

エリアデータ連携基盤推奨モジュールの管理及び
自治体への運用支援等業務

共同利用を支える技術基盤：推奨モジュールと実践事例

2025年11月

一般社団法人データ社会推進協議会



1. 都市OSと推奨モジュール

- 都市OSとは
- 都市OSと推奨モジュールの関係
- APIとは
- データブローカーとは
- APIゲートウェイとは
- 都市OSを実現するための技術要素 まとめ

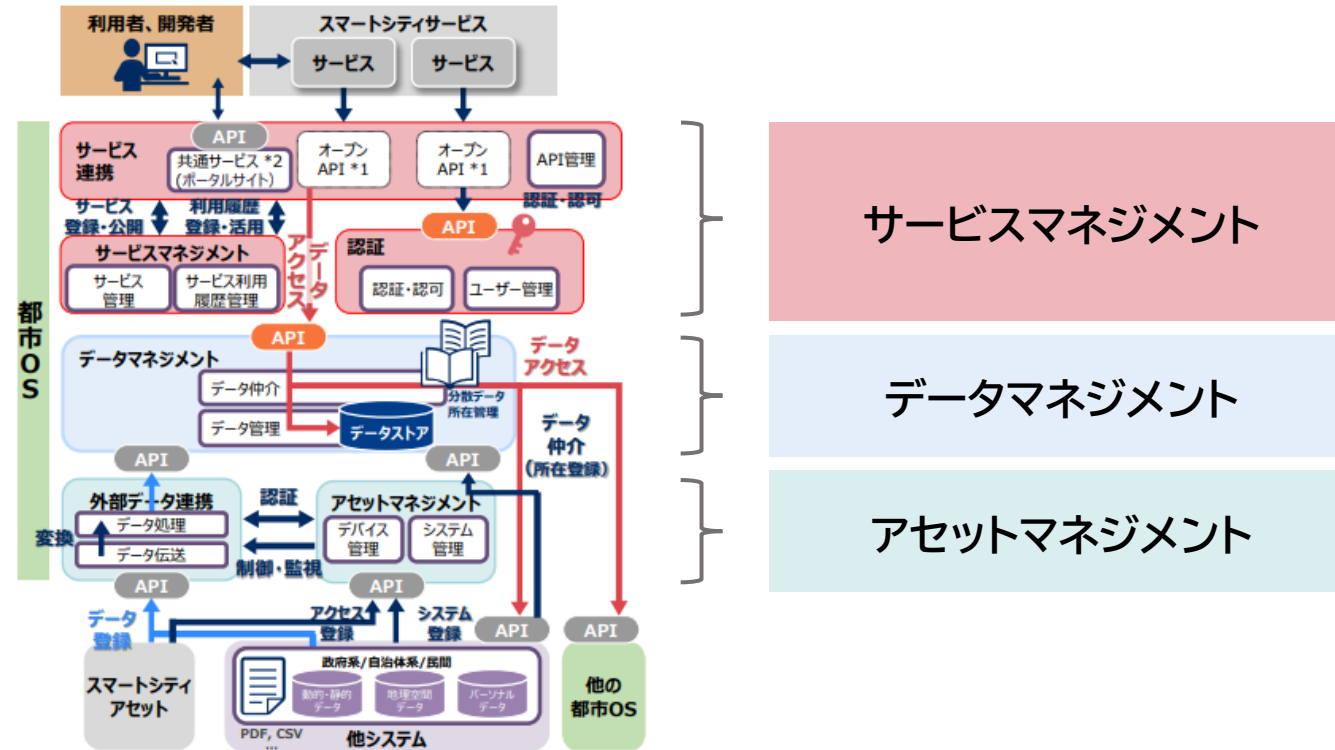
2. 共同利用とマルチテナントアーキテクチャ

- マルチテナントアーキテクチャのイメージ
- シングルテナント方式との比較
- Kong Gatewayにおけるマルチテナント設定の実践
- HTTPリクエストの構成要素
- HTTPリクエストの構成要素とKong Gatewayのルート設定
- Kong Gatewayでのポリシー適用
- 追加検討すべきポリシー設定 と 設定箇所
- FIWARE Orionにおけるマルチテナント設定の実践
- Kong GatewayとFIWARE Orionの連動
- Kong Gatewayと関連システムの連携設定
- マルチテナントの運用：まずはアクセスログからはじめる

都市OSと推奨モジュール

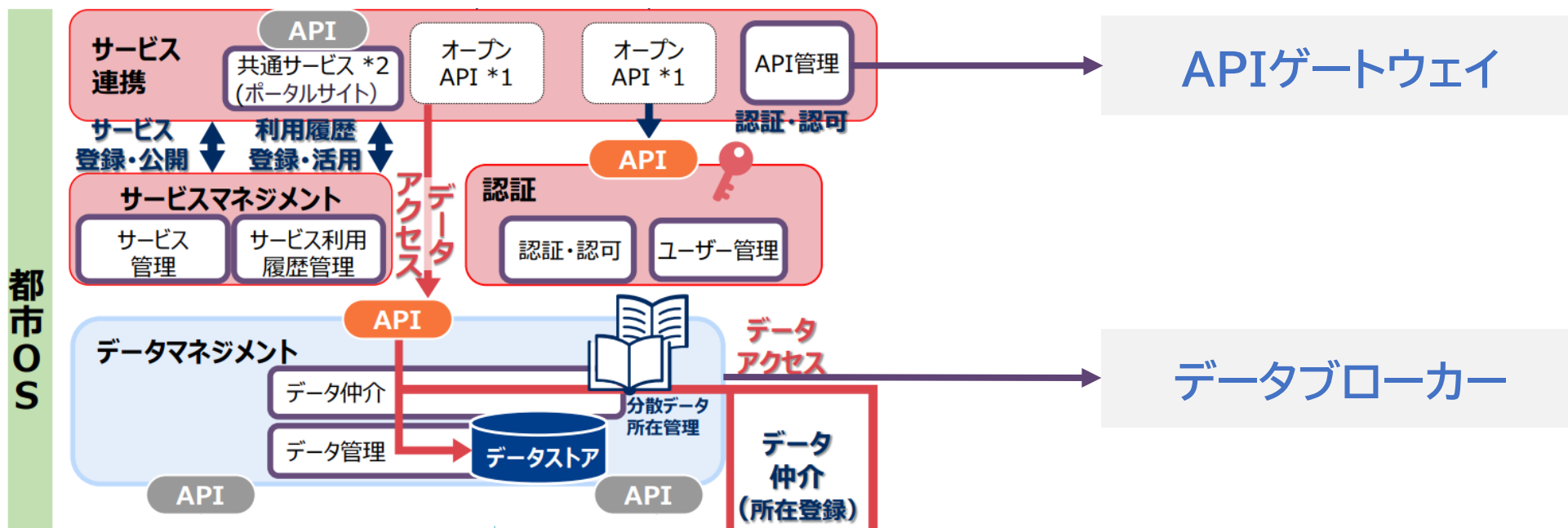


- 都市OSは、スマートシティの実現を目指す地域が運用するITシステムの総称。防災などのデータを収集/分析し、都市内外のサービスと連携する役割を担う。
- 主にサービス・データ・アセットの3つを管理することでサービスを提供している。



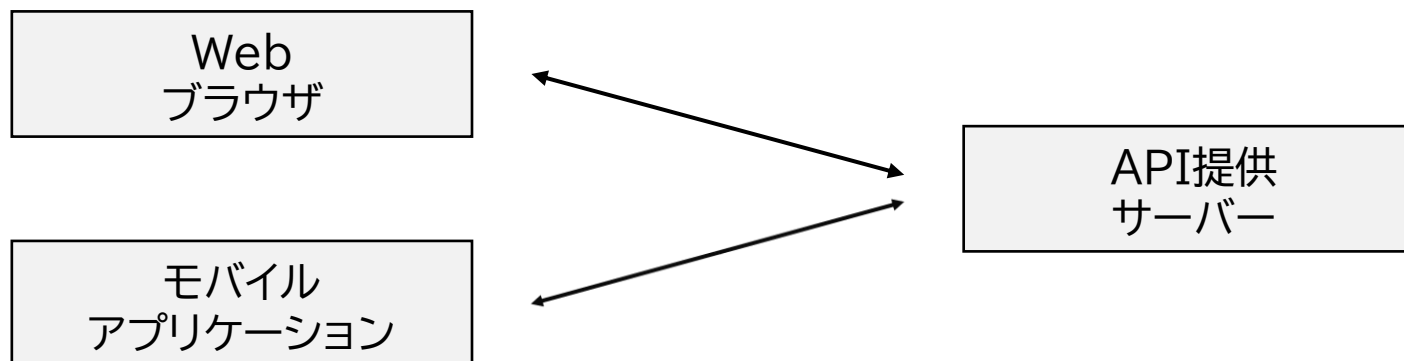
“スマートシティリファレンスアーキテクチャの使い方(導入ガイドブック)(第2版)” (p. 25).
https://www8.cao.go.jp/cstp/society5_0/smartcity/architecture.html

- 都市OSの構築で最低限必要となるコアモジュール(**APIゲートウェイ**、**データブローカー**)は**推奨モジュール**と定義されている。
- 「どんなデータを利活用したいのか」によって必要機能の拡充を検討することが重要である。



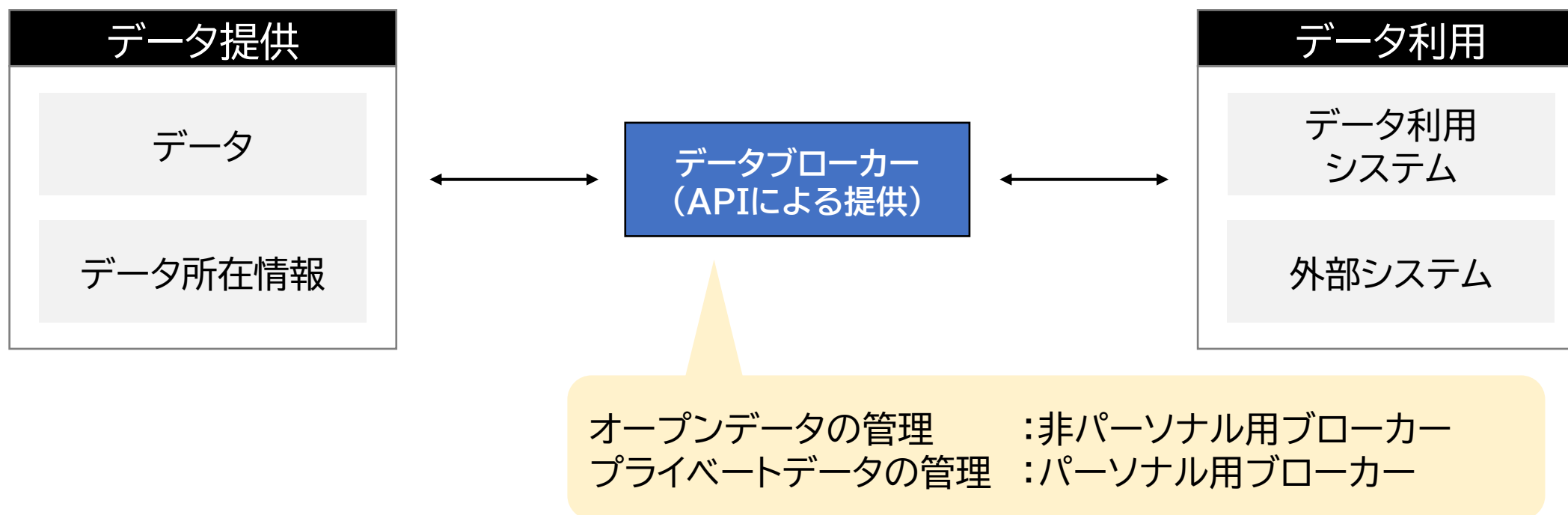
“スマートシティリファレンスアーキテクチャの使い方(導入ガイドブック)(第2版)” (p. 25).
https://www8.cao.go.jp/cstp/society5_0/smartcity/architecture.html

- APIは、プログラムから機能を利用するためのインターフェースの総称。
APIのうち、特にWebを介して利用するのはWeb APIと呼ばれ、主にインターネット経由で接続可能なWebサーバーが提供している。



API : Application Programming Interface

- データブローカーは、データ自体またはデータの所在情報を収集・管理し、これらを利用する内外のシステムにAPIを提供する役割を担う。
- 利活用したいデータの所有権によって2種類のブローカーを使い分けることが重要。

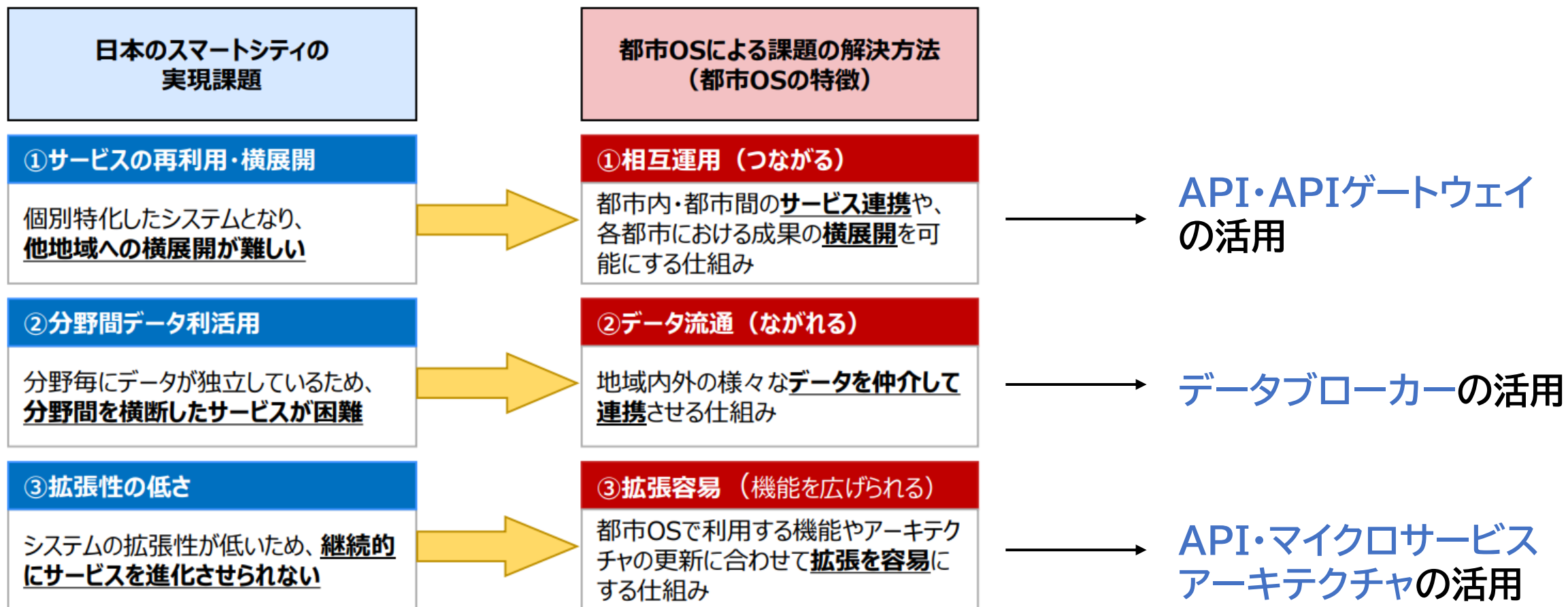


- APIゲートウェイは、マイクロサービスとして提供されたAPIを管理し、クライアントに提供する役割を担う（サービスと利用者の管理）。
- ブローカーはAPIゲートウェイで管理可能なAPI提供サーバーの一つで、都市OSではセットで利用することが一般的である。



データブローカーは、登録したデータ・データの所在情報を管理するAPI提供サーバーの一種。

都市OSを実現するための技術要素 まとめ



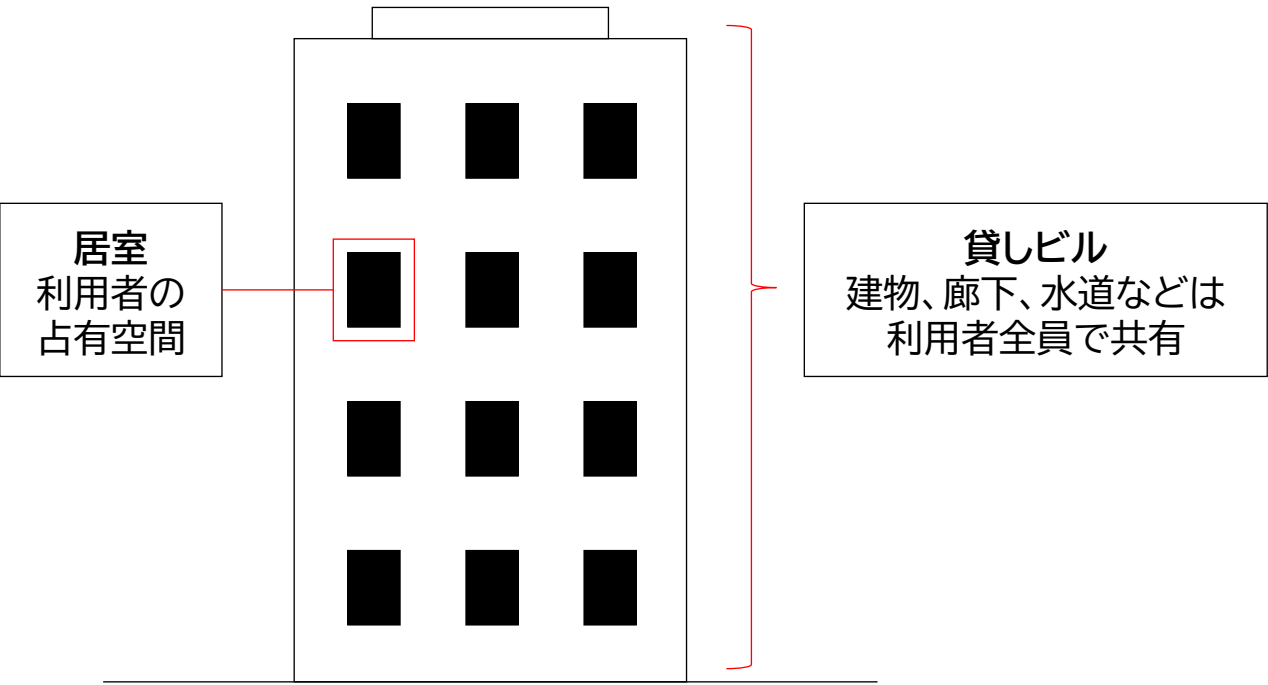
“スマートシティリファレンスアーキテクチャの使い方(導入ガイドブック)(第2版)” (p. 22).
https://www8.cao.go.jp/cstp/society5_0/smartcity/architecture.html

共同利用とマルチテナントアーキテクチャ



マルチテナントアーキテクチャのイメージ

- マルチテナントとは、同一システムを複数のユーザーで共有するモデルを指す。
- システム構成上の メリット や デメリット は、貸しビルの利用イメージと同様である。

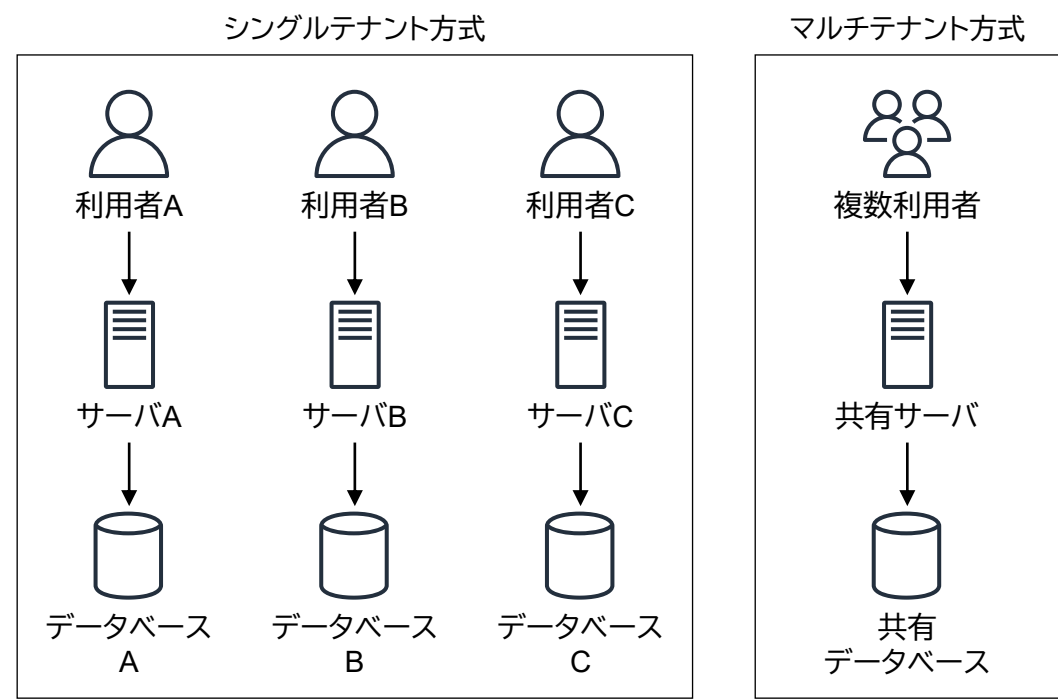


メリット

デメリット

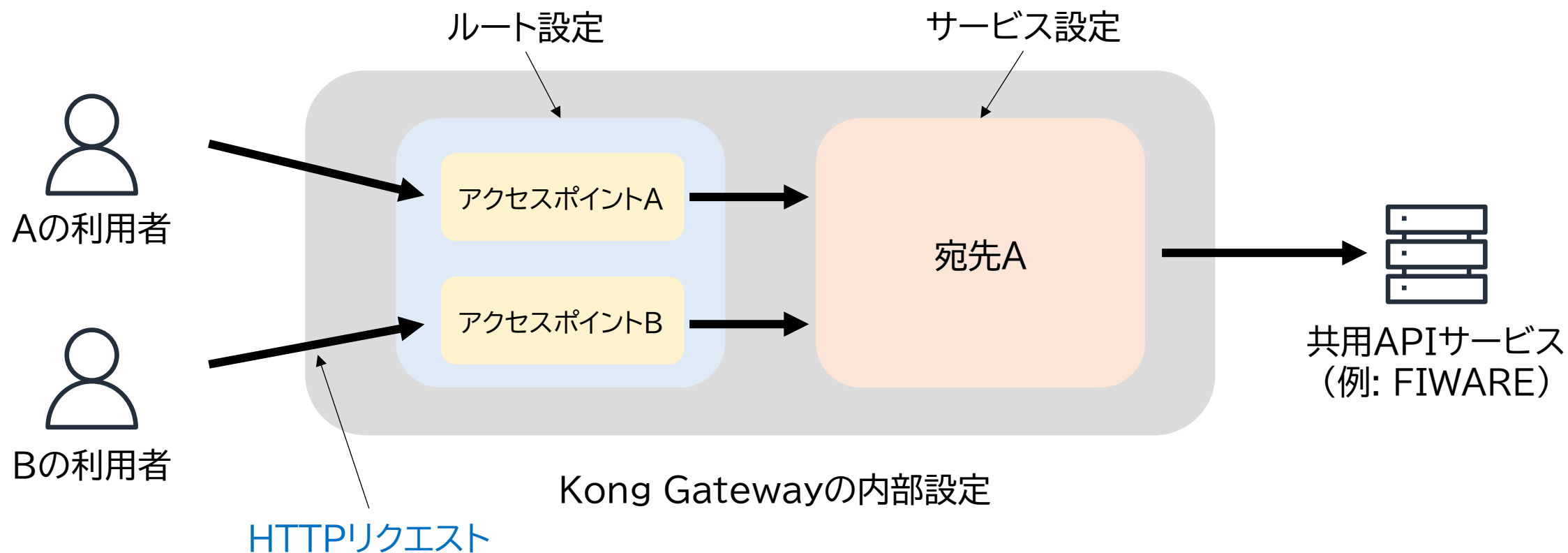
観点	貸しビルでの例
インフラの共有による全体コストの削減	建物の家賃を全員で払う
中央集権的な保守・運用の容易さ	管理人が一括で設備を点検する
テナント追加の容易さ	空き居室にすぐ入居できる
セキュリティやデータ分離の難しさ	騒音などの問題が発生しやすい
カスタマイズの制限	共有設備のリフォームが禁止されている
パフォーマンス競合のリスク	水道などを大量消費すると全体に影響がでる

- マルチテナントは コスト効率に優れる 反面、メンテナンス時間の共通化など 運用調整が必要 になる。
- 他の利用者と運用要件が乖離していないことを確認するなど、特性を理解した活用が重要。

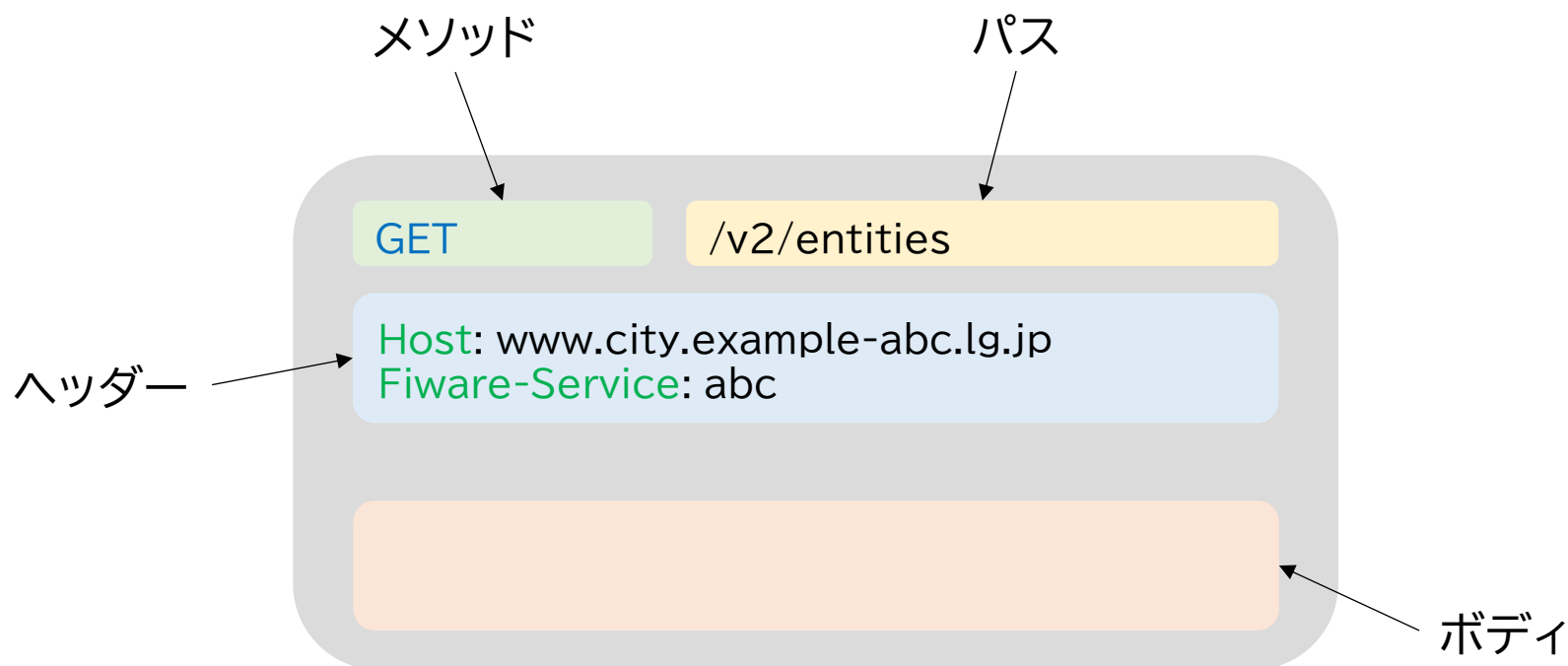


観点	シングルテナント	マルチテナント
環境分離	完全分離	論理分離
コスト効率	低い	高い
性能保証	個別の最適化が可能	利用者全体での最適化検討が必要
障害影響	他利用者への影響なし	1利用者の障害が全体に波及する可能性がある
運用調整	個別対応	共通メンテナンススケジュールの合意が必須

- APIゲートウェイには複数のアクセスポイントが設定でき、Kongでは ルート と呼ばれる。
- 同じ宛先(サービス)に対して、テナントごとに ルート を設定することで複数の独立したアクセスポイントを設けることができる。



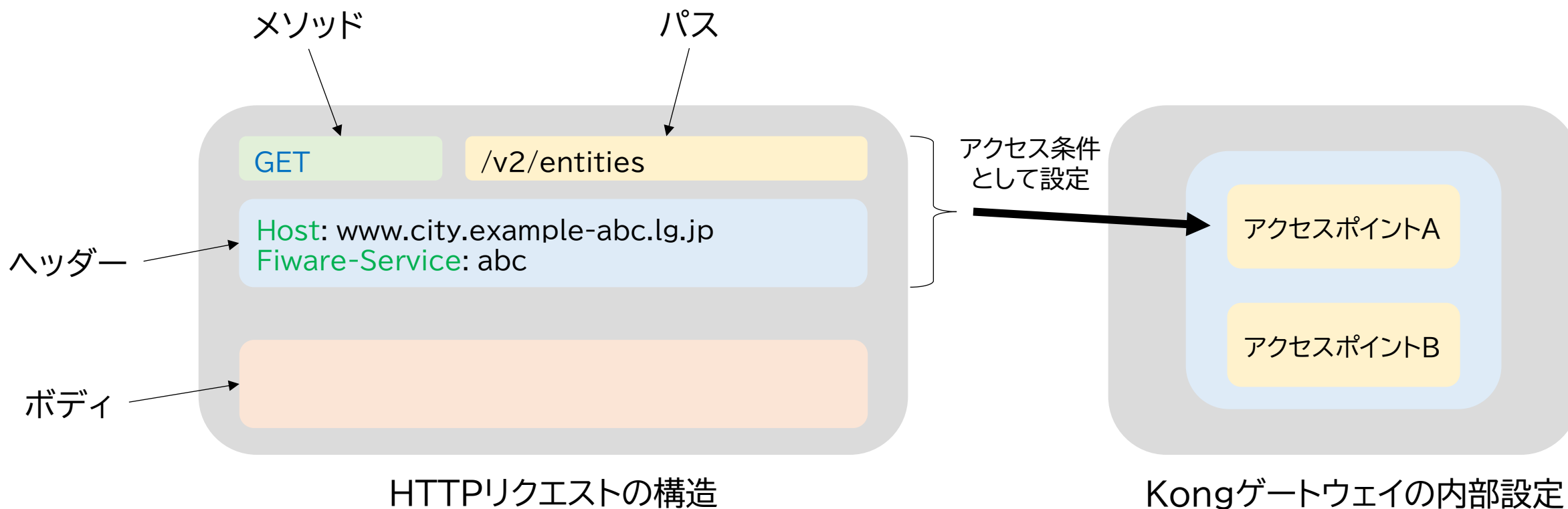
- Web APIの通信では、クライアントがサーバーに処理を要求する HTTPリクエスト を送信する。
- 通信内容に関連するメタ情報は、ボディ(本文)とは別で ヘッダー に記載される。
- 例えばテナントごとにドメインが違う場合は、Hostヘッダーが異なるHTTPリクエストが利用される。



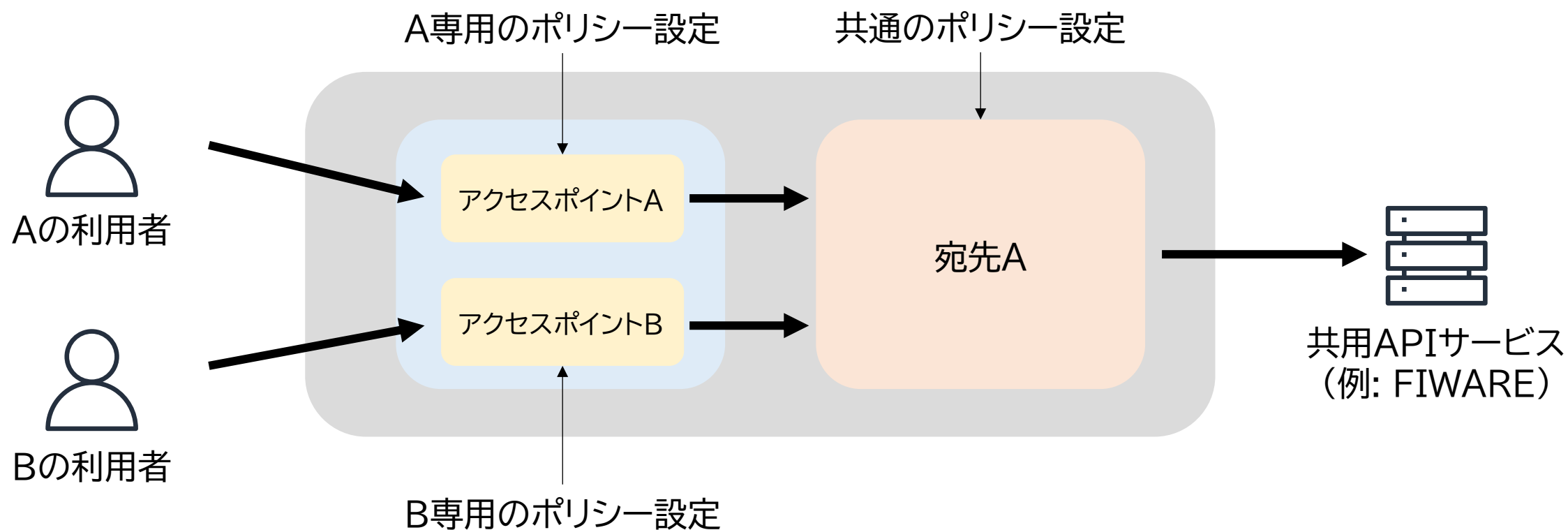
HTTPリクエストの構造

HTTPリクエストの構成要素 と Kong Gatewayのルート設定

- アクセスポイントには、テナント固有のヘッダー情報 をアクセス条件として設定する。
- Kongの場合は複数のドメインや証明書が指定できるため、ドメインによるルート設定が可能である。

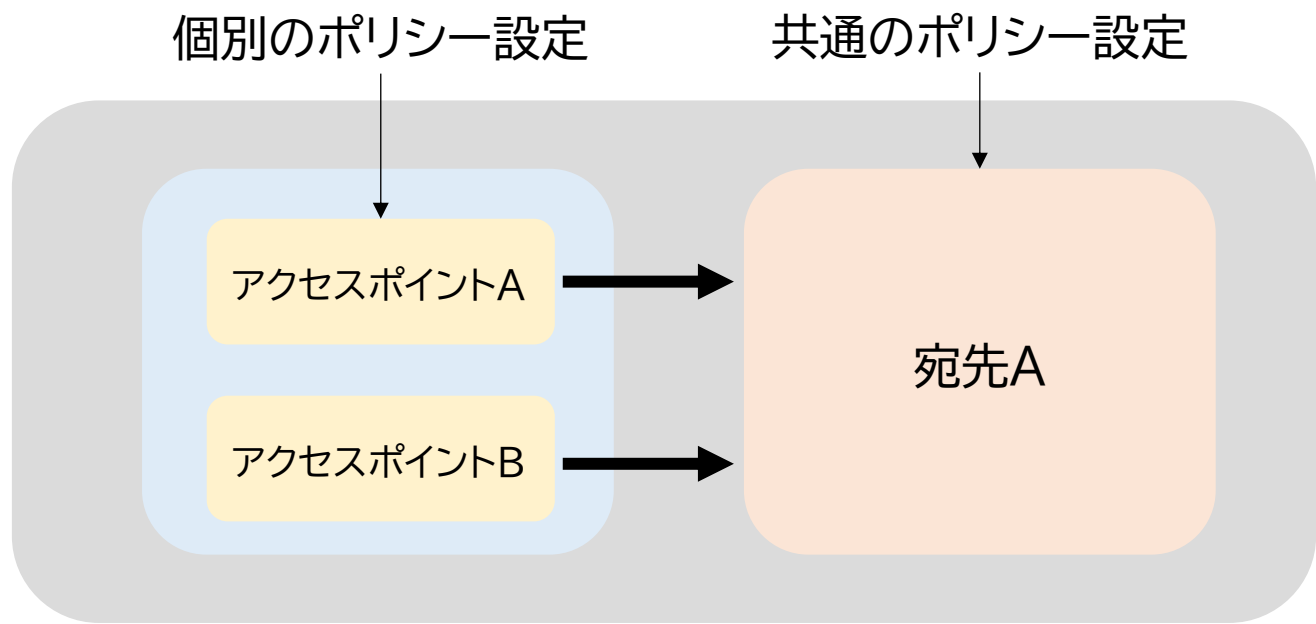


- APIゲートウェイでは、認証や流量制御などの ポリシー設定 を追加してHTTPリクエストを制御できる。
- Kongの場合は、プラグイン 設定をルートやサービスに適用することでポリシーが追加できる。
- アクセスポイントを分けることで、固有のポリシー設定を指定することができる。



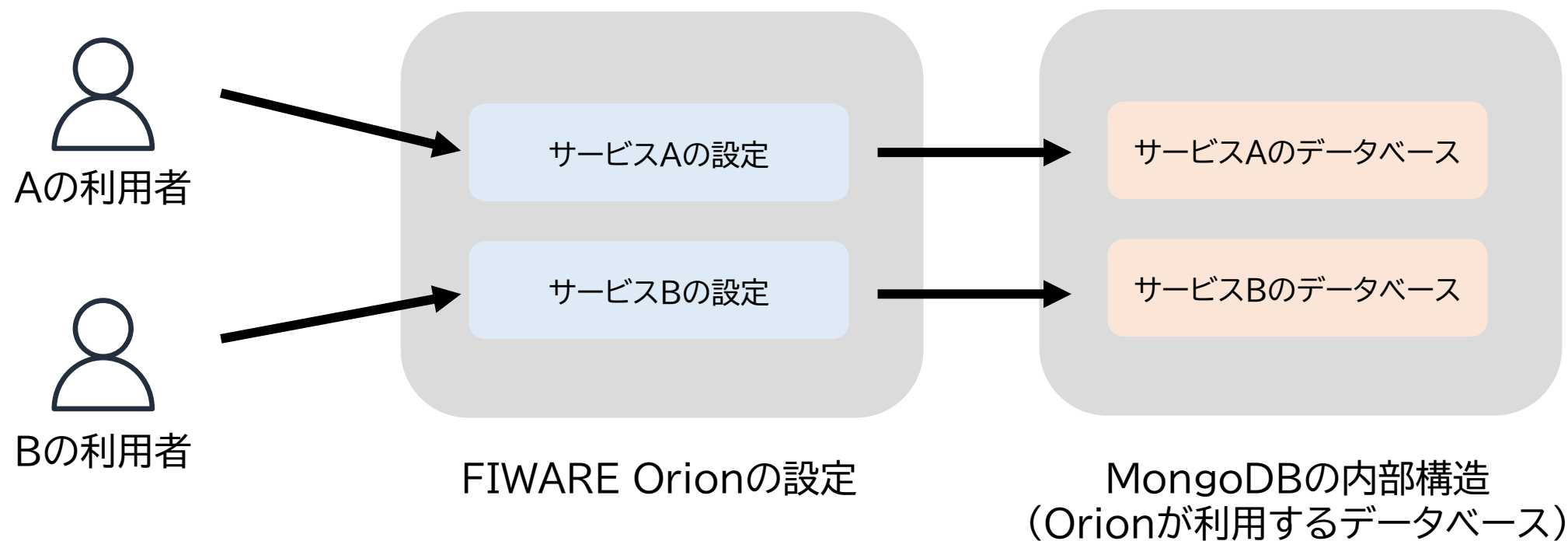
追加検討すべきポリシー設定 と 設定箇所

- 居室ごとに鍵が違うように、認証設定などは個別テナントが責任を持つことが望ましい。
- 安全性や可観測性に関連するアクセスログの収集は、共通ポリシー として全体適用することが望ましい。
これはテナントビルで、監視カメラによる入口のモニタリングが共通な点と同様である。



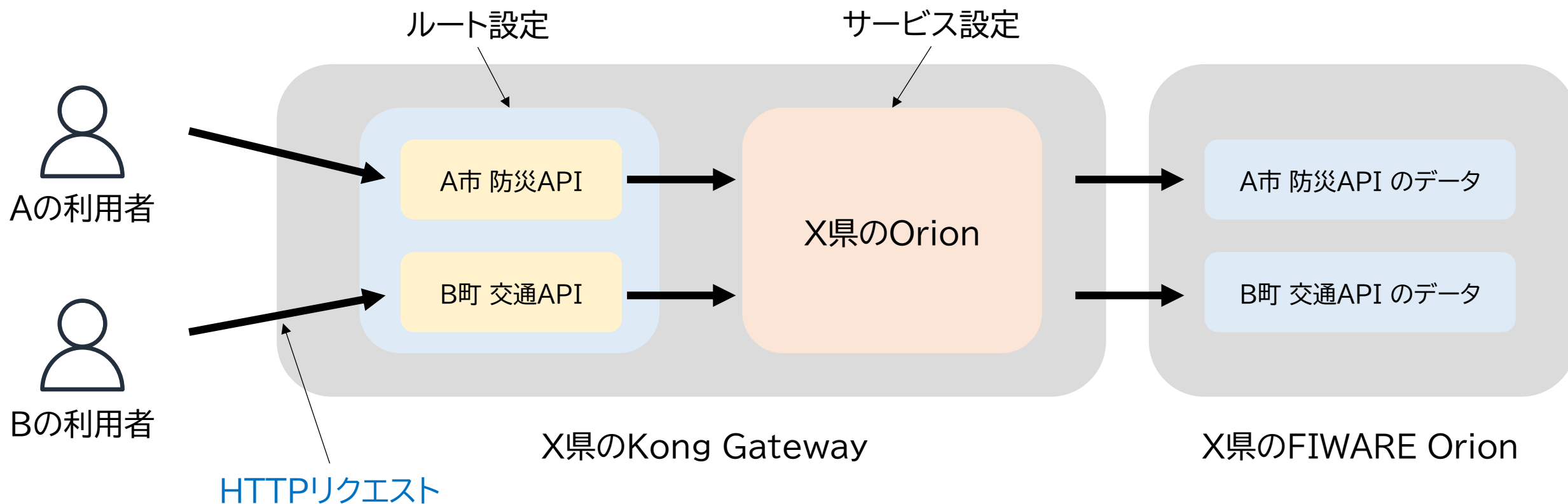
ポリシー名 (例)	設定箇所
認証・認可	個別
流量制限・IP制限	個別
メッセージ置換	個別
アクセスログ収集	共通
不正アクセス対策(Botなど)	共通
メンテナンス時の遮断通知	共通

- FIWARE Orionには複数のテナントが設定でき、これらは Fiware-Service と呼ばれる。
- Orionは、APIリクエストに指定されている Fiware-Serviceヘッダー でサービスを判断する。
- Orionは、サービスごとに異なるデータベースヘデータを登録する。



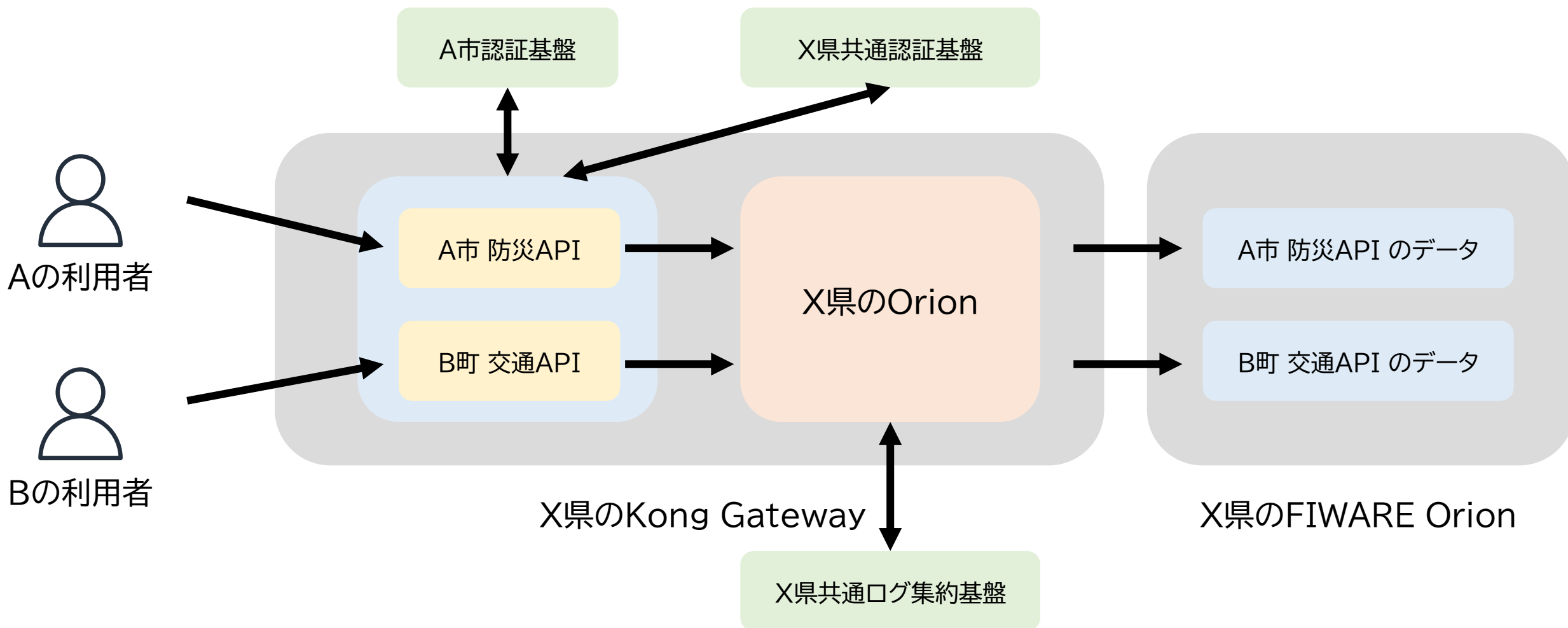
Kong GatewayとFIWARE Orionの連動

- 利用者は、HTTPリクエストヘッダーに、テナント固有情報を付与してリクエストする。
- Kongは、ヘッダー情報を元に個別のルート設定(とそのポリシー)を実行する。
- ヘッダー情報は 宛先のOrionにも転送される ため、ヘッダーに基づくデータが返却される。



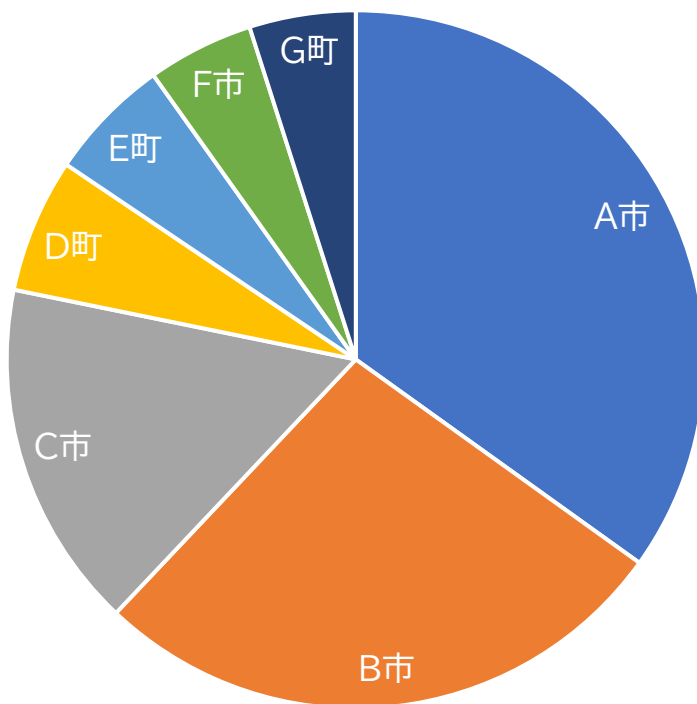
Kong Gatewayと関連システムの連携設定

- Kongを 認証基盤 や ログ集約基盤 と連動させることで、実践的なポリシー設定が適用できる。



マルチテナントの運用：まずはアクセスログからはじめる

- Web APIの利用実態調査は、アクセスログの収集・分析 が基本的かつ不可欠な要素になる。
- 監査に使えるほか、共有リソースの従量課金に対する根拠の一つとしても利用できる。
- 複雑な分析ができると理想だが、まずはログ集約と基本的な集計による 状況の可視化 が望ましい。



X県におけるAPI利用回数の集計結果イメージ

