているが、API 利用者が API 提供者の重要な秘密情報に触れる可能性が高い場合には、より詳しい秘密保持条項を定めることも検討すべきである。

第 15 条（個人情報の取扱）

API 提供者は、本 API の提供に関して個人情報を取り扱う場合、API 利用者情報及び本 API の利用に係る個人情報について、API 提供者が別途定める個人情報保護方針（プライバシーポリシー）に則り、適切に取り扱う。

解説：

本条は、API 提供者が、本 API の提供に関して個人情報を取り扱う場合、プライバシーポリシーを策定し、それに則り適切に取り扱う旨を宣明するものである。

第 16 条（規約の変更）

API 提供者は、本利用規約を変更する旨及び変更後の内容並びに変更の効力発生時期を予め周知することにより、API 利用者の個別の承諾を得ることなく、本利用規約の内容を変更することができるものとする。変更後の本利用規約の効力は、予め公開した時点より効力を生じるものとする。

解説：

本条は、本利用規約を変更する際の手続を定めるものである。

本 API は複数の API 利用者に同一条件で提供することが想定され、各 API 利用者に適用される規約内容を画一化する必要性が高い。そこで、API 提供者は、①本利用規約を変更する旨、②変更内容、③変更の効力発生時期を事前に周知することにより、API 利用者の個別の承諾を得ることなく、一方的に本利用規約の内容を変更することができるものとしている。なお、事前の周知期間は、変更によって API 利用者に及ぼす影響の程度等に応じて相当な期間を確保するべきである。

第 17 条（提供の終了）

API 提供者は、API 提供者の都合により、本 API の提供を終了することができる。この場合、API 提供者は、API 利用者に事前に通知するものとする。

解説：

本条は、本 API の提供を終了する場合について定めるものである。本 API の提供の終了とは、本 API の提供サービスを終了し、全ての API 利用者に対し永続的に本 API を利用できないようにすることを意味する。

諸事情により本 API の提供を終了する場合には、API 利用者に事前に通知することを要するものとしている。

第 18 条（連絡/通知）

(1) API 提供者から API 利用者への連絡は、書面の送付、電子メールの送信、又はインターネット上の API 提供者の Web サイトへの掲載等、API 提供者が適当と判断する通知手段によって行う。

(2) API 利用者は、API 提供者に対し、有効に受信可能なメールアドレスを届け出るものとし、不通、廃止その他の事情によって届出メールアドレスの変更を要する場合、直ちに所定の変更届出を提出するものとする。API 利用者は、API 提供者から API 利用者宛に送信される電子メールの受信を拒否する設定等を行ってはならない。

(3) API 提供者が API 利用者から届出のあった住所、メールアドレス等宛に書面又は電子メールによって連絡を発信した場合、届出事項の不備もしくは変更、受信拒否設定、通信事情その他 API 提供者の責めによらない事由により延着又は不着となった時であっても、通常到達すべき時に到達したものとみなす。

(4) API 利用者から API 提供者への連絡は、API 提供者が別途定める方法にて行うものとする。

解説：

本条は、API 提供者と API 利用者との連絡の方法について定めるものである。

(1)項は、API 提供者から API 利用者への連絡は、書面の送付、電子メールの送信、又はインターネット上の API 提供者の Web サイトへの掲載等、API 提供者が適当と判断する通知手段によって行うこととしている。

(2)項は、特に重要な連絡手段である電子メールについて、API 利用者にメールアドレスの届出義務を課すとともに、受信拒否設定等を行わないことを義務付けている。届出メールアドレスは、有効に受信可能なものでなければならず、メールアドレスの不通、廃止その他の事情によって変更が必要な場合には直ちに変更届出を提出する義務を課している。

(3)項は、API 利用者への連絡にあたって、API 利用者から届出のあった住所宛に書面を送付した場合や届出メールアドレス宛に電子メールを送信した場合には、例えば住所が変更されていたにもかかわらず変更届出が提出されていなかったり、メールアドレスが廃止済みであるにもかかわらず変更届出が提出されていなかった等の事情によって、書面や電子メールが遅れて届いたり、そもそも届かなかったような時であっても、通常到達すべき時に到達したものとみなすこととしている。

(4)項は、API 利用者から API 提供者への連絡は、API 提供者が別途定める方法で行うこととしている。

第 19 条（権利義務等の譲渡禁止）

API 利用者は、相手方の事前の書面等による承諾のない限り、本契約上の地位及び本契約によって生じる権利義務の全部若しくは一部を第三者に譲渡し、承継し、又は担保に供してはならない。

解説：

本条は、本 API の利用を認めるにあたっては、登録を受けた当該個別の API 利用者の信用が重要であることに鑑み、API 利用者が、本契約上の地位及び本契約によって生じる権利義務を、事前の承諾なく、第三者に譲渡、承継又は担保提供することを禁止するものである。

第 20 条（分離可能性）

本利用規約のいずれかの条項が法令又は裁判所により無効又は執行不能であるとされた場合であっても、本利用規約のその他の条項の有効性に影響を与えないものとする。

解説：

本条は、本利用規約の一部の条項が何らかの法令違反等によって無効又は執行不能であるとされた場合であっても、当該無効な条項のみを分離し、他の条項は有効のまま存続させ、本利用規約の大枠は引き続き維持することを定めるものである。

第 21 条（準拠法及び管轄裁判所）

(1) 本利用規約は日本法に準拠し、日本法に基づいて解釈される。

(2) 本 API の利用及び本利用規約に関する紛争については、（*裁判所名*）を、第一審の専属的な合意管轄裁判所とする。

解説：

本条は、本利用規約の準拠法及び裁判管轄について定めるものである。

5-2-3. ユースケース事例と関連法令

本項では前述の「表 5-2-1-1 先端的サービスに関連するオンラインサービス例」に記載した各オンラインサービスとその取り組みについて、関連法令と新たな規制の特例措置を求める対象として問題となり得る規制とその特例措置の例を紹介する。なお、実際に国家戦略特別区域法に基づき新たな規制の特例措置の整備を求めるにあたっては、先端的区域データ活用事業を実施するために特区内で当該特例措置を講じる必要性を示すとともに、既存の規制の保護法益を勘案し、代替措置を講じるなど既存の規制の趣旨を損なわないこと（相当性）を示すべきものと考えられる。

(1)金融の先端的サービス例と関連法令、想定される特例措置例

ここでは、金融の先端的サービスのユースケース事例として下記の具体的な取り組みを実施するにあたっての関連法令と想定される特例措置例を、その解説とともに整理する。

(a)具体的な取り組み

地方自治体等の組織が、ボランティア活動によって貯まるポイントを地域電子通貨として発行する。他の行政サービスでの支払いやボランティア以外の地域貢献活動などと広く連携することで利用シーンが広がることが期待される。

(b)関連法令

資金決済に関する法律（資金決済法）、金融商品取引法

(c)想定される特例措置

資金決済法上、発行者以外の者との間で使用することができ、物品やサービスの支払手段として利用可能で対価を得て発行されるポイントは、第三者型前払式支払手段に該当する（資金決済法 3 条 1 項・5 項）。対価には金銭に限らず財産的価値があるものは全て含まれるため、ボランティア（役務）の謝礼としての発行も「対価を得て」に該当する可能性がある。資金決済法上、第三者型前払式支払手段の発行者は、以下のような義務を負うこととなり、地方自治体の管理下で事業を行う場合でも例外はなく、通常の資金決済業者と同等の労力と運用コストが発生することとなる。

- 事前の登録義務（資金決済法 7 条）

- 情報提供義務（資金決済法 13 条）
- 発行保証金の供託義務（資金決済法 14 条。毎年 3 月末又は 9 月末における未使用残高が 1,000 万円を超える時は、その額の 2 分の 1 の額以上の額に相当する額を供託等する義務）

一方、地方自治体が発行する場合は資金決済法の適用除外とされているが（資金決済法 4 条 3 号）、NPO 法人や民間事業者が発行する場合には適用対象となる。また、発行者何如にかかわらず発行の日から 6 か月内に限り使用できるものも適用除外とされているが（資金決済法 4 条 2 号、資金決済に関する法律施行令 4 条 2 項）、それを超えるものは適用対象となる。

以上より、本取り組みを実施する上での特例措置例としては、前払式支払手段に対する規制が適用除外となる有効期間の要件（6 か月）の緩和、前払式支払手段発行者の事前登録要件・発行保証金の供託義務等の緩和が考えられる[23]。

さらに、ボランティアポイントを暗号資産として組成する場合には、暗号資産交換業者に係る資金決済法上の規制が問題となる。また、いわゆるセキュリティトークンとして組成する場合には、電子記録移転権利に係る金融商品取引法上の規制が問題となる可能性があることに注意が必要である。そのため、地域電子通貨を発行する場合は、暗号資産と同等の技術を使う場合でも前払式支払手段の方式を採用し、上述の特例措置を検討することが推奨される。

(2)医療の先端的サービス例と関連法令、想定される特例措置例

ここでは、医療の先端的サービスのユースケース事例として下記の具体的な取り組みを実施するにあたっての関連法令と想定される特例措置例を、その解説とともに整理する。

(a)具体的な取り組み

運動や食事データ等のライフログや医療データを連携することで、健康～未病～治療のサイクルをシームレスに繋ぐヘルスケアプラットフォームを構築する。また、あらゆる運動データをウェアラブルデバイスから収集し、健康状態と突合し、パーソナライズ化された運動メニューの推奨や発病リスクのアラート、健康 e ラーニングの配信など、健康～未病に関わる取り組みを健康アプリを通じて促進する。発病した際は、自覚症状が出る前に AI により受診勧奨の通知が行われると同時に、ライフログや健康診断データ等が電子カルテに統合され、最適なオンライン診療とオンライン服薬指導を可能にする。治療後は、再発予防のためのパーソナライズ化された最適な取り組み（運動・食事等）をレコメンドし、健康維持を促進する。これらを実現・促進するために、オンライン診療、オンライン服薬指導と混合診療を可能・拡大する。

(b)関連法令

オンライン診療の関連法令は医師法、オンライン服薬指導の関連法令は医薬品、医療機器等の品質、有効性及び安全性の確保等に関する法律（薬機法）、混合診療の関連法令は健康保険法

(c)想定される特例措置例

オンライン診療とは、遠隔医療のうち、医師 - 患者間において、情報通信機器を通じて、患者の診察及び診断、診断結果の伝達や処方等の診療行為をリアルタイムにより行う行為をいう。医師法上、医師には自ら診察する義務が課されているところ（医師法 20 条）、従来、診察とは対面で行うことであり、オンライン診療は認められないと解されてきた。平成 27 年以降に規制緩和が進み、平成 30 年

[23] 構造改革特区第 26 次検討要請における熊本県から金融庁への要望事項（地域通貨券の有効期限の要件緩和） https://www.kantei.go.jp/jp/singi/tiiki/kouzou2/boshu26/kaitou0302/k_02.pdf

の厚生労働省「オンライン診療の適切な実施に関する指針」（令和元年 7 月改訂）[24]などによって、オンライン診療が認められる範囲が大きく拡大された。もっとも、同指針によれば、例えば以下のような制限が設けられている。

- 初診及び急病急変患者は、原則として対面による診療を行うこと
- 新たな疾患に対して医薬品の処方を行う場合には、原則として直接の対面診療に基づいて行うこと
- オンライン診療では、リアルタイムの視覚及び聴覚の情報を含む情報通信手段を採用し、文字、写真及び録画動画のみのやり取りで完結してはならないこと
- オンライン診療を行う前に、オンライン診療で行う診療内容等を記載した診療計画を策定すること
- なお、新型コロナウイルス感染症の発生を踏まえ、令和 2 年 4 月 10 日、厚生労働省の事務連絡[25]により、時限的・特例的措置として、初診からのオンライン診療が全ての疾患について認められることになった。また、平成 30 年度診療報酬改定によりオンライン診療に対する診療報酬（オンライン診療料）が新たに算定できることとなった。もっとも、オンライン診療料の算定には例えば以下のような制限がある。
  - 算定対象となる疾患が限定されていること（特定疾患療養管理料、地域包括診療料、小児科療養指導料、認知症地域包括診療料、てんかん指導料、生活習慣病管理料、難病外来指導管理料、在宅時医学総合管理料、糖尿病透析予防指導管理料、精神科在宅患者支援管理料、在宅自己注射指導管理料（糖尿病、肝疾患、慢性ウイルス肝炎）の算定対象となる患者又は定期的に通院が必要な慢性頭痛患者）
  - 事前に 3 月以上の対面診療を行っていること
  - 対面診療を 3 か月以内ごとに実施すること
  - 緊急時には概ね 30 分以内に当該保険医療機関による対面診療が可能であること
  - 対面診療に比べ、算定できる管理料の点数が低いこと

以上より、本取り組みとしてオンライン診療を実施する上での特例措置例としては、オンライン診療の拡大及び恒常化とオンライン診療料の算定対象の拡大と要件緩和、点数引き上げといった措置が考えられる[26]。

オンライン服薬指導については、薬機法上、院外処方の場合には薬剤士の対面による情報提供及び指導が義務付けられており、オンライン服薬指導は認められていなかった（同法 9 条の 3）。令和 2 年 9 月 1 日施行の改正薬機法により、オンライン服薬指導が認められるようになった。ただし、対象は、オンライン診療又は訪問診療によって交付された処方箋に限られている（対面診療後のオンライン服薬指導は不可）。また、オンライン診療に先立って対面による服薬指導がなされていることが必要である。

以上より、本取り組みとしてオンライン服薬指導を実施する上での特例措置例としては、オンライン服薬指導の拡大措置が考えられる。

---

[24] https://www.mhlw.go.jp/content/000534254.pdf

[25] https://www.mhlw.go.jp/content/000621247.pdf

[26] 令和 2 年 10 月 21 日規制改革推進会議第 2 回医療・介護ワーキング・グループ　提出資料 1-1-1 https://www8.cao.go.jp/kisei-kaikaku/kisei/meeting/wg/iryou/20201021/agenda.html

混合診療とは、一連の診療の中で保険診療と保険外診療が混在することをいい、同一の疾患に対して保険診療と保険外診療を併用した場合、一連の治療行為とみなされ、原則として、全体について保険外診療となる。平成 18 年に導入された保険外併用療養費制度により、厚生労働大臣の定める療養については、保険外併用療養費が支給されることとなった（健康保険法 86 条）。もっとも、対象となる療養は以下のとおり限定されている[27]。

- 評価療養（厚生労働大臣が定める高度の医療技術を用いた療養その他の療養であって、前項の給付の対象とすべきものであるか否かについて、適正な医療の効率的な提供を図る観点から評価を行うことが必要な療養）
- 患者申出療養（高度の医療技術を用いた療養であって、当該療養を受けようとする者の申出に基づき、前項の給付の対象とすべきものであるか否かについて、適正な医療の効率的な提供を図る観点から評価を行うことが必要な療養として厚生労働大臣が定めるもの）
- 選定療養（被保険者の選定に係る特別の病室の提供その他の厚生労働大臣が定める療養）

以上より、本取り組みにおける混合診療に関する特例措置例としては、混合診療の適用範囲の拡大が考えられる。

(3)教育の先端的サービス例と関連法令、想定される特例措置例

ここでは、教育の先端的サービスのユースケース事例として下記の具体的な取り組みを実施するにあたっての関連法令と想定される特例措置例を、その解説とともに整理する。

(a)具体的な取り組み

多様なデジタルコンテンツを駆使したアクティブラーニングを採用したり、様々な分野の専門家に免許状を有する教員と同等に授業を担当してもらえるようにする。また、オンデマンド型のｅラーニング形式の授業を組み合わせた完全オンライン遠隔教育を実施する高等学校を実現し、遠隔授業で全単位取得可能にする。

(b)関連法令

デジタルコンテンツや専門家を活用したアクティブラーニングは著作権法と教育職員免許法、オンデマンド型遠隔教育の実施は学校教育法と著作権法。

(c)想定される特例措置

デジタルコンテンツを活用したアクティブラーニングの実現に関連しては、平成 30 年著作権法改正により、授業目的公衆送信補償金制度が創設され、それまでは個別に権利者の許諾が必要とされていた著作物のインターネット送信に関し、学校その他の教育機関が、授業の過程における利用に供することを目的とする場合には、指定管理団体(SARTRAS)に一括して補償金を支払うことで、個別の許諾を要することなく著作物の公衆送信を行うことが可能となった（同法 35 条 1 項・2 項）[28]。同改正法は、令和 3 年 4 月頃の施行が想定されていたが、新型コロナウイルス感染症の流行によるオンラインでの遠隔授業等のニーズの高まりを踏まえ、予定を早めて令和 2 年 4 月 28 日から施行されている。補償金の額は文部科学大臣が認可するものとされている（同法 104 条の 13。なお、令和 2 年度に限り、特例的に無償として補償金額の認可が行われている[29])。一方、専門家を活用したアクティブラー

[27] 厚生労働省　保険診療と保険外診療の併用について　https://www.mhlw.go.jp/topics/bukyoku/isei/sensiniryo/heiyou.html

[28] https://www.bunka.go.jp/seisaku/chosakuken/hokaisei/h30_hokaisei/

[29] 文化庁　授業目的公衆送信補償金の額の認可について　https://www.bunka.go.jp/seisaku/chosak

ニングの実現に関しては、昭和 63 年の教育職員免許法の改正により、大学での養成教育を受けていない者に、都道府県教育委員会の行う教育職員検定により免許状を授与する制度が創設されており、普通免許状を有しない者も一定の範囲において教育職員として授業を担当することが可能である（教育職員免許法 4 条 3 項)。しかしながら、特別免許状を授与するには、任命・雇用権者の推薦、都道府県教育委員会の教育職員検定の手続、学校教育に関し学識経験を有する者等からの意見聴取が必要とされており（教育職員免許法 5 条 3 項・4 項・5 項)、利用が進んでいない状況にある。

以上より、デジタルコンテンツや専門家を活用したアクティブラーニングを実施する上での特例措置例としては、著作権法上は授業目的公衆送信補償金の額の特例、教育職員免許法上は特別免許状の付与の特例といった措置が考えられる。

オンデマンド型遠隔教育の実施に関連しては、平成 27 年学校教育法施行規則改正により、高等学校においては、「高等学校が、対面により行う授業に相当する教育効果を有すると認めた」場合に、「授業を、多様なメディアを高度に利用して、当該授業を行う教室等以外の場所で履修させること」が認められるようになった（同施行規則 88 条の 3〔当時は 88 条の 2〕、平成 27 年文部科学省告示第 92 号[30])。しかし、「同時かつ双方向に行われるものである」ことが要件とされており（平成 27 年文部科学省告示第 92 号)、また、「原則として、受信側の教室に当該高等学校等の教員を配置する」ことが求められている（通知 27 文科初第 289 号[31]）ことから、オンデマンド型の e ラーニングによって生徒が時間や場所の制限を受けずに学ぶことは、正式な授業とは認められない。また、「多様なメディアを高度に利用して、当該授業を行う教室等以外の場所で履修させる」授業による習得する単位数は、全過程の修了要件 74 単位のうち 36 単位を超えてはならないものとされている（同施行規則 96 条 2 項)。

以上より、オンデマンド型遠隔教育を実施する上での特例措置例としては、学校教育法上は遠隔授業の実施に関する同時双方向要件や受信者側の教室への教員配置の撤廃や遠隔授業による単位取得上限の撤廃といった措置が考えられる。また、オンデマンド授業で使用する教材の公衆送信に係る著作権法上の扱いについては、上記のデジタルコンテンツを活用したアクティブラーニングの実現に関する説明と同等であるため、そちらを参照されたい。

uken/92728101.html

[30] https://www.mext.go.jp/component/a_menu/education/detail/__icsFiles/afieldfile/2015/05/20/1358054_02.pdf

[31] https://www.mext.go.jp/a_menu/shotou/kaikaku/1360985.htm

5-3. API 標準仕様案

5-3-1. はじめに

スーパーシティでは、API を公開することによる新たな先端的サービスの創出が期待されている一方で、我が国における API の提供方法に関しては検討が始められて間もない状況にあると認識している。例えば、内閣官房情報通信技術(IT)総合戦略室が発行する API テクニカルガイドブックにて、一般的な API の標準は定められているが、スーパーシティにおける API として準拠すべき標準仕様が明記されているわけではない。API によるデータの提供の本質的な意義は、サービスによってデータが活用されることで、サービスを利用する人々の QoL を向上することである。そのため、API 利用者であるサービス提供者が API を開発しやすい、標準化された API を提供することが求められる。本節では、インターネットを通じて Web で公開・提供するオープン API を主な対象とし、API 利用者の使い勝手向上と利活用促進を目的として、API 提供に際し共通化すべき標準仕様案を定める。

以降では、スーパーシティで提供する API が準拠すべき標準的な要件について、先頭に（必須）又は（推奨）を示し、太字下線の表記にて記載する。なお、API の相互運用性を確保する上で重要なもの、セキュリティ観点で特に遵守すべきと考えられるもの、調査した複数のガイドラインで定められており一般的なルールと認められるものについて（必須）とし、それらより重要度が低く、API を整備するプラットフォームや開発フレームワーク等の制約によって採用可否が変わり得るものについて（推奨）とした。

（必須）スーパーシティの API が準拠すべき必須要件
（推奨）スーパーシティの API が準拠すべき推奨要件

また、例示がある部分は次のように記載し、着目点を太字下線で強調する。

良い例：https://api.example-city.jp/v1/**magazines**/1234/**title**
悪い例：https://api.example-city.jp/v1/**zasshi**/1234/**toshomei**

5-3-2. 基本となる考え方

(1)API の通信プロトコル

（必須）API と API 利用者（サービス）を接続する通信路を暗号化する。
（推奨）API と API 利用者（サービス）を接続するプロトコルを HTTPS とする。

スーパーシティにおける API は、主として先端的サービスに利用されることが想定される。また、先端的サービスは、サービス利用者がインターネットを介してアクセスし、Web サービスとして提供されることが想定される。このようなスーパーシティの API では、広く一般に公開が可能なデータだけでなく、取り扱うデータの特性により公開範囲を制限すべきものや、個人情報などのプライバシーに関するものを取り扱う可能性が高い。インターネットを通じて API が実行される特性上、その通信経路上に信頼できないハードウェアやソフトウェアが介在するおそれがあるため、API と API 利用者の間の通信路を暗号化されている必要がある。HTTPS 通信に脆弱性が含まれないかどうかは、Qualys 社の SSLlabs が提供する SSL Server Test[32]などの仕組みを利用して定期的に脆弱性チェックを行い、必要に応じてセキュリティ対策を行うことが推奨される。また、クライアントアプリから非 SSL の URL アクセスがあるケースでは対応する HTTPS の URL にリダイレクトせず、HTTP アクセスしたことへのエラーを返却するよう実装することが推奨される。

[32] Qualys 社の SSLlabs が提供する SSL Server Test(https://www.ssllabs.com/ssltest/)

(2)API を呼び出すインターフェース

（必須）API のインターフェースを OpenAPI 仕様に準拠した設計とする。

（必須）OpenAPI 仕様のファイル名は openapi.json もしくは openapi.yaml ファイルとする。

スーパーシティで整備する API はインターネットを通じて広く公開され、様々な API 利用者に対して提供される。また、API 利用者は複数の API を複合的に利用して、分野を横断する新たなサービスを提供する場合もある。そのため、API 仕様の記載内容が統一されていないと、インターフェースやデータの互換性を判別しにくくなり、サービス開発に支障をきたすおそれがある。そのため、API の仕様は、Web API の仕様を記述するデファクトスタンダードである OpenAPI Specification（OAS、OpenAPI 仕様とも呼ばれる）での公開を前提とし、認証方式を記述できる OpenAPI 仕様 3 系に準拠して作成する。なお、OpenAPI 仕様は JSON 又は YAML での記載が可能であるが、どちらのフォーマットも相互変換可能であるため、記述方法はどちらを利用してもよい。

（推奨）API のインターフェースは、Pragmatic REST に基づく設計とする。

API を設計する際は、Representational State Transfer(REST)に基づいたインターフェース設計に統一することで、利用者にとって理解しやすく使いやすい API を提供することができる。REST の考え方に基づいて設計された API は、REST API 又は RESTful API と呼ばれる。REST API では、URL でリソースを、HTTP メソッドで操作（作成、取得、更新、削除）を指定し、操作に必要な情報を HTTP リクエストに含めるステートレスな設計が求められる。しかしながら、API を完全にステートレスなものとして設計・提供することは事実上困難である。そのため、スーパーシティにおける API では、実用的な範囲で REST に準拠する Pragmatic REST と呼ばれる考え方に準拠して API を設計することを推奨する。

(3)API で取り扱うデータモデルとフォーマット

（推奨）API が取り扱うデータモデルとして、業界標準の形式を採用する。

API で取り扱うデータがどのようなデータ構造で表現されるかは、複数の API を組み合わせて利用するサービスの開発難易度と期間に大きな影響を与える。そのため、スーパーシティの API が取り扱うデータモデルは、なるべく統一された業界標準の形式を採用することが望ましい。具体的なデータモデルの考え方については、「7 調査結果（ウ）データモデルの方針」を参照頂きたい。

（推奨）API が取り扱うリクエスト/レスポンスボディフォーマットを JSON 形式とする。

API は、データ作成に必要なパラメータ情報をリクエストボディに設定したり、作成されたデータや検索結果などの API 実行結果をレスポンスボディに設定することができる。リクエスト又はレスポンスのボディとして設定される情報は通常どのような形式でも動作するが、スーパーシティには多数の API が整備されることが想定されるため、一般的なデータ構造をテキストで表現することに長けた JSON 形式で統一することを推奨する。JSON は、XML や CSV と比べてデータ構造の検証や変換などコンピュータが取り扱いやすいフォーマットであり、多数の API で広く採用されている。ただし、音声や動画像など、JSON では表現が難しいバイナリデータを取り扱う場合は、この限りではない。

（推奨）属性値のフォーマットを、国際標準やデファクトスタンダード等に従って設定する。

日時や期間、言語などの情報を表現する場合には、国際標準やデファクトスタンダードが広く採用されており、これらと異なる独自のフォーマットを利用することは、API 利用者にとってデータ・フォーマットを調整する手間を増やしてしまう。そのため、下表に示すような標準に合わせてデータフォーマットを決定することが望ましい。

表 5-3-2-1　推奨する表記方法の例

内容	表記方法	表記例
日時	ISO8601 に準拠すること。 日付のみであれば、全て半角で YYYY-MM-DD。 日時（日本時間）の場合、日付と時間の間に T を挟み、全て半角で YYYY-MM-DDTHH:mm:ss+0900 と記載する。日本では UTC から 9 時間の時差があるため、+0900 とする。	日付のみであれば、『2017-02-06』、日時（日本時間）であれば、『2017-02-06T13:50:40+0900』で表す。
期間	ISO8601 に準拠すること。 P[n]Y[n]M[n]DT[n]H[n]M[n]S P は、期間表現の開始時に配置される期間指定子。	『P3Y6M4DT12H30M5S』は「3 年、6 か月、4 日、12 時間、30 分、5 秒」の期間を表す。
言語	ISO 639-1 に準拠すること。 日本語を必須、英語については推奨とする。多くの言語に対応する必要がある場合には、ISO639-2 や ISO639-3 などを検討する。	日本語の場合『ja』、英語の場合『en』で表す。
貨幣	ISO 4217 に準拠すること。	日本円の場合『JPY』、アメリカドルの場合『USD』で表す。 ※あくまで円やドルなどの貨幣単位について記述するもので、貨幣の量の情報は含まれない。
性別	ISO 5218 に準拠すること。	男性の場合『1』、女性は『2』、それ以外は『9』、不明な場合には『0』で表す。
国	ISO 3166-1 alpha-2 に準拠すること。	日本の場合『JP』、米国の場合『US』で表す。
都道府県	ISO 3166-2:2013JIS X 0401 に準拠すること（日本の場合、ISO3166-2:JP を参照する）。	東京都の場合は『JP-13』で表す。
市区町村	JIS X 0402 に準拠すること。	東京都千代田区の場合は『101』で表す。
経緯度	API で提供する場合は、地理空間情報（緯度経度データを含む）は GeoJSON の形式で提供することが望ましく、経緯度データは WGS84（米国世界測地系）に変換して格納することを推奨する（RFC7946「The GeoJSON Format」で、GeoJSON は WGS84 に準拠することを明記）。	東京都千代田区永田町 2 丁目 3-1 は、WGS84 で『35.672947,139.742622』と表す。
単位	ISO/IEC 80000 に記載のある国際単位系に準拠すること。	メートルは『meter』、グラムは『gram』、アンペアは『ampere』と表す。

(4)URL リソース名と属性名の命名規則

（必須）データモデルと属性名として、半角英数字、ハイフン、及びアンダースコアのみを使用する。

API で取り扱うリソース（データ）の名称が API を呼び出す際に URL に含まれる可能性がある。また、URL には「%」を使った記法（URL エンコード）を利用することで漢字や記号を含む表現が可能であるが、可読性が著しく低下する。API の URL をシンプルに保つため、リソース名には半角英数字、ハイフン、及びアンダースコアのみを用いる。ただし、API リクエスト時に指定するパラメータ（URL クエリなど）に

ついては、この限りではない。

（推奨）API で取り扱うデータモデル名と属性名として、標準的なアメリカ英語の名詞を使用する。
（推奨）API で取り扱うデータモデル名と属性名として、スネークケースを使用する。
（推奨）URL 中のリソース名を複数形で記載する。

多数の API が整備された時に、実質的に同一のリソースを指しているのに URL が異なると、それらの API を組み合わせて利用するサービス側の開発負荷が高くなる。そのような状況を回避するため、API で取り扱うリソース名とその属性値の記述方法を統一する。例えば、複数地域のイベントを横断的に検索するようなサービスを検討する場合、次のように良い例と悪い例の 2 種類の API があると、サービス内部でデータの変換を行わなければならない。

良い例：https://api.example-city.jp/v1/social_events/1234/start_date
悪い例：https://api.example-city.jp/v1/SyakouGyoji/1234/Kaisaibi
　　https://api.example-city.jp/v1/social-events/1234/start-date
※URL 中の 1234 はイベント ID を示す。

上記の良い例は、スネークケース（アンダーバーで単語を区切る記法）で記載されたリソース名「social_event」が URL 中にあるため複数形で記載されており、属性名「start_date」もスネークケースで記載されている。悪い例の 1 つ目では、リソース名と属性名が日本語ローマ字表記（英語ではない）かつ、キャメルケース（単語の 1 文字目を大文字にする記法）で記載されている。悪い例の 2 つ目では、ケバブケース（ハイフンで単語を区切る記法）で記載されている。

（推奨）プログラミング言語でよく利用されるキーワードとの競合を避けた名前を使用する。

API を呼び出すプログラムでは URL を取り扱うため、その中の一部が var、for、define などのプログラミング言語のキーワードと競合してバグを引き起こす可能性がある。そのため、このようなキーワードをリソース名として使用しないよう配慮する。

（推奨）一般的によく知られる名称については略語を使用する。

configuration、identifier、statistics など、API やサービスの開発でよく使用される単語については、config、id、stats などの略語を使用することで、スペルミスに起因するバグを防止したり、API を呼び出すプログラムのソースコードの可読性を改善する上で効果的な場合がある。略語を使用する場合は全て小文字で記載し、複数の単語の先頭文字だけを記述する場合は全て大文字で記載する。なお、日本語由来の略語（区分→kbn など）は読み間違い（kbn→区分、鞄、気分 など）が起きやすいため、省略しない。

良い例：configuration → config
　　identifier → id
　　specification → spec
　　statistics → stats
　　Amazon Web Services→AWS
悪い例：kubun → kbn

（推奨）属性名には前置詞を使わない短くシンプルな名前を使用し、形容詞を名詞の前に記載する。

短くシンプルで可読性の高い名前を使用することで、読み間違いや誤解を避ける。また、属性名はAPIを通して送受信されるデータに含まれ、何度も転送される可能性があるため、短い名称を使用することでデータ通信量の節約にも多少の効果がある。

良い例：error_reason imported_objects finished 悪い例：reason_for_error Objects_imported Is_finished

（必須）属性値が数値となる場合は、属性名に単位を含める。

属性値が整数又は小数で、一般に単位のある量を扱う場合、API利用者の誤解を招かないよう、属性名に単位を含める。単位のない値（割合など）についてはこの限りではない。属性名で利用する単位については、ISO/IEC 80000に記載のある国際単位系（英語表記）を使用する。

良い例：send_traffic_bytes nearest_station_distance_meters 悪い例：send_traffic nearest_station_distance

（必須）範囲を表す属性名は、属性名の先頭を「start」又は「end」とする。

期間や範囲を表す属性を取り扱う場合、属性名の先頭に「start」又は「end」を記載した共通の属性名を使用することで表現し、統一することで、API利用者の誤解を避ける。

良い例：start_date end_date 悪い例：begin_date finish_date

（必須）属性値が配列で表現される場合、その属性名を複数形とする。

イベントのスポンサー情報など、複数のデータを含む可能性のある属性を表す場合は、一般的に配列でデータが表現される。属性値が配列となる属性名は複数形で統一し、API利用者の誤解を避ける。

良い例：sponsors 悪い例：sponsor

（必須）商品数などを表す属性名には「count」を含める。

商品棚の情報や検索結果に含まれる商品数など、数量を示す属性値の末尾には「count」を記載することで、属性名がどのような情報を示すか表現を統一する。

良い例：items_count 悪い例：items

5-3-3. APIリクエストの標準

(1)リクエストURL

（必須）URLをRFC 3986に準拠し、RFC 6570に示されるURI Template記法で表現できる形式とする。

URL には使用可能な文字とそうでない文字があり、一般的なプログラムが URL として解釈可能なフォーマットが定められている。そのため、URL はその記述方法について定めた国際標準 RFC 3986 に準拠したものを使用する。また、URL を一定のパターンで表現する方法についても RFC 6570 として標準が定められているため、この表記方法にも適合するよう URL を設計する。

（必須）API のバージョン（v1、v2 等）を URL に含める。

（推奨）API のバージョンを上げた場合に下位互換性を保つ設計とする。

API はバージョンアップの際にインターフェース仕様が更新され、利用できる URL やメソッド、パラメータの設定方法などが変更される可能性がある。そのため、一般に API のインターフェース仕様が変更される場合は、一定の期間古いバージョンと共存させ、移行期間を設ける。そのため、API の URL にはバージョンを含める。また、API を利用する既存のプログラムに対する変更がなるべく少なくなるよう、API のバージョンアップ時は下位互換性をなるべく保つ方針をとることが望ましい。

良い例：https://api.example-city.jp/**v1**/social_events/
　　https://api.example-city.jp/**v2**/music_events/
悪い例：https://api.example-city.jp/social_events/
　　https://api.example-city.jp/music_events/

（推奨）地域が用意する全ての API でドメイン名を統一する。

スーパーシティでは個人情報など取り扱いに注意を要するデータを API で取り扱うことが想定され、地域のデータ連携基盤で提供される API が複数ドメインに分かれていると、どれが公式の API であるか混乱を招く。また、悪意ある第三者により類似する API が作られると、重要な情報が抜き取られるなどのおそれもある。そのため、地域で用意する API のドメインは全て統一し、分散管理とならないことが望ましい。

良い例：https://api.**example-city.jp**/v1/social_events/
　　https://api.**example-city.jp**/v1/music_events/
　　https://api.**example-city.jp**/v1/exhibition_events/
悪い例：https://api.**example-city.jp**/v1/social_events/
　　https://api.**example-city.net**/v1/music_events/
　　https://api.**example-city.com**/v1/exhibition_events/

（必須）URL に API を表す「api」を含める。

（推奨）サブドメイン名に API 提供を表す「api」を含める。

（必須）API の検証利用目的のための環境を用意する。

（推奨）API の検証利用目的のためのサブドメイン名を「api-sandbox」とする。

地域の Web ページが既にある場合、URL のパスのみで Web ページと API を分けて提供することは技術的に可能であるが、それがどちらの機能として提供されているのか URL から判読しづらい。そのため、サブドメイン名には API を示す「api」を含めて提供することが望ましい。また、本番用の API を公開すると本番環境にテストデータが残るなどの悪影響を与える可能性があるため、検証環境を別途用意する。そのサブドメイン名を「api-sandbox」とすることで、URL からも判読可能な API の検証環境を提供可能である。

良い例：https://api.example-city.jp/v1/social_events/
　　https://api-sandbox.example-city.jp/v1/social_events/
悪い例：https://supercity.example-city.jp/social_events/

（必須）地域の開発者ポータルの URL には「developer」を含める。
（推奨）地域の開発者ポータルのサブドメイン名を「developer」とする。

API のドキュメントを公開する Web サイトの URL には「developer」を含めることが一般的である。そのため、地域の開発者ポータルにおいても同様に、サブドメイン名を「developer」として提供することが望ましい。

良い例：https://developer.example-city.jp/tutorial
悪い例：https://www.example-city.jp/api/riyouhouhou

（必須）取り扱うデータ種が異なる場合は URL を分割する。

同じ URL パスの API で異なるデータを取り扱う設計にすると、API を呼び出すパラメータの指定ミスで Web 画面が正しく表示されないなどの不具合の原因となりやすい。また、URL でリソースを示すという REST の考え方にも反する。そのため、取り扱うデータ種が異なる（雑誌情報とイベント情報など）場合は URL を分割して提供する。

良い例：https://api.example-city.jp/v1/social_events
　　https://api.example-city.jp/v1/magazines
悪い例：https://api.example-city.jp/v1/objects?type=social_events
　　https://api.example-city.jp/v1/objects?type=magazines

（必須）計算や翻訳などの処理を実行する API の URL を名詞ではなく動詞とする。

REST API では、API のパスで対象のリソース（データモデル、ID、及びその属性値）を指定し、HTTP メソッド（POST、GET、PUT、PATCH、DELETE など）で操作を指定する。ただし、API 設計の都合上、計算、翻訳、変換処理など、それ自体がデータモデルではなく処理を表す API を作成する必要がある場合は、名詞ではなく動詞を使用し、リソース名との誤解を避ける。

良い例：https://api.example-city.jp/v1/translate
悪い例：https://api.example-city.jp/v1/translation
　　※与えられたデータを翻訳する API の例。

（必須）データ種の階層構造を表現する URL パスを設計する。
（必須）データ種の ID にはデータベースのシーケンス番号を指定しない。
（必須）データ種の ID には URL で使用可能な ASCII 文字を使用する。
（推奨）UUID などデータ種と関連しない値を ID として使用する。

API の URL パスは、取り扱うデータ種に合わせて設計し、スラッシュで区切るごとに範囲が限定されるような構造とする。例えば、ID を持つ特定のデータを指定する場合、データ種とその ID をスラッシュ区切りで順に記載する。さらにその属性値を取得する場合は、さらにスラッシュで区切って指定する。これらの ID や属性値は URL 中に含めてリクエストされ得ることを考慮し、URL で使用可能な ASCII 文字を使用する。

データ種の ID としてデータベースのシーケンス番号を使用すると、ID が推測されやすく攻撃の対象になりやすいため、シーケンス番号の使用を推奨しない。UUID など、データ種に関係なくランダムな ID を指定することで、外部からの攻撃によるセキュリティリスクを下げることができる。

JSON データの例：

```
{
  "id": "ABCD",
  "type": "social_events",
  "sponsors": [
    "Acme, Inc.",
    "John Doe"
  ],
  "location": "1-2-3, Example City, Tokyo, Japan",
  "startDate": "2025-12-20T10:00:00+09:00",
  "endDate": "2025-12-23T21:00:00+09:00"
}
```

上記データを指す URL について
良い例：https://api.example-city.jp/v1/social_events/ABCD
悪い例：https://api.example-city.jp/v1/ABCD/social_events
　　https://api.example-city.jp/v1/social_events/123456

上記データのスポンサー一覧を指す URL について
良い例：https://api.example-city.jp/v1/social_events/ABCD/sponsors
悪い例：https://api.example-city.jp/v1/social_events/sponsors/ABCD

（必須）URL 末尾のスラッシュ有無で API の動作を変更しない。

API の URL 末尾のスラッシュ有無で、API の動作に違い（呼び出される機能が変わるなど）があると、API 利用者の混乱を招く。そのため、スーパーシティで整備する API では、以下の例どちらの記載でも同じ結果を返却するよう設計する。

良い例：https://api.example-city.jp/v1/social_events
　　https://api.example-city.jp/v1/social_events/
　　※上記どちらの URL でも API の動作が変わらない。

（必須）URL には機密情報を含まない設計とする。

リクエストする URL 中にクレジットカード番号や個人番号、又は個人の氏名や電話番号などの機密情報が含まれていると、API とは別の中継サーバ(SSL 終端を担うリバースプロキシやロードバランサなど)のログにこれらの情報が記録される可能性がある。万一の情報流出や、情報開示範囲を最小限とするためにも、機密としたい情報は URL に含めず、アクセストークンなどの仕組みを使ってやり取りする方法を検討する。

良い例：https://api.example-city.jp/v1/Credentials/my_account/address
悪い例：https://api.example-city.jp/v1/Credentials/taro_yamada/address

(2)クエリパラメータ

（必須）クエリパラメータを指定しなくても API を利用できる設計とする。

API は、実行結果をフィルタリングしたり並べ替えたりする目的で、クエリパラメータを指定できる設計とする場合がある。しかし、API 実行時には既に URL で対象とするリソース（データ種）とその ID、ならびに属性名を含めることができ、HTTP メソッドでその操作を表現できるため、あくまでクエリパラメータはオプション機能として提供し、必須としない。

（推奨）クエリパラメータ名は、Web API で標準的に利用されるものを採用する。
（必須）クエリパラメータ名は半角英数字で始め、全て小文字で記載する。
（必須）1 つのクエリパラメータに複数の値を指定する際には、カンマ「,」を用いる。
（必須）複数の値を指定するクエリパラメータ名を複数形、それ以外を単数形で記載する。
（推奨）GET リクエストにおいて、返却データが 10 件以上になる場合、「limit」もしくは「since」「until」パラメータを指定し、返却件数を制限する。
（推奨）レスポンスデータ項目として 10 件以上の項目を含む場合、「fields」パラメータを指定し、要求する項目のみ返却する。
（推奨）リクエストデータに外字を含まない。

下表は、Web API で使用される標準的なクエリパラメータの例である。クエリパラメータの利用を検討する場合は、独自設計を避け、標準的に利用されているものの採用を推奨する。

表 5-3-3-1　Web API の標準的なクエリパラメータ例

クエリ パラメータ	用途
limit	1 回のリクエストにて返却されるデータ件数を指定する。
fields	不要なデータの返却を防ぎ、データ通信量やデータ検索負荷を削減するために利用する項目。指定した項目のみを返却する。項目が複数ある場合はカンマ「,」区切りで指定する。
offset	返却するデータの中で、先頭から「offset」で書かれた件数のデータを返却しないことを指定する。
sort	指定された条件を基に並び替えて結果を返却する。
encode	文字コードを指定する。
page	返却してほしいデータの開始位置を指定する。「limit」と合わせて用いることが多い。
since	指定された日付以降のデータを返却する。
until	指定された日付以前のデータを返却する。

(3)リクエストメソッド

（必須）API のデータ操作内容により、HTTP メソッドを使い分ける。

REST API では、URL でリソースを指定し、HTTP メソッドでその登録や取得などの操作を指定する。API を実行する際は、その操作により GET、POST、PUT、PATCH、DELETE の 5 種類のメソッドを使い分ける設計

とする。

GET メソッドを、URL で指定されたリソースを取得する際に使用する。指定されたリソースに対する変更は行われず、情報取得の目的で利用される。一般的にボディを指定しない。そのため、API 実行時に URL 指定されたリソースを誤って削除したり書き換えたりする心配なく利用できる。

POST メソッドを、新しいデータを作成する際に使用する。URL には作成するリソースを指定し、作成内容はボディに設定する。ボディには新しく作成するデータの基本的な情報を記載するが、データの ID や作成時刻など、API サーバ側で管理すべき属性値をボディに含めない。また、データ作成時のデフォルト値を決めておけば、ボディに指定する内容を削減できる。

PUT メソッドを、URL で指定したデータ全体をボディに指定した内容に変更する目的で使用する。変更できない属性値（対象リソースの ID や作成時刻、更新時刻など）を除き、全ての属性値が更新されるため、属性値を設定せずにリクエストを送信すると、その属性値の値を削除する処理になる。

PATCH メソッドを､URL で指定したデータの一部をボディに指定された内容に変更する目的で使用する。ボディで指定するが、PUT は異なり対応するデータモデル全体ではなく、変更箇所のみを抜粋したデータを送信する。

DELETE メソッドを、URL で指定されたデータを削除又は削除処理の予約をする目的で使用する。一般的にボディを指定しない。

GET、POST、PUT、PATCH、DELETE の 5 つのメソッドのうち、GET はリソースの状態が変更されない。一方、POST は対象のリソースが新規に作成され、DB に変更が反映される。リソース ID や更新日時が自動で作成されるため、何度も実行すると、同じ情報を持つリソースが複数できることがある。PUT、PATCH、DELETE は、同じリクエストを何度実行しても、対象のリソースは結果が同じ状態となる。

(4)リクエストヘッダ

（推奨）必要に応じて標準的なリクエストヘッダを設定する。

API リクエストには、ヘッダと呼ばれる部分にリクエストに関する付加的情報（メタデータ）を設定することができる。一般に、Accept、Accept-Charaset、Content-Type などのヘッダが使用される。これらのヘッダを必要に応じて使用し、API サーバでサポートすることが望ましい。

表 5-3-3-2　Web API の標準的なリクエストヘッダ例

ヘッダ名	説明
Accept	API クライアントが処理できるレスポンスのメディアタイプを指定する。サーバや URL で指定したリソースによっては対応しない場合がある。 例：Accept: application/json
Accept-Charset	API クライアントがレスポンスで処理できる文字セットを指定する。API サーバは、このヘッダで utf-8 が指定されていても対応できるよう開発する。 例：Accept-Charset: utf-8
Content-Type	リクエストボディに設定したデータのメディアタイプを示す。ボディに設定したデータが JSON などのテキスト形式の場合、Content-Type には文字セットパラメータを含める。文字セットは標準で UTF-8 とする。 例：Content-Type: application/json; charset=utf-8

ヘッダ名	説明
If-None-Match	リクエストしたリソースが変更されている場合のみレスポンスボディで新しいデータを返却するようリクエストを発行するためのヘッダ。設定する値は、以前に取得したレスポンスの ETag ヘッダの値を指定する。 例：If-None-Match: ″5e9b1c95f986e4b418de054181bfe72a0ffa484e″

(5)リクエストボディ

（推奨）リクエストボディを設定する場合は、JSON 形式を採用する。

API を呼び出す際に元となる情報を提供する必要がある場合、リクエストボディのその内容を設定する。基本的に、API 呼び出し時に指定する内容は JSON 形式のものを取り扱うが、画像や音声、動画など JSON 形式で取り扱いにくいデータを取り扱う場合はこの限りではない。

（推奨）テキストデータの文字コードは UTF-8 を使用する。

API リクエストのボディで取り扱うテキストデータは、多数のサービスで取り扱われることが想定される。そのため、どのプログラミング言語で開発されたサービスでも同様に取り扱うことができるよう、リクエストボディにテキスト形式のデータを設定する場合は、その文字コードを UTF-8 とすることが望ましい。

5-3-4. API レスポンスの標準

(1)ステータスコード

（必須）API の実行成否に関わらず、その結果を表す適切なステータスコードを返却する。

HTTP ステータスコードは、RFC 7231 で標準的なものとその用途が定められている。API サーバは、その実行成否に関わらず、その実行結果に対応するステータスコードを返却するよう実装する。ステータスコードは 100 番台から 500 番台までの 5 つのカテゴリに分けられており、100 番台は情報レスポンス、200 番台は成功レスポンス、300 番台はリダイレクト、400 番台はクライアントエラー、500 番台はサーバエラーが検出された際に返却する。API 開発では、200 番台、400 番台、500 番台を中心に検討し、100 番台と 300 番台のレスポンスコードは必要に応じて返却を検討する。

表 5-3-4-1　成功レスポンスで利用される HTTP ステータスコードの例

番号	名称	概要
200	OK	リクエスト成功。レスポンスで返される情報はリクエストで使用されたメソッドに依存する。レスポンスボディのエラーを伝える目的では使用しない。
201	Created	リクエストによりリソースが作成されたことを示す。POST リクエストによりリソースが作成された場合には、Location ヘッダにて作成されたリソースの URI を絶対パスにて返却する。
202	Accepted	リクエストが受付けられたが、処理は完了していないことを示す。通常、POST、PUT、PATCH 又は DELETE の非同期処理 API によるレスポンスで利用される。
204	No Content	リクエストに対して送信するコンテンツはないが、リクエストとしては有効という場合に利用される。更新処理や DELETE リクエストにて利用される。
206	Partial Content	指定されたリソースの全体ではなく一部が返却されたことを示す。クエリパラメータや

番号	名称	概要
		Range リクエストヘッダに要求範囲が指定された場合に、このステータスコードを返却する。
301	Moved Permanently	リクエストされたリソースが変更されたことを示す。レスポンスにて新しい URL を返却する。
304	Not Modified	キャッシュ済みのリクエストを使い続けてよいことを通知する目的で、リクエストに対して、レスポンスが変更されていないことを通知する。

表 5-3-4-2　エラーレスポンスにて利用される HTTP ステータスコードの例

番号	名称	概要
400	Bad Request	パラメータの内容や文字コードが誤っている、必須フィールドがないなど、API クライアント側のリクエストに問題がある場合に利用する。
401	Unauthorized	認証トークンが見つからないか、又は無効な認証トークンなど、認証されていない場合に利用する。
402	Payment Required	処理を実行するためには支払いが必要な場合に利用する。
403	Forbidden	認証に成功したが、認証されたユーザが対象リソースに対するアクセス権を持たない場合に利用する。
404	Not Found	指定されたリソースが存在しない場合に利用する。URL が正しくないか、リソースが利用できないことを意味する。
405	Method Not Allowed	URL で指定したリソースに対して、リクエストメソッドがサポートされていない場合に利用する。
406	Not Acceptable	Accept リクエストヘッダに対応する返却リソースがない場合に利用する。例えば、XML 形式をリクエストしたが、JSON 形式しかサポートしていない場合に利用する。
408	Request Timeout	リソース待ちでサーバがタイムアウトした場合に利用する。
409	Conflict	作成しようとしたリソースが既に存在する場合や、更新処理を実行する際にリソースがロックされている場合などに利用する。
410	Gone	今は存在しないリソース（廃止された API）などに利用する。
412	Precondition Failed	条件に合わないリクエストがされた場合に利用する。例えば If-Match を満たさないケースなど。
414	URI Too Long	サーバがリクエスト URL を処理する際に、取り扱える上限の長さを超えた URL が指定された場合に利用する。
415	Unsupported Media Type	Content-Type ヘッダを指定せずにリクエストボディを送るなど、リクエストデータ形式に問題がないが処理できない場合に利用する。
422	Unprocessable Entity	リクエストのフォーマットは問題ないが、リクエストされた処理を実行できない場合に利用する。
428	Precondition Required	サーバが条件付きリクエストを要求する場合に利用する。
429	Too Many Requests	ユーザ、アプリケーション、又はトークン等の利用回数制限をオーバーしたリクエストに対して応答する場合に利用する。
500	Internal Server Error	サーバ側での処理中に予期しないエラーが発生した場合に利用する。サーバ側のソフトウェアの欠陥又はサイトの停止を示す。
501	Not Implemented	サーバがリクエストを実行できない場合に利用する。また、暗に将来実装予定であ

番号	名称	概要
		ることを示す。
503	Service unavailable	サービス利用できない場合に返却する。サーバがメンテナンス中、外部サービスを利用していてそのサービスがエラーの場合など。

（必須）GET リクエストが成功した場合、通常は 200 OK を返す。

（必須）GET リクエストで 0 個のリソースを取得した場合は 200 OK を返す。

（必須）GET リクエストでリソースの取得ができなかった場合は 404 Not Found を返す。

リソース取得を目的とする GET メソッドでは、リクエスト成功時に 200 OK を返却する。なお、複数のリソースを取得する場合（条件に合致するリソースを取得する検索機能など）で返却されるリソース数が 0 の時、404 Not Found ではなく 200 OK を返却する。単一のリソースを ID などで指定して個別に取得するリクエストの場合は、そのリソースが見つからない場合に 404 Not Found を返却する。

（必須）POST リクエストが新しいリソースの作成に成功した場合、201 Created を返す。

POST リクエストで新しいリソースが作成された場合、200 OK ではなく 201 Created を返却する。なおこの際、Location レスポンスヘッダには作成されたリソースの URL を併せて記載し、作成されたリソース情報をレスポンスボディに設定する。

（必須）PUT リクエストが既存リソースの更新に成功した場合、200 OK 又は 204 No Content を返す。

PUT リクエストでは、リソース全体をボディに含めてリクエストを発行するため、API リクエストの発行元は対象リソースの全体を保持している。そのため、レスポンスボディを設定しない 204 No Content を返却しても、通常は API クライアント側の動作に影響はないため、200 OK でボディを含めて返却しても、204 No Content でボディを含めず返却してもよい。

（必須）PATCH リクエストが既存リソースの更新に成功した場合、200 OK 又は 204 No Content を返す。

PATCH リクエストでは、対象リソースの変更したい部分のみをボディに含めてリクエストを発行するため、通常 API リクエストの発行元は差分を意識できるよう対象リソースの全体を保持している。そのため、レスポンスボディを設定しない 204 No Content を返却しても、API クライアント側の動作に影響はないため、200 OK でボディを含めて返却しても、204 No Content でボディを含めず返却してもよい。

（必須）DELETE リクエストが対象リソースの削除に成功した場合、204 No Content を返す。

（必須）DELETE リクエストの対象リソースが削除済みの場合、204 No Content を返す。

DELETE リクエストは、対象のリソースが削除済みの状態にすることを目的としている。そのため、API リクエストでリソースを削除した場合も、リクエスト時点で対象リソースが削除されている場合も、同様にリクエストを満足しているとして 204 No Content を返却する。

(2) レスポンスヘッダ

（推奨）必要に応じて標準的なレスポンスヘッダを設定する。

API レスポンスにも、リクエスト時と同様に、レスポンスに関する付加的情報（メタデータ）を設定することができる。Content-Language や Content-Type などの一般的なヘッダに加え、X-Request-Id や X-RateLimit-Limit など API 提供時によく使用されるヘッダ、及びキャッシュ利用可否に関する情報を保持する ETag や Last-Modified などのヘッダが指定される。これらのヘッダを必要に応じて使用し、API サ

ーバでサポートすることが望ましい。

表 5-3-4-3　Web API の標準的なレスポンスヘッダ例

HTTP ヘッダ名	説明
Access-Control-Allow-Origin	サービスの設計上、Web ブラウザから直接 API を呼び出す場合は、オリジン間リソース共有(CORS)の設定を有効化する必要がある。必要に応じて、Access-Control-Allow-Origin ヘッダを追加して対応する。 例：Access-Control-Allow-Origin:https://api.example-city.jp
Content-Language	コンテンツの言語を指定するために使用する。API クライアントは、Content-Language ヘッダを使用してデータの言語を識別する。 例：Content-Language: en-US
Content-Type	レスポンスボディのメディアタイプを示す。レスポンスボディが含まれている場合、API サーバは Content-Type をレスポンスに含める。コンテンツが JSON などのテキストベースのタイプの場合、Content-Type には文字セットパラメータを含める。文字セットは UTF-8 とする。 例：Content-Type: application/json; charset=utf-8
ETag	指定されたリソースのハッシュ値など、リソースが変化しているかどうかを確認するための値。API クライアントはこの情報と合わせてレスポンスボディをキャッシュに保持し、次回リクエスト時に If-None-Match ヘッダに ETag 値を設定することで、キャッシュの再利用可否を確認できる。 例：ETag: "5e9b1c95f986e4b418de054181bfe72a0ffa484e"
Last-Modified	指定されたリソースをブラウザや中継サーバ等がキャッシュできる期限。指定された期限よりも前なら、保存したキャッシュを再利用してリクエストを省略できる。 Last-Modified: Wed, 24 Feb 2021 05:00:00 GMT
Location	POST メソッドで新しいリソースが作成され、ステータスコード 201 Created が返却された際に指定する。内容には新しく作成されたリソースの URL を設定する。 例：Location: /v1/SocialEvents/ABCD
X-Request-Id	クライアントからサーバ及びそのバックエンドへの Web サービス（REST API など）への個々の要求をトレースするためリクエスト ID。重複しないよう、UUID を生成して設定する。
X-RateLimit-Limit	特定期間において、クライアントに許可されるリクエストの最大数。
X-RateLimit-Remaining	特定期間において、残りのリクエスト可能な数を返す。
X-RateLimit-Reset	特定期間において、レート制限がリセットされる時間までの相対時間を秒単位で指定する。

(3)レスポンスボディ

（推奨）レスポンスボディを返却する場合は、JSON 形式を採用する。

API 実行結果としてレスポンスボディの設定が必要となる場合、JSON 形式のボディを取り扱う。ただし、画像や音声、動画など JSON 形式で取り扱いにくいデータを取り扱う場合はこの限りではない。

（推奨）テキストデータの文字コードには UTF-8 を使用する。

（必須）API のレスポンスボディに外字を使用しない。

（必須）日本語（全角文字）については『JIS X 0213』に定義される JIS 第 1 水準～JIS 第 4 水準を利用する。

（推奨）API に返却するデータについて、必要に応じて文字コードの変換、文字の縮退を行う。

API が返却するリクエストボディは、多数のサービスで取り扱われることが想定される。そのため、どのプログラミング言語でサービスが開発されていても同様に取り扱うことができるよう、リクエストボディのテキストデータは文字コードを UTF-8 とし、必要に応じて変換することが望ましい。また、文字化けや文字欠けに留意し、文字列の縮退を行う場合もある。文字列の縮退については、一般社団法人文字情報技術促進協議会が管理する『MJ 縮退マップ』を利用し、処理を行う。

（必須）レスポンスデータに複数行テキストが含まれる場合、改行コードには LF を使用する。

コンピュータにおける改行コードは LF、CR、CR+LF の 3 種類があるが、ネットワークを転送するプロトコル標準を踏まえて問題なくデータを転送できるよう、改行コードには LF を使用する。

（必須）JSON で 0 個のリソースを返却する場合、その属性値を NULL ではなく空配列（[]）とする。

複数のデータが返却される可能性のある GET リクエスト（検索など）の結果、条件に合致するリソースの数が 0 個だった場合、エラーと区別するために NULL ではなく空配列を返却する。

（必須）JSON の属性値にテキストを保持する場合、タブや改行などの特殊文字をエスケープする。

JSON の属性値の中にはテキストデータを保持することができるが、タブ、改行コードなどの特殊な文字を使用する場合はエスケープ処理を行うこととし、API クライアントが問題なく利用できるよう配慮する。

（推奨）処理ステータス、処理ログ、ドキュメント、パラメータ、データ件数、実行結果データ、画面遷移などのメタデータを API 実行結果に含める。

API 利用者はレスポンスデータを見て開発を行うため、返却される内容で実行結果が判断できるようにすることが望ましい。以下にメタデータの例を示す。

レスポンスデータに含めるメタデータ例

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Language: ja
{
    "metadata": {
        "title" : "書籍情報取得",
        "detail" : "正常実行されました。",
```

```
        "type" : "http://developer.example-city.jp/docs/",
        "parameter" : {
            "offset": 25,
            "limit": 25
        },
        "resultset": {
            "count": 227,
            "offset": 25,
            "limit": 25        }
    },
    "results": []
}
```

表 5-3-4-4　レスポンスに含めるメタデータ例

属性名	用途
metadata	API 実行結果について概要を格納する。
title	API の名称を表示する。
detail	API 利用者の処理結果を格納する。
type	API のドキュメントを示す URI を格納する。該当する URI がない時には、”about:blank”と入力する。
parameter	API が受け取ったパラメータを配列として格納する。パラメータを受け取っていない場合には、空の配列を格納する。
resultset	全データ件数、返却しなかった件数、返却したデータ件数 など、結果セットについてのメタ情報を表示する。
count	返却可能なレコード総数について格納する。
offset	返却したデータの中で、先頭から「offset」で書かれた件数のデータを返却しなかったことを記載する。
limit	返却されたデータ件数を記載する。
results	結果データを表示する。

（推奨）エラーレスポンス時として、エラー内容を示すレスポンスボディを返却する。

API 利用時に正常にデータのやり取りがなされなかった場合、エラー内容を示すメッセージを返すことで、不具合時の原因調査や解消作業に役立つ。RFC 7807「Problem Details for HTTP APIs」では、次の内容を含むレスポンスを返却することが推奨されている。

表 5-3-4-5　エラーレスポンスに含むデータ項目

名称	概要
status	処理に対応した HTTP ステータスコードを HTTP レスポンスヘッダに格納する。
type	エラーの種別を示す URI を格納する。参照先の URI では、API 利用者がエラー種別を理解できるドキュメントを用意していることが望ましい。該当する URI がない時には、”about:blank”と入力する。

名称	概要
title	エラーの名称を表示する。
detail	API 利用者がどこに問題があるか理解できるよう、エラーの説明文を格納する。
instance	エラーが発生した URI を記載する。

例えば、振込元の銀行口座に 3,000 円の残高がある時、別の口座に 8,000 円を振り込む API リクエストを発行した際に返却されるレスポンスは、次のようなフォーマットで記載することができる。

エラーレスポンス例

```
HTTP/1.1 400 Bad Request
Content-Type: application/problem+json
Content-Language: ja
{
"type": "https://example-city.jp/probs/out_of_credit",
"title": "口座残高が十分でありません。",
"detail": "口座残高は 3000 円です。5000 円の支払いが必要です。",
"instance": "/account/12345/msgs/abc",
"balance": 3000,
"accounts": [
  "/account/12345",
  "/account/67890"
]
}
```

※上記例では、「表 5-3-4-5」で例示したデータ項目に加え balance、accounts を返却している。
出所：RFC 7807「Problem Details for HTTP APIs」を参考に作成

5-3-5. 認証方式の標準

（推奨）API のクライアント認証では API Key を使用した認証方法を選択する。

API の認証には、ユーザが使用するサービスやアプリケーションの単位で API アクセスを認証するクライアント認証と、ユーザ単位で認証を行うユーザ認証の 2 パターンがある。API を利用するユーザ単位での細かな認証や認可を必要としない API の場合、クライアント認証を使用した、アプリケーション又はサービス単位で認証・認可の仕組みを利用できる。クライアント認証では、API Key を HTTP リクエストヘッダに設定する方法や、API クライアントごとに払い出された API Key と API Secret を組み合わせ、リクエストに署名を追加する方法がある。具体的な認証方法は API 個別の要件に従い検討する。

（推奨）API のユーザ認証では OpenID Connect を使用する。

API が個人に紐づく情報（ユーザ名、氏名、住所など）を取り扱う場合、API はユーザごとに異なる情報を返却する必要がある。このようなユーザを見分けた動作が必要なユースケースでは、OpenID Connect (OIDC)を利用した認証を使用する。OIDC にはスコープと呼ばれるアクセス範囲を設定する考え方が用意されており、ユーザごとに定めたスコープよって実行可能な API を制限できる。また OIDC では、API とは別に用意された認証・認可を行うサービス(Identity Provider: IdP)[33]と連携できるため、ユーザの認

[33] https://openid.net/certification/

証・認可に関する情報を一元管理できる。なお、OIDC の詳細な認証フローについては標準が定められており、それに従う。

（必須）個人情報など個人同意が必要となる情報の流通を行う場合はオプトイン管理を実装する。

個人情報を扱う API を提供する場合、ユーザの同意を得るオプトイン管理の実装が必須である。サービスに対して、オプトイン未済のユーザの個人情報を API で提供してはならない。OpenID Connect では、ユーザの同意を得るためのダイアログを提示するフローを規定しており[34]、これらを参考にオプトイン/アウトを管理する仕組みの実装を行う必要がある。ユーザのオプトイン/アウトはデータ連携基盤と接続するサービスごとに取得するものとし、ユーザのオプトインしたサービスにのみ API による情報提供を行う。これによりサービス提供者やデータ連携基盤事業者により事前承諾なくユーザの個人情報が利用されることを防ぐ必要がある。

5-3-6. システム構成

（推奨）API ゲートウェイを利用し、API の共通機能を取りまとめて提供する。

API を公開するには、認証・認可、レート制限、サイズ制限、ログ取得、不正アクセス対策などの API 一般に必要となる機能を実装し、適切に管理することが求められる。これらの機能を個別の API にプログラミングすると、採用するライブラリの違いなどの理由で細かな仕様にばらつきが出る可能性があったり機能の追加や削除、設定の変更時にプログラムの再リリースが必要になる。そのため、API ゲートウェイを採用し、このような共通機能をまとめて管理し、API 実装コストを削減し、安定した品質で API を提供することが望ましい。

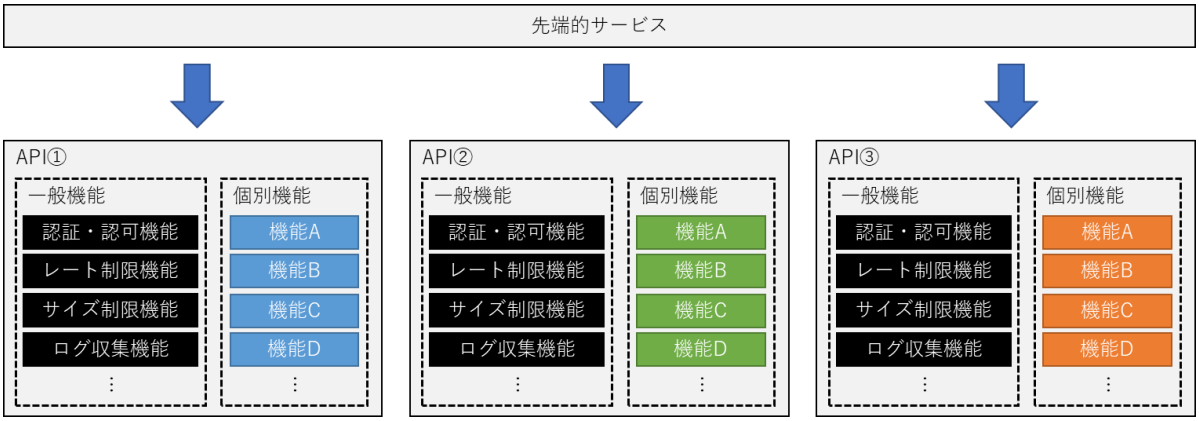


図 5-3-6-1 API Gateway のないシステム構成例

[34] https://openid.net/specs/openid-connect-basic-1_0.html#Consent

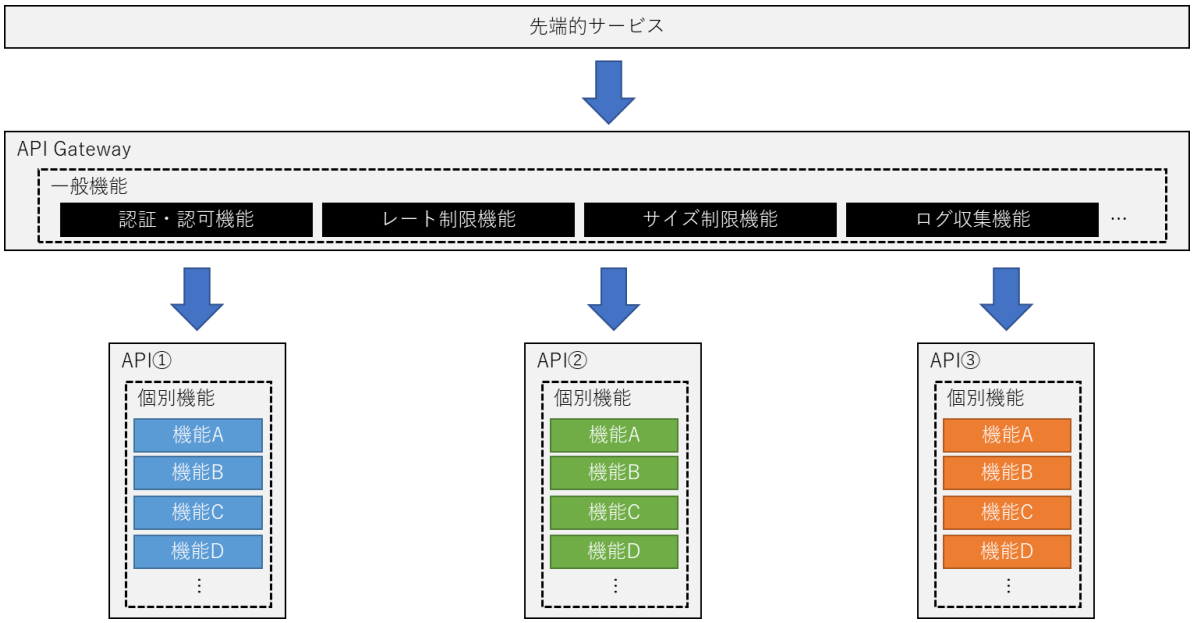


図 5-3-6-2　API Gateway を導入したシステム構成例

5-3-7.　API 標準仕様の今後について

本節で定めた API 標準仕様案は OpenAPI 仕様 3 系をベースとしているが、近年では下表に挙げるような通信規格/仕様も用途に合わせて採用されている。今後は、これらの通信規格/仕様の観点から本 API 標準仕様案の拡張を検討し、様々な先端的サービスの可能性を広げていくことが考えられる。また、OpenAPI 仕様も継続的にバージョンアップされているため、本 API 標準仕様は定期的に更新していくことが望まれる。

表 5-3-7-1　今後 API 標準仕様で検討すべき通信規格/仕様

通信規格/仕様	概要
gRPC[35]	gRPC は、Google 社が開発を開始したリモートプロシージャコールシステムで、独自のインターフェース記述言語を利用して様々なプログラミング言語に対応したリモートプロシージャコールの組み合わせを実現する。HTTP/2 プロトコルを利用しているためストリーム型の通信をサポートし、マイクロサービスアーキテクチャにおける接続サービスや、多人数参加型のチャットサービスにおけるインターフェースに利用することができる。
GraphQL[36]	GraphQL は、Facebook 社が中心となって開発された API 向けのクエリ言語及びその実装である。クライアントアプリケーションが呼び出すエンドポイントが単一でありながら、様々なクエリを投げることでバックエンド API から必要なデータ項目を取り出して必要なデータの形式に変換してフロントに返すことができる仕組みを提供する。クライアントアプリケーションが自らの要望に合わせて得られるデータ項目を決定できるというメリットがある。
Webhook	コールバック URL を設定することで Web ページ又は Web アプリケーションの動作を拡張又は変更する方法のこと。特に標準などはないが 2007 年に Jeff

[35] https://grpc.io/
[36] https://graphql.org/

通信規格/仕様	概要
	Lindsay によって言及されたプログラミング用語[37]である。
WebRTC[38]	WebRTC は、Web Real-Time Communication の略で、クライアントアプリケーションにシンプルな API 経由でリアルタイム通信を提供するオープンソースのプロジェクトである。ブラウザ間で直接ピア・ツー・ピア通信を行い、プラグインやネイティブアプリケーションのインストールなしで、ボイスチャット、ビデオチャット、ファイル共有などを実装できる。
AsyncAPI[39]	AayncAPI は、WebSocket や MQTT といった非同期プロトコルをサポートする API 仕様である。AsyncAPI Initiative によって仕様案策定が進められている。OpenAPI 仕様を拡張した形で仕様案策定が進められており、OpenAPI Initiative との交流も活発になされている。

[37] https://web.archive.org/web/20180630220036/http://progrium.com/blog/2007/05/03/web-hooks-to-revolutionize-the-web/

[38] https://webrtc.org/

[39] https://www.asyncapi.com/

6. 調査結果（イ）②API カタログ/開発者ポータル

6-1. 概要・目的

本報告書では、データ連携基盤が具備すべき機能・考慮すべき事項についての調査として、データ仲介機能や構築されるべき API の共通ルール・標準仕様、及びデータモデルの在り方といった、データ連携基盤自体の機能や内容ついての調査結果が記載されている。

それらの調査結果を基にデータ連携基盤が実際に構築され利活用される際、実装されている機能の利用方法や各種基準ドキュメント、API の利用方法、活用事例等を公開するための場が必要となる。

API の情報公開や開発者支援するための情報提供を行うポータルサイトの必要性については、既に他の事業の中でも検討がなされている。例えば、API テクニカルガイドブック[40]では、「API の利用を促進するためには、提供している API が広く開発者等に認知され、開発を支援するドキュメントやテスト環境等が用意されていることが重要」といった記載が存在し、公開すべき項目や告知方法等が検討されている。

また、スーパーシティ構想[41]では「スーパーシティでは各取組で実装される API に関して、情報の見つけやすさを向上させ、公開されている様々な API への接続をより容易とするために、API に関するメタデータやデベロッパーサイトの情報をまとめたカタログサイトの実装を行います。」という記載がなされており、各エリアで実装される API についてはカタログサイト上で集約・公開するものであると定義されている。さらに、各取り組みで実装される API の収集・公開だけでなく、「自団体のウェブサイト、あるいは信頼性の高いソースコードリポジトリ等、利用する技術者にわかりやすい場所においてデベロッパーサイト（開発者サイト）を作成し、そこで API に関する情報を、技術者にわかりやすい場所と形式で公開しなくてはなりません。」との記載から、各エリアにおけるデベロッパーサイト構築の必要性と、そこで提供される API に関するカタログサイトとの連携・住み分けの必要性が示唆されている。

さらに、各スーパーシティの取り組みにおける API を集約・公開するカタログサイトと、各エリアで構築されるデベロッパーサイトの連携、及びエリア同士の情報連携を実現するため、各エリアのデベロッパーサイトの構築においては連携を前提とした規格や品質の均一化が図られる必要がある。これについては、データ連携基盤活用のために最低限具備すべき機能を、中央から各エリアに対して提言することが望ましい。

上記を踏まえ、本章では、エリア横断で API の共通ルール・標準仕様等について中央から情報を公開するカタログサイトを「API カタログ」、そこからリンクされ、各エリア独自の API 詳細情報を公開したり、先端的サービスの開発に寄与する各種機能を各エリアの開発者に対して提供する場を「開発者ポータル」と定義する。

---

[40] API テクニカルガイドブック
https://cio.go.jp/sites/default/files/uploads/documents/1020_api_tecnical_guidebook.pdf

[41] 「スーパーシティ」構想について
https://www.kantei.go.jp/jp/singi/tiiki/kokusentoc/supercity/supercity.pdf

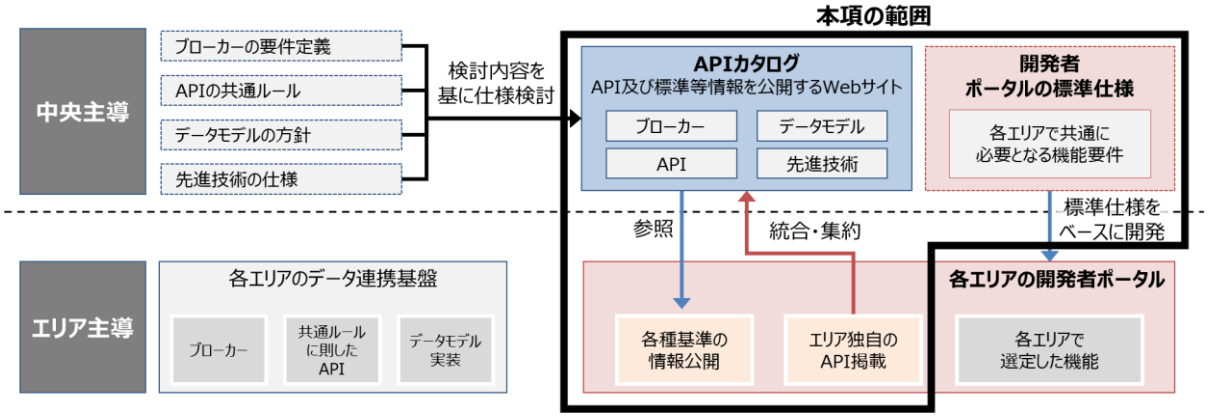


図 6-1-1　API カタログと開発者ポータルの本項における検討範囲（再掲）

本章では、スーパーシティ推進において必要となる API カタログ・開発者ポータルが具備すべき機能を調査するとともに、中央とエリアの連携実現の観点から、中央が担うべき役割と各エリアで提供されるべき役割を整理することで API カタログと開発者ポータルのあるべき姿を詳細化する。調査にあたっては既存事例を参考にするとともに、既存事例の開発者・運用者に対するヒアリングを実施する。

本章に記載の報告内容は、スーパーシティの取り組み主体が、各エリアに対して準拠させるべき基準や共通に実装させるべき機能といった内容を検討する際のインプットのひとつとして活用されることを想定している。また各エリアにおいては、本章を参照し、構築すべき開発者ポータルと全体における連携イメージ、及び必要となる準備事項等をより具体的にイメージしてもらうとともに、必ず実装されるべき機能とエリアの特色に合わせて独自で検討の余地がある機能とを検討する際の指針のひとつとなることを目的とする。

なお本章は、中央が整備すべき機能と各エリアで整備されるべき機能の住み分けに関する提言を主たる目的とし、詳細な実装方式や利用技術、及びそれらを基に検討されるべき非機能要件については検討に留めるものとする。また、各エリアで実装されるべき機能に関しても、各エリアの地域色の差を鑑み、必ずしも全国共通で必要ではないと想定される機能については、機能の概要レベルの提言に留めるものとし、詳細は各エリアが個別で実装検討すべきものとして位置付けている。

6-2. 仕様検討

6-2-1. 基礎調査によるモデルケースの選定

(1)調査の考え方

API カタログ・開発者ポータルに必要な機能は、政府 CIO ポータルにて標準ガイドライン群として公開されている API 導入実践ガイドブック[42]、API テクニカルガイドブックにて既に一部検討・定義されている。上記ドキュメントにおいては、ドキュメント公開・サンプルプログラム提供・開発者コミュニティといった機能の必要性やその効果については触れられているものの汎用的な記載となっており、スーパーシティにおける API カタログ・開発者ポータルに必要な機能としては詳細化の余地がある。

本項ではそれらの要件を詳細化するために、まずは基礎調査として、本事業における API カタログ・開発者ポータルに類する既存事例について調査を実施し、特徴的・典型的な機能や、それに対応した課題等の事例を収集した。調査対象は、API 利活用の先進的な事例を調査するために海外・金融・民間企業の事例を選定したほか、官公庁や自治体等が API やオープンデータといった情報を集約・提供している事例を対象とした。

(2)調査結果

既存事例の調査として、官公庁事例 4 件、自治体事例 3 件、民間事例 11 件、金融事例 3 件、海外事例 2 件の計 23 件を全体数として調査を実施した。

それらの事例から共通的に実装されている機能を洗い出し、API 導入実践ガイドブック、API テクニカルガイドブックといった既存検討における考え方を参考に、本事業において特に重要と想定される機能・特徴に対して適合性の高い事例をモデルケースとして選定した。選定したモデルケースとそれぞれの特徴を表 6-2-1-1 に示す。

表 6-2-1-1 モデルケースの比較

No.	分類	名称	特徴
1	自治体	Aizu WAKAMATSU Developer's Site(AWDS)	• ソーシャルログインが可能。 (Facebook,Twitter,Google,LINE,Yahoo,ゆうびん ID) • API を活用した事例を紹介するページがある。
2	自治体	BODIK オープンデータカタログサイト (BODIK ODCS)	• 複数自治体が参画し、オープンデータを公開できるカタログサイト。 • 複数自治体が関連した事例として以下が参考にできると想定。 1.新規に参画した自治体の表示方法 2.各参加自治体へのリンク方法（地図上から自治体を選択）
3	金融	三菱 UFJ 開発者ポータル	• API カタログ機能が API テクニカルガイドブックとの合致度が高い。 • 本ポータルの活用事例に加えて、Fintech や API 利活用、銀行 API アイデアソンレポート等幅広い技術情報が TIPS として掲載されており、外部からの新規利用者取り込みのパスとして機能することが期待できる。

[42] API 導入実践ガイドブック
https://cio.go.jp/sites/default/files/uploads/documents/1019_api_guidebook.pdf

No.	分類	名称	特徴
4	民間	楽天ウェブデータ	• API デモツールがプルダウンによる項目選択で直感的に操作できる点や、各種プルダウンを選択した際にガイドが表示される機能等、デモツール自体が仕様書として機能しており、仕様書なしで操作しやすい工夫がなされている。
5	民間	LINE Developers	• 開発者コミュニティの事例。
6	官公庁	Data.go.jp	• 政府事例であり、中央からの情報発信を行う場面におけるモデルケースになり得る。
7	民間・官公庁	農業データ連携基盤(WAGRI)	• 複数企業の API を収集し公開を行っている点が類似。 • API を機能、得られるデータの特徴、有償/無償等の分類等から検索することが可能。

これらの類似事例を基に本事業で必要と想定される機能を抽出した結果を表 6-2-1-2 に示す。なお、モデルケースの機能として特に特徴的であった内容を事例列に記載する。

表 6-2-1-2　類似事例において一般的に実装されている機能（モデルケース整理結果）

No.	機能		説明	事例
1	全画面共通	メニュー表示	サイトのトップページ、メニューボタンを表示。	―
2	サイトトップ	更新情報 お知らせ	サイトの更新情報を表示する。	BODIK ODCS： 当該サイトへの新規自治体の参加情報が表示される。 楽天ウェブデータ： カテゴリタグ（リリース、廃止等）が表示される。メンテナンス情報と併せて表示される。 三菱 UFJ 開発者ポータル： API 公開やブログ記事（TIPS 等）の更新情報が表示される。メンテナンス・障害情報はタブ分けされている。
3		障害・メンテナンス情報 表示	ポータルサイトの障害・メンテナンス情報を表示する。	楽天ウェブデータ： カテゴリタグ（メンテナンス等）が表示される。更新情報も併せて表示される。 三菱 UFJ 開発者ポータル： メンテナンス・障害情報が表示される。更新情報はタブ分けされている。
4	API 公開・提供機能	API 一覧	API の一覧を表示する。 API の公開開始日、公開終了日、ステータス（公開予定、公開中、公開終了）等を表示する。	三菱 UFJ 開発者ポータル： API 一覧、API 概要・詳細仕様・サンプルを 1 つの画面で確認可能。

No.	機能		説明	事例
5		API 詳細	各 API の仕様を公開する。 API のパラメータ、パラメータの説明、サンプル等の情報やメタデータ（カタログデータ、組織/人等）を公開する。	
6		利用規約公開	ライセンス、禁止事項、免責事項等を公開する。	―
7		利用手順公開	API を利用するための手順。 ユーザ登録、実行方法等	―
8	API 評価機能	API 開発評価環境提供（簡易検証環境）	開発評価向けに、API 動作仕様の簡易検証環境を提供する。	―
9		API 統計情報	API の利用状況を表示する。	―
10	オープンデータ公開・提供機能	オープンデータ一覧	オープンデータの一覧を表示する。 オープンデータの公開開始日、拡張子等を表示する。	BODIK ODCS・Data.go.jp： CKAN を活用して作成されている。
11		オープンデータ詳細	各オープンデータのメタデータ（カタログデータ、組織/人等）を公開する。	
12		利用規約	ライセンス、禁止事項、免責事項等を公開する。	―
13	情報交換機能	開発者コミュニティ	開発者間のやり取りを実現する。 開発者間で開発上の課題を共有・解決することで、問合せ対応の負荷の削減、改善点の発見やナレッジ蓄積を行う。	LINE： コミュニティ上でユーザ間で質問交換をすることで、問題解決を行っている。
14		FAQ	開発者コミュニティや問合せでよく受ける質問・回答例を整理し、提供する。	―
15	その他	ドキュメント公開機能	ブローカーや開発者支援サイト等、基盤の利活用に有益な機能のノウハウをまとめ開発者に提供する。	―
16		ツールキット提供	基盤利活用に有益なツールキットを提供する。	楽天： GitHub を活用して公開されている。
17		活用事例	公開 API が実際に活用されている事例を公開する。	AWDS： API の活用情報が WEB 形式で紹介されている。※PDF 等別ファイルの参照なし。
18		エリア状況確認	参画している団体の情報や一覧を表示する。	BODIK ODCS： 地図画像をクリックすることで、視覚的に

No.	機能		説明	事例
				わかりやすく団体一覧表示/掲載情報に移動できる。
19		関連サイトリンク集	関連するサイトへのリンク集。	―

6-2-2. 基礎調査に基づく仕様検討

(1)API カタログ/開発者ポータルのあるべき姿の整理

ここまで、既存の API 公開サイトや開発者支援サイト、オープンデータカタログの事例を基に、API の利活用に際して共通に実装されるべき機能を洗い出した。本事業における API カタログ/開発者ポータルの機能や仕様をより詳細に検討するにあたっては、それらの共通機能について中央が整備すべきものとエリアが整備すべきものに分類する必要がある。

機能の分類に先立ち、本項では、内閣府における既存の検討内容や各種ガイドブックに記載の指針等を基に、以下 5 つの観点から API カタログ/開発者ポータルの目的や役割を詳細化することで、あるべき姿を整理した。

- カタログサイトの役割は、スーパーシティで実装される API に関して、情報の見つけやすさを向上させ、公開されている様々な API への接続をより容易とすることである。API に関するメタデータやデベロッパーサイトの情報をまとめた実装を行う。
  （スーパーシティ構想[43]）
- デベロッパーサイト（開発者サイト）は、自団体のウェブサイト等、利用する技術者にわかりやすい場所において作成し、API に関する情報をわかりやすい形式で公開しなくてはならない。
  （スーパーシティ構想）
- API の利用を促進するためには、提供している API が広く利用者等に認知され、開発を支援するドキュメントやテスト環境等が用意されていることが重要。
  （API テクニカルガイドブック）
- API の再発明を避け、可能な限り機能の再利用を行うためには類似機能を提供している既存の他地域や他団体を検索できることによる相互運用性の向上が必要。
  （5-1 API 設計/開発・公開・運用プロセス）
- 国とエリアで設けるシステム間における相互運用性の観点から、各エリアにおいてスーパーシティサービスが効率的に開発されるよう、国から提供されるべき情報と、他エリアとの情報連携のために各エリアが準備すべき情報の検討が必要。
  （相互運用性の観点）

中央で整備される API カタログと、各エリアでデータ連携基盤ごとに構築される開発者ポータルの役割分担のイメージを表 6-2-2-1 及び図 6-2-2-1 に示す。

[43] 「スーパーシティ」構想について
https://www.kantei.go.jp/jp/singi/tiiki/kokusentoc/supercity/supercity.pdf

表 6-2-2-1　API カタログ/開発者ポータルの役割

対象	主体	目的
API カタログ	国	• 先端的サービスの開発の際に、各エリアのデータ連携基盤で提供される API やサービスについての情報が得られること。 • エリアがデータ連携基盤を構築・運用する際に、他エリアの事例や情報を入手できること。 • 各エリアにおけるデータ連携基盤構築・スーパーシティ実現促進のために、各エリアがデータ連携基盤及び開発者ポータルの構築に必要なドキュメント・ツール、API 標準仕様等の情報を公開すること。
開発者ポータル	エリア	• データ連携基盤の利用者が、データ連携基盤を用いた先端的サービスを円滑に開発できること。 • データ連携基盤で実装されている API の詳細情報や利用規約・手順等のドキュメント公開、開発者間で問題解決を図るためのコミュニティ機能等、基盤を有効に利活用できること。

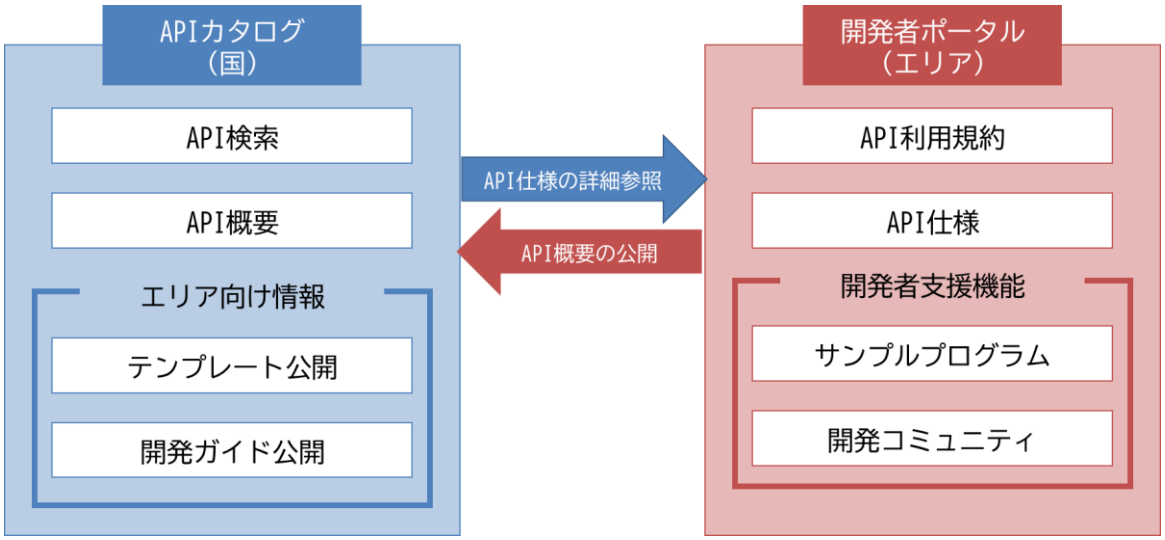


図 6-2-2-1　API カタログと開発者ポータルのイメージ

API の情報公開に関しては、国は API の概要情報のみを公開し、詳細情報は各エリアで公開・管理される想定とした。重複管理を防ぐ観点からは、国とエリアのどちらか一方でのみ API 情報を公開する方式も考えられるが、地域だけで公開すると、エリアに閉じた状況となり、他地域の情報が参照できずスーパーシティ構想の重要な要素である相互運用性が確保されない。また国だけで公開した場合、API の仕様や内容を国側が全て精査する必要があり、公開・管理の速度が落ちる等の利便性低下が危惧されるとともに、スーパーシティ構想として地域の自主性を重んじる方針にも反する。よって、重複管理を防ぎつつ相互運用性を担保する方式として、詳細情報は各エリアが取りまとめ、概要情報のみ API カタログで公開する方式をあるべき姿として整理した。API カタログにおけるエリア横断での概要情報公開にあたっては、適切な検索機能を設け、利用者がエリア横断・分野横断で必要とする概要情報にアクセスできる設計とされるべきである。

また、国が提供する API カタログはただ単に API 概要情報を提供するだけでなく、全国から参照されるポータルとなる立ち位置を鑑み、各エリアで構築されるデータ連携基盤や開発者ポータルに関するテ

ンプレートや開発ガイド、及び本調査事業において検討されている各種基準に関する情報も併せて提供する役割が想定される。基準やガイドを本サイトを通じて提供することにより、各エリアで構築される各種システムの品質を均一化するとともに、テンプレート公開によるエリアの開発工数削減を通じ、より多くの団体のスーパーシティ構想参画に寄与できる。また、基盤開発の初期段階から各エリアで API カタログを利用してもらうことで、各エリアにおける API カタログの認知度を向上させ、データ連携基盤や開発者ポータル稼働後にも API カタログを継続して利活用してもらうための導線にもなると想定する。

開発者を支援するためのサンプルプログラムや開発者コミュニティ等の機能については、内閣府スーパーシティ構想では「自団体のウェブサイト等、利用する技術者にわかりやすい場所において作成される」と記載されていることに加え、それらの機能の主たる利用者はスーパーシティ推進を目的とするサービス開発者であり、規模や分野、取り組むべき地域課題といった地域色に合わせた開発者向け支援がなされることが望ましいことから、各エリア単位で公開されるべき情報としてあるべき姿を整理した。

(2)想定利用シーンの検討

上記で検討したあるべき姿を基にして、本事業独自で必要となる機能が存在するか検討するために、API カタログと開発者ポータルの想定アクター、API カタログと開発者ポータルの開設準備から運用までの流れ、取り扱われる情報、及び想定利用シーンを検討した。

(a)想定アクターの検討

利用主体ごとの特色を考慮したニーズを仕様案へ反映し、機能充足性を確保するために、既存事例に基づく基礎調査に加えて利用シーンの検討を行った。まずは、API カタログ及び開発者ポータル各々の開設準備や運用、データ連携基盤の構築や運用、サービスや API の提供等、各役割を実行するアクターを事前に整理することで、利用シーン及びそれに関連する機能を利用者ニーズの観点から再度整理した。

API カタログの想定アクターを表 6-2-2-2、開発者ポータルの想定アクターを表 6-2-2-3 に示す。

表 6-2-2-2 API カタログの想定アクター

No.	区分	アクター	概要	アクター例
1	運営者	API カタログ管理者	• API カタログサイトの管理・運営を行う • API カタログで開示する API の概要、各種ガイドラインのメンテナンス、各エリアの状況確認等を行うことが想定される	国、支援事業者
2		API カタログ開設事業者	• 国によって選定される、API カタログの開設を支援する事業者。	民間事業者
3	利用者	自治体	• 各エリアのデータ連携基盤・開発者ポータルの構築と運営に関する業者選定 • 各エリアで稼働するサービスや API の管理 • API カタログ等を通じて各種基準やドキュメントを取得し、関連する事業者への情報展開を行うことが想定される。 • 必要に応じて、国に対してデータ連携基盤整備に向けた構築状況登録等の運用も想定される。	都道府県・市区町村及びそれらの共同体
4		データ連携基盤	• 各エリアにて選定される、データ連携基盤を構築す	都道府県・市区町村

No.	区分	アクター	概要	アクター例
		整備事業者	る民間事業者。 • 必要に応じて、国に対してデータ連携基盤整備状況の反映を行う。 • API カタログ等を通じて構築に関する各種基準やドキュメントを取得する。	に選定された民間企業
5		先端的サービス事業者	• API カタログに集約されたカタログ情報を基に各エリアの開発者ポータルから API を参照・活用し、スーパーシティ構想に寄与する先端的サービスを創出する。	サービサー （民間企業・個人）
6		開発者ポータル構築事業者	• 各エリアで構築されたデータ連携基盤を利活用するための開発者ポータルを構築する事業者。 • API カタログ等を通じて開発者ポータル構築に関する各種基準やドキュメントを取得する。	民間企業

表 6-2-2-3　開発者ポータルの想定アクター

No.	区分	アクター	概要	アクター例
1	運営者	開発者ポータル管理者	• 開発者ポータルの管理・運営を行う。 • 開発者ポータルで公開される、各エリアにおける API の詳細な情報や、コミュニティ・FAQ 等の各種情報を管理する。	都道府県・市区町村 及びそれらに選定された民間企業
2		データ連携基盤整備事業者	• 各エリアにて選定される、データ連携基盤を構築する民間事業者。 必要に応じて、国に対してデータ連携基盤整備状況の反映を行う。 • 他エリアの開発者ポータルを参考に、構築すべき API の詳細情報を既存事例として入手する運用が想定される。	都道府県・市区町村 及びそれらに選定された民間企業
3		開発者ポータル構築事業者	• 各エリアで選定される、開発者ポータルを開設する事業者。 • データ連携基盤整備事業者が開発者ポータル構築事業者を兼ねることも可能と想定する。	民間企業
4	利用者	自治体	• 各エリアのデータ連携基盤開発者ポータルの構築と運営に関する業者選定や提供されるサービスを管理する。	都道府県・市区町村 及びそれらの共同体
5		先端的サービス事業者	• 開発者ポータルを通じて各エリアのデータ連携基盤を利活用するための API を参照活用し、スーパーシティ構想に寄与する先端的サービスを創出する。 • 開発した API に関する情報を開発者ポータルに登録し、他者が利活用できるように公開する。	サービサー （民間企業・個人）

(b)想定運用フローの検討

API カタログと開発者ポータルに必要な手順や役割を漏れなく洗い出すために、ここまでに整理した各アクターが、API カタログと開発者ポータルの構築からサービス運用開始までのプロセスにおいて、どのような作業をどのような手順で実施するか、想定フローとして詳細化した。

API カタログと開発者ポータルにおける作業手順を詳細化した結果を以下に示す。

(ｱ)API カタログ開設準備

API カタログの構築事業者を選定、API カタログのコンテンツ作成・収集等を行い、API カタログを開設する。

(ｲ)データ連携基盤構築

各エリアにおいて、データ連携基盤を構築する整備事業者を選定し、データ連携基盤を構築する。データ連携基盤を構築する上で必要となるガイドラインや API 共通ルール、標準仕様の入手は API カタログを通して行い、それらに関連する問合せについても API カタログを通して行われることが想定される。

(ｳ)開発者ポータル開設

各エリアにおいて、開発者ポータルを開設する事業者を選定し、開発者ポータルの開設準備を行う。開発者ポータルを開設する上で必要なガイドラインの入手、関連する問合せ/回答については API カタログを利用してやり取りを行う。

開発者ポータルの開設にあたっては、中央から各種ガイドライン等の提供を通じて一定の品質や機能の統一が図られるが、より厳格に管理・統制する必要がある場合は API カタログ等で個々の開発者ポータルの開設認可制をとることも考えられる。

(ｴ)先端的サービス開発・提供開始

データ連携基盤上で稼働するサービスの開発を行う。サービスを円滑に開発する上では、データ連携基盤上に実装されている API の詳細情報や、効率的に開発を進めるためのサンプルプログラム等が共通的な部品として提供されていることが望ましく、これらは開発者ポータルを通じて提供されるべきものであると想定する。

また、開発においては開発者間の情報共有や公開されている API に関する問合せが発生することが想定され、これについても開発者ポータルにおいてコミュニティ機能や問合せ・回答機能が提供されるべきである。これらの開発者支援機能の提供にあたっては、先端的サービス事業者（サービス開発者）を開発者ポータルのユーザとして登録することも想定される。ユーザ登録を行うことで、開発者ポータル上で提供される各種機能へのアクセス、API 情報の登録、一般に公開されるべきでない情報等へのアクセス制御を行うことができる。また、データ連携基盤としてもユーザ登録がなされる場合は、統一的にログインできるような仕組みとして連携されることが望ましい。

サービス開始後には、当該サービスに利用された API や開発したサービス概要及び新たに開発された API 等の関連情報が開発者ポータルにも登録され、API 活用事例として収集・公開されることが望ましい。それらの情報を充実させることで、分野やサービスごとに使用されている API が明確になり、利用状況等の統計/評価情報の詳細化やエリアごとのスマートシティ戦略方針検討にも活用できると考える。

(オ)運用 統計/評価

データ連携基盤上で実行された API の情報（API の種類・実行回数・実行時間帯）や、開発者ポータル上で頻繁に参照されている API の属性について、分野やサービスごとに統計をとることで利用者の関心が明らかになる。また、API カタログ又は開発者ポータル上で、利用者が API を評価（5 段階の星評価等）できる機能を設けることで、開発者間・エリア間での協創を促すとともに、開発者へのフィードバックや、開発者ポータル・データ連携基盤の運営に役立てることができる。また、それらの各エリアの利用情報を API カタログ側に集約することができれば、各スーパーシティの特徴やサービス充足度等の評価や類似エリアへの戦略推薦等を行うことが可能であると想定する。

API カタログと開発者ポータルの構築からサービス運用開始までの流れを、図 6-2-2-2、図 6-2-2-3 に示す。

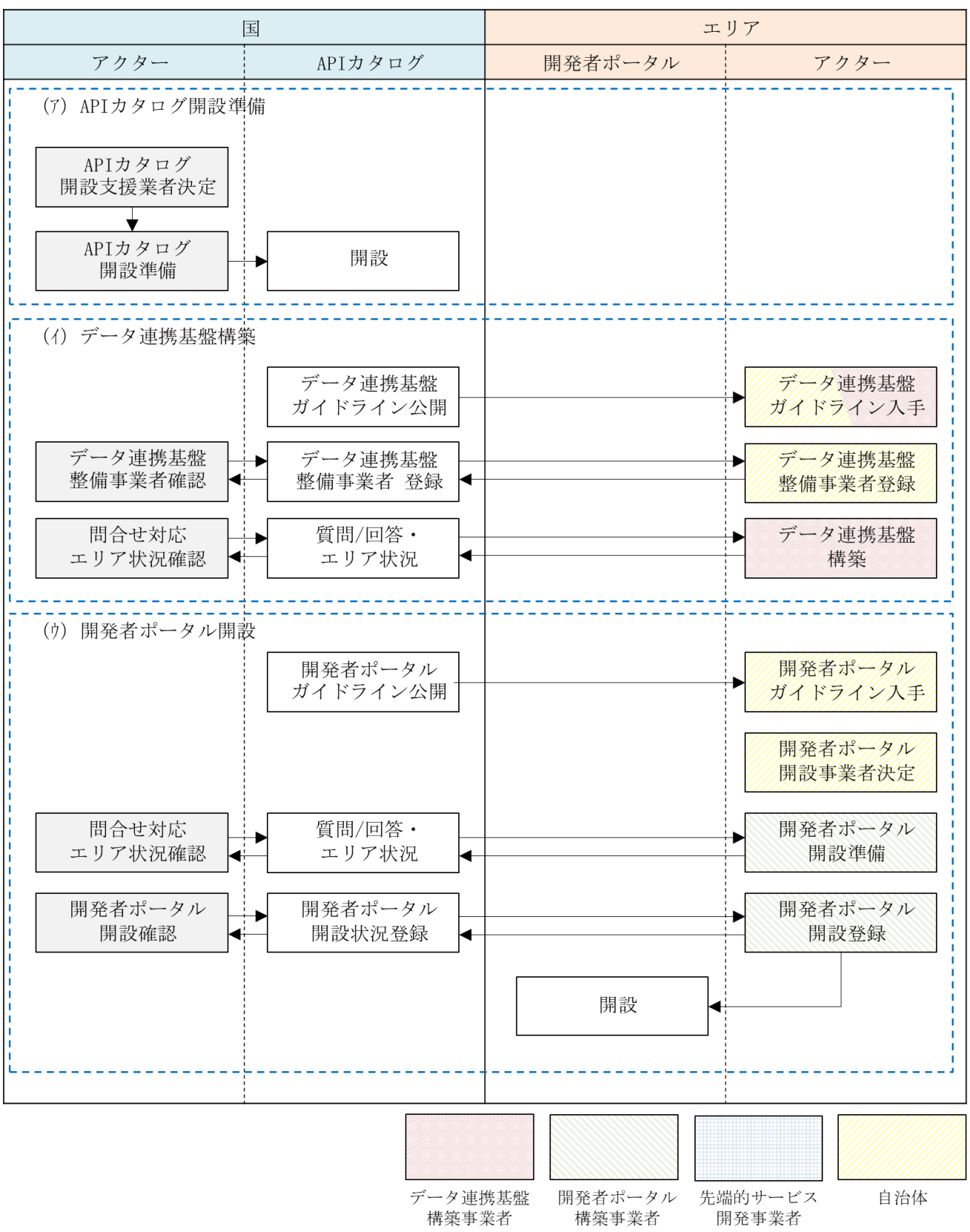


図 6-2-2-2　API カタログと開発者ポータルの構築から運用開始までの流れ①

凡例　□：アクターの行為　□：想定機能

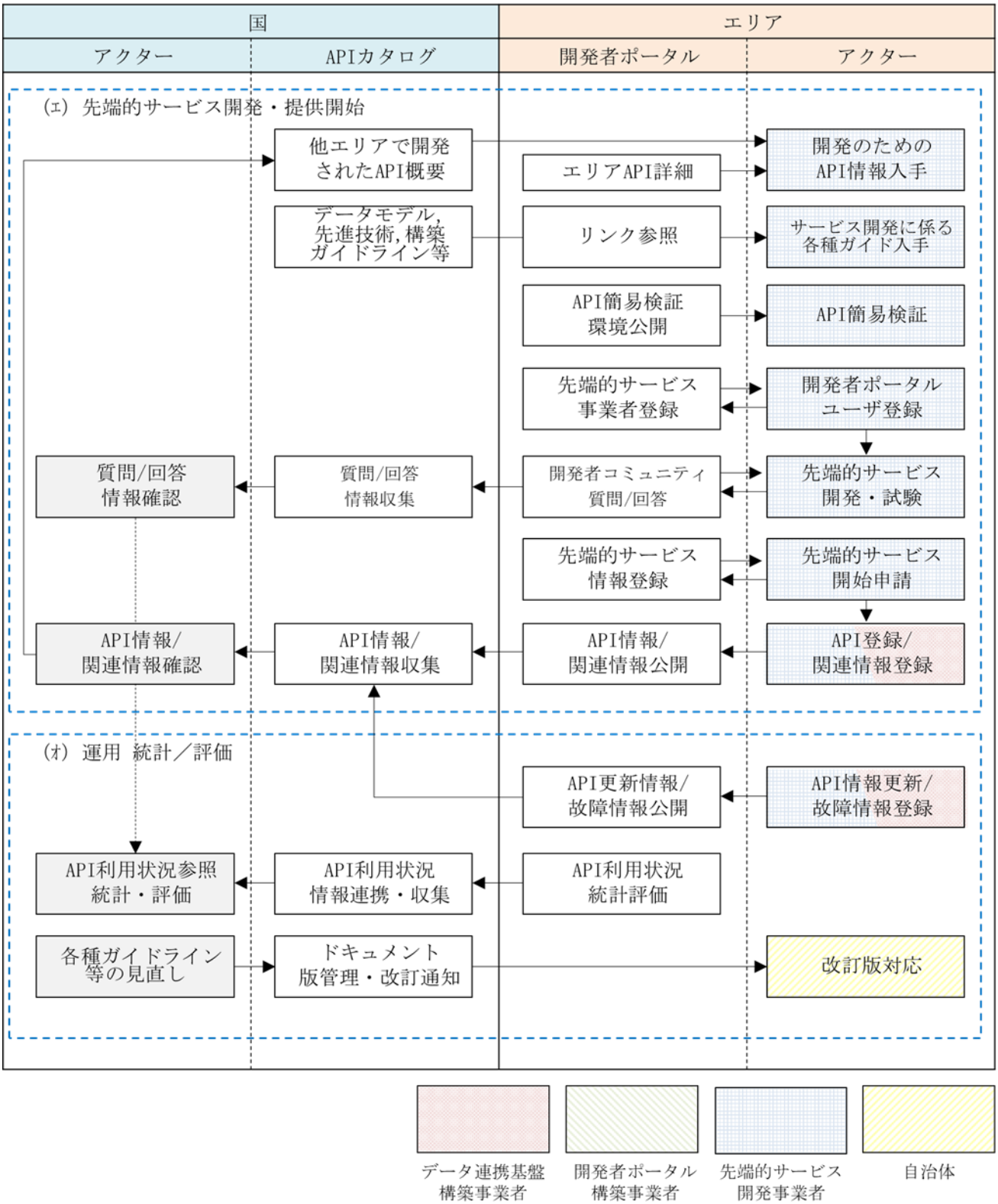


図 6-2-2-3　API カタログと開発者ポータルの構築からサービス運用開始までの流れ②

(c)取り扱い情報の整理

本項では、基礎調査にて抽出した機能項目、及び整理したアクター・想定作業手順の内容を基に、API カタログ及び開発者ポータルが取り扱うべき情報を整理する。

API カタログは、各エリアへの情報発信を統括して行い、各エリアの情報を必要に応じて収集・統合する役割であることから、以下の情報を取り扱うことが妥当であると想定される。

- ガイドライン情報、及びそれらに関連する FAQ 情報
- 先進技術情報、及びそれらに関連する FAQ 情報
- 他省庁等の参考情報
- 各エリアの API 概要情報

また、開発者ポータルは、各エリアを主管する自治体、及びサービス開発等を行う事業者間の迅速な情報共有の促進や連携を深める役割であることから、以下の情報を取り扱うことが妥当であると想定される。

- 開発者同士のコミュニティ情報
- 各エリアにおいて実装される API についての問合せ/回答情報
- 先進技術情報や開発ツール等
- 当該エリアで開発された API の詳細仕様、サービス情報（分野や使用データ）

上記の内容を基に、より具体的な例を交えながら API カタログと開発者ポータルにて取り扱うべき情報を表 6-2-2-4 に示す。

表 6-2-2-4　API カタログと開発者ポータルの取り扱い情報整理表

凡例　○：対象　△:リンク参照　　ー:対象外

No.	分類	情報名	内容	API カタログ	開発者ポータル
1	ドキュメント	利用規約	API カタログの利用規約	○	ー
2			開発者ポータルの利用規約	ー	○
3		利用手順	API カタログの利用手順 （必要であればユーザ登録手順等）	○	ー
4			開発者ポータルの利用手順 （必要であればユーザ登録手順等）	ー	○
5		ガイドライン	データ連携基盤構築ガイドライン データ連携基盤を構築するためのガイドライン	○	ー
6			先進技術情報 データ連携基盤構築、サービス開発に参考となるサービスモデルやデータモデルのガイドライン等の技術情報	○	△
7			開発者ポータル開設ガイドライン • 各エリアで開発者ポータルを開発・運用するためのガイドライン	○	△
8			テンプレート 開発者ポータルに記載する利用規約やコンテンツのひな形	○	ー

No.	分類	情報名	内容	API カタログ	開発者ポータル
9	API 情報	各エリアの API 情報	各エリアで開発した独自の API 利用規約（個別にある場合）や利用条件、使用分野やデータカテゴリ区分、仕様書、実装例サンプル（又はソースリスト)、更新情報（Version 等） ※ API カタログには概要/参照先リンクを掲載する。 仕様等詳細情報は各開発者ポータルにのみ掲載する。	△	○
10	関連リンク	他省庁リンク	他省庁への参考リンク • 各省庁にて実績のあるサービスや API 情報へのリンク	○	－
11		関連技術情報リンク	データ連携基盤構築、開発者ポータルやサービス開発に関連する技術情報や開発ツール等 API 導入ガイド、API テクニカルガイド 政府 CIO ポータルの手引き	○	△
12		API カタログ	API カタログへのリンク	－	○
13		開発者ポータルリンク	各エリアの開発者ポータルへのリンク	○	－
14	オープンデータ技術情報	オープンデータ技術情報	オープンデータの公開開始日や拡張子等、メタデータ（カタログデータ、組織/人等)、推奨データセット	△	○
15	コミュニティ	開発者コミュニティ	開発者等関連するアクター間のコミュニティ。 詳細技術情報や事例、ノウハウ等	－	○
16	問合せ/回答情報	問合せ/回答情報	API カタログ： 利用規約や手順、ガイドライン等における問合せと回答 開発者ポータル： 利用規約や手順、サービスや API に関する問合せと回答	○	○
17	FAQ 情報	FAQ	API カタログ、開発者ポータルに記載する利用規約や手順等において想定される問合せ/回答情報を FAQ 情報として掲載する。	○	○
18	お知らせ情報	更新情報	API カタログ、開発者ポータル各々の更新情報。 • リンク先の追加やガイドラインの改訂等	○	○
19		故障・メンテナンス情報	API カタログ、開発者ポータル自体の故障やメンテナンス情報 故障復旧の履歴や事由、計画停止の予定等	○	○
20	簡易検証環境	簡易検証環境	サービス開発のための API 検証環境	－	○

(3)機能要件の検討

基礎調査の結果から得られた共通的に実装されるべき機能をベースとし、API カタログ/開発者ポータルの役割及び想定利用シーンを考慮し、本事業において追加で必要な機能の検討を行った。検討された機能の全てが国・エリアで必須として実装されるべき機能ではなく、将来的に実装が検討されるべきものや、地域の特色に応じて有効性を判断した上で独自に開発される機能に関する情報等も含まれるため、検討した機能については下記 3 つの区分に分類を行った。

【必須機能】
API カタログや開発者ポータルの役割を満たすため最低限必要と想定される機能。
開発者ポータルの必須機能については、エリアの負荷軽減のため、中央から各エリアに対して構築手順やテンプレートが提供されることが望ましい。

【推奨機能】
API カタログや開発者ポータルをより利活用頂くために有効と想定される機能。
開発者ポータルの推奨機能については、エリアの特色やコスト面、利便性を踏まえて各エリアで実装要否が検討されることを想定する。

【将来機能】
将来的に実装されることが望ましいが、実現方法を含め今後より詳細な検討が必要と想定される機能。

検討・分類した機能の詳細については、別紙イ②-2 に「API カタログ機能一覧案（評価前)」及び「開発者ポータル機能一覧案（評価前)」として示す。

(a)アクター・利用シーンの検討を通じて追加した機能

API カタログの役割の整理及び想定アクター・想定利用シーンの検討を通じて、機能一覧案へ主に以下 4 点の機能を追加した。

(ｱ)開発者コミュニティ以外の問合せ受付/回答機能

基礎調査においては、API カタログ・開発者ポータルともに開発者コミュニティを設け、利用者同士での問題解決がなされることで問合せ対応に係る工数削減ができるとの検討がなされた。しかし、アクターの整理を通じた役割分担・公開されるべき情報の検討を踏まえた結果、API カタログでは各エリアの API 概要情報のみを公開し、詳細情報は各エリアの開発者ポータルが参照されるという分担となった。そのため、API カタログでは開発者コミュニティを通じて解決が促されるような技術的な問合せよりも、各種ガイドラインに対する問合せや構築・運用方針に関する相談がメインとなることが想定される。そういった掲載情報に関する問合せについては、内容としても似通ったものとなることが想定されることから、問合せフォーム等で受付けた内容を FAQ として他エリアにも展開する方式が効率的であると想定される。

開発者ポータルについても、API の詳細な問合せやサービスへ活用方法に関しては開発者ポータルを通じた技術者同士の問題解決やコミュニティ形成が促進されるべきであるものの、開発者ポータル自体の使い方や要望といった問合せを受付けるためにも問合せ機能自体は必要であるとして機能一覧案へ追加している。なお実装にあたっては、プルダウン選択式のような形であらかじめ想定した問合せに内容を制限することや、その他の問合せについてはコミュニティを活用するよう誘導することで運用に係る負荷軽減策が実施されることが望ましい。

(ｲ)各種情報収集（エリア状況、API 概要情報、活用事例）

これまでの検討の中で API カタログは API 概要情報の公開や、各エリアの開発者ポータルへのリンク､及び構築状況を公開するといった役割を有することが望ましいと想定された｡そのため、情報公開の前提機能として想定される各エリアから中央に情報を集約する仕組みについても機能一覧に追加した。これについては中央から各エリアの情報を吸い上げる Pull 型と、各エリアから情報を登録してもらう Push 型での実装方式が考えられる。

詳細な実装方式は将来的に検討が必要となるものの、Pull 型の場合は各エリアのデータ連携基盤上に API 概要や利活用状況といった情報を格納してもらい、中央からそれらの情報へアクセスする方式が考えられる。また、Push 型の場合は API カタログの画面を通じて登録してもらう方法や、API の公開や変更と同時に、自動で API カタログへの更新用 API がキックされる機能を各エリアの開発者ポータルやデータ連携基盤上に共通的に実装する方法等が想定される。

(ｳ)ユーザ登録・ユーザ認可

API カタログ・開発者ポータルともに、自治体やサービス事業者、住民に対して広く情報を公開し、幅広いユーザに利活用頂くことが前提ではあるものの、想定利用シーンの一部でユーザ登録を実装することによるアクセス制限が必要なシーンが想定されるため機能一覧案へ追加を行った（本項 (b)(c)にて詳述)。

(ｴ)エリア状況表示（API カタログ)、データ連携基盤稼働状況管理（開発者ポータル）

想定利用シーンの検討を通じ、API カタログにおいて各エリアのデータ連携基盤構築情報を公開することで、サービス開発者等の利用者がサービス開発計画や内容や対象エリアを検討しやすくする利用シーンが想定された。また、開発者ポータルにおいては、当該エリアのデータ連携基盤の稼働状況や、8 章で検討されるようなサービス開発で共通に役立つような先端的サービスの稼働状況の公開を通じて、サービス開発へ役立ててもらう利用シーンが抽出された。上記より、エリア状況表示機能を API カタログ･開発者ポータルの想定機能としてそれぞれ一覧へ追加した。

(ｵ)情報が更新された際のユーザへのお知らせ機能

開発者ポータルやデータ連携基盤に係る各種ガイドラインを公開する場合は、ユーザが常に最新の情報や重要な変更を参照できるよう、変更発生時にメール等でユーザへ通知できる機能が想定されることから、追加検討が必要な機能として一覧へ追加した。

(b)API カタログが提供すべき想定機能

6-2-1 において基礎調査で得られた一般的・共通的に実装されるべき機能に、上記(a)で検討した追加機能を踏まえ、API カタログとしての機能の在り方や必須度合いといった位置付けについて以下のとおり検討した。

(ｱ)API 概要情報公開・情報収集

API カタログの「各エリアで実装されている API の概要情報を横断的に公開する」という目的を実現するために、API 情報公開は必須の機能である。概要情報として連携されるべき項目は、API 名称、保有機能の概要、提供データに関する説明（データ更新日、提供元、データ内容、更新タイミング、その通知方法）等が想定される。

また、主たる想定利用者であるサービス開発者がサービス開発に必要な情報を迅速に入手できるよう、検索機能の実装も必須条件とし、検索可能項目についてもエリア・分野・データモデル

の形式等複数の項目から横断的に検索できることが望ましい。本機能については、CKAN 等を活用することで一覧形式での表示・検索等が可能となると想定する。なお、API 詳細仕様等については、API 提供元（各エリアの開発者ポータル）へのリンク先を併せて掲載し、そちらを参照する運用が想定される。

また、各エリアの情報を公開するためには、エリアから情報を集約する情報収集機能も必要となる。これにてついては、将来的な運用まで考慮すると各エリアから自動集約できるような仕組みが実装されるべきであるが、(a)に記載のとおり、Push 型・Pull 型それぞれで様々な方法が存在し、実装される機能が確定した後により詳細な検討がなされるべきであることから、将来機能と区分している。

(ｲ)API 概要以外の情報公開

1)ドキュメント

開発者ポータルやデータ連携基盤の構築・運営に係るガイドライン等のドキュメントやツールキットの公開については、API カタログ以外の場所で公開される運用も想定されるが、構築前段階から API カタログを活用頂くことで、API カタログ自体の認知度向上と利活用促進の効果が期待されることから API カタログ上の必須機能として定義している。

2)各エリアの開発状況表示

想定利用者であるサービス開発者が、データ連携基盤を構築している、もしくは今後構築が予定されているエリアへ到達できるよう、データ連携基盤を実装しているエリアの一覧や開発者ポータルへのリンクを掲載する必要がある。本機能は中央と各エリアを連携し、全国的な取り組みとして推進するために不可欠なものであるとして必須機能と区分している。

データ連携基盤の稼働状況（構築中、稼働中、稼働日等）や、稼働しているサービスの状況を表示できることが望ましい。また、表示するための情報の収集にあたっては、後述する各エリアをユーザ登録できるような将来機能の検討を通じ、各エリア側から申告がなされることを想定する。

3)リンク・その他

利用者が当カタログの概要を把握し、常に最新のコンテンツへ迅速にアクセスできるために、メニュー表示や更新情報等のお知らせ表示、API カタログ自体の利用規約・利用手順の掲載は必須と区分した。また、API カタログ上だけで網羅できない情報へのリンクとしては、各省庁にて実績のあるサービスや API 情報へのリンクや、関連する基準・技術情報へのリンク、スーパーシティ関連情報へのリンク等が想定される。

これらの情報は必要に応じて掲載要否の検討が必要であるため推奨機能として区分しているが、上記のとおり各エリアの開発者ポータルへのリンクは必須機能として定義している。

(ｳ)ユーザ登録

API カタログにおいて、各種基準・ドキュメント・API 概要といった情報は、ログイン不要で一般に対して広く公開されるべきものであると想定する。ただし、各エリアのデータ連携基盤・開発者ポータルの構築状況を中央で管理・フォローしたり、エリアの情報を API カタログへ連携する将来機能のために、各エリアをユーザとして登録する機能が候補の 1 つとして挙げられる。

情報収集機能の項でも記載のとおり、API カタログと開発者ポータル間の情報連携についての詳細は将来的に検討が必要な領域ではあるものの、各エリアがデータ連携基盤や開発者ポータル

の開発を開始する段階で、エリアにおける担当者情報（部署・氏名・メールアドレス 等)、開発事業者（データ連携基盤・データ連携基盤)、構築予定日現時点での代表的なサービス等を初期登録するための画面や、状況を一覧表示できるような機能を設けることが想定される。

構築状況については、構築開始・完了・運用開始時といったマイルストーンを設け、API カタログを通じて逐次状況を登録できる機能を設けること等により、中央で全エリアの状況を管理・フォローできるとともに、API カタログ上での状況公開を通じて、今後各エリアでサービス展開を検討しているサービサー等の利用者にとっても有益な情報となることが想定される。

(ｴ)問合せ受付/回答

本項(a)にて既に記載のとおり、API カタログの問合せ受付けに関しては、問合せ受付工数削減のためのコミュニティ機能を設けるのではなく、問合せフォーム等やメール運用等の形で問合せを受付ける機能を持たせることが望ましい。ただし、問合せ対象については API カタログそのものや、提供元として掲載しているコンテンツに関わる情報のみ等、最小限に絞った上で実施されるべきである。また、掲載するコンテンツの性質上、併せて FAQ を整備することで、問合せ受付に係る工数を削減できるものと想定される。

また、問合せ受付/回答に関しては、必ずしも本カタログ上に画面・機能としてフォーム等が設けられている必要は無く、メール等による外部運用の代替案も考慮されることから必須度合いは推奨機能として区分している。

(c)開発者ポータルが提供すべき想定機能

6-2-1 において基礎調査で得られた一般的・共通的に実装されるべき機能に、上記(a)で検討した追加機能を踏まえ、開発者ポータルとしての機能の在り方や必須度合いといった位置付けについて検討した結果を以下に示す。

(ｱ)API 詳細情報提供

1)API 詳細情報・一覧表示

6-2-2(1)に記載のとおり、開発者ポータルの一番の目的は利用者（サービス開発者）に対して API に関する情報をわかりやすく提供することであり、各エリアのデータ連携基盤上に構築されている API に関する情報を一覧表示・詳細表示できる機能は必須で実装されるべき機能として位置付けている。本機能の実現にあたっては、5 調査結果（イ）①API 共通ルール/標準仕様で検討がなされているとおり、各エリアの API 開発者により作成された OpenAPI 仕様ドキュメントを基に、SwaggerUI や ReDoc 等の機能が用いられることが想定される。また、主たる想定利用者であるサービス事業者が、求める情報に到達しやすくするために、API をキーワードから検索できる機能や、一覧表示機能の入り口をスーパーシティの各分野に分けること等が有効である。

2)情報登録/更新

上記の API 詳細表示機能を設けるためには、開発者ポータルへ API 情報を登録し運用の中で情報を更新できる機能も必要機能として想定される。本機能については各エリアにおけるデータ連携基盤の運用に併せて実装方式を選択できるよう「推奨機能」として定義し、機能一覧案上も実装方式まで規定しない形で、候補となる想定方式を提示している。

実装方式としては、利用者による API 登録用の画面を開発者ポータル上に設け、上記の仕様ドキュメントをアップロードできる方式とすることが想定されるほか、ポータル上の画面とし

ては実装せず、GitLab・GitHub 等により開発者ポータルサイト自体に対して API 登録者からのソースマージリクエストや、データ連携基盤上に登録したファイルの自動連係等を通じて開発者ポータルに対して直接登録する方式も想定される。いずれの場合においても、不適切なユーザからの登録や、不正な利用が想定される API が登録されないよう、後述するユーザ登録機能等による API 登録者の制限や、API 登録時に運営側による承認プロセス等が設けられることが望ましい。

(ｲ)API 詳細以外の情報公開

1)エリア内サービス・データ連携基盤稼働状況

8 章（エ）で検討されるような、当該エリアのデータ連携基盤上で稼働もしくは連携され、各エリアのサービス開発に共通的に利活用されることを目的とした先進技術や共通サービスが存在する場合、それらの一覧や稼働状況等の情報を利用者に公開する機能が必要となる。本機能については先進技術との連携や共通サービスの対象の精査に併せて将来的に検討が必要となる機能として区分している。また、本機能に併せてデータ連携基盤自体の稼働状況の表示がなされることや、運用負荷軽減のためにサービスの状況やデータ連携基盤の状況を収集する機能が存在することが望ましいが、こちらについてもデータ連携基盤の詳細仕様と併せて将来的に検討が必要となるため将来機能として区分している。

2)API 統計情報・活用事例

各エリアで、データ連携基盤上の API を活用したサービス・ソリューションの事例をモデルケースとして資料化し他事業者に参照頂くことで、データ連携基盤の利活用が推進され、新たなサービサーの参画も促されると想定する。

開発者ポータルとしては、それらの活用事例の資料を公開する場を設けることが望ましいが、登録される全ての API に対して事例資料の作成を義務化することは、運用負荷の増大や活発な API 登録を阻害する要因となり得る。そのため、作成の対象を特に利活用されている API に絞ったり、有償 API の宣伝目的で登録者側に作成してもらったものを掲載する等、想定運用や基準を中央から提示することが望ましい。また、それらを踏まえて資料の作成・掲載及び掲載のための機能実装の要否は各エリアにて判断されるべきと想定されるため、当機能は推奨機能として区分している。また本機能の関連として、各地域単位で実施されるコミュニティイベントや事業者向けイベント（アイデアソン・ハッカソン・各種勉強会等）の成果資料等も同様の機能を用いて公開されることが望ましい。

3)リンク・その他

開発者が本ポータルの概要を把握し、常に最新のコンテンツへアクセスできるために、サイト上の各コンテンツのメニュー・リンクや、直近で更新された情報に関するお知らせをトップ画面に表示すること、ならびに当ポータル自体の利用規約や利用手順を掲載することは全エリア共通で行われるべき事項として必須機能とした。また、各エリアにおけるサービス開発において基礎情報として共有されるべき情報へのリンク掲載も必須機能としており、中央で整備される API カタログへのリンクや、当該エリア自治体へのリンク、オープンデータ推奨データセットや API テクニカルガイドライン等の既に整備されている参考技術情報等へのリンク掲載がなされることで、開発者ポータルの利便性が向上すると想定される。

(ｳ)開発者向け支援機能

1)開発者コミュニティ機能

API テクニカルガイドにおいても推奨されているとおり、各エリアにおける問合せ削減及び開発ナレッジ共有のために、開発者同士が相互にコミュニケーションをとれる場である開発者コミュニティ機能は各エリア必須で導入されるべき機能であると想定される。本機能は、開発者ポータル上から直接操作できる形で BBS のような機能を作りこむ形態や、コミュニティ機能自体は GitHub や Slack 等の外部チームコラボレーションツールを活用し、開発者ポータル上にはツールへのリンクや加入方法・利用方法・利用手順のみを掲載する方式等が想定される。詳細な実装方式については、各エリアにおけるコミュニティ形成対象の考え方やポータル運用方針、実装・運用に係るコストを踏まえて各エリアで検討されるべきと想定される。

2)簡易検証環境機能

本ポータルの利用者として想定される先端的サービス事業者等に対して、各 API を簡易に検証するための環境を開発者ポータル上で提供できることで、サービス開発・評価の効率化や掲載する API の利活用が推進されると想定される。本機能については API テクニカルガイドブックにおいても重要機能と位置付けられていることから、仕様検討の段階においては各エリア必須で実装されるべきものとし、詳細なニーズについては 6-2-3 ヒアリングに基づく仕様検討の中で精査する。

(ｴ)問合せ受付/回答

各エリアにおける運用負荷軽減のためには、可能な限り開発者同士で問題解決がなされることが望ましく、開発者ポータル上の問合せ/回答の役割は上述している開発者コミュニティ機能に集約されるべきであると想定する。開発者コミュニティ機能を各エリアで実装必須とする場合、問合せ受付/回答機能を別で設けることは役割重複となるため、開発者コミュニティ機能でカバーできない質問の受付要否等、各エリアの判断に応じて将来的に実装検討がなされるべきものとして将来的に検討が必要な機能と区分している。

(ｵ)ユーザ登録

開発者ポータルにおいても、API カタログ同様、各種基準・ドキュメント・API 詳細といった情報は、ログイン不要で一般に対して広く公開されるべきものであると想定する。ただし、有償データ・API 等一部のユーザにアクセスを制御したいケースや、開発者ポータル上へ API を登録するユーザを制限したい場合等において、開発者ポータルにログイン機能を設けられるべきシーンが想定される。本機能に関しては、上記のようなデータ・API を取り扱うエリアに限定されるほか、API 登録に関してはデータ連携基盤側でユーザ認証を実施する方式も考えられることから、将来的に検討が必要な機能として定義している。

6-3. ヒアリングに基づく仕様の精査

6-3-1. ヒアリングの考え方（目的、ヒアリング先、ヒアリング項目）

前項までで検討・選定した想定利用シーンや機能案について精査を行うために、実際に類似サイトを開発・運用している事業者にヒアリングを行い、必要と想定される機能が過不足なく検討できているか再評価を行った。ヒアリング先は、以下に示すとおり「開発者の観点」と「運用事業者の観点」で選定した。

- 「開発者の観点」

API カタログ・開発者ポータルに類するシステムの開発経験を有し、行政・市民・企業との連携を通じた地域課題解決のためのコミュニティ形成を担うシビックテック推進団体をヒアリング先として選定した。ヒアリング項目は今回検討した機能・仕様に対する意見、関連事例の内容、実装にあたっての地域目線での課題や想定されるアプローチ、技術的・ボリューム的に今回の検討内容でエリアが構築できるか、等とした。

- 「運用事業者の観点」

特定分野でデータ連携基盤及び API 情報公開サイトを運用しているデータ連携基盤運用事業者を選定した。ヒアリング項目は、既存の類似事例における機能検討や運用方法の内容や、今回検討した機能・仕様に対する運用面からの意見を中心として取りまとめた。特に既存事例の運用内容については、机上検討だけでは想定しきれない利用シーンの検討や運営・運営に関する機能、設計の考え方、実装方法、課題・改善案を参考とすることを目的として、ヒアリング項目を検討した。

代表的なヒアリング項目の分類と、内容例について以下に示す。

(1)想定した運用や手順、各アクターの担当役割について
- 検討した運用フローは事業者から見て適切か、抜け漏れや追加すべき手順はないか
- 各アクターの役割は妥当か、負荷が高い箇所や不足する役割や利用シーンはないか
- 登録される API の品質をチェック/審査する仕組みについて
- 参画する事業者の審査フローについて

(2)API カタログの画面構成/機能について
- 機能案や掲載を想定しているコンテンツに過不足はないか
- 既存事例やユーザからの要望、問合せを通じた利用者ニーズに関する情報について

(3)開発者ポータルの画面構成/機能について
- 機能案や掲載想定しているコンテンツに過不足はないか
- 既存事例やユーザからの要望、問合せを通じた利用者ニーズに関する情報について

(4)開示/登録情報について
- 登録される API に関して、検索を容易にするための分野・種別の決定方法について
- 既存事例における、開設前に事前準備した/すべきであったコンテンツについて
- データ連携基盤の構築状況やサービス稼働状況の可視化の必要性について
- API の利用情報等、統計情報の利活用の方法について
- 簡易検証環境に対する開発者目線でのニーズと、考えられる実装方式について

(5)開発者コミュニティについて
- 開発者同士のコミュニティ形成を促進、活発化させる、運営面での取り組みや
  開発者コミュニティがツールとして備えるべき機能やサービス提供方法について

(6)問合せ/回答について
- 問合せ/回答における課題や、効率的に行うための工夫点について
- よくある問合せの傾向と対策について

(7)運用/保守について

- 掲載されているコンテンツの最新化や情報の更新タイミング等について
- ドキュメントや API 等に誤りがあった場合の検知トリガーやユーザへの通知方法について
- 既存の運用事例から参考にすべき点や、現状改善すべきと認識している点について

6-3-2. ヒアリング結果

(1)シビックテック推進団体

シビックテック推進団体に対して、類似サイトの開発者・利用者の視点で 6-3-1 に記載のとおりヒアリングを実施した結果の総括を以下に総括として述べる。なお、ヒアリング結果詳細は別紙イ②-3 ヒアリング結果（シビックテック推進団体）に示す。

(a)機能充足性の観点

ヒアリングの結果、検討した機能一覧案については、API カタログ・開発者ポータルとして必要な機能が列挙されているとの回答を得た。ただし、機能が多すぎると実装時に各エリアの負担となる可能性があるため、各エリア必須で開発が必要とするのは最低限の重要な機能に限定し、その他は推奨機能にすることが必要となる。例えば、機能一覧上エリア側で必須機能として検討されている「簡易検証機能」については、開発者の目線からは必ずしも開発者ポータル上の機能として作りこまれる必要はなく、出力データのサンプルが GitHub 上で公開されていたり、参考プログラムを Docker で試せる状態にしておく等の代替案が考えられる。

また、API カタログにて検索の際に使用される、カテゴリや分類項目等の整理が追加で必要となり、他の類似事業との関連や整合性等も併せて検討される必要がある。特に、API カタログの公開コンテンツのひとつとして、関連府省庁とのリンク等を公開すると検討されているが、現在各府省庁主導で行われているサービス・API 提供事業や基盤整備事業といった類似の取り組みと連携することや、将来的にそれらの他事業を含めた横断的な検索ができることが望ましいという意見を得た。

(b)相互運用性の観点

検討の中ではエリア単位で開発者コミュニティを設ける想定となっている。ただし、相互運用性の観点からは、エリア内の開発者コミュニティ形成だけでなくエリア横断でのコミュニティ形成を通じ各エリアの事例共有やハッカソン等を通じたノウハウ共有が行われることが重要となるとの意見を得た。

エリア横断のコミュニティの形成については、API カタログのような場でもコミュニティ機能を設ける等、国としてコミュニティの場を提供することも想定され、単なる情報連携のためだけではなく、エリア間のコミュニケーションを通じ有益な情報が得られるような仕組みを作ることで全国的なコミュニティの活性化を図る必要がある。

また、コミュニティの運営は中央主導ではなく、各エリアが持ち回りで運営する等、運営主体を各エリアに委ねることで、コミュニティに対する心理的安全性を高めたり、相対的な運営を通じたより一層のコミュニティ活性化に繋がる期待が持てるとの意見を得た。

そのように機能を追加していくとなると、国が構築する API カタログはただ単に API 概要情報を提供するものだけでは無くなるため、サイトの役割に沿った適切な名称の工夫等が今後必要となると想定される。

(2)既存事例の運用事業者

6-3-1 に記載のとおり「運用事業者の観点」に対してヒアリングを実施した結果を、以下に総括として

述べる。なお、ヒアリング結果の詳細は別紙イ②-4 ヒアリング結果（運用事業者）に示す。

(a)機能充足性の観点

現在運用されている開発者向け API 公開サイトでは、API の機能・用途から検索する方法と、アウトプットであるデータの特徴を指定して検索する方法が存在する。登録される API をどちらの検索メニューでヒットするものとするかは API を登録する事業者が決定しており、利用傾向としては API の機能・用途から検索する方法が主に使われている。

また、機能一覧案で触れられている簡易検証環境について、当該事例では実装を行っていない。その代わり、API 利用者である各種サービサーの API 動作を事前に検証したいというニーズに対しては、検証に必要となるデータや情報、マニュアル等を随時利用者に公開することで対応している。また、そういった運営状況に対して、利用者から簡易検証環境を実装してほしいという要望は挙げられていない。

(b)運用性・保守性の観点

- API の検索キーとして用いられる分野や分類といったメタデータについては、API の登録者が全て入力する形式としており、運営側での確認や追加情報の付与等は行っていない。API 登録者が検索キーを設定する際に、登録者ごとに入力基準がばらついていたり、入力漏れ等が発生してしまうと API 公開サイトとしての利便性が損なわれ、問合せ等による運営負荷の増大も想定される。そのため、分類や種別等のメタデータに関する定義付け及び誤記・認識齟齬を防止するための API 登録の際のマニュアルの事前の整備と、API 登録者への周知徹底が重要であるとの意見を得た。
- 円滑に運用を開始するために事前準備しておくべきドキュメント等について、少なくとも事例集等は稼働時には必須でない。また、その他のドキュメントに関してはヒアリング先が運用している既存事例においても開設後に徐々に資料を追加している状況である。その中でも、利用者とプラットフォーム間での利用規約や契約内容に関するドキュメントは稼働時から整備され、当該サイトで公開されておくべきものであったと想定する。
- 事業者から登録される API 全件に対する品質担保はできていない。登録事業者は事前にプラットフォームへの利用申請の中で、利用規約へ同意した上で運営側での審査を設け、その過程を通じて品質を担保している。また、上述のとおり API 全件の品質確認はしていないが、プラットフォームの利用状況について、ログ等からシステムの負荷状況の監視は行っている。
- API に寄せられている評価については、現時点では評価された結果を十分に運営に反映できていない状況である。また、統計については分野ごとの API 登録状況や利用状況、参照状況等に関するログ情報等を取得しているものの、利用用途は基盤への負荷状況の監視に留まっている。API の利活用状況を分析し、他の利用者へ推薦したり、サイト運営に反映させるといった活用方法は考えられるが、運用負荷が高い中で有効な利活用方法の検討までは至っていない状況である。本事業で各エリアに対して統計・評価といった機能を必須とする場合は、得られたデータを利活用するための分析手法等も併せて情報展開がなされるべきであるという意見を得た。
- 登録されている情報のメンテナンスについて、全て API 等の情報の有効性を手作業で確認することは現実的ではない。障害発生の検知方法については、クラウド側からの障害情報通知で発覚するものや、ユーザの問合せで発覚するもの、また、API 提供者側から情報展開がなされる場合等が存在する。

6-3-3. 仕様案へのフィードバック

6-3-2 に記載のヒアリングの結果を基に、機能一覧案に対する見直し観点を以下のとおり洗い出した。

- シビックテック推進団体へのヒアリングの際、必須機能が多すぎる場合は API カタログ・開発者ポータルの開発時に負担が増加する可能性があることが議論された。特に開発者ポータルにおいては、エリアごとに規模の違い等によって開発や運用にかけられるコストが異なることが想定される。必須機能を最低限にしつつ、エリアの規模や要望によりカスタマイズを可能とするため、機能一覧案に対して以下のとおり必須機能の見直しを行う。
  - ・FAQ を必須から推奨に変更

    開発者コミュニティ等の充実により、コミュニティ内で議論された内容を集約することで FAQ の役割を代替できるケースが想定されるため。
  - ・API 開発評価環境提供（簡易検証環境提供）を必須から推奨に変更

    API 開発評価環境提供（簡易検証環境提供）機能を必須機能として開発者ポータル上で提供するとした場合、開発・運用自体のコストや、API が追加になった際の改修コストがエリアの負担となり、導入のハードルが上がるとともに継続的な運用を阻害する要因となる可能性が高い。また、ヒアリングの結果から、既存事例ではサンプルプログラム等を提供すること等でも代替できているという意見が挙がったため、区分を推奨に変更した。

- 機能一覧案の検討段階において、開発者コミュニティ機能は開発者ポータルのみで実装を必須とし、API カタログの機能としては不要であると想定していた。しかし、シビックテック推進団体へのヒアリングの結果として、エリア間の連携促進のためには中央としてもコミュニティの場を提供する必要があるという意見が挙がった。このため、API カタログ側での開発者コミュニティ機能を将来検討が必要な機能として API カタログに追加した。

- 既存運用者の観点からは、運用の負荷が高いことが課題として挙がっており、今後の機能改善の方向性としても負荷軽減機能が重点的に検討なされていたり、運用の外注化が検討されている。各エリアにて構築・運用される開発者ポータルに関しても、運用開始後には同様の課題が発生すると想定される。よって、本調査事業の内容を基に各エリアに対する標準仕様等のテンプレートが公開される際は、各エリアの運用側の負荷を軽減し、持続性に寄与できる作りとする必要がある。具体的には、開発者コミュニティによる問合せ削減や、利用者によるデータ入力の仕組みによる修正作業減（ユーザが入力しやすい、エラーが発生しにくい、そのままの形で利活用できる）について検討がなされること等が想定される。また、中央として各エリアが抱える運用の課題や改善案について、API カタログの問合せ機能であったり、将来的に実装される API カタログ上のコミュニティ機能等で集約し、上記の標準仕様にフィードバックしていくことが望ましい。

既存事例運営事業者へのヒアリングの結果、統計・評価に関する機能について、情報収集しても負荷情報の確認や異常データの発見の用途に留まるケースが多く、課題の抽出等データを分析することはエリアの負荷が高くなる。本件については、上記の運用負荷削減施策と併せて、エリアにおけるデータ連携基盤・開発者ポータル運営に有用なデータの種類や取得方法、分析手法といった内容を、将来的に中央からエリアに対してまとめて展開することが望ましい。よって、機能概要に「本機能は初期段階では負荷情報の確認による異常データの発見のために実装を行う。統計・評価に関するデータ分析については収集した情報の活用方法も含めて、中央から提言することを今後検討する。」という旨を追記する。

6-4. 仕様検討結果

6-4-1. 仕様検討結果

API カタログと開発者ポータルに関する基礎調査・仕様検討、ヒアリング調査を通して、各仕様の整理を行った。

(1)API カタログの機能一覧案

API カタログの機能一覧案の仕様検討結果について表 6-4-1-1 に示す。なお、将来機能を含む機能の一覧については別紙イ②-5 に示す。

表 6-4-1-1 API カタログの機能検討結果

No.	画面名	機能名	機能概要	必須
1	全画面共通	メニュー機能	API カタログの全画面共通で、画面上部にサイトの各メニューへアクセスできるリンクを表示する。	○
2	サイト トップ	更新情報お知らせ	API カタログにて管理しているコンテンツ（ガイドラインや利用規約等）の更新情報を表示する。 また、API カタログに掲載されているエリアに追加があった場合も、この画面で表示される想定。ただし、各エリアの API の追加/更新/削除については表示しない。	○
3	サイト トップ	障害・メンテナンス 情報表示	API カタログの障害情報やメンテナンス時の計画停止情報等を表示する。	－
4	利用ガイド	利用規約公開	API カタログで各エリアに対して公開される情報（ガイドライン、API 仕様、標準データモデル等）に関するライセンス形態、禁止事項、免責事項を表示する。 API カタログに掲載される、各エリアの API の利用規約については、掲載元である各エリアの開発者ポータルを参照することとし、それに関する免責の記載が必要な場合はこの画面に表示する。	○
5	利用ガイド	利用手順公開	API カタログの概要、操作手順を表示する。API カタログでできることや、公開されている情報等をわかりやすく表示し、各メニューへリンクさせる。 また、API カタログの将来機能として、ユーザ情報登録機能等が実装される場合は、登録フロー等の情報が掲載されるべきと想定する。 ただし、API カタログに掲載される、各 API の利用手順については、各エリアの開発者ポータルを参照することとする。	○
6	利用ガイド	エリア状況確認	データ連携基盤を実装しているエリアの一覧を表示し、当該エリアの開発者ポータルへのリンクを掲載する。 また、データ連携基盤の稼働状況（構築中、稼働中、稼働日等）や、稼働しているサービスの状況を表示できることが望ましい。 元となる情報については、各エリア側から申告機能を将来実装することを想定する。	○

No.	画面名	機能名	機能概要	必須
7	利用ガイド	関連サイトリンク表示	関連するサイトへのリンクを表示する。 (1)関連する省庁のリンク 各省庁にて実績のあるサービスやAPI情報へのリンク (2)基準・技術情報へのリンク データ連携基盤構築、開発者ポータルやサービス開発に関連する技術情報や開発ツール等 (3)内閣府のスーパーシティ関連情報へのリンク	－
8	API公開・提供	API一覧表示	各エリアで提供している API の概要情報を一覧形式で表示する。概要情報として連携されるべき項目は「API名称、保有機能の概要、提供データに関する説明（データ更新日、提供元、データ内容、更新タイミング、その通知方法)」等が想定される。 また、主たる想定利用者である先端的サービス事業者がサービス開発に必要な情報を迅速に入手できるよう検索機能は必須とし、検索可能項目についても、エリア、分野、データモデルの形式等複数の項目から横断的に検索できることが望ましい。 なお、API詳細仕様等については、API提供元へのリンク先を併せて掲載し、そちらで参照される想定。	○
9	API公開・提供	API情報収集	各エリアのAPI情報を自動で連携するための方法として将来的に検討が必要となる機能。 中央から各エリアの情報を吸い上げるPull型と、各エリアから情報を登録してもらうPush型での実装方式が考えられる。詳細な実装方式は将来的に検討が必要となるものの、Pull型の場合は各エリアのデータ連携基盤上にAPI概要や利活用状況といった情報を格納してもらい、中央からそれらの情報へアクセスする方式が考えられる。また、Push型の場合はAPIカタログの画面を通じて登録してもらう方法や、APIの公開や変更と同時に、自動でAPIカタログへの更新用APIがキックされる機能を各エリアの開発者ポータルやデータ連携基盤上に共通実装してもらうこと等が想定される。	－
10	API公開・提供	API統計情報表示	API カタログで公開している各エリアの API について、API カタログ上で頻繁に参照されているものや、各エリアのデータ連携基盤において特に利活用されているAPIの情報を表示する。 各エリアの利活用状況については、個別に連携・登録される必要があるが、API カタログにおける被参照情報の収集にあたっては、Googleアナリティクス等のアクセス統計ツールが活用できると想定する。当該統計情報はアクセス負荷情報の把握にも利活用される想定。	－

No.	画面名	機能名	機能概要	必須
11	開発者向け情報提供	ドキュメント公開	先進技術情報、データ連携基盤構築や開発者ポータル開発ガイドライン等のドキュメントを表示、ダウンロードを可能とする。PDF や HTML 形式での公開も可能ではあるが、GitLab、GitHub、Note 等を活用することで、版管理や版ごとの変更差分を表示できる等のメリットが存在し、更新履歴や更新部分を説明するための資料作成等の手間を削減することが可能となる。	○
12	開発者向け情報提供	ツールキット提供	各エリアがデータ連携基盤構築や開発者ポータルを構築する際に活用できるテンプレート・ツールキット・SDK 等を公開する。公開にあたっては API カタログ上に直接ファイルを公開する方式に加え、GitLab、GitHub、Docker 等の外部ツールを提供内容や提供目的に応じて使い分けることが想定される。	○
13	開発者向け情報提供	開発者コミュニティ	開発者間のやり取りを実現する。開発者間で開発上の課題を共有・解決することで、問合せ対応の負荷軽減や改善点の発見、ナレッジの蓄積を行う。 外部のチームコラボレーションツールを利活用することを想定しており、API カタログ上にはそれらへのリンクや利活用方法が掲載されることを想定している。現段階で本機能に活用できると想定するチームコラボレーションツールの例は以下のとおり。 （Zendesk、phpMyFAQ、GitLab、GitHub、Slack、Teams）	－
14	FAQ 一覧	FAQ	開発者コミュニティ、又は問合せ回答でよく受ける質問・回答例を整理、表示する機能。	－
15	問合せ/回答	各種ガイドラインや API カタログ自体に関する 問合せ/回答	先進技術情報や、データ連携基盤構築ガイドライン、開発者ポータルガイドライン等に関して個別に問合せを受付ける画面。問合せ対応負荷を軽減するため、受付時にプルダウンメニューからカテゴリを選択する方式の実装や、FAQ への誘導がなされることが望ましい。	－

また、サイトツリーと画面イメージを図 6-4-1-1 に示す。

【凡例】
◎　：必須機能
- - - - - ：外部サイトへのリンク

画面名	画面イメージ	画面設計における 検討事項	機能名
サイトトップ		・更新情報は年月日形式で表示する。 ・更新箇所が分かるようにカテゴリ（利用規約、APIの追加等）を表示する。	◎更新情報お知らせ
利用ガイド	 関連サイト ・外部技術情報 ・各省庁の事例 ・各開発者ポータル等	・各エリアの開発者ポータルのリンクは、都道府県等から絞り込みが行えるようにする。 ・各種関連情報へのリンクを掲載する。 (1)関連する省庁 (2)基準・技術情報 (3)内閣府のスーパーシティ関連情報	◎利用規約 ◎利用手順 ・関連サイトリンク
API公開提供	 API詳細情報 └各エリアの開発者ポータル上で公開されるAPI仕様の詳細	CKAN形式でAPI一覧を表示しカテゴリやエリア等による検索を可能にする。 公開対象はエリアから集約したAPI概要とし、各APIの詳細はエリアが公開するものへのリンクを行う対応する。	◎API概要一覧 ・活用事例 ・API統計情報 ・API評価情報
開発者向け情報	 外部チームコラボレーションツール （Slack, GitLab, GitHub等）	外部のチームコラボレーションツールを活用する。	◎ドキュメント公開 ◎ツールキット提供 ・開発者コミュニティ
FAQ	－	－	・FAQ
問合せ	－	－	・先進技術情報問合せ ・標準API問合せ ・開発者ポータル ・ガイドライン問合せ

図 6-4-1-1　API カタログのサイトツリーと画面イメージ

(2)開発者ポータルの機能一覧案

開発者ポータルの機能一覧案の仕様検討結果について表 6-4-1-2 に示す。なお、将来機能を含む機能の一覧については別紙イ②-5 に示す。

表 6-4-1-2　開発者ポータルの機能検討結果

No.	画面名	機能名	機能概要	必須
1	全画面共通	メニュー機能	開発者ポータルの全画面共通で、画面上部にサイトの各メニューへアクセスできるリンクを表示する。	○
2	サイトトップ	更新情報お知らせ	開発者ポータルにて管理しているコンテンツ(各 API の詳細情報、簡易検証環境、利用規約等）の更新情報を表示する。	○
3	サイトトップ	障害・メンテナンス情報表示	当該エリアのデータ連携基盤や開発者ポータルの障害情報やメンテナンス時の計画停止情報等を表示する。 将来機能として検討されている当該エリアのサービス稼働状況が実装されている場合は、併せて表示する。	○
4	利用ガイド	利用規約公開	開発者ポータル上で公開される情報（各 API の詳細情報、オープンデータ、簡易検証環境・開発者コミュニティといった開発支援ツール)に関するライセンス形態、禁止事項、免責事項を表示する。	○
5	利用ガイド	利用手順公開	開発者ポータルの概要、操作手順を表示する。開発者ポータルでできることや、公開されている情報等をわかりやすく表示し、各メニューへリンクさせる。 また、開発者ポータルの将来機能としてユーザ情報登録機能が実装される場合は、登録フロー等の情報が掲載されるべきと想定する。	○
6	利用ガイド	関連サイトリンク表示	関連するサイトへのリンクを表示する。 (1)基準・技術情報へのリンク データ連携基盤構築、開発者ポータルやサービス開発に関連する技術情報や開発ツール等 (2)オープンデータ技術情報 オープンデータの公開開始日や拡張子等、メタデータ（カタログデータ、組織/人等)、推奨データセット (3)API カタログへのリンク	ー
7	API 仕様公開・提供	API 一覧表示	各エリアのデータ連携基盤で実装されている API の情報を一覧表示する。SwaggerUI や ReDoc 等の機能を使い OpenAPI 仕様ドキュメントを公開する想定。 また、主たる想定利用者であるサービス事業者が、求める情報にリーチしやすいよう、キーワード検索機能やスーパーシティの分野ごとに入り口を分けること等が有効と想定する。	○

No.	画面名	機能名	機能概要	必須
8	API 仕様公開・提供	API 詳細	各 API の詳細な仕様を公開する。 公開内容の詳細は 5-1-3(2)ドキュメントの公開に記載のとおり。 SwaggerUI や ReDoc 等のツールを使い OpenAPI 仕様ドキュメントを公開する想定。	○
9	API 仕様公開・提供	API 開発評価環境提供 （簡易検証環境）	先端的サービス事業者等に対して、サービス開発評価向けに API を簡易に検証するための環境を提供する。実装される場合の方式は以下のいずれかを想定。 (1)検証用のデータ連携基盤を構築してそこに対してリクエストを送ってもらう。 (2)モックアップのような形で、定型的なレスポンスを返す小規模な検証環境を構築する。 なお、費用対効果の面で全エリア必須機能として実装されるにはハードルが高いため、必ずしも本機能を実装せずとも、サンプルプログラムの提供やマニュアルの充実化等でも代替可能。	－
10	API 仕様公開・提供	活用事例	各エリアで、データ連携基盤上の API を活用したサービス・ソリューションの事例をモデルケースとして資料化し、他事業者に参照頂くことでデータ連携基盤の利活用が推進され、新たなサービサーの参画も促進される。当該機能では、それらの活用事例の資料を公開する。また、各地域単位で実施されるコミュニティイベントや事業者向けイベント（アイデアソン・ハッカソン・各種勉強会等）の資料等もここで公開されることが望ましい。	－
11	API 仕様公開・提供	API 登録	当該エリアのデータ連携基盤独自で開発した API の登録やメンテナンスを行う。API の活用事例を表示する。 当該エリアのデータ連携基盤で開発した API 情報の登録・更新・削除を行う。5-1-3(2)ドキュメントの公開に記載のとおり、OpenAPI 仕様ドキュメントが登録される想定。API 情報の登録にあたっては、ユーザ登録等の機能を通じて、承認されたユーザのみが登録できることが望ましい。	－
12	開発者コミュニティ	情報交換機能	開発者間のやり取りを実現する。開発者間で開発上の課題を共有・解決することで、問合せ対応の負荷軽減や改善点の発見、ナレッジの蓄積を行う。 外部のチームコラボレーションツールを利活用することを想定しており、開発者ポータル上にはそれらへのリンクや利活用方法が掲載されることを想定している。現段階で本機能に活用できると想定するチームコラボレーションツールの例は以下のとおり。 （Zendesk、phpMyFAQ、GitLab、GitHub、Slack、Teams）	○

No.	画面名	機能名	機能概要	必須
13	FAQ	FAQ 一覧	よく受ける質問・回答例を整理、表示する機能。 開発者コミュニティでも同様の役割は担えるため必須機能とはせず、各エリアにおけるコミュニティの在り方やユーザ管理の方法に応じて全ユーザが参照すべき内容がある場合は個別実装される想定。	－

また、サイトツリーと画面イメージを図 6-4-1-2 に示す。

【凡例】
◎　：必須機能
＿＿：外部サイトへのリンク

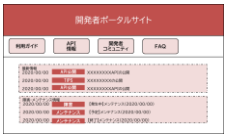
画面名	画面イメージ	画面設計における検討事項	機能名
サイトトップ		・更新情報は年月日形式で表示する。 ・更新された箇所が分かるようにカテゴリ（利用規約、APIの追加等）を表示するようにする。	◎更新情報お知らせ ◎障害・メンテナンス情報表示
利用ガイド		関連する情報のリンクを表示する。 リンク先はAPIテクニカルガイドブック、国が提供するAPIカタログ等を想定。	◎利用規約 ◎利用手順 ・関連サイトリンク
API公開提供		OpenAPI仕様ドキュメント公開のため、SwaggerUIやReDoc等を活用する。	◎API一覧表示 ◎API詳細 ・API開発評価環境提供（簡易検証環境提供） ・活用事例 ・独自API登録/更新
開発者コミュニティ		外部のチームコラボレーションツールを活用する。	◎開発者コミュニティ
FAQ	－	－	・FAQ

図 6-4-1-2　開発者ポータルのサイトツリーと画面イメージ

API カタログと開発者ポータルの基本的な考え方のイメージを図 6-4-1-3 に示す。

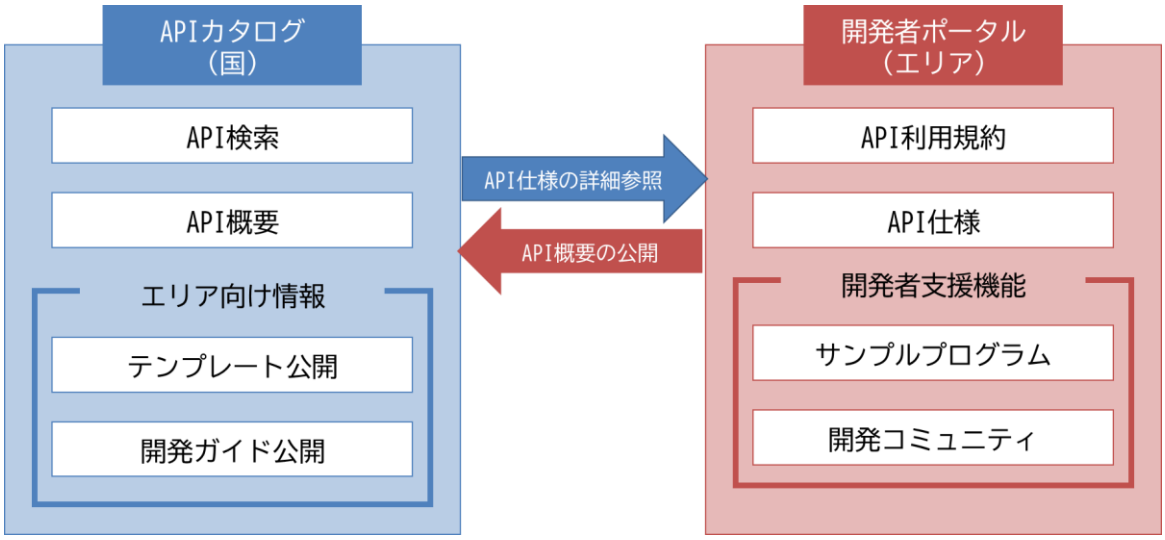


図 6-4-1-3　API カタログと開発者ポータルのイメージ（再掲）

また、API カタログと開発者ポータルの役割は表 6-4-1-3 のとおりである。

表 6-4-1-3　API カタログ/開発者ポータルの役割（再掲）

対象	主体	目的
API カタログ	国	• 先端的サービスの開発の際に、各エリアのデータ連携基盤で提供される API やサービスについての情報が得られること。 • エリアがデータ連携基盤を構築・運用する際に、他エリアの事例や情報を入手できること。 • 各エリアにおけるデータ連携基盤・スーパーシティの実現を促進するために、各エリアがデータ連携基盤及び開発者ポータルの構築に必要なドキュメント・ツール、API 標準仕様等の情報を公開すること。
開発者 ポータル	エリア	• データ連携基盤の利用者が、データ連携基盤を用いた先端的サービスを円滑に開発できること。 • データ連携基盤で実装されている API の詳細情報や利用規約・手順等のドキュメント公開、開発者間で問題解決を図るためのコミュニティ機能等、基盤を有効に利活用できること。

6-4-2.　将来的に発生し得る課題と想定対処法

(1)API カタログ上における API 詳細仕様の公開方法

本調査において API カタログの役割は、各エリアから API 仕様の概要を集約して検索を可能にするものと定義した。しかし今後のスーパーシティ推進に伴い、各省庁等の国の API 仕様の公開や、データ連携基盤を活用する上で共通的に利用できる API 詳細情報の公開等、各エリアから集約した API 以外の詳細な仕様を中央主導で公開するケースが発生することが想定される。

そのような場合において、API カタログ上で API 仕様の詳細まで掲載する場合、各エリアから集約した

API 概要情報と、国として API カタログで提供する API の詳細情報とで利用規約が不明瞭になることや、掲載情報が煩雑になりユーザビリティが低下すること等が懸念される。また、追加開発の際、API カタログ機能と API 詳細仕様公開機能が混在することにより追加開発自体や運用、他部分の改修に係るコスト等の負担が増加することも懸念される。

そのため、中央から API 詳細仕様を公開する必要が発生した際は、エリアが提供する開発者ポータルと同等の開発者ポータルを中央でも作成し、API カタログとは別サイトとして公開・運営することが選択肢のひとつとなり得ると想定する。この際、各エリアの開発者ポータルと中央の API カタログとの関係性と同様に、API カタログ上には API の概要情報のみを登録し、横断検索機能を設けた上で、リンク等による連携が図られることを想定する。

(2)API カタログ利活用促進のための評価機能の検討・搭載

API カタログを各エリアにとって有益な情報の公開場所とするためには、まず各エリアにおけるスーパーシティの取り組み活性化と、データ連携基盤・開発者ポータルの利活用を通じた活発な API 登録・更新が不可欠である。

しかし、エリアやサービサーにとって、API を登録・公開するメリットが十分に確保されない場合、API 登録数が伸び悩み、API カタログとしての利活用も十分に活性化されないことが懸念される。

API カタログにおける API・サービスの登録・公開によってエリアやサービサーが享受できるメリットとしては、既存のサービスや有償 API 等が本カタログ上で公開され、他エリアの目に留まることによって、新たなビジネスのオファーや、自エリアのブランディング等のチャンス創出に繋がることがひとつのインセンティブとして想定される。

本メリットを最大化し他エリアからの関心を集めるためには、より多く利活用され、利用者から高い評価を受けている優秀なサービス・API を、より目に留まる場所に表示できるような評価機能の詳細検討が必要となると想定される。

評価方法や評価値の算出方法については、機能一覧案に記載のとおり将来的に検討が必要となる領域ではあるものの、検索数や利用数といった統計情報に加え、レビューコメントを記入や食べログ等のレビューサイトで見られる 5 つ星評価、Twitter 等の SNS で見られる「いいね」ボタン等が代表的な評価方法として想定される。

(3)API カタログの名称検討

本事業の中で検討した結果として、中央が整備する API カタログは単に API の情報を横断的に公開するカタログとしての役割だけでなく、データ連携基盤に関する各種情報を展開する場としてや、他エリアとの事例情報共有やコミュニティ形成等、エリア間のハブとしての役割を担うことが相応しいと結論付けられる。最終的に実装される機能実態に即した名称を検討することで変更することで、構築後の利活用普及やより一層の知名度向上の観点に寄与できると想定する。

(4)機能一覧案で定義された情報以外の公開

本調査事業で検討した開発者ポータルの機能一覧案は、あくまで全エリア共通で実装されるべき必須機能と、エリアの判断で実証要否を検討されるべき推奨機能を整理したものとなっている。スーパーシティの取り組み推進過程においては、データ連携基盤の利活用を通じて解決されるべき課題の分野や深刻度、及びそこに参画するステークホルダの特徴に大きな差異があると想定され、本機能一覧案だけでは全てのエリア・全てのケースに対応できない可能性が存在する。特に API 詳細仕様の公開においては、5 調査結果（イ）①API 共通ルール/標準仕様の検討内容も考慮の上、OpenAPI 仕様のドキュメントを公開するケースを前提としている。しかし、スーパーシティ実現に向けた異なる分野間の多様なデータ連携の実現

においては、センサーデータの収集の際等で利用する IoT デバイス関連のインターフェース（MQTT 形式等）の取り扱い等、必ずしも統一仕様に合致しないケースが発生する可能性があると想定される。また、開発者ポータルでは API 詳細情報の公開を主たる役割と絞って検討しているが、エリアによってはデータカタログを併設したいといった今回検討した範囲外のコンテンツの追加要望が発生する可能性がある。

そのため、今後の検討では、機能一覧案の定義や必須機能の整理を行いつつも、エリアごとに機能を検討しコンテンツを拡充できるような仕組みを検討する必要がある。

(5)ユーザからの登録を受付ける場合の内容承認

開発者ポータルでは、ユーザ登録や API 登録等、不特定多数から登録が行われることを想定した機能を将来機能として定義している。しかし、ユーザの身元を確認せずに各種情報の登録機能の利用を可能とした場合、不正な API やサービスの公開や利用、不適切なユーザ登録等のリスクが発生する。

そのため、API 登録やユーザ登録等の際に審査後に承認を行うことで、リスクを低減することが可能である。API 登録やユーザ登録等の承認については、主にデータ連携基盤が認証機能を保有しているため、利用規約の選定や各種登録事項の承認等の各種判断基準を選定することが想定される。開発者ポータルではこれに従い、ユーザ登録や API 登録等の承認を行うと想定する。審査プロセスや基準・指針等のサンプルを中央から各エリアに対してガイドラインとして展開することで、不適切な利用を防止するために最低限必要な項目が漏れなく統一的に導入されると想定する。

(6)API カタログ/開発者ポータルの非機能要件

本章では、中央が整備すべき機能と各エリアで整備されるべき機能の住み分けに関する提言を目的とし、調査を行った。今後、詳細な実装方式や利用技術の選定を行い、API カタログ/開発者ポータルの非機能要件の検討が進められる必要がある。非機能要件の検討にあたっては、独立行政法人情報処理推進機構が定める非機能要求グレード[44]等を参考にし、運用スケジュールや稼働率、同時接続数等についての定が想定される。特に各エリアで整備する開発者ポータルについては、機能要件と同様にエリアごとの特徴に合わせながら、最低限必要となる非機能要件を策定できるよう、中央からガイドライン等を公開する必要がある。

[44] 非機能要求グレード https://www.ipa.go.jp/sec/softwareengineering/std/ent03-b.html

7. 調査結果（ウ）データモデルの方針

7-1. この文書について

本章[45]は、スーパーシティにおいて様々なサービスを実現する上で用いられるデータモデルについて、その取り扱いや作成の方針を示す。

なお、データモデルとは、データの項目、形式、構造、複数のデータ間の関係性等を示す仕様である[46]。

7-1-1. 位置付け

本書は、「スーパーシティのデータ連携基盤に関する調査業務」の趣旨に従い、データモデルの基本的な考え方とともに、具体的な推奨事例を示した。2021 年度以降各エリアがデータ連携基盤の整備を行っていく際、本書の示す推奨事例に沿ったデータモデル（以下、本データモデル）を参照、活用することにより、異なるスマートシティ間での相互運用性[47]を確保することを期待している。

なお、既に独自のデータモデルを使っている場合に、必ずしもこのデータモデルに変換することを強制するものではない。とはいえ、独自のデータモデルを使用している時にも、様々なサービス、サービスを実現するシステムやエリアを横断してデータ連携を行う際にこのデータモデルを参照することにより、連携に必要な変換等のデータ処理などが容易になることを期待している。

7-1-2. 想定読者

本書はスーパーシティのデータ連携に取り組む自治体の職員及び委託先、スーパーシティにおいてサービスを開発・提供する事業者やデータを提供する事業者等を対象読者とした。本書の読者には、データ処理、データ利活用に関する一定の知識、理解があることを前提としている。とはいえ、これからスーパーシティの構築や運営に取り組む読者にもわかりやすいように、可能な限り専門的な用語を避けた解説を試みた。

7-1-3. 想定する本書の利用場面

スーパーシティ、スマートシティにおける新たな価値をもたらすサービスの実現に必要なデータ連携の仕組みを検討、設計、構築、運用する際には、連携させるデータモデルを明らかにする必要がある。本書の作成にあたっては、どのようなサービス、仕組みであるかに関わらず、データについて共通認識を持つために、参加者や関係者が共通のデータモデルを参照・参考とすることが望ましいと考えた。

現実的には、新たにスーパーシティのデータ連携に取り組む場合であっても、事業者等がデータモデルそのものを新規に設計し、データ収集・蓄積から取り組むケースは少ない。多くの場合、特定の目的に沿って構築・運用されてきた既存システムのデータを再利用すると考えられる。このため本書では、スーパーシティ等における新規システム構築を前提とせず、既存データの相互運用性を高める取り組みにおいて活用可能なデータモデルの考え方と参照モデル、またどのデータセットであってもほぼ必要であろう基本データ項目について、その記述ルールを示すことを第一とした。

スーパーシティ、スマートシティの取り組みだけでなく、オープンデータのデータモデル設計や見直しにおいても、本書で示すデータモデルが参照されることを期待する。また、政府が進めるデータ戦略、ベース・レジストリにおいて参照されることにより、中長期的に相互運用性の高いデータの生成と活用に進み、社会全体の生産性と価値向上に寄与することも期待する。

なお、本章の内容は、いくつかの利用場面を想定してデータモデルの在り方を検討した結果を例示する

---

[45] 独立した文書「スーパーシティのデータモデルについて」（仮名）として取り扱われることを想定し、以下では「本書」と記載する。

[46] 7-1-6(5)参照。

[47] 7-1-6(9)参照。

ものであり、スーパーシティ、スマートシティにおけるあらゆる利用場面に適用できる完全性を保証するものではない。そのため、実際にスーパーシティ、スマートシティのサービスを検討、設計、構築、運用する際には、7-8 に示したスーパーシティ/スマートシティで展開され得るサービスのユースケースへのデータモデルの適用事例を含め本章の内容を参照しながら、より汎用的で拡張性のあるデータモデルの設計を検討されたい。

7-1-4. アーキテクチャや API との関係性

データモデルを整備しても、それを全体サービスの中でどのように使うのか、どのような制約条件があるか等の全体像の整理をしていく必要がある。全体像の中で本データモデルは、Society 5.0 の一環で整備されている「スマートシティリファレンスアーキテクチャ　ホワイトペーパー」[48]のデータ整備部分を担う。また、このホワイトペーパーに示されている「データを利活用するための API」は「スーパーシティ/スマートシティの相互運用性の確保等に関する検討会　最終報告書」[49]の考え方を参照し、報告書の第 5 章で定義される。

7-1-5. 対象範囲

「スマートシティリファレンスアーキテクチャ　ホワイトペーパー」では、「動的・静的データ」「地理空間データ」「パーソナルデータ」など多くの種類のデータを参考に示しているが、これら全てのデータモデルを一斉に整備することは困難である。そこで、本書では、地理空間データの基盤となるデータを対象とし優先的に着手した[50]。

なお、地理空間データについては別途調査を行い、考慮すべき事項について 7-9 にまとめた。また、別紙ウ-3 として調査結果報告書「地理空間情報の既存規格調査及び既存データモデル調査報告書」を添付した。地理空間データの基礎は地図（ベースマップ）であるが、地図データの形式は用途により異なるものが選ばれることも多い。そこで、今回のデータモデルでは、特定のデータ形式を対象とせず、付録で整理を行うこととした。

地理空間データ以外の「動的・静的データ」「パーソナルデータ」については、現時点で公式な定義がなされていないため、データモデルの対象テーマとしてデータ項目や要素を特定することが困難であることから、今後の検討対象とした。

本データモデルは、スマートシティにおける各種サービスで用いる地物（ちぶつ）と移動体を中心に検討し、将来、本データモデルを行政情報等へと展開を図っていくよう改版を予定している。また、本データモデルのデータ項目を定義するだけでは、実際の実装には十分でない点も多い。そこで、できる限り既存の参照情報を掲載することにより、スーパーシティやスマートシティ関係者のデータモデルの設計や活用の実装が容易になるように配慮を行った。

なお、データモデルを構成するデータ項目は、国内外の主要なスマートシティのデータモデルの中で共通的に使われている項目、国内でニーズの高い項目を、有識者ヒアリングを踏まえて選定した。

7-1-6. 用語定義

本書において文中で使用する用語を整理しておく。

---

[48] https://www8.cao.go.jp/cstp/stmain/20200318siparchitecture.html

[49] https://www.kantei.go.jp/jp/singi/tiiki/kokusentoc/supercity/pdf/sogowg_houkokusyo.pdf

[50] 都市計画やハザードマップなどの目的別地図は、一般に自治体ごとに作られるため、特定の対象範囲に限定されることが多い。また、その目的により様々なデータを階層的に重ねて作成する利用形態が多いことから、地図データに重ね合わせることを前提としたデータについて記載した。

(1)データ

データは、JIS X 0001 に「情報の表現であって、伝達、解釈又は処理に適するように形式化され、再度情報として解釈できるもの」と定義されている。また「データに対する処理は、人間が行ってもよいし、自動的手段で行ってもよい」とされている。

すなわち、データは、情報を構成する要素を示している。

(2)データセット

データセットとは、複数のデータから構成されるデータの集合である。一般に名前と年齢などのように、複数のデータとデータの属性などを示すメタデータなどから構成され、一群で取り扱われるデータの集合をデータセットという。

(3)情報

情報は、ISO/IEC 2382-1、JIS X 0001 で「事実、事象、事物、過程、着想及び概念により構成され、対象物に対して一定の文脈中で特定の意味をもつもの」[51]と定義されている。

情報には、(1)データで定義されているとおり、情報の表現を構成する要素であるデータやデータセットを含むことがある。データ連携、データ利活用を前提とするスーパーシティにおいて情報は、「データセットと付帯情報（事実、事象、事物、過程、着想）及び概念により構成され、対象物に対して一定の文脈中で特定の意味をもつもの」といえる。

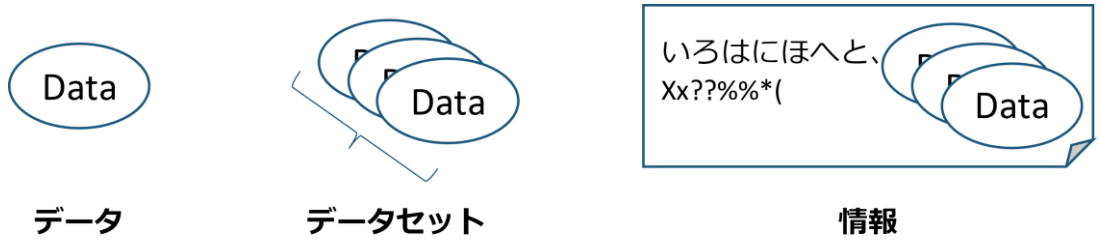


図 7-1-6-1　データ、データセット、情報の関係

(4)データセットの構成要素

データセットは、複数のデータから構成されるデータの集合であり、その取り扱い方法、実装により異なるが、基本的な構成と構成要素について「パーソナルデータリファレンスアーキテクチャ　DFFT(DATA FREE FLOW WITH TRUST) 実現のためのアーキテクチャ設計と国際標準化推進の研究開発」[52]と同様に、表形式のデータセットを事例に用いて以下に定義を示す。

[51] knowledge concerning objects, such as facts, events, things, processes, or ideas, including concepts, that within a certain context has a particular meaning

[52] 2019 年度、内閣府が実施し国立研究開発法人　新エネルギー・産業技術総合開発機構（NEDO）が管理法人を務める「戦略的イノベーション創造プログラム（SIP）第 2 期／ビッグデータ・AI を活用したサイバー空間基盤技術／パーソナルデータアーキテクチャ構築」事業の成果のひとつ。本書の説明は同文書に基づき一部を改変している。

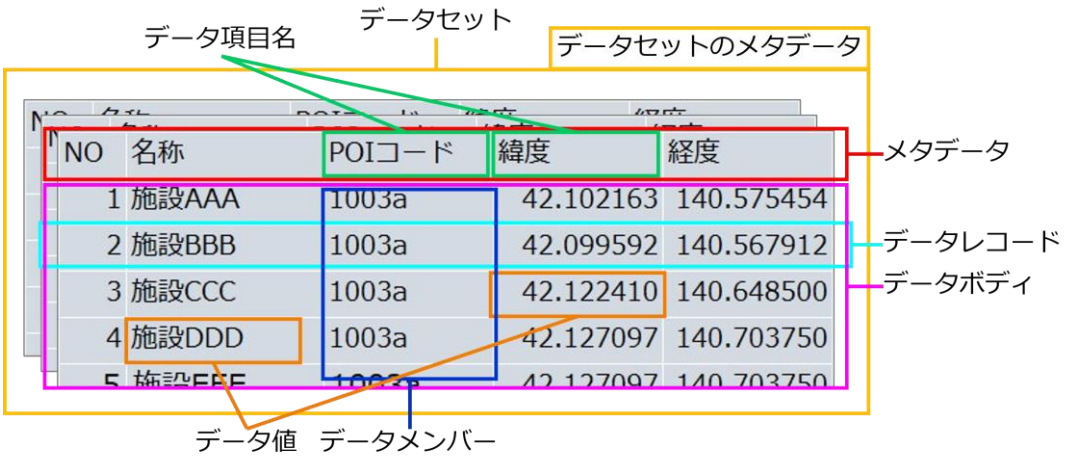


図 7-1-6-2　表形式データセットの構成要素

表 7-1-6-1　代表的な用語の一覧

用語	英語表記	本書での定義	出典	出典における定義
データボディ	Data Body	1 以上のデータの集合でメタデータを含まない。		
メタデータ	Meta Data	データのうち、データの属性などを示すデータ。	ISO/IEC 11179-3:2013, 3.2.74	data that defines and describes other data
データ値	Data Value	個々のデータの持つ値。	ISO/IEC 25000:2005	content of data item
データメンバ	Data Member	同一のメタデータに紐づくデータの集合。		
データレコード	Data Record	共通の識別子により 1 以上のデータ値が何らかの関連性によってまとめられた集合。	ISO 18739:2016(en), 3.1.13	one or more data items treated as a unit within a data set
データ項目名	Data Label	同一の計測、観測方法によるデータ値の集合を示す名称。		
データセット	Data Set	データボディとメタデータの集合。データセット自体もメタデータを持つ。	ISO 8000-2:2018, 3.2.4	logically meaningful group of data

図 7-1-6-2 のように、データセットには、計測値、観測値などのデータ値の集合だけでなく、データの属性などを示すメタデータが含まれる。また、データセットの全体に対する属性情報などであるデータセットのメタデータが含まれる場合がある。ここで、メタデータとは、データのうち、データの属性などを示すデータである。なお、データセットは、その取り扱いの形態により、これらの各要素を必ずしも同時に取り扱うとは限らない。例えば IoT 機器などのストリーミングソースによって継続的に生成されるデータなどでは、その伝送時には、メタデータを含まない場合もある。

なお、同一の計測、観測方法によるデータ値の集合は、その総称であるデータ項目名[53]と構成要素であるデータメンバから構成される。

また、データレコードは、共通の識別子（時刻やIDなど）により関連付けられた1以上のデータ値により構成される。

なお、実装においてはシステム上のデータ要件定義等で利用する「データ項目名」と、人間にとって使いやすい表記である「ラベル」は、同じ場合と異なる場合がある[54]。

(5)データモデル

ISO/IECでは、「データモデル(data model)」を“The numeric format in which the TIM[55] shall output or accept data.”[56]、「概念データモデル(conceptual data model)」を“data model that represents an abstract view of the real world”[57]と定義している。IMI情報共有基盤[58]では、「語彙やデータセットなどを機械処理するため、データ項目を構造化しその属性や値の制限などを定義したもの」と定義している。データマネジメント知識体系ガイド第二版(DMBOK)[59]では「モデルはドキュメントの一形式であり、モデリングプロセスで捉えられたデータ要件とデータ定義を表現する。データモデルはデータ要件を伝えるための主要な手段であり、業務からITへ、ITからITへ、アナリスト、モデラー、アーキテクトからデータベースデザイナーや開発者へ、データ要件を伝える際に使われる」と説明している。また、データモデルには、あるデータと他のデータの関係やデータの構造を表現する定義も含まれる。よって本書のデータモデルを参照することは、データの構造や意味を人間やデータを処理する機械に伝達し、相互運用性を確保するための手段と考えて良い。

様々なデータモデルの記述レベルのうち、本書では「データ項目名」とその「データ項目の説明」までのシンプルなデータ項目定義を対象とする[60]。7-3-3に記載のとおり、スーパーシティのデータ連携基盤には、誰でもどこでも何にでも活用できるオープンさが求められるため、特定のシステム仕様や具体的な利用場面を限定しすぎることは望ましくない。複雑な設計によってデータ項目の定義がシステムごとにばらばらに乱立することを避け、今後バージョンアップを重ねる前提で、まずはスーパーシティデータ連携基盤の基本となるデータモデルが共有されることを目指す。

---

[53] データ項目名は、メタデータの構成要素のうち、データ項目を識別するための名称だけを示す。メタデータには、7-1-6(7)で説明するとおり、データの属性などをより広範囲に示す役割を期待される。

[54] 例えばシステム仕様で利用可能なデータ項目名が半角英数字やコードのみに限定されているといったケースは少なくない。

[55] transducer interface module

[56] ISO/IEC/IEEE 21450:2010(en), Information technology - Smart transducer interface for sensors and actuators - Common functions, communication protocols, and Transducer Electronic Data Sheet (TEDS) formats, 3.1.8。

[57] ISO/IEC 11179-1:2015(en), Information technology - Metadata registries (MDR) - Part 1: Framework, 3.2.5。

[58] 共通語彙基盤と文字情報基盤からなり、行政データの相互運用性向上を図る。共通語彙基盤は、データに用いる様々な用語の表記、意味、構造の特徴を抽出し、体系的に整理した上で、分野を超えてデータの検索性向上やシステム連携強化を実現する。文字情報基盤は、行政で用いられる人名漢字等約6万文字の漢字を整備し、外字作成等のコストを解消した(2017年12月に国際規格化)。共通語彙基盤の取り組みはhttps://imi.go.jpを参照。

[59] 米国DAMAインターナショナルが編纂したデータマネジメントに関する知識体系。

[60] ただし、IMIやSchema.orgなど既存データモデルにおいて明確に定義されているデータ構造等が存在する場合は適宜提示する。

(6)データ連携

生成者、管理者や物理的な保管場所、システムに限定されず異なるアプリケーションやシステムをまたいでデータを共有、活用するために、異なる組織やサービが保有するデータセットが、一定の規則に基づき照合、補完、変換などを行い、新たな用途等に活用できる状態を、本書では「データ連携」という。

(7)メタデータ

メタデータは、(4)で説明したとおり、データのうち、データの属性などを示すデータである。ISO/IEC 11179 は、メタデータを「やり取りの対象となる何らかのデータセットを構成するデータ項目について、それらの意味や取りうる値、データ項目間の関係（構造）を定義した情報を指すもの」と定義している。本書では、データ提供者においてデータやデータセットが適切に管理され、データ提供先にその存在を認知、検索、発見され、アクセスされ、活用されるために、データやデータセットへ付与される情報としてメタデータを扱う。データセットに内包される個々のデータ項目を対象としたメタデータ定義の塊はデータモデルを構成し、データセットには内包されない、データセットを対象としたメタデータ定義の塊はデータカタログを構成する。

表 7-1-6-2　メタデータの付与対象による区分イメージ

付与対象	メタデータの塊	メタデータ項目例	主な利用場面
データセット	データカタログ	• 情報の管理者 • 作成日/更新日 • 利用条件　など	データセットを取得する前（データセットを検索する時等）
データセットに内包される個々のデータ項目	データモデル	• 基本属性 • 所在 • 用途 • 利用方法　など	データセットを取得し値を活用する時

スーパーシティのデータ連携の文脈において、データセットは、データモデルに基づき記述されたレコードの塊と、それらをグループ化し構造化した一塊のデータセット総体を処理しやすくするためのデータモデル、データを信頼して活用できるための利用条件等を含むメタデータから構成されている必要がある。ただし、メタデータの実装方法は複数存在し、サービス設計やサービスの実現に必要なシステム、データ処理等、それらに伴うコスト等により異なり、現時点で本書が標準を示すのは尚早である。したがって本書ではメタデータの実装方法は論じない。

(8)データカタログ

データセットを対象に記述された(7)メタデータの値の塊であるデータセットを、本書では「データカタログ」という。データカタログは、データモデルだけでは表現しきれないデータに関する情報を伝える機能を持つ。データカタログは、誰(どのシステム)が=Who、いつ=When、どこで=Where、誰(どのシステム)へ向けて=Whom、どのような目的・理由で=Why、どのようなものを=What、どのようなツールなどを用いて、どの基準を参照して、どう運用されている=How データセットなのかというメタデータから構成される。また、スーパーシティ内外でデータが活用されるためには、ライセンスや価格、提供先の制約などの利用条件を示すデータカタログが不可欠である。データカタログの既存データモデルには W3C DCAT 及び DTA データカタログ作成ガイドライン V1.1 をはじめとする汎用的なデータモデル、また地理空間に特化した geoDCAT などが存在する。

(9)データの相互運用性

JIS X 0001:1994 情報処理用語の定義は「それぞれの機能単位に固有な特性に関する知識を利用者がほとんど又は全く必要とせずに、各機能単位が互いに通信し、プログラムを実行し又はデータを転送する能力」、ISO/IEC 19941:2017 Information technology — Cloud computing — Interoperability and portability の定義は「Ability of two or more systems to exchange information and mutually use the information that has been exchanged.」である。本書では、「データの相互運用性向上のためのガイド」[61]の定義を踏まえ、組織や組織が運用するシステムを超えて、全ての参加者からそのデータに対する共通理解を得られる状態を指すこととする。

[61] 独立行政法人 情報処理推進機構「データの相互運用性向上のためのガイド」https://www.ipa.go.jp/ikc/our_activities/dt_data_guide01.html

7-2. データモデルの役割

7-2-1. 一般論

データモデルは、対象となる現実世界をデータとして抽象化し、その関係や構造を特定の表現形式で記述したものであり、人間にわかりやすくかつコンピュータ処理にも適したモデルが望ましいとされる。データモデルの設計には様々な手法や事例があるが、ここでは基本的な考え方を示す。

ある財務会計システムに、次のような形式のデータ(項目名と値)があったとする。

売上高	1,000,000

項目名は「売上高」で、値が「1,000,000」という数字だが、これだけを見ても、値の意味を正確には理解できない。どの企業の何の製品/サービスの売上高なのか不明である。「売上高」なので金額であろうと推察はできるが、定かではない。金額だったとしても 1,000,000「円」なのか、1,000,000「千円」なのか桁数がわからないし、1,000,000「ドル」かもしれない。また、いつの売上高なのかもわからない。

このようなデータを正しく理解し適切に利活用できるようにするためには、以下のように関連する複数のデータをまとめて構造化し、データが持つデータ項目や値の意味を定義する必要がある。

会社名	株式会社○○○○
会計年度	2020 年度
製品名	ABC
売上高	1,000,000
単位	円

金額項目
IFRS(国際会計基準)の売上高

この場合には株式会社○○○○社における製品 ABC の 2020 年度の売上高 1,000,000 円(IFRS に準拠)であることが読み取れる。

データモデルは、このように複数のデータ項目を関連付けてグループ化・構造化し、各データの値の意味を誰もが正確に解釈できるようにする役割を持つ。このようなデータ項目を含む構造によってある用語が定義されていれば、ラベルが「売上高」であったり「Sales」であったり、他の外国語表記になったりしていても、人が理解できるだけでなく、コンピュータもその中に含まれる値を同一のデータとして処理することができる。

また、データをシステム間で連携する場合には、そのデータの持つ意味と併せて、データの型(数値型か、文字列型か、あるいは「あり、なし」を示す識別子かなど)、桁数、単位なども定義して、その定義情報を開示・共有しなければ、有益な情報として他者がデータをサービスや事業などに活用することは難しい。データの共通理解を実現するためには、データの値と併せて、その意味・型・桁数・単位などを定義したデータモデルを組にして相手に伝える必要がある。

7-2-2. ユースケースから見たデータモデル

ここで、都市において今後どのようなデータモデルが必要となるのかを考察するにあたり、具体的なユースケースを想定してみたい。一例として、河川の氾濫など気象災害が差し迫っている時に、被災予想地域の病院に入院している怪我人や病人を避難させるために、病院から委託された民間の救急事業者やタクシー会社が、空きベッドのある近隣自治体の医療機関や避難所を探して予約し患者を搬送する、といったユースケースを考えてみる。具体的には、スマートフォンの専用アプリや PC の Web ブラウザを使って、民間救急事業者やタクシー会社の担当者が、施設検索予約システムにアクセスし、避難対象患者の人数、

病気やケガの種類、搬送先で必要とする医療設備や医療ケアサービスの内容、搬送元の出発地点の位置情報、搬送する時間数や距離の上限、搬送希望日時などを入力して、受け入れ可能な医療機関や避難所を検索する場合を想定する。

この場合、医療施設検索予約システムは、近隣の複数の自治体の施設情報データベースを横断的に検索して、入力された条件に合致する空き施設の情報を、利用者の希望条件に合致する度合いが高い順に並べて、検索結果データとして出力・表示する必要がある。その際、検索結果を構成する各データを、統一された配列で並べ、共通の項目名を付けて表示するためには、それぞれのデータの値がどのような意味・型・桁数・単位のデータであるかを示すデータモデルの定義情報を併せて共有する必要がある。さらに、いつ誰がどのシステムを使ってどんなデータを検索し利用したかを履歴として管理するために、送信日時、データを送信したシステムの ID、受信したシステムの ID などもやり取りする必要がある。

上記のようなことを考慮して、災害時に患者の避難先を探す施設検索予約のユースケースにおける検索結果出力用のデータモデルを例示すると、図のようになる。

（実際には、ID や番号などが入ったデータの値と、そのデータの配列順序や項目名や意味などを定義したデータモデルとが、別々に送受信されるものと想定されるが、ここでは読者にわかりやすいように便宜上それらを分けずにまとめて併記する。）

```
データモデル情報モデルの種類＝検索結果
        送信日時＝YYYY/MM/DD/HH:MM:SS（JST）
        送信元システム ID＝SSSSS(SXXXXX)
        送信先システム ID＝RRRRR(RXXXXX)
        検索種類＝空き施設確認
        検索条件

                検索対象レベル 1＝施設
                        検索対象レベル 2＝総合病院 or 専門病院 or 避難所 or 公民館
                地理条件
                        起点となる場所＝緯度経度（患者が入院している病院の住所）
                        直線距離＝100km 以内
                        車両での移動時間＝3 時間以内
                        標高＝海抜 5m 以上
                        安全基準
                                日本国○○市ハザードマップ
                                        洪水リスクレベル＝0
                利用予定日時
                        利用開始予定日時＝2021/04/03 05:00～
                        利用終了予定日時＝未定
                利用サービス
                        冷房＝要
                        暖房＝要
                        食事＝朝昼夕（部屋出し）
                        医療ケア＝要
                                看護＝要
                                点滴＝要
                                投薬＝要
                                人工透析＝要
                                医療用ベッド＝要
                                アレルギー除去食＝要
                                個室＝不要
                        介護ケア＝要
                                要介護度基準＝日本国法定要介護度
                                        要介護レベル＝1
                        停電時非常用電源設備＝有
                        入浴＝不要
                        言語＝日本語
                        健康保険適用＝要
```

利用希望者数
男性＝2
女性＝1
その他＝0
幼児＝0
乳児＝0　　（上記の検索条件と同一）

条件合致施設一覧
施設 1
施設 ID＝9999999999
施設情報 URL＝https://xxxxxxxxxxxxxxxxxxxx
利用可能人数
男性＝2
女性＝1
位置
距離＝XXkm
予想移動時間＝99 時間
料金
通貨＝日本円
単位＝1 日あたり
金額＝10,000
施設 2
施設 ID＝9999999999
施設情報 URL＝https://xxxxxxxxxxxxxxxxxxxx
利用可能人数
男女＝1
位置
距離＝XXkm
予想移動時間＝99 時間
料金
通貨＝日本円
単位＝1 日あたり
金額＝9,000

図 7-2-2-1　情報検索システムから出力される回答結果のデータモデルの例

このデータモデルの例を見るとわかるように、現代の都市には様々な種類の施設・設備・サービス・人・マネーなどが混在するため、それらを正確にデータ化して複数のシステム間で相互運用し利活用するためには、それぞれの対象物の情報を表す異なるデータ項目を用途に合わせて柔軟に階層的に組み合わせて構造化・体系化し、そのデータモデルをシステム間で共有できるようにする必要があると考えられる。

7-2-3.　データモデルと相互運用性

データモデルを用いた相互運用性の考え方は図 7-2-3-1 で整理した。どれが正解ということはなく、システムや組織を超えてデータを活用するステークホルダがビジネス目的や費用対効果を考慮し合意しやすい手段が選択されている。「①個別にすり合わせ」は社内システムや特定の業界などステークホルダが固定されている場合に、精緻なレベルでデータの意味を共有しやすい。従来のシステム連携やデータ連携では一般的にこの手法が採用されてきたと考えられるが、多様なデータ源で生成・処理されるデータを連携させるスーパーシティにおいては、自由度の低さが課題になる。

「②モデルを統一」は全ての領域に適用することは難しいが、新たにデータセットを整備したり、EC サイトで定型フォームへ購入情報を登録してもらったりするように、値に制約をかけられる場合はデータ

の質を確保しやすい。OPC UA（IEC62541）[62]やXBRL[63]はこれに近い。全て統一ではなくレイヤーによってローカル定義が許容されている。「推奨データセット[64]」や「中間標準レイアウト[65]」といった自治体の共通フォーマットとして適用しやすいと考えられるが、既存データセットが持つ値を記載するデータ項目がない場合などの対応方法に留意が求められる[66]。

「③参照モデルを共有する」手法は、各システムのデータモデル定義はそのままに、参照モデルとの違いや共通点を明らかにすることでデータの相互運用性を確保しようとする。IMI 共通語彙基盤やNIEM[67]、Data Transfer Project[68]等で採用されている考え方である。表現方法を揃える手法は汎用性の高い概念で意味を伝えようとするため、「個別にすり合わせ」と比べると精緻な定義は困難だが、幅広い分野に跨る相互運用での現実解と考えられる。

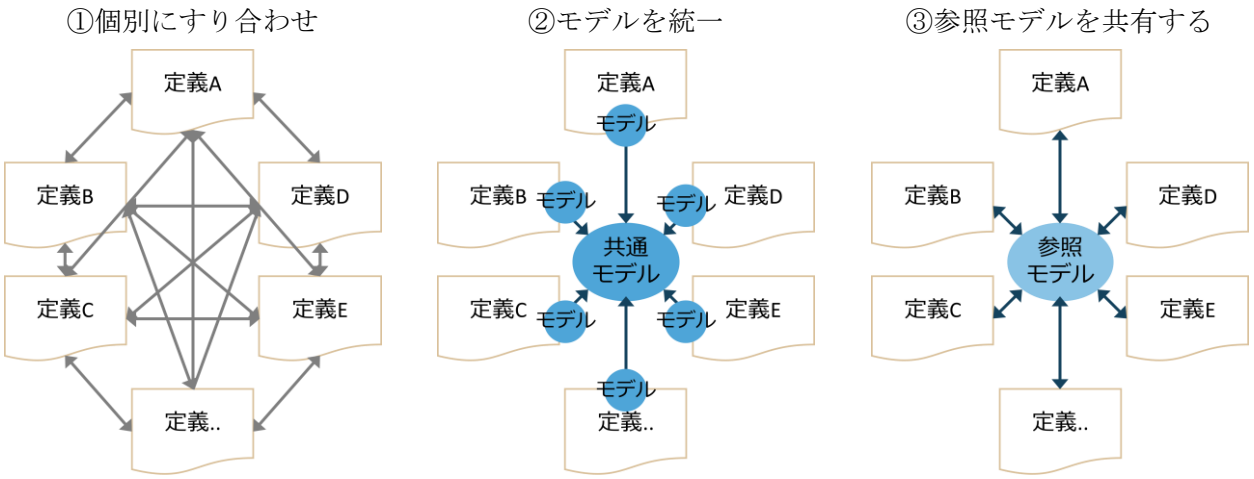


図 7-2-3-1　相互運用性の考え方

[62] 産業オートメーションなどの業界で、安全で信頼性あるデータ交換を行うために策定されたオープンな国際標準規格。OPC Foundation が 2008 年に発表し IEC62541 として国際標準化された。

[63] eXtensibleeXtensible Business Reporting Language。各種財務報告に用いられる情報を作成、流通、利用できるように標準化された XML ベースの言語で、国内外の監督官庁、投資家などが企業等の財務チェックや分析で利用されている。

[64] オープンデータ公開とその利活用促進目的に、政府として公開を推奨するデータセット。
https://cio.go.jp/policy-opendata

[65] 総務省が公開する、自治体業務システムの円滑なデータ移行のために移行データの項目名称及びデータ型、桁数、その他の属性情報等を標準的な形式として定めたデータ移行用の仕様。
https://www.j-lis.go.jp/rdd/jititaicloud/standard_layout/standard_layout.html

[66] 例えば、指定緊急避難場所のデータセットに備品のデータ項目が定義されていない場合、「バール」などのデータ値があっても、記載を諦めるか、備考欄に記載するといった対応が散見される。

[67] National Information Exchange Model。米連邦政府におけるデータ連携標準で、IMI 情報共有基盤が構築時参考にした標準である。政府機関や州ごとに異なる業務や語彙を連携して組織の壁を超えてデータを活用するためのスキーマを整備している。https://www.niem.gov/

[68] Apple、Google、Facebook、Microsoft、Twitter が取り組む各社サービス間でのポータビリティ実現へ向けた枠組み。https://datatransferproject.dev/

7-2-4. スーパーシティにおけるデータモデルの役割

スーパーシティのデータは、特定の自治体の 1 つの情報システムの中だけに閉じて利用されるのではなく、他の自治体や企業・団体、居住者、来街者、さらには海外の政府・自治体や企業・団体が管理する情報システムに受け渡され、異なる情報源から得た複数のデータを組み合わせたり集計したり加工したりして、様々な用途に使われる。そのため、それらのデータを扱う全ての情報システムが、それぞれのデータの形式や意味を正しく認識できるように、一定の共通ルールを定め、各システムがそのルールを共有しておく必要がある。

また、スーパーシティにおいてデータを生成するデバイスや装置や情報システムには多種多様な新旧の種類のものがあり、データの形式や構造や単位などが必ずしも全て統一されていない。それらの異なる種類のデータをサービスや情報システム間で連携、流通させ適切に利活用できるようにするためには、各データを活用可能なパターンに変換して受け渡し正しく認識させる必要がある。その手法は 7-2-3 のとおり複数考えられるが、いずれにせよスーパーシティのサービスを提供する各システムそれぞれがデータを適切に処理できる必要があり、データをそれらシステムの仕様に合致する形式に変換するルールの設計が欠かせない。スーパーシティにおけるデータモデルは、このルール設計の際必要な連携データの仕様を伝える役割を担う。

さらに、スーパーシティでは、土地や建物、店舗、土木構造物、機械、車両、人、エネルギー、催事イベントなど、種類が異なるものを管理するため、管理対象の種類によって取り扱うデータの形式や構造が異なる。システム間でデータを受け渡して利活用するためには、各データがどのような形式や構造で記述されているのかを全てのシステムが共通して認識する必要がある。

他方、今後、技術の進化や事業モデルの変遷、法規制の変化などによって施設・設備・サービスの種類や管理すべき属性が増えたり、社会通念や倫理観の変化によって人や組織を表す属性が追加されたり、AI の普及によってデータ利活用のユースケースが増えたりすることも想定される。したがって、あらゆるパターンに対応できる完全で万能なデータモデルを、ある時点で全て網羅し一律に定義することは困難である。そのため、スーパーシティ、スマートシティのデータ連携基盤を実際に設計・構築・運用する際には、既存のデータモデルに適合させるだけでなく、将来にわたり継続的に、新しいデータモデルを追加したり、複数のデータモデルを階層的に構造化する組み合わせ方を追加変更したりできるように、なるべく柔軟で拡張性のある技術や方式を採用することが望まれる。

また、そのようにして将来継続的に更新されていくデータモデルの定義が、自治体やシステムごとに異なる仕様になってしまうと、共有・連携されるデータの意味を相互に正しく解釈することができずデータの利活用が円滑に行えなくなる。そうした事態を避けるため、データをシステム間で共有・連携させる際のデータモデルの定義情報を全国の自治体で共有できるようにしておき、全てのシステムがその最新のデータモデル定義情報を参照してデータを送受信し利活用できるようにするのが理想である。

このような理由から、スーパーシティのデータを利活用するためには、共通のルールに基づいてデータの構造や形式を体系的網羅的に定めたデータモデルを定義し、そのデータモデルをあらゆる情報システム間で共有できるようにすることが望ましい（②モデルを統一）。

ただし処理・活用可能なデータやデータを扱う技術は日々進化すること、またスーパーシティでは新旧多様なステークホルダが関係することから、あらゆるシステムが同一データモデルを共有できるとは限らない。実装にあたっては、少なくとも各データセットの持つデータ項目や値の根拠をデータモデルとして定義し公開しておくことにより、データを活用したサービスやシステムを担う他のステークホルダが、データの変換ルールを設計しやすくなる。

他方、7-2-3 のとおりデータモデル自体を共通の標準として管理運用する取り組みも進んでいる。スーパーシティはスタート段階にあり、各地域の自発的な取り組みに委ねられているため、政策のトップダウンや特定のベンダが主導権を独占することは望ましくないものの、実際のスーパーシティのニーズに対

応しながら多様なステークホルダの関与のもとで相互運用性の高い標準が形成され持続されていくと期待される。そうした取り組みには参加のインセンティブや持続可能な運用体制が不可欠である。したがって政策には、各スーパーシティやその参加者からフィードバックを得つつ、国・地域やベンダ、業界の違いを超えた国際標準化をリードする責任が求められる。

データモデルの定義にあたっては、国・地域や開発ベンダや業界の違いを超えてデータ連携の相互運用性を確保するため、産業界における Industry4.0 のデータ連携インターフェース規格 OPC UA（IEC62541）のように国際標準化して仕様を継続的に管理・更新し、連携対象となる全ての情報システムがその標準に準拠するよう促す仕組みも重要である。今後、スーパーシティ、スマートシティのデータが国境を越えてグローバルに活用されていく可能性があることも考慮すると、データ連携基盤の設計にあたっては、システム間でデータの共有・連携を行うためのデータモデルの定義情報が国際標準化されることも想定して、国際的にも相互運用性の高いアーキテクチャで設計しておくことが望ましい。

7-2-5.　都市 OS とデータモデル

スマートシティでの活用が想定されるデータは、オープンデータや非識別加工匿名情報、スマートデバイスや IoT 機器などを通じて得られる交通、エネルギー、土地、建物、人流や医療、健康、教育といった行政サービス等に関わる官民データである。個別業務システムで生成されてきたこれらデータの活用を実現するため、①相互運用（つながる）、②データ流通（ながれる）、③拡張容易（つづけられる）を特徴として設計されたのが都市 OS[69]である。都市 OS における相互運用は、都市 OS が提供する API やデータが各種スマートシティサービスや他都市 OS と連携できる状態を指す。

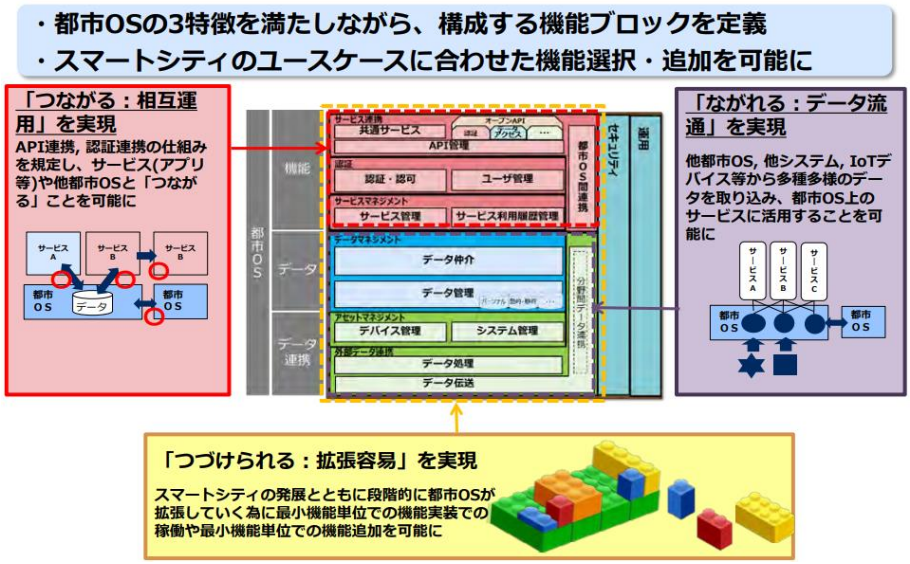


図 7-2-5-1　都市 OS のアーキテクチャ[70]

都市 OS の相互運用とデータ流通を支える基盤のひとつが、政府が整備を進めてきた

[69] 「スマートシティリファレンスアーキテクチャホワイトペーパー」は「スマートシティ実現のために、スマートシティを実現しようとする地域が共通的に活用する機能が集約され、スマートシティで導入する様々な分野のサービスの導入を容易にさせることを実現する IT システムの総称」と定義している。https://www8.cao.go.jp/cstp/stmain/a-whitepaper1_200318.pdf

[70] 戦略的イノベーション創造プログラム（SIP）第 2 期／ビッグデータ・AI を活用したサイバー空間基盤技術スマートシティアーキテクチャ構築「スマートシティアーキテクチャ設計と関連実証研究の推進」

IMI(Infrastructure for Multilayer Interoperability:情報共有基盤)である。IMI は、データに用いる文字や用語の同一性を特徴付ける概念を体系的に分析・整理し、コンピュータ処理に適した定義情報としてデータに付加することで、既存システムに原則手を加えず運用を相互に維持しながら情報の共有や活用を円滑に行う。

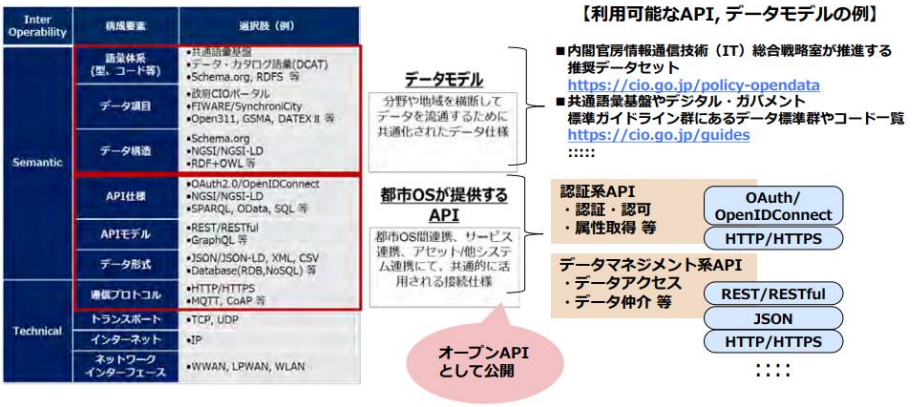


図 7-2-5-2　都市 OS に求められる相互運用性の構成要素[71]

また IMI に対応したデータモデルとして、政府は既に「推奨データセット」や「行政データ連携標準」などを公開し、2018 年現在 47.5%の自治体がこれに沿ったデータを整備・公開している[72]。他方、国土交通省や総務省をはじめ省庁が主導し整備・公開されているデータセットも多数存在する。また、スーパーシティに限らずあまねく流通されるデータについては、Schema.org をはじめとする世界的に認知され利用されているデータモデルが存在する。スーパーシティデータ連携基盤はこれら既存データ資産を積極的に活用すべきであるため、本書では、推奨データセット、国土数値情報、Schema.org の定義が存在する場合はこれらをデータモデル検討の土台としている。

[71] 戦略的イノベーション創造プログラム(SIP)第 2 期／ビッグデータ・AI を活用したサイバー空間基盤技術スマートシティアーキテクチャ構築「スマートシティアーキテクチャ設計と関連実証研究の推進」

[72] https://cio.go.jp/sites/default/files/uploads/documents/survey_results.pdf

7-3. 推奨データモデル概要

7-3-1. 検討の背景と目的

現在、国内で多くのスマートシティ・プロジェクトが行われている。我が国は、もともと正確な交通ネットワークや気象予報、インフラ保守などのスマートシティを構成する基盤が整備されているが、そこに先端センサー、アクチュエータ、高度なネットワークを組み合わせることで新たなサービスを効率的に構築できると期待されている。また、高齢者が多いことから、アクセシブルな都市環境を世界に先駆けて構築できるのではないかとの期待もある。

また、今後は自都市でのスマートシティ化に留まることなく、他都市さらには海外への展開も考えられる。世界の最先端のサービスを自都市に取り入れることも考えられる。

そのためには相互運用性が確保されたスマートシティを実現していく必要がある。スマートシティアーキテクチャの整備や API の検討が進められてきたが、さらにスマートシティの取り組みを進めるために、スマートシティのサービスを実現するドライバーであり血液であるデータを、あらゆる参加者が処理しやすくする必要がある。そのためにはデータモデルの整備が不可欠であり、世界各国でデータモデル整備の取り組みが進められている。

しかし先行事例から明らかなようにデータモデル、データ整備の取り組みは一朝一夕ではなく、10 年 20 年の単位で活用と改善のサイクルを継続する中で初めて「基盤」として機能する。本データモデルは、スーパーシティの中で様々なサービスを効率的に実現するための基盤を目指す第一歩である。このデータモデルを多くの参加者が改善しながら活用することで、サービス提供者がデータ処理の手間を最小化することができる。また、本データモデルは、スマートシティサービスの高度化に専念できる環境を提供するとととともに、移行性の高い都市基盤の実現を目指す。

7-3-2. 推奨データモデルの整理原則

本書におけるデータモデルは以下の原則に基づいて整理する。完全性を目指すものではなく、継続的な改善の対象とすることを原則としている。

(1)参照モデルとしての整備

ここで示すデータモデルは、参照モデルである。本データモデルをそのまま実装してもよいし、データモデルを拡張もしくはサブセットにすることで相互運用性を確保して導入することが可能である。また、高速処理をするために、シンプルなデータモデルで実装し、外部とデータ交換する時に参照モデルにデータモデルに合わせるといった実装もある。

(2)既存データモデルの活用

スーパーシティ/スマートシティだけでなく、既存のデータモデルが存在する場合には、可能な限りそのデータモデルを使用する。既存のデータモデルを使うことにより、先人の知見を活用するとともに既存のサービスとの相互運用性を確保する。ただし、既存モデルの設計思想が古いなどにより、再利用することでデータ流通が円滑に進まないと考えられる場合には、既存のデータモデルと相互運用性をできる形でデータモデルを新たに設計し、異なるデータモデルを相互に変換できる仕組みを導入する場合もある。

特に、現在検討が進められているベース・レジストリとの相互運用は必須の要素である。

(3)スモールスタートと成果の可視化

データの整備は一気にできない場合も多い。時間や地域、サービスを絞ってのスモールスタートにより成果を利用者に理解してもらい、参加者、応援者を集めながら推進していくことが重要である。また、一過性の実証に終わってもいけない。全体ロードマップと成果を可視化しながら推進を図っていくことが

重要である。

(4)グローバル標準との整合性

データモデルは、可能な限りグローバルな体系を意識して整備する。グローバルな体系に合わせることにより、国際展開を容易にするとともに世界中の先端のサービスを導入しやすくなる。

(5)イノベーションのための成長の仕組み

データに関連する技術は年々進歩しており、最新技術を使ったイノベーティブな取り組みには継続的に取り組んでいく必要がある。そこで、最新技術への適応などで参照モデル以外のモデルを使う場合には、そのモデルや取り組み内容を公開し、本参照モデルの改善に資する情報のフィードバックを求めていく。また、数年おきにモデルの検証を行い、データモデルを変更した場合にはデータ変換ツールやマニュアル提供の検討も行うものとする。

(6)多様な用途に永く使える仕組み

データの通信方法、処理方法、処理単位などは、時代、地域、用途などによって変わるため、特定の利用者による特定の用途だけを想定して固定的なデータモデルを定義することは望ましくない。多様な条件でデータ項目を抽出したり、異なる種類のデータを集計したり、雑多なデータの相関を分析したりすることを想定して、柔軟なデータ構造を検討する。また、データを扱う情報システムの更改やデータストレージ技術の進化に合わせて、将来、データ構造を組み替えることになる可能性も視野に入れ、拡張性のあるデータモデルを検討する。

7-3-3. 具体的な要素

上述の検討内容を考慮し、さらに今後スーパーシティ、スマートシティのデータが国内外の様々な個人・法人によって長期間にわたって永く利用され、プライバシーやデータ主権、コンプライアンス、公平性、経済性、持続可能性といった点にも配慮が必要となることを踏まえると、スーパーシティ、スマートシティのデータモデルは、以下のような要件を満たすものであることが望ましい[73]。

- ポータビリティ（他の地域・国でも同じデータを利用できること）
- ベンダフリー（異なるベンダの機器やシステムで利用できること）
- 汎用性（業界や分野を越えてデータを連携し利活用できること）
- 国際標準化（規格を統一し認証制度で相互接続性を保証できること）
- アクセシビリティ（誰でも必要なデータに公平にアクセスできること）
- ダイバーシティ（国籍や言語、文化、習慣の異なる人も利用できること）
- アクセス制御（個人情報や機密情報へのアクセスを管理できること）
- 完全性（対象に紐づくデータが一意に特定され重複や誤りがないこと）
- 柔軟性（多様な用途、多様なシステムアーキテクチャに対応できること）
- 拡張性（データ項目の新規追加やデータ構造の変更が容易にできること）
- 検索性（必要なデータを任意の方法で簡単に探し出せること）
- トレーサビリティ（いつどこで誰がどのデータを何のために使ったかわかること）
- 時系列性（現在のデータだけでなく過去や未来のデータも管理できること）
- 時限性（データの有効期限を設定し期限後に消去/無効化できること）

[73] 本書に提示するデータモデルが、必ずしもこれらの要件を全て完全に満たすことを保証するものではない。

- 民主性（誰もが自由にデータを作成し既存のデータと連携できること）
- データ主権（データの利用に関する権利を誰が持っているかわかること）
- 実現性（様々な機器やソフトウェアに容易に実装できる）
- 継続性（法制度や国・自治体の形態が変わっても利用できる）

これら要件の全てがデータモデル定義で実現するものではなく、例えばデータ品質の確保に関しては「データ品質管理ガイドブック」等に基づく改善の取り組みが望ましい。そうした品質管理においても、データモデルが持つ値型や値域等の定義は有益である。

7-3-4. データモデルの検討対象

本書においては、以下国内外の主要スマートシティで共通的に使われているデータモデル、国内でニーズの高いデータモデルを対象に調査・取りまとめを行った。

(1)有識者から提示されるデータモデル

内閣府技術検討会で取りまとめる「スーパーシティのデータモデルについて」が対象とするデータモデルである。本書では、2 調査プロセスで実施した既存データモデルの検討結果を踏まえ、以下(1)～(16)のデータモデルを提示する。検討結果は 7-6 に示した。

検討の過程において明らかになった、最低限記述を統一すべき基本データの値については、7-4 に共通記述方法を取りまとめた。これらのルールは政府 CIO ポータルで公開されている「行政データ連携標準」と対応しており、スマートシティ、スーパーシティ以外の行政システムやオープンデータの取り組みにおいても有効である。

表 7-3-4-1 推奨データモデル検討対象

対象	サブセット	基になるモデル	補足	ユースケースとの対応
(1)土地		国土数値情報	ic:土地型を参照	
(2)建物		国土数値情報	ic:建物型を参照	
(3)施設		推奨データセット(施設)	ic:施設型を参照、サービスサブセット例示	○
	公共機関	推奨データセット		
	医療機関	推奨データセット		
	社会福祉法人	推奨データセット		
	観光施設	推奨データセット		
	子育て施設	推奨データセット		
	事業所			
	食品等営業許可･届出事業所			
	公園			
	指定緊急避難場所	推奨データセット		
	交番			
	入浴施設			
	クリーニング施設			
(4)出入口		推奨データセット	ic:設備型を参照	
(5)設備		推奨データセット	ic:設備型を参照	○

対象	サブセット	基になるモデル	補足	ユースケースとの対応
	AED	推奨データセット		
	公衆無線 LAN アクセスポイント	推奨データセット		
	公衆トイレ	推奨データセット		
	消防水利施設	推奨データセット		
	文化財	推奨データセット		
(6)道路	道路	対象外		○
	道路規制情報	交通規制情報		
(7)地物		推奨データセット(施設)	ic:地物型を参照	
	観光ポイント	推奨データセット(施設)		
	位置			
(8)交通	空港、港、駅、バス停	推奨データセット(施設)	GTFS	○
	鉄道、バス路線	GTFS		
	駐車場、駐車スペース、駐輪場	推奨データセット(設備)	ic:駐車場型を参照	
	シェアカー、レンタル自転車スポット	OMA		
	ガソリン、EV、水素ステーション	推奨データセット(施設)	OMA	
(9)イベント		推奨データセット(イベント)		○
(10)センサーデータ	デバイス	推奨データセット(設備)	ic:設備型を参照	
	観測データ			
(11)建物内、地下街	対象外とする。			
(12)地下埋設物	対象外とする。			
(13)オブジェクト	人、物、乗り物など	Schema.org		
(14)自然	天気	気象庁		
(15)緊急情報		公共コモンズ		
(16)行政情報		申請データ、証明データ		
データカタログ	追加で 7-7 に定義	DCAT、DTA データカタログ		

データモデル検討の過程において、7-1-6(7)メタデータのうち、データセットを対象に付与するメタデータの主な役割は、データセットが内包する値に対するメタデータの役割とは異なることが明らかになった。このため、データセットの属性や種別、活用手段に関わらず不可欠であるデータセットそのもののデータモデルを、7-7 データカタログのデータモデル定義として切り分けて提示することとした。

(2)ユースケースシナリオを起点としたデータモデル

スーパーシティ/スマートシティ上で展開され得るサービスのユースケースを想定し、「戦略的イノベ

ーション創造プログラム（SIP）第2期/ビッグデータ・AIを活用したサイバー空間基盤技術/パーソナルデータアーキテクチャ構築」事業[74]の成果であるユースケースシナリオテンプレート[75]を用いてユースケースシナリオを整理した。本ユースケースは「2-5-3(4)ユースケースの検討」にて作成されるものをベースとし、ユースケースシナリオに基づき導かれる、サービスにおいて必要と考えられるデータモデルである。検討結果は7-8に示した。

7-3-5. 推奨データモデル検討の実施内容

7-3-4(1)有識者から提示されるデータモデルを対象に、実在するデータモデルのデータ項目名及びその説明の定義を検討、整理して記載した。なお、7-3-4(2)ユースケースシナリオを起点としたデータモデルに関しては、7-8及び別紙ウ-2にて推奨データモデルとの対応を例示した。

(1)データモデルの検討手順

検討手順は以下のとおりである。

対象となるデータのテーマに関して、推奨データセット、IMI共通語彙基盤、国土数値情報及びGTFS、Schema.orgなどの既存規格で示されているデータモデルを抽出した。

データモデルに含まれるデータ項目定義の構造化を検討した。例えば、推奨データセット「公共施設」を構成するデータ項目のうち、都道府県コード又は市区町村コード、名称などは「情報の管理主体」、施設の管理者を示す団体名や法人番号、連絡先は「管理者」、施設の名称を示す名称、名称_カナなどは施設自体の「名称」とグループ化することができる。

表 7-3-5-1 データ項目構造の例(施設の例を抜粋)

中分類	小分類	項目名
情報の管理主体	基本属性	法人名
		法人番号
	連絡先	連絡先
施設の基本属性	名称	名称
	ID	ID
	用途	用途分類
施設の管理者	基本属性	団体名
		法人番号
		認可等年月日
	連絡先	連絡先
施設の所在		住所
		緯度
		経度
		標高
施設のサービス	提供者	団体名
		法人番号

[74] https://www8.cao.go.jp/cstp/stmain/20200318siparchitecture.html

[75] パーソナルデータを取り扱う事業者が活用するテンプレートドキュメント。
https://data-trading.org/sipb-1_personaldataarchitecuture_dta/

中分類	小分類	項目名
	主要サービス	種別
		POI コード
		災害時対応
	対象者	ライフイベント
		日常生活
		緊急時
		定員
		バリアフリー情報
		駐車場情報
	利用時間	利用可能曜日
		開始時間
		終了時間
	利用方法	料金(基本)
		アクセス方法
	説明	URL
		説明
状況(ステータス)	中期稼働状況	
	短期稼働状況	営業時間
		利用可否
施設の設備	基本属性	設備 ID
施設のサービス	基本属性	サービス ID

(a)なお、表 7-3-5-1　データ項目構造の例(施設の例を抜粋)中分類のうち、「情報の管理主体」は、7-1-6(7)メタデータで分類した「データセット」に対するメタデータに該当すると考えられる。これらデータセットに関するデータ定義は、7-6 と別に 7-7 データカタログのデータモデル定義で整理する。

(b)例えば「施設」には 1)公共施設から 15)クリーニング施設まで 15 の分類が示されている。これらの分類に共通する基本情報を推奨データセット及び IMI 共通語彙基盤コア語彙[76]のクラス用語「ic:施設型」より抽出し、施設共通基本項目として整理した。

(c)(b)の整理結果を踏まえ、「施設」の用途である「サービス」に着目して整理を行った。施設が提供するサービスの内容には、利用可能時間や対象者、定員、利用方法といった共通項目が多いことから、これらを「施設のサービス」共通項目として整理した。(b)と(c)に該当するデータ項目は、「施設」に共通して必要なデータモデル定義と判断した。バリアフリー情報については対象者と相関関係にあると考えられることからやはり共通項目とし、アクセシビリティオプションとして提示した。

[76] IMI 共通語彙基盤が定義する、氏名、住所、組織等、あらゆる社会活動で使用される中核的な用語の集合。用語名に付加している「ic:」は特定の用語がコア語彙で定義されている用語であることを示す時に慣用的に使われる接頭辞で、IMI Core vocabulary の略表記。

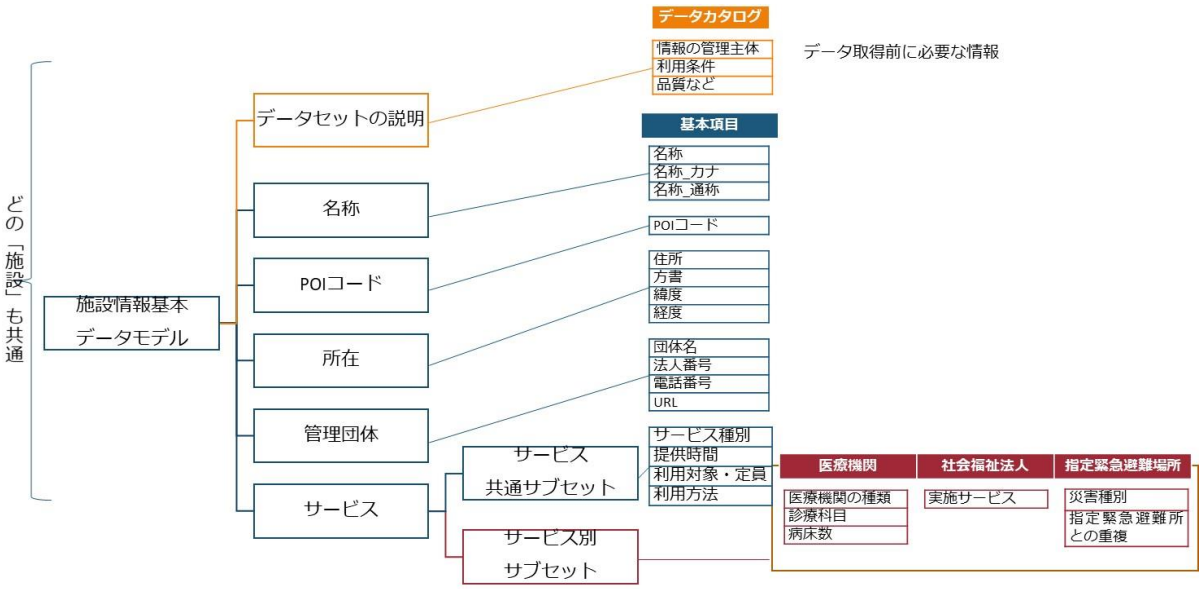


図 7-3-5-1　基本データモデルとサービス別サブセットの関係イメージ

(d)スーパーシティで想定されるサービスを実現するには、「施設」において複数の事業者等が提供する「サービス」や「設備」、「イベント」における様々な講演者等が実施する「付帯イベント」を定義する必要がある。そのため、各データモデルの定義対象にはデータセット内で一意に対象を特定できる ID の付与を必須とした。施設内に設置された「設備」の例では、設置されている施設の ID を示すことにより、「設備の所在」をあらためて記述する必要がなくなる。

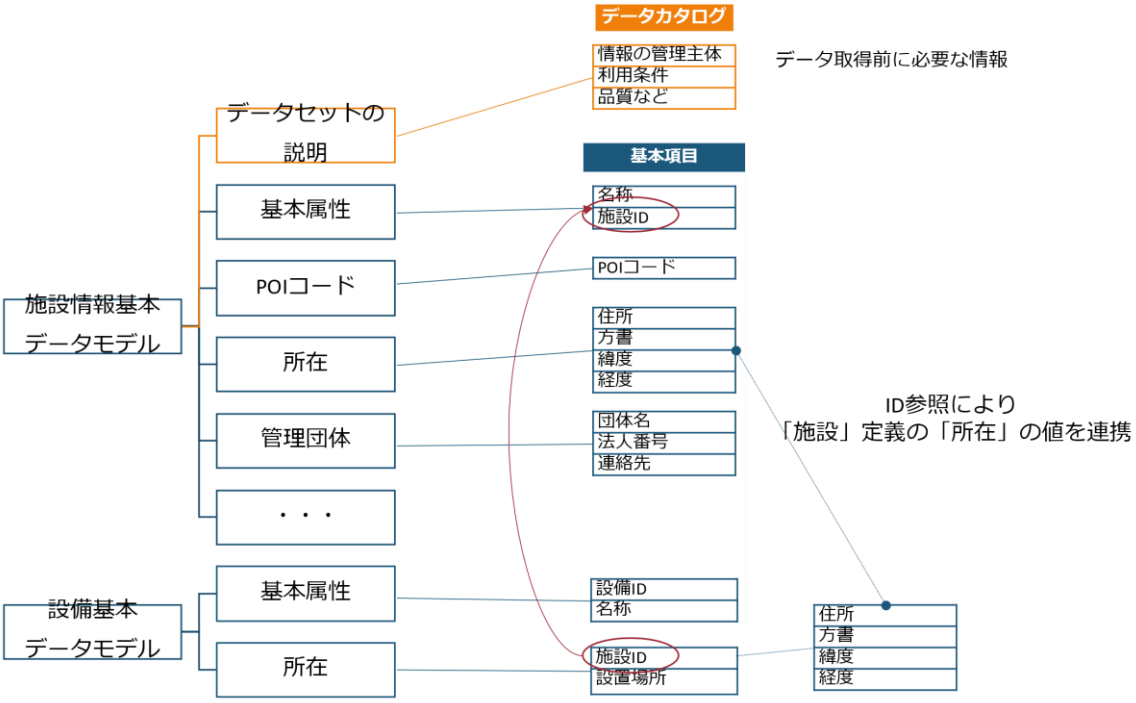


図 7-3-5-2　ID 連携のイメージ

(e)可能な限り、コントロールド・ボキャブラリー(統制語彙)として定義された列挙型コードを提示する。例えば施設の区分を示すコードとしては国土数値情報の「公共施設分類コード」[77]や一般社団法人ユニバーサルメニュー普及協会が運営管理する「ユニバーサルメニュー」[78]などを示した。

(f)「医療機関」「教育機関」「指定緊急避難場所」といった用途別の施設が提供するサービスのデータの意味を標準的に理解しやすいよう、用途ごとの独自のデータ項目定義を整理した。極力データの相互運用性を確保するため、データモデルやコード体系、データベース等が存在する場合には、それらを参照して記述するように定義を示している。

(g)組織やシステムの用語定義の違いに起因する表記ゆれ等のデータ項目定義については、今後の検討課題とした[79]。例えば、「住所」「所在」「所在地」のような同じ意味に対する表現上の違いがある。また、「関東」「西日本」などの「地域」のように、法律や制度による行政の所管により、その用語の示す対象範囲の違いが異なるケースがある。さらには、「住民数」等のデータにおいても、データの基準日が異なるケースもある。本データモデルでは、代表的な表現を使うように作成するが、基準日の違い等の詳細については、今後の検討課題として取り扱わないこととした。

(2)データモデルの適用

スマートシティ間、スーパーシティ間で相互運用性を確保することが前提であるため、以下推奨データモデルの最小構成には極力沿うこととする。

(3)データモデルとデータカタログ

スーパーシティでデータ連携される全てのデータセットは、データセット提供者(情報管理主体)とデータセットの情報、利用条件といったデータセットを説明するメタデータすなわちデータカタログの記載内容(オレンジ色の枠)、データ項目に関するメタデータすなわちデータモデル(青色の枠)を付与される必要がある。

---

[77] https://nlftp.mlit.go.jp/ksj/gml/codelist/PubFacMaclassCd.html

[78] 市区町村等自治体による身近な行政サービスの情報を、誰もが簡単に検索・活用できる行政サービス関連標準メニュー体系。https://universalmenu.org/openum/um/about.html

[79] ただし、本データモデルで参照する「IMI 共通語彙基盤コア語彙」とデータ項目の対応付け(マッピング)を行うことによりデータ間の相互運用性を確保する方法がある。また同じ用語で意味が異なる場合は、参照した列挙コード等の定義をメタデータとして示すことで値の意味を伝えやすくなる。独立行政法人 情報処理推進機構「データの相互運用性向上のためのガイド」https://www.ipa.go.jp/ikc/our_activities/dt_data_guide01.html を参照。また、基準日は 7-7 で示すデータカタログに記載するといった対処が考えられる。

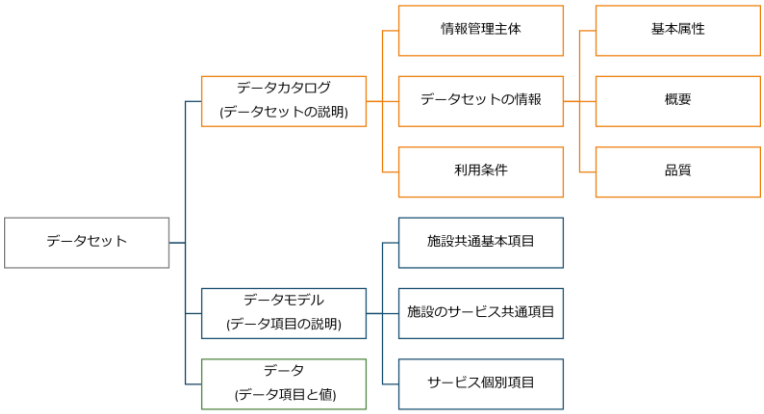


図 7-3-5-3　データモデルとデータカタログ

(4)実装方法

データカタログ及びデータモデルの内容を利用者へ伝える実装方法については、本書では論じない。例えば「施設」データセットとしてメタデータを付与する単位は、図 7-3-5-4 のように様々な施設をひとつの対象とする方法も、図 7-3-5-5 のように施設に含まれる公共施設や医療機関を対象とする方法もあり得る[80]。あるデータ提供サイトがサイト全体として統一された情報管理主体や利用条件を持つ場合は、データセット単位ではなくサイトとして情報提供を行うことも考えられる。また、システム構築時等に一度データモデルを伝達すればよい場合も、データを受け渡す都度データモデルを伝達すべき場合も考えられる。

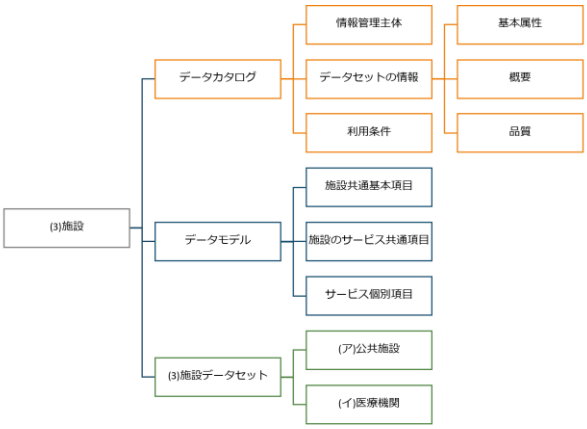


図 7-3-5-4　「施設」の単位とする例

[80] ここでは論じないが、図 7-3-5-4 の「(3)施設」や図 7-3-5-5 の「(3)施設(ア)公共施設」のように、データカタログ、データモデル、値を含むデータセットから構成される塊を「オブジェクト」と呼ぶ場合もある。

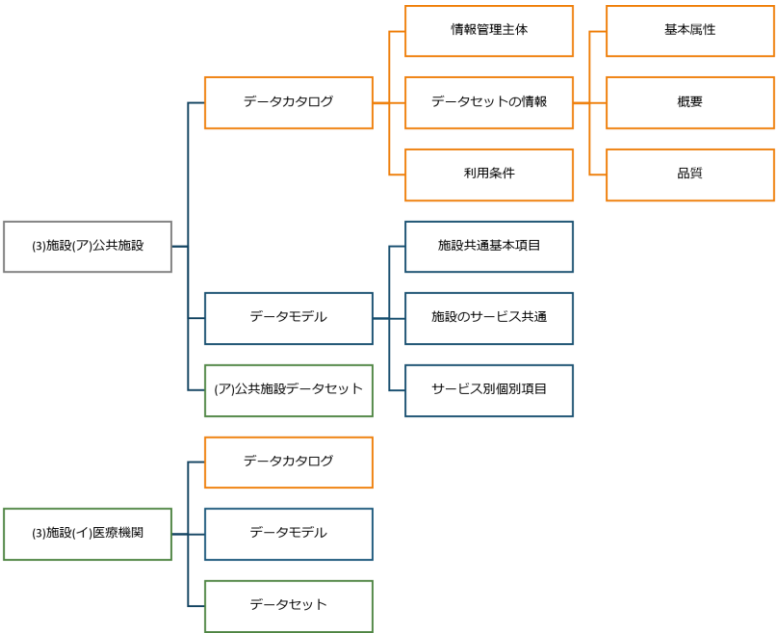


図 7-3-5-5　施設>公共施設の単位とする例

7-4. 基本データの共通記述方法

7-5 コード体系・ID 体系、7-7 データカタログのデータモデル定義に共通して、全てのメタデータで共通的なデータモデル、データモデルを構成する要素の値の共通記述方法を説明する。

7-4-1. 文字

文字は、スマートフォン等の一般的な機器に搭載されている JIS X 0213（JIS 第 4 水準までの 1 万文字）の範囲内とする。また文字符号化は UTF-8 を使用する。また、氏名や地名、法人名等の固有名詞にはヨミガナをデータ項目として定義することとする。詳細は、文字環境導入実践ガイドブック[81]を参照することとする。

7-4-2. 外国語表記・ピクトグラム

スーパーシティの各サービスは日本語を理解しない人が利用することが想定される。そのため、最低限、英字対応することが求められる。

地名の英字については、国土交通省「地名等の英語表記規程」[82]の英字表記に基づき決定する。施設名等の英字表記は施設等の提示する名称を基本とする。また、観光庁「観光立国実現に向けた 多言語対応の改善・強化のためのガイドライン」[83]や東京都「国内外旅行者のためのわかりやすい案内サイン標準化指針」[84]を参考にする。

さらに、都市サービスでは言語がわからなくても直感的に理解ができるように多くのピクトグラムが使用される。観光庁「観光立国実現に向けた 多言語対応の改善・強化のためのガイドライン」[85]や東京都「国内外旅行者のためのわかりやすい案内サイン標準化指針」[86]を参考に活用する。

7-4-3. 日時

「行政データ連携標準（日付時刻)」[87]に基づき、ISO 8601 の拡張形式である YYYY-MM-DD を使用する。これはデータの形式であり、表示や印字においては自由な表記形式に変換して利用することする。時間や時期をコードで記述する場合は、7-5 コード体系・ID 体系を参照する。

開始時間、終了時間などの期間がある場合には、データ項目を分離して案内などの処理をしやすいようにする。曜日指定、季節、時間帯、タイムゾーン等も「行政データ連携標準（日付時刻)」を活用することとする。

利用可能日、利用不可日を表す時には、利用可能日を記述する。例えば「休館：土日祝日」ではなく「開館：月火水木金（祝日は除く)」とする。

(1)定期スケジュール

日時が特定せずに定期スケジュールで行われるイベントなどがある。種別は、「年次スケジュール」「月次スケジュール」「週次スケジュール」がある。

定期スケジュールの記述は、ニーズが高いが自由記述が多く、記述方法に標準がない。本検討では以下

---

[81] https://cio.go.jp/guides

[82] http://www.gsi.go.jp/common/000138865.pdf

[83] https://www.mlit.go.jp/common/001029742.pdf

[84] https://www.sangyo-rodo.metro.tokyo.lg.jp/tourism/signs/

[85] https://www.mlit.go.jp/common/001029742.pdf

[86] https://www.sangyo-rodo.metro.tokyo.lg.jp/tourism/signs/

[87] https://cio.go.jp/guides

のように定義する。

(a)年次スケジュール

年の特定日に行うスケジュール事項である。

- 月日を MM-DD で、カンマ区切りで列挙する。

(b)月次スケジュール

月の特定日に行うスケジュールである。

- 日を DD で、カンマ区切りで列挙する。

隔月イベントは、備考欄に「隔月であること」を記載する。

(c)週次スケジュール

週の特定日に行うスケジュールである。

- 曜日は Mo、Tu、We、Th、Fr、Sa、Su で表す。
- 毎週は 0、月曜 1 週から日曜 5 週は 1-5、最終は 6、隔週は 7 とする。

記述例

毎週水曜日	0We
第 1、3 月曜日	13Mo
第 1、4 火曜日と第 4 木曜日	14Tu4Th
最終金曜	6Fr
隔週日曜	7Su

(2)特定の日付時刻

認定日や開始日、終了日など特定の日付時刻の記述方法は、「行政データ連携標準(日付時刻)」[88]に従う。

7-4-4. 緯度、経度

緯度経度は、1m 程度の誤差を許容し、「行政データ連携標準（緯度・経度)」に従い、小数点以下 6 桁を基本とする。ただし、測量法が定める誤差等、作るデータの目的により許容される誤差が異なる場合がある。その場合は精度の変換をしたり注記で記載する等で対応することとなる。基となるデータの誤差範囲を制約するものではない。

特別な精度を必要とするサービスの場合は、サービス側でリアルタイム測地点による補正や小数点以下 7 桁情報の利用をすることなどを検討する。

7-4-5. 住所等

住所等の所在地を表す情報は、「行政データ連携標準（住所)」に従うこととする。ただし、旧システムからのデータ公開などが想定され、用途に応じたデータ変換は比較的容易であることから、(1)から(3)に例示した複数の方式を許容する。

(1)都道府県分離表記方式

最も一般的な入力方式である。表示する時には、「都道府県」「住所市区町村町名」が統合されることも多い。

[88] https://cio.go.jp/guides

表 7-4-5-1　都道府県分離表記方式のデータ項目

データ項目	説明
都道府県	都道府県名を記入。(例：東京都)
市区町村町名	住所を市区町村から記入。(丁目以下省略)
丁目以下	住所の丁目以下、地番を半角数字とハイフンで記入。
建物名等	建物名等の情報がある場合に使用。

(2)都道府県一体表記方式

一般的な入力方式である。

表 7-4-5-2　都道府県一体表記方式のデータ項目

データ項目	説明
住所町名	住所を都道府県から記入。(丁目以下省略)
住所丁目以下	住所の丁目以下、地番を半角数字とハイフンで記入。
建物名等	建物名等の情報がある場合に使用。

(3)全住所一体方式

多くの従来データで使われる方式である。

表 7-4-5-3　全住所一体方式のデータ項目

データ項目	説明
住所	所在地を記入。(都道府県名から、丁目、号までを１データで記載)
建物名等	建物名等の情報がある場合に使用。

7-4-6.　連絡先

連絡先を表す情報は、以下のデータ項目を持つ。担当者は明記しない場合も多く、任意項目とする。コア語彙「ic:連絡先型」を参照する。必要に応じてLINE IDなどを追加しても良い。

表 7-4-6-1　連絡先のデータ項目

データ項目	説明
役割	連絡先の役割
法人番号	
法人名	
担当部署	担当部署名
担当者役職	担当者の役職(任意記述)
担当者名の氏	担当者の氏(任意記述)
担当者名の氏（カナ）	担当者の氏のカナ表記(任意記述)

担当者名の名	担当者の名(任意記述)
担当者名の名（カナ）	担当者の名のカナ表記(任意記述)
電話番号	担当部署の電話番号（市外局番にカッコをつけ、以降の番号はハイフンで接続。半角）
内線	担当部署の電話番号の内線番号 電話番号に「直通」「代表」と記載したい場合は、この欄に記入
メールアドレス	連絡先のメールアドレス
住所	7-4-5 を参照。
web フォーム	連絡先の Web フォーム URL

7-4-7. センサーデバイス

センサーデバイスは、データ項目が多く、以下の OASC Shared Data Models for Smart City domains (SynchroniCity Data Models)のデバイス[89]を参照することとする。全ての項目を使う必要はない。参照情報であるため、下表の翻訳は行わない。

表 7-4-7-1 センサーデバイスのデータ項目

データ項目	説明
id	Unique identifier.
type	Entity type. It must be equal to Device.
source	A sequence of characters giving the source of the entity data.
dataProvider	Specifies the URL to information about the provider of this information
category	See attribute category from DeviceModel. Optional but recommended to optimize queries.
Controlled Property	See attribute controlledProperty from DeviceModel.
controlledAsset	The asset(s) (building, object, etc.) controlled by the device.
mnc	This property identifies the Mobile Network Code (MNC) of the network the device is attached to. The MNC is used in combination with a Mobile Country Code (MCC) (also known as a "MCC / MNC tuple") to uniquely identify a mobile phone operator/carrier using the GSM, CDMA, iDEN, TETRA and 3G / 4G public land mobile networks and some satellite mobile networks.
mcc	Mobile Country Code - This property identifies univoquely the country of the mobile network the device is attached to.
macAddress	The MAC address of the device.
ipAddress	The IP address of the device. It can be a comma separated list of values if the device has more than one IP address.
Supported Protocol	See attribute supportedProtocol from DeviceModel. Needed if due to a software update new protocols are supported. Otherwise it is better to

[89] https://github.com/smart-data-models/dataModel.Device/blob/master/Device/doc/spec.md

データ項目	説明
	convey it at DeviceModel level.
configuration	Device's technical configuration. This attribute is intended to be a dictionary of properties which capture parameters which have to do with the configuration of a device (timeouts, reporting periods, etc.) and which are not currently covered by the standard attributes defined by this model.
location	Location of this device represented by a GeoJSON geometry of type point.
distance	Location of this device represented by a distance from a starting point.
depth	Location of this device represented by a depth from a starting point.
relativePosition	Location of this device in a coordinate system according to its local emplacement.
direction	Location of this device represented by its direction according to its local emplacement.
name	A mnemonic name given to the device.
description	Device's description.
dateInstalled	A timestamp which denotes when the device was installed (if it requires installation).
dateFirstUsed	A timestamp which denotes when the device was first used.
dateManufactured	A timestamp which denotes when the device was manufactured.
hardwareVersion	The hardware version of this device.
softwareVersion	The software version of this device.
firmwareVersion	The firmware version of this device.
osVersion	The version of the host operating system device.
dateLast Calibration	A timestamp which denotes when the last calibration of the device happened.
serialNumber	The serial number assigned by the manufacturer.
provider	The provider of the device.
refDeviceModel	The device's model.
batteryLevel	Device's battery level. It must be equal to 1.0 when battery is full. 0.0 when battery is empty. -1 when transiently cannot be determined.
rssi	Received signal strength indicator for a wireless enabled device. It must be expressed in dBm or mW, use unitcode to set it out.
deviceState	State of this device from an operational point of view. Its value can be vendor dependent.
dateLastValue Reported	A timestamp which denotes the last time when the device successfully reported data to the cloud.
value	A observed or reported value. For actuator devices, it is an attribute that allows a controlling application to change the actuation setting. For instance, a switch device which is currently on can report a value "on"of type Text. Obviously, in order to toggle the referred switch, this attribute value will have to be changed to "off".
dateModified	Last update timestamp of this entity.

データ項目	説明
dateCreated	Entity's creation timestamp.
owner	The owners of a Device.

7-5. コード体系・ID 体系

7-2 データモデルの役割で概説したとおり、スーパーシティを構成するデータには、社会全体で共通認識できるコードと ID が必要である。ただし現時点で公式に共通標準と認められるコード体系や ID 体系は限定されるため、本書では可能な範囲で既存コードリストを提示することとした。

コード体系、ID 体系整備の必要性は「データ戦略タスクフォース第一次とりまとめ」[90]でも言及されており、今後標準的に利用すべきコード体系、ID 体系の整備が進む見込みである。将来の標準コードと別のコード体系、ID 体系を現在各スーパーシティにおいて採用した場合も、一定のルールに従ったコード体系、ID 体系であれば変換テーブル等と用いた相互運用性確保の可能性は高まると考えられる[91]ため、可能な限りコードで値を記述できるデータモデルが望ましい。

7-5-1. 地理空間に関するコード

(1)自治体コード

総務省自治行政局全国地方公共団体コード

https://www.soumu.go.jp/denshijiti/code.html

(2)POI コード

観光施設、公共施設など地理的目標物を POI（Point Of Interest）という。政府 CIO ポータルサイトの「行政データ連携標準（PoI)」を参照する。

https://cio.go.jp/guides

(3)建物コード

調査中

7-5-2. 時間、時期に関するコード

時間や時期に関する情報は、物流や施設等、スマートシティに関連するサービスで使われることが多いが、運営者や主催者により様々な区分で管理されている。「行政データ連携標準（日付時刻)」を活用することとする。

桜の開花時期、大潮等の日時の定義が困難なデータは備考に記入する。

(1)時間帯コード

「行政データ連携標準（日付時刻)」を活用することとする。

表 7-5-2-1 時間帯コード

コード	時間帯	コード	時間帯
00	00:00/01:00	12	12:00/13:00

[91] 例えば経済産業省「gBizINFO」は変換テーブルを用いて各省庁ごとに異なるコード体系で構築されたデータベース上のデータを統合的に処理している。https://www.ipa.go.jp/files/000087414.pdf
また農林水産省は既存コード体系を作り直すのではなく、各コード体系内の語彙同士の完全一致、部分一致、不一致を機械的に判別できるようにして、農産物検査や経営所得安定対策、農地台帳といった省内コード体系、農林水産省や総務省の統計、貿易の HS コードでそれぞれ異なる農産品目マスタの変換テーブルを整備し相互運用性を確保している。

コード	時間帯	コード	時間帯
01	01:00/02:00	13	13:00/14:00
02	02:00/03:00	14	14:00/15:00
03	03:00/04:00	15	15:00/16:00
04	04:00/05:00	16	16:00/17:00
05	05:00/06:00	17	17:00/18:00
06	06:00/07:00	18	18:00/19:00
07	07:00/08:00	19	19:00/20:00
08	08:00/09:00	20	20:00/21:00
09	09:00/10:00	21	21:00/22:00
10	10:00/11:00	22	22:00/23:00
11	11:00/12:00	23	23:00/24:00
		99	不明・未定

表 7-5-2-2　目標時間コード

コード	目標時間	コード	目標時間
00	00:00	12	12:00
01	01:00	13	13:00
02	02:00	14	14:00
03	03:00	15	15:00
04	04:00	16	16:00
05	05:00	17	17:00
06	06:00	18	18:00
07	07:00	19	19:00
08	08:00	20	20:00
09	09:00	21	21:00
10	10:00	22	22:00
11	11:00	23	23:00
		99	不明・未定

(2)季節、旬コード

「行政データ連携標準（日付時刻)」を活用することする。

表 7-5-2-3　季節コード

コード	季節	コード	季節
10	春	30	秋
11	初春	31	初秋
12	仲春	32	仲秋
13	晩春	33	晩秋
20	夏	40	冬

コード	季節	コード	季節
21	初春	41	初冬
22	仲夏	42	仲冬
23	晩夏	43	晩冬

表 7-5-2-4　旬コード

コード	旬
10	上旬
20	中旬
30	下旬
99	不明・未定

7-5-3.　状態に関するコード

状態に関する情報はスマートシティに関連するサービスで使われることが多いが、運営者や主催者により様々な区分で管理されており、標準といえるコードは確立していない。しかし、情報管理主体やデータセットの単位で統一した状態に関するコードを使用することは有益と考えられる。したがってここでは、代表的な列挙型コード（コントロールドボキャブラリ）を選択肢候補として示す。

(1)施設やイベントの状況

施設、施設やイベントでは、開催状況に関する情報を必要とする場合がある。統一的なコードはないが、以下のような列挙型コードが考えられる。

(a)中期的な状況

Google Maps Platform では稼働(OPERATIONAL)、一時閉鎖(CLOSED_TEMPORARY)、閉鎖(CLOSED_PERMANENTLY)の区分を設定している。7-6 では中期的な稼働状況をこの区分で示した。

(b)短期の状況

「準備中」、「受付中」、「開催中」、「中止」、「休止中」、「復旧中[92]」、「延期」などの区分が考えられる。

(2)混雑の状況

施設、施設やイベント、交通機関では、混雑に関する情報を必要とする場合がある。統一的なコードはないが、以下のような選択肢候補が考えられる。

「空あり」、「混雑」、「空なし」

(3)予約や利用可否の状況

施設、施設やイベント、交通機関では、予約やある日時時刻における利用可否に関する情報を必要とする場合がある。統一的なコードはないが、以下のような選択肢候補が考えられる。

「予約可」、「要問合せ」、「予約不可」、「当日可」、「キャンセル待可」、
「予約不要」、「先着順」

[92] 「復旧中」は、故障、事故や災害時に使用される。

7-6. 推奨データモデル

スーパーシティ、スマートシティでニーズが高いと考えられるデータのデータモデルを示す。相互運用性を確保するため、7-3-2 推奨データモデルの整理原則にしたがってできる限り既存のデータモデルを参照した。

各データモデルは、そのデータが主に参照した既存データモデルを示す「参照データモデル」、データ項目の構造、名称と説明を示す「データモデル」、データモデルの中で自由記述ではなく選択肢コードとして定義されることが望ましい「列挙型コード」、その他参考になる「既存のデータセット・データモデル等」により構成される。

また、施設や設備等は、個々の類型で項目の差異が小さいため、基本形を示した上で施設や設備の特性に応じて必要なデータ項目を記述した。

各データ項目には、必須◎、推奨○の重要度レベル（L）をつけた。空欄の場合は任意項目である。

なお、コンピュータ処理の必要から、全てのデータ項目は 7-2 で説明したとおり共通の標準コードに基づくコードと ID で記述されることが望ましい。しかし、現時点で公式に共通標準とするコード体系や ID 体系は限定される[93]ため、本参照モデルでは可能な範囲で既存コードリスト等を提示することとした。

7-6-1. 土地

特定の用途に用いられている土地を記述するためのデータモデルである。土地データに求められるデータモデルは、利用目的によって異なる。主には、現実世界に存在する(a)位置等の情報を記述するデータモデル、(b)所有権等を示す不動産としての土地情報を記述するデータモデルが想定される。

(1)土地共通の参照データモデル

表 7-6-1-1　土地共通の参照データモデル

名称	英語名称	参照元	参照 url
土地	Land	IMI 共通語彙基盤	https://imi.go.jp/ns/core/Core242.html#ic:土地型

(2)土地の基本データモデル

(a)位置等の情報を記述するデータモデル

表 7-6-1-2　位置等の情報を記述するデータモデル

	データ項目	L	参照	説明
基本属性	不動産 ID		ic:ID	不動産に一意に付与された ID。（将来使用）
	名称		ic:名称	場所を特定する名称（地名など）を記入。
	通称		ic:通称	場所の正式な名称以外に一般的に使用されている名称がある場合に記入。
	用途	○	ic:種別	土地使用用途を記入。（都市計画基礎調査の区分）
所在	住所	◎	ic:住所	住所を記入。
	地理座標	○	ic:地理座標	場所の経緯度座標を記入。

[93] 一部を 7-27 に示した。

	データ項目	L	参照	説明
説明	面積	○	ic:面積	土地の面積を記入。(m2)
	ポリゴン	○	ic:参照	ポリゴン情報を記入。
	説明		ic:説明	説明を記入。(必要な場合)
	画像		ic:画像	画像を参照する URL がある場合に記入。

(b)所有権等を示す不動産としての土地情報を記述するデータモデル

表 7-6-1-3　所有権等を示す不動産としての土地情報を記述するデータモデル

分類	データ項目	L	参照	説明
基本属性	不動産 ID		ic:ID	不動産に一意に付与された ID。(将来使用)
	名称		ic:名称	場所を特定する名称（地名など）を記入。
	用途	○	ic:種別	土地使用用途を記入。(都市計画基礎調査の区分)
所在	住所	◎	ic:住所	住所を記入。
	地理座標	○	ic:地理座標	場所の経緯度座標を記入。
説明	面積	◎	ic:面積	土地の面積を記入。(m2)
	説明		ic:説明	説明を記入。(必要な場合)
管理者	連絡先	○	ic:連絡先	連絡先を記入。
	管理者		ic:連絡先	管理者を記入。

(3)土地共通の列挙型コード

(a)土地の用途

都市計画基礎調査実施要領[94]（平成 31 年 3 月国土交通省都市局）の区分を使用する。

表 7-6-1-4　土地の用途区分

	用途区分	細分用途
自然的土地利用	田	水田
	畑	畑、樹園地、採草地、養鶏（牛、豚）場
	山林	樹林地
	水面	河川水面、湖沼、ため池、用水路、濠、運河水面
	その他自然地	原野・牧野、荒れ地※、低湿地、河川敷･河原、海浜、湖岸
都市的土地利用	住宅用地	建物用途現況図の 5～9
	商業用地	同 1～4
	工業用地	同 13
	農林漁業施設用地	同 14
	公益施設用地	同 10、11、15
	道路用地	道路、駅前広場
	交通施設用地	同 12
	公共空地	公園・緑地、広場、運動場、墓園

[94] https://www.mlit.go.jp/common/001282174.pdf

	用途区分	細分用途
	その他公的施設用地	防衛施設用地
	その他の空地①	ゴルフ場
	その他の空地②	太陽光発電のシステムを直接整備している土地
	その他の空地③	平面駐車場
	その他の空地④	その他の空地①～③以外の都市的土地利用 （建物跡地、資材置場、改変工事中の土地、法面〈道路、造成地 等の主利用に含まれない法面〉）
不明		不明な土地

(4)既存のデータセット・データモデル等

(a)政府関連土地関連基礎情報

政府内の専門領域のデータ構造やコードが参照できる。

- DID 地区（国土交通省都市計画基礎調査）
- 土地利用：区域区分（国土交通省都市計画基礎調査）
- 都市計画区域及び準都市計画区域
- 土地利用現況（国土交通省都市計画基礎調査）
- 農地の区画情報（農林水産省筆ポリゴンデータ）
  https://www.maff.go.jp/j/tokei/porigon/
- 土地利用 3 次メッシュ、土地利用細分メッシュ、都市地域土地利用細分メッシュ、土地利用詳細メッシュ、森林地域、国有林野、農業地域、都市地域、用途地域（国土数値情報）
  https://nlftp.mlit.go.jp/ksj/indexindex.html#kokudo
- 浸水想定区域［防災計画（ハザードマップ)］(国土交通省 水管理・国土保全局 河川環境課 水防企画室)
  https://www.mlit.go.jp/river/shishin_guideline/bousai/saigai/tisiki/syozaiti/pdf/e-guideline_1910.pdf

(b)Schema.org

国際的な web データ標準である schema.org の「場所」を参照できる。https://schema.org/Place

(c)不動産登記情報（参考）

不動産登記の表題部から所有者情報を除いたデータ項目が参照できる。

表 7-6-1-5　不動産登記情報のデータ項目

データ項目	説明
調製	登記記録作成日を記入する。
不動産番号	不動産固有の ID を記入する。
地図番号	登記所に備え付けられている精度の高い地図情報の番号を記入する。（登記所に地図が備え付けていない場合は空白）
筆界特定	筆界特定所が作成されている場合にその旨を記入する。
所在	市区町村と町字を記入する。
地番	地番を記入する。

データ項目	説明
地目	「宅地」、「田」、「畑」、「山林」、「原野」、「用悪水路」、「公衆用道路」、「公園」、「雑種地」、「境内地」、「牧場」、「鉱泉地」、「池沼」、「墓地」、「水道用地」、「運河用地」、「ため池」、「保安林」、「堤」、「井溝」、「保安林」、「鉄道用地」、「学校用地」から選択。
地積	水平投影面積により、m2（平方メートル）で小数点以下2桁まで記入する。
原因	発生年月日と「売買」等の事象を記入する。

7-6-2. 建物

居住、仕事、娯楽等何らかの役割のために建てられた建物(ビル、建造物、建築物、家屋)を記述するためのデータモデルである。建物自体に備わった情報を主に記述する。建物が施設として用いられる時は、施設としての情報は施設型を使って記述し、この建物を使う施設の情報として分けて記述することもできる。

(1)建物共通の参照データモデル

表 7-6-2-1 建物共通の参照データモデル

名称	英語名称	参照元	参照 url
建物	Building	IMI 共通語彙基盤	https://imi.go.jp/ns/core/Core242.html#ic: 建物型

(2)建物の基本データモデル

表 7-6-2-2 建物の基本データモデル

分類	データ項目	L	参照	説明
基本属性	不動産 ID		ic:ID	不動産に一意に付与された ID。(将来使用)
	種別	○	ic:種別	種別を記述する。種別は 7-6-2(3)(a)を参照。
	名称	○	ic:名称	場所を特定する名称（地名など）を記述する。
	通称		ic:通称	場所の正式な名称以外に一般的に使用されている名称を記述する。
	説明		ic:説明	説明を記述する。
所在	住所	◎	ic:住所	建物の所在の住所を記載。記述は「基本データの共通記述方法」を参照。
	地理座標	○	ic:地理座標	建物の所在の住所を記載。記述は「基本データの共通記述方法」を参照。
説明	敷地面積	○	ic:敷地面積	建物の敷地面積を記述する。
	主要用途	○	ic:主要用途	建物の主要用途の表記を記述する。
	建築面積	○	ic:建築面積	建物の建築面積を記述する。
	延べ面積		ic:延べ面	建物の延べ床面積を記述する。

分類	データ項目	L	参照	説明
			積	
	最高の高さ		ic:最高の高さ	建物の最高点の高さを記述する。
	地上階数		ic:地上階数	建物の地上階数を記述する。。
	地下階数		ic:地下階数	建物の地下階数を記述する
	構造	○	ic:構造	建物の構造の表記を記述する。構造は 7-6-2(4)(a)を参照。
	竣工日	○	ic:竣工日	建物の竣工日を記述する。
管理者	連絡先	○	ic:連絡先	建物の連絡先を記述する。
	管理者		ic:管理者	建物を管理している組織や人を記述する
説明	ic:参照		ic:参照	追加情報などを参照する URL を記述するためのプロパティ用語。
	ic:画像		ic:画像	画像を参照する URL を記述するためのプロパティ用語。

(3)建物共通の列挙型コード

(a)種別

一戸建ての住宅、工場等、建築基準法施行規則（別記様式）に定める建築物用途区分コードを使用できる。

https://elaws.e-gov.go.jp/document?lawid=325M50004000040_20200907_502M60000800074

(b)構造

不動産登記の「構造」区分を使用することができる。次項の不動産登記情報に参照情報を示す。

(4)既存のデータモデル等

(a)不動産登記情報

不動産登記の表題部から所有者情報を除いたデータ項目を参考にできる。

表 7-6-2-3　不動産登記情報のデータ項目

データ項目	説明
調製	登記記録作成日を記入する。
所在図番号	建物所在図が備え付けられている場合に記入する。
所在	町字に加えて地番を記入する（土地投機と異なるので注意が必要)。
家屋番号	家屋に固有の番号を記入する（地番が入る場合とさらに枝番が入る場合がある)。
種類	不動産登記規則第 113 条で示す種類 「居宅」、「店舗」、「寄宿舎」、「共同住宅」、「事務所」、「旅館」、「料理店」、「工場」、「倉庫」、「車庫」、「発電所」、「変電所」 不動産登記事務取扱手続準則第 80 条で示す種類 「校舎」、「講堂」、「研究所」、「病院」、「診療所」、「集会所」、「公会堂」、「停車場」、「劇場」、「映画館」、「遊技場」、「競技場」、「野球場」、「競馬場」、「公衆浴場」、「火葬場」、「守衛所」、「茶室」、「温室」、「蚕室」、「物置」、「便所」、「鶏舎」、「酪農舎」、「給油所」

データ項目	説明
	これらの区分に該当しない建物については、これに準じて定める。
構造	構成材料による区分 「木造」、「土蔵」、「石」、「れんが」、「コンクリートブロック」、「鉄骨」、「鉄筋コンクリート」、「鉄骨鉄筋コンクリート」 屋根の種類による区分 「かわらぶき」、「スレートぶき」、「亜鉛メッキ鋼板ぶき」、「草ぶき」、「陸屋根」 階数による区分 「平家建」、「二階建（三階建以上の建物にあっては、これに準ずるものとする。）」 これらの区分に該当しない建物については、これに準じて定める。
床面積	各階の床面積を m2（平方メートル）で小数点以下 2 桁まで記入する。
原因	発生年月日と「売買」等の事象を記入する。

(b)スマートシティ・データモデル

スマートシティ・データモデルのビルディングモデルを参照することで詳細な表現ができる。

https://github.com/smart-data-models/dataModel.Building/tree/1ce9e7483ac51cfca6d3d3dd4859257b80746300/Building

(c)FIWARE

FIWARE データモデルビルディングモデルを参照することで詳細な表現ができる。

https://fiware-datamodels.readthedocs.io/en/latest/Building/Building/doc/spec/index.html

(d)CityGML

CityGML データモデルのビルディングを参照することで都市レベルでの建物の表現や階層構造などの表現ができる。

https://www.ogc.org/standards/citygml

(e)Industry Foundation Classes(IFC)

buildingSMART が定義する、建物を構成する全てのオブジェクトのシステム的な表現方法の仕様のデータモデル群。配管や電設なども含めた詳細の建物表現ができる。

https://www.buildingsmart.or

7-6-3. 施設

施設の種類は多岐にわたる。施設の「基本モデル」+「施設のサービス別サブセット」の組み合わせで、目的に応じてデータ項目を付加して活用する。

娯楽施設、医療施設、交通施設など、ある目的のために用いられる建造物やその設備を施設として記述するためのデータモデルである。多くの施設は建物が設置されているが、施設型には施設として持つ情報を記述する。設置されている建物の情報の記述が必要な時は 7-6-2 建物のデータモデル、施設内に含まれる設備の情報の記述が必要な場合は 7-6-5 設備のデータモデルを用いて記述し、施設データセットに紐づく属性として参照させる。

なお、大型商業施設や役所、総合病院のように、多くの施設には管理者や利用方法等が異なる複数のサービスや機能が含まれる。利用目的によってデータセットに求められる粒度は異なるため、ここでは施設全体として持つ情報が記述されることを想定する。施設データセットに含まれる個々のサービスについ

て記述する場合は、「施設のサービス」に該当する項目で記述し、施設に紐付く属性として参照させる。

(1)施設共通の参照データモデル

表 7-6-3-1　施設共通の参照データモデル

名称	英語名称	参照元	参照 url
施設	Facility	IMI 共通語彙基盤	https://imi.go.jp/ns/core/Core242.html#ic:施設型
公共施設	Public Facility	推奨データセット	https://cio.go.jp/policy-opendata
観光施設	Tourism Facility	推奨データセット	https://cio.go.jp/policy-opendata

(2)施設の基本データモデル(基本形)

表 7-6-3-2　施設の基本データモデル(基本形)

中分類	小分類	項目名	L	説明
	名称	名称	◎	施設の名称を記載。
		名称_カナ	◎	施設の名称をカナで記載。
		名称_通称	○	施設の通称を記載。
		名称_英語		施設の名称を英語で記載。
	ID	ID	○	データセット内で施設が一意に決まるよう、ID を設定し記載。施設カルテなど既存のデータセットに ID がある場合は流用しても良い。
	用途	用途分類		建築基準法の「用途地域」分類から選択。住宅、店舗等、事務所等、ホテル・旅館、遊戯施設・風俗施設、公共施設・病院・学校等、工場・倉庫等
		公共施設分類コード		公共施設の場合は、国土数値情報「公共施設小分類コード」を適用し、施設の用途を記載する。
	基本属性	団体名		施設の管理団体の名称を記載。
		法人番号		施設の管理者の法人番号を記載。
		認可等年月日		管理者が認可された日付を年月日で記載。記述方法は「基本データの共通記述方法」の「日時」を参照。
	連絡先	連絡先	◎	施設の設置者の連絡先を記載。記述は「基本データの共通記述方法」の「連絡先」を参照。
施設の設置者		団体名		管理者と設置者が異なる場合、施設の設置団体の名称を記載。
		法人番号		施設の設置者の法人番号を記載。
	連絡先	連絡先		施設の設置者の連絡先を記載。記述は「基本データの共通記述方法」の「連絡先」を参照。
施設の所在		住所	◎	施設の所在の住所を記載。記述は「基本データの共通記述方法」の「住所等」を参照。
		緯度	○	施設の所在の経度を記載。記述は「基本データの共通記述方法」の「緯度、経度」を参照。

中分類	小分類	項目名	L	説明
		経度	○	施設の所在の経度を記載。記述は「基本データの共通記述方法」の「緯度、経度」を参照。
		標高		施設の所在の標高を記載。
		団体名		サービス提供者の団体名を記載。施設の管理者と同一の場合もある。 施設内のテナントや個々のサービスを列記する場合は、施設データセットのサブセットとしてサービス ID を付与した施設サービスのデータセットを整備する。サービスからはサービスが提供される施設 ID を参照することで関係を示す。
		法人番号		サービス提供者の法人番号を記載。
		種別		施設の主要なサービスを記載。住宅、店舗等、事務所等、ホテル・旅館、遊戯施設・風俗施設、公共施設・病院・学校等、工場・倉庫等と対応するが、複数兼ねる場合もある。店舗等は飲食、物品販売、サービス提供に分けても良い。
		POI コード		施設の POI コードを記載。
		災害時対応		災害などの非常時に提供可能な機能を記載。避難場所、避難所、給水拠点、その他支援のうち該当するものを選択。指定緊急避難場所は指定緊急避難場所のサブセットを用いて詳細を定義する。
		ライフイベント		「妊娠・出産」「子育て」「学校教育」「結婚・離婚」「引っ越し・住まい」「就職・退職」「高齢者・介護」「ご不幸」から該当するものを選択。
		日常生活		「ごみ・水道・生活インフラ」「衛生と食の安全」「健康・医療」「交通機関と駐車・駐輪」「文化・スポーツ」「生涯学習・市民活動」「まちづくり(市民協働)」「障害者支援」「税金」「国保・年金」「届出・証明」「ビジネス」から該当するものを選択。
		緊急時		「救急」「消防」「防犯」「防災」「生活保護」から該当するものを選択。
		外部利用可否		可能、条件付き可能、不可の区分を記載。
		定員		施設全体の定員を整数で記載。
		バリアフリー情報		バリアフリー情報を記載。
		駐車場情報		「交通」データモデルの駐車場・駐輪場を参照。
		利用可能曜日	○	利用可能な曜日を記載。記述方法は「基本データの共通記述方法」を参照。
		利用可能曜日特記事項		利用可能曜日についての特記事項・例外（祝日、年末年始の変更点など）等があれば記載。
		開始時間		開始時間を記載。時刻の記述方法は「基本データの共通記述方法」の「日時」を参照。
		終了時間		終了時間を記載。時刻の記述方法は「基本データの共通記述方法」の「日時」を参照。

中分類	小分類	項目名	L	説明
		利用可能日時特記事項		利用可能曜日、開始時間、終了時間についての特記事項・例外（祝日、年末年始の変更点など）等があれば記載。
		時間外における対応		利用時間外における対応内容を記載。
		料金(基本)		利用するために必要な基本料金を日本円で記載。(1 円単位)
		料金(詳細)		利用するために必要な各種料金を日本円で記載。(1 円単位)
		アクセス方法		施設への公共交通や車でのアクセス方法を記載。
		出入口		4.出入口で施設のサブセットとして定義する。
		URL	○	施設の URL を記載。
		説明		施設の説明を記載
		備考		
	中期稼働状況			稼働中、一時閉鎖中、閉鎖の区分を記載。「基本データの共通記述方法」を参照。
	短期稼働状況	営業時間		営業中、営業時間外の区分を記載。利用時間と現在時刻あるいは指定した時刻から機械的に表示されることを想定するが、実装方法はデータ処理の設計に委ねる。
		利用可否		ある時点における利用可能、利用不可の区分を記載する。定員と予約状況から機械的に表示されることを想定するが、実装方法はデータ処理の設計に委ねる。
施設の関連情報	設備	設備データセット ID		施設の提供する設備は設備のデータモデルで記述し、設備ごとに設備が所在する施設の施設 ID を付与して関係を示す。設備データセットにはデータセットの ID を付与し、施設に紐付く設備を参照できるようにする。
	サービス	サービスデータセット ID		施設の提供するサービスが複数に渡る場合は、同一施設 ID に紐付くサービスのデータセットとして定義し、施設のサービスのデータセット ID を付与することで関係を示す。
	関連施設	関連施設データセット ID		施設の関連施設は施設のデータモデルで記述し、施設ごとに関連する施設の施設 ID を付与して関係を示す。関連施設データセットにはデータセットの ID を付与し、施設に紐づく関連施設を参照できるようにする。

(3)施設共通の列挙型コード

(a)用途

建築基準法の「用途地域」分類[95]から選択する。

表 7-6-3-3　用途の分類

住宅	住宅、共同住宅、寄宿舎、下宿
店舗等	日用品販売店舗、喫茶店、理髪店、建具屋等のサービス業用店舗

[95] https://www.toshiseibi.metro.tokyo.lg.jp/kanko/area_ree/youto_seigen.pdf

	物販販売店舗、飲食店、損保代理店、銀行の支店・宅地建物取引業者等のサービス業用店舗
事務所等	
ホテル、旅館	
遊戯施設・風俗施設	
公共施設・病院・学校等	
工場・倉庫等	

(b)POI コード

Point of Interest を示す POI コードには行政基本情報データ連携モデルの POI コード[96]を使用する。

(c)サービス対象者がサービスを求める場面

スーパーシティデータ連携基盤では、住民や訪問者などの対象者が必要とする場面に応じたサービスを提供するためのデータ処理が求められる。これを容易にするため、ユニバーサルメニュー・マップを推奨する。

表 7-6-3-4 ユニバーサルメニュー・マップに基づく列挙型選択肢(対象者がおかれた場面別)

ライフイベント	「妊娠・出産」「子育て」「学校教育」「結婚・離婚」「引っ越し・住まい」「就職・退職」「高齢者・介護」「ご不幸」から該当するものを選択。
日常生活	「ごみ・水道・生活インフラ」「衛生と食の安全」「健康・医療」「交通機関と駐車・駐輪」「文化・スポーツ」「生涯学習・市民活動」「まちづくり(市民協働)」「障害者支援」「税金」「国保・年金」「届出・証明」「ビジネス」[97]から該当するものを選択。
緊急時	「救急」「消防」「防犯」「防災」「生活保護」から該当するものを選択。

(4)施設共通の推奨オプション

施設の特性に応じて、アクセシビリティ、子育て支援、外国語対応のオプションを付加する。対応可否を「可=1」、「不可=0」、「不明、未確認=空白」で記載する。

- 全ての施設においてアクセシビリティオプションがあることが望ましい。
- 子どもが来る可能性がある施設には子ども子育て支援オプションがあることが望ましい。
- 外国人が利用する可能性のある施設には外国語オプションがあることが望ましい。

(a)アクセシビリティオプション

アクセシビリティについて情報提供する場合には、以下のデータ項目を使用する。

表 7-6-3-5 アクセシビリティのデータ項目

分類	データ項目
移動に関する支援	車椅子可

[96] 政府 CIO ポータル「データ連携モデル」https://cio.go.jp/guides

[97] ユニバーサルメニュー・マップは生活者としての市民を想定しているが、スーパーシティではビジネスに関するニーズを満たすサービスも考えられるため、本書では「ビジネス」を追加した。

	車椅子貸出
	ツエ貸出
	多目的トイレ
	スロープ、エレベーター、エスカレーター
視覚に関する支援	点字ブロック等の移動支援
	点字や読上による支援
	盲導犬・介助犬同伴
聴覚に関する支援	字幕
	筆談対応
	介助犬・聴導犬同伴
駐車場	優先駐車場
その他の支援	オストメイト対応トイレ
備考	自由記述

※備考は自由記述項目。

(b)子育て支援オプション

子育て支援情報について情報提供する場合には以下のデータ項目を使用する。

表 7-6-3-6　子育て支援情報のデータ項目

分類	データ項目
子供預かり	子供預かり（無料）
	子供預かり（有料）
	最少年齢
	最大年齢
	子供預かり開所時間
	子供預かり閉所時間
設備	授乳室
	おむつ替えコーナー
ベビーカー	ベビーカー貸出
	ベビーカー利用
備考	自由記述

※備考は自由記述項目。

(c)外国語オプション

外国語対応情報について情報提供する場合には以下のデータ項目を使用する。

表 7-6-3-7　外国語対応情報のデータ項目

データ項目	説明
言語	2 文字言語コードを使用する。
会話	説明や会話による対応。
資料	外国語資料、説明表示による対応。
備考	要予約など。

複数言語ある時には、言語ごとにデータを作成する。

(5)既存のデータモデル等

国土数値情報の「公共施設」を参照。https://nlftp.mlit.go.jp/ksj/index.html

(6)施設のサービス別サブセット

7-6-3 施設のデータセットをサービス別に整備する際、7-6-3 施設データモデルと組み合わせるデータモデルのサブセットを例示する。ここではスマートシティでニーズが高いと考えられる公共、医療、教育、社会福祉などを対象とした。

(a)公共機関

行政機関が運営する施設。博物館、図書館などの文化施設も含む。

(ｱ)列挙型コード

1)POI には行政データ連携標準の POI コードを使用する。

2)国土数値情報の平成 27 年度版「公共施設小分類コード」98を施設の基本属性に記載する。

(ｲ)既存データモデル

1)国土数値情報「公共施設」が参照できる。

https://nlftp.mlit.go.jp/ksj/gml/datalist/KsjTmplt-P02-v4_0.html

2)Schema.org　Schema.org の「CivicStructure」が参照できる。

https://schema.org/CivicStructure

(b)医療機関

医療機関に関する情報。薬局なども含む。

(ｱ)データモデル

表 7-6-3-2　施設の基本データモデル(基本形)に以下を付加して利用する。診療時間は「施設＞サービス＞利用時間」のモデルを使って記述する。

[98] 平成 27 年度　公共施設小分類コード https://nlftp.mlit.go.jp/ksj/gml/codelist/PubFacMiclassCd_wf.html

表 7-6-3-8　医療機関のデータ項目

データ項目	説明
医療機関区分	医療機関の種類を記載。(ｲ)を参照。
医療機関番号	医療機関の医療機関番号を記載。(ｲ)を参照。
時間外における対応	医療機関の診療時間外における対応内容を記載。
診療科目	医療機関の診療科目を「;」（半角のセミコロン）区切りで記載。(ｲ)を参照。
病床数	医療機関の病床数を記載。

災害対応情報として、以下の項目を付加することも考えられる。

表 7-6-3-9　災害対応情報に関するデータ項目

データ項目		説明
災害拠点分類		災害拠点病院等の分類を記載。
状況		医療機関の状況を記載。

(ｲ)列挙型コード

1)医療機関区分

厚生労働省が示す医療施設の区分[99]を参照する（診療所は、オープンデータ推奨データセットに従い、無床、有床に分割)。

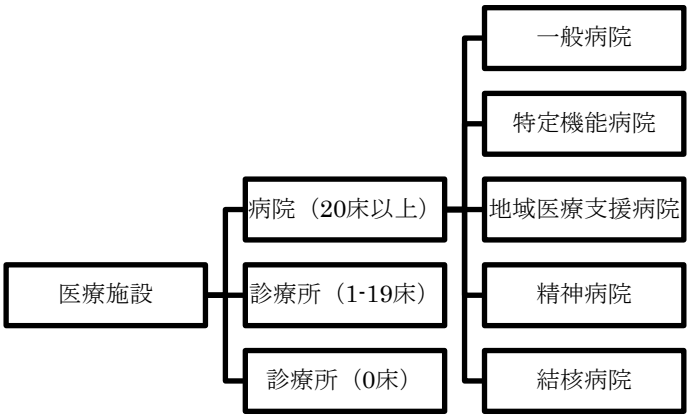


図 7-6-3-1　医療施設の区分

2)診療科目

診療科区分（厚生労働省 様式コード表）のコード診療科目は診療科区分（厚生労働省 様式コード表）のコード[100]を使用する。

[99] https://www.mhlw.go.jp/wp/hakusyo/kousei/10-2/kousei-data/PDF/22010206.pdf

[100] https://www.mhlw.go.jp/bunya/iryouhoken/iryouhoken15/dl/other1-1-17.pdf

表 7-6-3-10　診療科区分のコード番号

診療科目	コード番号	診療科目	コード番号	診療科目	コード番号
内科	010	整形外科	120	眼科	230
心療内科	020	形成外科	130	耳鼻咽喉科	240
精神科	030	美容外科	140	気管食道科	250
神経科	040	脳神経外科	150	リハビリテーション科	260
呼吸器科	050	呼吸器外科	160	放射線科	270
消化器科	060	心臓血管外科	170	神経内科	280
循環器科	070	小児外科	180	胃腸科	290
アレルギー科	080	皮膚泌尿器科	190	皮膚科	300
リウマチ科	090	性病科	200	泌尿器科	310
小児科	100	肛門科	210	産科	320
外科	110	産婦人科	220	婦人科	330
呼吸器内科	340	代謝内科	440	大腸肛門科	540
循環器内科	350	内分泌内科	450	眼形成眼窩外科	550
歯科	360	救急医学科	460	不妊内分泌科	560
歯科矯正科	370	血液科	470	膠原病リウマチ内科	570
小児歯科	380	血液内科	480	脳卒中科	580
歯科口腔外科	390	麻酔科	490	腫瘍治療科	590
糖尿病科	400	消化器内科	500	総合診療科	600
腎臓内科	410	消化器外科	510	乳腺甲状腺外科	610
腎移植科	420	肝胆膵外科	520	新生児科	620
血液透析科	430	糖尿内科	530	小児循環器科	630

3)医療機関番号

地方厚生局のコード内容別医療機関一覧表の医療機関番号を使用する

- 北海道厚生局
  https://kouseikyoku.mhlw.go.jp/hokkaido/iryo_shido/hoken-kikan.html
- 東北厚生局
  https://kouseikyoku.mhlw.go.jp/tohoku/gyomu/gyomu/hoken_kikan/itiran.html
- 関東信越厚生局 https://kouseikyoku.mhlw.go.jp/kantoshinetsu/chousa/shitei.html
- 東海北陸厚生局
  https://kouseikyoku.mhlw.go.jp/tokaihokuriku/gyomu/gyomu/hoken_kikan/shitei.html
- 近畿厚生局 https://kouseikyoku.mhlw.go.jp/kinki/tyousa/shinkishitei.html
- 中国四国厚生局
  https://kouseikyoku.mhlw.go.jp/chugokushikoku/chousaka/iryoukikanshitei.html
- 四国厚生支局
  https://kouseikyoku.mhlw.go.jp/shikoku/gyomu/gyomu/hoken_kikan/shitei/
- 九州厚生局
  https://kouseikyoku.mhlw.go.jp/kyushu/gyomu/gyomu/hoken_kikan/index.html

4)災害拠点分類

「期間災害医療センター」「地域災害医療センター」「救護所」を記載する。

(ｳ)既存データモデル等

1)厚生労働省各地方局の「保険医療機関・保険薬局の指定等一覧」

厚生労働省各地方局が毎月「保険医療機関・保険薬局の指定等一覧」を更新している。

2)国土数値情報の「医療機関」

国土数値情報の医療機関が参照できる。

https://nlftp.mlit.go.jp/ksj/gml/datalist/KsjTmplt-P04-v2_1.html

3)Schema.org

Schema.org の「Hospital」のデータ項目が参照できる。https://schema.org/Hospital

(c)教育機関の施設

教育機関の施設に関する情報を、施設ごとに定義する。塾など民間教育機関は事業所のデータモデルに含まれるので、このデータモデルには含まない。

(ｱ)参照データモデル

表 7-6-3-11　教育機関の施設の参照データモデル

名称	英語名称	参照元	参照 url
学校	School	文部科学省	https://www.mext.go.jp/b_menu/toukei/mext_01087.html

(ｲ)データモデル

(2)施設の基本データモデル(基本形)を使用し、以下文部科学省「学校コード」のデータモデルの項目を追加する。学校コードを基本属性の ID として設定することが望ましい。また、全ての教育機関施設は、極力(2)施設の基本データモデル(基本形)に示した「災害時対応」を記載することが望ましい。

表 7-6-3-12　教育機関の施設のデータモデル

項目	説明
学校コード	学校コードを記入する
学校種	A1：幼稚園 A2：幼保連携型 認定こども園 B1：小学校 C1：中学校 C2：義務教育学校 D1：高等学校 D2：中等教育学校 E1：特別支援学校 F1：大学 F2：短期大学 G1：高等専門学校 H1：専修学校 H2：各種学校
都道府県番号	学校コードの都道府県番号の規則にならう。01-47
設置区分	1：国立 2：公立 3：私立
本分校	1：本校 2：分校 9：廃校
学校名	設置者名は入れないことを原則とするが、学校コード上の学校種、都道府県番号及び設置区分が同一となる､同じ名称の学校が同一都 道府県内に存在する場合（例：「公立」となる県立と市立で同じ名称の高校が存在する場合）には、区別を容易にするために設置者名を含めて記載（例：「○○市立○○高等学校」と

項目	説明
	記載）することを原則とする。
属性情報設定年月日	YYYY-MM-DD
属性情報廃止年月日	YYYY-MM-DD
旧学校調査番号	学校コードへの移行前に当該学校に設定されていた都道府県番号-学校調査番号を記載する。ただし、大学、短期大学、高等専門学校に ついては、都道府県番号を含めず学校調査番号のみを記載する。
移行後の学校コード	現行の学校コードを廃止した上で別の学校コードに移行する場合に本データを設定する。

(ｳ)既存のデータセット・データモデル等

1)Schema.org

Schema.org の「EducationalOrganization」が参照できる。

https://schema.org/EducationalOrganization

(d)社会福祉法人(介護サービス事業者含む)

介護サービス事業所及び介護関連事業者の情報。

(ｱ)参照データモデル

表 7-6-3-13　社会福祉法人（介護サービス事業者含む）の参照データモデル

名称	英語名称	参照元	参照 url
介護サービス事業所	Nursing home	推奨データセット	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)施設の基本データモデル(基本形)に、以下を付加して利用する。

表 7-6-3-14　社会福祉法人（介護サービス事業者含む）のデータモデル

データ項目	L	説明
実施サービス	◎	実施サービスを記載。(ｳ)を参照。
事業所番号		介護サービス事業所の事業所番号を記載。
定員	○	サービスの定員を記載。

(ｳ)列挙型コード

1)実施サービス

実施サービスは、推奨データセットに記載された以下の項目を選択する。複数サービスを提供している場合は「;」で区切り列挙する。

- 居宅介護支援
- 訪問介護（ホームヘルプ）

- 訪問入浴
- 訪問看護
- 訪問リハビリ
- 夜間対応型訪問介護
- 定期巡回・随時対応型訪問介護看護
- 通所介護（デイサービス）
- 通所リハビリ
- 地域密着型通所介護
- 療養通所介護
- 認知症対応型通所介護
- 小規模多機能型居宅介護
- 複合型サービス（看護小規模多機能型居宅介護）
- 短期入所生活介護（ショートステイ）
- 短期入所療養介護
- 介護老人福祉施設（特別養護老人ホーム）
- 介護老人保健施設（老健）
- 介護療養型医療施設
- 特定施設入居者生活介護（有料老人ホーム・軽費老人ホーム等）
- 認知症対応型共同生活介護（グループホーム）
- 地域密着型介護老人福祉施設入所者生活介護
- 地域密着型特定施設入居者生活介護
- 福祉用具貸与
- 特定福祉用具販売

(e)子育て施設

幼稚園、保育園、認定こども園を記述するためのデータモデルである。

(ｱ)参照データモデル

表 7-6-3-15　子育て施設の参照データモデル

名称	英語名称	参照元	
子育て施設	Child support facility	推奨データセット	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)施設の基本データモデル(基本形)に、オープンデータの推奨データセットの「子育て施設」から以下を付加して利用する。

表 7-6-3-16　子育て施設のデータモデル

データ項目	L	説明
認可等年月日		認可又は認定をもらった時の年月日を記載。
収容定員	○	収容できる定員数を記載。
受入年齢	○	利用可能年齢を記載。
一時預かりの有無	○	一時預かりがあるかどうかを記載。

また、以下の項目も追加が検討できる。

表 7-6-3-17　子育て施設のデータモデルの追加項目

データ項目	L	説明
病児保育		病児保育の可否

(ｳ)列挙型コード

1)POI コード

施設項目の POI コードに、以下から選択したコードを記入する。

授乳室、保育園、児童館（児童センター)、学童保育クラブ、認定こども園、幼稚園、その他の社会福祉施設

(ｴ)事業所

支店や営業所など、経済活動が行われている場所[101]ごとの単位で事業所を記述するためのデータモデルである。

(ｱ)参照データモデル

表 7-6-3-18　事業所の参照データモデル

名称	英語名称	参照元	
事業所	Business facility	gBizInfo	https://info.gbiz.go.jp/

[101] 経済センサスでは以下のように定義している。
「経済活動が行われている場所ごとの単位で、原則として次の要件を備えているものをいう。
一定の場所（1 区画）を占めて、単一の経営主体のもとで経済活動が行われていること。
従業者と設備を有して、物の生産や販売、サービスの提供が継続的に行われていること。
「出向・派遣従業者のみの事業所」は当該事業所に所属する従業者が一人もおらず、他の会社など別経営の事業所から出向又は派遣されている人のみで経済活動が行われている事業所をいう。「事業内容等不詳の事業所」は事業所として存在しているが、回答不備等で事業内容が不明の事業所をいう。」
https://www.stat.go.jp/data/e-census/2016/kekka/k_yougo.html#e01

(ｲ)データモデル

(2)施設の基本データモデル(基本形)を使用し、gBizInfo のデータモデルを参照する。

表 7-6-3-19　事業所のデータモデル

項目名	L	説明
事業所 ID		事業所の ID を記載。(現在未定義の予約領域)
事業所名	◎	事業所名称を記載。
事業所名_カナ		事業所名称のカナを記載。
事業所名_英語		事業所名称の英字を記載。
事業所所在地	◎	事業所所在地を記載。記述は「基本データの共通記述方法」を参照。
業種		事業所が主に提供するサービスの業種コードを記載。0 日本産業分類を参照。
法人番号	○	法人番号を記載。
法人名	◎	法人名を記載。

(ｳ)列挙型コード

日本産業分類　https://www.stat.go.jp/data/e-census/2016/kekka/bunrui.html

表 7-6-3-20　日本産業分類(大分類を抜粋)[102]

A 農業，林業
B 漁業
C 鉱業，採石業，砂利採取業
D 建設業　★
E 製造業
F 電気・ガス・熱供給・水道業　★
G 情報通信業　☆
H 運輸業，郵便業　★
I 卸売業，小売業
J 金融業，保険業　★
K 不動産業，物品賃貸業
L 学術研究，専門・技術サービス業
M 宿泊業，飲食サービス業
N 生活関連サービス業，娯楽業
O 教育，学習支援業　☆
P 医療，福祉

[102] 事業所単位の売上（収入）金額の把握ができない産業については、産業分類項目名の右側に★、構成する大分類に事業所単位の売上(収入)金額の把握ができる産業とできない産業の両方がある場合は、大分類項目名に☆が付与されている。https://www.stat.go.jp/data/e-census/2016/kekka/bunrui.html

Q 複合サービス事業　☆ R サービス業（他に分類されないもの）　☆

(ｴ)既存のデータセット・データモデル等

1)事業所母集団データベース（ビジネスレジスター)

http://www.stat.go.jp/data/jsdb/index.html

2)gBizInfo　　https://info.gbiz.go.jp/

(g)食品等営業許可・届出事業所

食品等事業者から都道府県、各保健所設置市、特別区の衛生主管部局食品衛生担当課へ提出される食品等営業許可・届出をデータセットとして活用する際のデータモデルである。

(ｱ)参照データモデル

表 7-6-3-21　食品等営業許可・届出事業所の参照データモデル

名称	英語名称	参照元	
食品等営業許可・届出事業所	Food shop	推奨データセット	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)施設の基本データモデル(基本形)に以下を付加して利用する。

表 7-6-3-22　食品等営業許可・届出事業所のデータモデル

分類	データ項目	L	説明
基本属性	名称、ID、用途	◎	(2)施設の基本データモデル(基本形)で施設ごとに記載る。
管理者	法人名	○	営業許可申請の申請主体が法人の場合、法人名を記載。営業許可申請が個人の場合は記載不要。
	法人番号		営業許可申請の申請主体が法人の場合、法人番号を記載。営業許可申請が個人の場合は記載不要。
	連絡先		(2)施設の基本データモデル(基本形)で施設ごとに記載。
所在	住所、方書		(2)施設の基本データモデル(基本形)で施設ごとに記載。
食品等事業者のサービス	営業の種類	◎	食品衛生法施行令第 35 条にて定める 34 の営業の分類と、各地方公共団体における条例で定めた分類の中から該当するものを記載。(ｳ)を参照。
	業態	○	営業施設の業態に係る情報で、「営業の種類」の項目より詳細な情報を記載。(「自動販売機」、「移動販売」、「臨時営業」等の情報も本項目に記載。)
認可	許可番号	◎	営業許可申請に対する各地方公共団体で管理する許可番号を記載。
	初回許可年月日	○	初回に営業許可を出した時の年月日を記載。記述方法は「基本データの共通記述方法」の「日時」を参照。
	許可年月日	◎	直近で営業許可を出した時の年月日を記載。記述方法は「基本データ

分類	データ項目	L	説明
			の共通記述方法」の「日時」を参照。
	許可開始日	○	営業許可開始日の年月日を記載。記述方法は「基本データの共通記述方法」の「日時」を参照。
	許可満了日	◎	営業許可満了日の年月日を記載。記述方法は「基本データの共通記述方法」の「日時」を参照。
	廃業年月日	○	廃業届を受理した際の廃業日の年月日を記載。記述方法は「基本データの共通記述方法」の「日時」を参照。
	申請区分	○	受理した申請の区分を記載。「新規」「変更」「継続」「廃業」
	許可条件		営業許可にあたり、付けている条件がある場合、その内容を記載。

(ｳ)列挙型コード

1)食品衛生法施行令第 35 条[103]

営業の種類を以下の中から選択する。

①飲食店営業、②喫茶店営業、③菓子製造業、④あん類製造業、⑤アイスクリーム類製造業、⑥ 乳処理業、⑦特別牛乳搾取処理業⑧ 乳製品製造業、⑨集乳業⑩ 乳類販売業、⑪食肉処理業、⑫食肉販売業、⑬食肉製品製造業、⑭魚介類販売業、⑮魚介類せり売営業、⑯魚肉ねり製品製造業⑰食品の冷凍又は冷蔵業⑱食品の放射線照射業、⑲ 清涼飲料水製造業、⑳乳酸菌飲料製造業、㉑氷雪製造業、㉒氷雪販売業㉓食用油脂製造業、㉔マーガリン又はショートニング製造業、㉕みそ製造業、㉖しょう油製造業、㉗ソース類製造業、㉘酒類製造業、㉙豆腐製造業、㉚納豆製造業、㉛めん類製造業、㉜そうざい製造業、㉝缶詰又は瓶詰製造業、㉞添加物製造業

https://www.mhlw.go.jp/content/11121000/000349946.pdf

(h)公園

公園を記述するためのデータモデルである。

(ｱ)参照データモデル

表 7-6-3-23　公園の参照データモデル

名称	英語名称	参照元	
公園	Park	推奨データセット	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)施設の基本データモデル(基本形)に以下を付加して利用する。

[103] 食品衛生法改正に伴い 2021 年 6 月 1 日より以下の 32 業種に変更される。1 飲食店営業、2 調理の機能を有する自動販売機により調理し、調理された食品を販売する営業、3 食肉販売業、4 魚介類販売業、5 魚介類競り売り営業、6 集乳業、7 乳処理業、8 特別牛乳搾取処理業、9 食肉処理業、10 食品の放射線照射業、11 菓子製造業、12 アイスクリーム類製造業、13 乳製品製造業、14 清涼飲料水製造業、15 食肉製品製造業、16 水産製品製造業、17 氷雪製造業、18 液卵製造業、19 食用油脂製造業、20 みそ又はしょうゆ製造業、21 酒類製造業、22 豆腐製造業、23 納豆製造業、24 麺類製造業、25 そうざい製造業、26 複合型そうざい製造業、27 冷凍食品製造業、28 複合型冷凍食品製造業、29 漬物製造業、30 密封包装食品製造業、31 食品の小分け業、32 添加物製造業

表 7-6-3-24　公園のデータモデル

データ項目	L	説明
目的		「体を動かす」「自然に触れる」「スポーツをする」「施設を利用する」や自由記述の特徴語を;区切りで列挙。
公園種別		国土数値情報「都市公園」公園種別コードから選択する。0 を参照。
特徴		「芝生広場」、「水遊び」、「ボート」、「森」、「歴史」、「庭園」や自由記述の特徴語を;区切りで列挙。
主な遊具		「健康遊具」「滑り台」「ブランコ」「鉄棒」「ジャングルジム」「複合遊具」「砂場」や自由記述の特徴語を;区切りで列挙。
主な植物		「アジサイ」「イチョウ」「ウメ」「サクラ」「スイセン」「チューリップ」「ツツジ」「バラ」「フジ」「モミジ」や自由記述の特徴語を;区切りで列挙。
スポーツ		「ジョギング」「テニス」「野球」「サッカー」「バスケットボール」「アスレチックコース」「プール」「ゲートボール」や自由記述の特徴語を;区切りで列挙。
面積		公園の面積（m2）
設備		「レストラン」「売店」「トイレ」「集会場」や自由記述の特徴語を;区切りで列挙。
禁止事項		「火気、球技」「ペット」「自転車」や自由記述の特徴語を;区切りで列挙。

(ｳ)列挙型コード

1)国土数値情報

「都市公園」公園種別コードから選択する。

1 街区公園
2 近隣公園
3 地区公園（カントリーパーク）
4 総合公園
5 運動公園
6 広域公園
7 レクリエーション都市
8 国営公園
9 特殊公園（風致公園、動植物公園、歴史公園、墓園）
10 緩衝緑地
11 都市緑地
12 緑道
13 都市林
14 広場公園

(ｴ)既存データモデル等

1)国土数値情報

「都市公園」のデータモデルが参照できる。

https://nlftp.mlit.go.jp/ksj/gml/datalist/KsjTmplt-P13.html

2)Schema.org

Park のモデルが参照できる。https://schema.org/Park

3)PARKFUL

公園専門メディア PARKFUL は遊びや運動などの区分を提示している。

https://search.parkful.net/

(i)指定緊急避難場所

市区町村から提供される指定緊急避難場所の情報を記述するデータモデルである。

(ｱ)参照データモデル

表 7-6-3-25　指定緊急避難場所の参照データモデル

名称	英語名称	参照元	
指定緊急避難場所	Evacuation place	推奨データセット	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)施設の基本データモデル(基本形)に以下を付加して施設ごとに記載する。

表 7-6-3-26　指定緊急避難場所のデータモデル

データ項目	L	説明
災害種別_洪水	◎	指定緊急避難場所が対応している災害(洪水)を記載。
災害種別_崖崩れ、土石流及び地滑り	◎	指定緊急避難場所が対応している災害(崖崩れ、土石流及び地滑り)を記載。
災害種別_高潮	◎	指定緊急避難場所が対応している災害(高潮)を記載。
災害種別_地震	◎	指定緊急避難場所が対応している災害(地震)を記載。
災害種別_津波	◎	指定緊急避難場所が対応している災害(津波)を記載。
災害種別_大規模な火事	◎	指定緊急避難場所が対応している災害(大規模な火事)を記載。
災害種別_内水氾濫	◎	指定緊急避難場所が対応している災害(内水氾濫)を記載。
災害種別_火山現象	◎	指定緊急避難場所が対応している災害(火山現象)を記載。
指定避難所との重複	◎	指定避難所との重複している施設の際に記載。
想定収容人数	○	指定緊急避難場所に収容可能な人数を記載。
対象となる町会・自治会	○	指定緊急避難場所へ避難する対象の地域等を「;」(半角のセミコロン) 区切りで記載。

※災害種別は該当する場合は「1」、該当しない場合は「0」を記入。未確認の場合は空白

(j)交番

交番の情報を記述するためのデータモデルである。

(ｱ)参照データモデル

表 7-6-3-27　交番の参照データモデル

名称	英語名称	参照元
交番	Police station	

(ｲ)データモデル

(2)施設の基本データモデル(基本形)を使って記載する。

表 7-6-3-28　交番のデータモデル

分類	項目名		項目の説明
基本属性	ID		交番に一意の ID。(2)施設の基本データモデル(基本形)で施設ごとに記載する。
	名称	◎	交番の名称。
	名称(カナ)		交番のカナ名称
	名称(英語)		交番の英語名称
用途	行政サービス拠点種別		交番の POI コード「1212」を記入。
管理者	警察署	◎	警察署名を記載。
	連絡先		(2)施設の基本データモデル(基本形)施設ごとに記載する。
所在	住所	◎	(2)施設の基本データモデル(基本形)で施設ごとに記載する。
受持区	地域表記		受持区を;で区切って列記する。
	ポリゴン		地域のポリゴンのファイルがある場合にはファイル名を記入。
	備考		市区町、町丁字の一部の場合には、それぞれの項目で少しでも含まれている場合には記入の上、この項目で、一部である解説をする。

(ｳ)既存のデータセット・データモデル等

国土数値情報「警察署データ」

https://nlftp.mlit.go.jp/ksj/gml/datalist/KsjTmplt-P18.html

(k)入浴施設

公衆浴場などの入浴施設情報を記述するデータモデルである。

(ｱ)参照データモデル

表 7-6-3-29　入浴施設の参照データモデル

名称	英語名称	参照元	
入浴施設	Bath	推奨データセット「観光施設」	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)施設の基本データモデル(基本形)に以下を付加して利用する。

表 7-6-3-30　入浴施設のデータモデル

項目名	項目の説明
湯の種類	温泉、鉱泉、炭酸泉、湯
サービス内容	露天風呂、無料シャンプー・ソープ、タオルレンタル、サウナ、水風呂、

	休息所、ランドリー、wifi

(1)クリーニング施設

(ｱ)参照データモデル

表 7-6-3-31　クリーニング施設の参照データモデル

名称	英語名称	参照元	
クリーニング施設	Cleaning	推奨データセット「公共施設」	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)施設の基本データモデル(基本形)に以下を付加して利用する。

表 7-6-3-32　クリーニング施設のデータモデル

項目名	項目の説明
サービス内容	「クリーニング」、「コインランドリ」「クリーニング;コインランドリ」から選択する。
コインランドリ台数	台数を記入する。
洗剤	販売の有無

7-6-4.　出入口

建物や公園等の出入口、駐車場、車両搬入口等の情報を記述するためのデータモデルである。設備のデータモデルを参照する。ただし 7-6-5 設備と異なり、施設や建物に付随する設備の一部として定義されることが想定される。

(1)参照データモデル

表 7-6-4-1　出入口の参照データモデル

名称	英語名称	参照元	
出入口	Gate	IMI 共通語彙基盤	https://imi.go.jp/ns/core/Core242.html#ic：設備型

(2)データモデル

7-6-5(2)設備の基本データモデルに以下を追加して記述する。

表 7-6-4-2　出入口のデータモデル

データ項目	L	説明
対象		7-6-13 移動オブジェクトの列挙型コードから該当する対象を選択して記述する。人、乗用車、トラック、バス、農耕車両、工事車両、オートバイ、自転車、車椅子、動力付きスクーター、UAV (Unmanned aerial vehicle)

データ項目	L	説明
出入口情報		入口、出口、出入口から選択する。
エスカレーター		人の出入口の場合のエスカレーター有無
エレベーター		人の出入口の場合のエレベーター有無
アクセシビリティ情報		人の出入口の場合のスロープ、点字ブロック等の情報を記入する。
車高制限		車の出入口の場合の車高制限
車幅制限		車の出入口の場合の車幅制限
備考		出入口に関する関連情報を記入する。

GTFS の STOP を使用する場合には、Location_type に出入口を表す「2」を指定する。

7-6-5. 設備

施設や建物に備わっている設備を記述するためのデータモデルである。施設や建物の記述の一部として定義される場合と、設備単体のデータセットとして定義される場合がある。施設や建物に固定されていることが多いが、可搬な設備を扱うことも考えられる。

(1)設備共通の参照データモデル

表 7-6-5-1 設備共通の参照データモデル

名称	英語名称	参照元	参照 url
設備	Equipment	IMI 共通語彙基盤	https://imi.go.jp/ns/core/Core242.html#ic:設備型

(2)設備の基本データモデル

表 7-6-5-2 設備共通の基本データモデル

中分類	小分類	項目名	L	説明
基本属性		名称	◎	設備の名称を記載
		名称_カナ	◎	設備の名称をカナで記載
		名称_通称		設備の通称を記載
		名称_英語		設備の名称を英語で記載
	ID	ID		データセット内で設備が一意に決まるよう、ID を設定し記載。既存のデータセットに ID がある場合は流用しても良い。
設備の管理者		団体名		設備の管理団体の名称を記載
		法人番号		設備の管理者の法人番号を記載
	連絡先	連絡先		設備の管理者の電話番号を記載。記述は「基本データの共通記述方法」の「連絡先」を参照。
設備の設置者		団体名		管理者と異なる場合、設備の設置者の名称を記載。
		法人番号		設備の設置者の法人番号を記載

中分類	小分類	項目名	L	説明
	連絡先	連絡先		設備の設置者の電話番号を記載。記述は「基本データの共通記述方法」の「連絡先」を参照。
設備の所在	施設	施設 ID	◎	設備が設置されている施設の施設 ID を記載する。施設の所在、サービス、管理者はデータ連携により取得することを想定するが、データ処理の設計に委ねる。
		住所	◎	施設内に設置された設備ではない場合、設備が所在する住所を記載する。記述は「基本データの共通記述方法」を参照。
		緯度		施設内に設置された設備ではない場合、設備が所在する住所を記載する。記述は「基本データの共通記述方法」を参照。
		経度		施設内に設置された設備ではない場合、設備が所在する住所を記載する。記述は「基本データの共通記述方法」を参照。
	設置場所	設置位置		設置位置の詳細を記載する。
設備の機能	主要サービス	種別		設備の主要なサービス機能を記載。衛生(便器・風呂桶など)、衛生(給水)、衛生(排水)、厨房、熱源・エネルギー、電気、通信、空気調和、搬送、消防・防災、警備・防犯・監視、その他から該当する機能を選択。
		POI コード		設備の POI コードを記載する。
		ライフイベント		「妊娠・出産」「子育て」「学校教育」「結婚・離婚」「引っ越し・住まい」「就職・退職」「高齢者・介護」「ご不幸」から該当するものを選択。
		日常生活		「ごみ・水道・生活インフラ」「衛生と食の安全」「健康・医療」「交通機関と駐車・駐輪」「文化・スポーツ」「生涯学習・市民活動」「まちづくり(市民協働)」「障害者支援」「税金」「国保・年金」「届出・証明」「ビジネス」から該当するものを選択。
		緊急時		「救急」「消防」「防犯」「防災」「生活保護」から該当するものを選択。
		外部利用可否		可能、条件付き可能、不可の区分を記載。
		バリアフリー情報		バリアフリーオプションを参照。
		駐車場情報		交通の駐車場・駐輪場を参照。
		定員		設備単位の定員を整数で記載する。
	認可等年月日	認可等年月日		設備が認可された日付を年月日で記載。記述方法は「基本データの共通記述方法」の「日時」を参照。
	利用時間	利用可能曜日		利用可能な曜日を記載。時刻の記述方法は「基本データの共通記述方法」の「日時」を参照。
		利用可能曜日特記事項		利用可能曜日についての特記事項・例外（祝日、年末年始の変更点など）等があれば記載。
		開始時間		開始時間を記載。時刻の記述方法は「基本データの共通記述方法」の「日時」を参照。

中分類	小分類	項目名	L	説明
		終了時間		終了時間を記載。時刻の記述方法は「基本データの共通記述方法」の「日時」を参照。
		利用可能日時特記事項		利用可能曜日、開始時間、終了時間についての特記事項・例外（祝日、年末年始の変更点など）等があれば記載。
		時間外における対応		利用時間外における対応内容を記載。
	利用方法	料金(基本)		利用するために必要な基本料金を日本円で記載。(1 円単位)
		料金(詳細)		利用するために必要な各種料金を日本円で記載。(1 円単位)
		アクセス方法		設備への公共交通や車でのアクセス方法を記載。
	説明	URL		設備の URL を記載。
		説明		設備の説明を記載
		画像		画像の格納先 URL を記載。
		画像_ライセンス		画像に対するライセンスについて記載。
		備考		
	中期稼働状況			稼働中、一時閉鎖中、閉鎖の区分を記載する。
	短期稼働状況	営業時間		営業中、営業時間外の区分を記載する。利用時間と現在時刻あるいは指定した時刻から機械的に表示されることを想定するが、実装方法はデータ処理の設計に委ねる。
		利用可否		利用可能、利用不可の区分を記載する。定員と予約状況から機械的に表示されることを想定するが、実装方法はデータ処理の設計に委ねる。

(3)設備共通の列挙型コード

(a)POI コード

Point of Interest を示す POI コードには行政基本情報データ連携モデルの POI コード[104]を使用する。

(b)サービス対象者がサービスを求める場面

スーパーシティデータ連携基盤では、住民や訪問者が必要とする場面に応じたサービスを提供するためのデータ処理が求められる。これを容易にするため、ユニバーサルメニュー・マップを使用することが望ましい。

表 7-6-5-3　ユニバーサルメニュー・マップに基づく列挙型選択肢(シーン別)

ライフイベント	「妊娠・出産」「子育て」「学校教育」「結婚・離婚」「引っ越し・住まい」「就職・退職」「高齢者・介護」「ご不幸」から該当するものを選択。

[104] 政府 CIO ポータル「データ連携モデル」https://cio.go.jp/guides

日常生活	「ごみ・水道・生活インフラ」「衛生と食の安全」「健康・医療」「交通機関と駐車・駐輪」「文化・スポーツ」「生涯学習・市民活動」「まちづくり(市民協働)」「障害者支援」「税金」「国保・年金」「届出・証明」「ビジネス」[105]から該当するものを選択する。
緊急時	「救急」「消防」「防犯」「防災」「生活保護」から該当するものを選択。

(ｱ)Wikipedia の分類

衛生(便器・風呂桶など)、衛生(給水)、衛生(排水)、厨房、熱源・エネルギー、電気、通信、空気調和、搬送、消防・防災、警備・防犯・監視、その他の分類が示されている。

(4)設備のサービス別サブセット

(a)文化財

文化財の情報を記述するデータモデルである。

(ｱ)参照データモデル

表 7-6-5-4　文化財の参照データモデル

名称	英語名称	参照元	
文化財		推奨データセット	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)設備の基本データモデルに以下を付加して利用する。

表 7-6-5-5　文化財のデータモデル

項目名	説明
種類	文化財の種類を記載。(ｳ)1)を参照。
文化財分類	文化財の分類を「;」(半角のセミコロン) 区切りで記載。(ｳ)2)を参照。
員数（数）	当該文化財内の員数を数値部分のみ記載。
員数（単位）	当該文化財内の員数の単位を記載。
文化財指定日	文化財として指定された指定日を記載。記述方法は「基本データの共通記述方法」の「日時」を参照。

[105] ユニバーサルメニュー・マップは生活者としての市民を想定しているが、スーパーシティではビジネスに関するニーズを満たすサービスも考えられるため、本書では「ビジネス」を追加した。

(ｳ)列挙型コード

1)文化財の種類

表 7-6-5-6　文化財の種類

種類	備考
建造物	有形文化財の場合に指定
美術工芸品	有形文化財の場合に指定
芸能	無形文化財の場合に指定
工芸技術	無形文化財の場合に指定
その他	無形文化財の場合に指定

2)文化財種類

表 7-6-5-7　文化財種類

文化財分類	備考
重要文化財	有形文化財の場合に指定
国宝	有形文化財の場合に指定
登録有形文化財	有形文化財の場合に指定
重要無形文化財	無形文化財の場合に指定
記録作成等の措置を講ずべき無形文化財	無形文化財の場合に指定
重要有形民俗文化財	有形民俗文化財の場合に指定
登録有形民俗文化財	有形民俗文化財の場合に指定
重要無形民俗文化財	無形民俗文化財の場合に指定
記録作成等の措置を講ずべき無形の民俗文化財	無形民俗文化財の場合に指定
史跡	記念物の場合に指定
特別史跡	記念物の場合に指定
名勝	記念物の場合に指定
特別名勝	記念物の場合に指定
天然記念物	記念物の場合に指定
特別天然記念物	記念物の場合に指定
登録記念物	記念物の場合に指定
重要文化的景観	文化的景観の場合に指定
伝統的建造物群保存地区	伝統的建造物群の場合に指定
重要伝統的建造物群保存地区	伝統的建造物群の場合に指定

(b)AED

(ｱ)参照データモデル

表 7-6-5-8　AED の参照データモデル

名称	英語名称	参照元	
AED	AED	推奨データセット	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)設備の基本データモデルを利用する。

(c)公衆無線 LAN アクセスポイント

(ｱ)参照データモデル

表 7-6-5-9　公衆無線 LAN アクセスポイントの参照データモデル

名称	英語名称	参照元	
公衆無線 LAN アクセスポイント	Wifi	推奨データセット	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)設備の基本データモデルに以下を付加して利用する。

表 7-6-5-10　公衆無線 LAN アクセスポイントのデータモデル

データ項目	L	説明
SSID		公衆無線 LAN の SSID を記載。
提供エリア		公衆無線 LAN の設置施設の施設内の提供エリアを記載。
URL		公衆無線 LAN の設置施設の Web サイトの URL を記載。

(d)公衆トイレ

(ｱ)参照データモデル

表 7-6-5-11　公衆トイレの参照データモデル

名称	英語名称	参照元	
公衆トイレ	Public Toilet	推奨データセット	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)設備の基本データモデルに以下を付加して利用する。

表 7-6-5-12　公衆トイレのデータモデル

データ項目	L	説明
車椅子使用者用トイレ	◎	車椅子使用者用トイレの設置有無を記載。有(1)無(0)
乳幼児用設備設置トイレ	◎	乳幼児用設備設置トイレの設置有無を記載。有(1)無(0)
オストメイト設置トイレ	◎	オストメイト設置トイレの設置有無を記載。有(1)無(0)

(ｳ)既存データモデル等

https://schema.org/PublicToilet

(e)消防水利施設

(ｱ)参照データモデル

表 7-6-5-13　消防水利施設の参照データモデル

名称	英語名称	参照元	
消防水利施設	Water supply for fire	推奨データセット	https://cio.go.jp/policy-opendata

(ｲ)データモデル

(2)設備の基本データモデルに以下を付加して利用する。内閣官房が公表するオープンデータ：推奨データセットを参照する。

表 7-6-5-14　消防水利施設のデータモデル

データ項目	L	説明
種別	◎	消防水利施設の種類を記載。(ｳ)を参照。
口径	○	消防水利施設が対応する口径をミリメートル単位で記載。
備考		特記事項があれば記載。

(ｳ)列挙型コード

1)オープンデータの推奨データセット

以下から選択する。

- 消火栓
- 私設消火栓
- 防火水そう
- プール
- 河川・溝等
- 濠・池等
- 海・湖

- 井戸
- 下水道
- その他

7-6-6. 道路

(1)道路

道路のデータモデルは、道路名（路線名）、道路管理者名、位置など最小限の地図表示が目的のもの、起点終点、最小幅員幅、通行方向なども含んだ経路検索が目的のもの、勾配など 3 次元形状を含む自動運転支援が目的のもの等、目的に合わせて様々な要素が取り扱われている。ここでは、地図表示を目的とする単純な道路のデータモデルについて検討した。

(a)データモデル

(ｱ)国土数値情報

国土数値情報の道路データを参照することで、国土交通省の整備する全国の道路と同じレベルの表現が可能となる。詳細は別紙ウ-3 参照。

https://nlftp.mlit.go.jp/ksj/gmlold/datalist/gmlold_KsjTmplt-N01.html

(2)交通規制情報

交通規制情報は、事故、工事、イベントなどにより随時発生し、都市情報として重要である。

公益財団法人日本道路交通情報センターの公開する「一般道路の「交通規制情報(11 種別)」[106]を参照する。

No. 規制種別
1 歩行者用道路
2 自転車用道路
3 自転車及び歩行者用道路
4 通行止め
5 車両通行止め
6 指定方向外進行禁止
7 一方通行
8 一時停止
9 信号機
10 ゾーン 30
11 環状交差点

[106] https://www.jartic.or.jp/d/opendata/202102010000/typeA_kisei_11.pdf

7-6-7. その他の地物

地上にある施設や建物、土地などで、なんらかの管理や運営がなされている対象を記述するために用意されたデータモデルである。IMI 共通語彙基盤コア語彙では地物型を継承して施設、建物、土地型が定義されているため、記述対象がこれらに相当する場合はそれらのデータモデルを利用した方が使いやすいことが多い。

(1)地物共通の参照データモデル

表 7-6-7-1 地物共通の参照データモデル

名称	英語名称	参照元	
その他の地物	Feature	IMI コア語彙	https://cio.go.jp/policy-opendata

(2)地物の基本データモデル

施設の基本モデルを利用する。

(3)地物のサービス別サブセット

(a)観光ポイント

石碑、岬等を、共通語彙基盤を参照してモデル化する。

(ｱ)参照データモデル

表 7-6-7-2 観光ポイントの参照データモデル

名称	英語名称	参照元
観光ポイント		推奨データセット

(ｲ)データモデル

施設の基本モデルに POI コードを付加して利用する。

(b)位置

なにか位置を示したい時に記載

7-6-8. 交通

(1)交通施設　空港・港湾・駅・バス停

(a)参照データモデル

表 7-6-8-1　交通施設の参照データモデル

名称	英語名称	参照元
空港	Airport	schema.org
港湾	Port	GTFS
駅	Station	GTFS
バス停	Bus Stop	GTFS

(b)データモデル

施設の基本モデルに以下を付加して利用する。GTFS を参照する。

表 7-6-8-2　交通施設のデータモデル

データ項目	GTFS フィールド名	必須	説明
場所タイプ	location_type	必須	空港、港湾、駅、バス停の区分を記載する。
親 ID	parent_station	任意	stops.txt で定義されている様々な場所間の階層を定義する。親である場所の ID が含まれる。
車いす対応	wheelchair_boarding	任意	車椅子での乗車が可能かどうかを記載する。
階 ID	level_id	任意	場所の階。互いに連結されていない複数の駅では、同じ階の ID を使用することができる。
プラットフォーム番号	platform_code	任意	駅などのプラットフォームや乗り場の情報。

(c)既存データモデル

(ｱ)https://schema.org/TrainStation

(ｲ)https://schema.org/BusStation

(ｳ)https://schema.org/BusStop

(2)鉄道路線・バス路線

(a)参照データモデル

表 7-6-8-3　鉄道路線・バス路線の参照データモデル

名称	英語名称	参照元
鉄道路線、バス路線	rout	GTFS

(b)データモデル

GTFS を参照する。

表 7-6-8-4　鉄道路線・バス路線のデータモデル

データ項目名	GTFS フィールド名	必須	説明
路線 ID	route_id	必須	路線 ID を指定
路線正式名	route_long_name	必須	路線の正式名称
通称	route_short_name	任意	路線に通称がある場合に記入
運行会社 ID	agency_id	任意	運行会社の法人番号
運行会社名		必須	運行会社名称
説明	route_desc	任意	路線の説明。
路線タイプ	route_type	必須	路線で使用される輸送手段のタイプ
URL	route_url	任意	ウェブページの URL。

(c)列挙型コード

(ｱ)GTFS のルートタイプ

鉄道、バス、フェリー等を表す経路タイプは、欧州全域 TPEG 規格の階層型車種（HVT）コードを参照して検討されている GTFS のルートタイプを使用する。

0 - 路面電車、市街電車、ライトレール。都市圏内を運行する軽量軌道交通又は路面電車システム。

1 - 地下鉄、メトロ。都市圏内を運行する地下鉄システム。

2 - 電車。都市間又は長距離の鉄道路線。

3 - バス。短距離及び長距離のバス路線。

4 - フェリー。短距離及び長距離の航路。

5 - ケーブルトラム。ケーブルが車両の下を通る、路面を走る鉄道車両（例: サンフランシスコのケーブルカー)。

6 - 索道（例: リフト、ケーブルカー、ゴンドラ、ロープウェイ）1 本又は複数本のケーブルに吊り下げた客車、車両、ゴンドラ、椅子で移動する交通機関。

7 - 鋼索鉄道。急斜面用に設計された全ての鉄道システム。

11 - トロリーバス。道路上空の架線から棹状の集電装置で集めた電気を動力として走るバス。

12 - モノレール。1 本のレール又は桁を軌道とする鉄道。

(3)タクシー乗り場

https://schema.org/TaxiStand

(a)参照データモデル

表 7-6-8-5　タクシー乗り場の参照データモデル

名称	英語名称	参照元
タクシー乗り場	Taxi stand	Schema.org

(b)データモデル

7-6-5(2)設備の基本データモデルに以下を付加して利用する。

表 7-6-8-6　タクシー乗り場のデータモデル

データ項目	Schema.org フィールド名	必須	説明
乗り場 ID	Identifier	必須	駅、停車場を識別する ID を指定
名称	stop_name	必須	駅や停車場の名前
説明	stop_desc	任意	駅や停車場に関する説明
配車時間	openingHours	任意	配車時間。

(4)駐車場

(a)参照データモデル

表 7-6-8-7　駐車場の参照データモデル

名称	英語名称	参照元
駐車場	Off street Parking	スマートシティモデル

(b)データモデル

7-6-5(2)設備の基本データモデルに以下を付加して利用する。共通語彙基盤を参照してモデル化する。

表 7-6-8-8　駐車場のデータモデル

データ項目		説明
契約期間		ic:駐車場種別で記述する。時間貸し、月極め、日貸し、年払い、提起契約などの選択肢が考えられる。
収容台数		駐車場の収容台数を記載する。ic:収容台数を参照する。
設置場所		設置場所を記載する。平面、立体、タワー、地下、パレット式などが考えられる。

(c)既存データモデル

OMA を参照する

(ｱ)スマートシティ・データモデル

車種、車両条件、料金、平置き等を詳細に機技術するためのデータ項目が提供されている。

https://github.com/smart-data-models/dataModel.Parking/blob/master/OffStreetParking/doc/spec.md

(ｲ)Schema.org

https://schema.org/ParkingFacility

(5)駐車スペース（路上）

(a)参照データモデル

表 7-6-8-9　駐車スペース（路上）の参照データモデル

名称	英語名称	参照元	
駐車スペース（路上）	OnStreetParking	スマートシティデータセット	https://github.com/smart-data-models/dataModel.Parking/blob/master/OnStreetParking/doc/spec.md

(b)データモデル

7-6-5(2)設備の基本データモデルに許容時間を付加して利用する。OMA を参照する

(6)駐輪場

(a)参照データモデル

表 7-6-8-10　駐車スペース（路上）のデータモデル

名称	英語名称	参照元	
駐輪場	Bike station	Schema.org (Parking Facility を参照)	https://cio.go.jp/policy-opendata

(b)データモデル

7-6-5(2)設備の基本データモデルを利用する。OMA を参照する。

https://schema.org/ParkingFacility

(7)シェアードカーステーション

OMA を参照する。

(8)レンタル自転車スポット

(a)参照データモデル

表 7-6-8-11　レンタル自転車スポットの参照データモデル

名称	英語名称	参照元	
レンタル自転車スポット	BikeHireDockingStation	スマートデータモデル	https://github.com/smart-data-models/dataModel.Transportation/blob/master/BikeHireDockingStation/doc/spec.md

(b)データモデル

7-6-3(2)施設の基本データモデル(基本形)に以下を追加して利用する。

表 7-6-8-12　レンタル自転車スポットのデータモデル

データ項目	smart-data-models フィールド名	説明
運営組織	provider	レンタルバイク運営組織（ブランド）
利用可能エリア	areaServed	レンタルバイク等の利用可能エリア
空きスロット数	freeSlotNumber	レンタルバイク等を返却するための空きスロット数
利用可能バイク数	availableBikeNumbe	レンタルバイク等の利用可能バイク数
故障スロット数	outOfServiceSlotNumber	故障スロット数
全スロット数	totalSlotNumber	レンタルバイクステーションの全スロット数
データ更新時間	dateModified	利用状況データのデータ更新時間
所有者	owner	所有者

(c)既存データモデル

https://github.com/smart-data-models/dataModel.Transportation/blob/master/BikeHireDockingStation/doc/spec.md

(9)ガソリンスタンド、水素ステーション

(a)参照データモデル

表 7-6-8-13　ガソリンスタンド、水素ステーションの参照データモデル

名称	英語名称	参照元	
ガソリンスタンド、水素ステーション	Fuel station	Schema.org（gasStation） スマートデータモデル （EVstation を参照）	https://schema.org/GasStation https://github.com/smart-data-models/dataModel.Transportation/blob/master/EVChargingStation/doc/spec.md

(b)データモデル

7-6-3(2)施設の基本データモデル(基本形)に以下を追加して利用する。

表 7-6-8-14　ガソリンスタンド、水素ステーションのデータモデル

データ項目	smart-data-models フィールド名
ブランド	network
事業所名	name
別名	alternateName
対象車種	allowedVehicleType
利用可能車両数	availableCapacity
最大車両数数	capacity
燃料	chargeType
支払方法	acceptedPaymentMethod
連絡先	contactPoint
データ更新時間	dateModified
運営者	operator
所有者	owner

(c)既存データモデル

https://schema.org/GasStation

https://github.com/smart-data-models/dataModel.Transportation/blob/master/EVChargingStation/doc/spec.md

(10)EV スタンド

(a)参照データモデル

表 7-6-8-15　EV スタンドの参照データモデル

名称	英語名称	参照元	
EV スタンド	EV stand	smart-data-models	https://github.com/smart-data-models/dataModel.Transportation/blob/master/EVChargingStation/doc/spec.md

(b)データモデル

7-6-3(2)施設の基本データモデル(基本形)に以下を追加して利用する。

表 7-6-8-16　EV スタンドの参照データモデル

データ項目	smart-data-models フィールド名
ID	id
ブランド	network
事業所名	name

データ項目	smart-data-models フィールド名
別名	alternateName
説明	description
対象車種	allowedVehicleType
利用可能車両数	availableCapacity
最大車両数数	capacity
充電タイプ	chargeType
支払方法	acceptedPaymentMethod
ソケット数	socketNumber
ソケットタイプ	socketType
電圧	voltage
最大アンペア数	amperage
連絡先	contactPoint
データ更新時間	dateModified
運営者	operator
所有者	owner

(c)既存データモデル

https://github.com/smart-data-models/dataModel.Transportation/blob/master/EVChargingStation/doc/spec.md

7-6-9. イベント

催事やセミナーといったイベント情報を記述するためのデータモデル。典型的なイベントとは、主催者がいて、ある目的を持って日時と場所を決めて行われるできごとである。1 回しか行われないイベント、週 1 回等定期に行われるイベント、不定期に繰り返し行われるイベントなどが含まれる。

(1)イベント共通の参照データモデル

表 7-6-9-1 イベント共通の参照データモデル

名称	英語名称	参照元	
イベント	Event	推奨データセット	https://cio.go.jp/policy-opendata

(2)イベントの基本データモデル

表 7-6-9-2 イベント共通の基本データモデル

中分類	小分類	項目名	説明
基本属性	名称	イベント名	イベントの名称を記載。
		イベント名_カナ	イベントの名称をカナで記載。
		イベント名_英語	イベントの名称を英語で記載。
	ID	ID	データセット内でイベントが一意に決まるよう、ID を設定し

中分類	小分類	項目名	説明
			記載。既存のデータセットに ID がある場合は流用しても良い。
	開催方法	開催方法	オンライン、オフライン、併催の区分を記載。
	規模	規模	予想されるイベントの参加人数を記載。
		規制情報	イベントに伴う規制(道路、船舶、施設など)に関する情報へのリンクとなる情報を記載。7-6-6(1)(a)で定義されている場合は規制情報 ID を記載。
管理者	基本属性	主催者	イベントの主催者の法人名を記載。
		法人番号	イベントの主催者の法人番号を記載。
		連絡先	イベントの主催者の連絡先を記載。記述は「基本データの共通記述方法」の「連絡先」を参照。
関係者		共催者	イベントの共催者を列挙。
		後援者	イベントの後援者を列挙。
実施主体		名称	イベントの実施を管理する主体が主催者と異なる場合、その名称を記載。
		法人番号	イベントの実施を管理する主体が主催者と異なる場合、その法人番号を記載。
		連絡先	イベントの実施を管理する主体が主催者と異なる場合、その連絡先を記載。記述は「基本データの共通記述方法」の「連絡先」を参照。
サービス	説明	説明	イベントの説明を記載。
		URL	イベントの説明が掲載されているウェブサイトの URL を記載。
	開催時間	開始日	イベントの開始日を記載。日付の記述方法は「基本データの共通記述方法」の「日時」を参照。
		終了日	イベントの終了日を記載。日付の記述方法は「基本データの共通記述方法」の「日時」を参照。
		開始時間	イベントの開始時間を記載。時刻の記述方法は「基本データの共通記述方法」の「日時」を参照。
		終了時間	イベントの開始時間を記載。時刻の記述方法は「基本データの共通記述方法」の「日時」を参照。
		開始日時特記事項	開始日、終了日、開始時間、終了時間についての特記事項・例外等があれば記載。
	所在	住所	イベントの開催場所の所在の住所を記載。記述は「基本データの共通記述方法」の「住所等」を参照。
		場所名称	イベント会場の名称を記載。
		所在の区分	集合場所、開催場所、解散場所が異なる場合は、該当する所在の区分を記載する。集合場所、開催場所、解散場所ごとに所在と場所名称を記載する。
	参加方法	料金(基本)	イベントに参加するために必要な基本料金を日本円で記載。無料の場合は 0 と記述する。(1 円単位)
		料金(詳細)	イベントに参加するために必要な各種料金を日本円で記載。(1 円単位)

中分類	小分類	項目名	説明
		料金の説明	イベントの参加料金に関する説明を記載。
		料金支払方法	
		予約の要否	予約要否の区分を記載。7-5-3(3)を参照。
		予約方法	予約が必須又は可能な場合、予約方法を記載。
		予約状況	7-5-3(3)を参照。
		参加資格	参加に資格が必要な場合に説明を記載する。
		バリアフリー情報	バリアフリーオプションを参照。
		駐車場情報	「交通」データモデルの駐車場・駐輪場を参照。
	対象者	定員	イベントの定員を整数で記載する。
		目的などの区分	「子ども・子育て」「観光・文化・芸術」「スポーツ・教養」「産業・事業者向け」「健康・福祉」「環境・地域活動」から該当するものを選択。
状況(ステータス)	中期稼働状況	開催区分	開催の有無や延期などを記載。(開催、中止、延期、一部変更)
		実施条件	天候や集客状況等により開催区分が変更される場合は説明を記載する。
	短期稼働状況	開催区分	開催期間中などに短期の状況を記載する。7-5-3(1)(b)を参照。
		混雑状況	混雑状況を記載。7-5-3(2)を参照。

(3)既存のデータモデル等

(a)しずみち.info「大型イベント基本情報」

http://opendata-api-wiki-dot-shizuokashi-road.appspot.com/resourcelist/bigEvents

(4)イベントのサービス別サブセット

(a)大規模イベント

表 7-6-9-3　大規模イベントのサブセット

データ項目	説明
規制情報	イベントに伴う規制(道路、船舶、施設など)に関する情報へのリンクとなる情報を記載する。

(b)オンラインイベント

表 7-6-9-4　オンラインイベントのサブセット

データ項目	説明
配信サービス	ウェブ会議、動画配信などのサービス名を記載。
対応媒体	スマートフォン、PC、ブラウザ、専用アプリケーションなどの対応媒体を列挙する。
配信 URL	配信 URL を記載する。

データ項目	説明
開催後視聴	可能(1)、不可(0)を記載する。

7-6-10. センサーデータ

センサーデータは、センサーデバイスを利用した観測活動によって得られるデータである。

観測活動は、センサーデバイスに搭載されるセンサーを利用して、観測対象の有する観測特性を測定する行為の集合を意味する。スーパーシティにおいては、先端的なサービスを実現するために複数の観測活動が実施されることが想定される。

観測活動には、1 つ以上の観測対象が存在する。観測対象の有する複数の観測特性に対して、観測活動の目的に応じて定義されるスコープに基づき、1 つ以上のセンサーデバイスを利用して、センサーデバイスに搭載された 1 つ以上のセンサーによって、対象となる観測特性を測定する。

下に観測活動の概念図を示す。センサーによって測定された観測特性は、データに変換され、センサーデバイスから送信、収集されデータセットとなる。

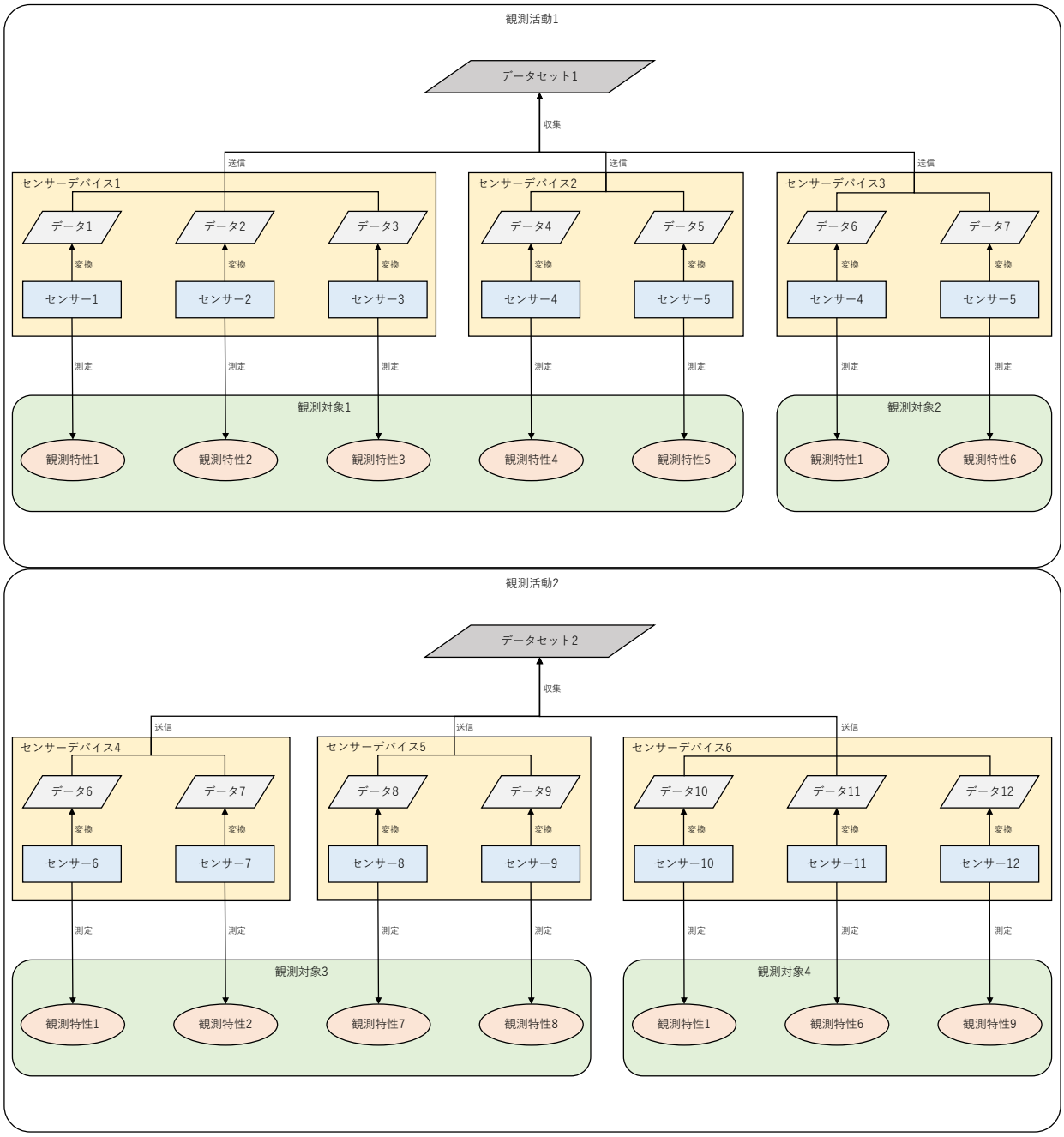


図 7-6-10-1　観測活動の概念図

観測活動の種類や目的に応じて、複数の観測対象に対する観測対象とセンサーデバイス、センサーの組み合わせは同一の場合もあれば、観測対象によって観測対象とセンサーデバイス、センサーの組み合わせが異なる場合もある。

図に示した例では、観測特性 1 は全ての観測対象で共通して測定される特性であり、観測特性 2 は観測対象 1、3 で共通して測定される特性、観測特性 6 は観測対象 2、4 で共通して測定される特性、観測特性 3、4、5、7、8、9 は各観測対象特有の特性となっている。

例えば、河川に対する観測活動では、河川を流れる水や、護岸といった観測対象がそれぞれ複数箇所存在する。観測対象には、水位、水温、流速、pH、濁度、残存酸素、護岸侵食量、構造物変異などの観測特性がある。観測活動の目的が防災のための河川氾濫状況のモニタリングとすると、水位と流速を計測するセンサーを搭載したセンサーデバイスを氾濫のリスクの高い箇所に設置してモニタリングするといったケースが想定される。

観測活動の概念図に示したように、観測活動の構成要素は、センサーを中心として双方向に配置される関係性を有する。したがって、センサーは異なっても観測特性は同一のこともあり、必ずしもセンサーとデータ、観測特性は 1 対 1 対 1 で対応するとは限らない。実装の際には、これらの要素が疎結合となっていることに留意が必要である。ただし、観測特性の重複を認めることにより、下図のように、観測活動＞観測対象＞センサーデバイス＞センサー＞観測特性＞データ　の階層構造でこれを表現することもできる。

図 7-6-10-2　観測特性に重複を認めた場合の階層構造の例

以上を踏まえてセンサーデバイス、センサーデータのデータモデルを示す。

(1)センサーデータ共通の参照データモデル

表 7-6-10-1　センサーデータ共通の参照データモデル

名称	英語名称	参照元	
センサーデータ	Sensing data	推奨データセット	https://cio.go.jp/policy-opendata

(2)データモデル

7-6-5 設備のデータモデルを基に必要なデータ項目を追加した。

(a)センサーデバイス

表 7-6-10-2　センサーデバイスのデータモデル

中分類	小分類	項目名	説明
	名称	名称	デバイスの名称を記載。
		名称_カナ	デバイスの名称をカナで記載。
		名称_通称	デバイスの通称を記載。
		名称_英語	デバイスの名称を英語で記載。
	ID	ID	データセット内でデバイスが一意に決まるよう、ID を設定し記載。既存のデータセットに ID がある場合は流用しても良い。
		認定情報	国や地方公共団体の定める各種制度に適合し、認定を受けたデバイスである場合、その認定情報を記載。
		認可等年月日	デバイスが認可された日付を年月日で記載。記述方法は「基本データの共通記述方法」の「日時」を参照。
	仕様	色	デバイスの色を記載。
		ブランド名	デバイスのブランド名を記載。
		メーカー名	デバイスの製造元名を記載。
		モデル名	デバイスのモデル名を記載。
		シリアルナンバー	製造者によって割り当てられたシリアルナンバーを記載。
デバイスの管理者	基本属性	団体名	デバイスの管理団体の名称を記載。
		法人番号	デバイスの管理者の法人番号を記載。
	連絡先		デバイスの管理者の電話番号を記載。記述は「基本データの共通記述方法」の「連絡先」を参照。
デバイスの設置者	基本属性	団体名	管理者と異なる場合、デバイスの設置者の名称を記載。
		法人番号	デバイスの設置者の法人番号を記載。
	連絡先		デバイスの設置者の電話番号を記載。記述は「基本データの共通記述方法」の「連絡先」を参照。
デバイスの所在	施設	施設 ID	デバイスが設置されている施設の施設 ID を記載する。施設の所在、サービス、管理者はデータ連携により取得することを想定するが、データ処理の設計に委ねる。
	設置場所	設置位置	設置位置の詳細を記載する。
		住所	施設に設置されたデバイスではない場合、デバイスの所在の住所を記載。記述は「基本データの共通記述方法」の「住所等」を参照。
		緯度	施設に設置されたデバイスではない場合、デバイスの所在の経度を記載。記述は「基本データの共通記述方法」の「緯度、経度」を参照。
		経度	施設に設置されたデバイスではない場合、デバイスの所在の

中分類	小分類	項目名	説明
			経度を記載。記述は「基本データの共通記述方法」の「緯度、経度」を参照。
		標高	デバイスの所在の標高を記載。
	説明	説明	デバイスの機能や性能、仕様などについての説明を記載。
		ソフトウェア名称	デバイスにインストールされているソフトウェア（ファームウェア）の名称を記載。
		ソフトウェアバージョン	デバイスにインストールされているソフトウェア（ファームウェア）のバージョンを記載。
		処理能力	RFC7228 で定義される処理能力のクラス。「C0」「C1」「C2」から該当するものを選択。詳細は RFC 7228 - Terminology for Constrained-Node Networks を参照。
		瞬時消費電力計測値	デバイスの瞬時消費電力を W で記載。
		積算消費電力計測値	デバイスの積算消費電力を 0.001kWh で記載。
		デバイス関連文書	デバイスに関連する文書について、文書名称と Web 空間上の場所を示す URL を記載。
		構成情報	デバイスの構成情報を記載。
		電気通信番号	電気通信事業者を識別するコード。Mobile Network Code (MNC) 。「00」～「99」の中から該当するものを選択。日本国外の場合 Mobile Country Code (MCC) も必要。
		プロトコル	デバイスの通信プロトコルを記載。どの層のプロトコルを対象とするか別途定義が必要。
		MAC アドレス	デバイスの MAC アドレスを記載。
		信号強度	ワイヤレスデバイスの信号強度を dBm で記載。
		設定	デバイスの技術的な設定。本モデルでカバーできないデバイスの構成についての属性を記載。
	利用期間	設置日	デバイスを設置した日時。記述方法は「基本データの共通記述方法」の「日時」を参照。
		初回仕様日	デバイスが初めて使用された日時。記述方法は「基本データの共通記述方法」の「日時」を参照。
		インストール日	デバイスがインストールされた日時。記述方法は「基本データの共通記述方法」の「日時」を参照。
		キャリブレーション日	デバイスが最後にキャリブレーションされた日時。記述方法は「基本データの共通記述方法」の「日時」を参照。
		製造日	デバイスの製造日。記述方法は「基本データの共通記述方法」の「日時」を参照。
	状態	稼働状態	デバイスの ON/OFF の状態。「ON」「OFF」から該当するものを選択。稼働状態について別の定義が必要なセンサーデバイスの場合は別途考慮が必要。
		異常発生状態	異常内容が示す異常状態の発生の有無。「異常発生有」「異常発生無」から該当するものを選択。

中分類	小分類	項目名	説明
		異常内容	異常内容のコードを記載。別途異常内容の具体的な定義が必要。例えば、「ECHONET 機器オブジェクト詳細規定 2．5 異常内容プロパティ」を参照。
		バッテリー残量	バッテリーが満充電の場合 1.0、バッテリーが空の場合 0.0 としてバッテリーの残量を記載。
センサー	名称	名称	センサーの名称を記載。
		名称_カナ	センサーの名称をカナで記載。
		名称_通称	センサーの通称を記載。
		名称_英語	センサーの名称を英語で記載。
	ID	ID	データセット内でセンサーが一意に決まるよう、ID を設定し記載。既存のデータセットに ID がある場合は流用しても良い。
	説明	センサー説明	センサーの種類や仕様などについての説明を記載。
		センサー関連文書	製品のカタログや仕様書など関連する文書について、文書名称と Web 空間上の場所を示す URL を記載。
		センサー観測特性	センサーと、それが計測する観測特性との関係を記載。
		測定単位	センサーの観測値の単位系。SI 単位から該当するものを選択。「国際単位系（SI）第 9 版」を参照。
		最小測定範囲値	センサーによって測定することができる最小値を記載。
		最大測定範囲値	センサーによって測定することができる最大値を記載。

(b)センサーデータ

中分類	小分類	項目名	説明
観測活動	名称	観測活動名称	観測活動の名称を記載。
		観測活動名称_カナ	観測活動の名称をカナで記載。
		観測活動名称_通称	観測活動の通称を記載。
		観測活動名称_英語	観測活動の名称を英語で記載。
	ID	観測活動 ID	データセット内で観測活動が一意に決まるよう、ID を設定し記載。既存のデータセットに ID がある場合は流用しても良い。
	説明	観測活動説明	観測活動（観測の対象の特性の値を推定あるいは計算する手続きを実行する行為）の内容についての説明を記載。
		期間	観測対象に対して、観測、作動、サンプリングを行った結果の時間、期間あるいは、瞬間時刻、あるいは複合的な時間を記載。
		頻度	観測対象に対して、観測、作動、サンプリングを行った頻度を記載。
		観測活動関連文書	観測活動に関連する文書について、文書名称と Web 空間上の場所を示す URL を記載。
観測対象	名称	観測対象名称	観測対象の名称を記載。
		観測対象名称_カナ	観測対象の名称をカナで記載。

		観測対象名称_通称	観測対象の通称を記載。
		観測対象名称_英語	観測対象の名称を英語で記載。
	ID	観測対象 ID	データセット内で観測対象が一意に決まるよう、ID を設定し記載。既存のデータセットに ID がある場合は流用しても良い。
	説明	観測対象説明	観測対象（観測により推定されたり計算される特性を持つモノ）の説明。
		観測対象関連文書	観測対象に関連する文書について、文書名称と Web 空間上の場所を示す URL を記載。
		観測対象特性	観測対象と、それが有する観測特性との関係を記載。
観測特性	名称	観測特性名称	観測特性の名称を記載。
	ID	観測特性 ID	データセット内で観測特性が一意に決まるよう、ID を設定し記載。既存のデータセットに ID がある場合は流用しても良い。
	説明	観測特性説明	観測特性（観測対象における観測可能な質、特性、特徴）についての説明。
		観測特性関連文書	観測特性に関連する文書について、文書名称と Web 空間上の場所を示す URL を記載。
関連情報	センサーデバイス	デバイス ID	センサーデバイスはセンサーデバイスのデータモデルで記述し、ID を付与してデータとデバイスの関係を示す。デバイスデータセットにはデータセットの ID を付与し、センサーデータに紐づくデバイスを参照できるようにする。
観測データ	名称	データ名称	観測データの名称を記載
		データ名称_英語	観測データの名称を英語で記載
	ID	データ ID	データセット内で観測データが一意に決まるよう、ID を設定し記載。既存のデータセットに ID がある場合は流用しても良い。
	説明	データ説明	観測データについての説明。
	データ	値	観測値。
		単位	観測値の単位系。SI 単位から該当するものを選択。「国際単位系（SI）第 9 版」を参照。
		検知閾値	異常状態を検出するための閾値。
		検知状態	値と検知閾値の関係から導出される状態。
		日時	観測データのタイムスタンプ。記述方法は「基本データの共通記述方法」の「日時」を参照。
		精度	スマートセンサーによって報告された測定品質指標又は測定品質レベル。

7-6-11. 建物内、地下街

今後の検討課題とする。

7-6-12. 地下埋設物

水道管など、今後の検討課題とする。

7-6-13. 移動オブジェクト

都市を構成する要素として、人や自動車等の移動する対象物を考える必要がある。各対象物には詳細なデータモデルが定義されているが、ここでは、都市内を移動する対象物という観点からデータモデルを整理した。

(1)基本形

(a)参照データモデル

表 7-6-13-1　移動オブジェクトの参照データモデル

名称	英語名称	参照元	
物	Product	Schema.org	https://schema.org/Product

(b)データモデル

表 7-6-13-2　移動オブジェクトのデータモデル

項目	Schema.org	説明
ID	productId	一意に付番された ID
カテゴリ	category	移動オブジェクトのカテゴリ。(c)列挙型コードを参照。
高さ	size	高さ
長さ	size	長さ
幅	size	幅
重量	weight	重量

(c)列挙型コード

人、物、乗り物、その他

(2)人

都市の中の人の特性のデータ項目のみ対象とし、個人情報は扱わない。

(a)参照データモデル

表 7-6-13-3　人の参照データモデル

名称	英語名称	参照元	
人	Person	Schema.org	https://schema.org/Person

(b)データモデル

追加している項目は、移動速度、配慮事項などが異なること、パーソントリップ等の人流データ解析に使うことから追加している。

表 7-6-13-4　人のデータモデル

項目	Schema.org	説明
ID	Identifier	一意に付番された ID
身長	Height	身長を記入。
重量	weight	重量を記入。
性別	gender	性別をコードで記入。(ｱ)を参照。
年齢		年齢を記入。
移動条件	additionalType	障害の有無を記入。
視覚条件	additionalType	障害の有無を記入。
聴覚条件	additionalType	障害の有無を記入。

(c)列挙型コード

(ｱ)性別

国際標準 ISO 5218 に準拠する。

0:不明　1:男性　2:女性　9:適用外

(ｲ)移動、視覚、聴覚条件

アクセシビリティを求める条件を以下のコードで記入する。

0:無　1:有

(d)既存のデータセット・データモデル等(参考)

(ｱ)人の流れ（パーソントリップ）に関するデータモデル

1)「東京大学 CSIS 人の流れデータセット」
https://pflow.csis.u-tokyo.ac.jp/wp-content/uploads/webapi.pdf

2)「交通流動量 パーソントリップＯＤ量データ」
https://nlftp.mlit.go.jp/ksj/jpgis/datalist/KsjTmplt-S05-b.html

(3)乗り物（自動車、オートバイ、自転車、船、飛行物、その他）

陸海空の移動物についてのデータモデル。

(a)参照データモデル

表 7-6-13-5　乗り物（自動車、オートバイ、自転車、船、飛行物、その他）の参照データモデル

名称	英語名称	参照元
乗り物	Vehicle	Schema.org

(b)データモデル

表 7-6-13-6　乗り物（自動車、オートバイ、自転車、船、飛行物、その他）のデータモデル

項目	Schema.org	説明
ID	Identifier	一意に付番された ID
乗り物タイプ	bodyType	乗り物やボディの形状を記入。(ｱ)を参照。
高さ	Height	高さを記入。(m)
長さ	Size	長さを記入。(m)
幅	Width	幅を記入。(m)
重量	weight	重量を記入。(kg)
燃料種別	fuelType	燃料の種類を記載。(ｲ)を参照。

(c)列挙型コード

(ｱ)乗り物タイプ

- 乗用車、トラック、バス
- 農耕車両、工事車両
- オートバイ
- 自転車、車いす、動力付きスクーター
- UAV (Unmanned aerial vehicle)

(ｲ)燃料種別

- 電気
- ガソリン
- ディーゼル
- なし

(4)物

無人搬送機、無人清掃ロボットなどは基本形を活用する。

7-6-14. 自然

植生、降雨範囲など自然データは今後の検討課題とする。

(1)天気

気象庁で国内用として定めた 15 種類を使用する。(国際的には 96 種類が決められている。)

(a)列挙型コード

快晴、晴れ、薄曇り、曇り、煙霧、砂じん嵐、地ふぶき、霧、霧雨、雨、みぞれ、雪、あられ、ひょう、雷

項番	記号	天気	観測者による観測	自動観測
1	○	快晴	○	−
2	⦶	晴	○	○
3	⦷	薄曇	○	−
4	◎	曇	○	○
5	∞	煙霧	○	○
6	⚲	砂じん嵐	○	−
7	+	高い地ふぶき	○	−
8	≡	霧	○	○
9	●	霧雨	○	○
10	⚲	しゅう雨または止み間のある雨	−	○
11	P	降水	−	○
12	●	雨	○	○
13	⚹	みぞれ	○	○
14	✕	雪	○	○
15	∿	着氷性の雨	−	○
16	∿	着氷性の霧雨	−	○
17	△	凍雨	−	○
18	⌓	霧雪	−	○
19	⚲	しゅう雪または止み間のある雪	−	○
20	△	あられ	○	−
21	▲	ひょう	○	○
22	=	もや	−	○
23	↔	細氷	−	○
24	⚡	雷	○	○
注　観測者による観測で該当する天気が複数存在する場合は、項番の大きい方を優先する。				

https://www.data.jma.go.jp/obd/stats/data/mdrr/man/tenki_kigou.html

図 7-6-14-1　天気の列挙型コード

7-6-15. 緊急情報

公共コモンズなどを参照し今後の検討課題とする。

7-6-16. 行政情報

申請データ、証明データを対象に今後の検討課題とする。

7-7. データカタログのデータモデル定義

スーパーシティで多様なステークホルダによるデータ連携を容易にするためには、データセットを検索・特定し取得する前に必要なデータセットのメタデータを入手できなくてはならない。データカタログはこうしたデータセットに内包されないメタデータの交換機能を担う。データ、データモデルとセットで提供され、データモデルで定義できない情報を伝達する。

データやデータセットが生成された目的と異なる用途でのデータ利活用が期待されるスーパーシティで、多様な参加者がデータの価値を見出すためには、極力標準的なメタデータで記述されたデータカタログが求められる。このデータモデルは、データ提供者がデータセットの所在を伝える用途だけでなく、スーパーシティにおいてサービスを提供する事業者が、必要なデータの要件をデータ連携基盤等を通じて要望し、要件に近いデータセットを保有する者が要望に応じるといった用途にも活用され得る。また、データ提供者やデータ所有者が管理すべきデータセットを特定し運用する際にも、データカタログが機能する。

データカタログは W3C DCAT[107]や地理空間に特化した geoDCAT AP[108]、W3C DCAT を拡張して定義された一般社団法人データ社会推進協議会(DSA)が提供する「データカタログ作成ガイドライン V1.1」などの既存データモデルが存在する。本書では DSA「データカタログ作成ガイドライン V1.1」を参照し、スーパーシティのデータ連携に必要と考えられるメタデータを抜粋して説明を加えた[109]。

表 7-7-1　ひとつのデータセットに対するデータカタログのデータモデル概要

大分類	L	小分類	項目名
	◎	基本属性	法人名
			法人番号
		連絡先	連絡先
	◎	ID	ID
		名称	名称
		言語	文字
			文字コード
			使用言語
		時間帯	時間帯
		地域	地域
	◎	説明	説明
		分類テーマ	分類テーマ
		発生源	データセット提供者法人名
			データセット提供者法人番号
			データセットが生成された営み

[107] Data Catalog Vocabulary。国際標準化団体 W3C が定めるデータカタログ標準仕様。

[108] 地理メタデータを表現するための欧州のデータポータル用の DCAT アプリケーションプロファイル。

[109] DSA「データカタログガイドライン V1.1」は複数のデータカタログを共通項目で管理したい場合に利用するデータカタログ本体部、最低限の必要事項を速やかに記入し、他の方に伝えたい場合に利用するデータジャケット部、データを取得した時の状況や方法を詳しく表現したい場合に利用するデータ詳細部、データの外部提供を前提に権利帰属やライセンス、価格を決める場合に利用するデータ利用条件部の合計 144 データ項目から構成されるが、本書ではスーパーシティにおけるサービス提供の観点から重要と思われるものを選び掲載した。https://data-trading.org/public-release/

大分類	L	小分類	項目名
			データセットを生成したシステム
		取り扱いデータ	データ種別
			同意区分
			データの対象
			ID の有無
		提供方法	配信方法
			形式
	○	最新性	データセットの作成日
			データセットの発行日
			データセットの更新日又は修正日
			データセットの提供頻度
			データセットの更新頻度
			データセットの対象期間
			データセットの有効期間
			データセットの対象地域
		データ量	容量
			メタデータ(データ項目)数
			レコード数
		品質指標	データ品質評価
			サービス品質評価
			データ管理プロセス評価
データセットの利用条件	◎	利用条件	利用条件
		データ利用範囲	地域
			法人種別
			用途
		価格及び支払い	価格及び支払い
		取り扱い条件	データ保護要件
			ライセンス
			利用期間

7-7-1. 情報の管理主体

全てのデータセットには、情報の管理主体が存在する。7-1-6 で触れたとおり情報の管理主体はデータ利用者にとって不可欠な情報である。IC:メタデータ型を参照する。

表 7-7-1-1 情報の管理主体(必須)

項目名	説明
法人名	情報の管理主体の法人名を記載。
法人番号	情報の管理主体の法人番号を記載。
連絡先	情報の管理者の連絡先を記載。記述は「基本データの共通記述方法」の「連絡先」を参照。

7-7-2. データセットの基本属性

データセットの基本的な属性を記載するためのデータモデルである。言語や地域、時間帯など値に関するデータ項目はデータセットの利用可否を判断する前提と考えられるため、ここでは基本属性に含めた。

表 7-7-2-1 データセットの基本属性(必須)

小分類	データ項目	説明
ID	ID	データカタログ内でデータセットが一意に決まるよう、ID を設定し記載。
名称	名称	データセットの名称を記載。
言語	文字	日本語のデータセットの場合、常用漢字、JIS X 0213 漢字、住民基本台帳ネットワークシステム統一文字、戸籍統一文字、文字情報基盤のいずれか該当するものを記載。
	文字コード	データセット中のデータ記述に使われている文字コードを記載。
	使用言語	データセット中のデータ記述に使われている言語を記載。
時間帯	時間帯	データセット中に日時や期間が含まれる場合、該当する標準時時間帯を記載。
地域	地域	データセットが生成された地域を記載。記述方法は「地理空間に関するコード」を参照。

7-7-3. データセットの概要

データセットに含まれるデータの概要を記載するデータモデルである。スーパーシティデータ連携基盤では、データセットが生成された業務目的とは異なる目的での再利用が見込まれるため、テーマに加えてデータ発生源や取り扱いデータなどの説明が必要と考えられる。

表 7-7-3-1 データセットの概要(必須)

小分類	データ項目	説明
説明		データセットの特徴を第三者に理解してもらうための説明を記載。
分類テーマ	テーマ	7-6 推奨データモデルの対象分野から該当するものを記載。 土地、建物、施設、設備、道路、地物、交通、イベント、移動オブジェクト、自然、緊急情報、行政情報、地理空間、その他
発生源	データセット提供者法人名	データセットの提供者の法人名を記載。
	データセット提供者法人番号	データセットの提供者の法人番号を記載。
	データセットが生成された営み	データセットが生成された営みについて説明を記載。不動産登記、防火・防災管理講習、○○補助金申請など。
	データセットを生成したシステム	データセットを生成したシステムについて説明を記載。○○市住民記録システム、ボランティア管理システムなど。
取り扱いデータ	データ種別	産業データ、行政データ、パーソナルデータのうち該当するものを選択する。
	同意区分	パーソナルデータが含まれる場合、区分を記載。同意済、同意不要、同意なし、未確認から該当するものを選択する。

	データの対象	文書型、イベント型、サービス型、設備型、人型、組織型、法人型、場所型、地物型、土地型、建物型、施設型、駐車場型、製品型[110]から該当するものを選択する。
	レコード ID の有無	データセットに含まれる個々のレコードを特定する ID が付与されている場合はあり、されていない場合はなし、未確認の場合は空白を記載。
提供方法	配信方法	API、ダウンロードによるファイル提供、個別に定める方法でのファイル提供などの配信方法を記載する。
	形式	ファイル提供の場合、形式を記載する。文書ファイル、アプリケーションファイル、スキーマ、動画、音声、書籍、記事から選択。
地域	データセットの対象地域	データセットが持つレコードの対象地域情報を記載する。記述方法は「地理空間に関するコード」を参照。

7-7-4. データの品質

データの品質に関わる情報[111]は、データセットを入手する前に必要な情報である。とくにスーパーシティのサービスにおいては、データセットの更新頻度や対象とするデータ項目(メタデータ)が連携対象データとして採用するか否かの判断に欠かせないため、データカタログのメタデータとして記述することが望ましい。DTA データカタログガイドラインのデータセット部より、データセットの最新性に関わるデータ項目と説明を記載した。また、データの量に関するデータ項目と説明を記載した。

なお、データに求められる「品質」は用途やコストによって異なり、必ずしも高品質すなわち最善とは限らない。データセットの提供者と提供先が同じ基準であらかじめ合意し、データ提供に先立つ情報の非対称性を削減することが、本モデルの目的である。

表 7-7-4-1 データの品質(推奨)

小分類	データ項目	説明
最新性	データセットの作成日	データセットが作成された日付を年月日で記載。記述方法は「基本データの共通記述方法」の「日時」を参照。
	データセットの発行日	データセットが発行された日付を年月日で記載。記述方法は「基本データの共通記述方法」の「日時」を参照。
	データセットの更新日又は修正日	データセットが更新又は修正された日付を年月日で記載。記述方法は「基本データの共通記述方法」の「日時」を参照。
	データセットの提供頻度	データセットの提供頻度について記載する。記述方法は「基本データの共通記述方法」の「日時」を参照。
	データセットの	データセットの更新頻度について記載する。記述方法は「基本データの共通記述

[110] IMI コア語彙「コア語彙解説書 バージョン 1.0 (2020-02-28 公開)」「3.1 主にデータセットのルートクラスとして用いるクラス用語」より例示。

[111] 本書ではとりあげないが、標準ガイドライン群「データ品質管理ガイドブック」データ品質評価モデルでは、ISO/IEC 25012 に基づき正確性、完全性、一貫性、最新性など 13 の基礎評価指標と効率性、追跡可能性の 2 つの応用評価指標を定義している。併せて ISO/IEC 25024 に基づくデータを使ったサービス実現プロセスの評価指標、ISO/TS 8000-61 に基づくデータ管理プロセスの評価指標とツールを提示している。

小分類	データ項目	説明
	更新頻度	方法」の「日時」を参照。
	データセットの対象期間	データセットの対象期間開始日及び終了日を、記述方法は「基本データの共通記述方法」の「日時」を参照。
	データセットの有効期間	データセットの有効期間開始日及び終了日を、記述方法は「基本データの共通記述方法」の「日時」を参照。
データ量	容量	データセットの容量を記載する。
	メタデータ(データ項目)数	提供可能なメタデータ(データ項目)の数を記載する。代表的な項目を例示してもよい。
	レコード数	提供可能なレコード数を記載する。
品質指標	保証データ品質評価	このデータ利用条件が品質管理ガイドブック等に沿った品質管理を行っている場合は結果を記載の対象としている保証のインスタンスへのリンク情報を値として記載する。
	サービス品質評価	データ品質管理ガイドブック等に沿った品質管理を行っている場合は結果を記載。
	データ管理プロセス評価	データ品質管理ガイドブック等に沿った品質管理を行っている場合は結果を記載。

7-7-5. データセットの利用条件

データセットの利用に必要な契約ポリシー、利用期間や利用範囲、有償/無償の区分や支払条件等の利用条件を示すためのデータモデルである。

表 7-7-5-1 データセットの利用条件(必須)

小分類	データ項目	説明
契約ポリシー	契約形態	契約形態について自由形式のテキストで記載する。列挙型コードは譲渡/利用許諾/共同利用が考えられる。
	秘密保持義務	購入者に同意を求めるか、求めないか、についてテキストで記載する。
	利用用途	許諾する利用用途について自由形式のテキストで記載する。列挙型コードは商用利用/研究利用/教育利用/制限なし/個別調整などが考えられる。
データ利用範囲	地域	データが利用される地域を特定する場合、地域を記載。記述方法は「地理空間に関するコード」を参照。
	法人/個人種別	データ利用する法人/個人の種別を特定する場合、利用できる法人/個人の種別を記載。
	用途	学術研究、公衆衛生、災害時対応などとデータ利用する用途を特定する場合、利用できる用途を記載。
取り扱い条件	データ保護要件	このデータ利用条件が記載の対象としているデータ保護要件を記載する。
	ライセンス	ライセンスの種別(クリエイティブ・コモンズ<https://creativecommons.org/licenses/>、標準ライセンスなど)をテキストで記載する。
	利用期間	データの利用期間を限定する場合、利用開始日と利用終了日を記載す

小分類	データ項目	説明
		る。記述方法は「基本データの共通記述方法」の「日時」を参照。
	法人種別	特定の法人種別だけを対象にデータ提供が可能な場合、取り扱うことが可能な法人の種別を記載する。
	用途	特定の用途だけを対象にデータ提供が可能な場合、取り扱うことが可能な用途を記載する。
価格及び支払	支払方法	データセット提供先に求める支払方法を自由形式のテキストで記載する。列挙型コードは固定料金、従量課金、売上配分、無償、その他から該当するものを選択することが考えられる。
	課金単位	有料の場合、課金の単位を記載する。レコード数あたり、利用者数あたり、API コールあたり、利用期間あたり、その他から該当するものを選択することが考えられる。
	価格帯	データセットの価格帯を記載する。下限価格、上限価格、価格、通貨単位などを記載することが考えられる。
	支払頻度	データ購入者の支払頻度を記載する。記述方法は「基本データの共通記述方法」の「日時」の「定期スケジュール」区分(年次、月次、週次)に加え、日次、利用ごとに課金、購入時に一括、その他などが考えられる。

7-8. スーパーシティにおけるデータモデル検討手順（データモデルの適用）

本調査業務では、ユースケースシナリオに基づき導かれる、サービスにおいて必要と考えられるデータモデルを検討し、7-6 推奨データモデルとの対応を確認した。ユースケースは「2-5-3(4)ユースケースの検討」にて作成されるものをベースとし、スーパーシティ/スマートシティ上で展開され得るサービスのユースケースを想定し、「戦略的イノベーション創造プログラム（SIP）第 2 期/ビッグデータ・AI を活用したサイバー空間基盤技術/パーソナルデータアーキテクチャ構築」事業の成果であるユースケースシナリオテンプレートを用いて整理した。

スーパーシティで実現されるサービスは、何らかのデータが利用可能であることに起因して創出されるものではなく、「住民が参画し、住民目線で、2030 年頃に実現される未来社会を先行実現することを目指[112]」し、サービス設計 12 箇条[113]等を踏まえユーザ中心に設計され創出されると期待する。各エリアの課題解決を起点としたサービスを実現するための、実際のデータモデル設計においては、本節で示す検討手順を参照されたい。

7-8-1. 想定されるスーパーシティサービスのユースケースシナリオ

本 7-7 データカタログのデータモデル定義においては､本読者である自治体の職員及び委託先等にて、スーパーシティサービスを検討する際に、前述された推奨データモデルを含むデータモデルをどのように整理されるべきか、その手順を検証し、結果を記した。なお、検証に使われるスーパーシティサービス例は内閣府地方創生推進事務局『「スーパーシティ」構想について』[114]にて記載されている､『参考２.「スーパーシティ」構想イメージ』の事例等をモデルとしてサービス仮説立てしたユースケースをベースとした。

7-8-2. スーパーシティサービスにおけるデータセット及びデータモデルを整理するまでの手順

各自治体の職員及び委託先等にて、スーパーシティサービス検討において様々なサービス構想が検討されるものと想定する。各自治体等にて検討されたサービスを環境実装していくためには、データモデル定義に必要なサービスシナリオだけでなく、地域としての取り組み課題定義や関連事業者のケイパビリティ・保有アセットの確認等、前段で検討しなければならないプロセスが存在する。ここでは、本調査事業にて検証するにあたって経たプロセスとそのプロセスの位置付けを以下に記すものとする。

なお、各プロセスにおけるドキュメント作成においては、内閣府より公表されている『戦略的イノベーション創造プログラム（SIP）第 2 期/ビッグデータ・AI を活用した サイバー空間基盤技術のアーキテクチャ構築ならびに実証研究事業』における『２.パーソナルデータ分野アーキテクチャ構築 「DFFT（Data Free Flow With Trust）実現のためのアーキテクチャ設計と国際標準化推進の研究開発」（一般社団法人データ流通推進協議会)』の『活用編 アーキテクチャ設計書の利用』[115]にて定義されているドキュメントを活用した。各ユースケースに作成されたドキュメントは【別紙ウ-1】を参照すること。

- エリア概要

  位置付け：スーパーシティを検討する自治体の概要の定義。

  記載観点：各自治体の職員及び委託先だけでなく、第三者目線で地域概要、特性を整理。

- 現状と課題

---

[112] 2021 年 3 月「「スーパーシティ」構想について」https://www.kantei.go.jp/jp/singi/tiiki/kokusentoc/supercity/supercity.pdf

[113] 2020 年 12 月 25 日「デジタル・ガバメント実行計画」http://www.kantei.go.jp/jp/singi/it2/dgov/201225/siryou4.pdf

[114] 引用：https://www.kantei.go.jp/jp/singi/tiiki/kokusentoc/supercity/supercity.pdf

[115] 引用：https://www8.cao.go.jp/cstp/stmain/b-1-10_200331-2.pdf

位置付け：スーパーシティサービスを提供して解決すべき課題を定義。
記載観点：「エリア概要」の地域現状から導出される課題を整理。

- 解決策

位置付け：スーパーシティサービスが取り組む分野課題を定義。
記載観点：データ連携基盤として連携するサービス及び取り扱いデータ対象領域を整理。

- サービスシナリオ

位置付け：サービスシナリオ一連の概要、ユースケースの全体像を定義。
記載観点：解決する対象課題、及び登場アクターとその事業関係性を整理。

- ステークホルダリスト

位置付け：サービスシナリオに登場するステークホルダを一覧で定義。
記載観点：サービスシナリオを input に各事業者役割及び用語定義上の分類を整理。

- サービスアセット

位置付け：サービスシナリオで各事業者が提供するサービスアセットを一覧で定義。
記載観点：サービスシナリオを input にサービス管理事業者、種別、提供機能を整理。

- ビジネス関係

位置付け：サービスシナリオを成立させる事業者間契約関係を定義。
記載観点：サービスシナリオ、ステークホルダリスト、サービスアセットを input に各事業者間で発生する契約関係及びそのサービス対価を整理。

- データリソースマップ

位置付け：サービスシナリオを成立させるデータ流通の関係を定義。
記載観点：サービスシナリオ、ステークホルダリスト、サービスアセット、ビジネス関係を input に各事業者間でやり取りされるデータ及びそのデータ種別を整理。

- トラスト関係

位置付け：サービスシナリオを成立させる信頼関係（互いの組織認知）を定義。
記載観点：サービスシナリオ、ステークホルダリスト、サービスアセット、ビジネス関係、データリソースマップを input に各事業者間でやり取りされる認証関係を整理。

7-8-3. スーパーシティサービスにおけるデータセット定義及び推奨データモデルの適用

ここでは、7-7-2 の手順にて整理されたスーパーシティサービスシナリオを基に、必要となるデータセット定義と推奨データモデルがどのように適用されるか、関係性を書き記す。

(1)データセット定義

データセット定義を対象とするスーパーシティサービスシナリオは『3.1.3 高齢者通院サポートサービス』『3.1.4 高齢者避難支援サポートサービス』『3.2.3 ものづくりツーリズムサービス』『4.1.2 シニア向け旅行サポートサービス』とする。データセットを定義する各項目は内閣官房情報通信技術（IT）総合戦略室にて公表されている『推奨データセット　データ項目定義書』の『データ項目』[116]をベースとして整理した。以下、データセット定義の各項目の説明とする。

(a)データ項目

- 管理主体：データ管理事業者
- フェーズ：サービス申込、サービス提供、次の連続したサービスなどの利用されるフェーズ
- 管理情報：フェーズの中で利用されるデータ種別（大項目）

[116] 引用：https://cio.go.jp/policy-opendata

- 項目名：サービスで利用されるデータセット（詳細項目）
- 区分：I:SC 産業データ、G:SC 行政データ、P:SC パーソナルデータ
- データ連携先：データ管理主体からデータ提供され提供先組織
- データソース：データ原本管理されている参照先番号 ［管理主体№. 管理情報№. 項目№.］
- 説明：データ項目名に対する意味定義
- 形式：データ格納形式
- 記入例：データサンプル例
- 推奨データモデルとの関係性：表 7-3-4-1　推奨データモデル検討対象（案）における(1)～(13)までの対象項目との関係性

(b)参考情報

以下の項目は内閣官房情報通信技術（IT）総合戦略室にて公表されている『推奨データセット　データ項目定義書』の定義を採用する。

- 先進自治体公開有無
- 共通語彙基盤
- 共通語彙基盤での値型

(2)推奨データモデルの適用

前述のとおり、スーパーシティサービスシナリオにおけるデータセットを定義し、推奨データモデルがどのように適用されるか、検討した。結果として、各スーパーシティサービスシナリオにて共通的に必要なデータモデル項目であることが下記例のように確認できた。特に、今回推奨データモデルとして定義した 7-6-3 施設、7-6-6 道路においては、Society5.0 の世界観をサービス実現する上で必要なデータモデル項目であるということを確認した。詳細については【別紙ウ-2】を参照する。今後、スーパーシティ/スマートシティが展開されていくと同時に、共通的に使われてくるデータモデル項目については、推奨データモデルとしての反映及び公開の検討がされていくことが望まれる。

図 7-8-3-1　推奨データモデル項目の適応項目例

データ項目													
管理主体No.	管理主体	フェーズ	管理情報No.	管理情報	項目No.	項目名	区分	データ連携先	データソース	説明	形式	記入例	推奨データモデルとの関係性
1	A市スーパーシティ拠点コンソーシアム	サービス申込フェーズ	1	住民	1	氏名	P	Z認証	–	住民の氏名	文字列	○○太郎	(13)オブジェクトのデータモデル参照
					2	性別	P	同上	–	住民の性別	文字列（1: 男/2: 女）	1	
					3	生年月日	P	同上	–	住民の生年月日	日付（YYYY-MM-DD）	2017-02-10	
					4	保険証番号	P	同上	–	住民の保険証番号	文字列（半角数字）	00000222	
			2	事業者	1	事業者コード	G	×	–	事業者コード	文字列（半角数字）	0000022200	
					2	名称	G	×	–	事業者の名称	文字列	○○会社	(3)施設のデータモデル参照
					3	住所	G	×	–	事業者の所在地	文字列	北海道札幌市厚別区2-○-○	
			3	医療施設	1	施設ID	G	×	–	情報の管理主体である地方公共団体内でデータが一意に決まるよう、NOを設定	文字列（半角数字）	0000022200	
					2	名称	G	×	–	医療機関の名称	文字列	○○病院	
					3	住所	G	×	–	医療機関の開設場所の住所	文字列	北海道札幌市厚別区2-○-○	
					4	緯度	G	×	–	医療機関の開設場所の位置情報	文字列（半角数字）	35.23456	(3)施設のデータモデル参照
					5	経度	G	×	–	医療機関の開設場所の位置情報	文字列（半角数字）	139.23456	
					6	座標参照系	G	×	–	医療機関の開設場所の位置情報の座標参照系	文字列（半角数字）	6668	
			4	避難所施設	1	施設ID	G	A市役所	–	情報の管理主体である地方公共団体内でデータが一意に決まるよう、NOを設定	文字列（半角数字）	0000022200	
					2	名称	G	同上	–	指定緊急避難場所の通称や建物等の名前	文字列	○○小学校	
					3	住所	G	同上	–	指定緊急避難場所の住所	文字列	北海道札幌市厚別区2-○-○	
					4	緯度	G	同上	–	指定緊急避難場所の位置情報	文字列（半角数字）	35.23456	
					5	経度	G	同上	–	指定緊急避難場所の位置情報	文字列（半角数字）	139.23456	(3)施設のデータモデル参照
					6	標高			–	指定緊急避難場所の位置情報の標高を記載	文字列（半角文字）	30.5	
					7	座標参照系	G	同上	–	指定緊急避難場所の位置情報の座標参照系	文字列（半角数字）	6668	
					5	対象となる町会・自治会	G	同上	–	指定緊急避難場所へ避難する対象の地域等	文字列	○○町;△△町	
			5	設備(AED)	1	設備ID	G	A市役所	–	情報の管理主体である地方公共団体内でデータが一意に決まるよう、NOを設定	文字列（半角数字）	0000022200	
					2	名称	G	同上	–	AEDが設置場所の建物等の名称	文字列	○○会館	
					3	住所	G	同上	–	AED設置場所の住所	文字列	北海道札幌市厚別区2-○-○	
					4	緯度	G	同上	–	AED設置場所の位置情報	文字列（半角数字）	35.23456	
					5	経度	G	同上	–	AED設置場所の位置情報	文字列（半角数字）	139.23456	(5)設備のデータモデル参照
					6	座標参照系	G	同上	–	AED設置場所の位置情報の座標参照系	文字列（半角数字）	6668	
					5	設置位置	G	同上	–	AED設置場所における詳細位置	文字列	1階事務室	

7-9. 共通的に使われる地理空間データの既存標準規格に関する考慮

7-9-1.　既存標準規格の調査

スーパーシティサービスにて共通的に使われるデータモデル項目は推奨データモデルとして定義を行う一方、データモデルにて参照される既存の標準規格に関する考慮が必要となる。ここでは共通的に利用されるデータの代表的な例として、地理空間データを対象に調査を実施し、データ形式や規格、取り扱い団体について整理を行った。【別紙ウ-3】にて参考情報として記す。

7-9-2.　既存標準規格の調査結果から見るデータセット定義の考察

地理空間情報の既存規格では、対象とする空間の大きさによって、記述されるデータ項目や詳細度が異なっていることが確認できた。建物を例にとると地図では建物の外周をデータ化するが、BIM(Building Information Model)では建物内部の階層構造や詳細な配管情報などが建物単位でデータ化される。

ここで、内閣官房情報通信技術（IT）総合戦略室にて公表されている『推奨データセット　データ項目定義書』の『データ項目』(引用：https://cio.go.jp/policy-opendata) をベースに作成した【別紙ウ-2】ドキュメントにおける、「3.1.4 高齢者避難支援サポートサービス」「設備(AED)」を確認すると、設備タイプの管理データには、設備が設置されている施設の経緯度に関する項目が盛り込まれている。

設備によって設置状況は異なるが、1 つの施設に多数設置されている設備（AED 等）は多いと考えられるためこのような場合には同じ内容の項目が繰り返し含まれることになる。システム運用面を考慮すると、設備には、設置してある施設をユニークに特定できる ID を付与し、位置情報は施設の経緯度を参照する構造が望ましいと考える。

地理情報を同一座標系で取り扱うために必要な座標参照系データについても同じことが言える。今回検討を行った【別紙ウ-2】ドキュメントでは、経緯度の項目と同列に座標参照系も項目として盛り込んでいるが、同じひとまとまりのデータでは共通するためメタデータとして整備されるのが望ましいと考えられる。

このように実社会での関係がデータ構造にもある程度反映されていれば、データ整備のハードルが下がり、利活用フェーズでの使い分けも自然に行われると考えられる。

7-10. スマートシティ構築における知見を踏まえたデータモデル検証

スマートシティの共通設計図かつ運用マニュアルであるアーキテクチャ「都市 OS」開発の知見を踏まえ、本データモデル方針の検討結果について検証を行い、実効性を確認した。

検証結果を以下記載する。

7-10-1. 個々のデータモデルに関する検証結果

7-6 推奨データモデルに記載されたデータモデルについて検証を行った。検証の結果指摘された問題点や課題については、可能な限りデータモデルに反映し定義などを精査した。

7-10-2. データモデルの方針に関する評価

7-1 から 7-9 の記載内容全体について検証を行った。結果を踏まえ、課題を以下のとおり報告する。

(1)解説の内容等

データモデル設計には本来相応の専門知識が必要である。本書ではスーパーシティを推進する自治体職員を想定読者に含めてかみ砕いた解説を試みているものの、全ての読者が全ての内容を理解しデータモデルを検討するには難解である。また、参照モデルとしたことにより抽象度が高く、設計者によって異なる解釈や実装につながり得る点を危惧する。7-8 などを参照してサービスにおける各データ項目定義の使い方に対するイメージを描いた上で検討を進めることが望ましいと考える。

他方、スーパーシティにおけるデータモデルの必要性を理解し、自治体及びアーキテクトとサービスを開発・提供する事業者やデータを提供する事業者とのコミュニケーションの土台としてこうした方針を示した意義は評価できる。自治体に馴染みのある「推奨データセット」のデータ項目を構造化し、組み合わせ可能な要素としてデータモデルを捉えることは、スーパーシティの相互運用性確保へ向けた第一歩であり、システムに依拠したデータ項目定義の乱立を避ける効果に期待する。

(2)データモデルの対象範囲

7-8 や図　7-2-2-1 に見られるように、実際のスーパーシティ、スマートシティサービスで連携が期待されるデータはより広範であり、かつより精緻な定義が求められる。各スーパーシティ、スマートシティ及び事業者等のニーズと期待効果を取り入れながら優先順位を検討し、今回対象外としたデータへの拡大を図るべきである。

(3)今後のデータモデル確立へ向けた課題

本書でも指摘されているように、本来データをシステム間で共有・連携させる際は、全国で統一されたデータモデル定義が運用されており、全てのシステムがその定義を参照してデータを送受信できる状態が理想である。機械判読性を考慮すると、データモデルが定義する値域の多くは、コード又は ID で記述されるべきである。また、メタデータが依拠する法制度等の改正がデジタルにデータモデルへ反映され、個々のシステムが仕様を変更しなくてもスーパーシティのサービスがアップデートされる仕組みが望まれる。各スーパーシティ、スマートシティの独自性に関わらず不可欠な基盤としてこうした「日本公式」の環境の確立を優先事項と捉え、その工程を一刻も早く示していくことが期待される。

また、データ連携において実際に必要なデータやデータ連携方法は、実現すべきサービスやスーパーシティごとのビジョン、また技術環境等によって異なり、さらにそれらの成熟に伴って変化を続けていく。この変化を大前提として、政府、自治体、アーキテクト、事業者等が協調しながらデータモデルの改善を重ねていけるよう、効果や課題の共有と議論の場を持続的に運用することが求められる。

8. 調査結果（エ）先端的サービスとデータ連携基盤の接続検証

8-1. 概要・目的

本章では、「2-6-2 先端的サービスを支える先進技術とデータ連携基盤の接続検証」に記載した考え方と接続検証のポイントに従い実施した調査及び検証結果を記載し、各エリアに推奨する先進技術を示す。基本的な考え方として、各エリアにおける先進技術の活用方法・実装方法は各エリアにて検討する方針とし、本調査業務の対象外とする。

8-1-1. 先進技術の選定プロセス

本調査業務における先進技術の選定プロセスを以下に示す。

Step1 で先端的サービス候補の抽出を行う。

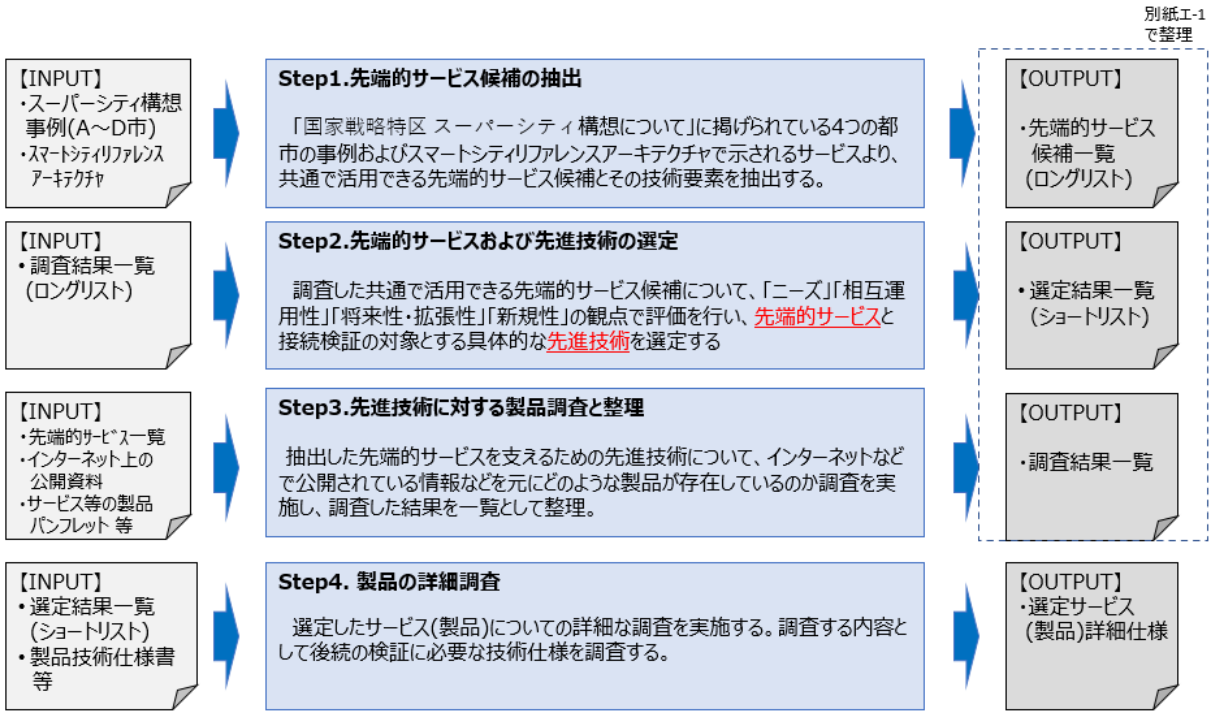


図 8-1-1-1 先進技術の選定プロセス

「国家戦略特区 スーパーシティ構想について」に掲げられた都市の構想例(A 市～D 市）及びスマートシティリファレンスアーキテクチャより、6 つの共通サービスに分類。これらの分類に当てはまる先端的サービスを抽出。先端的サービス候補を「表 8-1-1-1」に示す。

なお、「パーソナルデータ」は、個人情報保護法に規定する「個人情報」に限らず、かつ個人識別性の有無に関わらず、位置情報や購買履歴など広く個人に関する情報を構成し得るデータを指す。[117]

また、本章では行政及び民間企業が保有するパーソナルデータを対象とした。

[117] パーソナルデータの定義：「戦略的イノベーション創造プログラム（SIP）第 2 期／ビッグデータ・AI を活用したサイバー空間基盤技術／パーソナルデータアーキテクチャ構築」リファレンスアーキテクチャ概要書
https://data-trading.org/wp-content/uploads/2020/06/01_PersonalDataReferenceArchitecture_Overview_FirstEd.pdf

表 8-1-1-1　先端的サービス候補

No.	先端的サービス候補	概要
1	パーソナルデータに関するサービス	• 各エリアや様々な分野で多様なパーソナルデータを連携・活用するためのパーソナルデータを管理するサービス。
		• 市民への情報提供、健康管理、各エリアで住民の状況・特性に応じた行政施策や新規事業の立案などに 活用可能。
2	行政と民間の情報連携に関するサービス	• 官民・官官・民民それぞれで幅広く生活全般をカバーするための、行政と民間の情報連携サービス。
		• 安全な情報公開・共有を実現するため、証跡管理やブロックチェーン（分散台帳技術）による改ざん検知等を備え、広域での防災時の情報共有や、ライフログや医療データ連携による住民の健康管理などに活用可能。
3	生体認証による本人認証サービス	• 住民サービスの利便性の向上ならびに安全上の脅威を排除するための生体認証による本人認証サービス。
		• ボランティア・タクシー配車や観光における MaaS などに活用可能。
4	データ流通サービス	• エリア間や分野横断でのデータ利活用を促進するための、データ連携・相互流通のためのサービス。
		• 行政・民間などのハザード、道路、衛星、気象データを組み合わせ防災対策高度化などに活用可能。
5	情報提供ポータルサービス	• 住民の属性や嗜好に合わせて、最適な情報を提供するためのサービス。
		• 個人の属性や嗜好に合わせて情報提供の最適化サービスなどに活用可能。
6	可視化･分析ダッシュボードサービス	• 地域課題の解決を目的とした都市の状況を可視化や分析するためのダッシュボードサービス。
		• 他の都市のデータ分析、自分たちの都市の特徴理解などに活用可能。

スーパーシティにおける各サービスの位置付けを図 8-1-1-2 に示す。なお、図中の番号は表 8-1-1-1 に対応している。

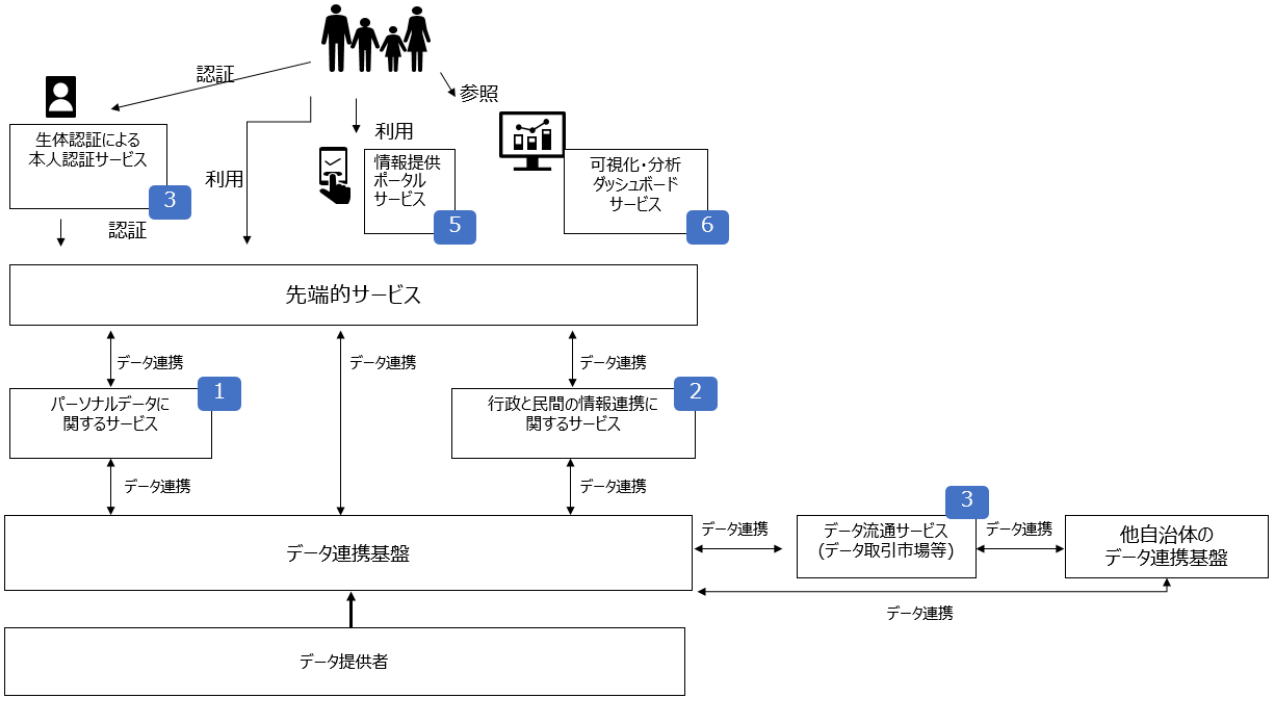


図 8-1-1-2　スーパーシティにおける各サービスの位置付け

8-1-2. 先進技術の選定

調査した先端的サービスについて、以下の表 8-1-2-1 に示す評価項目に基づき評価を行い、接続検証の対象とする先進技術を選定した。

表 8-1-2-1 先端的サービスの評価項目

No.	評価観点	評価内容
1	先端的サービスを支える先進技術としてのニーズが高いか。（ニーズ）	先端的サービスを支える先進技術としてニーズが高いか、また個別サービスに特化した内容ではなく複数のサービス間で共通に利用され得る技術であるかを評価する。
2	各種サービス分野との相互運用性や相乗効果が期待できるか。（相互運用性）	先端的サービス又は他の先進技術との親和性が高い技術であるか、また、スーパーシティ構想にける先端的サービスは複数のサービス分野（行政、金融、物流等）に跨って活用することができるか、各種サービス分野に相乗効果をもたらすことが期待できるかを評価する。
3	社会動向を考慮した将来性があるか。（将来性・拡張性）	新型コロナウイルス感染症の流行を受けたニューノーマルへの対応等、地域が直面し得る社会課題の解決を継続的に対応する先端的サービスに不可欠な先進技術であるかを評価する。
4	他であまり取り組みのない新規性がある技術であるか。（新規性）	世の中で幅広く用いられている技術について、本事業として改めて接続検証する意義は低いため、本調査業務における新規性を評価する。

8-1-3.　検証対象となる「先端的サービスを支える先進技術」

先端的サービスを表 8-1-2-1 で示す観点で比較・評価した。表 8-1-1-1 で示す 6 つのサービスは、A～D 市ユースケースのいずれかで用いられているサービスであり、ニーズが高いと認識。

残り「相互運用性」「将来性・拡張性」「新規性」の観点で評価し、赤枠で示した 4 つの先端的サービスを調査対象として表 8-1-3-1 のとおり選定した。

表 8-1-3-1　先端的サービスの絞り込み

先端的サービス	評価観点			評価結果
	相互運用性	将来性・拡張性	新規性	
パーソナルデータに関するサービス	パーソナルデータは複数の分野で活用でき、既存のサービスなどに対する相乗効果は高いと判断 ◎	様々な分野でパーソナルデータを活用したサービスが増えつつあり、将来性・拡張性が高いと判断 ◎	パーソナルデータを管理するようなサービスは、立ち上がりつつある状況であり、新規性が高いと判断。 ◎	9
行政と民間の情報連携に関するサービス	スーパーシティ内外のデータの相互連携をするためのサービスであり、複数分野での活用・相乗効果も高いと判断。 ◎	「官民データ活用推進基本法」制定以降、官のデータを活用した事例が増えてきており、将来性・拡張性が高い。また、ワンストップでできるサービスはこれからも増えていくと考え、将来性は高いと判断 ◎	特定の分野内のデータ連携は多数の取り組みが存在しているが、行政・民間での分野を跨ったデータ連携の取り組みは少なく、新規性が高いと判断。 ◎	9
生体認証による本人認証サービス	安全にサービスを活用するためには、生体認証による本人認証が必要であり、複数の分野で活用・相乗効果も高いと判断。 ◎	様々な生体認証技術が開発・製品化され、既に生体認証は身近な存在であるが、これからも普及の余地あり ○	生体認証は一般的なものになりつつあるが、スーパーシティとして、生体認証を活用した取り組みとしては新規性が高いと判断。 ○	7
データ流通サービス	データの活発的な活用に向けた、データを安全・安心に流通させるサービスであり、複数分野での活用・相乗効果も高いと判断。 ◎	諸外国や日本におけるデータ流通に対する取り組みが活発になってきており、将来性・拡張性が高いと判断 ◎	安全性・効率性・利便性を高めたデータ連携・相互流通に対する取り組みは少なく、新規性が高いと判断。 ◎	9
情報提供ポータルサービス	様々な分野のデータを集めて、個人に属性などに合わせた、データを提供するサービスは、複数分野での活用・相乗効果も高いと判断。 ◎	マーケティング業界では個人属性に合わせた広告など実現しており、スーパーシティにおいても将来性は高いと判断 ◎	様々な情報提供ポータルは存在しており、本事業としてあらためて検証するような新規性はないと判断。 ×	6
可視化・分析ダッシュボードサービス	様々の分野のデータに基づいた、スーパーシティの全体を俯瞰するような可視化・分析ダッシュボードは、複数分野での活用・相乗効果も高いと判断。 ◎	ダッシュボードそのものは、実現するサービスや製品も多々あり、サービスそのものの将来性・拡張性は未知数 ○	様々な可視化・分析ダッシュボードが存在しており、本事業としてあらためて検証するような新規性はないと判断 ×	5

＜評価結果凡例＞

◎合致する：3 点

○ある程度合致する：2 点

×合致しない：0 点

選定した先端的サービスの概要とそれらを支える先進技術を表 8-1-3-2 に示す。

表 8-1-3-2　検証対象の「先端的サービスを支える先進技術」

No.	先端的サービス分類	先端的サービス分類の概要	検証対象の先端的サービスを支える先進技術
1	パーソナルデータに関するサービス	• 各エリアや様々な分野で多様な個人情報を連携・活用するためのパーソナルデータを管理するサービス。 • 市民の健康管理、各エリアで住民の状況・特性に応じた行政施策や新規事業の立案などに活用可能	**パーソナルデータを保護しつつ安全に利活用するための先進技術** • 同意/個人情報の管理（匿名化技術） • データの安全な加工・分析（秘密計算技術）
2	行政と民間の情報連携に関するサービス	• 官民・官官・民民それぞれで幅広く生活全般をカバーするための、行政と民間の情報連携サービス。 • 安全な情報公開・共有を実現するため、証跡管理やブロックチェーン（分散台帳技術）による改ざん検知等を備え、広域での防災時の情報共有やライフログや医療データ連携による住民の健康管理などに活用可能。	**行政と民間事業者の情報連携に資する技術** • ブロックチェーンによる証跡管理や改ざん検知技術
3	生体認証による本人認証サービス	• 住民サービスの利便性の向上ならびに安全上の脅威を排除するための生体認証による本人認証サービス。 • ボランティア・タクシー配車や観光における MaaS などに活用可能。	**生体認証等を活用した本人認証技術** • 生体認証（顔認証）による本人認証技術
4	データ流通サービス	• エリア間や分野横断でのデータ利活用を促進するための、データ連携・相互流通のためのサービス。 • 行政・民間などのハザード、道路、衛星、気象データを組み合わせ防災対策高度化などに活用可能。	**データ取引市場を活用したデータ流通を促進する技術** • 利用者の円滑なデータ検索を支援する統合検索技術

ここまでで調査した先進技術リストを別紙エ-1 に示す。

8-1-4. 調査結果

(1)選定した先進技術に関する調査結果

(a)パーソナルデータを保護しつつ安全に利活用するための先進技術

(ｱ)必要性

スーパーシティ構想における先端的サービスとしては、各地域の住民特性に適合したサービスや個々の住民にカスタマイズされたサービスが求められることから、これまで以上に各エリアで多様なパーソナルデータが連携・活用されると考える。それらのサービスを支える技術として、機微な情報の漏洩等のリスクを低減しつつ活用範囲を最大化し、確実な本人同意に基づいたパーソナルデータの管理・提供ができるよう、各エリアとして以下のような技術が共通的に必要になると考える。

1)住民の同意に基づくパーソナルデータ管理及び住民の意思を反映した施策立案支援

住民自身が用途を把握し、パーソナルデータの提供の同意を自身で管理することを可能にする技術や、提供の停止を依頼したい場合も容易に手続き可能にするための管理技術が必要となる。これにより、住民の意思に反してパーソナルデータが提供されることを防ぐことができると考える。また、当該技術を活用し、定量的な住民同意に基づいて自治体や研究所等公益団体に住民のパーソナルデータを提供し、かつ公益団体でデータを分析することで、各エリアで住民の状況・特性に応じた行政施策や新規事業の立案が期待できる。

2)パーソナルデータの匿名化・秘匿化

各エリア単位でパーソナルデータを保管・提供する運用を想定した際、情報漏洩等が発生した場合、地域の信頼の失墜にまで発展する。そこで、情報漏洩が発生しないようなセキュリティ対策を実施した上で、万が一情報漏洩した場合にも個人が特定されないよう、保管されるパーソナルデータの秘匿化・匿名化技術が重要となる。また、それらが秘匿化された状態でも利活用を可能とする技術や、改ざんや不正な利用を検知するための証跡管理技術も重要と考える。

(ｲ)先進技術の概要

パーソナルデータを保護しつつ安全に利活用する仕組みとして「PDS」と「情報銀行」がある。以下に説明を記載する。

1)PDS について

PDS（Personal Data Store）とは、他者保有データの集約を含め、個人が自らの意思で自らのデータを蓄積・管理するための仕組み（システム）であって、第三者への提供に係る制御機能（移管を含む）を有するもの。運用形態としては、個人が自ら保有する端末等でデータを蓄積・管理する（事業者は本人の同意によりデータを活用できる）分散型と、事業者が提供するサーバ等でデータを蓄積・管理する（個人は当該事業者にデータの蓄積・管理を委託する）集中型がある。（実際にデータをやり取りする形態と、データをやり取りせず必要な時にアクセス権（閲覧のみ可、コピー不可など）を提供・管理する形態もある。）[118]

[118] https://www.kantei.go.jp/jp/singi/it2/senmon_bunka/data_ryutsuseibi/dai2/siryou2.pdf

2)情報銀行について

情報銀行（情報利用信用銀行）とは、個人とのデータ活用に関する契約等に基づき、PDS 等のシステムを活用して個人のデータを管理するとともに、個人の指示又はあらかじめ指定した条件に基づき個人に代わり妥当性を判断の上、データを第三者（他の事業者）に提供する事業。（データの提供・活用に関する便益は、データ受領事業者から直接的又は間接的に本人に還元される。）[119]

(ｳ)製品例について

PDS の製品例を「表 8-1-4-1」に示す。

表 8-1-4-1　PDS の製品例

No.	サービス（製品）名	提供元	概要
1	匿名バンク	株式会社日立製作所	• サーバ型 PDS • パーソナルデータの同意管理機能を提供する • 個人情報を匿名化して管理する。
2	Personium[120]	OSS（オープンソースソフトウェア）	• オープンソース分散型 PDS • PDS を実現するための PF
3	パーソナルデータ管理サービス「paspit」	DataSign	• PDS 内蔵の情報銀行サービス。個人のパーソナルデータを保管し、様な場所にある個人のデータを個人の指示で集約。

[119] https://www.kantei.go.jp/jp/singi/it2/senmon_bunka/data_ryutsuseibi/dai2/siryou2.pdf
[120] https://personium.io/ja/

(b)行政と民間事業者の情報連携に資する技術（ブロックチェーン）

(ｱ)必要性

先端的サービスの実現のためには、官民・官官・民民それぞれで幅広く生活全般をカバーする情報連携を行う必要がある。データ連携基盤においてはオープン API 等の活用により、基礎機能としてオープンな形での情報公開・取得が可能となる。一方で、ビジネスの内容等の機微な情報等については、一部関連事業者や自治体等ある程度クローズな範囲で安全に情報の公開・共有を実現するため、証跡の管理やブロックチェーン（分散台帳技術）による改ざん検知等が必要であると想定する。これらの活用により官民のデータ連携による手続きのワンストップ化や、行政手続きの IT 化（コネクテッド・ワンストップ等)、及び官民の独自保有情報を活用し、かつ事業者単独では実現が困難であった新たなサービスの創出が期待できる。また、民間事業者と自治体の情報連携をスムーズに行うために、自治体の業務システム環境と通信を安全に連携させる仕組みが必要となる。

(ｲ)先進技術の概要

行政と民間事業者の情報連携に資する技術としてブロックチェーンによる証跡管理や改ざん検知技術について説明する。ブロックチェーンとは中央管理者不在の分散型ネットワーク(P2P)の参加者同士が分散台帳を共有し、相互監視、合意形成・承認することで、信頼性を担保し、データの改ざんを防ぐ技術のこと。分散台帳には、一定期間の取引データをブロック単位にまとめて記録する（時系列にチェーンへ連結・蓄積)。

ブロックチェーンは 3 つの要素技術「暗号技術」「合意形成アルゴリズム」「P2P 技術」の組み合わせで機能実現している。ブロックチェーンの特性は「真正性」「透明性」「信頼性」である。

以上の要素技術・特性により、以下の用途を実現している。

1)用途

- データ連携・共有（医療カルテ）
- トレーサビリティ（履歴・証跡）
- 通貨（電子決済）
- 投資（証券・債券）
- ユーティリティサービス（チケット・電子）
- 契約・取引（CO2 削減）
- 認証・証明(KYC)
- 権利（著作権・所有権）

(ｳ)製品例

ブロックチェーン関連製品の例を「表 8-1-4-2」に示す。

表 8-1-4-2　ブロックチェーン関連製品例

No.	製品	提供元	概要
1	Hyperledger Fabric	OSS（オープンソースソフトウェア）	• 企業向けブロックチェーンフレームワーク • OSS をベースとした BaaS（ブロックチェーン環境サービス）が提供されている。
2	Corda	R3	Corda は企業間取引での利用に特化したブロックチェーンプラットフォーム。世界で 300 社を超える金融機関、規制当局、中央銀行、業界団体、システム・インテグレーターやソフトウェアベンダにより構成される R3 エコシステムにより、エンドユーザ目線で設計・開発されている。
3	企業間情報連携基盤(NEXCHAIN)	企業間情報連携推進コンソーシアム	データを活用したサービスの更なる高度化を目的に、企業がそれぞれ保持する独自情報を、お客さま本人の同意のもとで持ち寄り共有することで、企業同士が垣根を越えた情報連携を実現する枠組み（コミュニティ・技術支援）。

No.3 の企業間情報連携基盤(NEXCHAIN)は、サービス型で機能を提供しており、実機検証の準備が短期間で可能なため、評価対象とした。

(c)生体認証等を活用した本人認証技術

(ｱ)必要性

先端的サービスの 1 つとして、生体認証等を活用した本人認証を前提とした住民サービスの実現がある。近年、ID とパスワードを主体としたインターネット空間上のサービス活用ならびにリアル社会における ID カード等による認証に基づくサービスが普及している。それに伴い、なりすまし等によるサービスの安全・安心性を脅かす脅威や、パスワードや大量のカード管理等、利用者の利便性を著しく低下させること等が、もはや発生リスクやその脅威といった懸念レベルではなく、現実の犯罪や問題として可及的速やかに解決すべき社会課題化している。社会課題である住民サービスの利便性の向上ならびに安全上の脅威の排除に向け、顔認証をはじめとする生体認証による個人認証の活用は注目すべき技術の 1 つだと考える。

(ｲ)技術の概要

生体認証等を活用した本人認証技術については複数の技術要素があり、代表的な要素について下記に記載する。

1)顔認証　　：顔の輪郭、目や鼻の形状や配置の特徴を分析して人物を識別する技術
2)虹彩認証　：目の虹彩の特徴を分析して人物を識別する技術
3)指紋認証　：指紋の特徴を分析して人物を識別する技術
4)静脈認証　：手や指の内側にある静脈（血管）の特徴を分析して人物を識別する技術
5)耳音響認証：外耳道の形状によって異なる反響音の特徴を分析して人物を識別する技術
6)音声認証　：マイクを用いて音声を拾い、その特徴を分析して人物を識別する技術

(ｳ)調査対象の選定

各技術要素に対して、本取り組みにおけるデータ連携基盤と接続検証の対象を選定するための評価を行う。評価は「表 8-1-2-1」の項目をベースに行う。

1)ニーズ

先端的サービスを支える先進技術としてのニーズが高いかを、スーパーシティ構想におけるユースケース事例（A 市～D 市）と各先端的サービスとの適応可否の観点で評価する。

- ユースケース事例（A 市～D 市）への適応

A 市、B 市及び D 市の事例では本人確認を行うシーンがあり、生体認証の活用が見込める。技術要素としては、顔認証、虹彩認証、指紋認証、静脈認証、音声認証が該当する。
また、C 市の事例では災害現場での本人特定に顔認証の活用が見込める。顔認証は本人が意識せずに認証することが可能な点で他の要素技術にはない特徴があり、C 市の事例に該当する。
耳音響認証は利用者ごとに専用機器が必要となり、スーパーシティのユースケースには適さない。

- 各先端的サービスとの適応

顔認証と指紋認証は導入のし易さから、行政手続き、交通、観光、社会福祉、教育現場など数多くのサービスに対して導入実績がある。虹彩認証と静脈認証は行政手続き、空港、銀行など精度が求められる場合に導入される事例があるが、専用機器のコスト面等から利用されるサービスは限定される。音声認証は通話時などリモートでの認証において導入事例があり、耳音響認証は工場などでのマスクや手袋の着用により他生体認証が活用できない特殊なケース等でのニーズがある。

以上より、幅広いサービス事業で顔認証と指紋認証のニーズが高いと評価する。

2)相互運用性

各種サービス分野との相互運用性や相乗効果が期待できるかを、他先進技術との親和性、標準規格への適応、複数サービス分野を跨った利用可否の観点で評価する。

- 他先進技術との親和性

顔認証は「(b)行政と民間事業者の情報連携に資する技術（ブロックチェーン)」で挙げているブロックチェーンと連携した導入事例がある。

- 標準規格への適応

顔認証、指紋認証、音声認証はオンライン認証の国際標準規格（FIDO）に対応した認証技術を取り入れた製品が公表されている。また、本書の（ア）データ仲介機能や（イ）API 共通ルールで推奨している OAuth2.0 や OIDC の本人認証時に顔認証や指紋認証が使用されている事例もある。

- 複数サービス分野を跨った利用可否

空港、飲食店、ホテルのサービスを顔認証で連携した導入事例がある。

以上より、相互運用性の観点では顔認証が最も対応が進んでいると評価する。

3)将来性・拡張性

社会動向を考慮した将来性があるかを、ニューノーマルへの適応、Society5.0 との関連性の観点で評価する。

- ニューノーマルへの適応

全ての要素技術は非接触かつマスク着用時での認証が可能だが、音声認証は発声が必要となるため対面での利用は向かない。また、顔認証は感染症対策をテーマとした実証実験の事例があり（認証時の体表温度測定、マスク非着用者の検知、混雑度検知、人物トラッキング等)、ニューノーマルへの適応に向けた取り組みが進んでいる。

- Society5.0 との関連性

Society5.0 で課題となる、高齢期（要介護期)、災害発生時（避難時）など所持や記憶による認証情報の提示が必ずしも期待できないケースにおいて、顔認証は利用者の動作を必要としないため他技術と比較して優位性が高いと評価する。

以上より、将来性・拡張性の観点では顔認証が他要素技術よりも高いと評価する。

1)～3)の評価結果を「表 8-1-4-3」に示す。調査した結果、総合的に顔認証が最も評価点が高いため、接続検証の対象は顔認証技術とする。

表 8-1-4-3　4 つの評価観点での技術評価結果

技術要素	ニーズ	相互運用性	将来性・拡張性
顔認証	○	○	○
虹彩認証	△	×	△
指紋認証	△	△	△
静脈認証	△	×	△
耳音響認証	×	×	△
音声認証	△	△	×

※凡例　○：満たす、△：一部満たす、×：満たさない

(ｴ)顔認証の製品事例

以下に、代表的な顔認証技術の関連製品の例を記載する。本調査業務では「Bio-IDiom Services」を使用して接続検証を行う。

1)Bio-IDiom Services　(NEC)マスク着用時でも高精度な認証を実現する新たな顔認証エンジンを搭載した製品。

2)KPAS　(Panasonic)顔認証による入退セキュリティ&オフィス可視化システム。

3)顔認証パッケージソフト（LFMnex-key)（日立国際電気）顔画像をリアルタイムで照合、人物を特定。 顔照合の結果を各設備へ配信できるため、共通の「鍵」として利用可能な製品。

(d)データ取引市場を活用したデータ流通を促進する技術

(ｱ)必要性

スーパーシティにおいて、様々なエンティティからデータが生成され、それらのデータが分散管理されている状況では、データ提供者とデータ利用者を繋ぎ、都市間や分野間での安全性・効率性・利便性を高めたデータ連携・相互流通の仕組みが必要となる。

実現方法としては、データ取引市場を活用し中立性を確保したデータ流通を実現することで、エリア間や分野横断でのデータ利活用が促進され、複数分野のデータを融合した新しい価値・サービスの創出､データを活用した技術の発展により､住民 QoL の向上に繋がることが期待できる。

データ取引市場を構成する技術の例として、なりすましの防止や情報の改ざんを防止するセキュリティ技術（電子署名）や、データの意味を高精度に推定することで関連データを高精度&広範囲で見出す AI 技術（データ意味理解）等が挙げられ、データの安全性や価値を高め利活用を広げる技術が共通的に活用可能になると考えられる。

スーパーシティのデータ連携基盤や民間企業が運用するデータカタログ、あるいはデータ販売サービス独自の API により相互接続性を具備するようになると、それぞれ特性を持ったデータカタログが連携した膨大なデータカタログ群が形成されるとともに、業界・業種、都市や地域横断のデータ利活用が期待される。

一方で、データカタログそのものが物理的に分散して管理されている状況においては、利用されるインターフェースによってデータ検索が特定範囲内に限定的となるなど、データ利用者のデータ検索に非効率を生じてはならないことから、データ利用者には効率的かつ効果的なカタログ横断（統合）検索を行えるとともに「求めるデータ」への高い到達性が求められる。

(ｲ)技術の概要

データカタログを運営する主体者によってそこで扱われるデータには特性・特徴があると考えられる。それぞれ特性を持ったデータカタログ同士が連携し検索範囲が拡張する場合、カタログごとに使用されている語彙や意味・知識等から生成されたデータが存在しており、全文検索のような単純な語彙や用語の完全、又は部分的な一致だけでは、最善の検索結果が得られない可能性がある。

データ利用者は、データ意味理解技術が具備されたデータ取引市場で、他のデータカタログと連携された広範なデータカタログ群に対し検索を実行すると、メタデータに記載された人間可読な内容の他、語彙関係の類推、データ自体の意味（データ構造）を理解した検索結果が参照可能になる。また、データ同士の類似度も合わせて指し示すことが可能となり、間接的に相関があると考えられるデータの価値向上、複数データを組み合わせたデータ利用、効率的なデータ比較も可能となる。このようにデータ意味理解技術は、データ利用者の効率的なデータ検索、データ提供者とのマッチング率の向上、データ利用機会の創出を支援するものである。データ取引市場へのデータ意味理解エンジン実装イメージを「図 8-1-4-1」に示す。

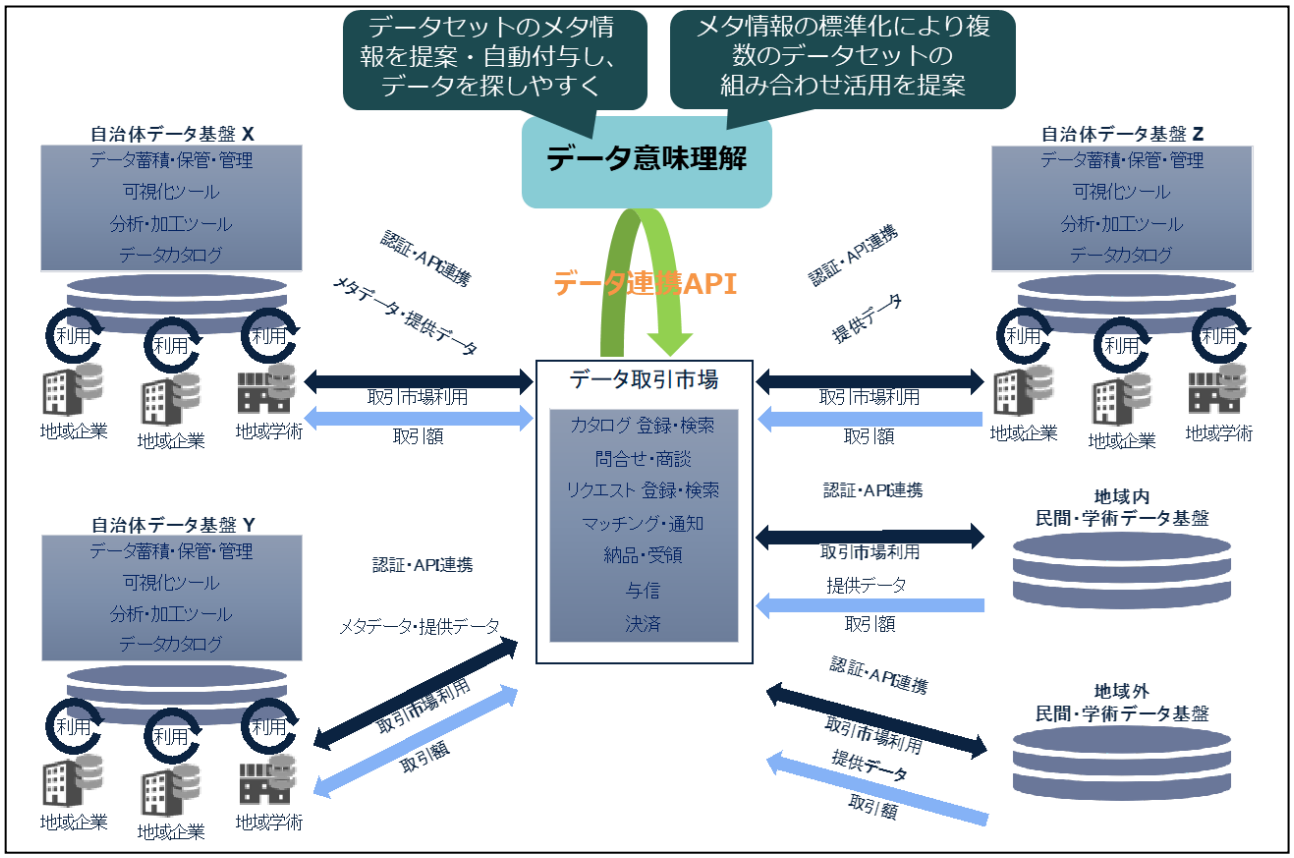


図 8-1-4-1　データ取引市場へのデータ意味理解エンジン実装イメージ

都市や地域で活動する人、文化、企業、業界、業種によって、使用する語彙、意味、知識は異なる。この非統一性を社会全体でのデータ流通活性化の観点で鑑みると、共通語彙の普及により語彙、意味、知識の同一化が進んでいくが、既に存在する人や企業、都市、地域の言語的、用語的な慣習や慣例を尊重し、データ意味理解技術のようにシステム側でその違いによる問題の解決、データ変換・統合や共通化等の利便性を提供することも、データ活用社会参加者全体への便益に資すると考える。

このような観点から、スーパーシティの各都市、地域間及び業界・業種といった分野横断でデータ利活用を促進するためには高度なデータ共有の仕組みが必要であり、そのひとつとして「データ意味理解技術」が有用であると考える。データ意味理解技術のイメージを「図 8-1-4-2」に示す。

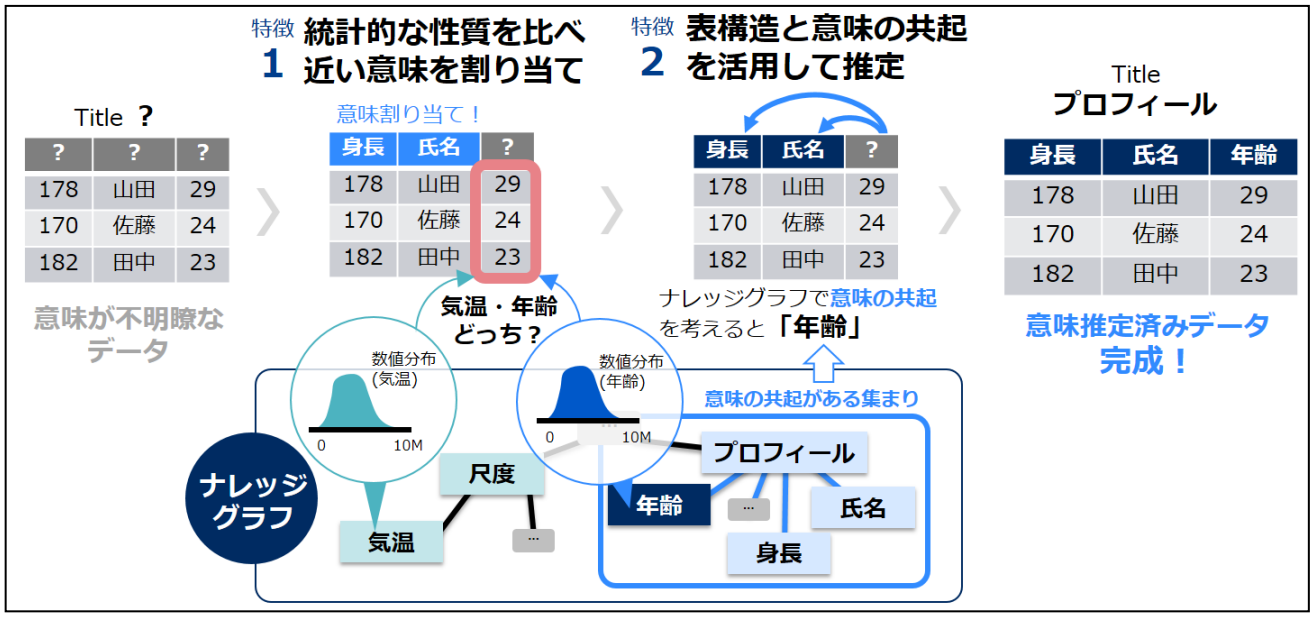


図 8-1-4-2　データ意味理解技術とは

(ｳ)データ意味理解技術の製品事例

以下に、代表的なデータ意味理解技術の関連製品の例を記載する。

- データ意味理解技術（NEC）[121]

多種多様なデータの本質的な意味を AI で推定する。

[121] https://jpn.nec.com/press/201908/20190805_02.html

8-2. 先端的サービスを支える先進技術とデータ連携基盤の接続検証

先端的サービスを支える先進技術とデータ連携基盤の接続検証について記載する。

8-2-1. 接続検証のポイント

接続検証（机上及び実機検証）における評価項目を示す（表 2-6-2-1　机上評価及び実環境を用いた接続検証による検討評価項目で示した評価項目の再掲）。

表 8-2-1-1　接続検証（机上及び実機検証）における評価項目

分類	検討評価項目	検討・評価実施タイミング		
		共通に推奨すべき技術の選定	机上評価	実環境を用いた実証評価
スーパーシティに活用される技術・ニーズ	先端的サービスを支える先進技術としてのニーズがあるか	●		
	各種サービス分野との相互運用性や相乗効果が期待できるか	●		
	社会動向を考慮した将来性・拡張性があるか	●		
相互運用性を考慮した接続・連携が可能か	データ連携基盤との相互運用性に資する標準的な仕様に則しているか		●	
	データ連携層である API の共通ルールとの親和性		●	●
	データ層であるデータモデル・データ仲介（ブローカー）機能との親和性		●	●
複数サービスを同時に利活用した場合に発生し得る問題の観点	複数サービスの間での連携時の接続容易性や技術的課題		●	
	複数サービスを同時に連携する際の動作における課題		●	
	複数サービスを同時に連携する際のデータの一貫性に関する課題		●	

8-2-2.　接続検証について

(1)パーソナルデータを保護しつつ安全に利活用するための先進技術

データ連携基盤と個人情報の同意管理基盤である匿名バンクの接続検証結果を記載する。

(a)利用ユースケースについて

「図　8-2-2-1」」において「運動や食事データ等のライフログや医療データを連携することで健康～未病～治療　のサイクルをシームレスに繋ぐヘルスケアプラットフォームを構築。」と示されている。これら先端的サービスの実現のためには、パーソナルデータを提供に対し個人の同意を取得する仕組みが必要である。それら個人情報の同意管理基盤として匿名バンクが活用できると考える。ユースケースへの適用事例を「図 8-2-2-2」に示す。

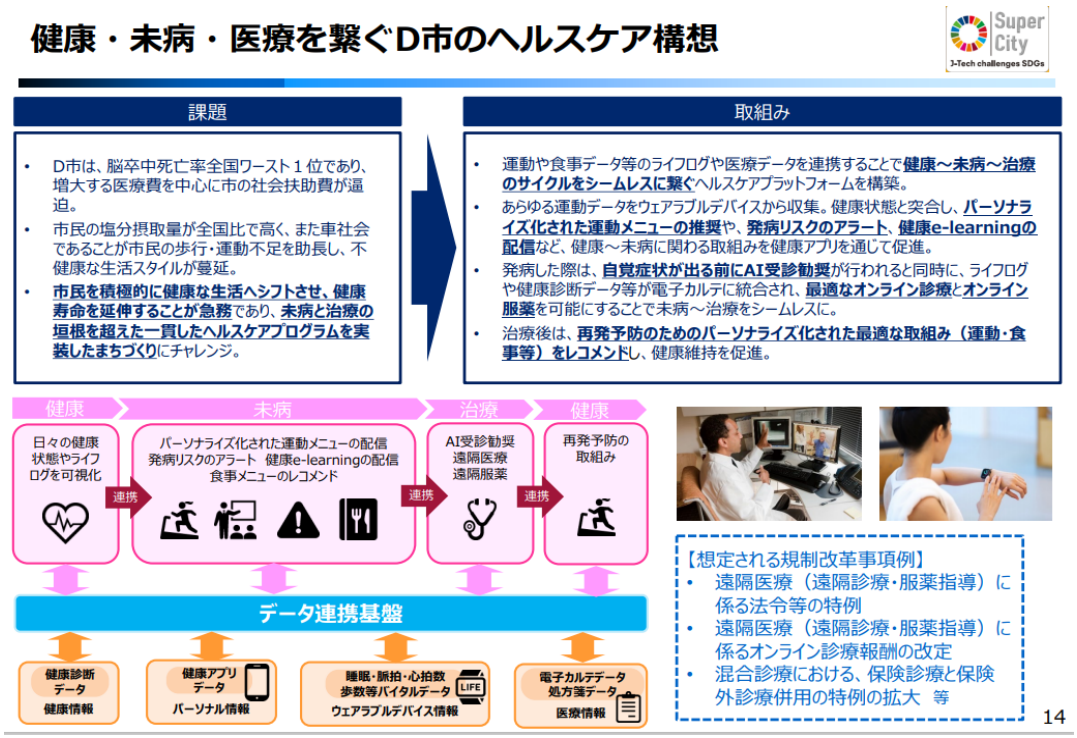


図 8-2-2-1　D 市のユースケース[122]

[122] D 市のユースケース　内閣府「スーパーシティ構想について」

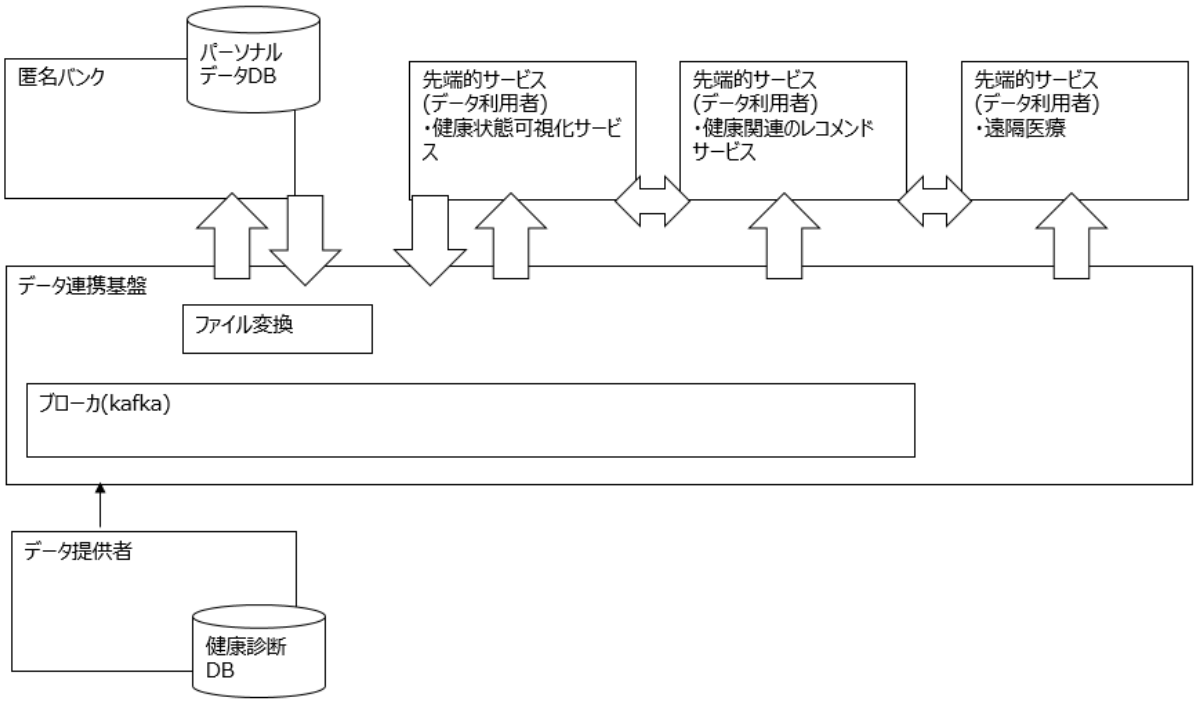


図 8-2-2-2　D 市ユースケースへの適用事例

(b)実機検証環境の説明

(ｱ)検証環境のシステム構成

匿名バンクとブローカーとの実機による接続検証のシステム構成を「図 8-2-2-3」に示す。

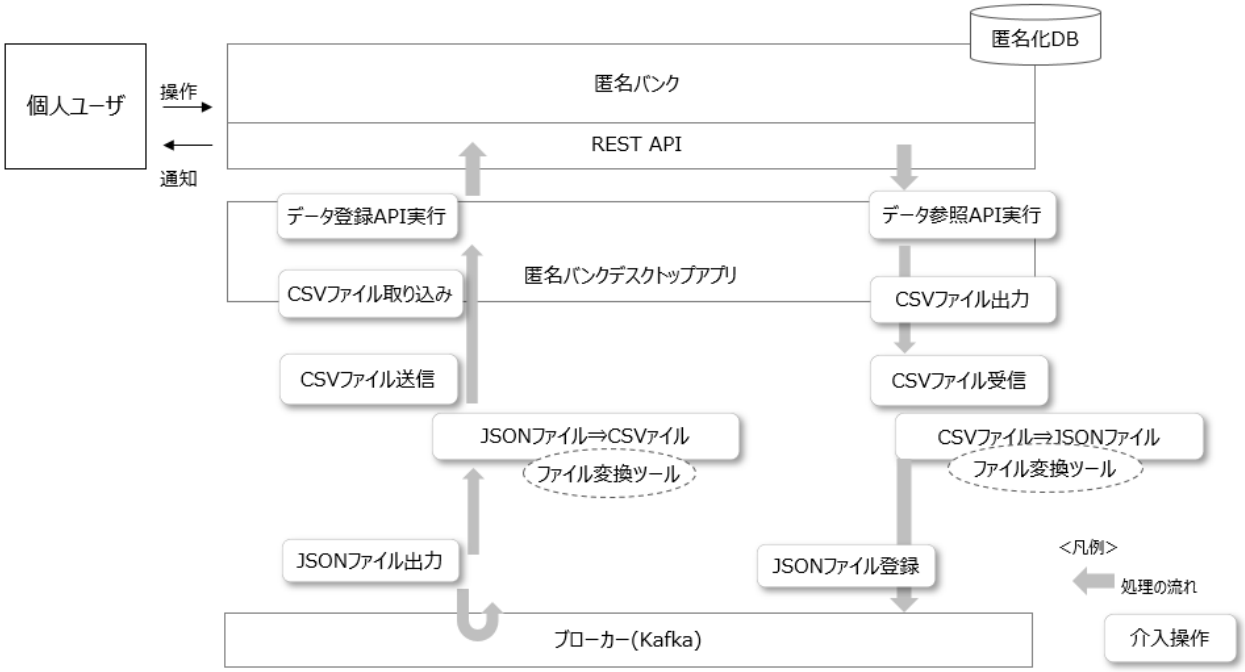


図 8-2-2-3　検証環境のシステム構成

(ｲ)各機能の概要

- ブローカー(Kafka)：データ連携を中継する。本検証では Kafka を利用する。
- 匿名バンク：パーソナルデータを管理する。
- 匿名バンクデスクトップアプリ：匿名バンクとのデータ入出力やユーザインターフェースを担う。

(c)接続検証結果

実機/机上検証を行った結果を基に以下の観点で検証結果を記載する。

- データ連携基盤との接続性
- API の共通ルールとの整合
- データモデルの方針との整合
- サービス間連携の実現性
- サービスの運用性

(ｱ)データ連携基盤との接続性

ブローカーとの接続検証（実機検証）結果について記載する。

「匿名バンク」ではデスクトップアプリ経由のデータ入出力が必要となる。検証実施時点では JSON ファイルでの入出力は未対応だったため、CSV ファイルでのデータ入出力を行った。そのため、Kafka から JSON で出力した連携データを CSV ファイルに変換し匿名バンクへ連携した。また匿名バンクから CSV ファイルで出力したデータを JSON に変換し、Kafka へ登録した。

実機検証手順を以下に示す。

- Kafka からデータを JSON 形式で取得する。
- ツールを用いて JSON ファイルを CSV ファイルに変換する。
- 匿名バンクデスクトップアプリで CSV ファイルを取り込む。
- 匿名バンク上の画面操作で個人情報の提供に関する同意を行う。
- 匿名バンクデスクトップアプリで CSV ファイルを出力する。
- ツールを用いて CSV ファイルを JSON ファイルに変換する。
- JSON ファイルを Kafka に取り込む。

(ｲ)接続検証の結果

上記手順でデータ連携基盤と匿名バンクのデータ連携を確認した。

(ｳ)API の共通ルールとの整合

検証で使用した匿名バンクでは CSV ファイルと JSON ファイルの変換が必要となったが、ファイル変換機能を整備する。もしくは REST API に対応した製品を使用することで対応可能となる。

(ｴ)データ連携（検証時のデータモデル）

検証実施のデータモデルの標準化への対応状況、課題について検証する。本検証の対象とした匿名バンクでは、製品の仕様の範囲内で登録するデータを規定できるため、標準データモデルに対応した形でデータを登録することで対応可能となる。

(ｵ)サービス間連携の実現性

サービス間のデータ連携の実現性について図 8-2-2-4 の例で説明する。

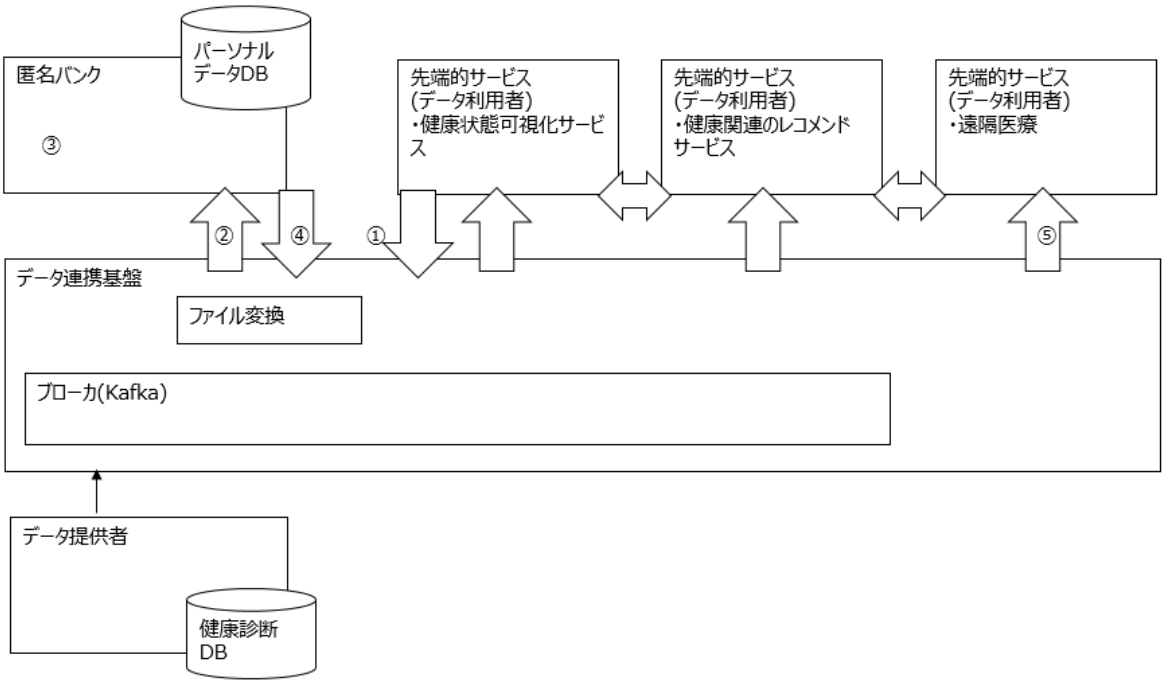


図 8-2-2-4　サービス間連携イメージ図

分野間のデータ連携は、ブローカーを仲介して行う。 加えてブロックチェーン基盤を仲介させることで履歴管理などが可能となる。連携の流れは以下のとおり。

- 先端的サービス（健康状態可視化サービス）からブローカーにデータを連携する。(図中①)
- ブローカーから匿名バンクにデータを連携する。匿名バンクに記録される。(図中②)
- 匿名バンク上で同意管理の画面操作を行う。(図中③)
- 匿名バンク上データ連携基盤へデータを連携する。(図中④)
- ブローカーから先端的サービス（遠隔医療）にデータを連携する。(図中⑤)

サービス間連携における考慮点としては以下が考えられる。

- 各種サービス分野間での連携時においては、データモデルの統一など先端的サービス間で連携するデータ項目を調整することが必要となる。
- サービス間連携は、データ連携基盤が仲介して行うため、サービスとして要求する性能などを考慮の上で設計する必要がある。

(ｶ)サービスの運用性

- [課題①]リアルタイム性について

  匿名バンクによるパーソナルデータの管理では、パーソナルデータの持ち主によるオプトイン・オプトアウト操作があるため、リアルタイムのユースケースには適さない可能性がある。そのため、実際の業務を考慮した上での設計が必要となる。

(d)結果の考察・分析

匿名バンクとブローカーとの接続検証を行い、接続における課題を確認した。なお、スーパーシティにおける実装については、各エリアのデータ連携基盤との機能の整理や、同意管理機能の最適な方式を設計する必要があると考える。また、エリアでの導入の際には実環境、実業務に近い環境で実証実験を行い導入時の課題などを検証する必要があると考える。

(2)行政と民間事業者の情報連携に資する技術（ブロックチェーン）

ブローカーとブロックチェーン基盤(NEXCHAIN)の接続検証結果を記載する。ブローカーと NEXCHAIN 間の REST API での接続確認に加え、連携における課題などについても既存の製品情報などを基に机上検証を行った。

(a)利用ユースケース

D 市の構想にて「運動や食事データ等のライフログや医療データを連携することで健康～未病～治療 のサイクルをシームレスに繋ぐヘルスケアプラットフォームを構築。」と示されている。

これら先端的サービスの実現のためには、官民・官官・民民それぞれで幅広く生活全般をカバーする情報連携を行う必要がある。データ連携基盤においてはオープン API 等の活用により、基礎機能としてオープンな形での情報公開・取得が可能となる。一方で、ビジネスの内容等の機微な情報等については、一部関連事業者や自治体等ある程度クローズな範囲で安全に情報の公開・共有を実現するため、証跡の管理やブロックチェーン（分散台帳技術）による改ざん検知等が活用できると考える。

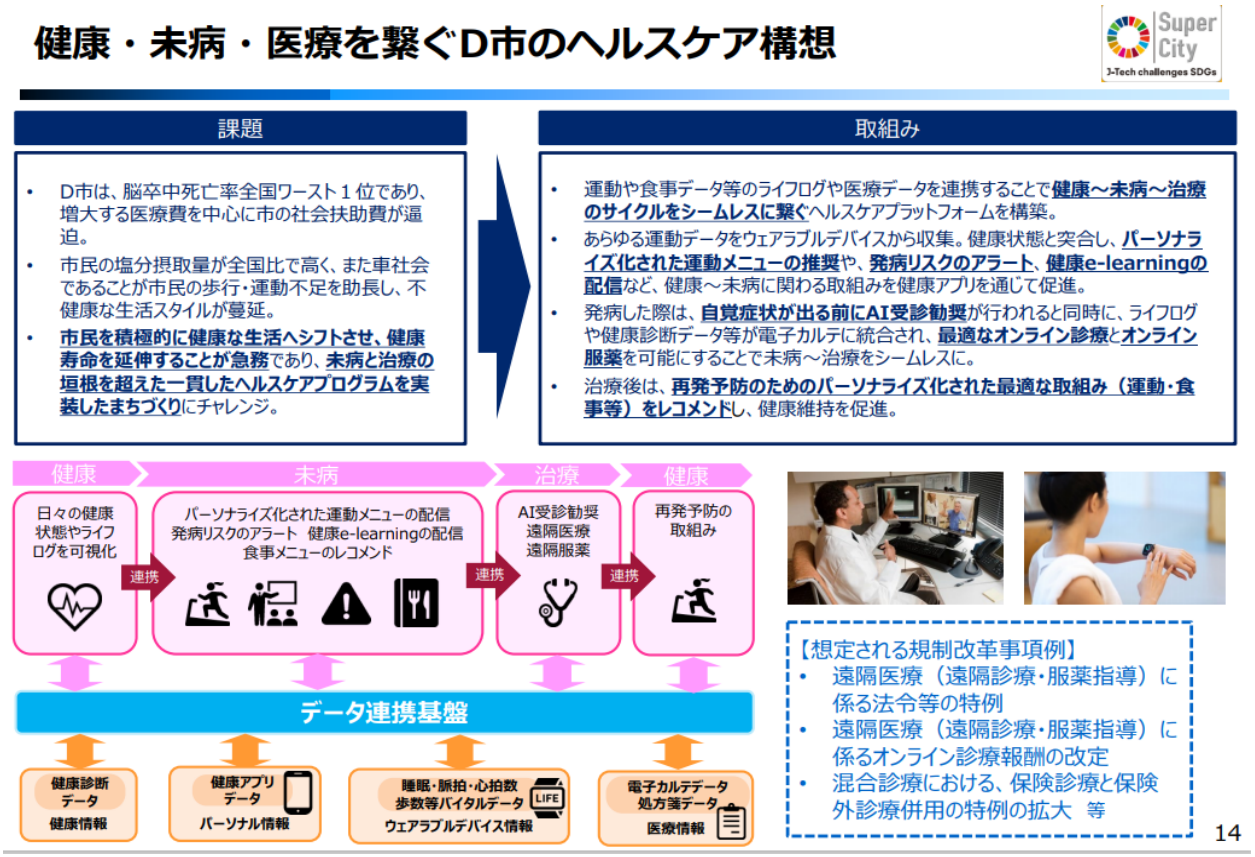


図 8-2-2-5　D 市のユースケース

D 市の構想にブロックチェーン技術を適用した場合のイメージ例を「図 8-2-2-6」に示す。ブロッ

クチェーン基盤を経由してデータ連携を行うことで履歴管理などを行うことができる。

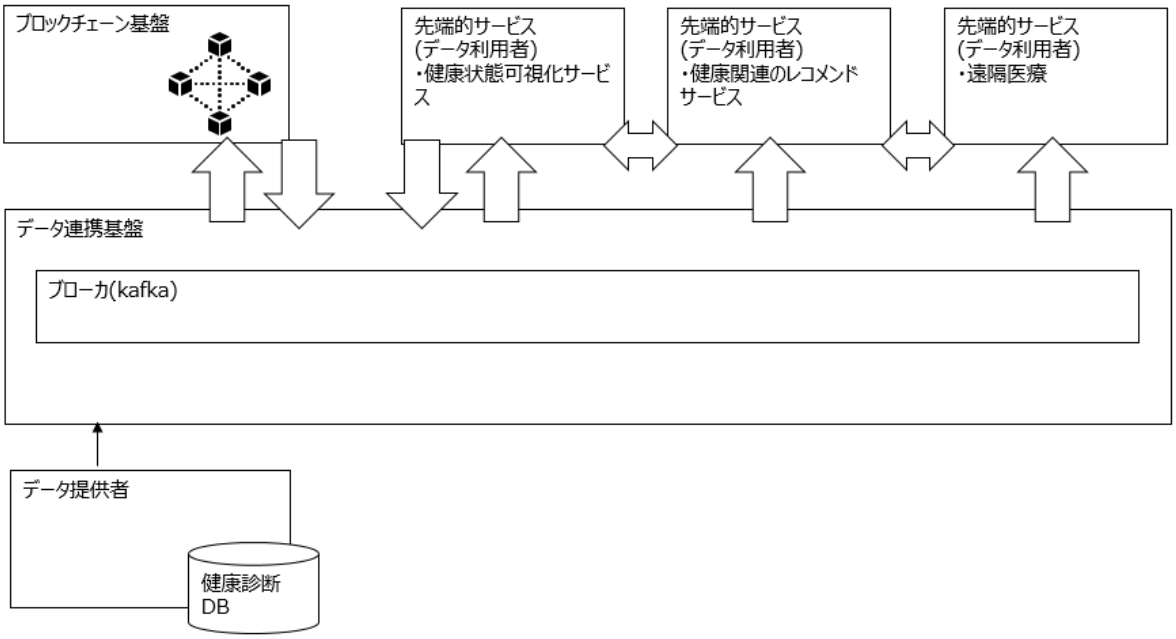


図 8-2-2-6　D 市ユースケースへの適用事例

(b)検証環境の説明

(ｱ)検証環境のシステム構成

ブロックチェーン基盤(NEXCHAIN)とブローカーとの実機検証のシステム構成を「図 8-2-2-7」に示す。

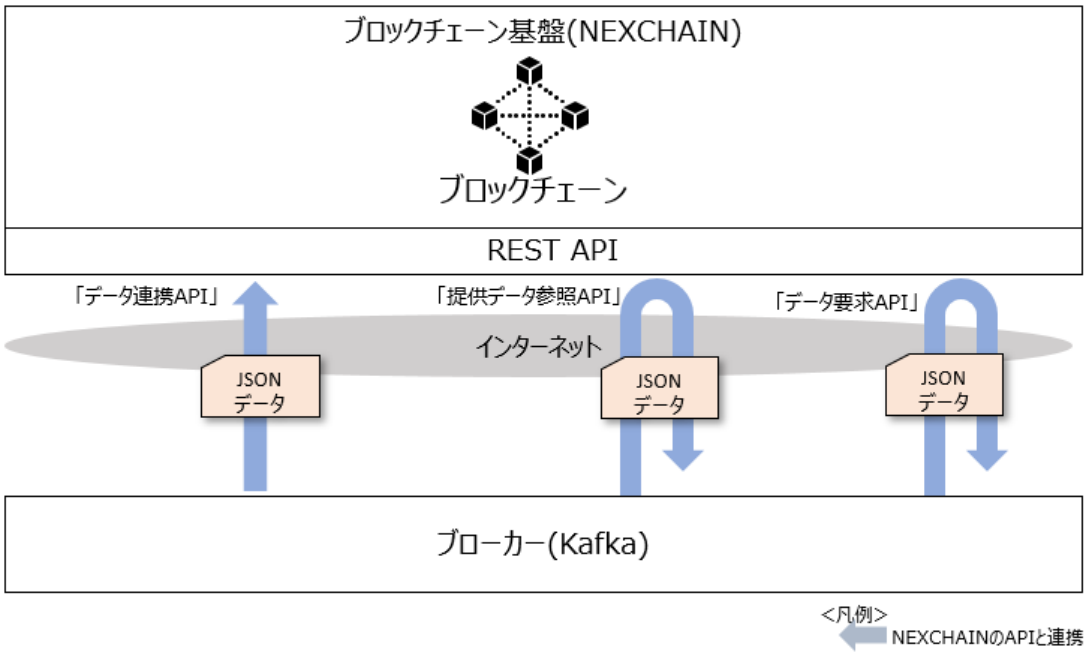


図 8-2-2-7　検証環境のシステム構成

(ｲ)各機能概要

- ブローカー(Kafka)：データ連携を中継する。
- ブロックチェーン基盤(NEXCHAIN)：ブロックチェーンを用いてデータ及びデータの処理履歴を管理する。

実機検証ではNEXCHAINの以下APIを使用した。

- データ連携API：ブローカー側から実行し、NEXCHAINにデータを登録するAPI
- 提供データ参照API：ブローカー側から実行し、NEXCHAINに格納されているデータを参照するAPI
- データ要求API：ブローカー側から実行し、NEXCHAINに格納されているデータをJSON形式で取得するAPI

(c)接続検証結果

実機/机上検証を行った結果を基に以下の観点で見解を記載する。

- データ連携基盤との接続性
- APIの共通ルールとの整合
- データモデルの方針との整合
- サービス間連携の実現性
- サービスの運用性

(ｱ)データ連携基盤との接続性

データ連携基盤との接続仕様、接続検証結果について記載する。実機検証では以下の方法で確認した。

- データ提供者相当のサーバからコマンドを実行しAPIを介してブロックチェーン基盤にデータを登録する。
- データ利用者相当のサーバからコマンドを実行しAPIを介してブロックチェーン基盤からデータを取得する。

上記接続検証の結果、ブロックチェーン基盤のREST APIのインターフェースを使用し、ブロックチェーンを意識せずにブローカー(Kafka)データ登録・取得ができることが確認できた。

(ｲ)APIの共通ルールとの整合

実機検証の結果から、「(イ) API の共通ルール」で推奨している内容に該当するか確認を行った。その結果、以下の2点において該当することを確認した。

- 接続検証で使用したブロックチェーン基盤はREST APIでの呼び出しが可能で、本検証においてもデータ連携基盤側からREST API形式での呼び出しを行った。
- ブロックチェーン基盤はJSONファイルで受け取ることが可能で本検証でもデータ連携基盤からJSONファイルでデータ連携を行った。

なお、APIの公開性について、NEXCHAINコンソーシアムの会員企業に制限してAPIを公開している。これは企業間の情報連携基盤の性質上、お互い信頼している組織同士内でのクローズドな公開としているからである。スーパーシティで利用する場合でも同様にクローズドな形での公開になると考える。

(ｳ)データモデルの方針との整合

NEXCHAIN では、ブロックチェーン基盤側ではデータの中身については関与せず、利用システム側で登録データを規定できる。そのため、スーパーシティで活用する際には、連携データの仕様に関するサービス間の取り決めや標準データモデルの使用により、ブロックチェーン基盤を仲介したデータ連携を行うことができる。

(ｴ)サービス間連携の実現性

サービス間のデータ連携の実現性について「図 8-2-2-8」の例で説明する。

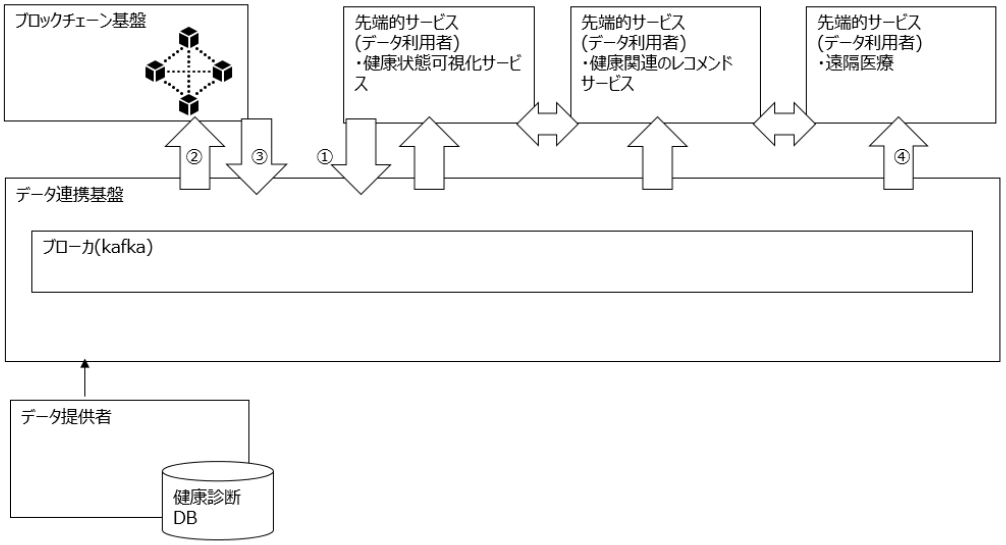


図 8-2-2-8　サービス間連携イメージ図

分野間のデータ連携は、ブローカーを仲介して行う。 加えてブロックチェーン基盤を仲介させることで履歴管理などが可能となる。連携の流れは以下のとおり。

- 先端的サービス（健康状態可視化サービス）からブローカーにデータを連携する（図中①）
- ブローカーからブロックチェーン基盤にデータを連携する。ブロックチェーンに記録される。（図中②）
- ブロックチェーン基盤からデータ連携基盤へデータを連携する（図中③）
- ブローカーから先端的サービス（遠隔医療）にデータを連携する。(図中④)

サービス間連携における考慮点としては以下が考えられる。

- 各種サービス分野間での連携時においては、サービス間で連携するデータ仕様（形式、データ項目）をブロックチェーン基盤と合わせて仕様調整する必要がある。
- 複数サービスの同時利用時においては、データ連携基盤、ブロックチェーン基盤を通した性能を考慮する必要がある。例えば、サービス提供型のブロックチェーン基盤を利用する場合、サービスが示す性能レベルが、要求性能を満たすのかを確認・検証する必要がある

(ｵ)サービスの運用性

A 市のユースケース事例に対して「(b)検証環境の説明」で記載したシステム構成で実現した場

合の運用面での課題と対策案を記載する。

- [課題①]データモデルの統一について
  各サービス同士で連携するデータモデルを取り決める必要がある。その際には、7 章で検討したような標準データモデルを活用することで取り決めに関するハードルを下げることが期待される。
- [課題②]ブロックチェーン基盤の整備主体について
  データ連携基盤においてどこに位置付けるのか、また整備主体を検討する必要がある。先端的サービス事業者個別に構築するのは重複投資回避の観点と運営の煩雑回避の観点からエリアのデータ連携基盤を整備する主体が中心となり整備することが望ましいと考える。

(d)結果の考察・分析

実機検証により、ブローカー(Kafka)とブロックチェーン基盤(NEXCHAIN)が REST API でデータ連携できることを確認できた。なお、実際にスーパーシティ、スマートシティのサービスを検討、設計、構築、運用する際には、各エリアのデータ連携基盤整備と合わせて最適な方式を設計する必要があると考える。

(3)生体認証を活用した本人認証技術

「8-1-4 調査結果」の技術の中から顔認証技術を用いた接続検証結果について記載する。

(a)利用ユースケース

A 市の構想にて通院及び配車予約を行うユースケース（図 8-2-2-9）では、個人を特定する本人認証が必要である。

通院及び配車予約のそれぞれのサービスでは、通常は個人 ID の入力などで個人を特定して予約を行うが、今回は顔認証を利用した本人認証を用いて接続検証を実施する。

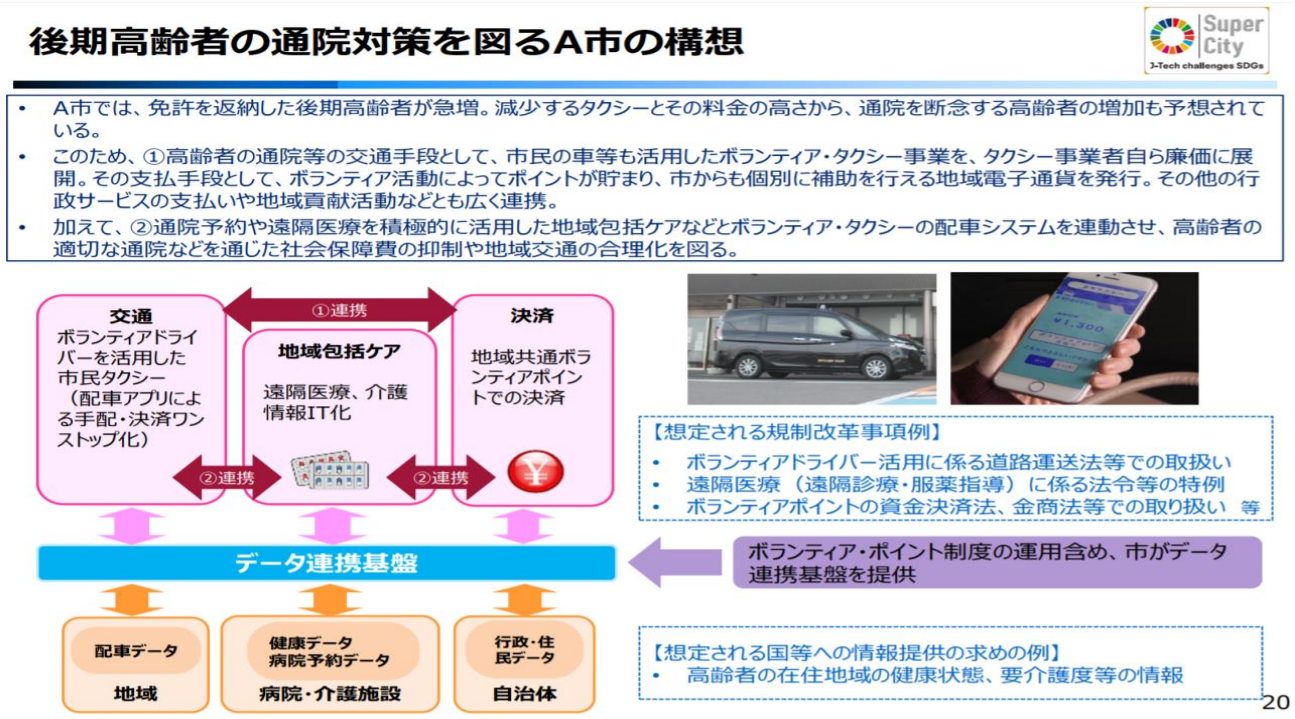


図 8-2-2-9　A 市のユースケース

(b)検証環境の説明

(ｱ)実機検証環境のシステム構成

A 市のユースケース事例を実現するための構成イメージを以下に記載する。

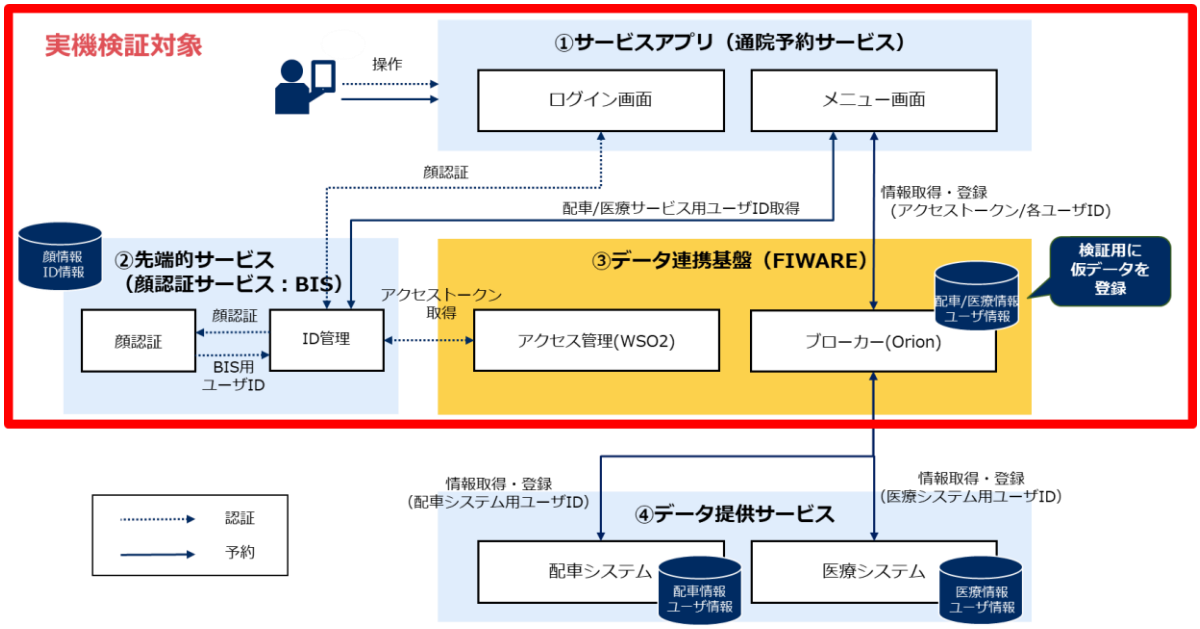


図 8-2-2-10　実機検証イメージ図

(ｲ)各機能概要

①サービスアプリ（通院予約サービス）

ユーザが利用するデモ用のモバイル AP サービス。

②先端的サービス（顔認証サービス(BIS：Bio-IDiom Services)）

顔認証を行う機能とユーザ ID 情報の管理を行うサービス。

③データ連携基盤(FIWARE)

データ連携を中継する。今回は FIWARE が所有するブローカー機能(Orion)とアクセス管理機能(WSO2)を使用する。

- データ提供サービス

配車予約を行う配車システムと診察予約を行う医療システムを対象とする。

（※）今回の接続検証の目的は先端的サービス（顔認証サービス）とデータ連携基盤の接続性を確認することであり、データ連携基盤とデータ提供サービスのインターフェースは省略する。

(c)接続検証結果

以下の観点で接続検証結果を記載する。

- データ連携基盤との接続性
- API の共通ルールとの整合
- データモデルの方針との整合
- サービス間連携の実現性
- サービスの運用性

(ｱ)データ連携基盤との接続性

先端的サービス(顔認証)とデータ連携基盤の接続性を確認するため、BIS(Bio-IDiom Services)と FIWARE との接続試験を実施した。接続試験としては、「(b)検証環境の説明」で記載したシステム構成で検証環境を構築し、以下のシーケンスでデータが連携できることを検証した。

1) サービスアプリが顔認証サービスに対して顔認証の要求を行う。

2) 顔認証サービスはデータ連携基盤からアクセストークンを取得する。顔認証を実施して特定した本人の顔認証用のユーザ ID とともにアクセストークンをサービスアプリに返却する。

3) サービスアプリからアクセストークンと顔認証用ユーザ ID をインターフェースして、データ連携基盤へ予約状況の取得依頼を行う。

4) データ連携基盤はアクセストークンの有効性を確認し、顔認証用ユーザ ID を基に予約状況を取得してサービスアプリに返却する。

5) サービスアプリは予約状況を表示し、データ連携基盤に対してユーザが選択した予約時間を基に予約依頼を行う。

6) 予約情報の更新処理を行い、結果をサービスアプリに返却する。

7) サービスアプリが予約結果を表示する。

上記のシーケンスにおいて、検証ポイントとなるのは以下の 2 点である。

2)で顔認証サービスがデータ連携基盤からアクセストークンを取得して使用できるか。

3)で取得したアクセストークンで認証をデータ連携基盤で行えるか。

データ連携基盤（FIWARE）の標準仕様として一般公開されているドキュメントに従って実装をした結果、上記 2 点は実現できたことを確認した。

この結果から接続性は問題ないと評価する。

以下に検証結果として、サービスアプリの実行画面を記載する。

図 8-2-2-11　実機検証結果

(ｲ)API の共通ルールとの整合

実機検証の結果から、「(イ) API の共通ルール」で推奨している内容に該当するか確認を行った。その結果、以下の 3 点において該当することを確認した。

1) 実機検証で使用した顔認証サービスは REST API での呼び出しが可能であり、本検証でもサービスアプリから REST API 形式での呼び出しを行った。

2) 顔認証を行うための顔画像は JSON ファイルで受け取ることが可能で本検証でもサービスアプリから JSON ファイルでデータ連携を行った。

3) 顔認証サービスからデータ連携基盤への接続において、＜データ連携基盤との接続性＞で記載したとおり、アクセストークンを活用した接続（OAuth2.0 認証方式）を行った。

(ｳ)データモデルの方針との整合

実機検証の結果から、「(ウ) データモデルの方針」で記載している内容に該当するかの確認を行った。その結果を以下に記載する。

本接続検証において、配車システムと医療システムで保持するデータは「表 7-3-5-1　データ項目構造の例(施設の例を抜粋)」に従ったデータ構造で作成した。作成したデータのうちキー項目となる情報を顔認証サービスに保持し、データ連携基盤が顔認証サービスからキー項目を取得した上で、配車システムと医療システムのデータを取得・更新を行った。この結果より、各データ提供者で保持するデータのキー項目を顔認証サービスに連携することで、「(ウ) データモデルの方針」に沿ったデータ連携が可能なことを確認した。

(ｴ)サービス間連携の実現性

実機検証ではデータ連携基盤が配車システムと医療システムの役割を代行し、データ連携基盤と顔認証サービスの接続性を確認したが、データ連携基盤・配車システム・医療システムが「(b) 検証環境の説明」にて記載した本来の役割を担ったケースでの机上検証を行う。ここでポイント

に挙げるのは、サービスアプリが顔認証サービスに対して ID 変換要求を行う点である。顔認証サービスが顔認証用ユーザ ID と医療システム用 ID、配車システム用 ID を紐付けて管理することで、サービスアプリは顔認証用ユーザ ID を基に処理要求を行いたいシステムのユーザ ID を取得することができる。これにより、複数サービスを跨ったデータ連携が可能となる。下図で、サービスアプリが顔認証サービスに対しての ID 変換要求する部分を赤線で示す。

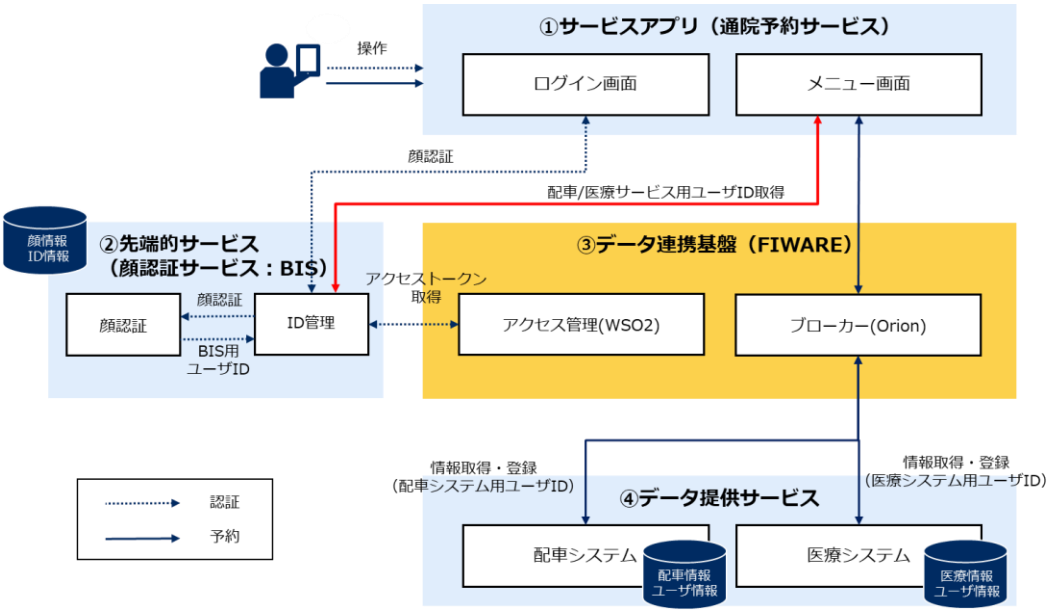


図 8-2-2-12　A 市ユースケース実装イメージ図

1)検出した課題と対策

上記の流れは、顔認証サービスが各サービスの ID を管理することで実現しているが、スーパーシティでは複数の認証システムの活用が想定される中で、各認証システムが ID 管理を行う方式では運用の煩雑化が懸念される。また、特定の製品で ID 管理を行うことはベンダロックしないというスーパーシティの思想に反する。そのため、ID 管理機能として共通的に利用できる基盤が必要となることが想定される。

(ｵ)サービスの運用性

A 市のユースケース事例に対して「(b)検証環境の説明」で記載したシステム構成で実現した場合の運用面での課題と対策案を記載する。

- ［課題①］顔認証システムの管理主体について

顔認証システム（本実機検証では BIS）では顔認証に必要な個人情報を保持するため、管理主体をどこに置くかが課題になる。実際に複数サービスを連携した実証実験の事例においても自治体での管理が難しいことから本格的な導入に至らないケースが発生している。
顔認証システムは複数サービス間で利用される位置付けとなるため、データ連携基盤同様に横断的な立場となる団体が管理主体となることが望ましい。

- ［課題②］データ提供サービスのユーザ ID 管理について

データ提供サービスで管理するユーザ情報に追加・更新・削除が行われた場合、＜サービス間

連携の実現性＞で挙げた ID を管理するシステムと乖離が生じてデータ連携が行えないことが懸念される。そのため、ID を管理するシステムと各データ提供サービスは常にデータ連携を行い、ユーザ情報の最新化を維持する仕組みの検討が必要となる。

- ［課題③］生体認証の選択、実施場所について

  顔認証は照度の条件等、認証を行う場所の環境に応じて認証精度に影響が及ぶ場合があり、実施場所の環境維持が課題事項となる。そのため、認証を行う場所での基本的な環境条件を把握しておく必要がある。また、例えば暗い場所では虹彩認証と組み合わせる等、他の認証技術と連携した仕組みの検討が必要となる。

- ［課題④］顔情報の管理・運用について

  顔認証を利用するにあたり、事前に登録した時点から長時間経過すると経年変化により、認証精度が維持できなくなることが想定される。そのため、利用者が定期的に登録画像を更新する運用を取り入れる必要がある。

(d)結果の考察・分析

以上の検証結果より、顔認証技術を用いて A 市のユースケース事例での接続性、実現性を確認できた。スーパーシティにおいては、複数サービスを跨ったデータ連携が期待される一方で、各データ提供者に ID・パスワードや ID カードの利用が必要となると利用者の利便性を著しく低下させる。本人確認において一貫して生体認証を活用できれば利用者の利便性向上が期待でき、かつ、なりすまし等によるサービスの安全・安心性を脅かす脅威に対する対策としても有効であると考える。生体認証の中でも本取り組みで検証した顔認証は ID・パスワードや ID カードの代替として様々なサービスで普及が進んでおり、スーパーシティとして複数のサービス間で共通的に活用することで更なる利便性の向上が期待できる。

(ｱ)サービス間連携の実現性

実機検証では FIWARE が配車システムと医療システムの役割を代行し、FIWARE と BIS の接続性を確認したが、FIWARE・配車システム・医療システムが「(b)検証環境の説明」にて記載した本来の役割を担ったケースでの机上検証を行う。

ここでポイントに挙げるのは、サービスアプリが BIS に対して ID 返還要求を行う点である。BIS が BIS 用ユーザ ID と医療システム用 ID、配車システム用 ID を紐付けて管理することで、サービスアプリは BIS 用ユーザ ID を基に処理要求を行いたいシステムのユーザ ID を取得することができる。これにより、複数サービスを跨ったデータ連携が可能となる。

「図 8-2-2-13」にて、サービスアプリが BIS に対しての ID 変換要求する部分を赤線で示す。

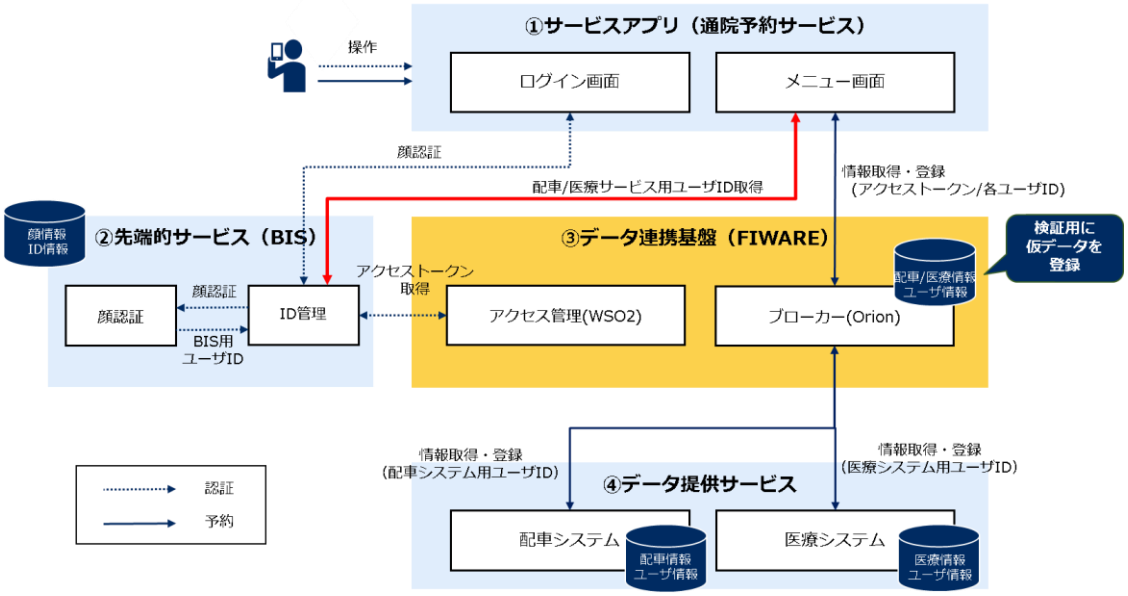


図 8-2-2-13　A 市ユースケース実装イメージ図

1)検出した課題と対策

上記の流れは、BIS が各サービスの ID を管理することで実現しているが、スーパーシティでは複数の認証システムの活用が想定される中で、各認証システムが ID 管理を行う方式では運用の煩雑化が懸念される。また、特定の製品で ID 管理を行うことはベンダロックしないというスーパーシティの思想に反する。そのため、ID 管理機能として共通的に利用できる基盤が必要となることが想定される。

(ｲ)サービスの運用性

A 市のユースケース事例に対して「(b)検証環境の説明」で記載したシステム構成で実現した場合の運用面での課題と対策案を記載する。

- [課題①]顔認証システムの管理主体について

顔認証システム（本実機検証では BIS）では顔認証に必要な個人情報を保持するため、管理主体をどこに置くかが課題になる。実際に複数サービスを連携した実証実験の事例においても自治体での管理が難しいことから本格的な導入に至らないケースが発生している。そのため、顔認証システムの管理主体をどこにするかという課題が挙げられる。

顔認証システムは複数サービス間で利用される位置付けとなるため、データ連携基盤同様に横断的な立場となる団体が管理主体となることが望ましい。

- [課題②]データ提供サービスのユーザ ID 管理について

データ提供サービスで管理するユーザ情報に追加・更新・削除が行われた場合、(ｱ)サービス間連携の実現性で挙げた ID を管理するシステムと乖離が生じてデータ連携が行えないことが懸念される。そのため、ID を管理するシステムと各データ提供サービスは常にデータ連携を行い、ユーザ情報の最新化を維持する仕組みの検討が必要となる。

- [課題③]生体認証の選択、実施場所について

顔認証は照度の条件等、認証を行う場所の環境に応じて認証精度に影響が及ぶ場合があり、実施場所の環境維持が課題事項となる。そのため、認証を行う場所での利用者が基本的な環境条件を把握しておく必要がある。

また、例えば暗い場所では虹彩認証と組み合わせる等、他の認証技術と連携した仕組みの検討が必要となる。

- ［課題④］顔情報の管理・運用について

  顔認証を利用するにあたり、事前に登録した時点から長時間経過すると経年変化により、認証精度が維持できなくなることが想定される。そのため、利用者が定期的に登録画像を更新する運用を取り入れる必要がある。

(e)結果の考察・分析

以上の検証結果より、顔認証技術を用いてA市のユースケース事例の実現性を確認できた。スーパーシティにおいては、複数サービスを跨ったデータ連携が期待される一方で、データ提供者ごとにID・パスワードやIDカードの利用が必要となると利用者の利便性を著しく低下させる。本人確認において一貫して生体認証を活用できれば利用者の利便性向上が期待でき、かつ、なりすまし等によるサービスの安全・安心性を脅かす脅威に対する対策としても有効であると考える。生体認証の中でも本取り組みで検証した顔認証はID・パスワードやIDカードの代替として様々なサービスで普及が進んでおり、スーパーシティとして複数のサービス間で共通的に活用することで更なる利便性の向上が期待できる。

(4)データ取引市場を活用したデータ流通を促進する技術

(a)利用ユースケース

「国家戦略特区 スーパーシティ構想について」では、整備するデータ連携基盤はデータ分散方式を想定しており、都市間でのデータ連携、共有の迅速化が期待されている。

都市間でのデータ共有を考えた場合として、都市間及び複数の都市で管理しているデータを相互検索するユースケースにて接続検証を実施する。

(b)検証環境の説明

(ｱ)実機検証環境のシステム構成

上記ユースケース事例を実現するための検証環境のイメージを以下に記載する。

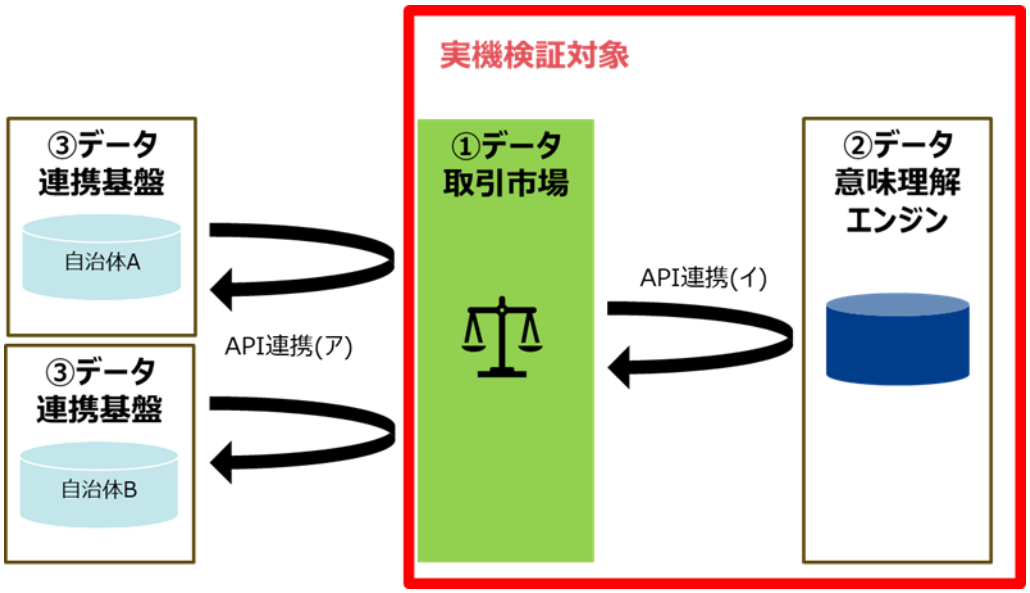


図 8-2-2-14 検証環境のイメージ

(ｲ)各機能概要

1) 先端的サービス（データ取引市場）

都市間、分野間でのデータ連携を仲介する。

本検証では、データ取引市場として、EverySense Pro を用いて検証を行う。

2) 先端的サービス（データ意味理解エンジン）

データ連携時のデータの検索性向上を支援する。

3) データ連携基盤

データ連携を中継する。ブローカー機能(Orion)にて検証済み。

(ｳ)検証内容

スーパーシティの各エリア間及び分野横断でのデータ利活用を想定し、データ意味理解エンジン API を実装（API 連携(イ)）したデータ取引市場と各自治体が整備するデータ連携基盤を相互接続し（API 連携(ア)）、データ意味理解エンジンで語彙やデータの意味を理解し、データカタログの検索性能が実装前と後でどう変化が起きるかを検証する。各自治体が整備するデータ連携基盤とデータ取引市場との相互接続（API 連携(ア)）は、2019 年度 SIP 第 2 期「スマートシティに

おけるパーソナルデータと産業データのデータ取引市場による共有基盤の実証」研究テーマ 2 において、本書の「調査結果（ア）データ仲介機能」で検証した Orion を対象とした検証が完了しているため、今回はデータ取引市場とデータ意味理解エンジンとの接続（API 連携(イ)）について、検証を実施する。

(c)接続検証結果

API 連携(イ)の接続検証結果について、以下のとおり記述する。API 連携(イ)の接続構成は以下となる。

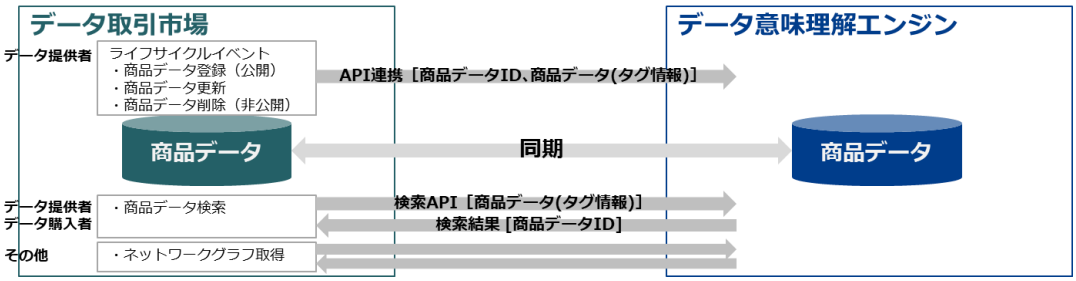


図 8-2-2-15　API 連携(イ)の接続構成

(ア)接続性

データ取引市場サービスとデータ意味理解技術との相互接続（API 連携(イ)）を実施した。

AI 技術によって、データの意味を理解し、関連データを効率的と広範囲に見出すことを目的に、データ意味理解エンジンの使用前と使用後を比較した。

図 8-2-2-16　データ意味理解エンジン接続前

図 8-2-2-17　データ意味理解エンジン接続後

例として、データ取引市場のサイトから同一条件（”終身医療保険”）で検索をかけ、データ意味理解エンジン接続前と後で検索結果の比較を行った。

- 関連データの検索結果（ヒット数）

  データ意味理解エンジン接続前　：　65 件

  データ意味理解エンジン接続後　：　100 件

件数の増加要因としては、データ取引市場に登録されているデータカタログの情報（タイトル、説明文、サンプルファイル等）をデータ意味理解エンジンが理解したためであり、データ意味理解エンジンが関連データをより広範囲に見つけ出していることが確認できる。

同様に、データ利用者の効率的なデータ検索、データ提供者とのマッチング率の向上、データ利用機会の創出を目的として、データ取引市場に「商品データ」を登録する際、データの意味を AI で理解した上で、“検索キーワード”の推奨値を導出する。

サンプルとして、「生命保険の未加入者特徴データ」を商品データとして登録する。

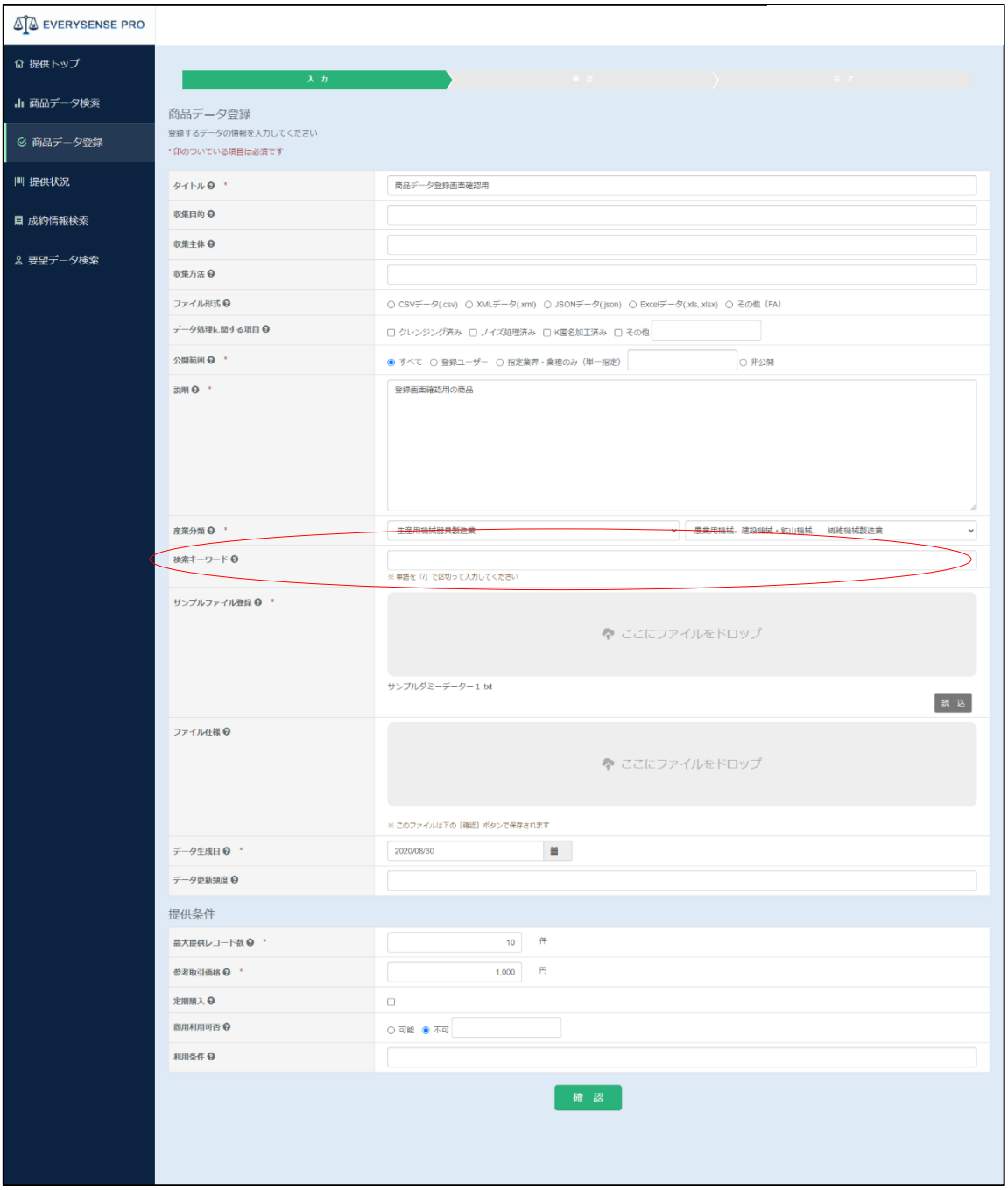


図 8-2-2-18　商品データ登録画面

通常は、データ提供者が想定する「検索キーワード」を手動で入力する必要がある。本検証では、あえて「ブランク」で登録し、データ意味理解エンジン接続前と続後を比較する。

図 8-2-2-19　データ意味理解エンジン接続前の商品データ登録完了画面

データ意味理解エンジン接続前では、検索キーワードが未入力のため、結果はブランクとなる。

図 8-2-2-20　データ意味理解エンジン接続後の商品データ登録完了画面

データ意味理解エンジン接続後では、データの意味を AI が理解し、“アンケート”“個人起点データ”“終身医療保険”“金融商品”“国勢調査”“商圏分析”“ジオデモ”“ターゲティング”といったキーワードが生成される。

これにより、データ提供者の利便性が大幅に向上するとともに、データ利用者にとっても検索キーワードが増えることで様々なニーズに合致したデータ検索につながる。

(ｲ)サービス間連携

各種サービス分野間での連携時の接続、データ連携、技術的課題について検証する。

今回は、データ取引市場に掲載されている単一のデータのみを対象として検証を実施したため、その有効性の検証は限定的だが、今後各スーパーシティ間や分野間でのデータ連携が進むと、より広範に複数データを組み合わせた利用が求められ、さらにデータ意味理解技術が有効になると考える。また、複数のデータで学習を積み重ねることにより、さらにデータ意味理解の精度の向上が見込める。一方で、データ連携が進み、読み込むデータ量の増加に伴い、結果の表示速度な

ど性能面へ影響を考慮したシステム設計が求められる。

(d)結果の考察・分析

以上の検証結果より、データ意味理解技術の利用によって、データの検索精度の向上、データ提供者とのマッチング精度の向上につながることを、一定程度確認することができた。

スーパーシティにおいては、業界・業種、都市や地域横断のデータ利活用が期待される一方で、データカタログが分散して管理される状況でデータ検索に非効率性が生じ、利用者の利便性を大きく損なうおそれがあり、検索データへの到達性やデータ意味理解技術を活用することで、分野間・地域間で異なるデータの意味の定義や形式を推定しデータ検索の効率性が増すことで、この課題克服につながると考える。

また、さらなるデータ意味理解技術の活用として、データ提供者がデータ登録を完了する前に、意味推定によってPII（Personally Identifiable Information）[123]らしきデータ項目を特定し、登録を未然に防いだり、自動的に該当するPII列をマスクすることも可能となり、データ提供者、データ利用者双方に対し、データを扱う際に懸念されるリスクの低減に寄与することも想定される。

スーパーシティの各都市、地域間及び業界・業種といった分野横断でデータ利活用を促進、さらに安全・安心なデータ共有のためにも、高度なデータ共有の仕組みとして本技術が有効であると考える。

8-3. まとめ

本章では、「先端的サービスを支える先進技術」としてエリアに推奨する先進技術として「情報銀行(PDS)」「ブロックチェーン」「顔認証」「データ取引市場におけるデータ意味理解技術」について、スーパーシティにおける活用における課題やデータ連携基盤との親和性について評価した。

なお、本調査業務で評価対象とした「先進技術」を各エリアでのスーパーシティ整備に導入する際には、データ連携基盤との機能の住み分けや実装方法の検討を各エリアの実情に合わせて最適な方式を設計する必要がある。

[123] PII：個人を特定できる情報（Personally Identifiable Information）
https://jp.twilio.com/docs/glossary/what-is-personally-identifiable-information-pii

9. 調査結果（オ）基盤を長期運用する上で必要となる事項

本章では、2-7（オ）その他データ分散型のデータ連携基盤を運用する上で必要となる事項の調査結果として、データ連携基盤を長期運用する上で必要となる課題の抽出・整理及びその解決策の検討を行った。9-1 に、スーパーシティの取り組みを検討している自治体及び民間事業者に対し、実現するデータ連携基盤を長期運用する上で課題となる事項をヒアリングした結果と、ヒアリングにより抽出された課題及びその解決策案を示す。9-2 には、本調査業務体制の参画事業者が有する知見を基に検討した陳腐化しない運用の仕組みを示し、同様に 9-3 に安心してデータを扱える仕組みの検討結果を示す。

9-1. データ連携基盤を長期的に運用するための課題整理

自治体に限らず、データ連携基盤運用事業者、データ利用者やサービス提供者など、データ連携基盤の運用に携わるステークホルダがデータ連携基盤を長期的に運用するには様々な課題がある。ここでは、長期的に運用する際の課題を内閣府地方創生推進事務局より公表されている「『スーパーシティ』構想について」[124]（以下、「スーパーシティ構想」とする。）にある「後期高齢者の通院対策を図る A 市の構想」、「観光を起点とする B 市のスーパーシティ構想」、「被災者を受け入れる C 市の防災拠点構想」、「健康・未病・医療を繋ぐ D 市のヘルスケア構想」のユースケースから検討し、当該課題の解決策について、「スマートシティにおける連携レイヤー」の「ルール」、「機能」、「データ」、「データ連携」、「アセット」及び「セキュリティ・認証」の観点から分類する。分類した課題を基に実際にデータ連携基盤（同等な基盤を含む）。を導入している団体又は自治体、データ連携基盤の導入を検討している団体又は自治体に対して、当該課題に対しての解決方法や対処方法の検討について、ヒアリングを行い明らかにする。

9-1-1. ヒアリング方針とヒアリング対象

(1)ヒアリング方針

本ヒアリングは、データ連携基盤を導入している団体又は自治体、データ連携基盤の導入を検討している団体又は自治体に対して、データ連携基盤を長期的に運用するための課題を明確にするために行うが、データ連携基盤の導入状況やスーパーシティの取り組みにおける役割によって、課題に対する取り組みの検討・実施状況が、各ヒアリング先により異なると推測した。そのため、各ヒアリング先の回答者の知見やノウハウに応じてヒアリング項目を変更できるよう複数のパターンを準備し、ヒアリングを実施した。

(a)「企画」、「運用」、「利用」の分類

本ヒアリングでは、回答者に合わせて、「企画」、「運用」、「利用」の 3 種類のヒアリング項目を設定した。「企画」、「運用」、「利用」の詳細は、表 9-1-1-1 のとおり。

「企画」は、スーパーシティの取り組みやデータ連携基盤の運用の企画に携わっている回答者を対象とした。具体的には、自治体等のスーパーシティやデータ連携基盤の推進主体といった、「スマートシティにおける連携レイヤー」のスーパーシティの戦略・政策立案、ルールの策定、組織編制、ビジネス検討を主導する役割を指す。データ連携基盤の導入実績がある回答者とない回答者がいることが推測されるため、それぞれの状況に合わせてヒアリングを準備した。

「運用」は、データ連携基盤又は同等な基盤の導入実績があり、かつ、当該基盤を運用している回答者を対象とした。具体的には、データ連携基盤又は同等な基盤の推進主体を指す。「企画」よりも ICT やデータ連携基盤技術の運用に携わる又は当該技術に関する知見のある回答者を対象とした。「運用」では、「スマートシティにおける連携レイヤー」のスマートシティの機能、データ、データ連携、

[124] 情報提供：内閣府国家戦略特区 HP 「スーパーシティ」構想について　supercity.pdf (kantei.go.jp)

アセット、セキュリティに関する運用の視点を中心に、ヒアリング項目を準備した。

「利用」は、データ連携基盤又は同等な基盤を利用している回答者を対象とした。具体的には、データ連携基盤によるデータ連携機能又は同等な基盤を活用したサービスの提供者やデータ連携基盤へのデータ提供者を指し、公共・公益機関等を想定した。データ連携基盤の利用者の意見として、データ連携基盤に期待するサービスレベルや、評価、改善及び運用効率化等に関する課題についてヒアリング項目を準備した。

表 9-1-1-1　「企画」、「運用」、「利用」の詳細

分類	対象	回答者
企画	「スマートシティにおける連携レイヤー」のスーパーシティの戦略・政策立案、ルールの策定、組織編制、ビジネス検討を主導する役割を有する回答者。	• データ連携基盤推進主体（自治体など）
運用	「スマートシティにおける連携レイヤー」のスマートシティの機能、データ、データ連携、アセット、セキュリティに関する運用の役割を有する回答者。	• データ連携基盤推進主体（自治体など） • データ連携基盤整備事業者（民間企業、スマートシティ推進協議会など）
利用	データ連携基盤を利用し、サービス提供やデータ提供を行う回答者。	• データ提供者（民間企業、公共・公益機関など） • サービス提供者/データ利用者（民間企業など）

(b)ヒアリング項目の設計

ヒアリング項目は、複数のユースケースにてデータ連携基盤を長期に運用していくための課題を仮説立案し、図 3-2-2-2 に沿って各レイヤーに分類したのち、表 9-1-1-2 に示す大項目へ整理することで設計した。大項目は、さらに、表 9-1-1-1 に示す「企画」「運用」「利用」に合わせて課題を細分化した。

ユースケースは、内閣府地方創生推進事務局より公表されている「『スーパーシティ』構想について」にある「後期高齢者の通院対策を図る A 市の構想」、「観光を起点とする B 市のスーパーシティ構想」、「被災者を受け入れる C 市の防災拠点構想」、「健康・未病・医療を繋ぐ D 市のヘルスケア構想」を採用した。各ユースケースに対し、コンソーシアム各社の知見を持ち寄り、データ連携基盤の長期運用における仮説課題の抽出を行った。

表 9-1-1-2　ヒアリング大項目及び対象の分類（〇：ヒアリングの対象項目）

ヒアリング項目（大項目）	企画	運用	利用
スマートシティ/スーパーシティ戦略の実現	〇	〇	－
エコシステム環境※の構築	〇	－	－
データ管理	〇	〇	〇
アセット管理	－	〇	－
基盤サービスレベルの検討	－	－	〇
プライバシーポリシーの検討	〇	－	－
基盤サービスの評価	－	〇	〇
基盤サービスの改善（長期運用、保守）	－	〇	〇
運用効率化	－	－	〇

※エコシステム環境：スマートシティ/スーパーシティ推進組織、各種サービス、都市 OS などの事業者が、共通目的の下、1 つのスマートシティ/スーパーシティ事業内で協調・協創関係にあること

図 9-1-1-1 から図 9-1-1-3 に、「企画」「運用」「利用」に分類した各仮説課題の「スマートシティにおける連携レイヤー」へのマッピング結果を示す。本事業では図 3-2-2-2 の「機能」、「データ」、「データ連携」、「アセット」を主な検討対象とし、「スマートシティにおける連携レイヤー」の「スマートシティ機能」、「スマートシティデータ」、「スマートシティデータ連携」、「スマートシティアセット」の各層に焦点を当て課題を整理した。

一方で、データ連携基盤を長期に運用する上での「機能」、「データ」、「データ連携」、「アセット」の仮説課題は、その前提となる課題が図 3-2-2-2 の「戦略・政策」、「ルール」、「組織」、「ビジネス」に潜在していることも想定される。そのため、ヒアリング項目の検討にあたっては、「戦略・政策」、「ルール」、「組織」、「ビジネス」に起因する潜在課題を、法律・条令等による「ルール」と、スマートシティ戦略・政策、組織組成等による「組織・ビジネス」に整理し、抽出した仮説課題のマッピングを行った。

なお、ヒアリング先が容易に回答しやすくなるよう、仮説課題をそのままヒアリング項目には設定せず、できるだけ表現をかみ砕きヒアリング項目を策定した。仮説課題、ヒアリング項目の一覧を、別紙オ-1「企画」「運用」「利用」に示す。

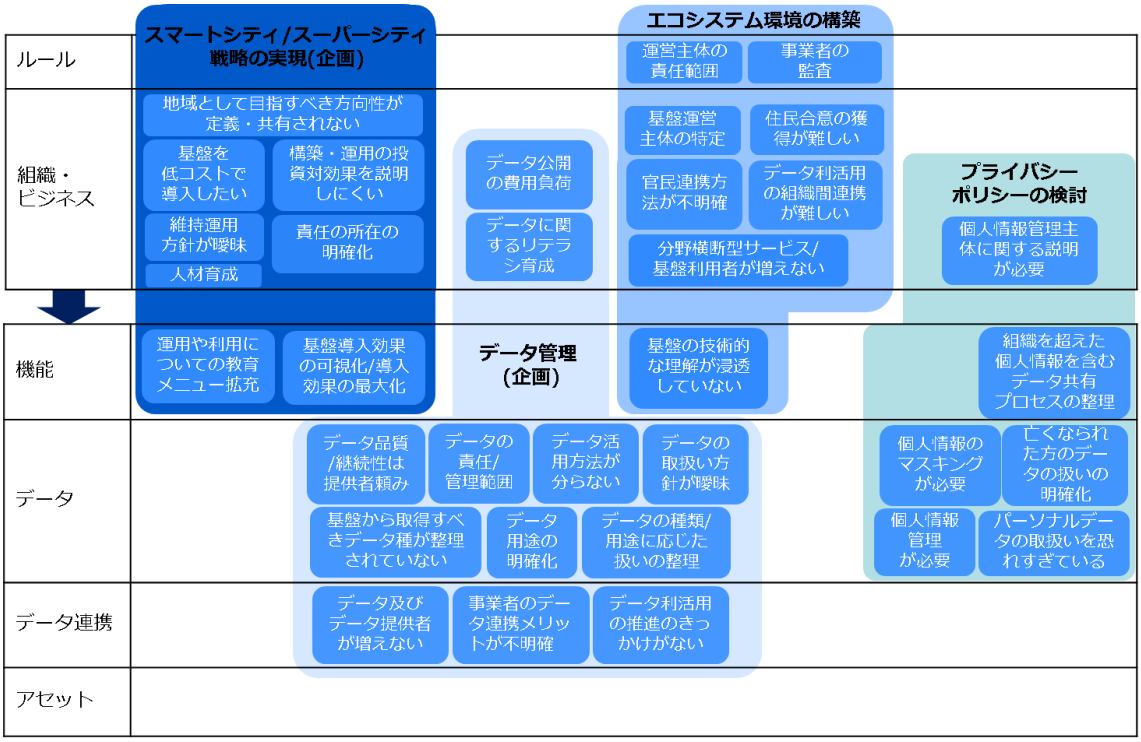


図 9-1-1-1　仮説課題のレイヤーマッピング（企画）

「企画」には、「スマートシティ/スーパーシティ戦略の実現」、「データ管理（企画）」、「エコシステム環境の構築」、「プライバシーポリシーの検討」の大項目に分類した仮説課題を図 9-1-1-1 に示すとおり各層へマッピングした。仮説課題の詳細は、別紙オ-1「企画」を参照のこと。なお、「データ管理」に関する課題は、課題の主な発生段階を基に「データ管理（企画）」と「データ管理（運用）」に分類したが、データ管理はデータ連携基盤の長期運用に際し重要度の高い項目であるため、回答者から課題を広く引き出すべく、「企画」「運用」ともに「データ管理（企画）」と「データ管理（運用）」の両方の仮説課題を基にヒアリング項目設計を行った。ヒアリング項目の詳細は、別紙オ-1「企画」及び別紙オ-2 を参照のこと。

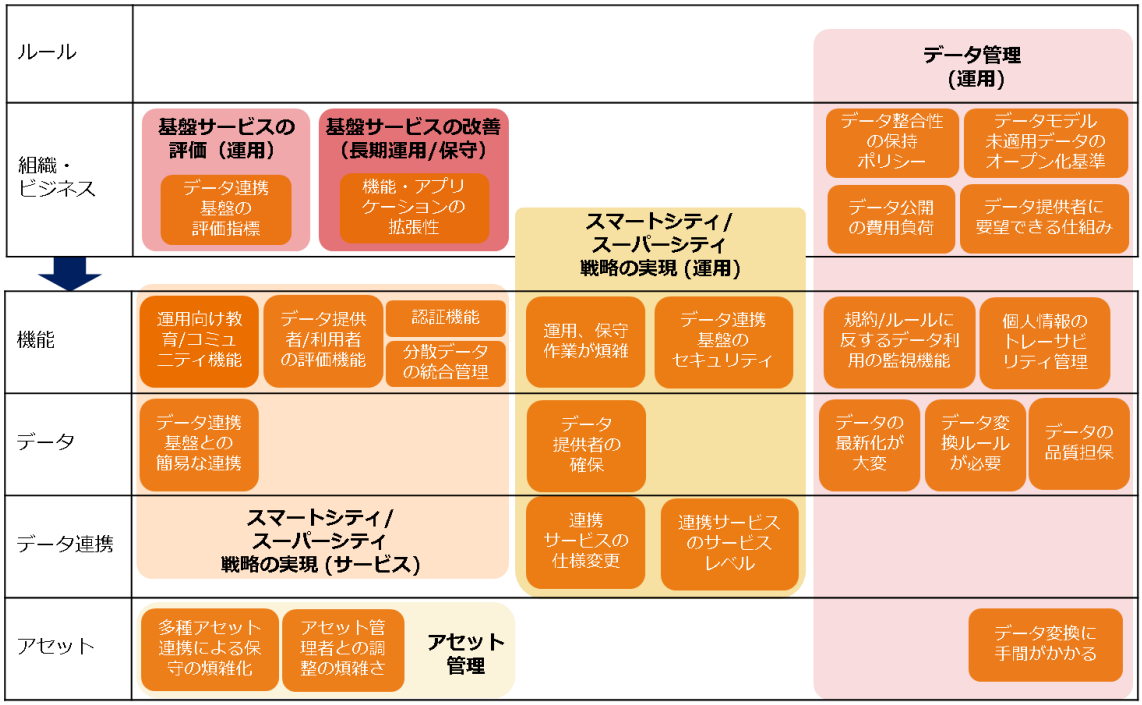


図 9-1-1-2　仮説課題のレイヤーマッピング（運用）

「運用」には、「スマートシティ/スーパーシティ戦略の実現（運用）（サービス）」、「データ管理（運用）」、「基盤サービスの評価（運用）」、「基盤サービスの改善（長期運用/保守）」、「アセット管理」の大項目に分類した仮説課題を図 9-1-1-2 に示すとおり各層へマッピングした。仮説課題の詳細は、別紙オ-1「運用」を参照のこと。

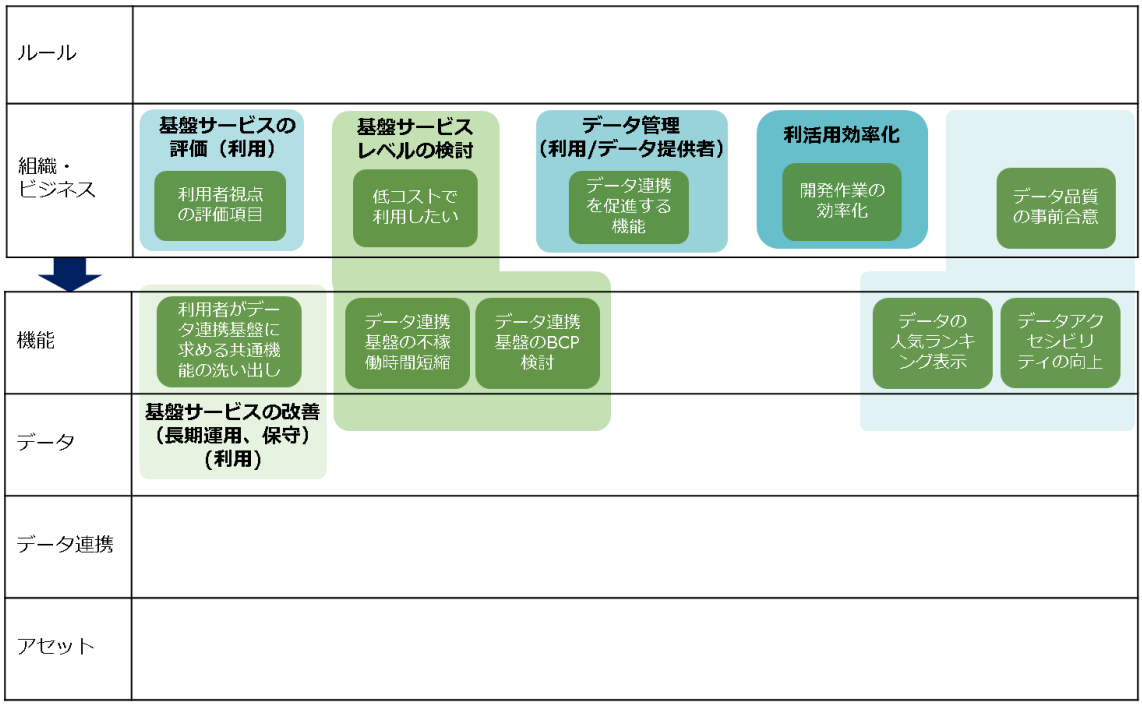


図 9-1-1-3　仮説課題のレイヤーマッピング（利用）

「利用」には、「データ管理（利用）」、「基盤サービスレベルの検討」、「基盤サービスの評価（利用）」、「基盤サービスの改善（長期運用/保守）」、「運用効率化」の大項目に分類した仮説課題を図 9-1-1-3

に示すとおり各層へマッピングした。仮説課題の詳細は、別紙オ-1「利用」を参照のこと。

(c)ヒアリングシート

(a)(b)を踏まえ、本ヒアリングにて使用するヒアリングシートを別紙オ-2 のとおり作成した。ヒアリングは共通編と、表 9-1-1-1 に沿った企画編、運用編、利用編で構成し、仮説課題に対する回答者の意見をヒアリングする選択回答方式と、回答者の自由な意見をヒアリングを通して収集する自由回答方式とした。

(2)ヒアリング先

本調査事業におけるヒアリング先として、次にあげる団体等において、データ連携基盤の運用に関する知見又は課題を既に持つと仮定し選定した。データ連携基盤の運用組織として、様々なデータを集約し、API を通じてデータを公開する基盤を運用している団体（2 件）、API 仕様や開発者ポータルサイトの仕様等に知見がある地域組織コミュニティの有識者（1 件）、スマートシティやシステムアーキテクチャに精通している有識者（1 件）、スマートシティに関与している（検討中を含む）官公庁（6 件）の合計 10 件にて行った。なお、本報告書において、各ヒアリング先や戦略等が特定されないようヒアリング先を明示して述べないものとする。

9-1-2. 自治体・団体におけるデータ連携基盤の長期運用課題

(1)ヒアリング結果

9-1-1 を基に、自治体・団体へデータ連携基盤の長期運用に関する課題のヒアリングを実施した。ヒアリングの回答者と、データ連携基盤の導入状況や知見、「企画」「運用」「利用」分類への回答状況を表 9-1-2-1 に示す。

表 9-1-2-1 回答者一覧

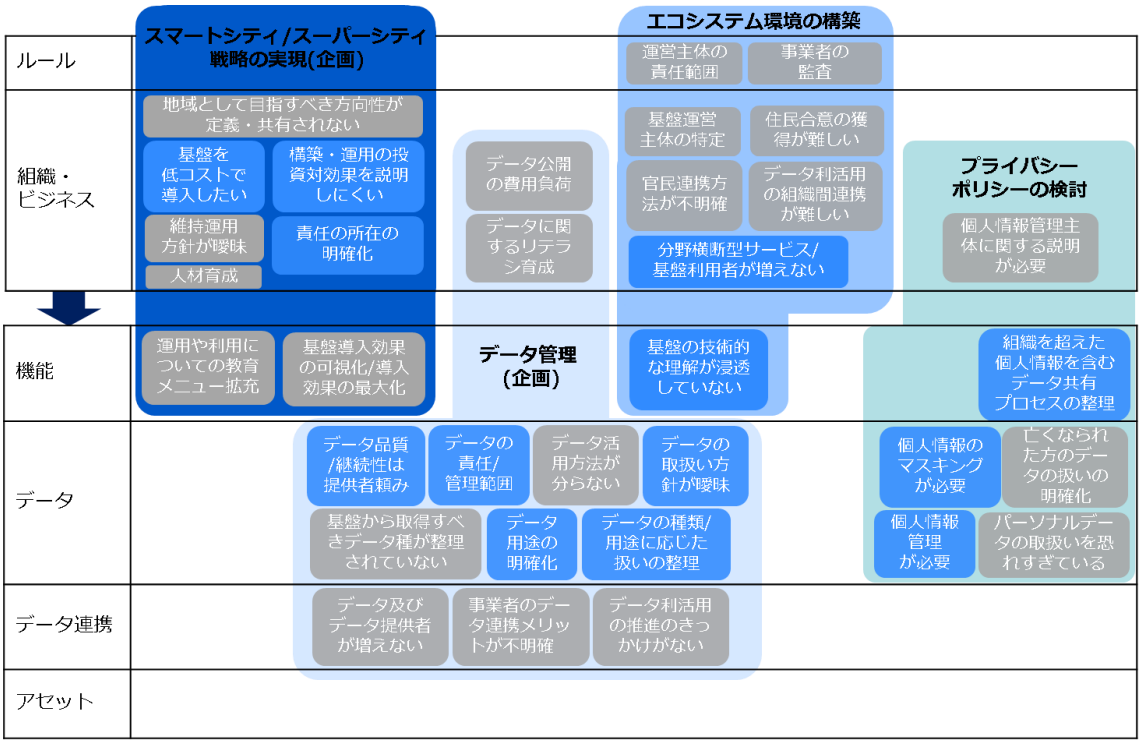
No.	回答者	データ連携基盤 導入状況	企画	運用	利用	備考
			回答	回答	回答	
1	自治体 A	導入する計画がある (決定事項)	○	―※	―	―
2	自治体 B	導入を検討している (未決定)	○	―※	―	―
3	自治体 C	導入を検討している (未決定)	○	―※	―	―
4	自治体 D	導入を検討している (未決定)	○	―※	―	―
5	自治体 E	導入する計画がある (決定事項)	○	―※	―	―
6	自治体 F	―	○	○	○	―
7	団体 G	既に導入している	○	○	○	データ提供者
8	団体 H	既に導入している	―	○	―	―
9	有識者 I	導入する計画がある (決定事項)	―	○	○	API 仕様や開発者ポータルサイトの仕様等に知見がある有識者(データ利用者)
10	有識者 J	―	○	―	―	スマートシティやシステムアーキテクチャに精通されている有識者
総計(回答自治体、団体数)			8	4	3	

※「データ管理(運用)」に関する課題のみ回答

「企画」「運用」「利用」に対するヒアリングの結果を、以下の(a)～(c)に示す。なお、本章では重点課題として、表 9-1-1-2 の大項目ごとに、自治体・団体から特に選択回答数の多かった上位 2 つの仮説課題を抽出(ただし、選択回答数が同数である場合は、同数の全ての仮説課題を抽出)した。加えて、仮説課題以外で、ヒアリングを通して自治体・団体から新たに得られた課題を(d)に示す。仮説課題ごとの選択回答数及びヒアリング結果の詳細は、別紙オ-1 を参照のこと。

(a) 企画

図 9-1-2-1 に、「企画」のヒアリング項目のうち、特に選択回答数の多かった仮説課題の抽出結果を示す。選択回答数や、選択理由の詳細は別紙オ-1「企画」を参照のこと。



(青色：選択回答数の多かった仮説課題、灰色：選択回答数の少なかった仮説課題)

図 9-1-2-1 「企画」ヒアリング結果

次に、大項目別に選択回答数の多かった仮説課題と、自治体・団体から得られた選択理由を示す。

(ア) スマートシティ/スーパーシティ戦略の実現（企画）

最も選択回答数が多かった仮説課題は、「基盤を低コストで導入したい」「構築・運用の投資対効果を説明しにくい」であり、回答数は 5 票であった。次点は「責任の所在の明確化」であり、回答数は 4 票であった。いずれも「組織・ビジネス」に分類される。

1) 組織・ビジネス

「基盤を低コストで導入したい」では、スマートシティの収益化やビジネスモデルの構築が各都市で課題になっている中で、データ連携基盤の構築や持続運用にかかる費用への関心が多く寄せられた。加えて、データ連携基盤の導入効果が明確でなく、数値として定量的に測れないため、自治体の予算を確保が難しいという意見も得られた。導入効果を測るために、ベストプラクティスのユースケースがあればよいとの意見もあった。さらに、費用面の課題は、導入フェーズよりも運用フェーズで深刻になりやすく自治体はデータ連携基盤を導入する際に、長期的な運用を見通して検討する必要があるとの意見も得られた。

「構築・運用の投資対効果を説明しにくい」では、データ連携基盤の導入効果について、参画する事業者数やサービス内容で変化する部分が大きく明確に示しにくいという意見が寄せ

られた。導入効果を明確にするために、データ連携基盤を活用したサービスの成功事例を、その取り組みの背景や目的といったストーリー併せて展開してほしいというコメントが得られた。ある自治体で成功したサービスそのものを他の自治体で導入するのではなく、その取り組みの背景や目的といったところを横展開する方が、自治体としては活用しやすいという意見も寄せられた。加えて、躓いた部分などを明らかにした失敗事例や、都市間連携の状況も展開されると良いとのことであった。

また、民間事業者のベネフィットを明らかにすべきとの声もあった。行政の立場では社会課題の解決を目的としてデータ連携基盤の導入を考えていると説明できるが、民間事業者が参画する上では、そのベネフィットが明らかになった方が参画促進につながるとのことであった。

「責任の所在の明確化」では、個人情報の取り扱いやデータ連携基盤におけるプライバシールールの明確化に関心が寄せられた。データ連携基盤では、行政の保有するデータを外部で活用していくため、個人情報やセキュリティに対する信頼性を損なわないよう配慮が必要となる。サイバー攻撃だけでなく、データ連携する事業者や相互接続している相手が信頼できるのか、どこまでデータを提供してよいのか等、様々な事項を明確に定めて信頼性や安全性を確保する必要があるとの意見が得られた。また、データ連携基盤で扱う際にリスクがあるデータの定義があるとよいとのコメントもあった。

加えて、データの責任についてのルールメイキングにも多くの意見が寄せられた。データ提供者とデータ連携基盤の運用主体が異なる場合のリスクヘッジや、事故初声時の責任等、透明性の高いルールが必要であるとの指摘も得られた。

ルールメイキングの仕方として、ルールの共通部分は政府が設計し、地域で協議を重ねる中でローカライズしていく部分があるような柔軟性の高い形での運用が望ましいとの意見もあった。また、信頼性や安全性の基準として、自治体としては、どこまでやれば信頼性や安全性を担保したと認められるのか、国が基準を提示し、国として基準を守る方法を示してほしいとのコメントもあった。

(ｲ)エコシステム環境の構築

最も選択回答数が多かった仮説課題は、「組織・ビジネス」の「分野横断型のサービス/基盤利用者が増えない」であり、回答数は 5 票であった。次点は「機能」の「基盤の技術的な理解が浸透していない」であり、回答数は 3 票であった。以下に詳細を示す。

1)組織・ビジネス

「分野横断型のサービス/基盤利用者が増えない」では、データ連携基盤が長期に広く活用されるためには、データ連携基盤がどのような役に立つものか、市民や庁内に向けて説明していく必要があるとの意見が得られた。加えて、民間事業者の参入を促すために、データ連携基盤の活用を通じて、民間事業者にどのようなベネフィットが生まれるのか明らかにする必要があるとのことであった。住民や民間事業者へのベネフィットが明確になれば、住民の合意や役割分担などは大きな障壁にならないと感じているとの意見も寄せられた。

2)機能

「基盤の技術的な理解が浸透していない」については、政府が示す方針が様々な立場の人に向けて発信する情報であるがゆえに、「API」など難しい表記が多く、自治体の理解が追い付かない状況があり、データ連携基盤の導入や長期的な利活用の障壁になっているとの意見が得られた。自治体側が理解を深めていくために、難しい表記をかみ砕いて説明できる人材が必要と

なり、その育成や役割を誰が担うかも課題であるとのことだった。

(ｳ)データ管理（企画）

最も選択回答数が多かった仮説課題は、データの品質に関する「データ品質/継続性は提供者頼み」「データ提供者の責任/管理範囲」と、データの取り扱いルールに関する「データの取り扱い方針が曖昧」「データ用途の明確化」「データの種類と用途に応じた保管の仕方や扱いの整理が必要」であり、いずれも回答数は 5 票であった。いずれも図 3-2-2-2 では「データ」に分類される。以下に詳細を示す。

1)データ

「データ品質/継続性は提供者頼み」と「データ提供者の責任/管理範囲」では、運用に向けた品質担保への課題意識が寄せられた。今後、データ連携基盤を活用したビジネスに取り組む事業者が増えていったとしても、連携されるデータの信頼性が不十分であれば、万が一の場合、そのビジネスの信頼性までも損なわれてしまうおそれがある。当該サービスが行政サービスの場合、行政が信頼を失うことになる。そうなら内容、データ品質や責任範囲を定めておくことが重要であるとの意見が得られた。加えて、自治体が様々なシステムをパッケージ標準で導入している中で、当該パッケージの中に含まれるダミーコードにより、不要あるいは意味をなさないデータが連携データに含まれるといった事例があり、データ連携基盤を通して得られるデータが生きているデータなのか、また、信頼性の高いデータなのか整備する必要があるとの指摘もあった。

データの取り扱いルールに関する「データの取り扱い方針が曖昧」「データ用途の明確化」「データの種類と用途に応じた保管の仕方や扱いの整理が必要」は、「規約/ルールに反するデータ利用の監視機能」と併せて選択されるケースが大半であった。データ連携基盤でデータを取り扱う上では、データ連携基盤を導入する段階でルールを明確に定める必要があるのはもちろんのこと、策定したルールが順守されているか確認する仕組みが必要であるという意見が多く寄せられた。例として、規約やルールに沿わないデータの利用を確認できる仕組みの構築が望ましいとのコメントが得られた。

(ｴ)プライバシーポリシーの検討

最も選択回答数が多かった仮説課題は、「機能」の「組織を超えた個人情報を含むデータ共有プロセスの整理」と、「データ」の「個人情報管理が必要」、「個人情報のマスキングが必要」であり、いずれも回答数は 2 票であった。以下に詳細を示す。なお、プライバシーポリシーはスーパーシティ取り組み状況により検討状況も様々であり、詳細な検討は今後実施するため未回答という自治体も複数あった。

1)機能

「組織を超えた個人情報を含むデータ共有プロセスの整理」では、民間事業者を巻き込んでいくための課題に意見が寄せられた。個人情報の取り扱いについては民間事業者間で考え方に温度差があり、リスクをとってまで個人情報を扱うか、という点で意識の差異が生じている。行政としては、民間事業者に対して個人情報の利活用を促すことができない中、どのように最適化を図っていくかが課題となっているとのことだった。

加えて、個人情報を属性情報に加工する際も、誰がデータを加工するのか、データ加工を担う事業者にメリットはあるか、といった課題があるとの意見が得られた。

また、自治体特有の組織間連携の難しさとして、各課の責任範囲が法律などで明確に定まっており、課をまたぐ連携はできない（やってはいけない）というのが実態であるとの知見が得られた。そのため、自課/自事業で手に入れた個人情報を他の目的で使うことはそもそも不可能であり、法改正できればいいが、それを待っていられないから、どういう形なら実現可能かをいろいろな自治体が考えている状態であるといったコメントも寄せられた。

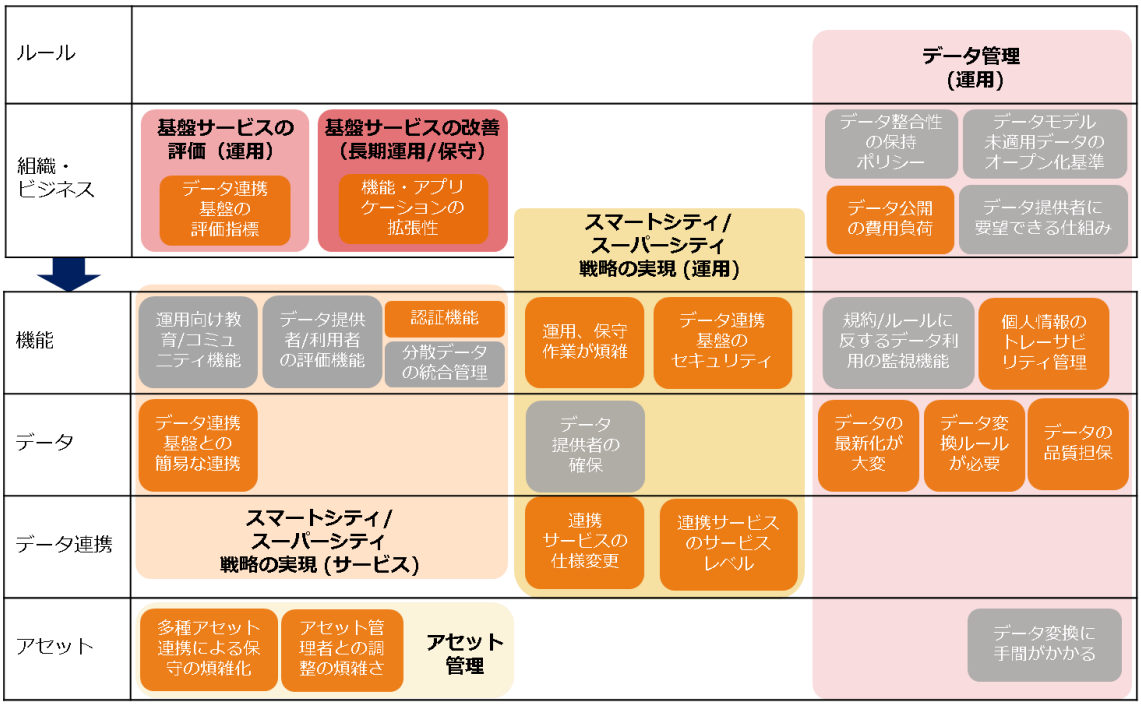
2)データ

「個人情報管理が必要」では、自治体としてデータ連携基盤で個人情報を管理したり連携したりするという点において、安全性に不安があるという意見が寄せられた。自治体によっては、個人情報をオプトイン方式で取得することを考えているとのことであった。

また、「個人情報のマスキングが必要」では、匿名加工の必要性について、個人情報は匿名加工をやるべきと考えており、生のデータをサービス提供事業者に渡して活用するというのは現時点で考えていないとのコメントがあった。匿名加工に際しては、その方法について国が基準を示してほしいという意見も得られた。

(b)運用

図 9-1-2-2 に、「運用」のヒアリング項目のうち、特に選択回答数の多かった仮説課題の抽出結果を示す。選択回答数や、選択理由の詳細は別紙オ-1「運用」を参照のこと。



（橙色：選択回答数の多かった仮説課題、灰色：選択回答数の少なかった仮説課題）

図 9-1-2-2 「運用」ヒアリング結果

次に、大項目別に選択回答数の多かった仮説課題と課題背景を示す。

(ｱ)スマートシティ/スーパーシティ戦略の実現（運用）

最も選択回答数が多かった仮説課題は、「機能」の「運用、保守作業が煩雑」及び「データ連携

基盤のセキュリティ」、「データ連携」の「連携サービスの仕様変更」及び「連携サービスのサービスレベル」であり、回答数は 2 票であった。以下に詳細を示す。

1)機能

データ連携基盤を長期に運用する上での不安や課題として、「運用、保守作業が煩雑」という点が多く寄せられた。データ連携基盤は単品開発で完了するシステムではなく、サービスや連携データの拡充を通して、年月をかけて発展させるべきシステムである。長期にわたり、利活用が促進されるシステムとして成立していくためには、初期設計の段階で運用、保守作業が煩雑にならないような検討が必要と考えられる。

また「データ連携基盤のセキュリティ」では、セキュリティ対策として実施すべき事項はある程度見えているが、どの程度の人出とコストをかけ、どこまで実施すべきかの決定が悩ましいという意見が得られた。

2)データ連携

「連携サービスの仕様変更」では、データ連携基盤と連携するサービスに仕様変更が生じた際の、データ連携基盤の運用ルールが決まっていないことを課題と感じる意見が寄せられた。データ連携基盤が連携する先端的サービスや各種アセットは、データ連携基盤とは別の主体が別の業務目的で運用している。

相手先の業務都合等により、サービスやデータの仕様変更が発生する場合があるため、いかなるケースにおいても安定したデータ連携を提供できるよう、連携サービス側の仕様変更に対する対処方針をあらかじめ決める必要がある。

「連携サービスのサービスレベル」では、データ連携基盤と連携するサービスやアセット類のサービスレベル（稼働時間、障害復旧時間等）に差異があることが課題として挙げられた。連携先のサービスやアセットは 24 時間稼働が原則となるが、状況により、業務システムのように休日や夜間の運用を想定していないサービスもあると考えられる。連携先サービスのサービスレベルや性能の差異を補完するための検討が必要である。

(ｲ)スマートシティ/スーパーシティ戦略の実現（サービス）

最も選択回答数が多かった仮説課題は、「機能」の「認証機能」、「データ」の「データ連携基盤との簡易な連携」であり、回答数は 3 票であった。以下に詳細を示す。

1)機能

データ連携基盤を長期に運用する上で必要とされるサービスとして、データ提供者及びデータ利用者の「認証機能」が必要であるという意見が多く寄せられた。データ連携基盤を活用したサービス等における本人確認/認証には、マイナンバーのような公的証明書等、様々な手段があると想定される。データ連携基盤の長期的な運用にあたり最適な証明手段を検討する必要がある。

2)データ

「データ連携基盤との簡易な連携」として、データ連携基盤上に簡易にデータを取込むツールが必要であるという回答が得られた。データ連携基盤と既存システムとの連携においては、既存システムの一部のデータを変換しデータ連携基盤上もしくは連携するプラットフォームにデータを蓄積ケースが想定される。あらゆるデータ連携パターンに簡易に対応できるよう、

プラットフォームやデータ連携基盤へデータを取り込むための施策（ツール等）が必要と考えられる。

(ｳ)データ管理（運用）

データ連携基盤の運用主体（団体G、団体H、有識者I）にて最も選択回答数が多かった仮説課題は、「組織・ビジネス」の「データ公開までの費用負荷」で、回答数は3票であった。次点は、「機能」の「個人情報のトレーサビリティ管理」、「データ」の「データの品質担保」であり、回答数は2票であった。

また、データ連携基盤の企画主体（自治体A～自治体F）からの意見を含めて選択回答数が多かったものは、「データ」の「データの最新化が大変」「データ変換ルールが必要」であった。以下に、課題背景や自治体か和えられた回答を示す。

1)組織・ビジネス

データ連携基盤の運営主体の目線で、分散されたデータを管理・利活用する上での不安や課題として、「データ公開までの業務・費用負荷」が多く選択された。データ連携基盤を長期間継続して運用し、常に新鮮なデータを提供するためには、データ連携基盤を経由してデータを連携又は公開するためにかかる作業負荷や費用負荷をできるだけ下げることが望まれる。

2)機能

長期間にわたるデータ連携においては、データを所有する個人への充分な配慮やデータのトレーサビリティ管理が必要と想定される。「個人情報のトレーサビリティ管理」では、データ連携基盤で個人情報を扱う上で、データ連携のトレーサビリティについて期待することとして、「プライバシーの配慮（例：アクセス制御との連動）」、「データ利用に関するアクセス権の付与」、「個人情報の利用規約を超えた不正な利用を追跡・管理する仕組みの構築」が望ましいという意見が多く寄せられた。

特に「個人情報の利用規約を超えた不正な利用を追跡・管理する仕組みの構築」は、「企画」において自治体からも多く意見の寄せられた、「データ管理（企画）」の「データ」の課題である「規約/ルールに反するデータ利用の監視機能」とも重複するため、特に重要視すべき点であると考えられる。

3)データ

「データの品質担保」については、今後、データを活用したビジネスに取り組む事業者が増えていくことを考えると、データに信頼性がなければそれらビジネスを潰すことになるうえ、行政サービスの場合は行政が信頼を失うことになることから、データの品質担保を重要視するという意見が得られた。

また、「データの最新化が大変」については、データを最新化するための作業負荷と、最新化による利便性のバランスの難しさについて意見が寄せられた。データを最新化する作業はデータ量が増加するほど大変な作業になる。作業負荷が大きく、他の業務がなかなか進まなくなるため、データを最新化する作業負荷の低減を重視しているといったコメントもあり、データの最新化作業が既にデータを取り扱う自治体・団体にとって大きな作業負荷となっているとのことであった。

「データ変換ルールが必要」では、データフォーマットの基本計画策定について、自治体ごとに整備すると都市間連携がしづらくなるため、例として日付データと場所データの順番とい

った細部も含め、国が主体となって策定してほしいといった意見が得られた。フォーマットの策定にあたっては、現場で行われている既存業務にできる限り合わせた形式になることが望ましいとの声もあった。

また、データフォーマットが示される際は、各分野の活用事例とセットで展開されると、自治体としてはその後のオペレーションが容易となるといった指摘が寄せられた。加えて、そもそもデータフォーマットの理解が普及していない実態があり、データ公開の足並みが自治体内部で揃いづらいといったコメントもあった。自治体内部で足並みを揃えた活動を促進するために、例えば国がベースレジストリを公開し活用事例を展開することで、国の取り組みのもと活用利点を説明しやすく、自治体内部の理解が進み協力体制を築きやすくなるとのことであった。

(ｴ)アセット管理

アセット管理に関する課題では、仮説課題として設定した「多種アセット連携や拡張による保守対象の煩雑化」と「アセット管理者との調整の煩雑さ」の両方とも多く選択され、回答数は 2 票であった。どちらも「アセット」に分類される仮説課題である。以下に詳細を示す。

1)アセット

データ連携基盤を長期に運用するための、アセット（例：センサー、監視カメラ映像、既存のデータサーバ等）管理に関する課題として、「多種アセット連携や拡張による保守対象の煩雑化」が多く選択された。長期間にわたるデータ連携基盤とのデータ連携においては、IoT 機器の増加により、アセットとデータ連携基盤の接続が増加する可能性がある。アセット管理者とデータ連携基盤の運用事業者が異なる場合、システム拡張に伴い両者の密な連携が必要となることから、「アセット管理者との調整の煩雑さ」が課題として多く選択されたと考えられる。また、ヒアリングではアセットそのものの保守作業に加え、ネットワーク負荷の軽減についても課題として意見が寄せられた。

(ｵ)基盤サービスの評価（運用）

データ連携基盤を継続的に運用するためには、利用者からのフィードバックによる継続的な改善が必要となる。本項目では、回答者が実施している、又は実施が必要と考えている基盤の評価・改善プロセスについてヒアリングを行った。詳細を以下に示す。

1)組織・ビジネス

「組織・ビジネス」に分類されるデータ連携基盤の評価・改善プロセスとして、「住民及びデータ連携基盤利用者を対象とした満足度調査」、「費用対効果の測定」、「利便性向上調査」が効果的であるとの意見が得られた。

2)機能

「機能」に分類されるデータ連携基盤の評価・改善プロセスとして、「データ欠損検知や正規化の測定」、「API に関する継続的な監視・エラー検出」、「ユーザ数、アクセス数、ダウンロード数等による利用効果の計量」が効果的であるとの意見が得られた。このような機能をデータ連携基盤に搭載することで、長期的な評価を簡易に行うことができ、改善プロセスの実行につなげやすいと考えられる。

(カ)基盤サービスの改善（長期運用/保守）（運用）

基盤を長期運用していくためには、長期運用を見据えたデータ連携基盤の拡張性を有することが必要である。本項目では、データ連携基盤の運用事業者が拡張性について期待することについてヒアリングを行った。詳細を以下に示す。

1)組織・ビジネス

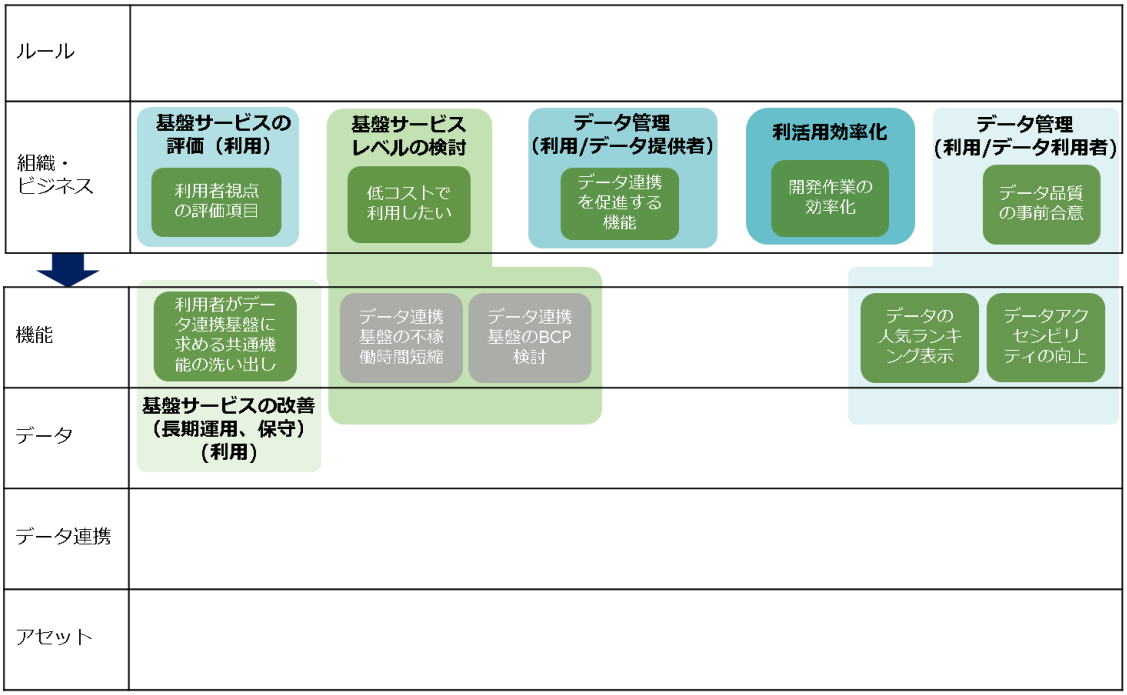
「組織・ビジネス」に分類される拡張性として、「相互運用性の確保を目的としたAPI・データ仕様等の公開（相互運用性）」、「PDCAサイクルによる段階的かつ定期的なサービス改善作業（継続性）」が多く選択された。

2)機能

「機能」に期待される拡張性として、「拡張に伴う作業（システム連携やデータ変換等）の簡略化（運用・保守性）」、「利用頻度に応じた柔軟な拡張や縮退が選択可能（性能）」が多く選択された。前者については、初期設計の段階から作り込んでいくことが求められると想定される。後者については、幅広い拡張を容易にするため、例としてビルディングブロックの概念に基づくシステム実装の仕方や、相互連携を実現するために採用すべき標準規格を明確にしておく必要があると考えられる。

(c)利用

図 9-1-2-3 に、「利用」のヒアリング項目のうち、特に選択回答数の多かった仮説課題の抽出結果を示す。選択回答数や、選択理由の詳細は別紙オ-1「利用」を参照のこと。



（緑色：選択回答が得られた仮説課題、灰色：選択回答数の少なかった仮説課題）

図 9-1-2-3　「利用」ヒアリング結果

次に、大項目別に選択回答が得られた仮説課題と課題背景を示す。

(ｱ)データ管理（利用（データ提供者））

データ提供者が長期的にデータ提供を継続するためには、データ提供者がデータ連携をしやすくする配慮が必要である。「データ連携を促進する機能」として、データ提供者がデータ連携基盤を活用してデータを提供する際に、どのようなことを期待するかヒアリングを行った。詳細を以下に示す。

1)組織・ビジネス

「組織・ビジネス」に分類される期待事項として、「データ提供のインセンティブがあること（例：課金制度、評価への反映）」が選択された。データ連携基盤に多くの有効なデータを連携するためには、データ提供者の協力は欠かすことができない。品質の良いデータを継続的に提供してもらうためにも、インセンティブは重要であると考えられる。

2)機能

「機能」に分類される機能として、「データ連携基盤にデータを連携するための準備作業（メタデータ作成）の支援環境があること（例：メタデータ作成支援ツール）」が選択された。データ提供者の作業を支援するような環境の整備により、連携データの拡充につながると考えられる。

(ｲ)データ管理（利用（データ利用者（サービス提供者）））

最も選択回答数の多かった仮説課題は、「機能」に分類される「データアクセシビリティの向上」であり、回答数は2票であった。次点で選択回答数の多かったものは、「組織・ビジネス」の「データ品質の事前合意」と、「機能」の「データの人気ランキング表示」であった。以下に課題背景を示す。

1)組織・ビジネス

「データ品質の事前合意」では、サービス提供者（データ利用者）が、データ連携基盤を経由して分散されたデータを利用する際に、当該データの品質をあらかじめ把握できることが望ましいという意見が得られた。

2)機能

「データアクセシビリティの向上」では、サービス提供者（データ利用者）が、データ連携基盤を経由して分散されたデータを利用する際に当該データへのアクセスのしやすさを期待するという意見が得られた。具体的には、データカタログ機能や、AIを実装した類語推定機能（異なる用語で記述された同義語に対し、類似語として推定し提示する機能）が望ましいとの回答であった。

また、「データの人気ランキング表示」として、データ取得時に当該データが有益であることがわかりやすいような仕組みの構築が期待されている。例として、人気ランキングや利用者数の表示があるとよいとの意見が得られた。

(ｳ)基盤サービスレベルの検討

最も選択回答数の多かった仮説課題は、「組織・ビジネス」に分類される「低コストで利用したい」であり、回答数は2票であった。以下に課題背景を示す。

1)組織・ビジネス

データ連携基盤を利用し続けてもらうためには、利用そのもののコストを下げることや、利用しやすいことが必要である。コスト低減例として、データ連携基盤を活用したシステム開発を容易にする開発支援ツールやドキュメントの提供等が効果的であると考えられる。

開発支援ツールのようなデータ連携基盤の運用者や利用者を支援するツールの提供は、「運用」の仮説課題大項目「スマートシティ/スーパーシティ戦略の実現（サービス)」の「データ連携基盤との簡易な連携」においても「データ連携基盤上に簡易にデータを取込むツール」の提供が求められている。したがって、データ連携基盤の長期的な利活用促進においては、“使いやすさ”を補う各種ツールの提供が有効であると考えられる。

(ｴ)サービスの評価（利用）

データ連携基盤を継続的に運用するためには、利用者からのフィードバックによる継続的な改善が必要となる。本項目では、データ提供者又はデータ利用者（サービス提供者）として、データ連携基盤を評価する上で評価項目となる点のヒアリングを実施した。結果を以下に示す。

1)組織・ビジネス

回答者から寄せられた評価項目として、「サービスを使いたい利用者が多いこと」、「API ドキュメントなどが揃っていること」、「データガバナンスについて明示されていること」が挙げられた。

特に、「サービスを使いたい利用者が多いこと」を目指すにあたっての課題として、データの提供者や利用者の絶対数が少なく、裾野を広げる活動が必要という意見が得られた。例として、データ連携基盤がないと実現しない事業モデル（又は、データ連携基盤があることでのみ実現効果が得られないこと）を示すのではなく、様々な選択肢を示し高い自由度の中で実現できるような事業モデルを示すことで、データ連携基盤が担う機能や役割もミニマムからスタートでき、利活用が促進されるのではないかという指摘が寄せられた。

(ｵ)基盤サービスの改善（長期運用/保守）（利用）

データ連携基盤を長期運用するにあたり、データ連携基盤と先端的サービスの連携において、共通的な機能をデータ連携基盤に実装することでコスト削減が可能となると考えられる。本項目では、利用者の視点で基盤に必要と想定される共通サービスのヒアリングを行った。詳細を以下に示す。

1)機能

回答者から選択された共通サービスは、「開発者ポータルサイト」、「双方向コミュニケーションポータルサイト」、「コンテンツ管理」、「地域ポイント管理」であった。コスト削減の観点においては、(ｳ)基盤サービスレベルの検討においても重要課題として回答者から選択されており、これらの共通サービスを実装しコスト削減につなげることで、より良いデータ連携基盤サービスを提供できると考えられる。

(ｶ)利活用効率化

データ連携基盤を利用し続けてもらうためには、利用者から見て利用しやすい環境（「開発しやすい環境)）を整備することが必要である。本項目では、利用者の視点で、データ連携基盤の利活

用を効率化させるために必要と想定される取り組みについてヒアリングを行った。詳細を以下に示す。

1)ルール、組織・ビジネス

データ連携基盤の利活用を効率化させるために、「ルールの整備」が必要であるという意見が 2 票寄せられた。また、ルールを整備するだけではなく、「ルールや規制への対応支援窓口等の整備」といった都市マネジメントに関する回答数も 2 票得られた。データ連携基盤の利活用を促進するためにはルールの整備が必須であり、かつ、当該ルールが利用者にとって身近なものになるよう、支援窓口等を通した配慮が必要であると考えられる。

(d)その他

図 9-1-2-4 に、仮説課題以外で、ヒアリングを通して自治体・団体から新たに得られた課題を示す。詳細は別紙オ-1「その他（ヒアリングにて得られた課題)」を参照のこと。

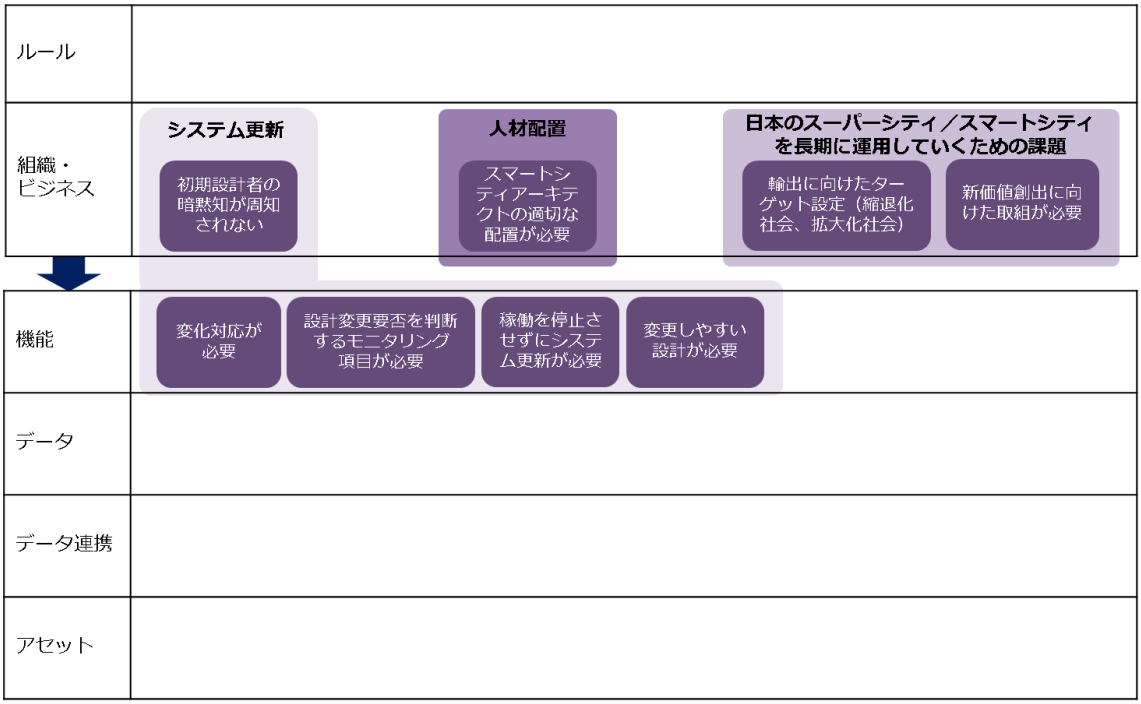


図 9-1-2-4　ヒアリングを通して自治体・団体から新たに得られた課題

(ア)システム更新

データ連携基盤を長期に運用するにあたっては、連携データや提供サービスの拡張により、度重なるシステム更新が発生することが予見される。データ連携基盤の稼働を停止せず、かつ煩雑になることなくシステム更新が行われるよう、以下の課題への配慮が必要であるとの意見が得られた。

1)組織・ビジネス

・「初期設計者の暗黙知が周知されない」:

データ連携基盤の長期運用を見据えた上では、初期設計者がいなくなってもデータ連携基盤を変化に対応させる必要がある。 データ連携基盤はプロダクトではなく後々変更するのは難

しいため、初期設計者がいる段階で知見を活用し、設計図の段階など早い段階から暗黙知を明文化していくべきであるとの意見が得られた。

2)機能

・「変化対応が必要」、「変更しやすい設計が必要」:

データ連携基盤を長期的に運用するには、テクノロジーや法制度の改正、人口移動などにいる地域住民の変化など、様々な変化に対応する必要がある。そのため、データ連携基盤を長期に運用していくために、データ連携のための通信プロトコルやコンピュータ性能など今後変化する可能性のあるものに対し、何が変化しやすいのかを予見し、設計に加味していくべきであるとの意見が得られた。

・「設計変更要否を判断するモニタリング項目が必要」:

データ連携基盤のようなオープンシステムを長期的に運用してくためには、周囲環境の変化や外部要求に伴うシステム設計変更に柔軟かつ簡単に対応できることが望まれる。そのために、設計段階から「何が変わったら、どの設計を関る必要があるか」を予測するとともに、適切なモニタリング項目を設け、運用において当該項目の意識的な確認を行う必要があるとの意見が得られた。(例:人口 1.000 人まで対応できるシステムを設計した場合、運用中に人口を意識的にモニタリングし、人口が 1,000 人に近づいた場合に修正対応を検討する、等)

・「稼働を停止させずにシステム更新が必要」:

データ連携基盤の特性上、データ連携基盤の稼働を止めずにシステム修正を行う必要がある。設計段階で考慮する必要があるとの指摘が得られた。

(ｲ)人材配置

1)組織・ビジネス

・「スマートシティアーキテクトの適切な配置が必要」:

スマートシティを長期に活性化させるためには、スマートシティ全体のメカニズムを担当する「スマートシティアーキテクト」の存在が重要だが、全部を 1 人が担当するのは難しいため、役割分担をする必要があるとの意見が寄せられた。また、1 つのプロジェクトにアーキテクトを多く配置しすぎるのも問題であり、スマートシティ/スーパーシティの規模に応じた人数の配置と役割分担が求められると予見されるとのことであった。

(ｳ)日本のスーパーシティ/スマートシティを長期に運用していくための課題

我が国において、データ連携基盤を含むスーパーシティ/スマートシティを長期に運用していくためには、スーパーシティ/スマートシティ事業の海外への輸出や、継続的な新価値創出が必要になり、それらに対する課題として下記の意見が得られた。

1)組織・ビジネス

・「輸出に向けたターゲット設定(縮退化社会、拡大化社会)」:

スーパーシティ/スマートシティの海外への輸出を行う場合、日本の取り組みは「縮退化していく日本社会」をターゲットとしているのに対し、輸出先の国々は「拡大していく社会」をターゲットとしているケースがあると想定され、日本の取り組みではそれらの国々に対応できない可能性がある。 今後拡大していく国々に対して、日本のスーパーシティ/スマートシティ

はどのようにアプローチすべきなのか考える必要がある。自治体単位で考えるのは難易度が高く、国として対応を考える必要があるとの意見が得られた。

一方で、今後拡大していく国々も、いずれは縮退に向かっていく。日本としては、拡大期を経て縮退期に向かっているということを見据えてスーパーシティ/スマートシティに取り組む必要があるとの指摘も上がった。

・「新価値創出に向けた取り組みが必要」:

スーパーシティ/スマートシティを長期に活性化させるためには、様々なプレイヤーが参画して多様な新価値を創造できる仕組みを構築することが必要となる。多様な目的に合わせた様々なサービスを提供しやすい環境や、様々なプレイヤーがデータを使いやすくする仕組みなどの体制の構築が求められるとの課題が得られた。

(2)ヒアリング結果に基づくデータ連携基盤の長期運用課題

0(1)を踏まえて、ヒアリングにより得られた自治体・団体が抱えているデータ連携基盤の長期運用に関する課題を、表 9-1-2-2 のとおり整理した。

表 9-1-2-2　ヒアリングによるデータ連携基盤の長期運用課題

課題	課題保持者	内容
費用対効果	• データ連携基盤推進主体	データ連携基盤の導入効果が不明瞭であり、数値として定量的に測りづらく、予算を獲得するための投資対効果の説明が難しい。
	• データ連携基盤推進主体	データ連携基盤の導入効果が不明瞭なため、データ連携基盤に関する多様なステークホルダの理解が得られにくく、データ連携基盤を利用したサービスや利用者が増えない。
ルール	• データ連携基盤推進主体 • データ連携基盤整備事業者 • データ提供者 • サービス提供者/ • データ利用者	データ連携基盤で扱われるデータに関するルールが明確でない。
長期運用を見据えた設計	• データ連携基盤整備事業者	データ追加やサービス追加に伴うデータ連携基盤の拡張作業や、保守・運用作業が煩雑であること。
データ整備	• データ連携基盤推進主体 • データ連携基盤整備事業者 • データ提供者 • サービス提供者/データ利用者	データの最新化にかかる作業負荷が高いこと。
	• データ連携基盤推進主体 • データ連携基盤整備事業者	規約やルールに沿わないデータの利活用を、データ連携基盤側で制限する手段がないこと。
	• データ提供者 • サービス提供者/データ利用者	データアクセシビリティが低く、有益なデータがどれかわかりづらいこと。

課題	課題保持者	内容
人材育成	• データ連携基盤推進主体	データ連携基盤の技術的な知識が浸透しておらず。十分な利活用が推進できないほか、自治体内でもその重要性に理解を得られにくいこと。
評価改善	• データ連携基盤推進主体 • データ連携基盤整備事業者	データ連携基盤が長期に利活用されるために、住民を含む利用者目線のサービス設計を行い、利用者のニーズを踏まえる必要がある。

9-1-3. 課題を解決するための仕組み

表 9-1-2-1 に示した、ヒアリングによるデータ連携基盤の長期運用課題を解決するための仕組みを以下に示す。詳細は別紙オ-1「課題解決策」を参照のこと。

(1)費用対効果

1 つ目の課題として、データ連携基盤の導入効果が不明瞭であり、数値として定量的に測りづらく、予算を獲得するための投資対効果の説明が難しいという点が挙げられる。この点に対する解決策として以下が考えられる。

(a)導入効果指標を設けること

サービス間や分野間のデータ連携により生まれる効果を表現する、客観的な指標を設けること。指標の設定例としては、リファレンスアーキテクチャ 3.2 「⑤ ：KGI 及び KPI の設定」をご参照いただきたい。また、指標項目と併せて、データ連携基盤を導入する自治体間で相互比較ができるよう、人口比換算や投資費換算等を推奨するなどの基準が設けられるとよい。また、データ連携基盤の導入と利活用に成功している先行事例での各指標値を、ユースケースと併せて可能な範囲で公開・共有することで、他の自治体も導入効果を説明しやすくなり、データ連携基盤の導入や運用に対する予算を確保しやすくなると考えられる。

(b)導入成功事例/失敗事例を共有すること（事例共有カタログの整備）

データ連携基盤を活用した成功事例を、取り組みの背景や目的などと併せてストーリーとして展開すること。成功事例を活用することで、自らの自治体で目指すデータ連携基盤を活用した取り組みの妥当性が説明しやすくなるほか、(a)で示す導入効果指標と併せることで投資対効果の説明がしやすくなると見込まれる。また、単なる事例の横展開ではなく、より自治体にフィットしたデータ連携基盤の利活用施策の検討につながり、結果としてより良いサービス提供に結びつくと考えられる。成功事例だけでなく、躓いた部分などを明らかにした失敗事例も展開するとよい。失敗事例は公表しにくい部分もあるため、展開の仕方として、クローズド環境で情報共有できるような工夫があるとよい。（自治体間のコミュニティの場を設けるなど。）

2 つ目の課題として、データ連携基盤の導入効果が不明瞭なため、データ連携基盤に関する多様なステークホルダの理解が得られにくく、データ連携基盤を利用したサービスや利用者が増えないという点が挙げられる。この点に対する解決策として以下が考えられる。

(c)丁寧な説明、双方向のコミュニケーションを行うこと

データ連携基盤の導入及び運用に対して多様なステークホルダの理解を得るには、それぞれの立場や理解度に合わせた説明が必要となる。国からの一方的な視点でのみ語るのではなく、双方向のコミ

ュニケーションを通じた認識の擦り合わせ、そして各ステークホルダの理解に繋がるまでの筋道を明確に示す必要がある。そのためには、データ連携基盤や関連する IT 技術についてわかりやすく説明し自治体や住民等の理解を推進できる人材や、ワークショップの開催、教材の開発などが必要となる。

(d)データ連携基盤の目的を共有すること

データ連携基盤を整備する意義を改めて議論し、データ連携基盤の目的を共有する必要がある。データ連携基盤の意義を広く周知し共有する手段として、(c)に上げた多様なステークホルダ間への丁寧な説明や双方向のコミュニケーションを行うことや、各ステークホルダにとっての導入効果やメリットの明確化、背景や目的も含めた成功事例/失敗事例の展開・共有を行うことなどが重要となる。

(2)ルール

(a)データについてのルールメイキングを行うこと

データ連携基盤で扱われるデータに関する個人情報管理、責任範囲、品質、信頼性、セキュリティ、データフォーマット等に関するルールを明らかにすること。ルールの制定にあたっては国が主導して大方針を定め、自治体は個別にローカライズすることができるなど、柔軟な運用形態であることが望ましい。また、個人情報の匿名化については、国が具体的な方法を示すことが望ましい。

(b)データフォーマットを決定すること

データフォーマットを示す際は、各分野のスマートシティ/スーパーシティにおける活用事例や使い方とともに展開すること。これによりデータフォーマットの利活用促進につながる。

データフォーマットの決定においては、日付データと場所データの順番など、詳細部分まで国が主導して決定できるとよい。これにより都市間での連携がしやすくなる。さらに、自治体の既存業務にできるだけ即した形になることが望ましい。

(3)長期運用を見据えた設計

(a)将来変化に柔軟に対応できる設計を行うこと

データ連携基盤を長期に運用するために、初期設計の段階から、変化に対応できるシステム設計を行うこと。機能面においてはビルディングブロック方式の利点を最大限活用し、機能単位で拡張/縮退可能な構成とするほか、通信プロトコルやコンピュータ性能、連携アセットの拡張や保守、ネットワーク負荷分散など、今後変化する可能性のあるものに対し、初期設計の時点である程度予見しておく必要がある。何が変化しやすいのかを考え、設計に加味していくことが必要となる。

また、初期設計担当者のノウハウを明文化し、後任の設計担当者へ確実に引き継いでいくことが望ましい。

なお、本課題及び解決策は、9-2 陳腐化しない運用の仕組みにて検討された 9-2-3 時代の変化に応じて柔軟にシステムを拡張する仕組みにおいても詳細な検討を実施している。そのため、9-2-3 も併せて参照頂きたい。

(4)データ整備

1 つ目の課題として、データの最新化にかかる作業負荷が高いことが挙げられる。これに対する解決策として、以下が考えられる。

(a)作業を簡易化するツールを提供すること

データの最新化にかかる作業負荷を軽減するため、データ連携基盤上に簡易にデータを取込むツー

ルを提供すること。例として、データ連携基盤にデータを連携するための準備作業（メタデータ作成）の支援環境があるとよい。

2つ目の課題は、規約やルールに沿わないデータの利活用を、データ連携基盤側で制限する手段がないことである。これに対する解決策として、以下が考えられる。

(b)データ提供者/利用者の認証機能、アクセス権機能、セキュリティ対策機能等を追加すること

データ連携基盤を導入する段階で、ルールを明確に定め、また、規約やルールに沿わないデータの利用を確認できる仕組みを構築すること。例として、データごとにデータ提供者/利用者の認証機能を設けたり、データへのアクセス権を付与したり、セキュリティ対策を講じたりすることが望まれる。

3つ目の課題は、データアクセシビリティが低く、有益なデータがどれかわかりづらいことである。解決策としては、以下が考えられる。

(c)有益なデータを見つけやすくする機能を設けること

データ利用者が適切なデータを探しやすくする機能を追加すること。例として、データの人気ランキングや利用者数の表示。データカタログ機能、データ提供者の評価結果がわかる機能[125]等があるとよい。

(5)人材育成

(a)適切なアーキテクチャ人材の育成、派遣を支援すること

各ステークホルダの立場や理解度に合わせた合意形成と全体アーキテクチャの最適化を推進可能な人材を派遣・育成すること。国と自治体を繋ぐ役割として、技術的な難易度の高い表現をかみ砕き説明できる能力や、技術だけでなく行政や企業活動も理解していることが望まれる。ただし、スーパーシティ/スマートシティのアーキテクチャ人材は対応すべき範囲が非常に広いため、複数人で担当するなど、スーパーシティ/スマートシティの規模に合わせた配置や役割分担の検討が必要である。

(6)評価改善

(a)住民を含む目線のデータ連携基盤評価指標を設け、PDCAサイクルを回すこと

データ連携基盤を継続的に運用するために、住民を含む利用者からのフィードバックを得る仕組みを設けること。例として、ユーザ数やアクセス数、ダウンロード数等による利用効果の計量、住民及びデータ連携基盤利用者を対象とした満足度調査、費用対効果の測定、利便性向上調査、データ欠損検知や正規化の測定、APIに関する継続的な監視・エラー検出があるとよい。これらの指標を継続的にモニタし、PDCAサイクルによる段階的かつ定期的なサービス改善作業を行うこと。

なお、本課題及び解決策は、9-2陳腐化しない運用の仕組みにて検討された9-2-2(2)利活用向上に向けた検討においてさらに詳細な検討を実施している。そのため、9-2-2(2)についても併せて参照頂きたい。

[125] 情報提供　内閣府データ連携基盤サブワーキンググループ（第3回）資料1　分野間データ連携基盤の整備に向けた方針案（内閣府）　データ連携基盤サブワーキンググループ（第3回）（cao.go.jp）

9-2. 陳腐化しない運用の仕組み

9-2-1.　調査方針と調査対象

データ連携基盤・API カタログ・開発者ポータルサイトの継続的な運用の実現に向け、既存のデータ連携基盤・API カタログ・開発者ポータルサイトの事例を調査し、想定される検討事項を整理する。

(1)調査方針

整理にあたりデータ連携基盤・API カタログ・開発者ポータルサイトが陳腐化せず、継続的に運用できている理想的な状態とはどのような状態であるか定義する。継続的な運用を行う上で必要な観点を「認知度向上・普及促進」「利活用の向上」「運営」の 3 つに分解し、それぞれについて継続的に運用できている状態を定義し、検討事項を整理する。

表 9-2-1-1　継続的に運用できている状態

運用の観点	継続的に運用できている状態
陳腐化しないための認知度向上・普及促進	データ連携基盤・API カタログ・開発者ポータルを認知してもらうための取り組みや、その利用者（サービス提供事業者、データ提供者、住民）を増やすための取り組みが行われている状態
陳腐化しないための利活用の向上	技術動向や利用者のニーズを取り込んだコンテンツの創出や改良が実施できている状態
陳腐化しない運営	運営主体と体制が定義され、運営計画が立案されている状態。また、立案した運営計画を実施する役割が明確である状態。

(2)調査対象

本調査では、データ連携基盤・API カタログ・開発者ポータルサイトを運用する上で必要な役割を、「データ連携基盤推進者」、「データ連携基盤構築事業者」、「データ連携基盤利用者」の 3 種類に分類した。調査対象及び選定理由を表 9-2-1-2 に示す。

表 9-2-1-2　調査対象と選定理由

分類	調査対象	選定理由
データ連携基盤推進主体者	エストニア	国が主導する 10 年以上の実績のあるデータ連携基盤を運用しており、スーパーシティのデータ連携基盤においても参考となるノウハウを保有していると想定される。
データ連携基盤構築事業者	農業データ連携基盤（WAGRI）	農業に関係する様々な公益団体や民間事業者が、API を介して農業分野のデータを流通するためのデータ連携基盤や開発者ポータルサイトの運用を行っており、スーパーシティのデータ連携基盤においても参考となるノウハウを保有していると想定される。
	G 空間情報センター	地理空間に関するデータを流通するためのデータ連携基盤の運用を行うとともに様々な活用事例を提供しており、スーパーシティのデータ連携基盤においても参考となるノウハウを保有していると想定される。
データ連携基盤利	Code for Japan	主に地域住民の目線から地域の課題を解決するための有志に

分類	調査対象	選定理由
用者		よる取り組みや自治体への提言を実施している。住民目線におけるスーパーシティにおける利活用向上に参考となるノウハウを保有していると想定される。

なお「データ連携基盤利用者」はサービスやデータの提供者及び利用者であり、自身がデータ連携基盤・API カタログ・開発者ポータルサイトを運用しないことから、「認知度向上・普及促進」と「運営」の調査の対象外とする。分類と調査する運用の観点の関連性は表 9-2-1-3 のとおり。

表 9-2-1-3　分類と調査する運用観点の関連性

分類	認知度向上・普及促進	利活用の向上	運営
データ連携基盤推進主体者	調査対象	調査対象	調査対象
データ連携基盤構築事業者	調査対象	調査対象	調査対象
データ連携基盤利用者	調査の対象外	調査対象	調査の対象外

9-2-2.　データ連携基盤が公開する API やデータの鮮度を維持する仕組み

(1)認知度向上・普及促進に向けた検討

継続的な運用を実現するためには、まずデータ連携基盤や API カタログ・開発者ポータルサイトの存在を周知し、利用者となり得る方々が認知し関心を持つことが重要である。関心を持った利用者は、インターネットなどを用いてデータ連携基盤や API カタログ・開発者ポータルサイトに関する詳細な内容、活用事例、先行利用者による評価などを検索することが想定される。これらの検索結果を踏まえて、データ連携基盤利用者にとってメリットを感じられたのち、行動を起こすための場として、ワークショップやコミュニティ活動などの情報共有や支援の場が提供されることにより普及が進んでいくと想定される。

調査対象が実施している認知度の向上と普及を促進させるための取り組みとして、利用者が認知し、関心を持ち、行動を起こすためにどのような施策を実施しているのか調査をした。調査結果を行動分析フレームワーク「AIDMA」の行動プロセス（図 9-2-2-1）に合わせて整理を行う。整理した結果を踏まえて、認知度の向上と普及を促進させる施策の実施にあたり発生し得る課題の抽出を行った。

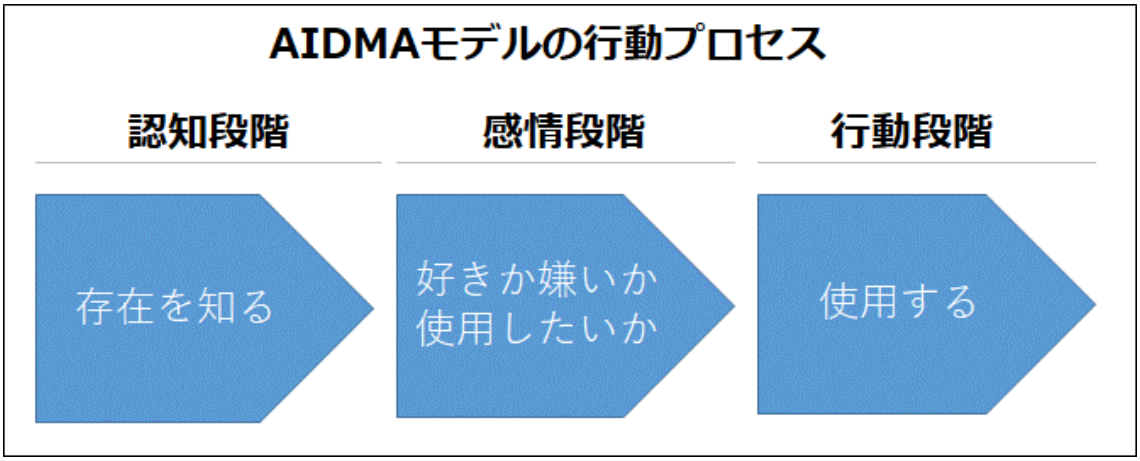


図 9-2-2-1　AIDMA モデルの行動プロセス

表 9-2-2-1　調査結果

No.	団体名	施策		
		認知のための施策	感情に訴えるための施策	行動を起こすための施策
1	エストニア	• ホームページ開設 • 政府によるトップダウンで取り組み、電子政府サービス「e-Estonia」を支える基盤を OSS として公開 • X-Road Community Event、Ultrahack、API Conference などのイベントの開催や参加	• データ連携基盤が導入されることで、ID カード 1 枚で行政や民間のサービスを受ける利便性を周知する • エストニア国民以外の外国人に対しても ID カードを発行し普及を促進する	• ID カードの交付 • スマートフォン上のデジタル ID アプリの提供 • Nordic Institute for Interoperability Solutions による各種開発者向けの資料や学習教材を提供
2	農業データ連携基盤協議会(WAGRI)	• ホームページ開設 • ニュース媒体を活用した取り組みの紹介やキーマンによる対談記事の紹介 • シンポジウムや展示会などによる講演の実施 • 所轄官庁である農林水産省のホームページでの取り組みの紹介 • YouTube 公式チャンネルによる情報発信	• ドキュメントや動画による活用事例の公開 • 導入メリットに関する情報提供 • 安心してデータ連携基盤を使って頂くため、農林水産省や弁護士等による「農業分野における AI・データに関する契約ガイドライン」の策定	• 技術者に向けた API 仕様やサンプルプログラムコードの提供 • 生産者に有益となるデータやサービスの発掘と登録の推進 •
3	G 空間情報センター	• ホームページ開設 • シンポジウムや展示会などによる講演の実施 • エキスポや展示会への出展 • 所轄官庁である国土交通省のホームページでの取り組みの紹介 • YouTube 公式チャンネルによる情報発信	• ドキュメントや動画による活用事例の公開 • ニュースレターの公開 • ユーザ会の活動資料の公開	• 技術者に向けた API 仕様の提供 • API を利用するための実習用の教材の提供 • データ利活用に関するコンサルティングなどによる地域活性化の支援

調査の結果、いずれの調査対象も利用者となり得る方々に対し、認知から行動につながるまでの様々な施策を実施していることが判明した。これら施策を実施する上で以下に示すような課題があると想定される。

● 利用者となり得る方々に対するリーチ

認知度を向上し普及を促進する対象が特定の分野であれば、該当する分野の業界団体などの活動を介して認知して頂くことは容易と想定される。しかしながら、スーパーシティのようにこれまでの業種・業態の枠組みを横断して様々な利用者に対して認知して頂くためには、どこへリーチすれば良いのか、またどのような内容を伝えれば誤解のないように認知いただけるのか検討する必要がある。

スーパーシティの場合において、認知して頂くために例えば以下に示すような施策案が考えられる。

・行政機関のホームページや施設の掲示板・ポスター展示による取り組み紹介、事業説明会の実施
・他の団体が運営・開催するホームページやセミナーでの取り組みの紹介
例）地方公共団体情報システム機構主催 地方自治情報化推進セミナー、独立行政法人情報処理推進機構(IPA)が運営する地方版 IoT 推進ラボ
・行政向けの機関紙誌への記事の掲載

● 施策における効果測定

施策を実施した結果としてどのような効果があったのか整理することは重要と考えられる。ただ、その場合の効果として、利用者が増えれば良いのか、データ連携基盤を流れるデータ量や API の利用頻度が増加すれば良いのかなど目的とする効果を明確にし、その効果を測定した結果に基づいて施策の見直しを行う必要がある。ただし、様々な施策を実施すると個々の効果が見えにくくなり、効果の出にくい施策を実施し続けることになりかねない。そのため施策に対する目的とする効果とその効果を測定する方法を定める必要があると想定される。

施策の効果を測定する方法としては以下のような方法が考えられる。
・ホームページへのアクセス数（総数、ユニークユーザ数）の推移や資料のダウンロード件数推移
・Google などの検索サービスが提供する検索キーワードのトレンド
・データ連携基盤利用者の登録者数と推移
・データ連携基盤の API 利用頻度とその推移
・データ連携基盤を利用したサービス数の推移

(2)利活用向上に向けた検討

認知度が向上し普及が進んだ後、データ連携基盤・API カタログ・開発者ポータルサイトに実装される技術や機能及びコンテンツが古いままでは、いつか利用者が減り利活用が進まなくなるおそれがある。利活用を向上させるためには、住民目線での更新や改良が必要であると考える。調査対象が実施している住民のニーズを取り込む仕組みや、既存の機能やコンテンツの更新や改良を促す仕組みを、表 9-2-2-2 に示す。

表 9-2-2-2 調査結果

No.	調査対象	施策
1	エストニア	• X-Road コミュニティによる X-Road Community Event や Ultrahack などのイベントを通じた活用事例の紹介や利活用を向上させるためのサービス紹介、slack によるユーザ（主にサービス開発事業者）のノウハウの共有 • 今後の開発ロードマップを公開 • X-Road サービスデスクを通した追加機能に関する要望を収集
2	農業データ連携基盤(WAGRI)	• 登録済み API の利用者レビュー機能を実装 • 農業活動に資するデータや API を調査し、有益なデータや API があれば WAGRI への登録を勧める活動 • 生産者からのデータ利活用に関する問合せがあれば、利用方法の説明やデータの充実を図る支援活動を行う • 利用が促進するよう、データ連携基盤利用規則やデータ提供利用規約を随時見直す活動
3	G 空間情報センター	• 提供するデータセット数、ファイル数、登録組織数を HP で常時公開 • 人気のデータセットをホームページ上にランキング形式で公開

No.	調査対象	施策
		• データセットに対するユーザからのフィードバックを受付ける機能を準備 • ショーケース一覧（活用事例の一覧）をホームページ上で公開 • G 空間 EXPO（例：地理空間情報フォーラム）での活用事例の紹介
4	Code for Japan	• データ連携基盤を構築するエリアからの依頼を受け、住民目線から改善・改良に繋がる意見を提示する • コミュニティを作り、現状に対する意見や解決につながるアイデアを提示するなど、データ連携基盤の運用者に対してフィードバックを行い、改善・改良につなげる活動 • 「参加型合意形成プラットフォーム」を導入し、住民の意見を集め、議論を集約し、政策に結びつけていく活動 • 定期的にサミット（カンファレンス）やワークショップを開催し、事例の共有や課題解決につなげる活動

調査の結果、いずれの調査対象も機能やコンテンツが継続的に利活用されるよう、利用者からのフィードバックを収集する施策や、コミュニティやイベントを開催し利活用の事例紹介、今後のエンハンス計画の公表を行っている。データ連携基盤の運用者と、サービスやデータ提供者との間で実施される利活用に関する活動内容は表 9-2-2-2 のとおりである。各団体で共通的に実施している「広報活動の施策」「フィードバックを収集する施策」について、施策内容を以下に整理する。

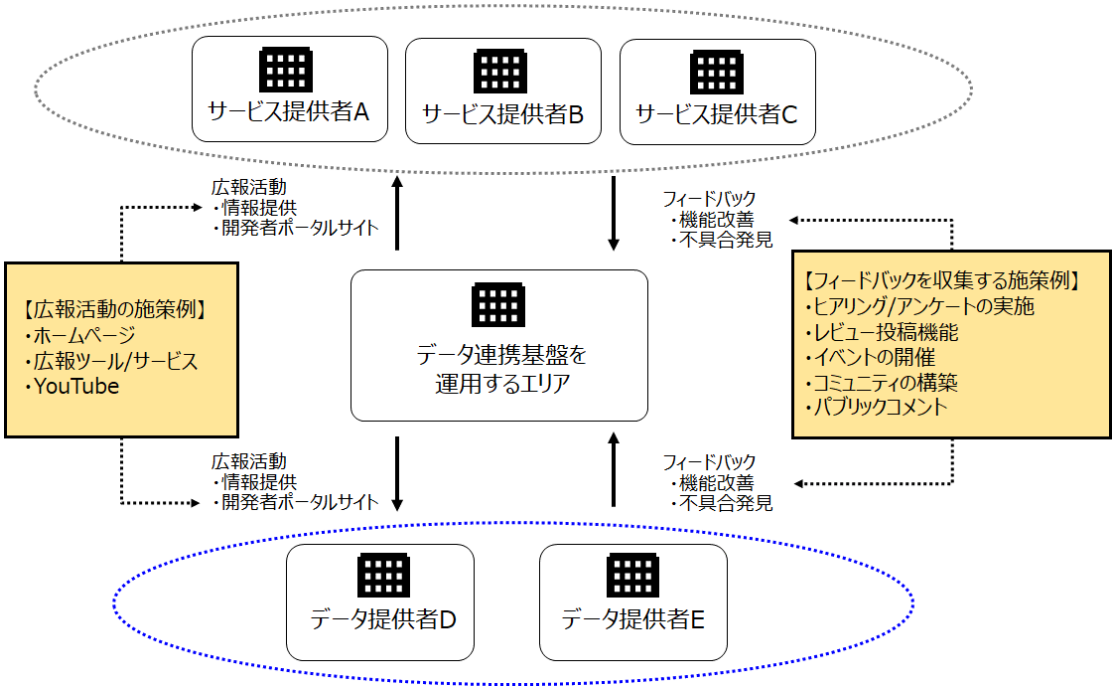


図 9-2-2-2　利活用に関する活動内容

(a)広報活動の施策

データ連携基盤・API カタログ・開発者ポータルサイトに実装される技術や機能の利活用を促すためには、サービス提供者・データ提供者に対して多様なアプローチを用いた広報活動を行い、利活用を促進させることが有効な施策である。

広報活動に関する施策例を表 9-2-2-3 に示す。

表 9-2-2-3　広報活動の施策例

No.	広報手段	施策例
1	ホームページ	• データ連携基盤のホームページ上において、活用事例の紹介を行う • データ連携基盤の改良に資する技術情報の提供依頼（情報提供依頼書(RFI)の公開）
2	広報ツール・サービスの活用	• 各種 SNS 公式アカウントで、活用事例の紹介などを行う • 技術情報に特化した情報交換を SNS 形式で行える Qiita を活用し、開発事業者間の新技術情報の共有や、記事の相互編集機能を用いた技術的課題の解決を行う • プレスリリース配信サービスである PRTimes を利用して、データ連携基盤に関するニュースリリースの発行や機能追加・改良に関する情報発信を行う
3	YouTube	• サービスやデータの利用方法、活用のメリットを動画にして YouTube 上で配信する • EXPO やフォーラムでの発表内容や展示内容を動画で配信する • サービスやデータの利活用方法や事例を映像で紹介することで、最新のサービス提供内容やデータ提供状況をよりわかりやすく伝える

(b)フィードバックを発信させ、収集する施策

データ連携基盤・API カタログ・開発者ポータルサイトの利活用を促すためには、利用者目線での機能の追加や改良が必要である。

適切な機能追加や改良を行うためには、利用者からのフィードバックを発信して頂く機会・手段を用意し、これを収集する施策が有用である。実装された機能に対する改善や追加機能の要望といったニーズを保有しているのは利用者である。

また、利用者から不具合情報を発信させ、タイムリーに受付ける仕組みを準備することで、実装された技術や機能の改良と不具合への対応速度を速めることが可能となる。

フィードバックに関する施策例を表 9-2-2-4 に示す。

表 9-2-2-4　フィードバックを収集する施策例

No.	フィードバック方法	施策例
1	アンケートやヒアリングの実施	• 現存する機能に対する利用者・開発者向けアンケートの実施 • サービス提供者となる団体に対してヒアリングシートを配布し、収集した意見から改良・改善の要素を調査する。
2	レビュー投稿の受付	• 利用者からの機能レビュー、コメント投稿を受付けることで、機能改善の要望や不具合情報を発信いただき、これを随時収集する • 投稿を受付ける機能をデータ連携基盤上の特定ページに集約することで、データ連携基盤の運用者と該当機能のサービス提供者両者に対して周知し、改善を促す
3	イベントの開催	• 住民も参加可能な展示会を開催し、サービスを試用した結果に対する意見や感想を収集する • エリア横断型の交流会を開催し、開発事例の共有を行うことで意見交換を通してサービスの改善を促す
4	コミュニティの構築	• 「地域住民参加型合意形成プラットフォーム」を活用し、住民からの要望や

No.	フィードバック方法	施策例
		意見を聞き取り、住民同士の議論を活発化させ、要望実現に向けた合意形成を図る • Slack や Qiita を活用した開発者同士のコミュニティを構築し、リアルタイムコミュニケーションが円滑に行われる状態を維持することで、発見した不具合や課題に対する機能改善の提案を収集する • GitHub の課題管理機能を活用して、不具合情報や、機能要件と実装された機能の間での齟齬情報を収集・共有する
5	パブリックコメントの活用	• 行政手続法に基づき意見公募の手続を行い、現行機能や新たに構築しようとする機能に関する意見を住民から幅広く収集する。 • 収集した意見を分析し、採用する案・採用されない案とその理由を再度公示することで住民の理解を得る。

(3)継続的な運営に関する検討

データ連携基盤等の利活用を推進したとしても、データ連携基盤・API カタログ・開発者ポータルサイトを継続的に運営するための計画や体制が整っていないと、利用者からの信用を失い利用者が減って利活用が進まなくなるおそれがある。調査対象において運営を担う主体者を調査し、その運営主体者がどのような運営をおこなっているのか、またデータ連携基盤などを利用するサービサーがどのような形で関りを持つのか整理した。調査結果を踏まえて、継続的に運営するために必要になると考えられる取り組み内容を記載する。

表 9-2-2-5　調査結果

No.	調査対象	運営の主体者	運営方法
1	エストニア (X-Road)	エストニア共和国情報システム局 RiigiInfosüsteemiAmet (RIA)	エストニアの X-Road は、エストニア共和国情報システム局(RIA)が運営を行っている。X-Road の機能拡張に向けた検討や開発は、2017 年 8 月にフィンランド政府とエストニア政府が X-Road の共同開発のために合同で設立した NPO 団体である、Nordic Institute for Interoperability Solutions (NIIS)が実施している。NIIS は開発だけではなく、データ提供者やサービサーなどからの要望の受付や教育などの支援を行っている。運営する組織と開発をする組織を明確に分けているが、お互いの組織が相互に連携した形での運営を実施している。
2	農業データ連携基盤(WAGRI)	国立研究開発法人 農業・食品産業技術総合研究機構（農研機構）農業情報研究センター	WAGRI は、農研機構 農業情報研究センターが運営を実施。API そのものについては WAGRI のサイトを通して、公益団体や民間事業者が提供及び運営を行っている。 データ連携基盤の運営に加えて、データ提供者やサービスを提供するサービサーから頂く多種多様な要望（機能・サービスの追加、データ種類の追加、性能の改善、利活用の仕方など）を踏まえた基盤の改善や各種支援も実施している。これらの支援を通じて、最終的なサービス受益者である農業生産者がメリットを享受できるような持続可能な基盤運営を目指した取り組みを実施している。
3	G 空間情報センター	一般社団法人	G 空間情報センターは、データカタログサイトとデータ提供のための

No.	調査対象	運営の主体者	運営方法
		社会基盤情報流通推進協議会	データ連携基盤を一般社団法人 社会基盤情報流通推進協議会が運営を実施。データについては、公益団体や民間事業者がG空間情報センターの基盤上を使いデータ提供を行っている。 基盤の運営に加えて、利用者の視点にたった「社会課題解決パッケージ」の提供、災害時の緊急対応を支援する情報提供、ソフトウェアの提供や人材の支援など、基盤の活用を促すためのサービスも提供している。

調査の結果、いずれの調査対象も運営体制が明確であり、計画的な活動を行っている。また、基盤の運営を行っているだけでは無く、基盤を継続的に運営するため、基盤の利用者の要望を収集する仕組みやサービス利活用を促すための人材支援などを実施していることが判明した。スーパーシティの要件の 1 つに「供給者、技術者目線ではなく、住民目線でより良い暮らしの実現を図るものであること」とうたわれている。有識者に対してヒアリングを実施した結果、「住民が関わることは、発展していく上で重要な施策である。」との意見を頂いている。

基盤の運営を行う上で、利用者からの要望の取り込みを実施することは重要ではあるが、一方でこれらの施策を実施する上で以下に示すような課題があると想定される。

● システム的な運営課題や利用者要望などに対する対応方針やルールの明確化

人や物などのリソースは有限であるため、全てに対応することは困難であると想定される。リソースの制約がある状況で、さらには基盤や地域の特性などを考慮した上で、何を優先に対応するべきなのか方針やルールを明確にする必要があると想定される。

また、明確な運営方針やルールは定めることは大切であるが、運営組織の中に閉じたままの方針やルールでは、サービサーや住民などの利用者からは、運営実態が見えてこない。そのため、方針やルールだけではなく、可能な範囲で検討の経緯も含めて公開し、利用者も運営に関わっていることを意識付け、継続的に改善できるような仕組みを構築する必要がある。

● 運営の公平性について

データ連携基盤は地域の課題解決に資する基盤である。課題解決に向けて地域の住民や事業者等から意見を集め、ステークホルダと合意形成を作ることになる。都市の主役は住民であり、住民に寄り添った対応が必要となるが、住民には老若男女、外国人、障害者・健常者等、様々な立場があり、様々な暮らしや経済活動を行っている。課題解決に向けて地域の住民や事業者等から意見を集めることは有益な活動ではあるが、意見の収集や意思決定には、公平性や公益性の配慮が必要である。

この課題に対して、有識者ヒアリングにおいて「地域住民参加型合意形成プラットフォームの活用が有効である。」と意見を頂いた。このプラットフォームは、施策を実現するにあたり、様々な地域住民に議論に参加頂き、意見の収集・整理を行い、集めた意見に対して地域住民自ら議論を通して合意形成を行う枠組みである。このプラットフォームを利用した地域住民との合意形成は、公平性の観点から意思決定方法の有効な手段の 1 つと考えられる。

9-2-3.　時代の変化に応じて柔軟にシステムを拡張する仕組み

データ連携基盤を長期運用していくためには、利用者の意見の基に改善・改良を重ねていく、いわゆるエコシステム環境の構築が重要となるが、システムの拡張性において、従来の個別特化したシステムでは、機能拡張によるコストや労力が大きくなり、継続的かつ容易にサービスを進化できないという課題がある。

本課題解決にあたっては、利用する機能やアーキテクチャの更新にあわせ、機能拡張や更新を容易にする仕組みが必要であり、「ビルディングブロック方式」といった機能の組み換えを柔軟に対応できる仕掛けを持つ必要がある。

ビルディングブロック方式とは、機能群の中から必要な機能を取捨選択し積み重ねることで、疎結合なシステム構築を行う方式のことである。ビルディングブロック間の API を統一もしくは公開することで、他のビルディングブロックへの影響を局所化した追加、更新が可能であり、必要な機能をあらかじめ定義してブロック化することで、必要最小限の機能でスモールスタートし、地域が解決する課題や目指すべき将来像に応じて（実証→社会実装→エコシステム環境の構築といった）段階的に機能拡張していくことも可能となる。また、障害発生時においては、障害範囲を限定することができ、障害対応やセキュリティパッチ適用などシステムに手を加える際に、既存部分への影響を最小限に抑えられ、保守性の向上も可能となる。ビルディングブロック方式による機能拡張例を図 9-2-3-1 に示す。

なお、ビルディングブロック方式を採用する場合、ビルディングブロック単位で各機能を提供する主体が異なることによりシステムの運用管理が煩雑になるため、統合運用管理等の仕組みを取り入れることが望ましい。また、各機能の提供主体が特定ベンダに依存しないよう、オープンソースを使うことで透明性を持たせることも望まれる。

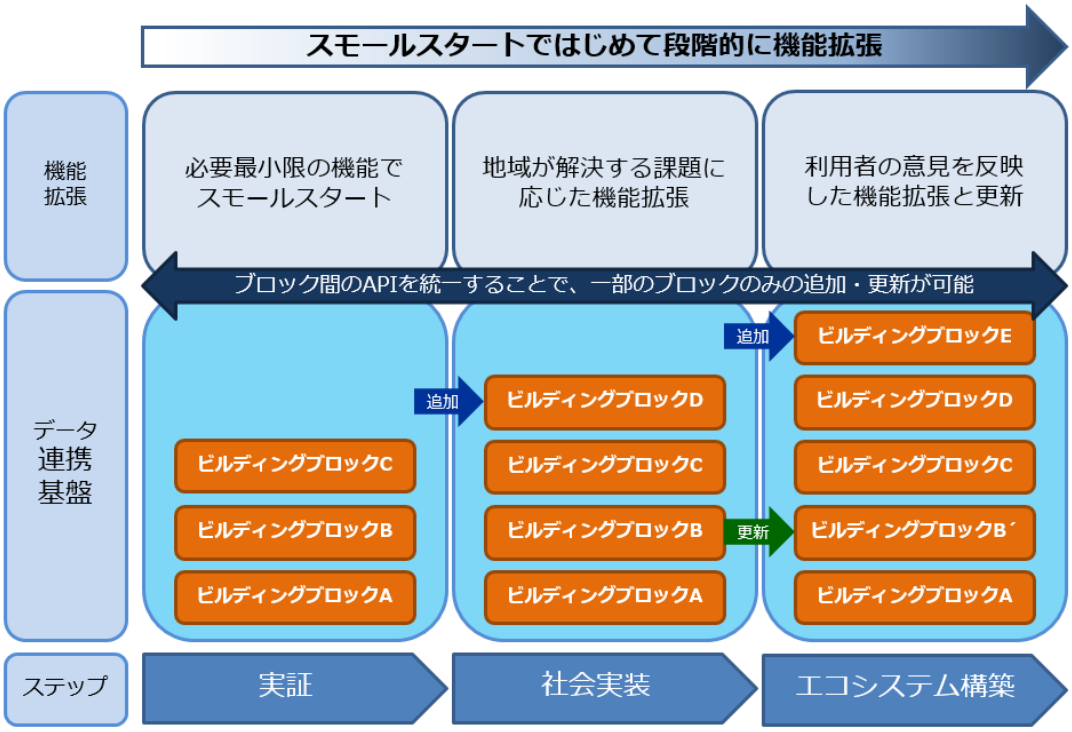


図 9-2-3-1　ビルディングブロック方式による機能拡張例

9-3. 安心してデータを扱える仕組み

住民が安心してサービス利用するためには、サービスが利用する自身のパーソナルデータについて把握できることが重要である。つまり、透明性を考慮したパーソナルデータ管理が実施されていることが重要である。透明性を考慮したパーソナルデータ管理としては、サービス利用開始時のサービス規約の明示とそれに対する同意管理（オプトイン）、サービス利用中にパーソナルデータがサービス利用規約に基づき扱われていることを確認する手段、パーソナルデータの開示・変更を請求するための手段、サービス利用終了時の解約手段（オプトアウト）などについて考慮する必要があると考えられる。

また、品質の高いサービスを継続して提供することも住民の安心につながる。そのためには、サービス品質を継続的に維持・改善する必要があり、データ品質管理（品質低下防止と継続的な品質改善）が重要となる。品質低下防止対策としては、サービスを提供するために必要十分なデータ項目を含むデータモデルであるか、収集されたデータ項目値が定められた精度となっているか、などについて考慮する必要がある。継続的な品質改善対策としては、定性的・定量的評価に基づくデータモデルの改善を継続的に行うことについて考慮する必要があると考えられる。

表 9-3-1　安心してデータを扱える仕組みに関する課題抽出、機能概要案の洗い出し手順

手順	手法
1	（ウ）データモデルの方針で検討した複数の先端的サービスのユースケースから、「ステークホルダの視点」、「タイミング」ごとの課題を抽出
2	抽出したユースケースごとの課題を整理、統合
3	統合された課題を解決するために必要な要件を、内閣府が定義したスマートシティリファレンスアーキテクチャの階層に対応付けて整理し、データ連携基盤で解決すべき課題を特定
4	特定された課題を解決するための機能概要案を検討

透明性を重視したパーソナルデータ管理と継続的なパーソナルデータ品質管理について、表 9-3-1 で述べた手順に従い別紙オ-3 に整理した。詳細は別紙参照ではあるが、本報告書では、ウ）データモデルの方針で検討した先端的サービスの例に記載された各ユースケースから、機能層、データ層、データ連携層、アセット層に関する課題に対し、発生タイミング、「サービス利用者」「先端的サービス事業者」の各ステークホルダ視点で表 9-3-1-1、表 9-3-2-1 に整理した。

なお、本報告書においては、システム層（機能層、データ層、データ連携層）についての機能概要案の検討を行うが、非システム層（戦略・政策層、ルール層、組織層、ビジネス層）で検討すべきことは検討範囲外とする。また、本報告書においては、（ウ）データモデルの方針で検討したユースケースを使用した検討のみを行っていため、更に他のユースケースを検討することにより課題抽出、機能概要案の洗い出しが必要であると考える。

9-3-1. 透明性を重視したパーソナルデータ管理の検討

透明性を重視したパーソナルデータ管理について、発生タイミング、「サービス利用者」「先端的サービス事業者」の各ステークホルダ視点での課題を表 9-3-1-1 に整理した。さらに、データ連携基盤を整備するために必要と考えられる機能概要案と関連する非機能要件に関して、IPA「非機能要求グレード」とのマッピングを行った。

表 9-3-1-1 透明性を重視したパーソナルデータ管理の課題、機能概要案、非機能要求グレード整理

項番	発生タイミング	ステークホルダ	課題	機能概要案	非機能要求グレード（注）
1	サービス利用申込時	サービス利用者	サービス利用者が、自身のパーソナルデータの利用に関する同意を先端的サービス事業者ごとに実施できること	先端的サービス事業者が、サービス利用者のオプトイン・オプトアウトの状態を把握できること	E5.アクセス・利用制限 E5.2.利用
				サービス利用者が、自身のパーソナルデータの利用に関する同意を先端的サービス事業者ごとに実施できること	E5.アクセス・利用制限 E5.2.利用制限
2	サービス利用中	サービス利用者	サービス利用者が、自身のパーソナルデータに対するアクセス記録を先端的サービス事業者ごとに、いつでも、簡単に確認できること	先端的サービス事業者が、パーソナルデータの利用履歴をサービスごとに取得できること	E7.不正追跡・監視 E7.1.不正監視
				サービス利用者が、自身のパーソナルデータに対するアクセス履歴を先端的サービス事業者ごとに確認できること	E5.アクセス・利用制限 E5.1.認証機能 E7.不正追跡・監視 E7.1.不正監視
3	サービス利用中	サービス利用者	サービス利用者が、自身のサービス利用記録をサービスごとに確認できること	先端的サービス事業者が、サービス利用記録を管理できること	E7.不正追跡・監視 E7.1.不正監視
				サービス利用者が、自身のサービス利用記録を先端的サービス事業者ごとに確認できること	E5.アクセス・利用制限 E5.1.認証機能
4	サービス利用中	サービス利用者	サービス利用者が、自身の機微なパーソナルデータの利用及び利用タイミングに関する同意を先端的サービス事業者職員の役割ごとに実施できること	先端的サービス事業者が、職員が利用する機微なパーソナルデータとその利用タイミングに関する利用権限を職員の役割ごとに管理できること	E5.アクセス・利用制限 E5.2.利用制限
				サービス利用者が、自身の機微なパーソナルデータの	E5.アクセス・利用制限

項番	発生タイミング	ステークホルダ	課題	機能概要案	非機能要求グレード（注）
				利用及び利用タイミングに関する同意を先端的サービス事業者職員の役割ごとに実施できること	E5.2.利用制限
5	サービス利用中	先端的サービス事業者	サービス利用者が、自身のパーソナルデータに関する開示請求を先端的サービス事業者に対してできること	先端的サービス事業者が、サービス利用者からのパーソナルデータに関する開示請求を受付けられること	－
				サービス利用者が、自身のパーソナルデータに関する開示請求を先端的サービス事業者ごとに実施できること	－
6	サービス利用解約時	サービス利用者	サービス利用者による利用解約時に、サービス利用規約に基づき自身のパーソナルデータが正しく処理（返却・削除など）されたことを確認できること	先端的サービス事業者が、サービスの利用解約を受け付けられること	－
				サービス利用者が、サービスの利用解約を実施できること	－
				サービス利用者が、サービスの利用解約時にサービス利用規約に基づき自身のパーソナルデータの返却を受けられること	E5.アクセス・利用制限 E5.1.認証機能
				サービス利用者が、サービスの利用解約時にサービス利用規約に基づき自身のパーソナルデータが削除されたことを確認できること	－
7	サービス利用解約時	サービス利用者	サービス利用者による利用解約後に、先端的サービス事業者間で自身のパーソナルデータが連携されていないことを確認できること	サービス利用者が、解約済みの先端的サービス事業者間で自身のパーソナルデータの連携が停止していることを確認できること	－

（注）E6.1 データ暗号化は全ての機能概要案に当てはまるため記載は省略している。

データ連携基盤運営組織が、データ連携基盤を整備する際に、本章で示した機能概要案及び非機能要求グレードを考慮して要件定義を実施することで、透明性を重視したパーソナルデータ管理に必要となる

基本的な要件を整理することができる。ただし、サービス利用者及び先端的サービス事業者の課題は、先端的サービスの特性や時間の経過とともに変化することが予想されるため、データ連携基盤運営組織による継続的な課題抽出と対策の実施が重要である。

9-3-2. 継続的なパーソナルデータ品質管理の検討

継続的なパーソナルデータ品質管理について、発生タイミング、「サービス利用者」「先端的サービス事業者」の各ステークホルダ視点での課題を表 9-3-2-1 に整理した。さらに、データ連携基盤を整備するために必要と考えられる機能概要案と関連する非機能要件に関して、IPA「非機能要求グレード」とのマッピングを行った。

表 9-3-2-1 継続的なパーソナルデータ品質管理の課題、機能概要案、非機能要求グレード整理

項番	発生タイミング	ステークホルダ	課題	機能概要案	非機能要求グレード（注）
1	サービス利用受付時	先端的サービス事業者	サービス利用者によるサービス利用申込時に、先端的サービス事業者ごとに管理している既存の個人 ID の紐づけが正しく実施されること	サービス利用申込時に、先端的サービス事業者が、先端的サービス事業者間で保有する既存の個人 ID 同士の紐づけを実施できること	－
				先端的サービス事業者が保有する既存の個人 ID の本人特定は、先端的サービス事業者が行うこと	－
				サービス利用申込時に、先端的サービス事業者が、誤った個人 ID の紐づけを実施した場合に訂正できること	－
				サービス利用申込時に、先端的サービス事業者が、個人 ID の紐づけを実施できなかった場合にサービス利用者に通知できること	－
2	サービス利用中	先端的サービス事業者	不正なパーソナルデータ（精度の不正、連携条件の違反）が収集及び連携されることを防止できること	サービス利用者のパーソナルデータ品質を上げるために、サービス利用者の手入力を極力減らし、信頼性の高いデバイス等を使用したデータ収集ができること	－
3	サービス利用中	先端的サービス事業者	定性的・定量的評価に基づくデータモデルの改善を継続的に行うことができること	パーソナルデータ品質を向上させるために、データセットを定期的に分析評価してデータセットをバージョンアップできること	－
				先端的サービス事業者が、データモデルのバージョン管理を行うことができるこ	E5.アクセス・利用制限 E5.2.利用制限

項番	発生タイミング	ステークホルダ	課題	機能概要案	非機能要求グレード（注）
				と	
				サービス利用者が、追加の個人情報に関わる項目が必要になった際には、再同意できること	E5.アクセス・利用制限 E5.2.利用制限
				先端的サービス事業者が、個人情報に紐づくデータ項目を含むデータモデルのバージョンアップ後に、サービス利用者が再同意したか否かの状態を把握できること	E5.アクセス・利用制限 E5.2.利用制限
				先端的サービス事業者が、新しいバージョンのデータモデルに対して、サービス利用者への追加同意を要求できること	E5.アクセス・利用制限 E5.2.利用制限
				サービス利用者が、自身のパーソナルデータの利用に関する同意を行う際、対象となるデータモデルのバージョンを確認できること	E5.アクセス・利用制限 E5.2.利用制限

（注）E6.1 データ暗号化は全ての機能概要案に当てはまるため記載は省略している。

データ連携基盤運営組織が、データ連携基盤を整備する際に、本章で示した機能概要案及び非機能要求グレードを考慮して要件定義を実施することで、継続的なパーソナルデータ品質管理に必要となる基本的な要件を整理することができる。ただし、サービス利用者及び先端的サービス事業者の課題は、先端的サービスの特性や時間の経過とともに変化することが予想されるため、データ連携基盤運営組織による継続的な課題抽出と対策の実施が重要である。

9-4. まとめ

本章では、データ連携基盤を長期的に運用するための課題整理、陳腐化しない運用の仕組み、安心してデータを扱える仕組みについて調査・検討を行った。

データ連携基盤を長期的に運用するための課題整理については、自治体等へヒアリングを実施した結果、表 9-1-2-1 に示す課題が明らかとなり、その解決策として 9-1-3 に示す仕組みを提案した。特に自治体から回答数の多かった課題として、データ連携基盤を長期運用するためには費用対効果やデータ連携基盤の導入効果を明らかにする必要がある。その解決策としては、スーパーシティ/スマートシティやデータ連携基盤の利活用の取り組みに関する自治体等の導入成功事例/失敗事例の共有（事例共有カタログ）の整備が有効と考えられる。また、データ連携基盤を安心して長期に利活用するためのデータ整備課題として、規約やルールに沿わないデータの利活用を、データ連携基盤側で制限する手段を設けるべきとの意見も回答数が多かった。その解決策として、データごとにデータ提供者/利用者の認証機能を設けたり、データへのアクセス権を付与したり、セキュリティ対策を講じたりするなどの対策が望まれる。

陳腐化しない運用の仕組みについては、運用を行う上で必要な観点を「認知度向上・普及促進」「利活用の向上」「運営」の 3 つに分解し、それぞれについて「データ連携基盤推進者」、「データ連携基盤構築事業者」、「データ連携基盤利用者」が何を実施しているか調査を行った。調査の結果から陳腐化しない運用に有効と考えられる施策案と、施策を実施する上での課題を提案した。データ連携基盤・API カタログ・開発者ポータルを運用する上で重要なことは、データ連携基盤推進者とデータ連携基盤構築事業者だけで運用を行うのではなく、データ連携基盤利用者や地域の住民、事業者等から意見を集め、地域と一体となった運用が望ましい。そのためには「地域住民参加型合意形成プラットフォーム」等を活用し、地域からの意見や要望、フィードバックを集め、改良・改善につなげることが有効な手段である。

安心してデータを扱える仕組みについては、9-3-1、9-3-2 で先端的サービスの例を起点に課題を洗い出し課題解決方法の検討を行った。その結果、「透明性を考慮したパーソナルデータ管理」としては、サービス利用開始時のサービス規約の明示とそれに対する同意管理（オプトイン）、サービス利用中にパーソナルデータがサービス利用規約に基づき扱われていることを確認する手段、パーソナルデータの開示・変更を請求するための手段、サービス利用終了時の解約手段（オプトアウト）などについて考慮する必要があることがわかった。次に、「継続的なパーソナルデータ品質管理」としては、サービスを提供するために必要十分なデータ項目を含むデータモデルが定義されていることを確認するための手段、収集されたデータ項目値が定められた精度を満たしていることを検証するための手段、定性的・定量的評価に基づくデータモデルの改善を継続的に行うために必要な手段などについて考慮する必要があることがわかった。

また、データ連携基盤を整備する際には、ここで定義した課題に紐付けた機能概要案以外に非機能要件についても提供する先端的サービスの特性を踏まえて定義することが重要である。非機能要件定義にあたっては IPA「非機能要求グレード」に沿って進めることが妥当であり、パーソナルデータの取り扱いに関する検討時には、表 9-3-1-1、表 9-3-2-1 に示した機能概要案を満たす非機能要求グレードを参照することを推奨する。

10. まとめ

本報告書のまとめとして、今回の調査結果から明らかになった事項のうち、データ連携基盤として、今後、深掘りした調査が必要と考える、以下 3 点についてまとめる。

- 都市間のデータ連携
- データ連携基盤を運用する上で必要となる事項
- 透明性を持ったパーソナルデータ管理

(1)都市間のデータ連携

「3 調査結果：データ連携基盤の全体像」で、「都市間連携における今後の課題」として挙げた事項を以下にまとめる。

表 10-1　都市間のデータ連携における課題と施策案

課題	施策案
各都市のデータ所在管理	分散型のデータ連携では、データのアクセス先（所在）の管理が必要となるが、連携対象となる都市が増大した場合、アクセス先の更新に関わる運用が肥大化する。全国を跨いだ都市全体のデータ所在を管理・制御する仕組みの検討を行う。
都市間の共通ルール	今回定義した API 共通ルールを実装することで都市間での相互接続は実現できると考えるが、コストや構築時期の関係により API 共通ルールを順守できないことも考えられる。指針となる API 仕様のサンプルの提示や、API の実装を容易にする開発ツールを提供、データモデルの継続的な整備等、今回策定した API 共通ルールを各エリアで採用するための仕組みが必要となると考える。
ユニークな ID 管理	データの分散管理を前提とした都市間連携の実現には、データ、利用者、組織、サービス、アセット（デバイス）をユニークな ID で管理する必要があり、このルール定義が必要であると考える。

(2)データ連携基盤を運用する上で必要となる事項

「9 調査結果（オ）基盤を長期運用する上で必要となる事項」で、ヒアリング等の調査結果や、陳腐化しない運用の仕組みで定義した、継続的な運用を行う上で必要な観点から導出した重要課題とその施策案を挙げる。

表 10-2　データ連携基盤を運用する上で必要となる事項における課題と施策案

課題	施策案
データ連携基盤の認知度向上・普及促進	• ホームページ、SNS 等のツールの活用等によるデータ連携基盤の広報活動 • 認知度向上・普及促進のための施策に対する効果測定と、その測定結果に基づき、施策を見直す PDCA サイクルの実施
データ連携基盤の利活用向上	• GitHub 等の活用、ヒアリング実施による利用者からのフィー

	ドバックを収集する仕組みの定義 • 開発者コミュニティの運営、展示会/勉強会/ハッカソン等のイベント開催
データ連携基盤の運用	• サービス利活用を促すための人材育成 • 運営課題や利用者要望に対する対応ルールの定義

(3)透明性を持ったパーソナルデータ管理

「9 調査結果（オ）基盤を長期運用する上で必要となる事項」の「安心してデータを扱える仕組み」で挙げたパーソナルデータ管理に必要な事項を挙げる。

表 10-3　透明性を持ったパーソナルデータ管理における課題と施策案

課題	施策案	
利用シーンに応じた透明性の確保	サービス利用開始時	サービス規約の明示とそれに対する同意管理(オプトイン)の定義
	サービス利用中	サービス利用規約に基づいて扱われていることを確認する手段、データの開示・変更を請求するための手段の定義
	サービス利用解約時	サービス利用終了時の解約手段(オプトアウト)の定義

上記のとおり、本報告書で詳細に調査した事項以外にも、さらなる調査が必要なものがあると考える。今回の報告書に留まらず、技術仕様、運用方法について、今後も整備、発展を続けていくことが必要である。