Guide pour le remplissage de la déclaration de conformité Jeux de données GBFS

Introduction

Ce document est la note d'accompagnement associée à la Déclaration de conformité relative à l'ouverture des données de l'information voyageur : il permet de préciser l'ensemble des informations demandées avec un focus particulier sur les éléments techniques pour la fourniture de données statiques et dynamiques en utilisant le format GBFS.

Il est ainsi à destination en particulier des producteurs de données de mobilité partagée. Les systèmes de mobilité partagés peuvent être de l'auto-partage, des vélos partagés, des trottinettes en libre service, etc. Ils sont classifiés sous l'appellation de "*Services à la demande et infrastructures de service à la demande*".

Le contenu de ce guide ne différencie en aucun cas le type de véhicule partagé. La seule différence faite entre les opérateurs est sur la base de leur structure opérationnelle, à savoir avec l'existence de stations ou non.

Guide de remplissage pour les systèmes avec stations

Pour l'ensemble des données listée dans la section "*FOURNITURE DES DONNÉES PAR L'INTERMÉDIAIRE DU POINT D'ACCES NATIONAL : transport.data.gouv.fr*", ce guide reprend les informations suivantes :

- type de données demandées;
- nom du fichier d'un jeu de données exprimé en GBFS;
- nom du(des) champ(s) dudit fichier correspondant au type de données, suivi de ses caractéristiques;
- un exemple.

Tous les champs marqués d'un astérisque (*) sont obligatoires.

À noter que le guide de conformité n'inclut pas de mentions aux données dites de "geofencing" qui permettent de définir des règles d'usage dans une zone de service donnée. Pour cela, le fichier geofencing_zones.json est utilisé. Il est recommandé de l'inclure dans tous les jeux de données en GBFS pour décrire finement les règles existantes des zones de services telles que les restrictions de parking, de vitesse, d'usage, etc.

Systèmes avec stations - Définition

Sont considérés comme des systèmes de mobilité partagée avec stations tout système pour lesquels existent des stations physiques installées dans la zone géographique couverte par l'opérateur. Ces stations doivent permettre à tout usager d'y commencer mais surtout terminer sa course. Pour se faire, l'usager n'a pas besoin de se connecter à l'application liée au service de mobilité et a uniquement besoin de remettre le véhicule loué dans l'un des emplacements de parking disponibles, c.à.d que l'action de verrouiller le véhicule sur une place de parking disponible à la station est automatiquement liée à la complétude d'un trajet.

Données statiques

<table>
<tr><td colspan="3">Type de données demandées :
• Recherche de lieu :
  ○ Identifiants d’adresse (numéro de bâtiment, rue, code postal)</td></tr>
<tr><td>Fichier GBFS</td><td colspan="2">station_information.json</td></tr>
<tr><td rowspan="4">Champs & caractéristiques</td><td>lat*</td><td>LATITUDE
Latitude of the station in decimal degrees. This field SHOULD have a precision of 6 decimal places (0.000001).</td></tr>
<tr><td>lon*</td><td>LONGITUDE
Longitude of the station in decimal degrees. This field SHOULD have a precision of 6 decimal places (0.000001).</td></tr>
<tr><td>address</td><td>STRING
Address (street number and name) where station is located. This MUST be a valid address, not a free-form text description. Example: 1234 Main Street</td></tr>
<tr><td>post_code</td><td>STRING
Postal code where station is located.</td></tr>
<tr><td>Exemple</td><td colspan="2"><pre>{
  "last_updated": 1609866247,
  "ttl": 0,
  "version": "2.2",
  "data": {
    "stations": [
      {
        "station_id": "abc",
        "name": "Parking ecole ABC",
        "lat": 12.345678,
        "lon": 45.678901,
        "address":"1 Place des Ecoles",</pre></td></tr>
</table>

	<pre>"post_code":12345,
	"vehicle_type_capacity": {
	"abc123": 7,
	"def456": 9
	}
	}
	]
	}
	}</pre>

Type de données demandées : • *Recherche de lieu :* ○ *Lieux topographiques (ville, localité, village, banlieue, unité administrative)*		
Fichier GBFS	station_information.json	
Champs & caractéristiques	region_id	ID Identifier of the region where station is located. See system_regions.json.

Exemple	<pre>{
	"last_updated": 1609866247,
	"ttl": 0,
	"version": "2.2",
	"data": {
	"stations": [
	{
	"station_id": "abc",
	"name": "Parking ecole ABC",
	"lat": 12.345678,
	"lon": 45.678901,
	"region_id":"region_12345",
	"vehicle_type_capacity": {</pre>	

	<pre>"abc123": 7,
	"def456": 9
	}
	}
	]
	}
	}</pre>

Type de données demandées : • *Recherche de lieu :* ◦ *Lieux intéressants (en relation avec les informations sur les transports), points de destination possible de voyageur*		
Fichier GBFS	station_information.json	
Champs & caractéristiques	cross_street	STRING Cross street or landmark where the station is located.

Exemple	<pre>{
	"last_updated": 1609866247,
	"ttl": 0,
	"version": "2.2",
	"data": {
	"stations": [
	{
	"station_id": "abc",
	"name": "Parking ecole ABC",
	"lat": 12.345678,
	"lon": 45.678901,
	"cross_street":"Ecole_123",
	"vehicle_type_capacity": {
	"abc123": 7,
	"def456": 9</pre>	

	<pre> }
	      }
	    ]
	  }
	}</pre>

Type de données demandées :
• *Recherche de lieux :*
  ○ *Stations de vélos partagés,*
  ○ *Stations de voitures partagées,*
  ○ *Stations d'autres engins de déplacement personnel partagés*		

Fichier GBFS	station_information.json	
Champs & caractéristiques	`station_id*`	ID Identifier of a station
	`name*`	STRING The public name of the station for display in maps, digital signage and other text applications. Names SHOULD reflect the station location through the use of a cross street or local landmark. Abbreviations SHOULD NOT be used for names and other text (e.g. St. for Street) unless a location is called by its abbreviated name (e.g. "JFK Airport"). Examples: * Broadway and East 22nd Street * Convention Center * Central Park South
	`short_name`	STRING Short name or other type of identifier.

	lat*	LATITUDE Latitude of the station in decimal degrees. This field SHOULD have a precision of 6 decimal places (0.000001).
	lon*	LONGITUDE Longitude of the station in decimal degrees. This field SHOULD have a precision of 6 decimal places (0.000001).
	address*	STRING Address (street number and name) where station is located. This MUST be a valid address, not a free-form text description. Example: 1234 Main Street
	post_code	STRING Postal code where station is located.
	region_id	ID Identifier of the region where station is located. See system_regions.json.
	cross_street	STRING Cross street or landmark where the station is located.
	is_virtual_station	BOOLEAN true - The station is a location without physical infrastructure, defined by a point (lat/lon) and/or station_area (below). **false - The station consists of physical infrastructure (docks).**
	capacity	NON-NEGATIVE INTEGER Number of total docking points installed at this station, both available and unavailable, regardless of what vehicle types are allowed at each dock.
	vehicle_type_capacity	OBJECT An object used to describe the docking capacity of a station where each key is a vehicle_type_id as described in vehicle_types.json and the value is a number representing the total docking points installed at this station, both available and unavailable for the specified vehicle type.

	is_valet_station	BOOLEAN Are valet services provided at this station? true - Valet services are provided at this station. false - Valet services are not provided at this station.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "stations": [
	  {
	  "station_id": "abc",
	  "name": "Parking ecole ABC",
	  "short_name": "Ecole ABC",
	  "lat": 12.345678,
	  "lon": 45.678901,
	  "address":"1 Place des Ecoles",
	  "cross_street":"Ecole_123",
	  "region_id":"region_12345",
	  "post_code":12345,
	  "is_valet_station":false,
	  "is_virtual_station":false,
	  "capacity":16,
	  "vehicle_type_capacity": {
	  "abc123": 7,
	  "def456": 9
	  }
	  }
	  ]
	  }
	}</pre>	

<table>
<tr><td colspan="3">Type de données demandées :
• Recherche de lieux :
○ Stations de vélos partagés,</td></tr>
<tr><td>Fichier GBFS</td><td colspan="2">vehicle_types.json</td></tr>
<tr><td>Champs & caractéristiques</td><td>form_factor*</td><td>ENUM
Must contain in the list at least of the following values:
- bicycle</td></tr>
<tr><td>Exemple</td><td colspan="2"><pre>{
  "last_updated": 1609866247,
  "ttl": 0,
  "version": "2.2",
  "data": {
    "vehicle_types": [
      {
        "vehicle_type_id": "abc123",
        "form_factor": "bicycle",
        "propulsion_type": "human",
        "name": "Example Basic Bike"
      },
    ]
  }
}</pre></td></tr>
</table>

Type de données demandées :

- *Recherche de lieux :*
  - *Stations de voitures partagées*

Fichier GBFS	vehicle_types.json	
Champs & caractéristiques	`form_factor*`	ENUM Must contain in the list at least of the following values: **- car**

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "vehicle_types": [
	  {
	  "vehicle_type_id": "car1",
	  "form_factor": "car",
	  "propulsion_type": "combustion",
	  "name": "Foor-door Sedan",
	  "max_range_meters": 523992
	  },
	  ]
	  }
	}</pre>	

<table>
<tr><td colspan="3">Type de données demandées :
• Recherche de lieux :
  ○ Stations d’autres engins de déplacement personnel partagés</td></tr>
<tr><td>Fichier GBFS</td><td colspan="2">vehicle_types.json</td></tr>
<tr><td>Champs & caractéristiques</td><td>form_factor*</td><td>ENUM
Must contain in the list at least of the following values:
- moped
- scooter
- other</td></tr>
<tr><td>Exemple</td><td colspan="2"><pre>{
  "last_updated": 1609866247,
  "ttl": 0,
  "version": "2.2",
  "data": {
    "vehicle_types": [
      {
        "vehicle_type_id": "def456",
        "form_factor": "scooter",
        "propulsion_type": "electric",
        "name": "Example E-scooter V2",
        "max_range_meters": 12345
      },
    ]
  }
}</pre></td></tr>
</table>

Type de données demandées : • *Itinéraires : calendrier opérationnel, reliant des types de journées à des dates :*		
Fichier GBFS	system_information.json	
Champs & caractéristiques	`timezone*`	TIMEZONE The time zone where the system is located.
Exemple	`{ "last_updated": 1609866247, "ttl": 0, "version": "2.2", "data":{ "system_id":"exemple_abc", "language":"fr", "name":"Exemple ABC", "timezone":"Europe/Paris", } }`	

Type de données demandées : • *Itinéraires : calendrier opérationnel, reliant des types de journées à des dates :*		
Fichier GBFS	system_calendar.json	
Champs & caractéristiques	`calendars*`	ARRAY Array of objects describing the system operational calendar. A minimum of one calendar object is REQUIRED. If start and end dates are the same every year, then start_year and end_year SHOULD be omitted.

	start_month*	NON-NEGATIVE INTEGER Starting month for the system operations (1-12).
	start_day*	NON-NEGATIVE INTEGER Starting month for the system operations (1-31).
	start_year	NON-NEGATIVE INTEGER Starting year for the system operations.
	end_month*	NON-NEGATIVE INTEGER Starting month for the system operations (1-12).
	end_day*	NON-NEGATIVE INTEGER Starting month for the system operations (1-31).
	end_year	NON-NEGATIVE INTEGER Ending year for the system operations.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data":{
	  "calendars":[
	  {
	  "start_month":4,
	  "start_day":1,
	  "start_year":2020,
	  "end_month":11,
	  "end_day":5,
	  "end_year":2020
	  }
	  ]
	  }</pre>	

	}

Type de données demandées : • *Itinéraires : calendrier opérationnel, reliant des types de journées à des dates :*		
Fichier GBFS	system_hours.json	
Champs & caractéristiques	rental_hours*	ARRAY Array of objects as defined below. The array MUST contain a minimum of one object identifying hours for every day of the week or a maximum of two for each day of the week objects (one for each user type).
	user_types*	ARRAY An array of member and/or nonmember value(s). This indicates that this set of rental hours applies to either members or non-members only.
	days*	ARRAY An array of abbreviations (first 3 letters) of English names of the days of the week for which this object applies (e.g. ["mon", "tue", "wed", "thu", "fri", "sat, "sun"]). Rental hours MUST NOT be defined more than once for each day and user type.
	start_time*	TIME Start time for the hours of operation of the system in the time zone indicated in system_information.json.
	end_time*	TIME End time for the hours of operation of the system in the time zone indicated in system_information.json.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",</pre>	

	<pre>{
	  "data": {
	  "rental_hours": [
	  {
	  "user_types": ["member"],
	  "days": ["mon", "tue", "wed", "thu", "fri", "sat", "sun"],
	  "start_time": "05:00:00",
	  "end_time": "18:29:59"
	  }
	  ]
	  }
	}</pre>

Type de données demandées :

- *Services d'information :*
  - *Lieux et modalités d'achat de billets pour les modes à la demande (y compris les canaux de détail, les méthodes d'exécution et les méthodes de paiement)*

Fichier GBFS	system_information.json	
Champs & caractéristiques	`purchase_url`	URL URL where a customer can purchase a membership.
	`android`	OBJECT Contains rental app download and app discovery information for the Android platform in the store_uri and discovery_uri fields.
	`store_uri`	URI URI where the rental Android app can be downloaded from. Typically this will be a URI to an app store such as Google Play. If the URI points to an app store such as Google Play, the URI SHOULD follow Android best practices so the viewing app can directly open the URI to the native app store app instead of a website. If a rental_uris.android field is populated then this field is REQUIRED, otherwise it is OPTIONAL.

		Example value: https://play.google.com/store/apps/details?id=com.abcrental.android
	discovery_uri	URI URI that can be used to discover if the rental Android app is installed on the device (e.g., using PackageManager.queryIntentActivities()). This intent is used by viewing apps to prioritize rental apps for a particular user based on whether they already have a particular rental app installed. This field is REQUIRED if a rental_uris.android field is populated, otherwise it is OPTIONAL. Example value: com.abcrental.android://
	ios	OBJECT Contains rental information for the iOS platform in the store_uri and discovery_uri fields.
	store_uri	URI URI where the rental iOS app can be downloaded from. Typically this will be a URI to an app store such as the Apple App Store. If the URI points to an app store such as the Apple App Store, the URI SHOULD follow iOS best practices so the viewing app can directly open the URI to the native app store app instead of a website. If a rental_uris.ios field is populated then this field is REQUIRED, otherwise it is OPTIONAL. Example value: https://apps.apple.com/app/apple-store/id123456789
	discovery_uri	URI URI that can be used to discover if the rental iOS app is installed on the device (e.g., using UIApplication canOpenURL:). This intent is used by viewing apps to prioritize rental apps for a particular user based on whether they already have a particular rental app installed. This field is REQUIRED if a rental_uris.ios field is populated, otherwise it is OPTIONAL. Example value: com.abcrental.ios://

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data":{
	  "rental_apps": {</pre>	

	<pre>"android": {
	"store_uri": "https://play.google.com/store/apps/details?id=com.abc",
	"discovery_uri": "exemple://"
	},
	"ios": {
	"store_uri": "https://apps.apple.com/fr/app/exemple/id123",
	"discovery_uri": "exemple://"
	}
	},
	"system_id":"exemple_abc",
	"language":"fr",
	"name":"Exemple ABC",
	"purchase_url":"https://www.exempleabc.fr",
	}
	}</pre>

Type de données demandées : • *Services d'information :* ◦ *Lieux et modalités d'achat de billets pour les modes à la demande (y compris les canaux de détail, les méthodes d'exécution et les méthodes de paiement)*		
Fichier GBFS	station_information.json	
Champs & caractéristiques	`rental_methods`	ARRAY Payment methods accepted at this station. Current valid values are: - key (e.g. operator issued vehicle key / fob / card) - creditcard - paypass - applepay - androidpay

		- transitcard - accountnumber - phone

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "stations": [
	  {
	  "station_id": "abc",
	  "name": "Parking ecole ABC",
	  "lat": 12.345678,
	  "lon": 45.678901,
	  "address":"1 Place des Ecoles",
	  "rental_methods": [
	  "KEY",
	  "APPLEPAY",
	  "ANDROIDPAY",
	  "TRANSITCARD",
	  "ACCOUNTNUMBER",
	  "PHONE"
	  ]
	  }
	  ]
	  }
	}</pre>	

Type de données demandées :

- *Services d'information :*
  - *Lieux et modalités d'achat de billets pour les modes à la demande (y compris les canaux de détail, les méthodes d'exécution et les*

méthodes de paiement)		
Fichier GBFS	free_bike_status.json	
Champs & caractéristiques	`pricing_plan_id`	ID The plan_id of the pricing plan this vehicle is eligible for as described in system_pricing_plans.json.
Exemple	`{ "last_updated":1609866247, "ttl":0, "version":"2.2", "data":{ "bikes":[ { "bike_id":"jkl012", "last_reported":1609866204, "is_reserved":false, "is_disabled":false, "vehicle_type_id":"def456", "current_range_meters":6543, "station_id":86, "pricing_plan_id":"plan3" } ] } }`	

Type de données demandées :

- *Services d'information :*
  - *Lieux et modalités d'achat de billets pour les modes à la demande (y compris les canaux de détail, les méthodes d'exécution et les méthodes de paiement)*

Fichier GBFS	system_pricing_plans.json	
Champs & caractéristiques	`plans*`	ARRAY Array of objects as defined below.
	`plan_id*`	ID Identifier for a pricing plan in the system.
	`url`	URL URL where the customer can learn more about this pricing plan.
	`name*`	STRING Name of this pricing plan.
	`currency*`	STRING Currency used to pay the fare. This pricing is in ISO 4217 code: http://en.wikipedia.org/wiki/ISO_4217 (e.g. CAD for Canadian dollars, EUR for euros, or JPY for Japanese yen.)
	`price*`	NON-NEGATIVE FLOAT OR STRING Fare price, in the unit specified by currency. If string, MUST be in decimal monetary value. (added in v2.2) Note: v3.0 may only allow non-negative float, therefore future implementations SHOULD be non-negative float. In case of non-rate price, this field is the total price. In case of rate price, this field is the base price that is charged only once per trip (e.g., price for unlocking) in addition to per_km_pricing and/or per_min_pricing.
	`is_taxable*`	BOOLEAN Will additional tax be added to the base price? true - Yes. false - No.
	`description*`	STRING

		Customer-readable description of the pricing plan. This SHOULD include the duration, price, conditions, etc. that the publisher would like users to see.
	per_km_pricing	ARRAY Array of segments when the price is a function of distance travelled, displayed in kilometers. Total price is the addition of price and all segments in per_km_pricing and per_min_pricing. If this array is not provided, there are no variable prices based on distance.
	start*	NON-NEGATIVE INTEGER The kilometer at which this segment rate starts being charged (inclusive).
	rate*	FLOAT Rate that is charged for each kilometer interval after the start. Can be a negative number, which indicates that the traveller will receive a discount.
	interval*	NON-NEGATIVE INTEGER Interval in kilometers at which the rate of this segment is either reapplied indefinitely, or if defined, up until (but not including) end kilometer. An interval of 0 indicates the rate is only charged once.
	end	NON-NEGATIVE INTEGER The kilometer at which the rate will no longer apply (exclusive) e.g. if end is 20 the rate no longer applies at 20.00 km. If this field is empty, the price issued for this segment is charged until the trip ends, in addition to following segments.
	per_min_pricing	ARRAY Array of segments when the price is a function of time travelled, displayed in minutes. Total price is the addition of price and all segments in per_km_pricing and per_min_pricing. If this array is not provided, there are no variable prices based on time.
	start*	NON-NEGATIVE INTEGER

		The minute at which this segment rate starts being charged (inclusive).
	rate*	FLOAT Rate that is charged for each minute interval after the start. Can be a negative number, which indicates that the traveller will receive a discount.
	interval*	NON-NEGATIVE INTEGER Interval in minutes at which the rate of this segment is either reapplied indefinitely, or if defined, up until (but not including) end minute. An interval of 0 indicates the rate is only charged once.
	end	NON-NEGATIVE INTEGER The minute at which the rate will no longer apply (exclusive) e.g. if end is 20 the rate no longer applies after 19:59. If this field is empty, the price issued for this segment is charged until the trip ends, in addition to following segments.
	surge_pricing	BOOLEAN Is there currently an increase in price in response to increased demand in this pricing plan? If this field is empty, it means these is no surge pricing in effect. true - Surge pricing is in effect. false - Surge pricing is not in effect.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "plans": [
	  {
	  "plan_id": "plan1",
	  "name": "Tarif simple",</pre>	

```
        "currency": "EUR",
        "price": 3,
        "is_taxable": true,
        "description": "3EUR pour la location, 0.25EUR par km et 0.50EUR par minute.",
        "per_km_pricing": [
          {
            "start": 0,
            "rate": 0.25,
            "interval": 1
          }
        ],
        "per_min_pricing": [
          {
            "start": 0,
            "rate": 0.50,
            "interval": 1
          }
        ]
      }
    ]
  }
}
```

Type de données demandées : • *Services d'information :* ○ *Caractéristiques des véhicules, telles que les différentes classes et le wifi à bord*	
Fichier GBFS	N/A

Type de données demandées : • *Services d'information :* ○ *Paramètres nécessaires pour calculer un facteur environnemental, tels que le carbone par type de véhicule ou voyageur-kilomètre*	
Fichier GBFS	N/A

Type de données demandées : • *Services d'information :* ○ *Paramètres tels que la consommation de carburant nécessaire pour le calcul du coût*		
Fichier GBFS	vehicle_types.json	
Champs & caractéristiques	`propulsion_type*`	ENUM The primary propulsion type of the vehicle. Current valid values are: - human (Pedal or foot propulsion) - electric_assist (Provides power only alongside human propulsion) - electric (Contains throttle mode with a battery-powered motor) **- combustion (Contains throttle mode with a gas engine-powered motor)**

	max_range_meters*	NON-NEGATIVE FLOAT If the vehicle has a motor (as indicated by having a value other than human in the propulsion_type field), this field is REQUIRED. This represents the furthest distance in meters that the vehicle can travel without recharging or refueling when it has the maximum amount of energy potential (for example, a full battery or full tank of gas).

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "vehicle_types": [
	  {
	  "vehicle_type_id": "car1",
	  "form_factor": "car",
	  "propulsion_type": "combustion",
	  "name": "Sedan 4_portes",
	  "max_range_meters": 523992
	  }
	  ]
	  }
	}</pre>	

Type de données demandées : • *Services d'information :* ○ *Modalités de réservation d'une voiture partagée, de taxis, de location de vélos, etc. (notamment les circuits de distribution, les méthodes d'exécution, les méthodes de paiement)*		
Fichier GBFS	station_information.json	
Champs &	rental_methods	ARRAY

caractéristiques		Payment methods accepted at this station. Current valid values are: - key (e.g. operator issued vehicle key / fob / card) - creditcard - paypass - applepay - androidpay - transitcard - accountnumber - phone
	rental_uris	OBJECT Contains rental URIs for Android, iOS, and web in the android, ios, and web fields.
	android	URI URI that can be passed to an Android app with an android.intent.action.VIEW Android intent to support Android Deep Links (https://developer.android.com/training/app-links/deep-linking). Please use Android App Links (https://developer.android.com/training/app-links) if possible so viewing apps don’t need to manually manage the redirect of the user to the app store if the user doesn’t have the application installed. This URI SHOULD be a deep link specific to this station, and SHOULD NOT be a general rental page that includes information for more than one station. The deep link SHOULD take users directly to this station, without any prompts, interstitial pages, or logins. Make sure that users can see this station even if they never previously opened the application. If this field is empty, it means deep linking isn’t supported in the native Android rental app. Note that URIs do not necessarily include the station_id for this station - other identifiers can be used by the rental app within the URI to uniquely identify this station. Android App Links example value: https://www.abc.com/app?sid=1234567890&platform=android
	ios	URI URI that can be used on iOS to launch the rental app for this station. More information on this iOS feature can be found here. Please use iOS Universal Links (https://developer.apple.com/ios/universal-links/) if possible so

		viewing apps don’t need to manually manage the redirect of the user to the app store if the user doesn’t have the application installed. This URI SHOULD be a deep link specific to this station, and SHOULD NOT be a general rental page that includes information for more than one station. The deep link SHOULD take users directly to this station, without any prompts, interstitial pages, or logins. Make sure that users can see this station even if they never previously opened the application. If this field is empty, it means deep linking isn’t supported in the native iOS rental app. Note that the URI does not necessarily include the station_id - other identifiers can be used by the rental app within the URL to uniquely identify this station. iOS Universal Links example value: https://www.abc.com/app?sid=1234567890&platform=ios
	web	URL URL that can be used by a web browser to show more information about renting a vehicle at this station. This URL SHOULD be a deep link specific to this station, and SHOULD NOT be a general rental page that includes information for more than one station. The deep link SHOULD take users directly to this station, without any prompts, interstitial pages, or logins. Make sure that users can see this station even if they never previously opened the application. If this field is empty, it means deep linking isn’t supported for web browsers. Example value: https://www.abc.com/app?sid=1234567890

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "stations": [
	  {
	  "station_id": "abc",
	  "name": "Parking ecole ABC",
	  "lat": 12.345678,
	  "lon": 45.678901,
	  "rental_uris": {
	  "ios":</pre>	

	"https://www.exempleabc.fr/applink?system_id=exemple_abc&station_id=abc&platform=iOS", "android": "https://www.exempleabc.fr/applink?system_id=exemple_abc&station_id=abc&platform=android" "web": “https://www.exempleabc.fr/app?sid=1234567890” }, "address":"1 Place des Ecoles", "rental_methods": ["KEY", "APPLEPAY", "ANDROIDPAY", "TRANSITCARD",] }] } }

Type de données demandées :

- *Services d’information :*
  - *Modalités de réservation d'une voiture partagée, de taxis, de location de vélos, etc. (notamment les circuits de distribution, les méthodes d'exécution, les méthodes de paiement)*

Fichier GBFS	free_bike_status.json	
Champs & caractéristiques	`rental_uris`	OBJECT JSON object that contains rental URIs for Android, iOS, and web in the android, ios, and web fields. See examples of how to use these fields and supported analytics.
	`android`	URI URI that can be passed to an Android app with an android.intent.action.VIEW Android intent to support Android Deep Links (https://developer.android.com/training/app-links/deep-linking). Please use Android App Links

		(https://developer.android.com/training/app-links) if possible, so viewing apps do not need to manually manage the redirect of the user to the app store if the user does not have the application installed. This URI SHOULD be a deep link specific to this vehicle, and SHOULD NOT be a general rental page that includes information for more than one vehicle. The deep link SHOULD take users directly to this vehicle, without any prompts, interstitial pages, or logins. Make sure that users can see this vehicle even if they never previously opened the application. Note that as a best practice providers SHOULD rotate identifiers within deep links after each rental to avoid unintentionally exposing private vehicle trip origins and destinations. If this field is empty, it means deep linking is not supported in the native Android rental app. Android App Links example value: https://www.abc.com/app?sid=1234567890&platform=android
	ios	URI URI that can be used on iOS to launch the rental app for this vehicle. More information on this iOS feature can be found here. Please use iOS Universal Links (https://developer.apple.com/ios/universal-links/) if possible, so viewing apps do not need to manually manage the redirect of the user to the app store if the user does not have the application installed. This URI SHOULD be a deep link specific to this vehicle, and SHOULD NOT be a general rental page that includes information for more than one vehicle. The deep link SHOULD take users directly to this vehicle, without any prompts, interstitial pages, or logins. Make sure that users can see this vehicle even if they never previously opened the application. Note that as a best practice providers SHOULD rotate identifiers within deep links after each rental to avoid unintentionally exposing private vehicle trip origins and destinations. If this field is empty, it means deep linking is not supported in the native iOS rental app. iOS Universal Links example value: https://www.abc.com/app?sid=1234567890&platform=ios
	web	URL URL that can be used by a web browser to show more information about renting a vehicle at this vehicle. This URL SHOULD be a deep link specific to this vehicle, and SHOULD NOT be a general rental page that includes information for more than one vehicle. The deep link SHOULD take users directly to this vehicle, without any prompts, interstitial pages, or logins. Make sure that users can see this vehicle even if they never previously opened the application. Note that as a best practice providers SHOULD rotate identifiers within deep links after each rental to avoid unintentionally exposing private vehicle trip origins and destinations. If this field is empty, it means deep linking isn’t supported for web browsers.

		Example value: https://www.abc.com/app?sid=1234567890

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl":0,
	  "version":"2.2",
	  "data":{
	  "bikes":[
	  {
	  "bike_id":"ghi789",
	  "last_reported":1609866109,
	  "lat":12.34,
	  "lon":56.78,
	  "is_reserved":false,
	  "is_disabled":false,
	  "vehicle_type_id":"abc123",
	  "rental_uris": {
	  "ios":
	"https://www.exempleabc.fr/applink?system_id=exemple_abc&station_id=abc&platform=iOS",
	  "android":
	"https://www.exempleabc.fr/applink?system_id=exemple_abc&station_id=abc&platform=android"
	  "web": “https://www.exempleabc.fr/app?sid=1234567890”
	  },
	  },
	  ]
	  }
	}</pre>	

Type de données demandées :

- *Calcul du plan de trajet :*
  - *Temps de trajet estimatifs par type de journée et zone horaire par mode de transport ou combinaison de mode de transport*

Fichier GBFS	N/A

Données dynamiques

Type de données demandées : • *Heures de passage, itinéraires et informations auxiliaires :* ○ *Perturbations*		
Fichier GBFS	system_alerts.json	
Champs & caractéristiques	`alerts*`	ARRAY Array of objects each indicating a system alert as defined below.
	`alert_id*`	ID Identifier for this alert.
	`type*`	ENUM Valid values are: - system_closure - station_closure - station_move - other
	`times`	ARRAY Array of objects with the fields start and end indicating when the alert is in effect (e.g. when the system or station is actually closed, or when it is scheduled to be moved).
	`start*`	TIMESTAMP Start time of the alert.

	end	TIMESTAMP End time of the alert. If there is currently no end time planned for the alert, this can be omitted.
	station_ids	ARRAY If this is an alert that affects one or more stations, include their ID(s). Otherwise omit this field. If both station_id and region_id are omitted, this alert affects the entire system.
	region_ids	ARRAY If this system has regions, and if this alert only affects certain regions, include their ID(s). Otherwise, omit this field. If both station_ids and region_ids are omitted, this alert affects the entire system.
	url	URL URL where the customer can learn more information about this alert.
	summary*	STRING A short summary of this alert to be displayed to the customer.
	description	STRING Detailed description of the alert.
	last_updated	TIMESTAMP Indicates the last time the info for the alert was updated.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl":0,
	  "version": "2.2",
	  "data":{
	  "alerts":[
	  {
	  "alert_id":"21",
	  "type":"station_closure",
	  "station_ids":[</pre>	

```
          "123",
          "456",
          "789"
        ],
        "times":[
          {
            "start":"1604448000",
            "end":"1604674800"
          }
        ],
        "url":"https://exempleabc.fr/more-info",
        "summary":"Interruption du service",
        "description":"Ces trois stations seront indisponibles entre le
          3 novembre minuit et le 6 novembre 15h pour travaux sur la voie publique",
        "last_updated":1604519393
      }
    ]
  }
}
```


Type de données demandées : • *Heures de passage, itinéraires et informations auxiliaires :* ○ *Information sur la situation en temps réel : retards, annulations, surveillance des correspondances garanties*	
Fichier GBFS	N/A

Type de données demandées : • *Heures de passage, itinéraires et informations auxiliaires :*   ○ *Heures de départ et d'arrivée estimatives*

Fichier GBFS	free_bike_status.json	
Champs & caractéristiques	`current_range_meters`	NON-NEGATIVE INTEGER If the corresponding vehicle_type definition for this vehicle has a motor, then this field is REQUIRED. This value represents the furthest distance in meters that the vehicle can travel without recharging or refueling with the vehicle's current charge or fuel.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl":0,
	  "version":"2.2",
	  "data":{
	  "bikes":[
	  {
	  "bike_id":"jkl012",
	  "last_reported":1609866204,
	  "is_reserved":false,
	  "is_disabled":false,
	  "vehicle_type_id":"def456",
	  "current_range_meters":6543,
	  "station_id":86,
	  "pricing_plan_id":"plan3"
	  }
	  ]
	  }
	}</pre>	

Type de données demandées :

- *Contrôle de disponibilité :*
  - *Disponibilité de voitures et de vélos partagés*

Fichier GBFS	station_status.json	
Champs & caractéristiques	station_id*	ID Identifier of a station
	num_bikes_available*	NON-NEGATIVE INTEGER Number of functional vehicles physically at the station that may be offered for rental. To know if the vehicles are available for rental, see is_renting. If is_renting = true this is the number of vehicles that are currently available for rent. If is_renting =false this is the number of vehicles that would be available for rent if the station were set to allow rentals.
	vehicle_types_available	ARRAY This field is REQUIRED if the vehicle_types.json file has been defined. This field's value is an array of objects. Each of these objects is used to model the total number of each defined vehicle type available at a station. The total number of vehicles from each of these objects SHOULD add up to match the value specified in the num_bikes_available field.
	vehicle_type_id*	ID The vehicle_type_id of each vehicle type at the station as described in vehicle_types.json. This field is REQUIRED if the vehicle_types.json is defined.
	count*	NON-NEGATIVE INTEGER A number representing the total number of available vehicles of the corresponding vehicle_type_id as defined in vehicle_types.json at the station.
	num_bikes_disabled	NON-NEGATIVE INTEGER Number of disabled vehicles of any type at the station. Vendors who do not want to publicize the number of disabled vehicles or docks in their system can opt to omit station capacity (in station_information.json, num_bikes_disabled, and num_docks_disabled (as of v2.0). If station capacity is published, then broken docks/vehicles can be inferred (though not specifically whether the decreased capacity is a broken vehicle or dock).

	num_docks_available*	NON-NEGATIVE INTEGER REQUIRED except for stations that have unlimited docking capacity (e.g. virtual stations) (as of v2.0). Number of functional docks physically at the station that are able to accept vehicles for return. To know if the docks are accepting vehicle returns, see is_returning. If is_returning = true this is the number of docks that are currently available to accept vehicle returns. If is_returning = false this is the number of docks that would be available if the station were set to allow returns.
	vehicle_docks_available	ARRAY This field is REQUIRED in feeds where the vehicle_types.json is defined and where certain docks are only able to accept certain vehicle types. If every dock at the station is able to accept any vehicle type, then this field is not REQUIRED. This field's value is an array of objects. Each of these objects is used to model the number of docks available for certain vehicle types. The total number of docks from each of these objects SHOULD add up to match the value specified in the num_docks_available field.
	vehicle_type_ids*	ID An array of strings where each string represents a vehicle_type_id that is able to use a particular type of dock at the station
	count*	NON-NEGATIVE INTEGER A number representing the total number of available vehicles of the corresponding vehicle type as defined in the vehicle_types array at the station that can accept vehicles of the specified types in the vehicle_types array.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "stations": [
	  {
	  "station_id": "station 1",
	  "is_installed": true,
	  "is_renting": true,
	  "is_returning": true,</pre>	

```
        "last_reported": 1609866125,
        "num_docks_available": 3,
        "vehicle_docks_available": [{
          "vehicle_type_ids": ["abc123"],
          "count": 2
        }, {
          "vehicle_type_ids": ["def456"],
          "count": 1
        }],
        "num_bikes_available": 1,
        "vehicle_types_available": [{
          "vehicle_type_id": "abc123",
          "count": 1
        }, {
          "vehicle_type_id": "def456",
          "count": 0
        }]
      }
    ]
  }
}
```


Type de données demandées :

- *Contrôle de disponibilité :*
  - *Disponibilité de voitures et de vélos partagés*

Fichier GBFS	free_bike_status.json	
Champs & caractéristiques	`bike_id*`	ID Identifier of a vehicle. The bike_id identifier MUST be rotated to a random string after each trip to protect user privacy (as of v2.0). Use of persistent vehicle IDs poses a threat to user privacy. The bike_id identifier SHOULD only be rotated once per trip.

	is_reserved*	BOOLEAN Is the vehicle currently reserved? true - Vehicle is currently reserved. false - Vehicle is not currently reserved.
	is_disabled*	BOOLEAN Is the vehicle currently disabled? true - Vehicle is currently disabled. false - Vehicle is not currently disabled. This field is used to indicate vehicles that are in the field but not available for rental. This may be due to a mechanical issue, low battery etc. Publishing this data may prevent users from attempting to rent vehicles that are disabled and not available for rental.
	vehicle_type_id	ID The vehicle_type_id of this vehicle as described in vehicle_types.json. This field is REQUIRED if the vehicle_types.json is defined.
	last_reported	TIMESTAMP The last time this vehicle reported its status to the operator's backend.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl":0,
	  "version":"2.2",
	  "data":{
	  "bikes":[
	  {
	  "bike_id":"ghi789",
	  "last_reported":1609866109,
	  "lat":12.34,
	  "lon":56.78,
	  "is_reserved":false,
	  "is_disabled":false,</pre>	

<table>
<tr><td></td><td><pre>
        "vehicle_type_id":"abc123"
      },
    ]
  }
}
</pre></td></tr>
</table>

<table>
<tr><td colspan="3">Type de données demandées :
• Localisation des véhicules, cycles et engins de déplacement personnel disponibles :</td></tr>
<tr><td>Fichier GBFS</td><td colspan="2">station_status.json</td></tr>
<tr><td rowspan="3">Champs & caractéristiques</td><td>is_installed*</td><td>BOOLEAN
Is the station currently on the street?
true - Station is installed on the street.
false - Station is not installed on the street.
Boolean SHOULD be set to true when equipment is present on the street. In seasonal systems where equipment is removed during winter, boolean SHOULD be set to false during the off season. May also be set to false to indicate planned (future) stations which have not yet been installed.</td></tr>
<tr><td>is_renting*</td><td>BOOLEAN
Is the station currently renting vehicles?
true - Station is renting vehicles. Even if the station is empty, if it would otherwise allow rentals, this value MUST be true.
false - Station is not renting vehicles.
If the station is temporarily taken out of service and not allowing rentals, this field MUST be set to false.
If a station becomes inaccessible to users due to road construction or other factors this field SHOULD be set to false. Field SHOULD be set to false during hours or days when the system is not offering vehicles for rent.</td></tr>
<tr><td>is_returning*</td><td>BOOLEAN
Is the station accepting vehicle returns?</td></tr>
</table>

		true - Station is accepting vehicle returns. Even if the station is full, if it would otherwise allow vehicle returns, this value MUST be true. false - Station is not accepting vehicle returns. If the station is temporarily taken out of service and not allowing vehicle returns, this field MUST be set to false. If a station becomes inaccessible to users due to road construction or other factors, this field SHOULD be set to false.
	`last_reported*`	TIMESTAMP The last time this station reported its status to the operator's backend.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "stations": [
	  {
	  "station_id": "station 1",
	  "is_installed": true,
	  "is_renting": true,
	  "is_returning": true,
	  "last_reported": 1609866125,
	  "num_docks_available": 3,
	  "vehicle_docks_available": [{
	  "vehicle_type_ids": ["abc123"],
	  "count": 2
	  }, {
	  "vehicle_type_ids": ["def456"],
	  "count": 1
	  }],
	  "num_bikes_available": 1,
	  "vehicle_types_available": [{
	  "vehicle_type_id": "abc123",</pre>	

<table>
<tr><td></td><td><pre>
          "count": 1
        }, {
          "vehicle_type_id": "def456",
          "count": 0
        }]
      }
    ]
  }
}
</pre></td></tr>
</table>

<table>
<tr><td colspan="3">Type de données demandées :
• Localisation des véhicules, cycles et engins de déplacement personnel disponibles :</td></tr>
<tr><td>Fichier GBFS</td><td colspan="2">free_bike_status.json</td></tr>
<tr><td>Champs & caractéristiques</td><td>station_id</td><td>ID
Identifier referencing the station_id field in system_information.json. REQUIRED only if the vehicle is currently at a station and the vehicle_types.json file has been defined.</td></tr>
<tr><td>Exemple</td><td colspan="2"><pre>
{
  "last_updated": 1609866247,
  "ttl":0,
  "version":"2.2",
  "data":{
    "bikes":[
      {
       "bike_id":"jkl012",
       "last_reported":1609866204,
       "is_reserved":false,
       "is_disabled":false,
       "vehicle_type_id":"def456",
</pre></td></tr>
</table>

	"current_range_meters":6543, "station_id":86, "pricing_plan_id":"plan3" },] } }

Guide de remplissage pour les systèmes sans stations (*free floating*)

Systèmes sans stations - Définition

Sont considérés comme des systèmes de mobilité partagée sans stations tout système de mobilité partagée se reposant sur une application mobile pour toute action de la part de l'usager (réservation d'un véhicule, usage, et cloture de la course). Que les véhicules soient soumis à des restrictions d'usage et/ou de parking (ex. stations virtuelles) ou non, ces systèmes ne peuvent être indépendants des données fournies par l'application de l'opérateur. Au contraire d'un système avec station, la complétude d'un trajet est obligatoirement liée à la confirmation manuelle de l'utilisateur.

À noter que les éléments ci-dessous reprennent certaines caractéristiques des systèmes avec stations pour prendre en compte l'existence de stations virtuelles (ex. emplacement dédié sur la voirie pour le parking) qui peuvent être définies dans les jeux de données GBFS.

Données statiques

Type de données demandées : ● *Recherche de lieu :* ○ *Identifiants d'adresse (numéro de bâtiment, rue, code postal)*	
Fichier GBFS	station_information.json

<table>
<tr><td rowspan="4">Champs & caractéristiques</td><td>lat*</td><td>LATITUDE
Latitude of the station in decimal degrees. This field SHOULD have a precision of 6 decimal places (0.000001).</td></tr>
<tr><td>lon*</td><td>LONGITUDE
Longitude of the station in decimal degrees. This field SHOULD have a precision of 6 decimal places (0.000001).</td></tr>
<tr><td>address</td><td>STRING
Address (street number and name) where station is located. This MUST be a valid address, not a free-form text description. Example: 1234 Main Street</td></tr>
<tr><td>post_code</td><td>STRING
Postal code where station is located.</td></tr>
<tr><td>Exemple</td><td colspan="2"><pre>{
  "last_updated": 1609866247,
  "ttl": 0,
  "version": "2.2",
  "data": {
    "stations": [
      {
        "station_id": "abc",
        "name": "Parking ecole ABC",
        "lat": 12.345678,
        "lon": 45.678901,
        "address":"1 Place des Ecoles",
        "post_code":12345,
        "vehicle_type_capacity": {
          "abc123": 7,
          "def456": 9
        }
      }
    ]
  }</pre></td></tr>
</table>

	}

Type de données demandées : • *Recherche de lieu :* ○ *Lieux topographiques (ville, localité, village, banlieue, unité administrative)*		
Fichier GBFS	station_information.json	
Champs & caractéristiques	region_id	ID Identifier of the region where station is located. See system_regions.json.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "stations": [
	  {
	  "station_id": "abc",
	  "name": "Parking ecole ABC",
	  "lat": 12.345678,
	  "lon": 45.678901,
	  "region_id":"region_12345",
	  "vehicle_type_capacity": {
	  "abc123": 7,
	  "def456": 9
	  }
	  }
	  ]
	  }
	}</pre>	

<table>
<tr><td colspan="3">Type de données demandées :
• Recherche de lieu :
  ○ Lieux intéressants (en relation avec les informations sur les transports), points de destination possible de voyageur</td></tr>
<tr><td>Fichier GBFS</td><td colspan="2">station_information.json</td></tr>
<tr><td>Champs & caractéristiques</td><td>cross_street</td><td>STRING
Cross street or landmark where the station is located.</td></tr>
<tr><td>Exemple</td><td colspan="2"><pre>{
  "last_updated": 1609866247,
  "ttl": 0,
  "version": "2.2",
  "data": {
    "stations": [
      {
        "station_id": "abc",
        "name": "Parking ecole ABC",
        "lat": 12.345678,
        "lon": 45.678901,
        "cross_street":"Ecole_123",
        "vehicle_type_capacity": {
          "abc123": 7,
          "def456": 9
        }
      }
    ]
  }
}</pre></td></tr>
</table>

Type de données demandées :
• *Recherche de lieux :*
  ○ *Stations de vélos partagés,*
  ○ *Stations de voitures partagées,*
  ○ *Stations d'autres engins de déplacement personnel partagés*		

Fichier GBFS	station_information.json	
Champs & caractéristiques	`station_id*`	ID Identifier of a station
	`name*`	STRING The public name of the station for display in maps, digital signage and other text applications. Names SHOULD reflect the station location through the use of a cross street or local landmark. Abbreviations SHOULD NOT be used for names and other text (e.g. St. for Street) unless a location is called by its abbreviated name (e.g. "JFK Airport"). Examples: * Broadway and East 22nd Street * Convention Center * Central Park South
	`short_name`	STRING Short name or other type of identifier.
	`lat*`	LATITUDE Latitude of the station in decimal degrees. This field SHOULD have a precision of 6 decimal places (0.000001).
	`lon*`	LONGITUDE Longitude of the station in decimal degrees. This field SHOULD have a precision of 6 decimal places (0.000001).

	address*	STRING Address (street number and name) where station is located. This MUST be a valid address, not a free-form text description. Example: 1234 Main Street
	post_code	STRING Postal code where station is located.
	region_id	ID Identifier of the region where station is located. See system_regions.json.
	cross_street	STRING Cross street or landmark where the station is located.
	is_virtual_station	BOOLEAN **true - The station is a location without physical infrastructure, defined by a point (lat/lon) and/or station_area (below).** false - The station consists of physical infrastructure (docks).
	station_area	"GeoJSON Multipolygon A GeoJSON multipolygon that describes the area of a virtual station. If station_area is supplied then the record describes a virtual station."
	capacity	NON-NEGATIVE INTEGER Number of total docking points installed at this station, both available and unavailable, regardless of what vehicle types are allowed at each dock.
	vehicle_capacity	OBJECT An object used to describe the parking capacity of virtual stations (defined using the is_virtual_station field), where each key is a vehicle_type_id as described in vehicle_types.json and the value is a number representing the total number of vehicles of this type that can park within the virtual station.
	vehicle_type_capacity	OBJECT An object used to describe the docking capacity of a station where each key is a vehicle_type_id as described in

		vehicle_types.json and the value is a number representing the total docking points installed at this station, both available and unavailable for the specified vehicle type.
	is_valet_station	BOOLEAN Are valet services provided at this station? true - Valet services are provided at this station. false - Valet services are not provided at this station.

Exemple	{
	"last_updated":1609866247,
	"ttl":0,
	"version":"2.2",
	"data":{
	"stations":[
	{
	"station_id":"station123",
	"name":"Station Ecole 123",
	"short_name":"Ecole 123",
	"is_valet_station":false,
	"is_virtual_station":true,
	"station_area":{
	"type":"MultiPolygon",
	"coordinates":[
	[
	[
	[
	-122.655775,
	45.516445
	],
	[
	-122.655705,
	45.516445
	],
	[	

```
                  -122.655705,
                  45.516495
                ],
                [
                  -122.655775,
                  45.516495
                ],
                [
                  -122.655775,
                  45.516445
                ]
              ]
            ]
          ]
        },
        "capacity":16,
        "vehicle_capacity":{
          "abc123":8,
          "def456":8,
          "ghi789":16
        }
      }
    ]
  }
}
```


Type de données demandées :

- *Recherche de lieux :*
  - *Stations de vélos partagés,*

Fichier GBFS	vehicle_types.json

Champs & caractéristiques	form_factor*	ENUM Must contain in the list at least of the following values: **- bicycle**

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "vehicle_types": [
	  {
	  "vehicle_type_id": "abc123",
	  "form_factor": "bicycle",
	  "propulsion_type": "human",
	  "name": "Example Basic Bike"
	  },
	  ]
	  }
	}</pre>	

Type de données demandées : • *Recherche de lieux :* ○ *Stations de voitures partagées*		
Fichier GBFS	vehicle_types.json	
Champs & caractéristiques	form_factor*	ENUM Must contain in the list at least of the following values: **- car**

Exemple	<pre>{
	  "last_updated": 1609866247,</pre>	

```
  "ttl": 0,
  "version": "2.2",
  "data": {
    "vehicle_types": [
      {
        "vehicle_type_id": "car1",
        "form_factor": "car",
        "propulsion_type": "combustion",
        "name": "Foor-door Sedan",
        "max_range_meters": 523992
      },
    ]
  }
}
```

<table>
<tr><td colspan="3">Type de données demandées :
• Recherche de lieux :
  ○ Stations d’autres engins de déplacement personnel partagés</td></tr>
<tr><td>Fichier GBFS</td><td colspan="2">vehicle_types.json</td></tr>
<tr><td>Champs & caractéristiques</td><td>form_factor*</td><td>ENUM
Must contain in the list at least of the following values:
- moped
- scooter
- other</td></tr>
<tr><td>Exemple</td><td colspan="2"><pre>{
  "last_updated": 1609866247,
  "ttl": 0,
  "version": "2.2",
  "data": {
    "vehicle_types": [
      {
        "vehicle_type_id": "def456",
        "form_factor": "scooter",
        "propulsion_type": "electric",
        "name": "Example E-scooter V2",
        "max_range_meters": 12345
      },
    ]
  }
}</pre></td></tr>
</table>

Type de données demandées : • *Itinéraires : calendrier opérationnel, reliant des types de journées à des dates :*		
Fichier GBFS	system_information.json	
Champs & caractéristiques	`timezone*`	TIMEZONE The time zone where the system is located.
Exemple	`{ "last_updated": 1609866247, "ttl": 0, "version": "2.2", "data":{ "system_id":"exemple_abc", "language":"fr", "name":"Exemple ABC", "timezone":"Europe/Paris", } }`	

Type de données demandées : • *Itinéraires : calendrier opérationnel, reliant des types de journées à des dates :*		
Fichier GBFS	system_calendar.json	
Champs & caractéristiques	`calendars*`	ARRAY Array of objects describing the system operational calendar. A minimum of one calendar object is REQUIRED. If start and end dates are the same every year, then start_year and end_year SHOULD be omitted.

	start_month*	NON-NEGATIVE INTEGER Starting month for the system operations (1-12).
	start_day*	NON-NEGATIVE INTEGER Starting month for the system operations (1-31).
	start_year	NON-NEGATIVE INTEGER Starting year for the system operations.
	end_month*	NON-NEGATIVE INTEGER Starting month for the system operations (1-12).
	end_day*	NON-NEGATIVE INTEGER Starting month for the system operations (1-31).
	end_year	NON-NEGATIVE INTEGER Ending year for the system operations.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data":{
	  "calendars":[
	  {
	  "start_month":4,
	  "start_day":1,
	  "start_year":2020,
	  "end_month":11,
	  "end_day":5,
	  "end_year":2020
	  }
	  ]
	  }</pre>	

	}

Type de données demandées : • *Itinéraires : calendrier opérationnel, reliant des types de journées à des dates :*		
Fichier GBFS	system_hours.json	
Champs & caractéristiques	rental_hours*	ARRAY Array of objects as defined below. The array MUST contain a minimum of one object identifying hours for every day of the week or a maximum of two for each day of the week objects (one for each user type).
	user_types*	ARRAY An array of member and/or nonmember value(s). This indicates that this set of rental hours applies to either members or non-members only.
	days*	ARRAY An array of abbreviations (first 3 letters) of English names of the days of the week for which this object applies (e.g. ["mon", "tue", "wed", "thu", "fri", "sat, "sun"]). Rental hours MUST NOT be defined more than once for each day and user type.
	start_time*	TIME Start time for the hours of operation of the system in the time zone indicated in system_information.json.
	end_time*	TIME End time for the hours of operation of the system in the time zone indicated in system_information.json.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",</pre>	

```
  "data": {
    "rental_hours": [
      {
        "user_types": [ "member"],
        "days": ["mon", "tue", "wed", "thu", "fri", "sat", "sun"],
        "start_time": "05:00:00",
        "end_time": "18:29:59"
      }
    ]
  }
}
```
 |

Type de données demandées :

- *Services d'information :*
  - *Lieux et modalités d'achat de billets pour les modes à la demande (y compris les canaux de détail, les méthodes d'exécution et les méthodes de paiement)*

Fichier GBFS	system_information.json	
Champs & caractéristiques	`purchase_url`	URL URL where a customer can purchase a membership.
	`android`	OBJECT Contains rental app download and app discovery information for the Android platform in the store_uri and discovery_uri fields.
	`store_uri`	URI URI where the rental Android app can be downloaded from. Typically this will be a URI to an app store such as Google Play. If the URI points to an app store such as Google Play, the URI SHOULD follow Android best practices so the viewing app can directly open the URI to the native app store app instead of a website. If a rental_uris.android field is populated then this field is REQUIRED, otherwise it is OPTIONAL.

		Example value: https://play.google.com/store/apps/details?id=com.abcrental.android
	discovery_uri	URI URI that can be used to discover if the rental Android app is installed on the device (e.g., using PackageManager.queryIntentActivities()). This intent is used by viewing apps to prioritize rental apps for a particular user based on whether they already have a particular rental app installed. This field is REQUIRED if a rental_uris.android field is populated, otherwise it is OPTIONAL. Example value: com.abcrental.android://
	ios	OBJECT Contains rental information for the iOS platform in the store_uri and discovery_uri fields.
	store_uri	URI URI where the rental iOS app can be downloaded from. Typically this will be a URI to an app store such as the Apple App Store. If the URI points to an app store such as the Apple App Store, the URI SHOULD follow iOS best practices so the viewing app can directly open the URI to the native app store app instead of a website. If a rental_uris.ios field is populated then this field is REQUIRED, otherwise it is OPTIONAL. Example value: https://apps.apple.com/app/apple-store/id123456789
	discovery_uri	URI URI that can be used to discover if the rental iOS app is installed on the device (e.g., using UIApplication canOpenURL:). This intent is used by viewing apps to prioritize rental apps for a particular user based on whether they already have a particular rental app installed. This field is REQUIRED if a rental_uris.ios field is populated, otherwise it is OPTIONAL. Example value: com.abcrental.ios://

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data":{
	  "rental_apps": {</pre>	

	"android": {
	"store_uri": "https://play.google.com/store/apps/details?id=com.abc",
	"discovery_uri": "exemple://"
	},
	"ios": {
	"store_uri": "https://apps.apple.com/fr/app/exemple/id123",
	"discovery_uri": "exemple://"
	}
	},
	"system_id":"exemple_abc",
	"language":"fr",
	"name":"Exemple ABC",
	"purchase_url":"https://www.exempleabc.fr",
	}
	}

Type de données demandées : • *Services d'information :* ◦ *Lieux et modalités d'achat de billets pour les modes à la demande (y compris les canaux de détail, les méthodes d'exécution et les méthodes de paiement)*		
Fichier GBFS	station_information.json	
Champs & caractéristiques	rental_methods	ARRAY Payment methods accepted at this station. Current valid values are: - key (e.g. operator issued vehicle key / fob / card) - creditcard - paypass - applepay - androidpay

		- transitcard - accountnumber - phone

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "stations": [
	  {
	  "station_id": "abc",
	  "name": "Parking ecole ABC",
	  "lat": 12.345678,
	  "lon": 45.678901,
	  "address":"1 Place des Ecoles",
	  "rental_methods": [
	  "KEY",
	  "APPLEPAY",
	  "ANDROIDPAY",
	  "TRANSITCARD",
	  "ACCOUNTNUMBER",
	  "PHONE"
	  ]
	  }
	  ]
	  }
	}</pre>	

Type de données demandées :

- *Services d'information :*
  - *Lieux et modalités d'achat de billets pour les modes à la demande (y compris les canaux de détail, les méthodes d'exécution et les*

méthodes de paiement)		
Fichier GBFS	free_bike_status.json	
Champs & caractéristiques	`pricing_plan_id`	ID The plan_id of the pricing plan this vehicle is eligible for as described in system_pricing_plans.json.
Exemple	`{ "last_updated":1609866247, "ttl":0, "version":"2.2", "data":{ "bikes":[ { "bike_id":"jkl012", "last_reported":1609866204, "is_reserved":false, "is_disabled":false, "vehicle_type_id":"def456", "current_range_meters":6543, "station_id":86, "pricing_plan_id":"plan3" } ] } }`	

Type de données demandées :

- *Services d'information :*
  - *Lieux et modalités d'achat de billets pour les modes à la demande (y compris les canaux de détail, les méthodes d'exécution et les méthodes de paiement)*

Fichier GBFS	system_pricing_plans.json	
Champs & caractéristiques	`plans*`	ARRAY Array of objects as defined below.
	`plan_id*`	ID Identifier for a pricing plan in the system.
	`url`	URL URL where the customer can learn more about this pricing plan.
	`name*`	STRING Name of this pricing plan.
	`currency*`	STRING Currency used to pay the fare. This pricing is in ISO 4217 code: http://en.wikipedia.org/wiki/ISO_4217 (e.g. CAD for Canadian dollars, EUR for euros, or JPY for Japanese yen.)
	`price*`	NON-NEGATIVE FLOAT OR STRING Fare price, in the unit specified by currency. If string, MUST be in decimal monetary value. (added in v2.2) Note: v3.0 may only allow non-negative float, therefore future implementations SHOULD be non-negative float. In case of non-rate price, this field is the total price. In case of rate price, this field is the base price that is charged only once per trip (e.g., price for unlocking) in addition to per_km_pricing and/or per_min_pricing.
	`is_taxable*`	BOOLEAN Will additional tax be added to the base price? true - Yes. false - No.
	`description*`	STRING

		Customer-readable description of the pricing plan. This SHOULD include the duration, price, conditions, etc. that the publisher would like users to see.
	per_km_pricing	ARRAY Array of segments when the price is a function of distance travelled, displayed in kilometers. Total price is the addition of price and all segments in per_km_pricing and per_min_pricing. If this array is not provided, there are no variable prices based on distance.
	start*	NON-NEGATIVE INTEGER The kilometer at which this segment rate starts being charged (inclusive).
	rate*	FLOAT Rate that is charged for each kilometer interval after the start. Can be a negative number, which indicates that the traveller will receive a discount.
	interval*	NON-NEGATIVE INTEGER Interval in kilometers at which the rate of this segment is either reapplied indefinitely, or if defined, up until (but not including) end kilometer. An interval of 0 indicates the rate is only charged once.
	end	NON-NEGATIVE INTEGER The kilometer at which the rate will no longer apply (exclusive) e.g. if end is 20 the rate no longer applies at 20.00 km. If this field is empty, the price issued for this segment is charged until the trip ends, in addition to following segments.
	per_min_pricing	ARRAY Array of segments when the price is a function of time travelled, displayed in minutes. Total price is the addition of price and all segments in per_km_pricing and per_min_pricing. If this array is not provided, there are no variable prices based on time.
	start*	NON-NEGATIVE INTEGER

		The minute at which this segment rate starts being charged (inclusive).
	rate*	FLOAT Rate that is charged for each minute interval after the start. Can be a negative number, which indicates that the traveller will receive a discount.
	interval*	NON-NEGATIVE INTEGER Interval in minutes at which the rate of this segment is either reapplied indefinitely, or if defined, up until (but not including) end minute. An interval of 0 indicates the rate is only charged once.
	end	NON-NEGATIVE INTEGER The minute at which the rate will no longer apply (exclusive) e.g. if end is 20 the rate no longer applies after 19:59. If this field is empty, the price issued for this segment is charged until the trip ends, in addition to following segments.
	surge_pricing	BOOLEAN Is there currently an increase in price in response to increased demand in this pricing plan? If this field is empty, it means these is no surge pricing in effect. true - Surge pricing is in effect. false - Surge pricing is not in effect.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "plans": [
	  {
	  "plan_id": "plan1",
	  "name": "Tarif simple",</pre>	

```
        "currency": "EUR",
        "price": 3,
        "is_taxable": true,
        "description": "3EUR pour la location, 0.25EUR par km et 0.50EUR par minute.",
        "per_km_pricing": [
          {
            "start": 0,
            "rate": 0.25,
            "interval": 1
          }
        ],
        "per_min_pricing": [
          {
            "start": 0,
            "rate": 0.50,
            "interval": 1
          }
        ]
      }
    ]
  }
}
```

Type de données demandées : • *Services d'information :* ○ *Caractéristiques des véhicules, telles que les différentes classes et le wifi à bord*	
Fichier GBFS	N/A

Type de données demandées : • *Services d'information :* ○ *Paramètres nécessaires pour calculer un facteur environnemental, tels que le carbone par type de véhicule ou voyageur-kilomètre*	
Fichier GBFS	N/A

Type de données demandées : • *Services d'information :* ○ *Paramètres tels que la consommation de carburant nécessaire pour le calcul du coût*		
Fichier GBFS	vehicle_types.json	
Champs & caractéristiques	`propulsion_type*`	ENUM The primary propulsion type of the vehicle. Current valid values are: - human (Pedal or foot propulsion) - electric_assist (Provides power only alongside human propulsion) - electric (Contains throttle mode with a battery-powered motor) **- combustion (Contains throttle mode with a gas engine-powered motor)**

	max_range_meters*	NON-NEGATIVE FLOAT If the vehicle has a motor (as indicated by having a value other than human in the propulsion_type field), this field is REQUIRED. This represents the furthest distance in meters that the vehicle can travel without recharging or refueling when it has the maximum amount of energy potential (for example, a full battery or full tank of gas).

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "vehicle_types": [
	  {
	  "vehicle_type_id": "car1",
	  "form_factor": "car",
	  "propulsion_type": "combustion",
	  "name": "Sedan 4_portes",
	  "max_range_meters": 523992
	  }
	  ]
	  }
	}</pre>	

Type de données demandées : • *Services d'information :* ○ *Modalités de réservation d'une voiture partagée, de taxis, de location de vélos, etc. (notamment les circuits de distribution, les méthodes d'exécution, les méthodes de paiement)*		
Fichier GBFS	station_information.json	
Champs &	rental_methods	ARRAY

caractéristiques		Payment methods accepted at this station. Current valid values are: - key (e.g. operator issued vehicle key / fob / card) - creditcard - paypass - applepay - androidpay - transitcard - accountnumber - phone
	rental_uris	OBJECT Contains rental URIs for Android, iOS, and web in the android, ios, and web fields.
	android	URI URI that can be passed to an Android app with an android.intent.action.VIEW Android intent to support Android Deep Links (https://developer.android.com/training/app-links/deep-linking). Please use Android App Links (https://developer.android.com/training/app-links) if possible so viewing apps don’t need to manually manage the redirect of the user to the app store if the user doesn’t have the application installed. This URI SHOULD be a deep link specific to this station, and SHOULD NOT be a general rental page that includes information for more than one station. The deep link SHOULD take users directly to this station, without any prompts, interstitial pages, or logins. Make sure that users can see this station even if they never previously opened the application. If this field is empty, it means deep linking isn’t supported in the native Android rental app. Note that URIs do not necessarily include the station_id for this station - other identifiers can be used by the rental app within the URI to uniquely identify this station. Android App Links example value: https://www.abc.com/app?sid=1234567890&platform=android
	ios	URI URI that can be used on iOS to launch the rental app for this station. More information on this iOS feature can be found here. Please use iOS Universal Links (https://developer.apple.com/ios/universal-links/) if possible so

		viewing apps don’t need to manually manage the redirect of the user to the app store if the user doesn’t have the application installed. This URI SHOULD be a deep link specific to this station, and SHOULD NOT be a general rental page that includes information for more than one station. The deep link SHOULD take users directly to this station, without any prompts, interstitial pages, or logins. Make sure that users can see this station even if they never previously opened the application. If this field is empty, it means deep linking isn’t supported in the native iOS rental app. Note that the URI does not necessarily include the station_id - other identifiers can be used by the rental app within the URL to uniquely identify this station. iOS Universal Links example value: https://www.abc.com/app?sid=1234567890&platform=ios
	web	URL URL that can be used by a web browser to show more information about renting a vehicle at this station. This URL SHOULD be a deep link specific to this station, and SHOULD NOT be a general rental page that includes information for more than one station. The deep link SHOULD take users directly to this station, without any prompts, interstitial pages, or logins. Make sure that users can see this station even if they never previously opened the application. If this field is empty, it means deep linking isn’t supported for web browsers. Example value: https://www.abc.com/app?sid=1234567890

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "stations": [
	  {
	  "station_id": "abc",
	  "name": "Parking ecole ABC",
	  "lat": 12.345678,
	  "lon": 45.678901,
	  "rental_uris": {
	  "ios":</pre>	

	"https://www.exempleabc.fr/applink?system_id=exemple_abc&station_id=abc&platform=iOS", "android": "https://www.exempleabc.fr/applink?system_id=exemple_abc&station_id=abc&platform=android" "web": “https://www.exempleabc.fr/app?sid=1234567890” }, "address":"1 Place des Ecoles", "rental_methods": ["KEY", "APPLEPAY", "ANDROIDPAY", "TRANSITCARD",] }] } }

Type de données demandées :

- *Services d’information :*
  - *Modalités de réservation d'une voiture partagée, de taxis, de location de vélos, etc. (notamment les circuits de distribution, les méthodes d'exécution, les méthodes de paiement)*

Fichier GBFS	free_bike_status.json	
Champs & caractéristiques	`rental_uris`	OBJECT JSON object that contains rental URIs for Android, iOS, and web in the android, ios, and web fields. See examples of how to use these fields and supported analytics.
	`android`	URI URI that can be passed to an Android app with an android.intent.action.VIEW Android intent to support Android Deep Links (https://developer.android.com/training/app-links/deep-linking). Please use Android App Links

		(https://developer.android.com/training/app-links) if possible, so viewing apps do not need to manually manage the redirect of the user to the app store if the user does not have the application installed. This URI SHOULD be a deep link specific to this vehicle, and SHOULD NOT be a general rental page that includes information for more than one vehicle. The deep link SHOULD take users directly to this vehicle, without any prompts, interstitial pages, or logins. Make sure that users can see this vehicle even if they never previously opened the application. Note that as a best practice providers SHOULD rotate identifiers within deep links after each rental to avoid unintentionally exposing private vehicle trip origins and destinations. If this field is empty, it means deep linking is not supported in the native Android rental app. Android App Links example value: https://www.abc.com/app?sid=1234567890&platform=android
	ios	URI URI that can be used on iOS to launch the rental app for this vehicle. More information on this iOS feature can be found here. Please use iOS Universal Links (https://developer.apple.com/ios/universal-links/) if possible, so viewing apps do not need to manually manage the redirect of the user to the app store if the user does not have the application installed. This URI SHOULD be a deep link specific to this vehicle, and SHOULD NOT be a general rental page that includes information for more than one vehicle. The deep link SHOULD take users directly to this vehicle, without any prompts, interstitial pages, or logins. Make sure that users can see this vehicle even if they never previously opened the application. Note that as a best practice providers SHOULD rotate identifiers within deep links after each rental to avoid unintentionally exposing private vehicle trip origins and destinations. If this field is empty, it means deep linking is not supported in the native iOS rental app. iOS Universal Links example value: https://www.abc.com/app?sid=1234567890&platform=ios
	web	URL URL that can be used by a web browser to show more information about renting a vehicle at this vehicle. This URL SHOULD be a deep link specific to this vehicle, and SHOULD NOT be a general rental page that includes information for more than one vehicle. The deep link SHOULD take users directly to this vehicle, without any prompts, interstitial pages, or logins. Make sure that users can see this vehicle even if they never previously opened the application. Note that as a best practice providers SHOULD rotate identifiers within deep links after each rental to avoid unintentionally exposing private vehicle trip origins and destinations. If this field is empty, it means deep linking isn’t supported for web browsers.

		Example value: https://www.abc.com/app?sid=1234567890

Exemple	{
	"last_updated": 1609866247,
	"ttl":0,
	"version":"2.2",
	"data":{
	"bikes":[
	{
	"bike_id":"ghi789",
	"last_reported":1609866109,
	"lat":12.34,
	"lon":56.78,
	"is_reserved":false,
	"is_disabled":false,
	"vehicle_type_id":"abc123",
	"rental_uris": {
	"ios":
	"https://www.exempleabc.fr/applink?system_id=exemple_abc&station_id=abc&platform=iOS",
	"android":
	"https://www.exempleabc.fr/applink?system_id=exemple_abc&station_id=abc&platform=android"
	"web": “https://www.exempleabc.fr/app?sid=1234567890”
	},
	},
	]
	}
	}	

Type de données demandées :

- *Calcul du plan de trajet :*
  - *Temps de trajet estimatifs par type de journée et zone horaire par mode de transport ou combinaison de mode de transport*

Fichier GBFS	N/A

Données dynamiques

Type de données demandées : • *Heures de passage, itinéraires et informations auxiliaires :* ○ *Perturbations*		
Fichier GBFS	system_alerts.json	
Champs & caractéristiques	`alerts*`	ARRAY Array of objects each indicating a system alert as defined below.
	`alert_id*`	ID Identifier for this alert.
	`type*`	ENUM Valid values are: - system_closure - station_closure - station_move - other
	`times`	ARRAY Array of objects with the fields start and end indicating when the alert is in effect (e.g. when the system or station is actually closed, or when it is scheduled to be moved).
	`start*`	TIMESTAMP Start time of the alert.

	end	TIMESTAMP End time of the alert. If there is currently no end time planned for the alert, this can be omitted.
	station_ids	ARRAY If this is an alert that affects one or more stations, include their ID(s). Otherwise omit this field. If both station_id and region_id are omitted, this alert affects the entire system.
	region_ids	ARRAY If this system has regions, and if this alert only affects certain regions, include their ID(s). Otherwise, omit this field. If both station_ids and region_ids are omitted, this alert affects the entire system.
	url	URL URL where the customer can learn more information about this alert.
	summary*	STRING A short summary of this alert to be displayed to the customer.
	description	STRING Detailed description of the alert.
	last_updated	TIMESTAMP Indicates the last time the info for the alert was updated.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl":0,
	  "version": "2.2",
	  "data":{
	  "alerts":[
	  {
	  "alert_id":"21",
	  "type":"station_closure",
	  "station_ids":[</pre>	

```
          "123",
          "456",
          "789"
        ],
        "times":[
          {
            "start":"1604448000",
            "end":"1604674800"
          }
        ],
        "url":"https://exempleabc.fr/more-info",
        "summary":"Interruption du service",
        "description":"Ces trois stations seront indisponibles entre le
          3 novembre minuit et le 6 novembre 15h pour travaux sur la voie publique",
        "last_updated":1604519393
      }
    ]
  }
}
```


Type de données demandées : • *Heures de passage, itinéraires et informations auxiliaires :* ○ *Information sur la situation en temps réel : retards, annulations, surveillance des correspondances garanties*	
Fichier GBFS	N/A

Type de données demandées : • *Heures de passage, itinéraires et informations auxiliaires :*   ○ *Heures de départ et d'arrivée estimatives*

Fichier GBFS	free_bike_status.json	
Champs & caractéristiques	`current_range_meters`	NON-NEGATIVE INTEGER If the corresponding vehicle_type definition for this vehicle has a motor, then this field is REQUIRED. This value represents the furthest distance in meters that the vehicle can travel without recharging or refueling with the vehicle's current charge or fuel.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl":0,
	  "version":"2.2",
	  "data":{
	  "bikes":[
	  {
	  "bike_id":"jkl012",
	  "last_reported":1609866204,
	  "is_reserved":false,
	  "is_disabled":false,
	  "vehicle_type_id":"def456",
	  "current_range_meters":6543,
	  "station_id":86,
	  "pricing_plan_id":"plan3"
	  }
	  ]
	  }
	}</pre>	

Type de données demandées :

- *Contrôle de disponibilité :*
  - *Disponibilité de voitures et de vélos partagés*

Fichier GBFS	station_status.json	
Champs & caractéristiques	`station_id*`	ID Identifier of a station
	`num_bikes_available*`	NON-NEGATIVE INTEGER Number of functional vehicles physically at the station that may be offered for rental. To know if the vehicles are available for rental, see is_renting. If is_renting = true this is the number of vehicles that are currently available for rent. If is_renting =false this is the number of vehicles that would be available for rent if the station were set to allow rentals.
	`vehicle_types_available`	ARRAY This field is REQUIRED if the vehicle_types.json file has been defined. This field's value is an array of objects. Each of these objects is used to model the total number of each defined vehicle type available at a station. The total number of vehicles from each of these objects SHOULD add up to match the value specified in the num_bikes_available field.
	`vehicle_type_id*`	ID The vehicle_type_id of each vehicle type at the station as described in vehicle_types.json. This field is REQUIRED if the vehicle_types.json is defined.
	`count*`	NON-NEGATIVE INTEGER A number representing the total number of available vehicles of the corresponding vehicle_type_id as defined in vehicle_types.json at the station.

	num_bikes_disabled	NON-NEGATIVE INTEGER Number of disabled vehicles of any type at the station. Vendors who do not want to publicize the number of disabled vehicles or docks in their system can opt to omit station capacity (in station_information.json, num_bikes_disabled, and num_docks_disabled (as of v2.0). If station capacity is published, then broken docks/vehicles can be inferred (though not specifically whether the decreased capacity is a broken vehicle or dock).
	num_docks_available	NON-NEGATIVE INTEGER REQUIRED except for stations that have unlimited docking capacity (e.g. virtual stations) (as of v2.0). Number of functional docks physically at the station that are able to accept vehicles for return. To know if the docks are accepting vehicle returns, see is_returning. If is_returning = true this is the number of docks that are currently available to accept vehicle returns. If is_returning = false this is the number of docks that would be available if the station were set to allow returns.
	vehicle_docks_available	ARRAY This field is REQUIRED in feeds where the vehicle_types.json is defined and where certain docks are only able to accept certain vehicle types. If every dock at the station is able to accept any vehicle type, then this field is not REQUIRED. This field's value is an array of objects. Each of these objects is used to model the number of docks available for certain vehicle types. The total number of docks from each of these objects SHOULD add up to match the value specified in the num_docks_available field.
	vehicle_type_ids*	ID An array of strings where each string represents a vehicle_type_id that is able to use a particular type of dock at the station
	count*	NON-NEGATIVE INTEGER

		A number representing the total number of available vehicles of the corresponding vehicle type as defined in the vehicle_types array at the station that can accept vehicles of the specified types in the vehicle_types array.

Exemple	{
	"last_updated": 1609866247,
	"ttl": 0,
	"version": "2.2",
	"data": {
	"stations": [
	{
	"station_id": "station 1",
	"is_installed": true,
	"is_renting": true,
	"is_returning": true,
	"last_reported": 1609866125,
	"num_docks_available": 3,
	"vehicle_docks_available": [{
	"vehicle_type_ids": ["abc123"],
	"count": 2
	}, {
	"vehicle_type_ids": ["def456"],
	"count": 1
	}],
	"num_bikes_available": 1,
	"vehicle_types_available": [{
	"vehicle_type_id": "abc123",
	"count": 1
	}, {
	"vehicle_type_id": "def456",
	"count": 0
	}]
	}
	]
	}	

	}

Type de données demandées : • *Contrôle de disponibilité :* ○ *Disponibilité de voitures et de vélos partagés*		
Fichier GBFS	free_bike_status.json	
Champs & caractéristiques	bike_id*	ID Identifier of a vehicle. The bike_id identifier MUST be rotated to a random string after each trip to protect user privacy (as of v2.0). Use of persistent vehicle IDs poses a threat to user privacy. The bike_id identifier SHOULD only be rotated once per trip.
	is_reserved*	BOOLEAN Is the vehicle currently reserved? true - Vehicle is currently reserved. false - Vehicle is not currently reserved.
	is_disabled*	BOOLEAN Is the vehicle currently disabled? true - Vehicle is currently disabled. false - Vehicle is not currently disabled. This field is used to indicate vehicles that are in the field but not available for rental. This may be due to a mechanical issue, low battery etc. Publishing this data may prevent users from attempting to rent vehicles that are disabled and not available for rental.
	vehicle_type_id	ID The vehicle_type_id of this vehicle as described in vehicle_types.json. This field is REQUIRED if the vehicle_types.json is defined.

	last_reported	TIMESTAMP The last time this vehicle reported its status to the operator's backend.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl":0,
	  "version":"2.2",
	  "data":{
	  "bikes":[
	  {
	  "bike_id":"ghi789",
	  "last_reported":1609866109,
	  "lat":12.34,
	  "lon":56.78,
	  "is_reserved":false,
	  "is_disabled":false,
	  "vehicle_type_id":"abc123"
	  },
	  ]
	  }
	}</pre>	

Type de données demandées : • *Localisation des véhicules, cycles et engins de déplacement personnel disponibles :*		
Fichier GBFS	station_status.json	
Champs & caractéristiques	`is_installed*`	BOOLEAN Is the station currently on the street? true - Station is installed on the street. false - Station is not installed on the street. Boolean SHOULD be set to true when equipment is present on the street. In seasonal systems where equipment is removed during winter, boolean SHOULD be set to false during the off season. May also be set to false to indicate planned (future) stations which have not yet been installed.
	`is_renting*`	BOOLEAN Is the station currently renting vehicles? true - Station is renting vehicles. Even if the station is empty, if it would otherwise allow rentals, this value MUST be true. false - Station is not renting vehicles. If the station is temporarily taken out of service and not allowing rentals, this field MUST be set to false. If a station becomes inaccessible to users due to road construction or other factors this field SHOULD be set to false. Field SHOULD be set to false during hours or days when the system is not offering vehicles for rent.
	`is_returning*`	BOOLEAN Is the station accepting vehicle returns? true - Station is accepting vehicle returns. Even if the station is full, if it would otherwise allow vehicle returns, this value MUST be true. false - Station is not accepting vehicle returns. If the station is temporarily taken out of service and not allowing vehicle returns, this field MUST be set to false. If a station becomes inaccessible to users due to road construction or other factors, this field SHOULD be set to

		false.
	last_reported*	TIMESTAMP The last time this station reported its status to the operator's backend.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "stations": [
	  {
	  "station_id": "station 1",
	  "is_installed": true,
	  "is_renting": true,
	  "is_returning": true,
	  "last_reported": 1609866125,
	  "num_docks_available": 3,
	  "vehicle_docks_available": [{
	  "vehicle_type_ids": ["abc123"],
	  "count": 2
	  }, {
	  "vehicle_type_ids": ["def456"],
	  "count": 1
	  }],
	  "num_bikes_available": 1,
	  "vehicle_types_available": [{
	  "vehicle_type_id": "abc123",
	  "count": 1
	  }, {
	  "vehicle_type_id": "def456",
	  "count": 0
	  }]
	  }</pre>	

	] } }

Type de données demandées : • *Localisation des véhicules, cycles et engins de déplacement personnel disponibles :*		
Fichier GBFS	free_bike_status.json	
Champs & caractéristiques	station_id	ID Identifier referencing the station_id field in system_information.json. REQUIRED only if the vehicle is currently at a station and the vehicle_types.json file has been defined.
	lat	LATITUDE Latitude of the vehicle in decimal degrees. (as of v2.1) This field is REQUIRED if station_id is not provided for this vehicle (free floating). This field SHOULD have a precision of 6 decimal places (0.000001).
	lon	LONGITUDE Longitude of the vehicle. (as of v2.1) This field is REQUIRED if station_id is not provided for this vehicle (free floating).

Exemple	<pre>{
	"last_updated": 1609866247,
	"ttl":0,
	"version":"2.2",
	"data":{
	"bikes":[
	{
	"bike_id":"ghi789",
	"last_reported":1609866109,
	"lat":12.34,</pre>	

	<pre>"lon":56.78,
	"is_reserved":false,
	"is_disabled":false,
	"vehicle_type_id":"abc123"
	},
	]
	}
	}</pre>

Guide de remplissage commun à tous les opérateurs

Formats et versions

Type de données demandées : • *format d'échange des données et la version du format le cas échéant*		
Fichier GBFS	gbfs.json	
Champs & caractéristiques	feeds*	ARRAY An array of all of the feeds that are published by this auto-discovery file. Each element in the array is an object with the keys below.
	name*	STRING Key identifying the type of feed this is. The key MUST be the base file name defined in the spec for the corresponding feed type (system_information for system_information.json file, station_information for station_information.json file).
	url*	URL URL for the feed. Note that the actual feed endpoints (urls) MAY NOT be defined in the file_name.json format. For example, a valid feed endpoint could end with station_info instead of station_information.json.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "fr" : {
	  "feeds": [
	  {
	  "name": "system_information",
	  "url": "https://www.example.com/gbfs/1/fr/system_information"
	  },
	  {
	  "name": "station_information",
	  "url": "https://www.example.com/gbfs/1/fr/station_information"
	  }
	  ]
	  }
	  }
	}</pre>

Type de données demandées : • *format d'échange des données et la version du format le cas échéant*		
Fichier GBFS	Tous les fichiers du flux de données	
Champs & caractéristiques	`version*`	GBFS version number to which the feed confirms, according to the versioning framework.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2"</pre>	

	}

Type de données demandées : • *format d'échange des données et la version du format le cas échéant*		
Fichier GBFS	gbfs_versions.json	
Champs & caractéristiques	versions*	ARRAY Contains one object, as defined below, for each of the available versions of a feed. The array MUST be sorted by increasing MAJOR and MINOR version number.
	version*	STRING The semantic version of the feed in the form X.Y.
	url*	URL URL of the corresponding gbfs.json endpoint.

Exemple	<pre>{
	  "last_updated": 1609866247,
	  "ttl": 0,
	  "version": "2.2",
	  "data": {
	  "versions": [
	  {
	  "version":"2.2",
	  "url":"https://www.example.com/gbfs/2-2/gbfs"
	  }
	  ]
	  }
	}</pre>	

<table>
<tr><td colspan="3">Type de données demandées :
• couverture géographique des données</td></tr>
<tr><td>Fichier GBFS</td><td colspan="2">system_regions.json</td></tr>
<tr><td rowspan="3">Champs & caractéristiques</td><td>regions*</td><td>ARRAY
Array of objects as defined below.</td></tr>
<tr><td>regions_id*</td><td>ID
Identifier for the region.</td></tr>
<tr><td>name*</td><td>STRING
Public name for this region.</td></tr>
<tr><td>Exemple</td><td colspan="2"><pre>{
  "last_updated": 1609866247,
  "ttl": 0,
  "version": "2.2",
  "data":{
    "regions":[
      {
        "name":"North",
        "region_id":"3"
      },
      {
        "name":"East",
        "region_id":"4"
      },
    ]
  }
}</pre></td></tr>
</table>

Type de données demandées : • *restrictions de couverture pour certaines catégories de données* • *identifiants des jeux de données publiés sur le point d'accès national*	
Fichier GBFS	N/A

Métadonnées

Type de données demandées :
• *Description des métadonnées*
  ○ *Entité publiant les données*
  ○ *Entité propriétaire des données*		

Fichier GBFS	system_information.json	
Champs & caractéristiques	`system_id*`	ID This is a globally unique identifier for the vehicle share system. It is up to the publisher of the feed to guarantee uniqueness and MUST be checked against existing system_id fields in systems.txt to ensure this. This value is intended to remain the same over the life of the system. Each distinct system or geographic area in which vehicles are operated SHOULD have its own system_id. Systems IDs SHOULD be recognizable as belonging to a particular system as opposed to random strings - for example, bcycle_austin or biketown_pdx.
	`name*`	STRING Name of the system to be displayed to customers.
	`short_name`	STRING abbreviation for a system.

	operator	STRING Name of the system operator.
	email	EMAIL This OPTIONAL field SHOULD contain a single contact email address actively monitored by the operator's customer service department. This email address SHOULD be a direct contact point where riders can reach a customer service representative.
	feed_contact_email	EMAIL This OPTIONAL field SHOULD contain a single contact email for feed consumers to report technical issues with the feed.
	phone_number	PHONE NUMBER This OPTIONAL field SHOULD contain a single voice telephone number for the specified system's customer service department. It can and SHOULD contain punctuation marks to group the digits of the number. Dialable text (for example, Capital Bikeshare's "877-430-BIKE") is permitted, but the field MUST NOT contain any other descriptive text.
	url	URL The URL of the vehicle share system.

Exemple	<pre>{
	"last_updated": 1609866247,
	"ttl": 0,
	"version": "2.2",
	"data":{
	"system_id":"exemple_abc",
	"name":"Exemple ABC",
	"short_name":"ABC",
	"operator":"Exemple ABC",
	"email":"customerservice@exampleride.org",
	"feed_contact_email": datafeed@exampleride.org,
	"phone_number":"1-800-555-1234",</pre>	

	<pre>"url":"http://www.exampleride.org",
	}
	}</pre>

Type de données demandées : • *Description des métadonnées* ◦ *Date de publication*		
Fichier GBFS	Tous les fichiers du flux de données	
Champs & caractéristiques	`last_updated*`	TIMESTAMP Indicates the last time data in the feed was updated. This timestamp represents the publisher's knowledge of the current state of the system at this point in time.

Exemple	<pre>{
	"last_updated": 1609866247,
	"ttl": 0,
	"version": "2.2"
	}</pre>	

<table>
<tr><td colspan="3">Type de données demandées :
• Description des métadonnées
  ○ Langue</td></tr>
<tr><td>Fichier GBFS</td><td colspan="2">gbfs.json</td></tr>
<tr><td>Champs & caractéristiques</td><td>language*</td><td>LANGUAGE
The language that will be used throughout the rest of the files. It MUST match the value in the system_information.json file.</td></tr>
<tr><td>Exemple</td><td colspan="2"><pre>{
  "last_updated": 1609866247,
  "ttl": 0,
  "version": "2.2",
  "data": {
    "fr" : {
      "feeds": [
        {
          "name": "system_information",
          "url": "https://www.example.com/gbfs/1/fr/system_information"
        },
        {
          "name": "station_information",
          "url": "https://www.example.com/gbfs/1/fr/station_information"
        }
      ]
    }
  }
}</pre></td></tr>
</table>

<table>
<tr><td colspan="3">Type de données demandées :
• Description des métadonnées
  ○ Langue</td></tr>
<tr><td>Fichier GBFS</td><td colspan="2">system_information.json</td></tr>
<tr><td>Champs & caractéristiques</td><td>language*</td><td>LANGUAGE
The language that will be used throughout the rest of the files. It MUST match the value in the system_information.json file.</td></tr>
<tr><td>Exemple</td><td colspan="2"><pre>{
  "last_updated":1611598155,
  "ttl":1800,
  "version": "2.2",
  "data":{
    "system_id":"exemple_abc",
    "name":"Exemple ABC",
    "short_name":"ABC",
    "operator":"Exemple ABC",
    "email":"customerservice@exampleride.org",
    "feed_contact_email": datafeed@exampleride.org,
    "phone_number":"1-800-555-1234",
    "url":"http://www.exampleride.org",
    "language":"fr",
  }
}</pre></td></tr>
</table>

Type de données demandées : ● *Description des métadonnées* ○ *Entité concernée par les données* ○ *Nom* ○ *Adresse mail* ○ *Adresse* ○ *Téléphone* ○ *Url du site web* ○ *Date de début de validité* ○ *Date de fin de validité* ○ *Couverture géographique* ○ *Couverture du réseau* ○ *Le cas échéant, licence utilisée et condition d'utilisation (lien http:// de la licence à inclure)* ○ *Indicateur de qualité (veuillez les indiquer) :*	
Fichier GBFS	system_information.json system_regions.json gbfs.json
Notes	L'ensemble de ces éléments sont repris dans les tableaux précédents pour l'identification des métadonnées en lien avec l'opérateur

Type de données demandées :
• *Description des métadonnées*
  ○ *Nom du jeu de données*
  ○ *Description du jeu de données*
  ○ *Catégorie de données publiées et détails*
  ○ *Le cas échéant, le format ou la spécification technique utilisé pour la publication (lien http:// de la spécification technique)*
  ○ *Interface d'accès*
  ○ *Lien URL*
  ○ *Fréquence de mise à jour*	

Fichier GBFS	N/A