GTFS-PATHWAYS METHODOLOGY HANDBOOK

MobilityData

kisio

Google

GTFS-PATHWAYS METHODOLOGY HANDBOOK

Version 1.0 × September 2019

Written by Charlotte Chebassier and Felix Motot
Contributions by Charlotte Berthier, Sei Cabrol, Guillaume Fournier, Leo Frachet, Thi Hoa NGuyen, David O'Neill, Alex Pool, Pascal Rhod and Sandrine Salaun

MobilityData   Kisio   Google

Credits

Summary

1. Introduction

1.1. About this handbook

This handbook is a comprehensive methodology to perform data collection regarding accessibility in public transport stations, according to the GTFS-Pathways format. GTFS (General Transit Feed Specification) is the format used by most of the transit agencies around the world to publish information related to their networks (lines, stops, stop times, …).

The main goal of the GTFS-Pathways extension is to provide more accurate information about transit stations, to route people through it, and to provide the most comprehensive accessibility information for all type of users. There are currently several extensions proposal under review under the lead of Mobilitydata[1]. This methodology is based on the current full extension proposal available online[2].

This handbook first describes the aim and the content of the GTFS-Pathways extensions (*Section 2. GTFS-Pathways basics, page 7*). It then explains how to represent transit spaces into graphs (*Section 3. GTFS-Pathways graph representation, page 8*). In the next section, it describes which information to collect and how to complete this data collection and survey in the most efficient and simple way (*Section 4. Data collection methodology, page 15*). The last section explains how to perform quality checks to ensure the data collected will be usable by consumers (section E).

This methodology for GTFS-Pathways is meant to be shared, used and improved per the community inputs. We intend to share our learning and hopefully help cities to produce GTFS-Pathways more easily. Access to accessibility information is key to improve the use of public transportation systems by all type of users.

It is important to note that this handbook is not exhaustive and will not cover all situations. In all cities, odd cases can be found. This format is intended to cover most of them, and be flexible enough to adapt for other cases. In all cases, we do recommend to use common sense when doing data collection or modeling. For instance, if many escalators and elevators are starting within a perimeter of 10 to 20 meters, it would complicate the model to multiplie nodes within this perimeter, and the multiple nodes would likely not add any value to the pathway modeling.

1.2. About the authors

As part of the project *Go Deeper Paris*, Kisio wrote this handbook based on a pilot project in Paris. Kisio completed a GTFS-Pathways dataset for two Parisian subway and train stations: Réaumur Sébastopol and Saint-Lazare – Haussmann (May 2019). This dataset, meant for testing

1 Rocky Mountain Institute, the incubator of MobilityData, is an independent nonprofit that works to transform global energy use to create a clean, prosperous, and secure low-carbon future. *https://mobilitydata.org/*

2 Link toward GTFS-Pathways extensions proposal : *https://docs.google.com/document/d/1qJOTe4m_a4dcJnvXYt4smYj4QQ1ejZ8CvLBYzDM5IyM/edit#heading=h.97lvzislswnw*

purpose only, is published and available in open data on *kisioanalysis.io*[3]. Some representations of the dataset are available in the appendix (*Section § appendix, page 37*). Kisio collected all the data during the first semester of 2019 with the help of a property data collection software.

Kisio would like to thanks *Mobilitydata.org* and Leo Frachet for their meaningful help and feedback during this project. Their strong expertise challenged our findings and contributed to a more qualitative work.

The Kisio firm, as a transportation data producer and consumer, has brought a transverse expertise to the project. It is in our firm best interest and the entire transportation community to produce higher quality GTFS. Kisio is involved with all mobility stakeholders to create, deploy and animate mobility services in order to improve the lives of citizens and territories. Our core three disciplines are digital solutions, mobility services and consulting.

If you are interested by our work and would like to provide feedback or ask questions, please feel free to contact us at *datalab@kisio.com*.

3 direct link : *https://kisioanalysis.io/gtfs-pathways*

2. GTFS-Pathways basics

GTFS-Pathways extensions are primarily meant to inform users about the physical and visual accessibility of transit indoor spaces. This format can be used for several other important purposes, like providing accurate and precise information about routing and transfer time.

The extensions are made of additional columns on the *stops.txt* file, that already exists in GTFS-Static, as well as four new files:

- *pathways.txt*, that describes information related to pathways within transit areas;
- *levels.txt*, that describes the different levels within a station and their relationship to pathways. We strongly recommend to provide this file only when there are elevators in the station and therefore level signage available to the public;
- *traversal_times.txt*, meant to provide information about waiting time in controls areas (e.g. passport control, lines);
- *pathway_evolutions.txt* that describes the evolution of pathways depending on the time (opening and closure, change of directions, …).

Unlike GTFS-Static, most of the information needs to be collected on the field because it is based on the real user experience and its lecture of the available signage on site. We do strongly recommend to work with transit authorities and operators, for legal issues as well as practical challenges : some of the information can only known by the operators and authorities and would be very difficult to obtain (e.g. mechanical pathways evolutions). In addition, as indicated in *Section 3.1.2. Dimensioning, page 9*, large stations or hubs require the use of background plans to complete the graph modeling due to complexity.

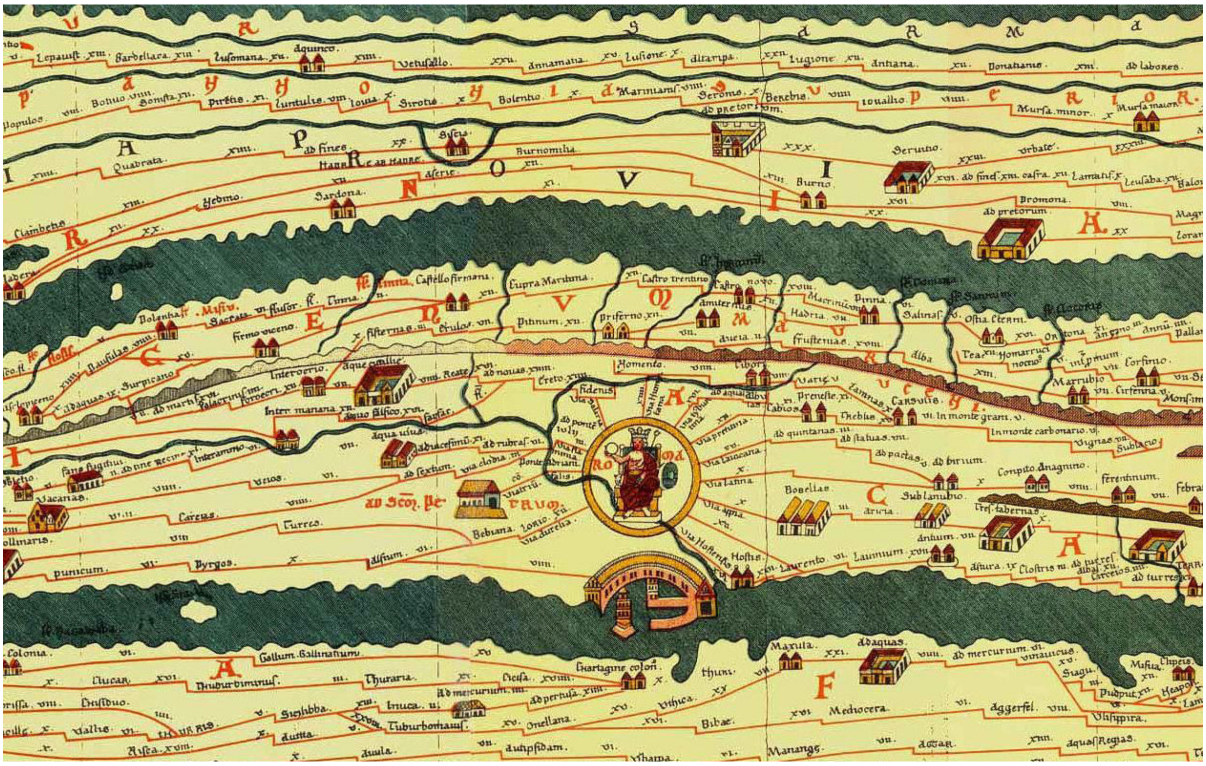
As explained in *Section III – Large station or hub (PE ≥ 30), page 10*, most of large station or hubs are shared between several operators and therefore do not necessarily have continuous signage, naming convention and level of accessibility. Since GTFS-Pathways is meant to help all passengers throughout a complete journey within the station, we do strongly recommend that the data collection is made by only one entity. Following this recommendation would ensure consistency throughout the dataset and therefore would provide the best possible guidance to the final users.

3. GTFS-Pathways graph representation

3.1. How to model?

3.1.1. Definition

It is essential to understand that GTFS-Pathways is not meant to map indoor space. Mapping should be done with mapping tools and technologies like GIS, BIM and CAD. GTFS-Pathways is meant to indicate an itinerary and describe the relationship of different pathways within the same network. The Peutinger Map[1], an ancient roman road map, is an accurate historical comparison: a very schematical map showing road network and connections between them to provide an itinerary. The network is out of scale, but this doesn't matter because we mostly care about features that can guide and direct people through a journey. The transposition of this concept with transit indoor spaces is similar: we do indicate pathways network (edges) and their connections (nodes) into a graph model (matrix).



Tabula Peutingeriana, 1-4th century CE. Facsimile edition by Conradi Millieri, 1887-1888

1 Tabula Peutingeriana, also referred to as Peutinger's Tabula, is an illustrated ancient Roman road map showing the layout of the cursus publicus, the road network of the Roman Empire. The parchment dates from the XIIIth century, but may be a copy of an original map from the IVth or Vth centuries. 555 cities and 3,500 spatial references are indicated. Wikimedia commons - *http://www.fh-augsburg.de/~harsch/Chronologia/Lspost03/Tabula/tab_pe00.html Ulrich Harsch Bibliotheca Augustana*

3.1.2. Dimensioning

The complexity of a station modelization, and therefore its data collection, depends on three criteria:

1. Number of platforms: the more platforms, the more connections and pathways there will be.
2. Number of entrances and exits: the more entrances and exits, the more pathways, fare gates and exit gates there will be.
3. Number of pathway modes: the more mechanical pathways, the more complex and duplicated will be each pathway (assuming a stair is always required for fire safety reasons).

To assess the dimension and level of complexity of the stations, a coefficient PE will help estimation at an early stage of the projects.

Coefficient PE = Number of platforms x Number of entrances and exits

This coefficient does not use the third criteria "number of pathway modes" since this value can only be known once the graph is modeled and verified on the field. Based on our previous data collection and transit expertise, we propose to classify each station into three categories, depending on their PE value.

CATEGORY	COMPLEXITY	PE VALUE
I - Small station	Easy, accessible to any profile	PE < 10
II - Medium station	Some complexity, needs expert profile	10 ≤ PE < 30
III - Large station or Hub	Very complex, needs expert profile	PE ≥ 30

I - Small Stations : Paris/Louvre-Rivoli (PE = 2), New York City/Canal St (PE = 8)

II - Medium stations : Paris/Réaumur-Sébastopol (PE = 16), New York City/Bleeker St-Lafayette (PE = 10)

III - Large station or Hub : Paris/Saint-Lazare - Haussmann (PE = 1064), New York City/Times Square-42St (PE = 952), London/Kings Cross-St-Pancras (PE = 209)

I - Simple station (PE < 10)

For stations with a PE coefficient between 1 and 9, the conceptualization of transit spaces into GTFS-Pathways is fairly simple and does not require a great level of abstraction.

If accurate plans of the station are available easily, they should be used to model the station ahead. Although in most cases, complete plans are not necessarily available from one entity and do not always provide the most accurate information. Therefore, graph modelization can be conceptually done ahead by informing the number of platforms and the number of exits and entrances. It can be assumed there will be one fare gate area for 2 entrances and exits in average. Then, the worst case scenario shall be modeled: all pathways from entrances/exits to fare/exit fare gates are connected, and all fare/exit fare gates are connected to all platforms. This conceptual modelization shall be checked in the field, and all pathways that do not exist shall be removed. If some pathways have various pathway_modes, they shall be added to the model on site.

I - Medium station (10 ≤ PE < 30)

Medium stations have a PE coefficient comprised between 10 and 29. The conceptualization of GTFS-Pathways is slightly more complicated since there will be, by definition, more pathways to add. Most of these stations will have at least one transfer, or many entrances and exits. If accurate plans of the station are available easily, they should be used to model the station ahead.

The same methodology as simple stations shall be used when no plan is available. Although it shall be noted that likely only trained and experimented surveyors with a good sense of orientation could perform the graph representation.

III – Large station or hub (PE ≥ 30)

For a PE coefficient of 30 and above, we consider the transit space as a large station or hub. In this case, the station is at all time complex, if it is not extremely complex. In general, there will be several transportation modes located within the station (subway, commuter, tramway, buses and train lines) and several transportation authorities will often share that space. In these circumstances, GTFS-Pathways will be particularly relevant because wayfindings, signage and naming conventions may not be consistent, making it difficult for the public to travel from one point to another.

Due to its complexity, the graph representation of such category will require the use of plans and backgrounds. Each graph will require the expertise of architectural experts that do understand the space organization and can read architectural drawings or 3D.

3.2. What to model?

3.2.1. Perimeter

GTFS-Pathways should be used to model all pathways within a transportation space: metro station, train station, tramway station, bus station, airport, ... It shall be noted that GTFS-Pathways could also be used to model other kind of spaces, like commercial centers. These spaces are mostly indoor, but can occasionally be outdoor (e.g. bus station).

As explained in the specifications, GTFS-Pathways is assumed to be exhaustive. All public pathways should be modelled within the station. The perimeter of the modeling should be within the property limits of the transit station, often delimited by a gate, a fence, a wall or a building. For the specific case of station entrances and exits, see *Section 3.2.2. Schematic modeling per stop location type, page 11* for methodology.

To reiterate our previous statement in *Section 2. GTFS-Pathways basics, page 7*, a large station is generally managed by one transportation authority and can be integrated within a cluster of several stations managed by different transportation authorities. Since the main goal of GTFS-Pathways is to inform transit users about the full accessibility of a path from platforms to platforms or from entrances/exits to platforms, we strongly recommend to model GTFS-

Pathways and collect the data for an entire cluster at once. Joining different pathways collected by different entities may be challenging. Also, information related to pathways between end points of each data collection would be missing, cancelling the main benefits of GTFS-pathways.

3.2.2. Schematic modeling per stop location type

0: Stop or Platform

`A location where passengers board or disembark from a transit vehicle is called a "platform" when defined within a parent_station.`

Each platform giving access to railway tracks or bus boarding should be specified individually, even when two platforms are positioned next to each other and are continuous. If many train or bus lines use the platform, the naming shall indicate the platform code - numbering or lettering – or naming (e.g. "12", "C"). If the platform is permanently allocated to one or several lines, the naming shall indicate the platform signage (e.g. "Metropolitan Ave", "M3, Galliéni").

Platforms normally already exist in the GTFS-Static feed: in that case, the same identifier has to be used. It happens that one stop identifies more than one platform: then new platforms have to be instantiated, and the GTFS-Static feed has to be adapted accordingly (changes have to be reflected in stop_times.txt).

◊ Advised naming convention and examples

```
[related station initials]:P[line code or platform number/letter][A-Z*]
RS:P3A (Platform of line 3, direction Galliéni, Réaumur Sébastopol)
SL:P23 (platform 23, Saint-Lazare station)
```

1: Station

`A physical structure or area that contains one or more platform.`

Stations already exist in GTFS-Static: the same identifiers have to be used.

◊ Advised naming convention and example

```
[related station initials]:S[1000-1999]
RS:S1 (Réaumur Sébastpol)
```

2: Station Entrance/Exit

`A location where passengers can enter or exit a station from the street. The entrance/exit should also specify a parent_station value referencing the stop_id of the parent station for the entrance. If an entrance/exit belongs to multiple stations, it will be linked by pathways to both, and the data provider can pick one of the stations as parent.`

A station entrance/exit node shall be located at the edge of a public pathway, in most case within the street right of way. Choosing the right of way as an origin provides the most comprehensive information regarding access within the station and toward the platforms. For instance, if the station entrance/exit has a stairway leading toward one level down, and the main gate is located one level down, GTFS-Pathways should include the stairway.

When two different exit/entrance pathways lead to exits/entries very close to each other (less than 20 meters), that are identified as only one exit/entry in the signage, there should be only one stop in the GTFS.

Station entrances/exits normally already exist in the GTFS-Static feed: in that case, the same identifier has to be used. It happens that some are missing, then they have to be instantiated.

◊ Advised naming convention and example

`[related station initials]:E[0-999] – if the entrance/exit is identified by a number, it has to be used.`

`RS:E2(Exit 2, Réaumur Sébastopol station)`

3: Generic Node

`A location within a station, not matching any other location_type, which can be used to link together pathways defined in pathways.txt.`

GTFS-Pathways specifications provide several examples and guidelines regarding the use of generic nodes. The main idea is to simplify modelization with as little nodes and complexity as possible.

In addition to this recommendation, here are a few specific use cases related to generic nodes:

Mechanical pathways

Since all mechanical pathways/edges will be referred within pathway_evolutions.txt for real-time updates, they need to be specified only once to avoid redundancy. To ensure they are unique, and to ease data collection, we do recommend adding a generic node at each beginning and end of mechanical pathways for escalators, travellators and elevators. If multiple pathways start or end at the same location within a diameter of about 10 meters, generic nodes can be shared and do not need to be multiplied (e.g. an elevator entrance is located 8m away from an escalator ramp, two travellators are following each other with a 10 meters walkway in-between)

Mechanical pathways/edges shall be bidirectional for all elevators unless clearly specified by the transportation authority. Escalators and travellators shall be mono-directional. If they do change direction during certain times of the day, this information will be specified within pathway_evolutions.txt.

◊ Advised naming convention

`[related station initials]:N[4000-4999]`

Fare Gates and Exit Fare Gates

Although it is fully justified and coherent within GTFS-Pathways, the concept of specifying a fare or exit gate as a pathway can be a little difficult to grasp and potentially leads to confusion during the data collection.

First, if several fare/exit fare gates are located within the same area (about 30 meters perimeter), they will be modeled as a single ensemble.

Second, we do recommend a specific way to model all fare/exit fare gates. To avoid mistakes, it is better to position one generic node before the fare/exit fare gate and another after, in all circumstances. Pathways, either fare gates or exit fare gates, connect these two nodes. Fare and exit fare gates pathways shall be mono-directional in all cases. If a fare or exit fare gate is bidirectional and can enable both entrance and exit within the transit system, it should be specified twice with one pathway as a fare gate, and one pathway as an exit fare gate. If some fare/exit fare gates have different dimensions or types, a different pathway shall be modeled for each type (e.g. accessible gate, automated gate, ...)

If a gate leads from one transit agency area to another transit agency area, it should be specified as a fare gate.

When a fare gate is not restricting passage and is only used for fare payment, it should be refered within the *fare.txt* extension file. There is not need to model the gate since it is not part of pathways.

◊ Advised naming convention and exemple

```
[related station initials]:F[3000-3999]
RS:F3002(a fare gate node, Réaumur Sébastopol station)
```

4: Boarding Area

`A specific location on a platform, where passengers can board and/or alight vehicles.`

In all circumstances, we do recommend to specify a boarding area at the beginning and at the end of a platform. The beginning and the ending boarding areas should be placed relatively to the platform rather than the vehicles. If the position of each carriage is clearly identified on the platform, then a boarding area node should be specified for each sign (e.g. "A", "Car n°6", "Car n°9 - Nozomi, Hiraki", letter sign "A" "B" "C"). Otherwise if the platform is longer than 2 coaches or carriages or is longer than 50 meters, a boarding area in the middle should be added to provide a more accurate positioning for users.

When signage is positioned along the platform to mark some specific information relatively to the vehicles positions, the position should be identified by an additional boarding area and named after the signage (e.g. "end of short trains", "beginning of short train", etc).

Other than beginning, middle, ending and specific signs, no other boarding area that is not identified and marked should be specified.

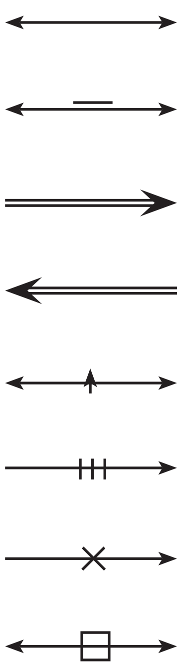
Each specified boarding areas next to each other within the same platform should be connected together by bidirectional pathways.

◊ Advised naming convention and exemple

```
[related station initials]:B[2000-2999]
RS:B2002(a boarding area, Réaumur Sébastopol station)
```

3.2.3. Schematic modeling convention

	STOPS.TXT *(nodes)*	**Numbering** *(Even numbers, 4 digits XXXX)*
S	Station	S + 1000 =< XXX =< 1999
L	Level	L + elevator level numbering
P	Platform	P + line code + A or B
B	Boarding Area	B + 2000 =< XXX =< 2999
E	Entrance / Exit	E + < 1000 (based on station numbering)
F	Fare gate / Exit fare gate	F + 3000 =< XXX =< 3999
N	Generic Node	N + 4000 =< XXX =< 4999
	Node on another level	Z + XXXX



PATHWAYS.TXT *(edges)*	**Direction**
Walkway	bidirectionnel / unidirectionnel
Accessible ramp (walkway)	bidirectionnel / unidirectionnel
Fare gate	unidirectionnel
Exit fare gate	unidirectionnel
Stair	bidirectionnel / unidirectionnel
Escalator	unidirectionnel
Travelator	unidirectionnel
Elevator	bidirectionnel / unidirectionnel

4. Data collection methodology

All fields are defined online and can be found in the specification at the following link : https://developers.google.com/transit/gtfs/reference/. It should be noted that `all text in this font` is an excerpt from this reference document. The aim of the following section is to provide uniform guidelines for data collection. These guidelines were created thanks to Kisio transportation expertise and almost 200 hours of field work entirely dedicated to GTFS-Pathways. They are proposing solutions to collect all the data in cheap, safe and appropriate manner. By proposing a practical methodology for each variable, the data collection shall be eased. If followed properly, they shall lead to a high-quality dataset.

For each field, we indicate the **field name**, the **type**, the **requirement** and the **description** directly extracted from the Google GTFS extensions reference document. All fields where GTFS-Static is indicated are not GTFS-Pathways dedicated. This handbook provides recommendations only if there is a specific use regarding GTFS-Pathways.

4.1. Stops.txt

Field name	**stop_id (GTFS-Static)**
Type	`ID`
Required	`Required`
Definition	`Identifies a stop, station, or station entrance. The term «station entrance» refers to both station entrances and station exits. Stops, stations, and station entrances are collectively referred to as locations. Multiple routes can use the same stop.`
Methodology	

Field name	**stop_code (GTFS-Static)**
Type	`Text`
Required	`Optional`
Definition	`Contains some short text or a number that uniquely identifies the stop for riders. Stop codes are often used in phone-based transit information systems or printed on stop signage to make it easier for riders to get information for a particular stop.` `The stop_code can be the same as stop_id if the ID is public facing. Leave this field blank for stops without a code presented to riders.`

Field name	**stop_name (GTFS-Static)**
Type	`Text`
Required	`Conditionally required`

Definition	Contains the name of a location. Use a name that people understand in the local and tourist vernacular. When the location is a boarding area, with location_type=4, include the name of the boarding area as displayed by the agency in the stop_name. It can be just one letter or text like «Wheelchair boarding area» or «Head of short trains.» This field is required for locations that are stops, stations, or entrances/exits, which have location_type fields of 0, 1, and 2 respectively. This field is optional for locations that are generic nodes or boarding areas, which have location_type fields of 3 and 4 respectively.
Methodology	- 0 (platforms): if many train or bus lines use the platform, the naming shall indicate the platform code - numbering or lettering – or naming (e.g. "12", "C"), - 1 (stations): the station name (e.g. "Chatelet"), - 2 (entrances/exits): the entrance or exit name (e.g. "1 - rue Réaumur") - 3 (boarding areas): reflect the signage if relevant (e.g. "A")

Field name	**stop_desc (GTFS-Static)**
Type	Text
Required	Optional
Definition	Describes a location. Provide useful, quality information. Don't simply duplicate the name of the location.

Field name	**stop_lat**
Type	Latitude
Required	Conditonally required
Definition	Contains the latitude of a stop, station, or station entrance. This field is required for locations that are stops, stations, or entrances/exits, which have location_type fields of 0, 1, and 2 respectively. This field is optional for locations that are generic nodes or boarding areas, which have location_type fields of 3 and 4 respectively.
Methodology	When a platform is underground and its GPS location cannot be catched, use the parent station position.
Source	GPS

Field name	**stop_lon**
Type	Longitude
Required	Conditonally required

Definition	Contains the longitude of a stop, station, or station entrance. This field is required for locations that are stops, stations, or entrances/exits, which have location_type fields of 0, 1, and 2 respectively. This field is optional for locations that are generic nodes or boarding areas, which have location_type fields of 3 and 4 respectively.
Methodology	When a platform is underground and its GPS location cannot be catched, use the parent station position.
Source	GPS

Field name	**zone_id (GTFS-Static)**
Type	ID
Required	Conditionally required
Definition	Defines the fare zone for a stop. This field is required if you want to provide fare information with fare_rules.txt. If this record represents a station or station entrance, the zone_id is ignored.

Field name	**stop_url (GTFS-Static)**
Type	URL
Required	Optional
Definition	Contains the URL of a web page about a particular stop. Make this different from the agency_url and route_url fields.

Field name	**location_type (GTFS-Static)**
Type	Enum
Required	Optional
Definition	Defines the type of the location. The location_type field can have the following values: 0 or (empty): Stop (or « Platform »). A location where passengers board or disembark from a transit vehicle. Stops are called a «platform» when they're defined within a parent_station. 1: Station. A physical structure or area that contains one or more platforms. 2: Station entrance or exit. A location where passengers can enter or exit a station from the street. The stop entry must also specify a parent_station value that references the stop_id of the parent station for the entrance. If an entrance/exit belongs to multiple stations, it's linked by pathways to both, and the data provider can either pick one station as parent, or put no parent station at all. 3: Generic node. A location within a station that doesn't match any other location_type. Generic nodes are used to link together the pathways defined in pathways.txt. 4: Boarding area. A specific location on a platform where passengers can board or exit vehicles.

Methodology	Refer to *Section 3.2.2. Schematic modeling per stop location type, page 11* for graph modeling methodology.
Source	Observation, plans

Field name	**parent_station (GTFS-Static)**
Type	`ID that references stops.stop_id`
Required	`Conditionally required`
Definition	The parent_station field defines hierarchy between the different locations defined in stops.txt. It contains the ID of the parent location, as followed: Stop/platform (location_type=0): the parent_station field contains the ID of a station. Station (location_type=1): this field must be empty. Entrance/exit or generic node (location_type=2 or 3): the parent_station field contains the ID of a station (location_type=1) Boarding Area (location_type=4): the parent_station field contains ID of a platform. Conditionally Required: Required for location_type 2, 3 & 4. Optional for location_type 0. Forbidden: location_type 1.
Methodology	

Field name	**wheelchair_boarding (GTFS-Static)**
Type	`Enum`
Required	`Optional`

Definition	Identifies whether wheelchair boardings are possible from the specified stop, station, or station entrance. This field can have the following values: 0 or (empty): Indicates that there's no accessibility information available for this stop. 1: Indicates that at least some vehicles at this stop can be boarded by a rider in a wheelchair. 2: Indicates that wheelchair boarding isn't possible at this stop. When a stop is part of a larger station complex, as indicated by the presence of a parent_station value, the stop's wheelchair_boarding field has the following additional semantics: 0 or (empty): The stop inherits its wheelchair_boarding value from the parent station if it exists. 1: Some accessible path exists from outside the station to the specific stop or platform. 2: There are no accessible paths from outside the station to the specific stop or platform. For station entrances/exits, the wheelchair_boarding field has the following additional semantics: 0 or (empty): The station entrance inherits its wheelchair_boarding value from the parent station if it exists. 1: The station entrance is wheelchair accessible, such as when an elevator is available to reach platforms that aren't at-grade. 2: There are no accessible paths from the entrance to the station platforms.
Source	This field is meant to provide the public transit authority regulation and official recommendation. GTFS-Pathways is more complete to inform about accessibility and step-free pathways.

Field name	**level_id**
Type	ID that references levels.level_id
Required	Optional
Definition	Provides the level of the location. The same level can be used by multiple unlinked stations.
Methodology	It is recommended to use this field only if elevators are available in the station.

Field name	**platform_code (GTFS-Static)**
Type	Text
Required	Optional
Definition	Provides the platform identifier for a platform stop, which is a stop that belongs to a station. Just include the platform identifier, such as G or 3. Don't include words like «platform» or «track» or the feed's language-specific equivalent. This allows feed consumers to more easily internationalize and localize the platform identifier into other languages.

Field name	**boarding_edge**
Type	Enum
Required	Optional
Definition	The boarding_edge field specifies if the platform or boarding area is protected: 0 or empty: No information 1: Open platform 2: Manual platform edge doors 3: Automated platform edge doors This field is recommended for location_type=4 (boarding area).
Methodology	If the platform is not continuous along its edge, the boarding edge shall be informed either specifically for each boarding area, either with the lowest protection (open platform-1, then manual platform edge doors-2)
Source	Observation

Field name	**boarding_height**
Type	Float
Required	Optional
Definition	The stop_height field specifies the height in meters between the floor (defined as the surface on which the wheels are rolling) and the edge of the platform or quay or sidewalk. This value would be negative for suspended vehicles (on rail or cable). The difference between this value and the vehicle.floor_height aims to give information about a possible step between the platform and the vehicle. The height of the step being: height = floor_height - boarding_height
Preset	0 or empty: The boarding height is accessible (less than 2cm (3/4in) threshold) 2: The boarding height has an accessible step (between 2cm and 16cm (6in) step) 16: The boarding height is non accessible (more than 16cm step)
Methodology	Since this information needs to be collected in a hazardous location (rail tracks) and requires specific permits and temporary shutdown of the transit system, we strongly recommend to collect this information through observation and visual approximation. Refer to the classification above to describe the boarding height.
Source	Observation

Field name	**boarding_distance**
Type	Float
Required	Optional

Definition	The boarding_distance field specifies the distance in meters between the edge of the platform or quay or sidewalk to the middle of the track. This value should be provided only if there is a guidance system (e.g. rail, cable), and should not be provided if each driver can stop at a different distance of the edge of the platform. This field, used with vehicle.vehicle_width, aims to give information about a possible gap between the platform and the vehicle. The size of the gap being: gap = boarding_distance - vehicle_width/2
Preset	0 or empty: The boarding distance is accessible (less than 2cm (~3/4in) gap) 2: The boarding distance has an accessible gap (between 2 and 5cm (~2in) gap) 5: The boarding distance is non accessible (more than 5cm gap)
Methodology	Since this information needs to be collected in a hazardous location (rail tracks) and requires specific permits and temporary shutdown of the transit system, we strongly recommend to collect this information through observation and visual approximation. Refer to the classification above to describe the boarding height.
Source	Observation

4.2. Levels.txt

Field name	**level_id**
Type	`ID`
Required	`Required`
Definition	`Identifies the level of the station.`

Field name	**level_index**
Type	`Float`
Required	`Required`
Definition	`Indicates the relative position of this level in relation to other levels. Levels with higher indices are assumed to be located above levels with lower indices.` `Use an index of 0 to indicate ground level, with levels above ground indicated by positive indices and levels below ground by negative indices.`
Methodology	Half level can be used when relevant. In some instances, be aware that the level index may not match the level name.
Source	Observation, plans & sections

Field name	**level_name**
Type	Text
Required	Required
Definition	Provides the name of the level. Be sure the name matches the level lettering or numbering used inside the building or station. This field is useful for elevator routing, such as directions to take the elevator to the levels «Mezzanine,» «Platforms,» or «-1.»
Methodology	When levels names are not consistent along the signage (on the elevator, inside the elevator and within the station), the name may combine all the names being used.
Source	Observation, plans

Field name	**elevation**
Type	Float
Required	Optional
Definition	Elevation above ground, in meters (with negative values indicating underground locations.
Methodology	Elevation shall be informed only if the information is available on survey documents and will provide relevant information to the database.
Source	Survey documents

4.3. Pathways.txt

Field name	**pathway_id**
Type	ID
Required	Required
Definition	Identifies the pathway. Different pathways can connect from the same from_stop_id to the same to_stop_id. For example, this happens when two escalators are side by side in opposite directions, or when a stair and elevator both go from the same place to the same place.
Preset	Concatenation of station id, from_stop_id and to_stop_id.

Field name	**from_stop_id**
Type	ID that references stops.stop_id
Required	Required
Definition	Defines the location at which the pathway begins. It contains a stop_id that identifies a platform, entrance/exit, generic node, or boarding area from the stops.txt file.
Preset	ID to be extracted from the graph representation Refer to modeling convention for naming

Field name	**to_stop_id**
Type	`ID that references stops.stop_id`
Required	`Required`
Definition	`Defines the location at which the pathway ends. It contains a stop_id that identifies a platform, entrance/exit, generic node, or boarding area from the stops.txt file.`
Preset	ID to be extracted from the graph representation Refer to modeling convention for naming

Field name	**pathway_mode**
Type	`Enum`
Required	`Required`

Definition	`Specifies the type of pathway between the specified pair (from_stop_id, to_stop_id). The following are valid values for this field:`
	`1: Walkway`
	`2: Stairs`
	`3: Moving sidewalk/travelator`
	`4: Escalator`
	`5: Elevator`
	`6: Fare gate (or payment gate). A pathway that crosses into an area of the station where a proof of payment is required. This is usually a physical payment gate.`
	`Fare gates might either separate paid areas of the station from unpaid ones or separate different payment areas within the same station from each other. This information can be used to route passengers through stations in a way that helps them avoid unnecessary payments.`
	`7: Exit gate. A pathway where passengers exit an area that required proof-of-payment to enter and which connects to an area where proof-of-payment is no longer required.`

Methodology	Refer to the graph modelling sections for the methodology of each pathway mode. Refer to the *Section 3.2.2. Schematic modeling per stop location type, page 11* to know which variable shall be collected per each pathway mode. When a pathway is composite with several non-mechanical pathway modes, the most constraining mode should be indicated, in priority order: control-8, fare gate-6 and exit fare gate-7, stairs-2, walkway-1. It should be noted that we strongly do not recommend to combine mechanical paths (travelator-3, escalator-4 and elevator-5) with any other pathway modes. Since the purpose of pathway_evolutions is to provide real time information on mechanical pathways (repairs, breakdowns, ...), they do have to be isolated from each others. When several mechanical paths are located close to each other (within 10 meters – 30ft), the walkway in between them can be included with the pathway to avoid the creation of an additional pathway.
Source	Observation, plans

Field name	**is_bidirectional**
Type	`Enum`
Required	`Required`

Definition	`Indicates in which directions the pathway can be used. The following are valid values for this field:`
	`0: Unidirectional pathway. It can only be used in the direction of from_stop_id toward to_stop_id.`
	`1: Bidirectional pathway. It can be used in either direction.`
	`Fare gates (pathway_mode=6) and exit gates (pathway_mode=7) can't be bidirectional.`

Methodology	All pathways can be assumed to be bidirectional. In the following circumstances, they should be specified as unidirectional : • For fare gates (pathway_mode=6) and exit fare gates (pathway_mode=7) • For moving travelators and escalators • When on-site signage does not permit to use the path in one direction (prohibited direction sign, emergency exit only, ...)
Source	Observation based on signage

Field name	**cover_type**
Type	Enum
Required	Optional
Definition	The cover_type field specifies the protection that the pathway has against the weather: • 0 or empty: No information • 1: Outside pathway • 2: Covered pathway (with a roof; protection against vertical rain) • 3: Inside pathway (with roof & walls; protection against rain & wind)
Methodology	If a pathway has different type of protection against the weather, it is recommended : • Either to specify two distinct pathways if the distinction is clear and for a significant distance (above 30 meters), • Or to specify the lowest level of weather available if the distinction is minor (e.g. pathways connected with outside entrance and exit stop should be specified as outside pathway (cover_type=1).
Source	Observation, plans

Field name	**length**
Type	`Non-negative float`
Required	`Optional`

Definition	`Specifies the horizontal length, in meters, of the pathway. The length is measured from the origin location, defined in from_stop_id, to the destination location, defined in to_stop_id.` `This field is recommended for walkways, fare gates, and exit gates, which have pathway_mode equal to 1, 6, and 7 respectively.`
Methodology	The length of each walkway, stairs and gates shall be measured from the beginning to the end. When a pathway is composite with both walkways and stairs or gates, the overall length shall be provided as a compound measure. Since the purpose of this length is to be converted into transfer duration as minutes, the meter or foot distance can be rounded to the nearest fifth.
Source	• Pedometer[2], to be calibrated with each surveyor before data collection. Many free apps are available for download on smartphones. • If no pedometer can be available, we do recommend the use of measuring wheel. • Plans

Field name	**mechanical_length**
Type	Non-negative float
Required	Optional
Definition	The mechanical_length field specifies the horizontal length in meters of a travelator. This field is recommended for pathway_mode=3 (travelator). When the travelator is functioning, the data consumer should use traversal_time and ignore mechanical_length. But during travelator downtime, the data consumer should consider the travelator as walkway, and use mechanical_length.
Methodology	The length of travelators shall be measured from the beginning to the end, by using the walkway on its side if available. Since the purpose of this length is to be converted into transfer duration as minutes, the meter or foot distance can be rounded to the nearest fifth.
Source	• Pedometer, to be calibrated with each surveyor before data collection. Many free apps are available for download on smartphones. • If no pedometer can be available, we do recommend the use of measuring wheel. • Plans

2 Accuracy of smartphone pedometer applications has been assessed in several research paper. Based on the use of different applications, the error can be fairly high at low speed (2km/h) (60.5% to 19.3%). But at normal walking speed (4km/h), pedometer applications indicate an absolute standard error between 12.5% and 1.1%. Presset, 2018. Therefore, for a 100m corridor, the error could be between 1m and 12m. This variation is marginal (in seconds) at the scale of an indoor transit space where we want to provide transfers duration in minutes based on a distance and a walking speed.

Field name	**traversal_time**
Type	`Positive integer`
Required	`Optional`

Definition	`Specifies the average time needed to walk through the pathway. The following are valid values for this field:(empty): Unknown traversal time.`
	`A positive integer: the number of seconds needed to traverse this pathway on foot.`
	`This field is recommended for mechanical pathways, such as moving sidewalks, escalators, and elevators, which have pathway_mode equal to 3, 4, and 5 respectively.`

Preset	For elevators, we recommend to provide a reference value of 30 seconds per floor.
Methodology	**For escalators and travelators** (pathway_mode 3 and 4) The traversal time shall always be measured in a static position to be able to calculate the speed of the mechanical path. The traversal time shall be measured from the moment someone steps on the moving path to the moment someone steps out of it. **For elevators** (pathway_mode 5) The traversal time shall be indicated only if a countdown timer indicates a specific frequency between elevator loops. We do not recommend providing traversal time for typical elevators, since the duration may vary depending on stops between floors, waiting time, fill factor, door closing time and programming.
Source	Chronometer

Field name	**stair_count**
Type	`Non-null integer`
Required	`Optional`

Definition	`Specifies the number of stairs of the pathway. A positive stair_count implies that the passenger walks up from from_stop_id to to_stop_id. A negative stair_count implies that the passenger walks down from from_stop_id to to_stop_id.`
	`This field is recommended for stairs and escalator pathways, which have pathway_mode equal to 2 and 4 respectively.`
	`As a best practice, we recommend the approximation of a change of one floor as equal to 15 stairs, or 12 for an escalator.`

Preset	On standard floor heights, one could use the approximation of 1 floor = 15 stairs to generate approximate values.
Methodology	If within the same pathway several flights of stairs are either going up and down, the total number of steps shall be provided as a positive addition, without negative count.
Source	Observation, Plans

Field name	**max_stair_flight**
Type	Non-null integer
Required	Optional
Definition	The max_stair_flight field specified the maximum number of steps in a row of the pathway (between landings). This field is recommended for stairs pathways, i.e. pathways with pathway_mode 2 (stairs).
Methodology	A flight of stairs is a set of steps from one floor or landing to another. To be considered as a flight, A landing shall be at least 1 meter (~3ft) deep.
Source	Observation, Plans

Field name	**mechanical_stair_count**
Type	Non-null integer
Required	Optional
Definition	The mechanical_stair_count field specified the number of stairs of an escalator. Valid values for this field are : • (empty): unknown number of stairs. • An integer: number of stairs of the escalator. This field is recommended for pathway_mode 4 (escalator). When the escalator is functioning, the data consumer should use traversal_time and ignore mechanical_stair_count. But during escalator downtime, the data consumer should consider the escalator as stairs, and use mechanical_stair_count.
Preset	On standard floor heights, one could use the approximation of 1 floor = 12 stairs to generate approximate values.
Methodology	During data collection, if the escalator is not functioning, the steps can be counted by observation. But if the escalator is running, two collectors are required to count: one will step on the escalator and stay on one step the whole time, while another one will count the number of steps until the first collector signals it is arrived at the end.
Source	Observation, plans

Field name	**max_slope**
Type	`Float`
Required	`Optional`

Definition	Specifies the maximum slope ratio of the pathway. The following are valid values for this field: 0 or (empty): No slope. A float: The slope ratio of the pathway, positive for upwards and negative for downwards. Only use this field with walkways and moving sidewalks, which have the pathway_mode 1 and 3 respectively. As an example, the maximum slope ratio allowed in the US for hand-propelled wheelchairs is 0.083 (also written 8.3%). This means that there's a vertical increase of 0.083 m for each 1 m horizontally.
Preset	To simplify data collection, we suggest using a simplified classification that will be converted into numerical values during exportation: 0 or empty: There is no slope or very little (less than 5%) 5: There is some slope, it is practicable (between 5% and 8%) 9: There is some significant slope (more than 8%)
Methodology	If different slopes can be observed within the same pathway, the worst case should be indicated.
Source	Observation should be sufficient to provide accurate and useful information. If the data collector is not familiar with surveying and accessibility, a digital slope measuring device shall be used when a slope is visible. Digital level application can also be downloaded on smartphones and tablet devices.

Field name	**max_cross_slope**
Type	Float
Required	Optional
Definition	The max_cross_slope field specifies the maximum cross slope ratio of the pathway. Valid values for this field are : • 0 or (empty): no slope. • A float: slope ratio of the pathway, positive for upwards, negative for downwards This field should be used only with the pathway_type 1 (walkway) and 3 (travelator).
Preset	To simplify data collection, we suggest using a simplified classification that will be converted into numerical values during exportation: 0 or empty: There is no cross slope or very little (less than 2%) 2: There is some cross slope, it is practicable (between 2% and 4%) 5: There is some significant cross slope (more than 4%)

Methodology	If different cross slopes can be observed within the same pathway, the worst case should be indicated. The cross slope or transverse slope is defined as the rate of change of roadway elevation with respect to distance perpendicular to the direction of travel. With important cross slope, it can be very difficult for some people with mobility impairments to walk across sloped surfaces than horizontal surfaces, particularly on extreme weather (rain, ice, snow).
Source	There is no standardized method for measuring cross-slope, but observation should be sufficient to provide accurate and useful information. If the data collector is not familiar with surveying and accessibility, a digital slope measuring device shall be used when a slope is visible. Digital level application can also be downloaded on smartphones and tablet devices.

Field name	**min-width**
Type	`Positive float`
Required	`Optional`
Definition	Contains the minimum width of the pathway, in meters. This field is highly recommended if the minimum width is less than one meter and half.
Preset	**The values can be specified before data collection for the known gate types. Then the collector would only have to select the gate type he is facing.**
Methodology	This field shall only be used in case width is lower than 1 meter. This field is mostly relevant in case of mechanical pathways and fare / exit fare gates.
Source	First observation, then use a meter to measure the exact width if relevant

Field name	**pathway_name**
Type	Text
Required	Optional
Definition	The pathway_name field contains the name of the pathway. Please use a name that people will understand in the local and tourist vernacular, if any.
Methodology	This field shall be used only if the pathway has a name mentioned in the signage (e.g. « passerelle Béthune », « rue Intérieure »)
Source	Observation, signage

Field name	**pathway_code**
Type	Text

Required	Optional
Definition	The pathway_code field contains short text or a number that uniquely identifies the pathway for passengers. The pathway_code can be the same as pathway_id if it is passenger-facing. This field should be left blank for pathway without a code presented to passengers. Example : Elevator "A"
Methodology	This field shall be used mostly for pathways that interact with pathway_ evolutions as elevators, escalators and travelators.
Source	Existing database, observation, signage

Field name	**signposted_as**
Type	Text
Required	Optional
Definition	Contains an exact string of text from physical signage that's visible to transit riders. The string can be used to provide text directions to users. Enter the language text in this field exactly how it's printed on the signs, without translation.
Methodology	If signage contains exactly the same information and as the next stop name, the field do not need to be filled (example: if the stop name is "Line 1, Direction downtown" and the signage is identical, the stop name will already contain that information). Signage is relevant to inform mostly if it is different from the direction taken or has some specific information that is relevant for the future user.
Source	Observation based on signage

Field name	**reversed_signposted_as**
Type	Text
Required	Optional
Definition	Same as the signposted_as field, but used when the pathway is traveled in the backward direction, that is, from the to_stop_id to the from_stop_id.
Methodology	This field shall be used only when the pathway is bidirectional. If signage contains exactly the same information as the next stop name, the field does not need to be filled (e.g. if the stop name is "Line 1, Direction downtown" and the signage is identical, the stop name will already contain that information). Signage is relevant to inform mostly if it is different from the direction taken or has some specific information that is relevant for the future user.
Source	Observation based on signage

Field name	**instructions**
Type	Text
Required	Optional
Definition	The instructions field indicates instructions for the pathway taken from the from_stop_id to to_stop_id (e.g. “Go downstairs to the station concourse”). The instructions should not contain abbreviations to allow being read aloud by screenreaders and voice assistants.
Methodology	Based on our pilot project, we identified very few use cases where it was relevant to provide instructions. On one end, providing instructions is complicating the data collection. On the other end, consumer transit applications do not provide instructions to date, they tend to show as little text as possible to simplify comprehension for the user. We recommend providing instructions only if signage or the space is not clear and easy to understand (e.g. an elevator hidden behind a structural pile, an exit barely visible, an inaccurate on-site signage, ...). In this case, instructions shall provide simple information to clarify the situation for the user.
Source	Observation, plans

Field name	**reversed_instructions**
Type	Text
Required	Optional
Definition	Same as the instructions field, but when the pathway is used backward, i.e. from the to_stop_id to the from_stop_id.
Methodology	Same comment as from field instructions.
Source	Observation, plans

Field name	**wheelchair_assistance**
Type	Enum
Required	Optional
Definition	The assistance field indicates if using this pathway requires an additional human assistance. • 0 or empty: This pathway does not require assistance • 1: Using this pathway requires assistance from the staff without prior notice (e.g. wheelchair/stroller door near turnstiles, ramp to install) • 2: Using this pathway requires assistance from staff with prior notice
Preset	This variable can be specified before data collection for the known gate types, then the data collector would only have to select the gate type.
Source	Observation, Public transit authority

Field name	**wheelchair_assistance_phone**
Type	Phone number

Required	Optional
Definition	Indicates the phone number to call to notice about assis-tance needs.
Preset	This variable can be specified before data collection for the known gate types, then the data collector would only have to select the gate type.
Methodology	In most case, this information shall be unique for a same network.
Source	Public transit authority

Field name	**tactile_strip**
Type	Enum
Required	Optional
Definition	• 0 or empty: The pathway has no guiding strips. • 1: The pathway has guiding strips for visually impaired and blind people
Methodology	Only tactile strip designed to guide visually impaired people shall be informed. It is considered that there is tactile strip along the pathway when a linear strip is mostly continuous, with occasional interruptions for guidance and stops. Tactile strips used to alert or avoid slipping shall not be indicated here.
Source	Observation

Field name	**report_phone**
Type	Phone number
Required	Optional
Definition	The report_phone field indicates the phone number to call to report a downtime for the pathways (e.g. broken elevator).
Methodology	In most cases, this information shall be unique for a same network.
Source	Public transit authority

Field name	**report_url**
Type	URL
Required	Optional
Definition	The report_url field indicates the URL on which one can report a downtime for the pathways (e.g. broken elevator).
Methodology	In most cases, this information shall be unique for a same network.
Source	Public transit authority

Field name	**commands_max_height**
Type	Non-negative float
Required	Optional

Definition	The commands_max_height field indicates the maximum height of the command in meters. This field is recommended if the pathway requires some action from the rider. Examples: fare gate validation zone, button to automatically open a door, button to call an elevator, button to choose the level in the elevator.
Preset	This variable can be specified before data collection for each gate type.
Methodology	The measurement shall be taken from the lowest point needed to reach out the highest command. This variable concerns only non-emergency controls.
Source	Tape measurer, Laser meter, Product specifications

Field name	**manual_activation**
Type	Enum
Required	Optional
Definition	Defines if the pathways requires a maneuvering action like pushing or pulling to pass through it: • 0 or empty: no information • 1: Do not requires manual activation • 2: Requires manual activation This can apply for example to elevators, fare gates, exit gates, walkway with manual, automatic or power-assisted door.
Preset	This variable can be specified before data collection for each gate type.
Methodology	Manual activation indicates if strength or maneuvering other than walking is required to use the pathway.
Source	Observation, Product specifications

4.4. Pathway_evolutions.txt

Pathway_evolutions.txt shall be used in different cases :

- When a pathway is not opened 24h 7/7 (e.g. station entrances and exits, access to commercial centers, ...),
- When a mechanical pathway switches direction during the day,
- When a pathway has special behaviour for a limited period of time, service_id shall be filled (e.g. construction within the station). The refered calendar.txt entry provides the dates when the evolution applies.

4.5. Traversal_times.txt

Traversal_times.txt shall only be used when a pathway is known to have significant waiting time before being able to cross (e.g. waiting line for passport control). We did not test this file within this pilot project.

5. Validation

Even though a consistent and rigorous data collection method can limit the number of errors (use of drop-down lists instead of free text input, …), all collected data must be checked, controlled and validated. Once the graph is modelized and the data is collected, remaining issues consist in checking the data global consistency and identify isolated errors.

There are easy checks to implement to guaranty that the GTFS-Pathways dataset will be usable.

5.1. Global consistency

5.1.1. Isolated nodes

There should be no isolated nodes: every single nodes with location_type 2, 3 or 4 have to be connected to at least another one by a pathway, except from platforms and stations (location_type 0 or 1).

5.1.2. Parent station

Boarding areas (location_type 4) always have a platform (location_type 0) as parent_station.

Generic nodes, station entrances/exits and platforms (location_type 0, 2, 3) always have a station as parent.

There should be no platforms with no children: there should always be at least one boarding area that has the platform as parent.

Every station should be the parent_station of at least 2 nodes (an entrance/exit and a platform).

5.1.3. Boarding areas

Every boarding area should be connected to at least another one (location_type 4), as a platform is always made of two boarding areas at least (beginning and end of the platform).

There should always be a route, even made of several pathways, from every entrance/exit to at least a boarding area.

5.2. Errors

5.2.1. Stops

Every station, station entrance/exit and platform (location_type 0, 1 and 2) should have GPS coordinates associated.

5.3. Pathways

Every mechanical pathway (pathway_mode 3, 4 and 5) should have a unique pathway_code.

There should be no bi-directional escalators or travelators (pathway_mode 3 and 4).

There should be no escalators with mechanical_stair_count = 0.

There should be no bi-directional fare gates or exit fare gates (pathway_mode 6 and 7).

There should be no stairs with stair_count = 0.

There should be no commas in every free text fields (signpostments, instructions, names).

6. Conclusion

Throughout our 200 hours of field work, we mostly learned that common sense is the key success factor in such a project. **The main goal of such a dataset is to provide the most accurate information in the simplest way to the largest number of users.** Modern digital tools did change the way we travel and we plan our trips. As they are largely adopted by active and younger users, we do believe they will be anchored in our future daily live. Although to date, these digital tools are mostly designed by active people for active people. Digitalisation has made our trips easier, but only for a part of the population. People with disabilities (e.g. wheelchair, visual impairment), difficulties to travel (e.g. luggage, stroller), anxiety in public transportation need even more than others access to accurate travel information before and during their trips. Facilitating the access to public transportation to all users is the next challenge for the transit planning softwares.

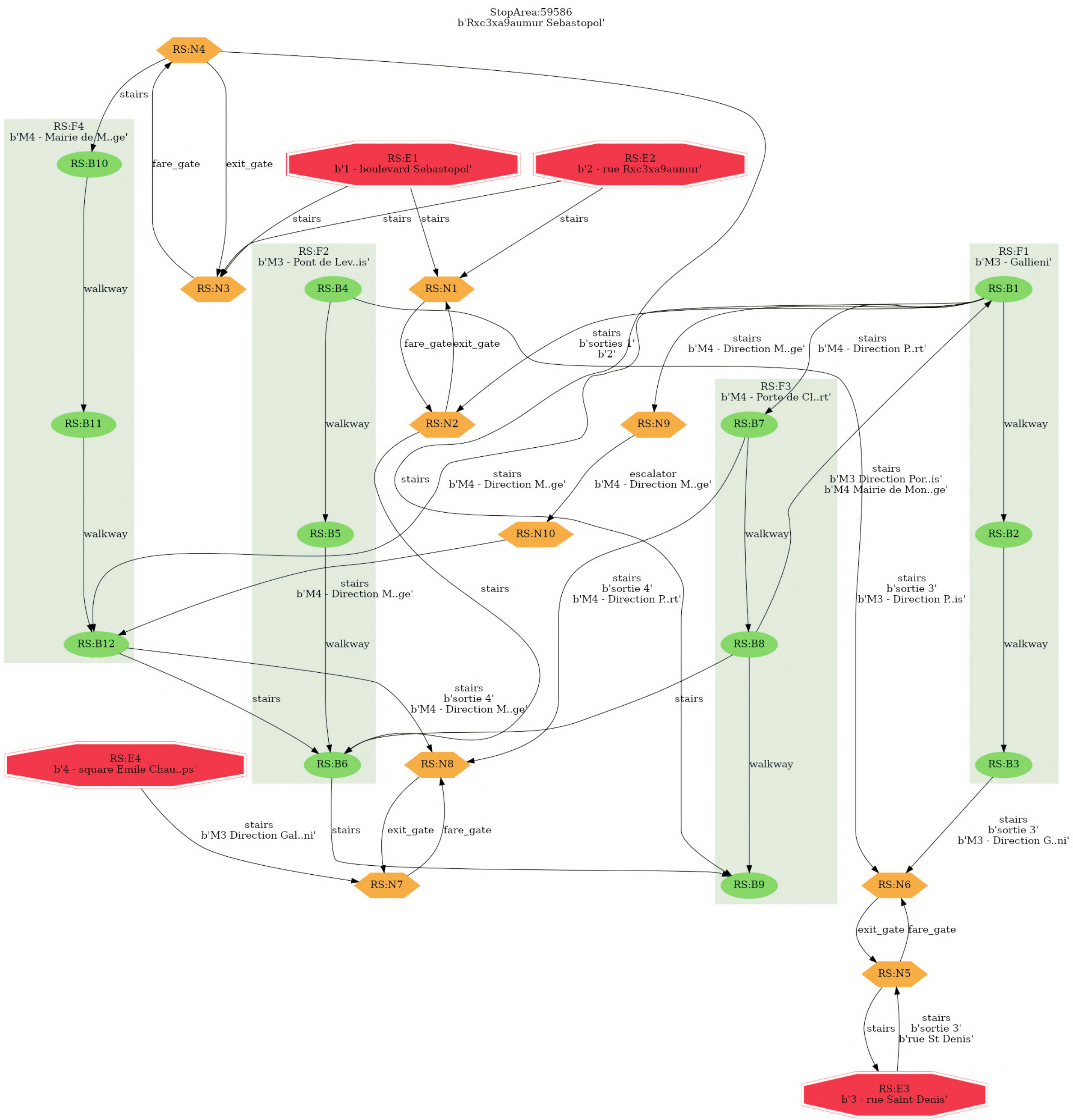
Throughout our 200 hours of field work, we also learned that data collection in large hubs can potentially lead to insanity. Surveying information in the field is an ancestral practice, although surveying this information within a subway or train system is slightly more challenging: constant flow of people in the pathways, underground situation, regular noise and announcements from the operators and trains, etc. Therefore, we developed a data collection software to ease this process. With this software, we divided our time in the field by two to three, we reduced the number of errors by pre-configuring a certain number of fields, and we eased the encoding process by limiting additional manual input after the field work. We are currently developing more features to optimize this data collection even more. This software can be used by any collectors, and is meant to be ease of use. Please feel free to contact us at *datalab@kisio*.com for more information.

Throughout our 200 hours of field work, we finally learned that graph modeling demands expert qualifications. This stage of the process can be very challenging to accomplish since it requires architectural, mathematical and GTFS understanding. We believe this step is extremely important because it will condition the data collection and therefore the quality of the future dataset. Public authorities and transporters should approach this stage with qualified professionals.

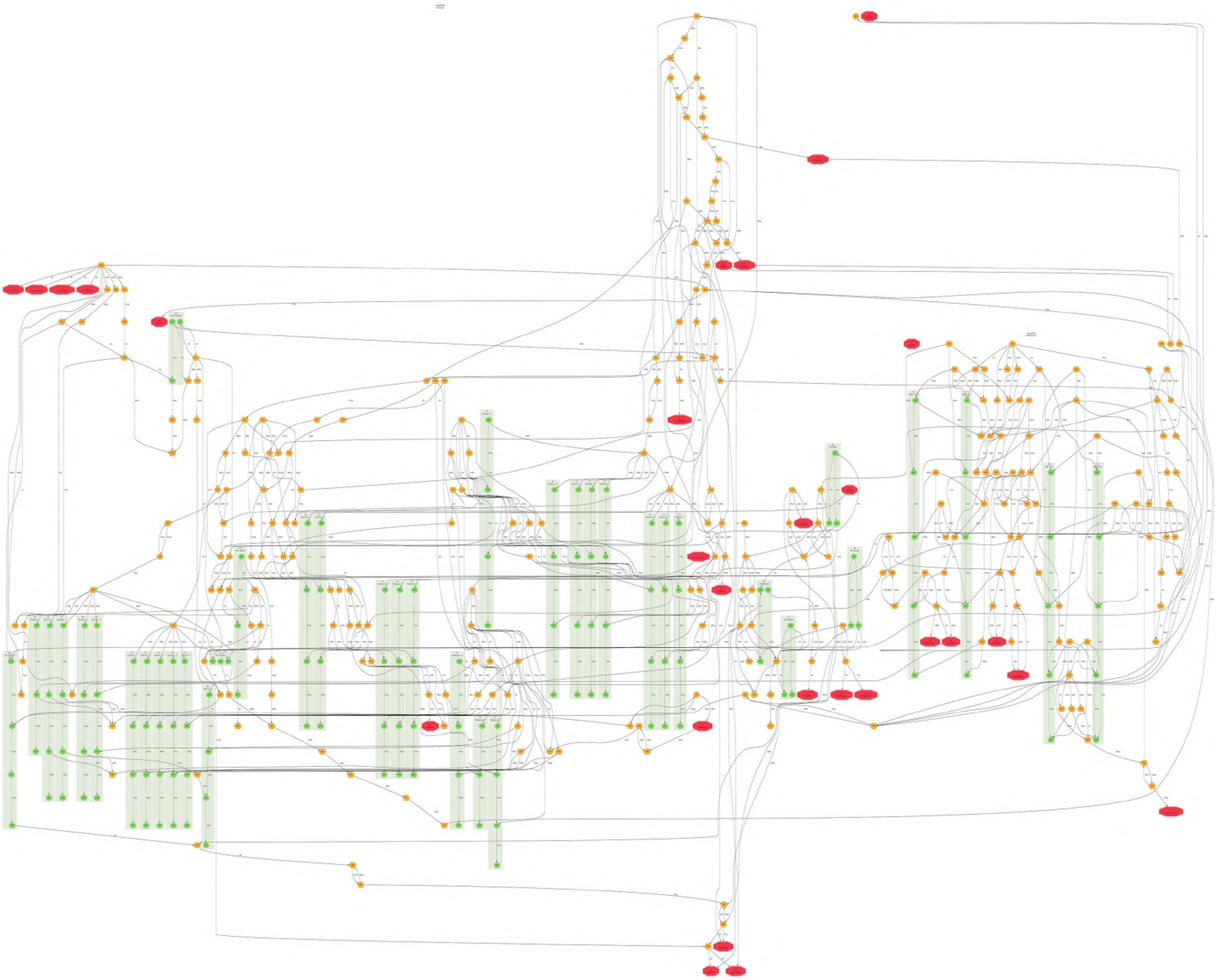
§ APPENDIX

Example of graph modeling

◊ Paris, Réaumur Sébastopol, PE = 16



◊ Paris, Gare Saint-Lazare & Haussman Saint-Lazare, PE= 1064



Example of GTFS-pathways

◊ Paris, Réaumur Sébastopol

The document is downloadable at this link : https://kisioanalysis.io/gtfs-pathways

◊ Paris, Gare Saint-Lazare & Haussman Saint-Lazare

The document is downloadable at this link : https://kisioanalysis.io/gtfs-pathways

§ INDEX

MobilityData

Kisio

Google