最終更新日：2023/3/16

トラックデータ
標準APIガイドライン0.1版

目次

※各項目の詳細内容については別途協議にて最終決定

目次
0.トラックデータ標準APIガイドラインについて
1.API設計
  1-1.URI設計
  1-2.パラメータ設計
  1-3.リクエストデータ設計
  1-4.レスポンスデータ設計
  1-5.エラーレスポンス設計
  1-6.個別データのパラメータ設計
2.API定義
3.認証認可方式

付録
1.全体構成図・機能分担例
2.API連携フロー

用語定義

※各項目の詳細内容については別途協議にて最終決定

用語	定義
商用車メーカー	本事業に参加するトラックを製造・販売するメーカーの総称
運送事業者	トラックを使用する運送事業者の総称
他事業者	商用車メーカー・運送事業者以外のその他民間事業者の総称
データ連携許諾	車両データに関する権利者が車両のデータ連携を他者に許すこと
商用車メーカーバックエンド	商用車メーカーシステムの総称
共通基盤	商用車メーカーシステムに共通で要求される一部機能を代替する基盤の総称

用語定義　データ連携におけるステークホルダー

※各項目の詳細内容については別途協議にて最終決定

商用車メーカー

- いすゞ自動車
- 日野自動車
- 三菱ふそう
- UDトラックス

運送事業者

- ヤマト運輸
- 佐川急便
- 日本通運
- 日本郵便
- 日立物流
- 山九
- ハンナ
- A社
- B社
- C社
- D社
- ……

他事業者

- デジタコメーカ
- サービスプロバイダー
- その他民間事業者

※詳細項目については別途協議にて最終決定

トラックデータ標準APIガイドライン0.1版 – トラックデータ標準APIガイドラインについて

■トラックデータ標準APIガイドライン策定の背景・目的

背景）トラック運送業界における長時間労働等を背景にドライバ不足が深刻化しており、将来の担い手確保のためにも、働き方改革は喫緊の課題である。また、我が国を取り巻くモビリティの自動化・電動化の流れ、カーボンニュートラル、Society 5.0 の実現や移動及び物流に係る社会課題の深刻化等の社会環境の変化を受けて、商用車分野で創出される様々なデータは、その分野に留まらず分野横断的に利活用することにより社会課題の解決や新たな価値・サービスの創出に貢献することが期待されている。上記をうけて、2019年より開始した経済産業省 物流MaaSの取り組みの中で、「トラックデータ情報連携基盤の確立」事業を通して本ガイドライン策定検討を実施している。

目的）物流分野における慢性的な人手不足等の社会課題や環境課題に対し、日本版トラックデータ標準を策定のうえで、標準的なトラックデータ情報連携の仕組みを確立し、その解決や付加価値向上を目指す。

■本APIガイドラインの基本的な考え方

・本ガイドラインは、内閣官房情報通信技術（IT）総合戦略室の定めるAPIテクニカルガイドブックをベースに作成
・APIテクニカルガイドブックは銀行分野のオープンAPIでもベースとされているもの
・本ガイドラインでは、トラックデータ連携に必要になるであろう項目をAPIテクニカルガイドブックから抜粋する形で構成されている
・本ガイドラインの0.1版はトラックデータ連携に必要であろう項目を目次レベルで記載している
・本ガイドラインの詳細はヒヤリハットマップの事例を参考に記載しているが、詳細は来年度以降で決定していくものとする

トラックデータ標準APIガイドライン0.1版 – API設計

※詳細項目については別途協議にて最終決定

■URI設計
URI設計では、以下のポイントを考慮して設計することとする

	概要
1.	動詞ではなく名詞を用い、複数形とする
2.	抽象的な名詞は避け、API利用者がリソースの内容を把握できる単語を用いる
3.	スネイクケース（単語間をアンダースコアで繋げる表記法）を利用する
4.	APIのバージョンをURIに含める。バージョン番号は「v」+「整数」で表記し、バージョン番号は整数にする
5.	リソースの操作はURIではなくHTTPメソッドで表す（GET:データ取得、POST:新規登録、PUT:更新、DELETE:削除）
6.	URIの階層が深くなる場合には、パラメータで指定する
7.	URI設計は、RFC 6570「URI Template」に示されるURI Templateで指定できる形式とする
8.	運用開始後には、基本的にベースURIを変更しない

例）ヒヤリハットマップ
・https://api.example.com/v1/risk_locations

トラックデータ標準APIガイドライン0.1版 – API設計

※詳細項目については別途協議にて最終決定

■パラメータ設計
パラメータ設計では、以下のポイントを考慮して設計することとする

	概要
1.	1つの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。
2.	ページネーションにおけるパラメータは、取得開始ページ（「offset」または「page」）と件数（「limit」とする） 返却するデータの初期値は100件以下を推奨とする
3.	レスポンスデータに10件以上の返却項目を含む場合、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、「fields」パラメータを指定し要求する項目のみ返却するようにする

例）パラメータ

名称	推奨度合	概要
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。
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
page	△	返却して欲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
sort	○	指定された条件を元に並び替えて結果を返却する
encode	△	文字コードを指定する
fields	○	指定した項目のみを返却する

トラックデータ標準APIガイドライン0.1版 – API設計

※詳細項目については別途協議にて最終決定

■リクエストデータ設計
リクエストデータ設計では、以下のポイントを考慮して設計することとする

	概要
1.	リクエストは、基本的に単純なパラメータによる値、又はカンマ区切りによる配列の受け渡しによって行う 単純なパラメータやカンマ区切りデータとしてリクエストすることが難しい場合には、リクエストデータの形式はレスポンスデータ形式と同一のものとする
2.	マルチバイト文字（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
3.	文字コードは UTF8 を用いる。元データが Shift-JIS など、 UTF8 ではない場合、文字を変換する際の文字化け・文字欠けに留意し、 必要に応じて、文字コードの変換、文字の縮退を行う
4.	APIのリクエストデータに外字は利用しない
5.	日本語のように、非英字を用いる場合にはスネイクケース（単語間をアンダースコアで繋げる表記法）を使用する
6.	パーセントエンコーディングで表現される文字列は URI に含めない

トラックデータ標準APIガイドライン0.1版 – API設計

※詳細項目については別途協議にて最終決定

■レスポンスデータ設計
レスポンスデータ設計では、以下のポイントを考慮して設計することとする

	概要
1.	レスポンスデータにはタグ情報を持たず、XMLに比べてデータ量が少なく、タグ処理にかかる負荷が少ないJSON形式（RFC4627「The application/json Media Type for JavaScript Object Notation(JSON)）を利用する
2.	API利用者の利便性の観点から、zip等のバイナリ形式のデータは使用しない
3.	リクエストデータと同様に、文字コードはUTF8を用いる。必要に応じて文字の縮退を行う
4.	APIのレスポンスデータに外字は利用しない
5.	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を判断することができるようにすることが望ましい
6.	単純なデータ形式では表現が容易ではない画面遷移などに代えて、JSON+Hypertext Application Language(HAL)形式を用いリソースへのリンクを示すことで、リソース間の遷移ができるようにする。特に手続きAPI（一連の処理を通じ、手続に関する機能を提供するもの）では、規定のフローに従って順次APIを呼び出すことが多いため、API間の繋がりを示すAPI管理等の仕組みがあることが望ましい

６の例）リソースリンク

```
{
  "_links": {
    "self": {
      "href": "/v2/risk_locations"
    },
    "next": {
      "href": "/v2/risk_locations?page=2"
    },
    "find": {
      "href": "/v2/risk_locations{?id}",
      "templated": true
    }
  }
}
```

トラックデータ標準APIガイドライン0.1版 – API設計

※詳細項目については別途協議にて最終決定

■エラーレスポンス設計
エラーレスポンス設計では、以下のポイントを考慮して設計することとする

	概要
1.	RFC7807「Problem Details for HTTP APIs」に示されるデータ項目を利用する

例）データ項目

名称	推奨度合	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が無い時には、”about:blank”と入力する
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	△	エラーが発生した場合のURIを記載する

例）エラーレスポンス

```
HTTP/1.1 404 Not Found
Content-Type: application/problem+json
Content-Language: ja
{
  "type": "https://example.com/risk_locations/not-found",
  "title": "ヒアリハットマップに地点が登録されていません",
  "detail": "ヒアリハットマップに地点を登録する必要があります"
}
```

トラックデータ標準APIガイドライン0.1版 – API設計

※詳細項目については別途協議にて最終決定

■個別データのパラメータ設計
個別データのパラメータ設計では、以下のポイントを考慮して設計することとする

	概要
1.	レスポンスデータ等は国際標準やデファクト標準等に従って、データ項目名及び内容を設定する

表記方法

内容	表記方法	表記例
日時	ISO08601に準拠	日付のみであれば、「2023-01-18」 日時（日本時間）であれば、「2023-01-18T09:00:00+0900」
言語	ISO 639-1に準拠	日本語の場合「ja」、英語の場合「en」
国	ISO 3166-1 alpha-2に準拠	日本の場合「JP」、米国の場合「US」
都道府県	JIS X 0401に準拠	東京都の場合「13」
市区町村	JIS X 0402に準拠	東京都千代田区の場合「101」
緯度経度	緯度経度データはWGS84（米国世界測地系）に準拠	東京都千代田区永田町2丁目3-1の場合、「35.672947,139.742622」

※詳細項目については別途協議にて最終決定

トラックデータ標準APIガイドライン0.1版 – API項目定義（サンプルAPI①）

■API設計
例）トークン取得API

項目	概要	サンプル
リクエストメソッド	POST	POST /v1/tokens
リクエストパス	/v1/tokens	

リクエスト
パラメータ
	REQUEST BODY SCHEMA: application/json
	id string ユーザID
	key string ユーザ鍵
		Payload
		Content type
		application/json
		Copy
		{
		"id": "123456",
		"key": "{$UserKey}"
		}

レスポンス
フォーマット
	RESPONSE SCHEMA: application/json
	token string トークン
		201
		Content type
		application/json
		Copy
		{
		"token": "XXXXXX"
		}

※詳細項目については別途協議にて最終決定

トラックデータ標準APIガイドライン0.1版 – API項目定義（サンプルAPI②）

■API設計
例）ヒヤリハットマップ情報取得API

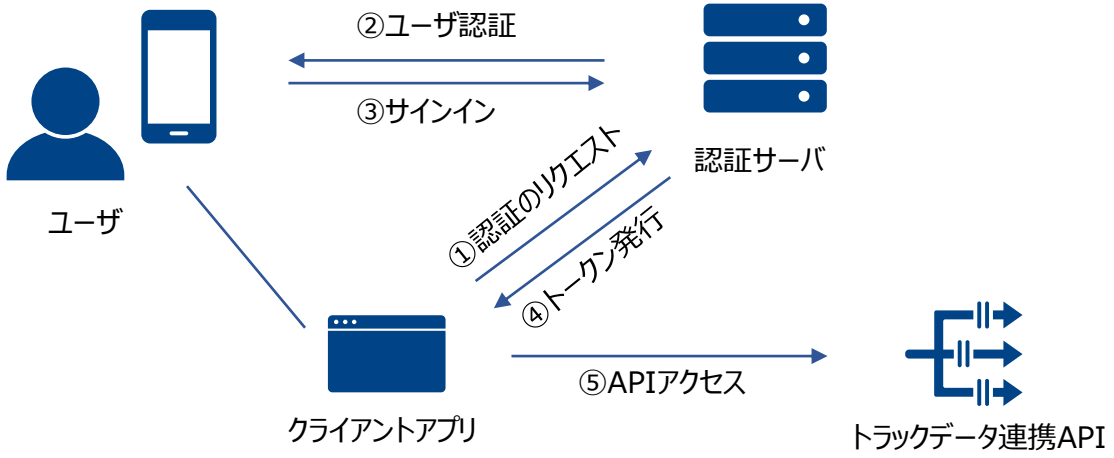
項目	概要	サンプル
リクエストメソッド	GET	GET /v1/risk_locations
リクエストパス	/v1/risk_locations	
リクエスト パラメータ	HEADER PARAMETERS token required — string 認証トークン	
レスポンス フォーマット	RESPONSE SCHEMA: application/json id — integer <int64> ヒヤリハットマップID date — string 発生日時(ISO08601準拠) latitude — number <float> 緯度(WSG84準拠) longitude — number <float> 経度(WSG84準拠) car_class — integer <int64> 車格（大型、中型、小型）	200 Content type application/json Copy Expand all Collapse all [{ "id": 1, "date": "2014-01-01T13:50:40+0900", "latitude": 34.49194, "longitude": 133.9459, "car_class": 1 }, { "id": 2, "date": "2015-10-20T13:50:40+0900", "latitude": 34.49194, "longitude": 133.9459, "car_class": 2 }, ...

トラックデータ標準APIガイドライン0.1版 – 認証認可方式

※詳細項目については別途協議にて最終決定

・API連携では、認証認可方式に OIDC を採用
- RCF 6749 で規定された認可フレームワーク OAuth2.0 をベースとし、認証まで行う拡張仕様
- OAuth2.0 はデジタル庁のデータ連携WTでも採用されている、データ連携の標準方式

OpenID Connect のフロー



トークンによる認証をベースとした方式を採用することのメリット

■セキュリティ面のメリット
- データ利活用事業者にユーザのID/PASSを預ける必要が無い
- 有効期限内が設定されており、万が一トークンが盗まれても被害が少ない

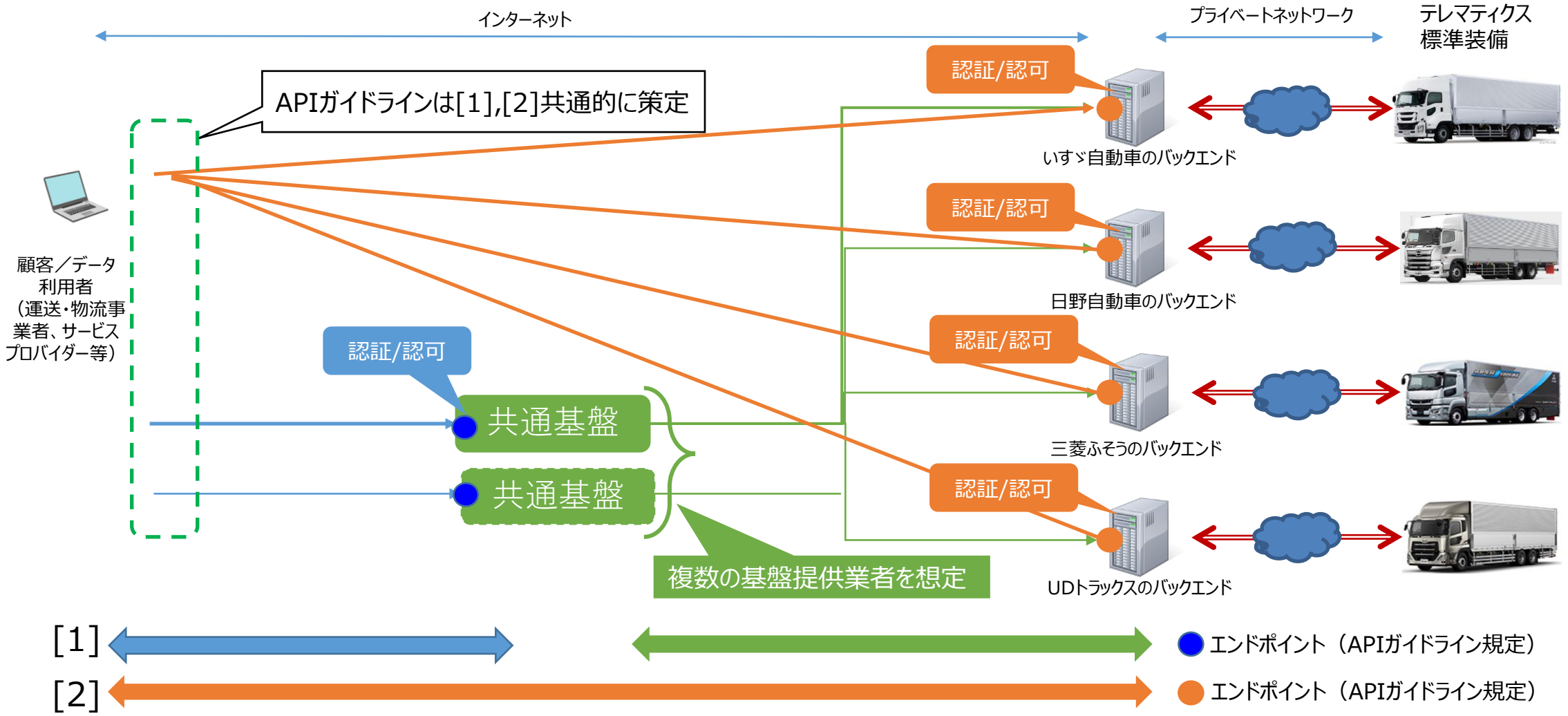
■利便性面のメリット
- ユーザが毎回ID/PASSによるログインを実施する必要が無い

トラックデータ標準APIガイドライン0.1版 – 全体構成パターン図

※詳細項目については別途協議にて最終決定

ガイドライン規定　ガイドライン規定　ガイドライン規定外

[1]共通基盤がAPIガイドラインに準拠した認証/認可を行い、共通基盤を介して複数商用車メーカーとのデータ連携を実施する構成パターン
[2]商用車メーカーがAPIガイドラインに準拠した認証/認可を行い、直接データ連携を実施する構成パターン



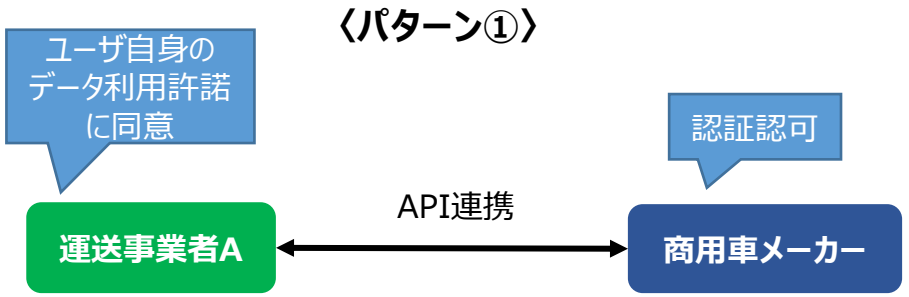
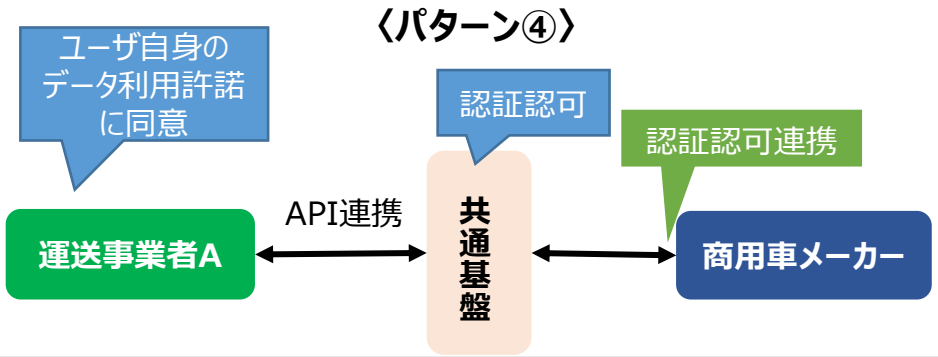
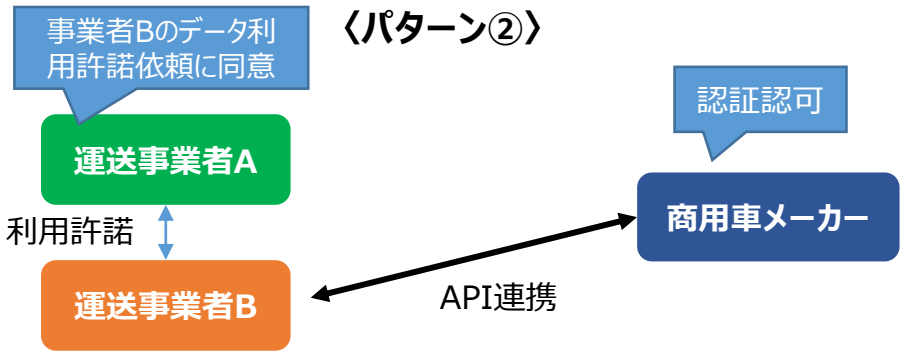
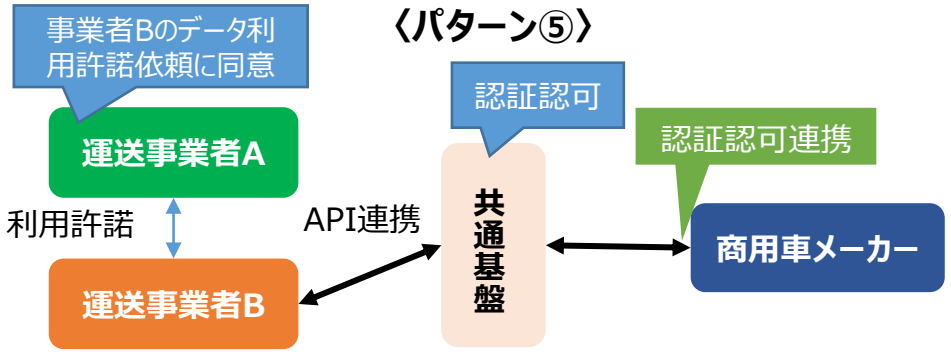
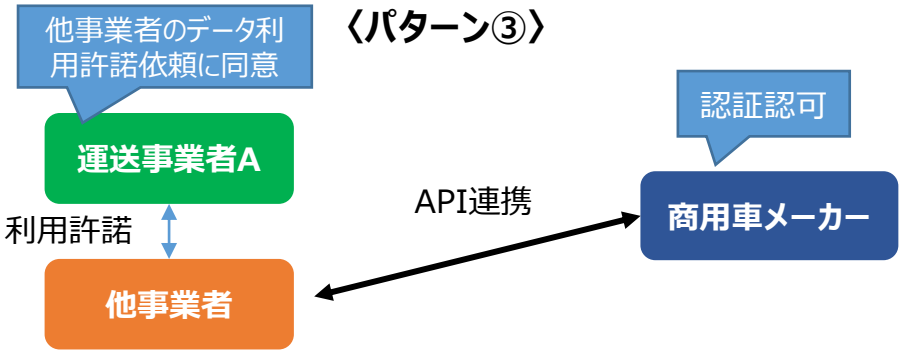
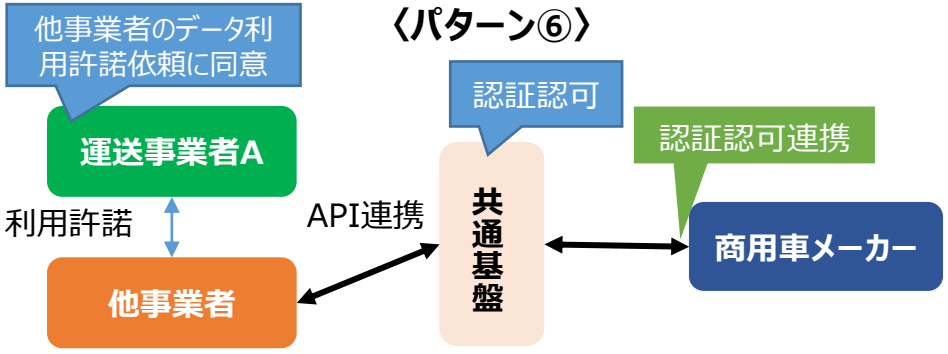
トラックデータ標準APIガイドライン0.1版 – 機能分担例

※詳細項目については別途協議にて最終決定

機能名	機能概要	構成パターン[1]での分担	構成パターン[2]での分担
API利用者	APIを利用する	運送事業者、他事業者	
API連携データ収集	車載器よりデータを収集する	商用車メーカーバックエンド	商用車メーカーバックエンド
API連携データ整形	収集したデータをAPIレスポンスフォーマットに合わせて整形する	未定義（商用車メーカーバックエンドor共通基盤）	商用車メーカーバックエンド
認証	ユーザを認証する	共通基盤	商用車メーカーバックエンド
認可	ユーザを認可する	共通基盤	商用車メーカーバックエンド

トラックデータ標準APIガイドライン0.1版 – API連携パターン

※詳細項目については別途協議にて最終決定

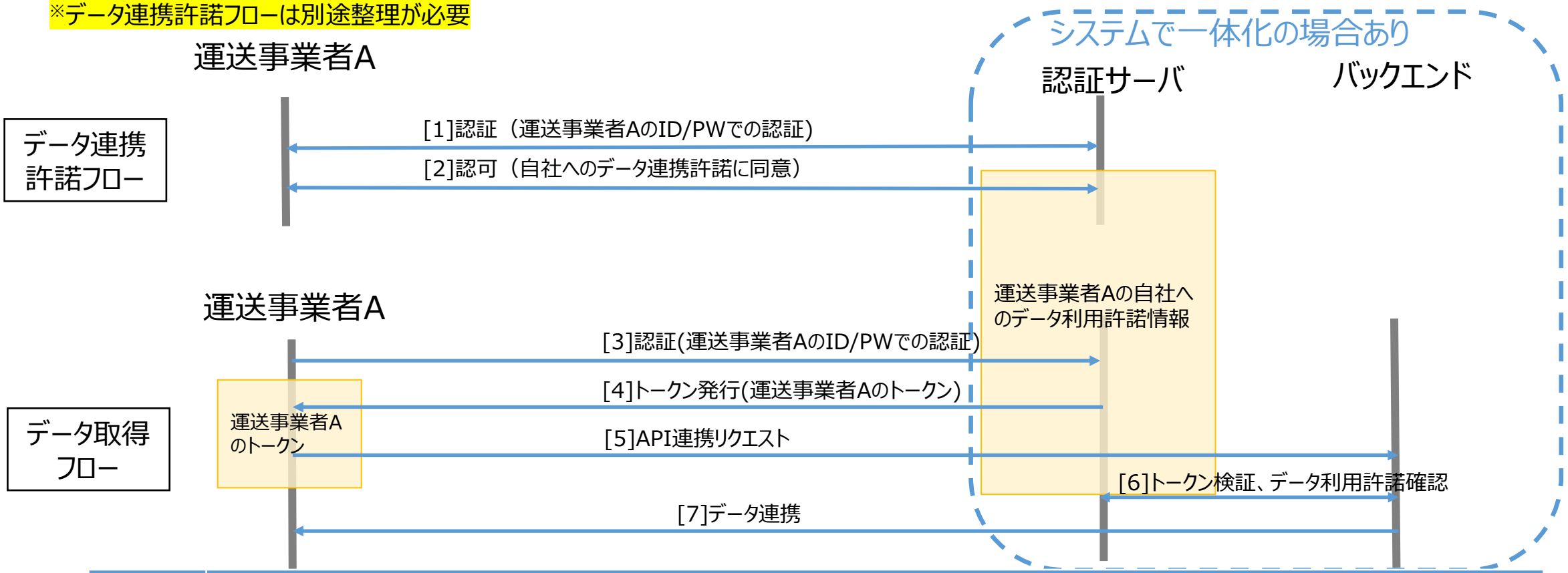
認証・認可	商用車メーカーによるAPI連携の認証認可	共通基盤によるAPI連携の認証認可
運送事業者（個社ごと）の連携		
運送事業者の会社間の連携（BがAの車両データを活用）		
運送事業者と他事業者間の連携（他事業者がAの車両データを活用）		

※データの利用許諾のフローについては別途整理要

※詳細項目については別途協議にて最終決定

トラックデータ標準APIガイドライン0.1版 – API連携フロー（パターン①）

※データ連携許諾フローは別途整理が必要

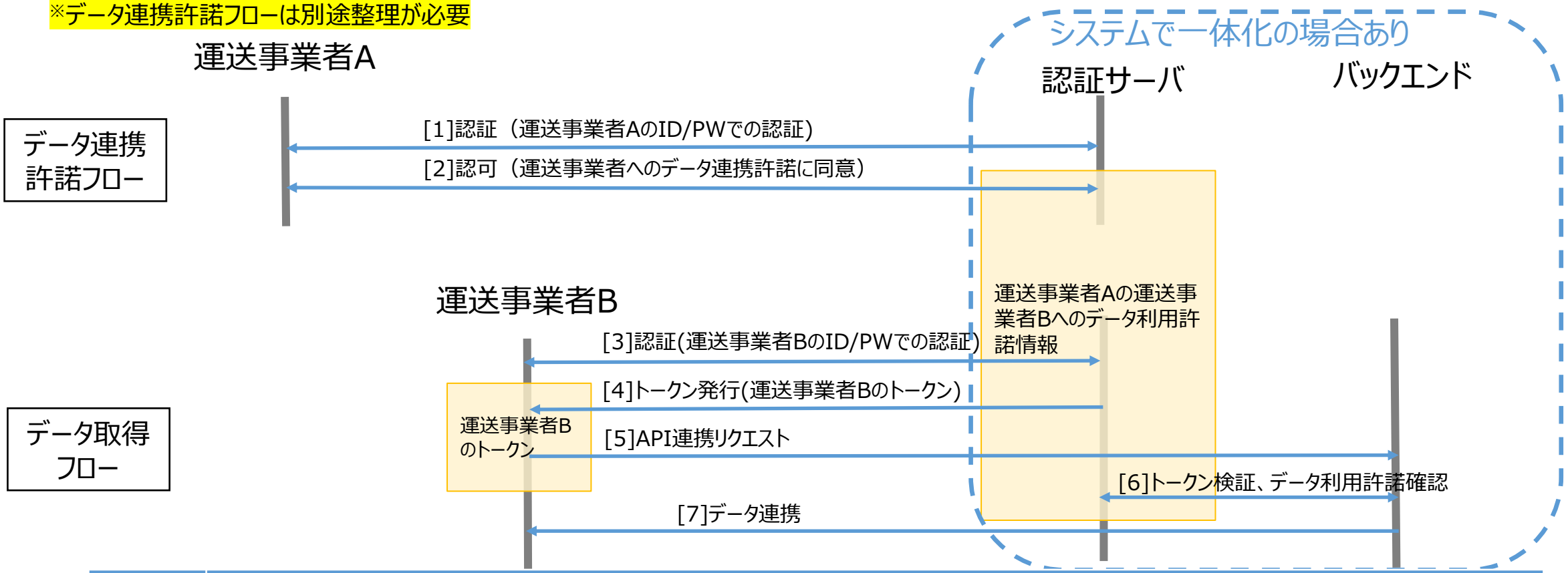


フローNo.	フロー詳細説明
1.	運送事業者Aが商用車メーカーの認証サーバ上でユーザID/PWで認証を行う
2.	運送事業者Aが自社へのデータ連携許諾に同意する
3.	運送事業者Aが商用車メーカーの認証サーバ上でユーザID/PWで認証を行う
4.	認証サーバが運送事業者Aへトークンを発行する
5.	運送事業者Aがトークンを付与してAPI連携リクエストを発出する
6.	API連携リクエストを受け付けたバックエンドサーバが認証サーバへトークン検証とデータ利用許諾の確認を実施する
7.	6.のトークン検証とデータ利用許諾の確認が取れた場合に、データを運送事業者Aへ連携する

※詳細項目については別途協議にて最終決定

トラックデータ標準APIガイドライン0.1版 – API連携フロー（パターン②）

※データ連携許諾フローは別途整理が必要

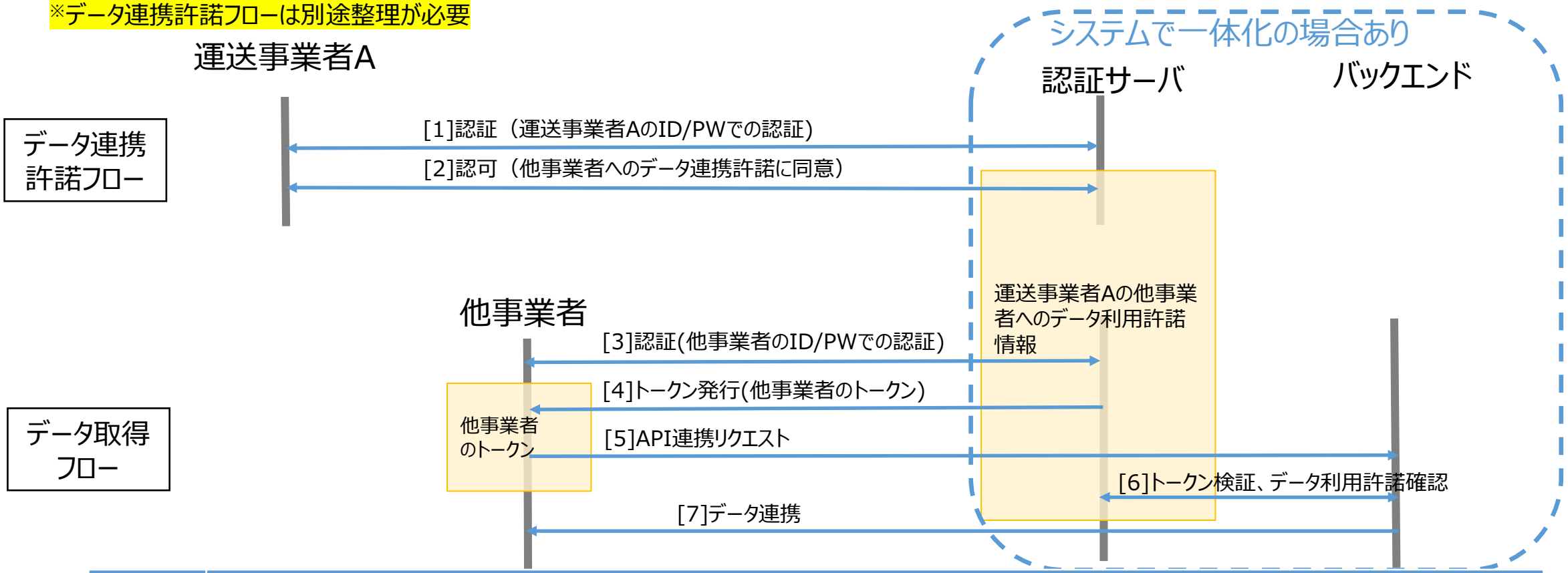


フローNo.	フロー詳細説明
1.	運送事業者Aが商用車メーカーの認証サーバ上でユーザID/PWで認証を行う
2.	運送事業者Aが運送事業者Bへのデータ連携許諾に同意する
3.	運送事業者Bが商用車メーカーの認証サーバ上でユーザID/PWで認証を行う
4.	認証サーバが運送事業者Bへトークンを発行する
5.	運送事業者Bがトークンを付与してAPI連携リクエストを発出する
6.	API連携リクエストを受け付けたバックエンドサーバが認証サーバへトークン検証とデータ利用許諾の確認を実施する
7.	6.のトークン検証とデータ利用許諾の確認が取れた場合に、データを運送事業者Bへ連携する

※詳細項目については別途協議にて最終決定

トラックデータ標準APIガイドライン0.1版 – API連携フロー（パターン③）

※データ連携許諾フローは別途整理が必要

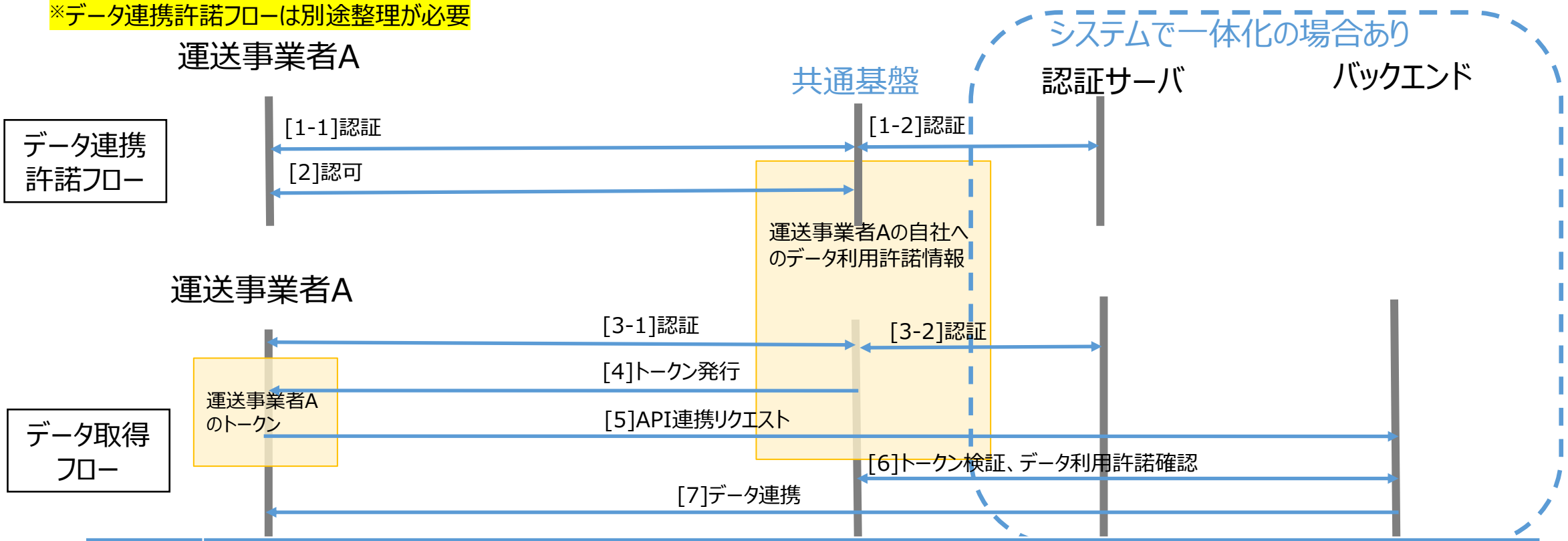


フローNo.	フロー詳細説明
1.	運送事業者Aが商用車メーカーの認証サーバ上でユーザID/PWで認証を行う
2.	運送事業者Aが他事業者へのデータ連携許諾に同意する
3.	他事業者が商用車メーカーの認証サーバ上でユーザID/PWで認証を行う
4.	認証サーバが他事業者へトークンを発行する
5.	他事業者がトークンを付与してAPI連携リクエストを発出する
6.	API連携リクエストを受け付けたバックエンドサーバが認証サーバへトークン検証とデータ利用許諾の確認を実施する
7.	6.のトークン検証とデータ利用許諾の確認が取れた場合に、データを他事業者へ連携する

※詳細項目については別途協議にて最終決定

トラックデータ標準APIガイドライン0.1版 – API連携フロー（パターン④）

※データ連携許諾フローは別途整理が必要

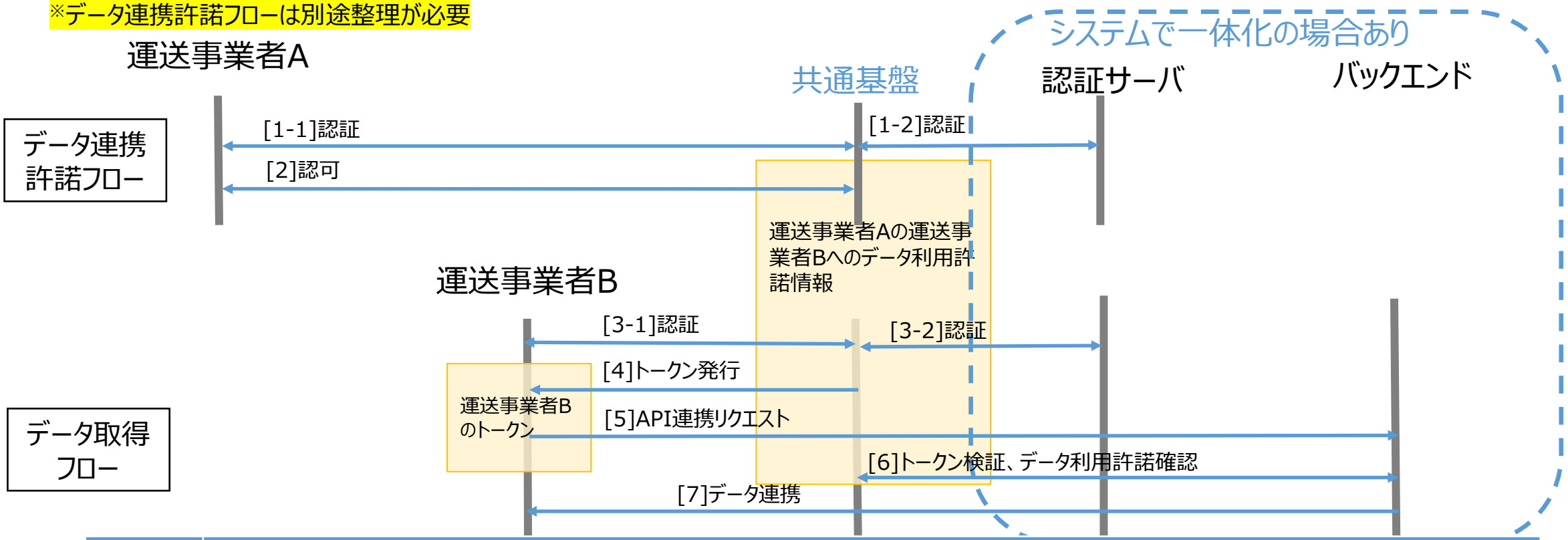


フローNo.	フロー詳細説明
1-1.	運送事業者Aが共通基盤上でユーザID/PWで認証を行う
1-2.	共通基盤と商用車メーカーの認証サーバ上で認証情報を連携する
2.	運送事業者Aが自社へのデータ連携許諾に同意する
3-1.	運送事業者Aが共通基盤上でユーザID/PWで認証を行う
3-1	共通基盤と商用車メーカーの認証サーバ上で認証情報を連携する
4.	共通基盤が運送事業者Aへトークンを発行する
5.	運送事業者Aがトークンを付与してAPI連携リクエストを発出する
6.	API連携リクエストを受け付けたバックエンドサーバが共通基盤へトークン検証とデータ利用許諾の確認を実施する
7.	6.のトークン検証とデータ利用許諾の確認が取れた場合に、データを運送事業者Aへ連携する

※詳細項目については別途協議にて最終決定

トラックデータ標準APIガイドライン0.1版 – API連携フロー（パターン⑤）

※データ連携許諾フローは別途整理が必要

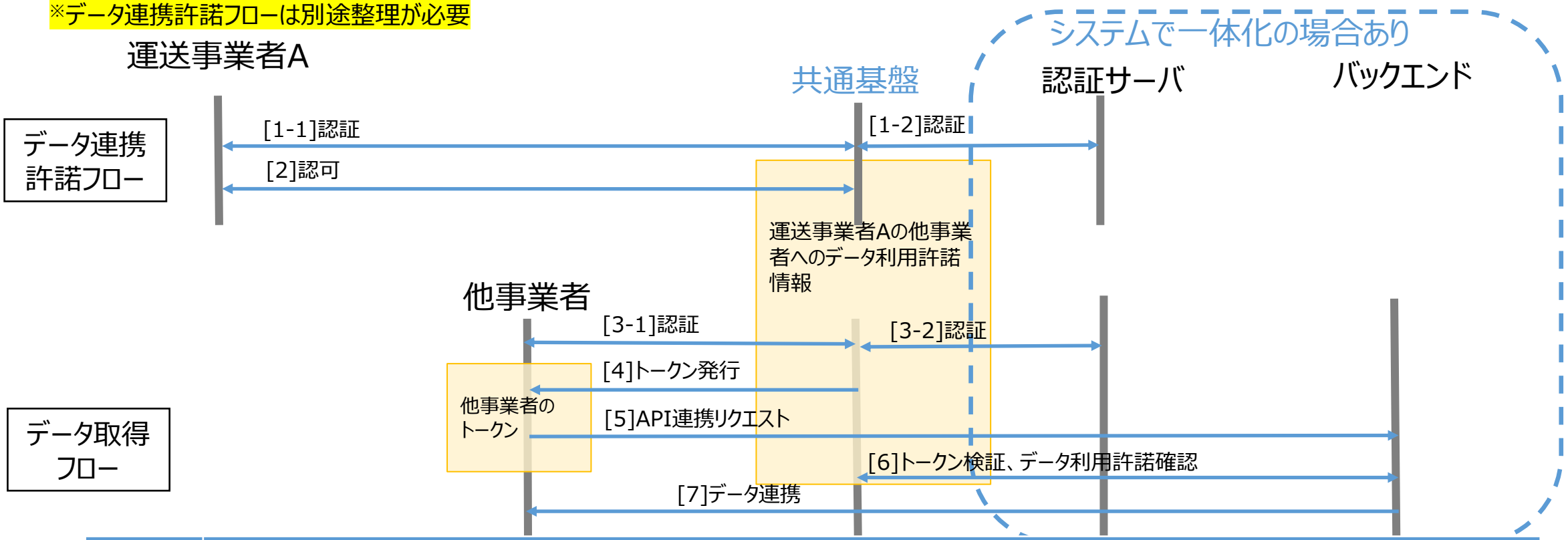


フローNo.	フロー詳細説明
1-1.	運送事業者Aが共通基盤上でユーザID/PWで認証を行う
1-2.	共通基盤と商用車メーカーの認証サーバ上で認証情報を連携する
2.	運送事業者Aが運送事業者Bへのデータ連携許諾に同意する
3-1.	運送事業者Bが共通基盤上でユーザID/PWで認証を行う
3-1	共通基盤と商用車メーカーの認証サーバ上で認証情報を連携する
4.	共通基盤が運送事業者Bへトークンを発行する
5.	運送事業者Bがトークンを付与してAPI連携リクエストを発出する
6.	API連携リクエストを受け付けたバックエンドサーバが共通基盤へトークン検証とデータ利用許諾の確認を実施する
7.	6.のトークン検証とデータ利用許諾の確認が取れた場合に、データを運送事業者Bへ連携する

※詳細項目については別途協議にて最終決定

トラックデータ標準APIガイドライン0.1版 – API連携フロー（パターン⑥）

※データ連携許諾フローは別途整理が必要



フローNo.	フロー詳細説明
1-1.	運送事業者Aが共通基盤上でユーザID/PWで認証を行う
1-2.	共通基盤と商用車メーカーの認証サーバ上で認証情報を連携する
2.	運送事業者Aが他事業者へのデータ連携許諾に同意する
3-1.	他事業者が共通基盤上でユーザID/PWで認証を行う
3-1	共通基盤と商用車メーカーの認証サーバ上で認証情報を連携する
4.	共通基盤が他事業者へトークンを発行する
5.	他事業者がトークンを付与してAPI連携リクエストを発出する
6.	API連携リクエストを受け付けたバックエンドサーバが共通基盤へトークン検証とデータ利用許諾の確認を実施する
7.	6.のトークン検証とデータ利用許諾の確認が取れた場合に、データを他事業者へ連携する