

GTFS-Ticketing

(fka GTFS-DeepLinks)

<http://bit.ly/gtfs-ticketing>

This document is a collaborative working tool for a GTFS extension proposal. As a living document, content may evolve based on community feedback. Therefore, any implementation may have to be updated. The status color scheme [A1] tracks the stages at which the contents of the GTFS extension proposal evolve. If you have any questions, please reach out to specifications@mobilitydata.org.

Goals

In transportation apps, riders search for O to D trips, check the different options, and choose among them the one they want to take.

But once their decision is made, a ticket has to be bought to make the ride. A growing number of public transit agencies already have ticketing apps allowing the rider to pay for such a ticket, but currently, links between transportation apps and ticketing apps have to be ad-hoc.

The goal of this proposal is to standardize the deep links between transportation apps and ticketing apps or websites, allowing the rider to have the following smooth experience:

- Search for O to D in transportation app.
- Choose the option they want in the transportation app.
- Be redirected in the ticketing app to buy the ticket, without having to re-enter the information about the OD nor the choice they made.
- Buy the ticket.

Requirements

Field `departure_time` in `stop_times.txt` are required.

Field Types added

- URI - A fully qualified URI that includes the scheme, and any special characters in the URI must be correctly escaped. See RFC 3986 “Uniform Resource Identifier (URI): Generic Syntax” at <https://tools.ietf.org/html/rfc3986> for a description of how to create fully qualified URI values.

Files extended or added

File Name	State	Defines
agency.txt	Extended	Adds agency.deep_link_id

routes.txt	Extended	Adds routes.deep_link_id
trips.txt	Extended	Adds trips.trip_ticketing_id
stop_times.txt	Extended	Adds stop_times.stop_time_ticketing_id
ticketing_identifiers.txt	Added	
ticketing_deep_links.txt	Added	

Tables Definitions

agency.txt (file extended)

Field Name	Details
ticketing_deep_link_id	(ID from ticketing_deep_links.txt, Optional) Defines the deep link that should be used for this agency. Can be overwritten in routes.txt

routes.txt (file extended)

Field Name	Details
ticketing_deep_link_id	(ID from ticketing_deep_links.txt, Optional) Defines the deep link that should be used for this route. If provided, overwrites the one in agency.txt

trips.txt (file extended)

Field Name	Details
ticketing_trip_id	(ID, Optional) Identifier that should be passed in the deep link. Doesn't have to be unique. Fallbacks on the trip_id.
ticketing_type	(Enum, Optional) Define if the ticketing through deep link is available for this trip: <ul style="list-style-type: none"> - 0 or empty: assumed to be: <ul style="list-style-type: none"> - Available if a ticketing_deep_link_id is defined on the route or agency. - Unavailable otherwise. - 1: Unavailable.

stop_times.txt (file extended)

Field Name	Details
ticketing_stop_time_id	(ID, Optional) Identifier that should be passed in the deep link. Doesn't have to be unique, even within the trip.

	Fallbacks on the <code>ticketing_stop_id</code> if defined. If not <code>ticketing_stop_id</code> is defined, fallbacks on the <code>stop_sequence</code> .
<code>ticketing_type</code>	(Enum, Optional) Define if the ticketing through deep link is available for this trip. If defined, override <ul style="list-style-type: none"> - Empty: As defined by <code>trip.ticketing_type</code>. - 0: assumed to be: <ul style="list-style-type: none"> - Available if a <code>ticketing_deep_link_id</code> is defined on the route or agency. - Unavailable otherwise. - 1: Unavailable.

ticketing_identifiers.txt (file added)

The most common case for ticketing is that every stop has an identifier that is used to process billing. This identifier isn't shared across agencies, so if two agencies are serving the same stop, this

Field Name	Details
<code>ticketing_stop_id</code>	(ID, Required) Define a default ticketing ID for this stop of this agency.
<code>stop_id</code>	(ID from <code>stops.txt</code> , Required) The stop for which the default <code>ticketing_stop_id</code> is defined.
<code>agency_id</code>	(ID from <code>agency.txt</code> , Required) The agency of the stop for which the default <code>ticketing_stop_id</code> is defined.

ticketing_deep_links.txt (file added)

Field Name	Details
<code>ticketing_deep_link_id</code>	(ID, Required) Define an ID for the deep link.
<code>web_url</code>	(URL, Optional) The URL to visit for deep linking. Translations can be supplied via <code>translations.txt</code> to send riders to locale-specific URLs.
<code>android_intent_uri</code>	(URI, Optional) The URI to pass to a native Android app with an <code>android.intent.action.VIEW</code> Android intent. If empty, it means no deep-linking to a native Android app. See https://developer.android.com/training/app-links/deep-linking for more information about deep links on Android. This field is not translatable via <code>translations.txt</code> .
<code>ios_universal_</code>	(URL, Optional) The universal link URL to invoke on iOS.

link_url	<p>The iOS mobile app must be designed to intercept the URL, failing which the URL will be directly opened on the browser as a fallback.</p> <p>If different fallback URLs are desired for different languages, then the appropriate translations must be provided in translations.txt, and the iOS app must intercept all possible translations.</p> <p>If empty, deep linking will not be shown on iOS.</p> <p>See https://developer.apple.com/ios/universal-links/ for more information about Universal Links on iOS.</p> <p>This field is not translatable via translations.txt.</p>
----------	---

Field Placeholders in the API call

The URLs defined above will be called with the following arguments:

Fields	Details
service_date	(Date, Required , Repeatable) Service day of the trip.
ticketing_trip_id	(ID from trips.txt, Required , Repeatable) Identifier of the trip.
from_ticketing_stop_time_id	(ID from stop_times.txt, Required , Repeatable) Identifier of the stop_time at which the leg starts.
to_ticketing_stop_time_id	(ID from stop_times.txt, Required , Repeatable) Identifier of the stop_time at which the leg ends.
boarding_time	<p>(ISO 8601, Required, Repeatable) Time of departure (departure_time) of the stop_time at which the leg starts.</p> <p>The time format must be converted to in ISO 8601, with the specific following format:</p> <p style="text-align: center;">YYYY-MM-DDThh:mm:ss±hh:mm</p> <p>All the times below are the same, in different time-zones:</p> <ul style="list-style-type: none"> - In London, UK: 2019-07-29T18:26:00+01:00 - In NYC, US-NY: 2019-07-29T13:26:00-04:00 - In Tokyo, JP: 2019-07-30T02:26:00+09:00

Examples

If the legs are:

- On service date 20190716, trip ticketing id ti1 from stop time ticketing id 11 to stop time ticketing id 12.

- On service date 20190716, trip ticketing id ti2 from stop time ticketing id 21 to stop time ticketing id 22

And the web_url is: <http://myawesomebookingwebsite.com>

The call will be done to:

[http://myawesomebookingwebsite.com?service_date\[\]=20190716&ticketing_trip_id\[\]=ti1&from_ticketing_stop_time_id\[\]=11&to_ticketing_stop_time_id\[\]=12&service_date\[\]=20190716&ticketing_trip_id\[\]=ti2&from_ticketing_stop_time_id\[\]=21&to_ticketing_stop_time_id\[\]=22](http://myawesomebookingwebsite.com?service_date[]=20190716&ticketing_trip_id[]=ti1&from_ticketing_stop_time_id[]=11&to_ticketing_stop_time_id[]=12&service_date[]=20190716&ticketing_trip_id[]=ti2&from_ticketing_stop_time_id[]=21&to_ticketing_stop_time_id[]=22)

Train number example

Trips with specific ids (e.g. SNCF trains identified by train numbers). Values in bold are the ones used in the web call described below the files.

```
stop.txt
```

```
stop_id,stop_name  
si1,"Paris Gare-de-Lyon"  
si2,"Lyon Part-Dieu"
```

```
routes.txt
```

```
route_id,route_long_name,ticketing_deep_link_id  
ri1,"TGV inOui Paris-Lyon",td11
```

```
trips.txt
```

```
trip_id,service_id,route_id,trip_short_name,ticketing_trip_id  
ti1,everyday,ri1,"TGV INOUI 6603",FR_SNCF_6603  
ti2,everyday,ri1,"TGV INOUI 6681",FR_SNCF_6681  
ti3,everyday,ri1,"TGV INOUI 6607",FR_SNCF_6607
```

```
stop_times.txt
```

```
trip_id,stop_sequence,stop_id,arrival_time,departure_time,ticketing_stop_time_id  
ti1,1,si1,06:59:00,06:59:00,4924  
ti1,2,si2,08:56:00,08:56:00,4676  
ti2,1,si1,07:53:00,07:53:00,4924  
ti2,2,si2,10:00:00,10:00:00,4676  
ti3,1,si1,08:59:00,08:59:00,4924  
ti3,2,si2,10:56:00,10:56:00,4676
```

```
ticketing_deep_links.txt
```

```
ticketing_deep_link_id,web_url,android_intent_url,ios_universal_url
tdl1,https://www.thetrainline.com/api/gtfs/web,https://www.thetrainline.com/api/gtfs/android,https://www.thetrainline.com/api/gtfs/ios
```

The call for the web, if requested on the 2019-07-19, will be:

[http://www.thetrainline.com/api/gtfs/web?service_date\[\]=20190719&ticketing_trip_id\[\]=FR_SNCF_6603&from_ticketing_stop_time_id\[\]=4924&to_ticketing_stop_time_id\[\]=4676](http://www.thetrainline.com/api/gtfs/web?service_date[]=20190719&ticketing_trip_id[]=FR_SNCF_6603&from_ticketing_stop_time_id[]=4924&to_ticketing_stop_time_id[]=4676)

Peak & off-peak hours fares when distance based fares

peak/non-peak hour fares where the fare depends on the source+destination and the departure time?

```
stop.txt
```

```
stop_id,stop_name
si1,"Town Hall"
si2,"Central"
si2,"Airport"
```

```
routes.txt
```

```
route_id,route_short_name,route_long_name,ticketing_deep_link_id
ri1,T8,"Airport & South Line",tdl1
```

```
trips.txt
```

```
trip_id,service_id,route_id,ticketing_trip_id
ti01,everyday,ri1,T8_offpeak
ti02,everyday,ri1,T8_offpeak
ti11,everyday,ri1,T8_peak
ti12,everyday,ri1,T8_peak
ti21,everyday,ri1,T8_offpeak
ti22,everyday,ri1,T8_offpeak
```

```
stop_times.txt
```

```
trip_id,stop_sequence,stop_id,arrival_time,departure_time,shape_dist_traveled,ticketing_stop_time_id
ti01,1,si1,06:00:00,06:00:00,0,si1
ti01,2,si2,06:05:00,06:05:00,10,si2
ti01,3,si3,06:10:00,06:10:00,20,si3

ti02,1,si1,06:30:00,06:30:00,0,si1
ti02,2,si2,06:35:00,06:35:00,10,si2
ti02,3,si3,06:40:00,06:40:00,20,si3
```

```
ti11,1,si1,07:00:00,07:00:00,0,si1
ti11,2,si2,07:05:00,07:05:00,10,si2
ti11,3,si3,07:10:00,07:10:00,20,si3
```

```
ti12,1,si1,07:30:00,07:30:00,0,si1
ti12,2,si2,07:35:00,07:35:00,10,si2
ti12,3,si3,07:40:00,07:40:00,20,si3
```

```
ti21,1,si1,08:00:00,08:00:00,0,si1
ti21,2,si2,08:05:00,08:05:00,10,si2
ti21,3,si3,08:10:00,08:10:00,20,si3
```

```
ti22,1,si1,08:30:00,08:30:00,0,si1
ti22,2,si2,08:35:00,08:35:00,10,si2
ti22,3,si3,08:40:00,08:40:00,20,si3
```

```
ticketing_deep_links.txt
```

```
ticketing_deep_link_id,web_url,android_intent_url,ios_universal_url
tdl1,https://transportnsw.info/api/gtfs/web,https://transportnsw.info/api/gtfs/android,https://transportnsw.info/api/gtfs/ios
```

The call for the web, if requested on the 2019-07-19 for trip ti11, will be:

[http://www.thetrainline.com/api/gtfs/web?service_date\[\]=20190719&ticketing_trip_id\[\]=T8_peak&from_ticketing_stop_time_id\[\]=si1&to_ticketing_stop_time_id\[\]=si3](http://www.thetrainline.com/api/gtfs/web?service_date[]=20190719&ticketing_trip_id[]=T8_peak&from_ticketing_stop_time_id[]=si1&to_ticketing_stop_time_id[]=si3)

Open questions

[Open Question A] Definition of boarding_time

With the quadruplet of values:

- service_date
- ticketing_trip_id
- from_ticketing_stop_time_id
- to_ticketing_stop_time_id

... the trip can be uniquely identified. But for simplification reasons, some agencies may prefer to provide the boarding time along with, e.g., the route_id as ticketing_trip_id. There are four different ways to define such boarding time:

- **[Option A1] The from_stop_time.departure_time value: but it may be not defined in the GTFS for this stop_time.**
- ~~[Option A2] The real-time override of the departure time, either as departure.delay or as departure.time: but it doesn't exist in some cases, and even when defined, the ticketing app may not be aware of it.~~
- ~~[Option A3] The real-actual boarding time: but it can be only be known after the fact.~~
- ~~[Option A4] The intended departure time, known in advance: but it may be completely off.~~

Sean: The easiest would be to choose option A1, and require the agency to provide stop_time.departure_time for that record. If the agency currently isn't providing a stop_time.departure_time value for that record, they would need to modify their GTFS stop_times.txt data to add the timepoint field and set timepoint=0 for interpolated times (new values they would need to add) and timepoint=1 for timepoints (the times from the existing stop_times.txt file).

=> Based on community feedback, we'll go with Option A1: departure_time.

GTFS-DeepLinks with callbacks [WORK-IN-PROGRESS]

Goal

Once the deep linked as been followed, the original trip planning app may be interested to know what happened: Did the rider successfully book a ticket for the trip they wanted? Did they even pay for it? Or was the trip full? Or canceled?

The proposal below offers to extend **GTFS-DeepLinks** to define callbacks providing such information.

Requirement

Extends **GTFS-DeepLinks**.

Fields in the API call

Additionally to the fields defined in **GTFS-DeepLinks**, extra fields would be allowed:

Fields	Details
callback_id	(String, Optional) Identifier provided by the apps which makes the call, to be passed back in callbacks.
callback_url	(URL, Optional) URL to be called as callback by the ticketing app.

Fields in the API callback

If a transaction is settled in the ticketing app, a callback can be called with the following arguments:

Fields	Details
callback_id	(String, Required) Identifier provided by the app which made the call.
service_date	Same definition as the call.
ticketing_trip_id	Same definition as the call.
from_ticketing_stop_time_id	Same definition as the call.

to_ticketing_stop_time_id	Same definition as the call.
status	(Enum, Required) Defines what has been done in the ticketing app. Allowed values are: <ul style="list-style-type: none"> - 21: Successfully booked - 22: Successfully paid - 23: Successfully booked and paid - 41: Error, some ticketing identifiers unknown - 42: Error, request contains canceled leg(s) - 43: Error, request contains full leg(s)

Appendices

[A1] Status color scheme

This document is a working tool for a GTFS extension proposal. As a living and collaborative document, content may evolve based on community feedback. The scheme below tracks the stages at which the contents of the GTFS extension proposal evolve. If you have any questions, please reach out to specifications@mobilitydata.org.

Working draft	<ul style="list-style-type: none"> - This content has not yet been adopted or put to a GitHub pull request (PR) on google/transit. - It may evolve based on community feedback, and is therefore subject to change. - Any implementation may have to be updated with iterations or official adoption.
Proposal	<ul style="list-style-type: none"> - This content is currently a GTFS extension proposal via a GitHub pull request (PR) on google/transit. - The hyperlinked to its relevant PR is embedded in the text. - The PR should serve as the sole venue for discussion and iteration. - Text of this color may no longer be up to date.
Adopted	<ul style="list-style-type: none"> - This content has been put to a GitHub pull request (PR) on google/transit as a GTFS extension proposal, and has been officially adopted. - Text of this color may no longer be up to date. - Any references to this material must be made from the official reference.md.