

GENERAL TRANSIT FEED SPECIFICATION (GTFS) PROFILE SWITZERLAND

Systemaufgaben Kundeninformation (SKI) – Team SKI+

<https://www.öv-info.ch/de/> & <https://opentransportdata.swiss/>

Status	Draft
Version	0.17
Date of publication	2024-12-13
Authors	Martin Griesser (SBB NDx), David Rudi (SBB SKI+), Matthias Günter (SBB SKI+)
Copyright	CC BY 4.0

Document information

Description	This document contains information, assessments, and explanations on GTFS used in Switzerland in the Swiss profile.
Target Group	People and organizations who want to obtain or provide GTFS data for the whole of Switzerland via the opentransportdata.swiss .
Repository	https://www.tp-info.ch/de/datenmanagement/ski/standards-der-ski

History

Version	Status	Remarks	Author	Valid from
0.17	Draft	Added feed_version in GTFS RT (p. 36). Added SJYID to GTFS Static and RT (p. 20 + 21 (static); p. 54).	D. Rudi	2024-12-13

Version	Status	Remarks	Author	Valid from
		Added modified_trip to GTFS RT (p. 53). Explanation of how to introduce changes to GTFS (p. 6)		
0.16	Draft	Draft for publication and discussion with included reviews and amendments by various instances, such as Tu-Tho Thai, André Murbach, and Mobility Data.	D. Rudi	2024-02-30
0.15	Draft	Draft for publication and discussion	D. Rudi M. Günter	2024-01-31
0.14	Draft	Draft for publication	M. Griesser D. Rudi M. Günter	2023-09-21
0.3	Draft	Version for discussion including GTFS Flex	M. Griesser	2023-08-07
0.1	Draft	First version	M. Griesser	2023-04-11

Content

1	What is it all about?	5
2	Description and Context	5
3	Who is responsible?	5
4	Important Links	6
5	Technology Stack and Standards	7
5.1	GTFS-Schedule	7
5.2	GTFS-Realtime	7
6	Usage	8
7	Datasets / Services Switzerland	8
8	Assessment	9
9	Conclusions	9
9.1	Comparison to other Standards (for example SIRI):	10
10	Specification and Recommendation	10
11	GTFS - Swiss Profile	12
11.1	GTFS-Schedule (Swiss Profile)	12
11.1.1	Useful Links:	12
11.1.2	Short Introduction	12
11.1.3	Dataset Files	13
11.2	GTFS-Realtime (Swiss Profile)	32
11.2.1	Introduction	32
11.2.2	Swiss Specific Information	33
11.2.3	Full Element Index	34
11.2.4	General Elements	35
11.2.5	Service Alerts (Provided by SBB)	39
11.2.6	Vehicle Position	50
11.2.7	Trip Updates (Provided by SBB)	51
12	Extension under Development: GTFS-Flex	63
12.1	GTFS-Flex	63
12.1.1	Links	63
12.1.2	Description	63

12.1.3	Data-Set Files	64
12.2	Extension under Development: GTFS-Flex (Swiss Profile).....	74
13	Possible other Extensions in the near Future	75

1 What is it all about?

This document is intended to lay the foundation for an efficient, standardized exchange of static and realtime information in terms of transport timetables and related geographic information based on GTFS.

The profile should serve for the following purposes:

- provide sufficient guidance for a coherent, interoperable adoption of GTFS in Switzerland,
- allow for a quick and easy GTFS connection
- lay the foundation for an effective and efficient data exchange infrastructure.

The profile is based on the standard GTFS.

2 Description and Context

General Transit Feed Specification (GTFS) is a digital exchange format was originally developed by Bibiana McHugh (TriMet, Portland, Oregon, USA). She was later supported by Google to make the specification usable by data consumers & trip planning apps. Google for public transport timetables and related geographic information, such as stop locations. The data is provided by transport companies and published as a collected database on the Open Data platform öV-Schweiz in GTFS format. The data is used for the development of applications related to public transport, such as timetable queries.

GTFS-Schedule describes public transportation in a particular region at a specific point in time. It contains information such as schedules, routes, stops, and fare information. This information typically changes infrequently and is usually maintained and updated by transport agencies. The aggregated static data feed of the whole Switzerland are available on the Open Data platform in GTFS-Schedule.

Data expressed in GTFS-Realtime, on the other hand, is provided in realtime and includes information such as delays, disruptions, vehicle positions, and estimated arrival times. It is generated and updated in realtime by transportation agencies or their partners.

In Switzerland, GTFS is used by many transportation agencies and public transportation authorities to provide data about public transportation.

Currently, GTFS version 1.0 is made available in Switzerland. In the future, extensions and adaptations to Google's range of functions are planned. This will be covered in sections 12 and 13.

3 Who is responsible?

GTFS was originally developed by TriMet & Google and released as an open-source project to facilitate the integration of public transportation into maps and trip planning applications like Google Maps. While Google still plays an important role in the continued development of GTFS, the project is now supported by a broader community of developers and organizations.

Responsibility for the continued development of GTFS lies with the community that hosts the project on GitHub and works on its development. Proposed changes and discussions about the specification are led by the GTFS developer community, which is composed of a variety of organizations

and individuals, including transportation agencies, developers, researchers, and advocates. The community is managed by MobilityData, a non-profit organization who has been given the maintainer role of the open-source project.

Additionally, there are also organizations like the Open Transit Standards Working Group that focus on the development and promotion of open standards in public transportation, including GTFS.

[GitHub - google/transit \(https://github.com/google/transit\)](https://github.com/google/transit)

The overall procedure for introducing a change to any of the GTFS specifications was defined as follows:

- Here is the short form version of how to proceed assuming you will assume the role of advocate:
 1. Create a Git Branch
 2. Open a PR
 3. Announce the PR on the [GTFS Changes Google Groups](#)
 4. Host a discussion of a minimum duration of 7 days. Can be held on here or a Zoom. (MobilityData can assist you with this if you find the necessity)
 5. The changes must be implemented by at least one consumer and one producer before calling a vote. Proof of implementations must be demonstrated in a public facing manner.
 6. Call a vote on GitHub. It lasts 14 days. There is a unanimous consensus yes with at least 3 votes requirement to pass. The advocate's vote doesn't count.
 7. Voting results must be announced on [GTFS Changes Google Groups](#)
 8. If passed the changes are merged.
- Here is [the full GTFS Spec amendment process](#)

In Switzerland, GTFS is being further developed and adapted to Swiss needs by the SKI+ team as part of the Open Journey Planner project. However, HRDF and NeTEx are the main timetable format and for realtime VDV 454/453 and SIRI are used. SKI+ moves in the direction of NeTEx and SIRI.

4 Important Links

No	Description	Link
1	Best Practices Guideline	https://gtfs.org/schedule/best-practices/
2	Google Description of the Schedule-Service	https://developers.google.com/transit/gtfs?hl=de
3	Google Description of the Realtime-Service	https://developers.google.com/transit/gtfs-realtime?hl=de
4	GitHub Community Link	https://github.com/google/transit
5	GTFS Realtime Description on the Open Transport Data platform	https://opentransportdata.swiss/de/cookbook/gtfs-rt/

6	GTFS-Schedule Description on the Open Transport Data platform	https://opentransportdata.swiss/de/cookbook/gtfs/
7	GTFS-Schedule Data for Switzerland	https://opentransportdata.swiss/de/dataset/timetable-2023-gtfs2020
8	GTFS FAQ	https://opentransportdata.swiss/de/faq/fahrplaene-gtfs/
9	GTFS Mapping with HRDF	https://opentransportdata.swiss/de/cookbook/verwendung-von-hrdf-fahrplaenen-zusammen-mit-gtfs-rt/

5 Technology Stack and Standards

The two services GTFS-Schedule and GTFS-Realtime are based on different technologies and stacks:

5.1 GTFS-Schedule

Some of the common technologies used to implement GTFS include:

- CSV files: GTFS uses CSV files to represent data in tabular form.
- Databases: GTFS data can be stored in relational databases or NoSQL databases such as MongoDB.
- RESTful APIs: Some transportation agencies and public transit authorities provide GTFS data via RESTful APIs, which enable developers to access the data and integrate it into applications.
- JSON: Although GTFS is based on CSV files by default, there are also ways to convert the data into JSON format.
- GeoJSON: As part of the changes done for GTFS-Flex geoJSON was introduced to represent geographical data.
- Mapping tools: There are also a variety of mapping tools that enable developers to integrate GTFS data into mapping applications such as Google Maps.

5.2 GTFS-Realtime

- ProtoBuf: GTFS Realtime is typically exchanged in the ProtoBuf format, as it provides an efficient and compact way to transmit and store realtime data. ProtoBuf also supports multilingual libraries that make it easier to read and write ProtoBuf data in a variety of programming languages.
- RESTful APIs: Some transportation agencies and public transit authorities provide GTFS Realtime data via RESTful APIs, which enable developers to access the data and integrate it into applications.
- WebSocket: WebSocket is a communication protocol used for transmitting realtime information. It provides a bidirectional connection between a client and a server, enabling data to be transmitted in realtime.

- Mapping tools: There are also mapping tools that enable developers to integrate GTFS Realtime data into mapping applications such as Google Map

6 Usage

General Usage

- GTFS is widely used worldwide and adopted by many public transit agencies to provide information about public transportation. In Switzerland, GTFS is supported by most public transit agencies, including Swiss Federal Railways (SBB), Zürcher Verkehrsbetriebe (VBZ), and Basler Verkehrsbetriebe (BVB). GTFS is also commonly used in the German-speaking region (DACH), such as by the Wiener Linien and Munich transport companies.
- GTFS is widely used in the United States by many public transportation operators, including the New York Metropolitan Transportation Authority (MTA), the Chicago Transit Authority (CTA), and the San Francisco Municipal Transportation Agency (SFMTA). In the US, GTFS is often used as a basis for mobile applications and online tools that provide public transportation information.
- In China, GTFS is also increasingly being used, especially in larger cities such as Beijing and Shanghai. The Chinese government has also been investing more in the modernization of public transportation in recent years, and has been relying on technologies such as GTFS.

Overall, GTFS has become an important tool for public transit and has contributed to making public transit data more accessible and promoting the integration of public transit into applications and services.

GTFS has a focus on pure customer information. All other formats are used for the exchange of planned data and realtime information between transport companies to ensure an end-to-end travel chain independent of the executing transport company. Consequently, standards like VDV, SIRI, NeTEx contain significantly more information.

GTFS is a suitable standard for basic use cases regarding passenger information in public transport. For more complex requirements (which need more detailed data) that involve, for example, disruptions, detours, substitute vehicles, specific parameters like capacity, accessibility or operational data, advanced standards like VDV-454, NeTEx, and SIRI are more appropriate. The choice of standard strongly depends on the specific requirements and goals of the project and interface.

More details can be found in chapter 9.

7 Datasets / Services Switzerland

GTFS-Schedule export for whole Switzerland can be found here:

<https://opentransportdata.swiss/de/group/timetables-gtfs>

A new upload is available weekly:

<https://opentransportdata.swiss/de/dataset/timetable-2023-gtfs2020>

A detailed sample with all elements could be found in the Google documentation:

<https://developers.google.com/transit/gtfs-realtime/examples/trip-updates-full?hl=de>

8 Assessment

Rough, qualitative evaluation or assessment of the SKI+ team ¹

P1 international	+++	GTFS is widely used and recognized as an international standard for public transit data exchange
P2 open	+++	GTFS is an open and free format, allowing for easy access and use by developers and public transit agencies
P3 simple	+++	GTFS is designed with simplicity in mind, making it easy for transit agencies to implement and maintain
P4 established	+++	GTFS has been in use for over a decade and is widely adopted by public transit agencies around the world
P5 evolutionary	++	GTFS continues to evolve with the changing needs of public transit, with regular updates and new features being added
P6 of high quality	+++	GTFS is a well-designed and well-documented format, with a high level of quality and reliability
P7 compliant	+++	GTFS is compliant with relevant international standards for public transit data exchange.
P8 interpretation-free	++	GTFS is designed to be unambiguous and interpretation-free, reducing the potential for errors or misinterpretation

9 Conclusions

In conclusion, GTFS is a widely used and important tool for public transportation providers worldwide, including in Switzerland and the DACH region. It enables the easy sharing and integration of public transportation data in various applications and services, and is established, of high quality, and compliant. The availability of GTFS data has greatly improved the accessibility and usability of public transportation information for travellers and developers alike. The focus when using GTFS data should be on the customer and their passenger information. GTFS is less suitable as an exchange format for realtime information or operational data between transport companies.

¹ The principles are defined in the standardization document for NADIM. Meaning: 0 = not existing, += low, ++ = medium, +++ = high.

9.1 Comparison to other Standards (for example SIRI):

- SIRI ET/PT is a highly flexible standard that allows for a wide range of realtime information to be exchanged between various systems and devices involved in public transport operations. This includes information such as vehicle positions, arrival and departure times, and service disruptions, as well as more detailed information about vehicle routes, stops, and schedules.
- One key advantage of SIRI ET/PT is its ability to provide a detailed and up-to-date view of the entire journey of a vehicle, down to the smallest detail. This can include information about detours, delays, and additional journeys, as well as realtime updates about the status of individual vehicles and services. This level of detail can be very valuable for transport authorities, as it allows them to closely monitor the performance of their services and make informed decisions about how to optimize them.
- At the same time, SIRI ET/PT can also provide valuable information to passengers, such as realtime updates about the status of their services, including delays and cancellations. This can help passengers better plan their journeys and avoid disruptions, which can improve the overall passenger experience.
- GTFS-Realtime, on the other hand, is more limited in scope than SIRI ET/PT, as it is primarily designed to provide realtime updates to the static schedule information provided by GTFS. However, it does offer several advantages over SIRI ET/PT in terms of simplicity and ease of use. The format is relatively simple and easy to understand, which can make it easier for transport authorities and developers to work with.
- One of the key advantages of GTFS-Realtime is that it is designed specifically for providing realtime information to passengers. It allows for realtime updates about the status of services to be displayed on digital signage, mobile apps, and other passenger information systems. This can be very valuable for passengers, as it allows them to stay informed about the status of their services and make informed decisions about how to plan their journeys.

In general, GTFS is from SKI+ (SBB) point of view a format to deliver/publish data (target group: passengers, apps, tools). For data exchange between PT operators or entities (data collection with the central data hubs), proven interfaces such as SIRI, VDV or NeTEx should be used. They allow for a greater level of detail in the information made available. This is important for further processing.

10 Specification and Recommendation

Basically, we orientate ourselves on the specifications of Google and the GTFS community and use the functions that are needed for a consistent passenger information scheme in Switzerland. In addition to the Google definition, the focus is on the unification and harmonization of data and thus the implementation of the SID4PT (Swiss Identification for Public Transportation), which will be also part of the GTFS implementations.

SBB (SKI+) pursues the strategy of collecting all realtime information of the whole of Switzerland and making it available to all consumers in different and target-oriented formats.

This also includes the conversion of VDV and SIRI as well as NeTEx data into the GTFS formats.

Consequently, SBB will make the following GTFS data freely available to all submitting companies

- GTFS-Schedule (planning data)
- GTFS-Realtime (Service Alerts, Trip Updates)
- GTFS-Flex

11 GTFS - Swiss Profile

The reference and starting point of this document is the GTFS documentation (version 2.0 of the document). This document is to be read together with the specification (Google / Best Practices / GitHub). It clarifies and details parts and offers restrictions to the original specification to improve interoperability.

The document lists the GTFS elements used in Switzerland. It also contains links to generally applicable implementation guidelines. This should give a good overview of the form in which data is made available in Switzerland. Important notice:

Used technologies for the **Swiss implementation**:

1. SBB provides the trip updates via a HTTPS-GET request
2. SBB provides the service alerts via a HTTPS-GET request
3. Data can be provided in the Protocol Buffer format or as Json (only for testing purposes)

For all elements, services or files the following information is available.



Used in Swiss implementation



Not used in Swiss implementation

11.1 GTFS-Schedule (Swiss Profile)

11.1.1 Useful Links:

4. Best Practices for the Implementation of GTFS-Schedule: [Best Practices - General Transit Feed Specification \(gtfs.org\) Best Practices](https://gtfs.org/schedule/best-practices/) (https://gtfs.org/schedule/best-practices/)
5. Detailed Description of the Google Implementation: [GTFS-Schedule - Überblick | Static Transit | Google Developers](https://developers.google.com/transit/gtfs?hl=de) (https://developers.google.com/transit/gtfs?hl=de)
6. Short Overview / Description of the GTFS-Schedule Service: [GTFS-Schedule Overview | Static Transit | Google Developers](https://developers.google.com/transit/gtfs?hl=de) (https://developers.google.com/transit/gtfs?hl=de)
7. Short Overview (Description of the GTFS Realtime Service: [GTFS Realtime Overview | Realtime Transit | Google Developers](https://developers.google.com/transit/gtfs-realtime?hl=de) (https://developers.google.com/transit/gtfs-realtime?hl=de)
8. GitHub Project for GTFS: [GitHub - google/transit](https://github.com/google/transit) (https://github.com/google/transit)
9. [FAQ to the Swiss Implementation](https://opentransportdata.swiss/de/faq/fahrplaene-gtfs/) (https://opentransportdata.swiss/de/faq/fahrplaene-gtfs/)
10. [Using HRDF timetable together with GTFS](https://opentransportdata.swiss/de/cook-book/verwendung-von-hrdf-fahrplaenen-zusammen-mit-gtfs-rt/) (https://opentransportdata.swiss/de/cook-book/verwendung-von-hrdf-fahrplaenen-zusammen-mit-gtfs-rt/)

11.1.2 Short Introduction

GTFS-Schedule is provided as a set of text files in ZIP format. Each file contains information about a particular aspect of the traffic information such as stops (stops.txt), routes (routes.txt), trips (trips.txt), agencies (agency.txt), and other schedule-related data. Some information is necessary

to provide valid GTFS data, others are optional and provide useful additional information (e.g., schedule changes on specific holidays = `calendar_dates.txt`).

11.1.3 Dataset Files

All available files for the Swiss Profile (GTFS-Schedule) can be found here: [Fahrpläne GTFS - Gruppen | Open-Data-Plattform Mobilität Schweiz \(opentransportdata.swiss\)](#)

No	Filename	Required Based on Google	Definition  Used in Swiss implementation  Not used in Swiss implementation
1	agency.txt	Required	 Used in Swiss implementation Transit agencies with service represented in this dataset
2	stops.txt	Required	 Used in Swiss implementation Stops where vehicles pick up or drop off riders. Also defines stations and station entrances.
3	routes.txt	Required	 Used in Swiss implementation Transit routes. A route is a group of trips that are displayed to riders as a single service
4	trips.txt	Required	 Used in Swiss implementation Trips for each route. A trip is a sequence of two or more stops that occur during a specific time period.
5	stop_times.txt	Required	 Used in Swiss implementation Times that a vehicle arrives at and departs from stops for each trip.
6	calendar.txt	Conditionally Required	 Used in Swiss implementation Service dates specified using a weekly schedule with start and end dates. This file is required unless all dates of service are defined in calendar_dates.txt .
7	calendar_dates.txt	Conditionally required	 Used in Swiss implementation Exceptions for the services defined in the calendar.txt . If calendar.txt is omitted, then calendar_dates.txt is required and must contain all dates of service.
8	fare_attributes.txt	Optional	 Not used in Swiss implementation
9	fare_rules.txt	Optional	 Not used in Swiss implementation
10	shapes.txt	Optional	 Not used in Swiss implementation We are considering the use. As our implementation is based on OSM, the shapes.txt will have to adhere to the OSM license.
11	frequencies.txt	Optional	 Not used in Swiss implementation
12	transfers.txt	Optional	 Used in Swiss implementation

			When calculating an itinerary, GTFS-consuming applications interpolate transfers based on allowable time and stop proximity
13	pathways.txt	Optional	✗ Not used in Swiss implementation We will consider it, as soon as the detailed stop modelling is complete. As our implementation is based on OSM, the pathways.txt will have to adhere to the OSM license.
14	levels.txt	Optional	✗ Not used in Swiss implementation
15	feed_info.txt	Conditionally required	✓ Used in Swiss implementation This file contains information about the dataset itself, rather than the services the dataset describes
16	translations.txt	Optional	✗ Not used in Swiss implementation
17	attributions.txt	Optional	✗ Not used in Swiss implementation

Figure 1: Overview GTFS-Schedule file set

Please refer to [Referenzdokument](#) | [Static Transit](#) | [Google Developers](#) for further information.

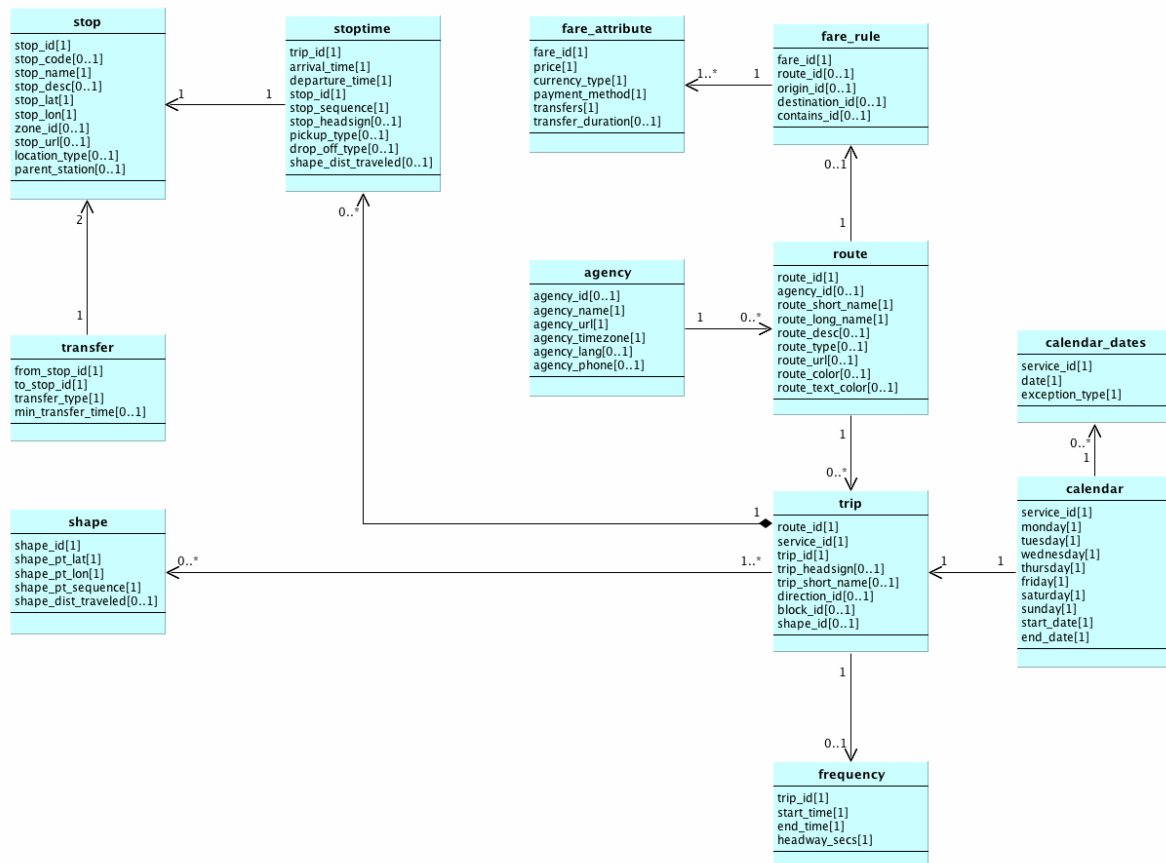


Figure 2: Connection between files according to Google

11.1.3.1 Agency.txt

An agency is the operator of public transport services. It is often a public agency. The operators are defined in the **agency.txt** file.

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs/reference#agencytxt Reference Static Transit Google Developers Best Practices - General Transit Feed Specification (gtfs.org)
---	--

11.1.3.1.1 File definitions: Required

Field Name	Required	Description ✔ Used in Swiss implementation ✘ Not used in Swiss implementation
agency_id	Conditionally Required	✔ Identifies a transit brand which is often synonymous with a transit agency. Currently we use the GO-Number out of DiDoK. In the future SBOID should be used.
agency_name	Required	✔ Full name of the transit agency.
agency_url	Required	✔ URL of the transit agency.
agency_time-zone	Required	✔ Time zone where the transit agency is located.
agency_lang	Optional	✔ Primary language used by this transit agency.
agency_phone	Optional	✔ A voice telephone number for the specified agency.
agency_fare_url	Optional	✘ URL of a web page that allows a rider to purchase tickets or other fare instruments for that agency online.
agency_email	Optional	✘ Email address actively monitored by the agency's customer service department.

Figure 3: Field definitions agency.txt

11.1.3.1.2 Example from OpenTransportData

agency_id,agency_name,agency_url,agency_timezone,agency_lang,agency_phone
"87_LEX","LEX","http://www.sbb.ch/","Europe/Berlin","DE","0848 44 66 88"
"11","Schweizerische Bundesbahnen SBB","http://www.sbb.ch/","Europe/Berlin","DE","0848 44 66 88"
"823","Basler Verkehrsbetriebe","http://www.sbb.ch/","Europe/Berlin","DE","0848 44 66 88"
"827","Städtische Verkehrsbetriebe Bern","http://www.sbb.ch/","Europe/Berlin","DE","0848 44 66 88"
"37","Baselland Transport","http://www.sbb.ch/","Europe/Berlin","DE","0848 44 66 88"
"96","Aargau Verkehr AG","http://www.sbb.ch/","Europe/Berlin","DE","0848 44 66 88"
"88","Regionalverkehr Bern-Solothurn","http://www.sbb.ch/","Europe/Berlin","DE","0848 44 66 88"
"78","Sihltal-Zürich-Uetliberg-Bahn","http://www.sbb.ch/","Europe/Berlin","DE","0848 44 66 88"
"44","Transports Publics Neuchâtelois SA (cmn)","http://www.sbb.ch/","Europe/Berlin","DE","0848 44 66 88"
"53","Transports publics fribourgeois","http://www.sbb.ch/","Europe/Berlin","DE","0848 44 66 88"

Figure 4: agency.txt Example OpenTransportData

11.1.3.2 Stops.txt

A stop is a place where a vehicle stops to allow passengers to get on and off. Stops are defined in the **stops.txt** file. Stops can be grouped together, for example, if there are multiple stops in a station. In this case, a parent stop is defined for all stops in this station.



Specific information to this section can be found here:

<https://developers.google.com/transit/gtfs/reference#stopstxt>

[Best Practices - General Transit Feed Specification \(gtfs.org\)](https://www.gtfs.org/en/best-practices-general)

The stop point must be modelled in GTFS as individual stops. A change of the stop point is then a change of the stop from the point of view of GTFS.

An example "8500010" (Basel SBB)

```
"8500010","Basel SBB","47.5474065481454","7.58955142623287","","8500010P"
"8500010:0:1","Basel SBB","47.5474671825912","7.58955142623287","","8500010P"
"8500010:0:10","Basel SBB","47.5483160574667","7.58955142623287","","8500010P"
"8500010:0:11","Basel SBB","47.548255424003","7.58955142623287","","8500010P"
"8500010:0:11AB","Basel SBB","47.5478916217475","7.58955142623287","","8500010P"
"8500010:0:12","Basel SBB","47.5481947904691","7.58955142623287","","8500010P"
"8500010:0:14","Basel SBB","47.548134156865","7.58955142623287","","8500010P"
"8500010:0:15","Basel SBB","47.5484979574371","7.58955142623287","","8500010P"
"8500010:0:16","Basel SBB","47.5480128894465","7.58955142623287","","8500010P"
"8500010:0:16C","Basel SBB","47.5489830209385","7.58955142623287","","8500010P"
"8500010:0:17","Basel SBB","47.5480735231908","7.58955142623287","","8500010P"
"8500010:0:2","Basel SBB","47.5478309877927","7.58955142623287","","8500010P"
"8500010:0:3","Basel SBB","47.5476490855076","7.58955142623287","","8500010P"
"8500010:0:30","Basel SBB","47.5489223882463","7.58955142623287","","8500010P"
"8500010:0:31","Basel SBB","47.5486192237333","7.58955142623287","","8500010P"
"8500010:0:33","Basel SBB","47.5486798567762","7.58955142623287","","8500010P"
"8500010:0:35","Basel SBB","47.548740489749","7.58955142623287","","8500010P"
"8500010:0:4","Basel SBB","47.5475884512723","7.58955142623287","","8500010P"
"8500010:0:5","Basel SBB","47.5477097196728","7.58955142623287","","8500010P"
"8500010:0:5AB","Basel SBB","47.5484373241838","7.58955142623287","","8500010P"
"8500010:0:6","Basel SBB","47.5477703537678","7.58955142623287","","8500010P"
"8500010:0:7","Basel SBB","47.5479522556321","7.58955142623287","","8500010P"
"8500010:0:7AB","Basel SBB","47.5488011226516","7.58955142623287","","8500010P"
"8500010:0:7CD","Basel SBB","47.5485585906203","7.58955142623287","","8500010P"
"8500010:0:8","Basel SBB","47.5483766908603","7.58955142623287","","8500010P"
"8500010:0:9","Basel SBB","47.5475278169668","7.58955142623287","","8500010P"
"8500010:0:9AB","Basel SBB","47.548861755484","7.58955142623287","","8500010P"
"8500010P","Basel SBB","47.5474065481454","7.58955142623287","1",""
```

Figure 5: Stop Point Example with Parent

All stops of Basel have a common "Parent" "8500010P". The individual tracks are separated with ":". A "track" can also be a track+sectors (ex: "8500010:0:7CD"). There are also very extraordinary additions, like "Bern Bfpl" so "8507000:0:Bfpl", when RBS uses it for rail replacement buses.

11.1.3.2.1 File definition: Required

Field Name	Required	Description ✓ Used in Swiss implementation ✗ Not used in Swiss implementation
stop_id	Required	✓ Identifies a stop, station, or station entrance. Currently we use BP-UIC (DIDOK-Code). In future SLOID is planned. Stop_Id is directly taken from the HRDF-Import.
stop_code	Optional	✗ Short text or a number that identifies the location for riders.
stop_name	Conditionally Required	✓ Name of the location. Use a name that people will understand in the local and tourist vernacular.
stop_desc	Optional	✗

		Description of the location that provides useful, quality information. Do not simply duplicate the name of the location.
stop_lat	Conditionally Required	✓ Latitude of the location, based on DIDOK
stop_lon	Conditionally Required	✓ Longitude of the location, based on DIDOK
zone_id	Conditionally Required	✗ Identifies the fare zone for a stop. This field is required if providing fare information using fare_rules.txt , otherwise it is optional.
stop_url	Optional	✗ URL of a web page about the location. This should be different from the agency.agency_url and the routes.route_url field values.
location_type	Optional	✓ (not filled) Type of the location:
parent_station	Conditionally Required	✓ Defines hierarchy between the different locations defined in stops.txt. It contains the ID of the parent location, as followed: a station.
stop_timezone	Optional	✗ Time zone of the location. If the location has a parent station, it inherits the parent station's time zone instead of applying its own.
wheelchair_boarding	Optional	✗ Indicates whether wheelchair boardings are possible from the location. Valid options are:
level_id	Optional	✗ Level of the location. The same level can be used by multiple unlinked stations.
platform_code	Optional	✗ Platform identifier for a platform stop (a stop belonging to a station).

Figure 6: Field definitions stops.txt

11.1.3.2.2 Example from OpenTransportData

stop_id,stop_name,stop_lat,stop_lon,location_type,parent_station
"1100008","Zell (Wiesental), Wilder Mann","47.7100842702352","7.85964788274668","",""
"1100009","Zell (Wiesental), Grönland","47.7131911044794","7.86290876722849","",""
"1100010","Atzenbach","47.7146175266411","7.8723500608659","",""
"1100011","Mambach, Brücke","47.7282088873189","7.8774704579861","",""
"1100012","Mambach, Mühlschau","47.7340818684375","7.8813871126254","",""
"1100013","Mambach, Silbersau","47.7395192233867","7.88223152899259","",""
"1100014","Fröhd (Schwarzw), Wühre","47.7543663509316","7.88913059037559","",""
"1100015","Fröhd (Schwarzw), Unterkastel","47.7605926689054","7.88553732923861","",""
"1100016","Wembach (Baden)","47.7728317637339","7.88772023537933","",""
"1100017","Schönau (Schw), Brand","47.7774561479519","7.89219384549486","",""
"1100018","Schönau (Schw), Zentrum","47.786208751796","7.89383776246503","",""
"1100019","Schönau (Schw), Friedhof","47.7896429916744","7.89809577691235","",""
"1100020","Schönau (Schw), Schönenbuchen","47.7936261842555","7.90221904406703","",""

Figure 7: stops.txt Example OpenTransportData

11.1.3.3 Routes.txt



Specific Information to this section could be found here:

<https://developers.google.com/transit/gtfs/reference#routestxt>

[Best Practices - General Transit Feed Specification \(gtfs.org\)](https://www.gtfs.org/docs/Best-Practices-General-Transit-Feed-Specification)

Attention:

Some routes in GTFS have an extension "-Y". The reason is that they have an artificially created route number. Only this way "**route_short_name**" can be valid. These artificial routes should not be displayed to users like this. For trains, "**trip_short_name**" will contain the train number.

This leads to the problem that unfortunately we do not have a match between the FAHRTID, the HRDF data or the Route_id. But the problem should be solved with the future implementation of SID4PT. Further information could be found here:

11. <https://opentransportdata.swiss/de/cookbook/verwendung-von-hrdf-fahrplaenen-zusammen-mit-gtfs-rt/>

The transport texts (Verkehrsmitteltext) that will be applied (the variable route_desc contains information of the column "Abbreviation"):

Please refer to the table on OpenTransportData for the detailed list with all available transport texts (Verkehrsmitteltexte) [GTFS | Open-Data-Plattform Mobilität Schweiz \(opentransportdata.swiss\)](https://opentransportdata.swiss/de/gtfs-open-data-plattform-mobilitaet-schweiz)

11.1.3.3.1 File definition: Required

Field Name	Required	Description ✔ Used in Swiss implementation ✘ Not used in Swiss implementation
route_id	Required	✔ Identifies a route (based on the DIVA Identifier). In Future SLNID is planned. Current Implementation: route_id = <DIVA Betriebszweig>-<DIVA Liniennummer>-<DIVA Projektkurzbezeichnung>-<DIVA Linienversionsnummer>
agency_id	Conditionally required	✔ Agency for the specified route.
route_short_name	Conditionally required	✔ Short name of a route.
route_long_name	Conditionally required	✔ Full name of a route. This name is generally more descriptive
route_desc	Optional	✔ Description of a route that provides useful, quality information. Do not simply duplicate the name of the route.
route_type	Required	✔ Indicates the type of transportation used on a route. Valid options that can be used in route_type for the Swiss

		implementation can be found on GTFS Open-Data-Plattform Mobilität Schweiz (opentransportdata.swiss)
route_url	Optional	 URL of a web page about the particular route.
route_color	Optional	 Route color designation that matches public facing material.
route_text_color	Optional	 Legible colour to use for text drawn against a background of route_color.
route_sort_order	Optional	 Orders the routes in a way which is ideal for presentation to customers. Routes with smaller route_sort_order values should be displayed first.
continuous_pickup	Optional	 Indicates whether a rider can board the transit vehicle anywhere along the vehicle's travel path.
contin- ous_drop_off	Optional	 Indicates whether a rider can alight from the transit vehicle at any point along the vehicle's travel path.

Figure 8: Field definitions routes.txt

11.1.3.3.2 Example from OpenTransportData

route_id	agency_id	route_short_name	route_long_name	route_desc	route_type
"91-10-A-j23-1"	"78"	"S10"	""	"S"	"109"
"91-10-B-j23-1"	"11"	"S10"	""	"S"	"109"
"91-10-C-j23-1"	"65"	"S10"	""	"S"	"109"
"91-10-E-j23-1"	"3849"	"10"	""	"T"	"900"
"91-10-G-j23-1"	"65"	"SN10"	""	"SN"	"109"
"91-10-j23-1"	"37"	"10"	""	"T"	"900"
"91-10-Y-j23-1"	"11"	"RE"	""	"RE"	"106"
"91-11-A-j23-1"	"11"	"S11"	""	"S"	"109"
"91-11-B-j23-1"	"327000"	"S11"	""	"S"	"109"
"91-11-C-j23-1"	"3849"	"11"	""	"T"	"900"
"91-11-j23-1"	"37"	"11"	""	"T"	"900"

Figure 9: routes.txt Example OpenTransportData

11.1.3.4 trips.txt

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs/reference#tripstxt Best Practices - General Transit Feed Specification (gtfs.org)
---	--

Attention:

For trains, "**trip_short_name**" contains the train number.

A "trip" according to GTFS represents in Transmodel the notion of a "DatedVehicleJourney" (in OJP a "DatedJourney"). In Transmodel and OJP a TRIP is a trip of the customer as returned by the Trip Planner. It consists of different partial journeys, transfers, and other paths to be taken using different modes. **In HRDF the GTFS "trip" corresponds to the journey and in NeTEx to the ServiceJourney.**

More information about the Trip and its structure or composition can be obtained here:

12. <https://opentransportdata.swiss/de/cookbook/verwendung-von-hrdf-fahrplaenen-zusammen-mit-gtfs-rt/>
13. <https://opentransportdata.swiss/de/faq/fahrplaene-gtfs/>

11.1.3.4.1 File definition: Required

Field Name	Required	Description
		 Used in Swiss implementation  Not used in Swiss implementation
route_id	Required	 Identifies a route.
service_id	Required	 Identifies a set of dates when service is available for one or more routes.
trip_id	Required	 Identifies a trip. The Trip ID is made up of different DIVA numbers/fields and has nothing to do with the HRDF data. The assignment is not quite 1:1, as the trip ID (= trip number) does not have to be unique within an HRDF data set, but using stops and stopping times, as the data user did, the appropriate trip can be determined as described here: https://opentransportdata.swiss/de/faq/fahrplaene-gtfs/
trip_headsign	Optional	 Text that appears on signage identifying the trip's destination to riders.
trip_short_name	Optional	

		Public facing text used to identify the trip to riders, for instance, to identify train numbers for commuter rail trips.
direction_id	Optional	 Indicates the direction of travel for a trip. This field is not used in routing; it provides a way to separate trips by direction when publishing timetables.
block_id	Optional	 Identifies the block to which the trip belongs.
shape_id	Conditionally required	 Identifies a geospatial shape that describes the vehicle travel path for a trip.
wheelchair_accessible	Optional	 Indicates wheelchair accessibility. Valid options are:
bikes_allowed	Optional	 Indicates whether bikes are allowed. Valid options are:
original_trip_id	Optional	 Contains the SJYID, which has a N:1 relationship with the trip_id. A SJYID is unique on a per-day basis. This is Swiss only (until further notice).

Figure 10: Field definitions trips.txt

11.1.3.4.2 Example from OpenTransportData

route_id	service_id	trip_id	trip_headsign	trip_short_name	direction_id	block_id
"91-10-A-j23-1"	"TA+gz"	"1.TA.91-10-A-j23-1.1.H"	"Zürich HB SZU"	"12908"	"0"	"
"91-10-A-j23-1"	"TA+gz"	"10.TA.91-10-A-j23-1.1.H"	"Zürich HB SZU"	"12872"	"0"	"
"91-10-A-j23-1"	"TA+em"	"100.TA.91-10-A-j23-1.7.H"	"Zürich HB SZU"	"12876"	"0"	"
"91-10-A-j23-1"	"TA"	"101.TA.91-10-A-j23-1.7.H"	"Zürich HB SZU"	"12844"	"0"	"
"91-10-A-j23-1"	"TA+em"	"102.TA.91-10-A-j23-1.7.H"	"Zürich HB SZU"	"12968"	"0"	"
"91-10-A-j23-1"	"TA+gz"	"103.TA.91-10-A-j23-1.7.H"	"Zürich HB SZU"	"12826"	"0"	"
"91-10-A-j23-1"	"TA"	"104.TA.91-10-A-j23-1.7.H"	"Zürich HB SZU"	"12788"	"0"	"
"91-10-A-j23-1"	"TA"	"105.TA.91-10-A-j23-1.7.H"	"Zürich HB SZU"	"12856"	"0"	"
"91-10-A-j23-1"	"TA+em"	"106.TA.91-10-A-j23-1.7.H"	"Zürich HB SZU"	"12924"	"0"	"
"91-10-A-j23-1"	"TA"	"107.TA.91-10-A-j23-1.7.H"	"Zürich HB SZU"	"12832"	"0"	"
"91-10-A-j23-1"	"TA"	"108.TA.91-10-A-j23-1.7.H"	"Zürich HB SZU"	"12952"	"0"	"

Figure 11: trips.txt Example OpenTransportData

11.1.3.5 Stop_times.txt

	Specific Information to this section could be found here:
	https://developers.google.com/transit/gtfs/reference#stop_timestxt Best Practices - General Transit Feed Specification (gtfs.org)

11.1.3.5.1 File definition: Required

Field Name	Required	Description  Used in Swiss implementation  Not used in Swiss implementation
trip_id	Required	 Identifies a trip.
arrival_time	Conditionally required	 Arrival time at a specific stop for a specific trip on a route. If there are not separate times for arrival and departure at a stop, enter the same value for arrival_time and departure_time.
departure_time	Conditionally required	 Departure time from a specific stop for a specific trip on a route.
stop_id	Required	 Identifies the serviced stop. All stops serviced during a trip must have a record in stop_times.txt .
stop_sequence	Required	 Order of stops for a particular trip. The values must increase along the trip but do not need to be consecutive.
stop_headsign	Optional	 Text that appears on signage identifying the trip's destination to riders.
pickup_type	Optional	 Indicates pickup method. Only valid option for Switzerland is: "0". In the specification, all available values are: 0 - or empty - Regularly scheduled pickup. 1 - No pickup available. 2 - Must phone agency to arrange pickup. 3 - Must coordinate with driver to arrange pickup.
drop_off_type	Optional	 Indicates drop off method. Only valid option for Switzerland is: "0". In the specification, all available values are: 0 or empty - Regularly scheduled drop off. 1 - No drop off available. 2 - Must phone agency to arrange drop off. 3 - Must coordinate with driver to arrange drop off.
continuous_pickup	Optional	 Indicates whether a rider can board the transit vehicle at any point along the vehicle's travel path.
continuous_drop_off	Optional	 Indicates whether a rider can alight from the transit vehicle at any point along the vehicle's travel path as described by shapes.txt, from this stop_time to the next stop_time in the trip's stop_sequence.
shape_dist_traveled	Optional	 Actual distance travelled along the associated shape, from the first stop to the stop specified in this record.

timepoint	Optional	 Indicates if arrival and departure times for a stop are strictly adhered to by the vehicle or if they are instead approximate and/or interpolated times.
------------------	----------	---

Figure 12: Field definitions stop_times.txt

11.1.3.5.2 Example from OpenTransportData

trip_id	arrival_time	departure_time	stop_id	stop_sequence	pickup_type	drop_off_type
"1.TA.91-C24-j23-1.1.H"	"07:50:00"	"07:50:00"	"8772202"	"1"	"0"	"0"
"1.TA.91-C24-j23-1.1.H"	"07:52:00"	"07:54:00"	"8728262"	"2"	"0"	"0"
"1.TA.91-C24-j23-1.1.H"	"07:58:00"	"07:59:00"	"8772850"	"3"	"0"	"0"
"1.TA.91-C24-j23-1.1.H"	"08:03:00"	"08:04:00"	"8772253"	"4"	"0"	"0"
"1.TA.91-C24-j23-1.1.H"	"08:08:00"	"08:09:00"	"8772254"	"5"	"0"	"0"
"1.TA.91-C24-j23-1.1.H"	"08:13:00"	"08:14:00"	"8772256"	"6"	"0"	"0"
"1.TA.91-C24-j23-1.1.H"	"08:19:00"	"08:20:00"	"8772257"	"7"	"0"	"0"
"1.TA.91-C24-j23-1.1.H"	"08:24:00"	"08:24:00"	"8772258"	"8"	"0"	"0"
"2.TA.91-C24-j23-1.1.H"	"07:50:00"	"07:50:00"	"8772202"	"1"	"0"	"0"
"2.TA.91-C24-j23-1.1.H"	"07:52:00"	"07:54:00"	"8728262"	"2"	"0"	"0"
"2.TA.91-C24-j23-1.1.H"	"07:58:00"	"07:59:00"	"8772850"	"3"	"0"	"0"
"2.TA.91-C24-j23-1.1.H"	"08:03:00"	"08:04:00"	"8772253"	"4"	"0"	"0"
"2.TA.91-C24-j23-1.1.H"	"08:08:00"	"08:09:00"	"8772254"	"5"	"0"	"0"

Figure 13: stop_times.txt Example OpenTransportData

11.1.3.6 Calendar.txt

Services define the date range and days of the week (e.g. Monday to Friday) when a trip is available. They are defined in the **calendar.txt** file. A single service can be applied to multiple trips.

If multiple schedules apply to a particular vehicle, e.g. one on weekdays and another on weekends, it is recommended to define two trips with the same stops but different services and StopTimes.

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs/reference#calendartxt Best Practices - General Transit Feed Specification (gtfs.org)
---	---

11.1.3.6.1 File definition: Conditionally Required

Field Name	Required	Description ✓ Used in Swiss implementation ✗ Not used in Swiss implementation
service_id	Required	✓ Uniquely identifies a set of dates when service is available for one or more routes. Each service_id value can appear at most once in a calendar.txt file.
monday	Required	✓ Indicates whether the service operates on all Mondays in the date range specified by the start_date and end_date fields. Note that exceptions for particular dates may be listed in calendar_dates.txt . Valid options are: 1 - Service is available for all Mondays in the date range. 0 - Service is not available for Mondays in the date range.
tuesday	Required	✓ Functions in the same way as monday except applies to Tuesdays
wednesday	Required	✓ Functions in the same way as monday except applies to Wednesdays
thursday	Required	✓ Functions in the same way as monday except applies to Thursdays
friday	Required	✓ Functions in the same way as monday except applies to Fridays
saturday	Required	✓ Functions in the same way as monday except applies to Saturdays.
sunday	Required	✓ Functions in the same way as monday except applies to Sundays.
start_date	Required	✓ Start service day for the service interval.
end_date	Required	✓ End service day for the service interval. This service day is included in the interval.

Figure 14: Field definitions caldener.txt

11.1.3.6.2 Example form OpenTransportData

service_id,monday,tuesday,wednesday,thursday,friday,saturday,sunday,start_date,end_date
"TA", "1", "1", "1", "1", "1", "1", "1", "1", "20221211", "20231209"
"TA+00000", "0", "0", "1", "0", "0", "0", "0", "0", "20221211", "20231209"
"TA+00200", "1", "1", "1", "1", "1", "0", "0", "0", "20221211", "20231209"
"TA+00330", "1", "1", "1", "1", "1", "1", "1", "1", "20221211", "20231209"
"TA+00520", "0", "0", "0", "0", "0", "0", "1", "1", "20221211", "20231209"
"TA+00630", "0", "0", "0", "0", "0", "0", "1", "1", "20221211", "20231209"
"TA+00930", "1", "1", "1", "1", "0", "0", "0", "0", "20221211", "20231209"

"TA+00c20", "1", "1", "1", "1", "1", "0", "0", "20221211", "20231209"
"TA+00c30", "0", "0", "0", "0", "0", "1", "1", "20221211", "20231209"
"TA+00d10", "0", "0", "0", "0", "0", "0", "1", "20221211", "20231209"
"TA+00d30", "0", "0", "0", "0", "0", "0", "1", "20221211", "20231209"

Figure 15: calender.txt Example OpenTransportData

11.1.3.7 Calendar_dates.txt

If a trip is not available on certain days, e.g. holidays, these days should be defined in the [calendar_dates.txt](#) file. This file can be used to define days when a trip is performed or not performed against the rule.

Perhaps you offer special services only on a public holiday. These would then need to be set as unavailable for that day in the [calendar.txt](#) file and as available in the [calendar_dates.txt](#) file

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs/reference#calendartxt Best Practices - General Transit Feed Specification (gtfs.org)
---	--

11.1.3.7.1 File definition: Conditionally Required

Field Name	Required	Description ✔ Used in Swiss implementation ✘ Not used in Swiss implementation
service_id	Required	✔ Identifies a set of dates when a service exception occurs for one or more routes.
date	Required	✔ Date when service exception occurs.
exception_type	Required	✔ Indicates whether service is available on the date specified in the date field. Valid options are: 1 - Service has been added for the specified date. 2 - Service has been removed for the specified date.

Figure 16 Field definitions calendar_dates.txt

11.1.3.7.2 Example from OpenTransportData

<pre>service_id,date,exception_type "TA+00000","20221214","2" "TA+00000","20221221","2" "TA+00000","20230111","2" "TA+00000","20230118","2" "TA+00000","20230125","2" "TA+00000","20230201","2" "TA+00000","20230208","2" "TA+00000","20230215","2" "TA+00000","20230301","2" "TA+00000","20230308","2" "TA+00000","20230315","2"</pre>

Figure 17: calendar_dates.txt Example OpenTransportData

11.1.3.8 Fare_attributes.txt

✗ Not used in Swiss Implementation

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs/reference#fare_attributestxt Best Practices - General Transit Feed Specification (gtfs.org)
---	--

11.1.3.9 Fare_rules.txt

✗ Not used in Swiss Implementation

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs/reference#fare_rulestxt Best Practices - General Transit Feed Specification (gtfs.org)
---	--

11.1.3.10 Shapes.txt

✗ Not used in Swiss Implementation

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs/reference#fare_attributestxt Best Practices - General Transit Feed Specification (gtfs.org)
---	--

11.1.3.11 Frequencies.txt

✗ Not used in Swiss Implementation

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs/reference#frequenciestxt Best Practices - General Transit Feed Specification (gtfs.org)
---	--

11.1.3.12 Transfers.txt

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs/reference#transferstxt Best Practices - General Transit Feed Specification (gtfs.org)
---	---

11.1.3.12.1 File definition: Optional

Field Name	Required	Description
		✓ Used in Swiss implementation ✗ Not used in Swiss implementation
from_stop_id	Required	✓ Identifies a stop or station where a connection between routes begins.
to_stop_id	Required	✓ Identifies a stop or station where a connection between routes ends.
transfer_type	Required	✓ Indicates the type of connection for the specified (from_stop_id, to_stop_id) pair. Valid options are:
min_transfer_time	Optional	✓ Amount of time, in seconds, that must be available to permit a transfer between routes at the specified stops. The min_transfer_time should be sufficient to permit a typical rider to move between the two stops, including buffer time to allow for schedule variance on each route.

Figure 18: Field definitions transfers.txt

11.1.3.12.2 Example from OpenTransportData

from_stop_id,to_stop_id,transfer_type,min_transfer_time
"1100079","8014441","2","240"
"1100079","8014441:0:2","2","240"
"1100079","8014441:0:3","2","240"
"1100079","8014441:0:1","2","240"
"1100084","8014440","2","180"
"1100084","8014440:0:1","2","180"
"1100097","8014447","2","240"
"1100097","8014447:0:2","2","240"
"1100097","8014447:0:1","2","240"
"1100097","8014447:0:3","2","240"
"1100102","8014446","2","240"
"1100102","8014446:0:1","2","240"

11.1.3.13 Levels.txt

 **Not used in Swiss implementation**

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs/reference#levelstxt
---	--

11.1.3.14 Feed_info.txt

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs/reference#feed_infotxt Best Practices - General Transit Feed Specification (gtfs.org)
---	--

11.1.3.14.1 File definition: Conditionally required

Field Name	Required	Description  Used in Swiss implementation  Not used in Swiss implementation
feed_publisher_name	Required	 Full name of the organization that publishes the dataset. This might be the same as one of the agency.agency_name values.
feed_publisher_url	Required	 URL of the dataset publishing organization's website. This may be the same as one of the agency.agency_url values.
feed_lang	Required	 Default language for the text in this dataset. This setting helps GTFS consumers choose capitalization rules and other language-specific settings for the dataset. To define another language, use the language field in translations.txt . For example, a dataset in Switzerland might set the original stops.stop_name field populated with stop names in different languages. Each stop name is written in accordance with the dominant language in that stop's geographic location. Stop names include Genève for the French-speaking city of Geneva, Zürich for the German-speaking city of Zurich, and Biel/Bienne for the bilingual city of Biel/Bienne. Set feed_lang=mul and provide the following translations in translations.txt : • German: "Genf," "Zürich," and "Biel" • French: "Genève," "Zurich," and "Bienne" • Italian: "Ginevra," "Zurigo," and "Bienna"

		• English: "Geneva," "Zurich," and "Biel/Bienne"
default_lang	Optional	 Defines the language used when the data consumer doesn't know the language of the rider. It's often defined as en, English.
feed_start_date	Optional	 The dataset provides complete and reliable schedule information for service in the period from the beginning of the feed_start_date day to the end of the feed_end_date day.
feed_end_date	Optional	 Refer to the feed_start_date row in this table.
feed_version	Optional	 String that indicates the current version of their GTFS dataset.
feed_contact_email	Optional	 Email address for communication regarding the GTFS dataset and data publishing practices.
feed_contact_url	Optional	 URL for contact information, a web-form, support desk, or other tools for communication regarding the GTFS dataset and data publishing practices

11.1.3.14.2 Example from OpenTransportData

```
feed_publisher_name,feed_publisher_url,feed_lang,feed_start_date,feed_end_date,feed_version
"SBB","http://www.sbb.ch/","DE","20221211","20231209","20230327"
```

Figure 19: feed_info.txt Example OpenTransportData

11.1.3.15 Translations.txt

 **Not used in Swiss implementation**



Specific information to this section could be found here:

<https://developers.google.com/transit/gtfs/reference#translationstxt>

11.1.3.16 Attibutions.txt

 **Not used in Swiss implementation**



Specific information to this section could be found here:

<https://developers.google.com/transit/gtfs/reference#attributionstxt>

11.2 GTFS-Realtime (Swiss Profile)

Best Practices: [Best Practices - General Transit Feed Specification \(gtfs.org\)](https://gtfs.org/realtime/best-practices/)
(<https://gtfs.org/realtime/best-practices/>)

11.2.1 Introduction

GTFS-Realtime is a standard developed by TriMet&Google in order to allow transit agencies to provide realtime information about their service.

There are three types of data a GTFS-Realtime feed provides:

1. Vehicle positions
2. Trip updates (provided by SBB)
3. Service alerts (provided by SBB)

Vehicle positions contain data about events that have already occurred (e.g. “the vehicle was at this location one minute ago”), whereas **trip updates** contain data about events that are yet to occur (e.g. “the bus will arrive in three minutes”). **Service alerts** contains information about delays, changed routes, replacement vehicles or cancellations for individual lines to enable passengers to plan as accurately as possible.

Typically, a single GTFS-Realtime feed contains only one of these three types of data. Many agencies therefore have multiple GTFS-Realtime feeds (that is, one for vehicle positions, one for trip updates and one for service alerts).

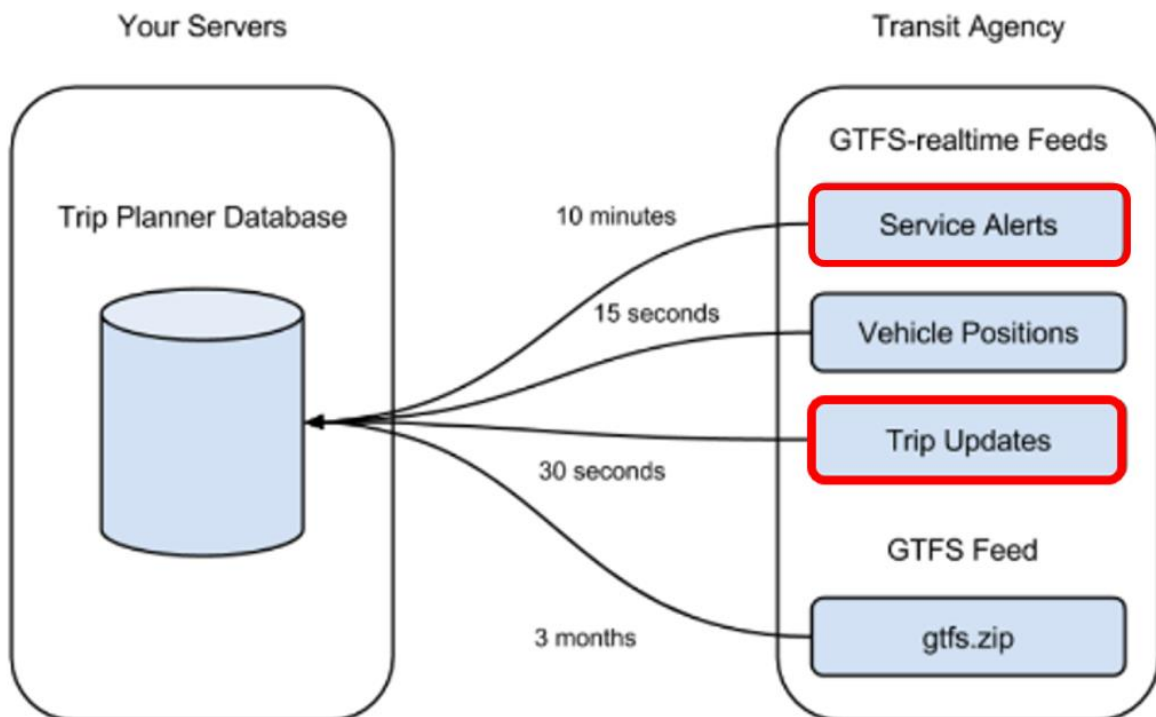


Figure 20:GTFS-Realtime feeds

In Switzerland, two of the three services are currently implemented, namely "Trip Update" and "Service Alerts".

11.2.2 Swiss Specific Information

1. The OpenDataPlatform is currently only supporting GTFS-RT "Trip Update" – Version 1.0. Please be aware that 2.0 elements are currently ignored.
2. The OpenDataPlatform is currently supporting GTFS-RT "Service Alerts"– Version 2.0. Please note these differences in the Service Versions
3. You can make a maximum of two queries per minute on the interface with your key. This is a sliding window.
4. The realtime feed includes all known changes in public transport Switzerland in the entire preview window (three hours) for all transport companies that provide realtime data.
5. For GTFS-RT there is an update rule for delays. If, for example, a whole trip is delayed by 5 minutes, this is only shown on the first stop. For all further stops the delay must be updated during import. The update must be done for arrival and departure times.
6. GTFS-RT only provides new data if something has changed. Only the departure forecast is considered by our system. If the departure forecast remains and only the arrival forecast changes, no GTFS-RT message is generated for this trip.
7. The trip update of GTFS-RT now includes the “feed_version” element which references the GTFS-Static file.

11.2.3 Full Element Index

[GTFS Realtime Reference](#) | [Realtime Transit](#) | [Google Developers](#)

Element index with link to Google documentation:

- [FeedMessage](#)
 - [FeedHeader](#)
 - [Incrementality](#)
 - [FeedEntity](#)
 - [TripUpdate](#)
 - [TripDescriptor](#)
 - [ScheduleRelationship](#)
 - [VehicleDescriptor](#)
 - [StopTimeUpdate](#)
 - [StopTimeEvent](#)
 - [ScheduleRelationship](#)
 - [VehiclePosition](#)
 - [TripDescriptor](#)
 - [ScheduleRelationship](#)
 - [VehicleDescriptor](#)
 - [Position](#)
 - [VehicleStopStatus](#)
 - [CongestionLevel](#)
 - [OccupancyStatus](#)
 - [Alert](#)
 - [TimeRange](#)
 - [EntitySelector](#)
 - [TripDescriptor](#)
 - [ScheduleRelationship](#)
 - [Cause](#)
 - [Effect](#)
 - [TranslatedString](#)
 - [Translation](#)

Figure 21: Element Index GTFS-RT

11.2.4 General Elements

- [FeedMessage](#)
 - [FeedHeader](#)
 - [Incrementality](#)
 - [FeedEntity](#)

11.2.4.1 FeedMessage

	Specific information to section can be found here: https://developers.google.com/transit/gtfs-realtime/reference#message-feedmessage Best Practices - General Transit Feed Specification (gtfs.org)
---	---

Field Name	Required	Cardinality	Description ✓ Used in Swiss implementation ✗ Not used in Swiss implementation
header	Required	One	✓ Metadata about this feed and feed message.
entity	Conditionally required	Many	✓ Contents of the feed. If there is realtime information available for the transit system, this field must be provided. If this field is empty, consumers should assume there is no realtime information available for the system.

Figure 22: Field definitions FeedMessage

```
{
  "Header": {
    "GtfsRealtimeVersion": "1.0",
    "Incrementality": "FullDataset",
    "Timestamp": 1680178351
  },
  "Entity": [
    {
      "Id": "42.TA.91-71A-j23-1.16.H",
      "IsDeleted": false,
      "TripUpdate": {
        "Trip": {
          "TripId": "42.TA.91-71A-j23-1.16.H",
          "RouteId": "91-71A-j23-1",
          "StartTime": "07:00:00",
          "StartDate": "20230330",
          "ScheduleRelationship": "Scheduled"
        }
      }
    }
  ]
}
```

Figure 23: FeedMessage - Example OpenTransportData

11.2.4.2 Message FeedHeader:

	Specific information to section can be found here: https://developers.google.com/transit/gtfs-realtime/reference#message-feedheader Best Practices - General Transit Feed Specification (gtfs.org)
---	--

Field Name	Required	Cardinality	Description  Used in Swiss implementation  Not used in Swiss implementation
gtfs_realtime_version	Required	One	 See Google description. Only version 1.0 is implemented.
incrementality	Optional	One	 In the Swiss implementation only "Full_Dataset" is used
timestamp	Optional	One	 This timestamp identifies the moment when the content of this feed has been created (in server time). In POSIX time (i.e., number of seconds since January 1st 1970 00:00:00 UTC).
feed_version	Optional	One	 The feed_version matches the feed_version in the feed_info.txt file of the GTFS Static / Schedule. This indicates which version of the GTFS file is currently being used by the realtime data, so the consumer knows exactly when to switch to a new file. It can also be used to identify when to check for a new GTFS file.

Figure 24: Field definitions - FeedHeader

<pre>{ "Header": { "GtfsRealtimeVersion": "1.0", "Incrementality": "FullDataset", "Timestamp": 1680178351, "FeedVersion": "20230327", }, "Entity": [{ "Id": "42.TA.91-71A-j23-1.16.H", "IsDeleted": false, "TripUpdate": { "Trip": { "TripId": "42.TA.91-71A-j23-1.16.H", "RouteId": "91-71A-j23-1", "StartTime": "07:00:00", "StartDate": "20230330", "ScheduleRelationship": "Scheduled" } } }] }</pre>

Figure 25: FeedHeader - Example OpenTransportData

Value	Description  Used in Swiss implementation  Not used in Swiss implementation
FULL_DATASET	

	This feed update will overwrite all preceding real-time information for the feed. Thus, this update is expected to provide a full snapshot of all known real-time information.
DIFFERENTIAL	 Currently, this mode is unsupported and behaviour is unspecified for feeds that use this mode.

Figure 26: Value definition: Enum Incrementality

11.2.4.3 Message FeedEntity

	Specific information to section can be found here: https://developers.google.com/transit/gtfs-real-time/reference#message-feedentity Best Practices - General Transit Feed Specification (gtfs.org)
--	--

Field Name	Required	Cardinality	Description ✓ Used in Swiss implementation ✗ Not used in Swiss implementation
id	Required	One	✓ Feed-unique identifier for this entity. The IDs are used only to provide incrementality support. DIVA Internal Number
is_deleted	Optional	One	✓ Whether this entity is to be deleted. Should be provided only for feeds with Incrementality of DIFFERENTIAL - this field should NOT be provided for feeds with Incrementality of FULL_DATASET. The Swiss Implementation always provides "IsDeleted": false
trip_update	Conditionally required	One	✓ Data about the real-time departure delays of a trip. At least one of the fields trip_update, vehicle, or alert must be provided - all these fields cannot be empty.
vehicle	Conditionally required	One	✗ Data about the real-time position of a vehicle. At least one of the fields trip_update, vehicle, or alert must be provided - all these fields cannot be empty.
alert	Conditionally required	One	✓ Data about the real-time alert. At least one of the fields trip_update, vehicle, or alert must be provided - all these fields cannot be empty.

Figure 27: Field definition - FeedEntity

<pre> { "Header": { "GtfsRealtimeVersion": "1.0", "Incrementality": "FullDataset", "Timestamp": 1680178351 }, "Entity": [{ "Id": "42.TA.91-71A-j23-1.16.H", "IsDeleted": false, "TripUpdate": { "Trip": { "TripId": "42.TA.91-71A-j23-1.16.H", "RouteId": "91-71A-j23-1", "StartTime": "07:00:00", "StartDate": "20230330", "ScheduleRelationship": "Scheduled" } } }] } </pre>

Figure 28: FeedEntity - Example OpenTransportData

11.2.5 Service Alerts (Provided by SBB)

✓ Used in Swiss implementation

11.2.5.1 Introduction

More information about the Service Alert implementation could as well be found here:

8. [GTFS-RT: Service-Alerts – \(Ereignisinformationen Schweiz\) | Open-Data-Plattform Mobilität Schweiz \(opentransportdata.swiss\)](#)

11.2.5.2 Elements

• Alert • TimeRange • EntitySelector • TripDescriptor • ScheduleRelationship • Cause • Effect • TranslatedString • Translation
--

11.2.5.3 message Alert

	General information in terms of Service Alerts can be found here: Service Alerts Realtime Transit Google for Developers Specific information to section can be found here: https://developers.google.com/transit/gtfs-real-time/reference#message-alert
---	---

Field Name	Required	Cardinality	Description ✓ Used in Swiss implementation ✗ Not used in Swiss implementation
active_period	Optional	Many	✓ Time when the alert should be shown to the user. If missing, the alert will be shown as long as it appears in the feed. If multiple ranges are given, the alert will be shown during all of them.
informed_entity	Required	Many	✓ Entities whose users we should notify of this alert. At least one informed_entity must be provided.
cause	Optional	One	✓
effect	Optional	One	✓
url	Optional	One	✓ The URL which provides additional information about the alert. Using Message Translation
header_text	Required	One	✓ Header for the alert. This plain-text string will be highlighted, for example in boldface. Using Message Translation
description_text	Required	One	✓

			Description for the alert. This plain-text string will be formatted as the body of the alert (or shown on an explicit "expand" request by the user). The information in the description should add to the information of the header. Using Message Translation
--	--	--	---

Table 1: Field definition - Alert

<pre> "id": "f8890c30-clbe-5b3c-a9f1-5fb4b3e22851", >alert": { "activePeriod": [{ "start": "1690538400", "end": "1690545600" }], "informedEntity": [{ "agencyId": "801", "routeId": "96-214-j23-1", "directionId": 0 }, { "agencyId": "801", "routeId": "96-214-j23-1", "directionId": 1 }], "cause": "UNKNOWN_CAUSE", "effect": "UNKNOWN_EFFECT", "url": { "translation": [{ "text": "https://www.PostAuto.ch", "language": "de" }, { "text": "https://www.PostAuto.ch", "language": "fr" }, { "text": "https://www.PostAuto.ch", "language": "it" }, { "text": "https://www.PostAuto.ch", "language": "en" }] }, "headerText": { "translation": [{ "text": "Unregelmässiger Busverkehr der Linie 475.", "language": "de" }, { "text": "Bus services of line 475 are running at irregular inter- vals.", "language": "en" }, { "text": "Circolazione degli autobus irregolare sulla linea 475.", "language": "it" }, { "text": "Circulation irrégulière des bus sur la ligne 475.", "language": "fr" }] }, "descriptionText": { "translation": [{ </pre>	
---	--

```

    "text": "Der Grund dafür ist ein Unfall. Die Einschränkung dauert bis
ca. 13:30. Es sind Verspätungen zu erwarten. Wir empfehlen, kurz vor jeder Fahrt den Online-Fahr-
plan zu konsultieren.",
    "language": "de"
  },
  {
    "text": "This is due to an accident. The restriction lasts until ap-
prox. 13:30. Expect delays. Please consult online timetable before each trip.",
    "language": "en"
  },
  {
    "text": "Il motivo è un incidente. La restrizione durerà fino alle
13:30 circa. Sono da prevedere ritardi. Raccomandiamo di consultare l'orario online poco prima di
ogni viaggio.",
    "language": "it"
  },
  {
    "text": "Un accident en est la cause. La perturbation dure jusqu'à
env. 13:30. Des retards sont à prévoir. Nous vous recommandons de consulter l'horaire en ligne peu
avant le départ.",
    "language": "fr"
  }
]
}
},

```

Table 2: Alert: Example OpenTransportData

11.2.5.4 Message TimeRange (Swiss implementation: «activePeriod»)

A time interval. The interval is considered active at time *t* if *t* is greater than or equal to the start time and less than the end time.

	General information in terms of Service Alerts can be found here: Service Alerts Realtime Transit Google for Developers Specific information to section can be found here: https://developers.google.com/transit/gtfs-real-time/reference#message-timerange
---	--

Field Name	Required	Cardinality	Description ✔ Used in Swiss implementation ✘ Not used in Swiss implementation
start	Conditionally required	One	✔ Start time, in POSIX time (i.e., number of seconds since January 1st 1970 00:00:00 UTC). If missing, the interval starts at minus infinity. If a TimeRange is provided, either start or end must be provided - both fields cannot be empty.
end	Conditionally required	One	✔ End time, in POSIX time (i.e., number of seconds since January 1st 1970 00:00:00 UTC). If missing, the interval ends at plus infinity. If a TimeRange is provided, either start or end must be provided - both fields cannot be empty.

Table 3: Field definition - TimeRange

<pre>"id": "0b781290-4776-5628-8aa4-fccc8dec62bd", "alert": { "activePeriod": [{ "start": "1690541760", "end": "1690545600" }], }</pre>

Table 4: TimeRange: Example OpenTransportData

11.2.5.5 Message EntitySelector (Swiss Implementation: «informedEntity»)

A selector for an entity in a GTFS feed. The values of the fields should correspond to the appropriate fields in the GTFS feed. At least one specifier must be given. If several are given, they should be interpreted as being joined by the logical AND operator. Additionally, the combination of specifiers must match the corresponding information in the GTFS feed.

In other words, for an alert to apply to an entity in GTFS it must match all provided EntitySelector fields.

For example, an EntitySelector that includes the fields route_id: "5" and route_type: "3" applies only to the route_id: "5" bus - it does not apply to any other routes of route_type: "3".

If a producer wants an alert to apply to route_id: "5" as well as route_type: "3", it should provide two separate EntitySelector fields, one referencing route_id: "5" and another referencing route_type: "3".

At least one specifier must be given - all fields in an EntitySelector cannot be empty.



General information in terms of Service Alerts can be found here: [Service Alerts | Realtime Transit | Google for Developers](#)

Specific Information to Section could be found here:

<https://developers.google.com/transit/gtfs-real-time/reference#message-entityselector>

Field Name	Required	Cardinality	Description ✓ Used in Swiss implementation ✗ Not used in Swiss implementation
agency_id	Conditionally required	One	✓ The agency_id from the GTFS feed that this selector refers to. Go-Nummer is used in Swiss implementation. See GTFS-Schedule implementation
route_id	Conditionally required	One	✓ The route_id from the GTFS feed that this selector refers to. If direction_id is provided, route_id must also be provided. Still DIVA Number is used for the identification of the route. See GTFS-Schedule implementation
route_type	Conditionally required	One	✗ The route_type from the GTFS feed that this selector refers to.
direction_id	Conditionally required	One	✓ The direction_id from the GTFS feed trips.txt file, used to select all trips in one direction for a route, specified by route_id . If direction_id is provided, route_id must also be provided.
trip	Conditionally required	One	✗ Currently not implemented in EMS system The trip instance from the GTFS feed that this selector refers to. This TripDescriptor must resolve to a single trip instance in the GTFS data (e.g., a producer cannot provide only a trip_id for exact_times=0 trips). If the ScheduleRelationship field is populated within this TripDescriptor it will be ignored by consumers when attempting to identify the GTFS trip.
stop_id	Conditionally required	One	✓ The stop_id from the GTFS feed that this selector refers to. SLOID is provide for StopID

Table 5: Field definition – EntitySelector

```

    "informedEntity": [
      {
        "agencyId": "801",
        "routeId": "96-214-j23-1",
        "directionId": 0
      },
      {
        "agencyId": "801",
        "routeId": "96-214-j23-1",
        "directionId": 1
      }
    ],

```

Second Example:

```

    "informedEntity": [
      {
        "agencyId": "65",
        "stopId": "ch:1:sloid:6201"
      },
      {
        "agencyId": "82",
        "stopId": "ch:1:sloid:6201"
      }
    ],

```

Table 6: EntitySelector: Example OpenTransportData

11.2.5.6 Message TripDescription



Not used in Swiss implementation



General information in terms of Service Alerts can be found here: [Service Alerts | Realtime Transit | Google for Developers](#)

Specific information to section could be found here:

<https://developers.google.com/transit/gtfs-real-time/reference#message-tripdescriptor>

11.2.5.7 Enum Cause



General information in terms of Service Alerts can be found here: [Service Alerts | Realtime Transit | Google for Developers](#)

Specific information to section can be found here:

<https://developers.google.com/transit/gtfs-real-time/reference#enum-cause>

<i>Value GTFS</i>	<i>Value EMS</i>
UNKNOWN_CAUSE ✓ Used in Swiss implementation	AlertCause=unknown
OTHER_CAUSE ✓ Used in Swiss implementation	AlertCause AlertCause = undefinedAlertCause AlertCause = serviceDisruption AlertCause = emergencyServicesCall AlertCause= routeBlockage

	AlertCause= specialEvent AlertCause=congestion
TECHNICAL_PROBLEM ✔ Used in Swiss implementation	AlertCause=vehicleFailure AlertCause=technicalProblem
STRIKE ✘ Not used in Swiss implementation	
DEMONSTRATION ✘ Not used in Swiss implementation	
ACCIDENT ✔ Used in Swiss implementation	AlertCause=accident
HOLIDAY	
WEATHER ✔ Used in Swiss implementation	AlertCause=poorWeather
MAINTENANCE ✔ Used in Swiss implementation	AlertCause=maintenaceWork
CONSTRUCTION ✔ Used in Swiss implementation	AlertCause=constructionWork
POLICE_ACTIVITY ✘ Not used in Swiss implementation	
MEDICAL_EMERGENCY ✘ Not used in Swiss implementation	

Table 7: Field definition cause and comparison to EMS

Ursachen

Ursache System	Ursache Interface
AlertCause	OTHER_CAUSE
AlertCause » undefinedAlertCause	OTHER_CAUSE
AlertCause » constructionWork	CONSTRUCTION
AlertCause » serviceDisruption	OTHER_CAUSE
AlertCause » emergencyServicesCall	OTHER_CAUSE
AlertCause » vehicleFailure	TECHNICAL_PROBLEM
AlertCause » poorWeather	WEATHER
AlertCause » routeBlockage	OTHER_CAUSE
AlertCause » technicalProblem	TECHNICAL_PROBLEM
AlertCause » unknown	UNKNOWN_CAUSE
AlertCause » accident	ACCIDENT
AlertCause » specialEvent	OTHER_CAUSE
AlertCause » congestion	OTHER_CAUSE
AlertCause » maintenanceWork	MAINTENANCE

Table 8: Converter Rules EMS SBB to GTFS

```

    "cause": "UNKNOWN_CAUSE",
    "effect": "UNKNOWN_EFFECT",
    "url": {
      "translation": [
        {
          "text": "https://www.PostAuto.ch",
          "language": "de"
        },
        {
          "text": "https://www.PostAuto.ch",
          "language": "fr"
        },
        {
          "text": "https://www.PostAuto.ch",
          "language": "it"
        },
        {
          "text": "https://www.PostAuto.ch",
          "language": "en"
        }
      ]
    },
  },

```

Table 9: Cause: Example OpenTransportData

11.2.5.8 Enum Effect

	General information in terms of Service Alerts can be found here: Service Alerts Realtime Transit Google for Developers Specific information to section can be found here: https://developers.google.com/transit/gtfs-real-time/reference#enum-effect
---	---

<i>Value GTFS</i>	<i>Value EMS</i>
NO_SERVICE ✗ Not used in Swiss implementation	
REDUCED_SERVICE ✗ Not used in Swiss implementation	
SIGNIFICANT_DELAYS ✗ Not used in Swiss implementation	
DETOUR ✗ Not used in Swiss implementation	
ADDITIONAL_SERVICE ✗ Not used in Swiss implementation	
MODIFIED_SERVICE ✗ Not used in Swiss implementation	
OTHER_EFFECT ✗ Not used in Swiss implementation	
UNKNOWN_EFFECT ✓ Used in Swiss implementation	Condition: Unknown
STOP_MOVED ✗ Not used in Swiss implementation	

Table 10: Field definition "Effect" and comparison to EMS

<pre> "cause": "UNKNOWN_CAUSE", "effect": "UNKNOWN_EFFECT", "url": { "translation": [{ "text": "https://www.PostAuto.ch", "language": "de" }, { "text": "https://www.PostAuto.ch", "language": "fr" }, { "text": "https://www.PostAuto.ch", "language": "it" }, { "text": "https://www.PostAuto.ch", "language": "en" }] }, </pre>
--

Table 11: Effect: Example OpenTransportData

11.2.5.9 Message TranslatedString

	General information in terms of Service Alerts can be found here: Service Alerts Realtime Transit Google for Developers Specific information to section can be found here: https://developers.google.com/transit/gtfs-real-time/reference#message-translatedstring
--	---

Field Name	Required	Cardinality	Description Used in Swiss implementation Not used in Swiss implementation
translation	Required	Many	At least one translation must be provided.

Table 12: Field definition - TranslatedString

Used for the elements "url", "headerText", "descriptionText"

<pre>"headerText": { "translation": [{ "text": "Unregelmässiger Busverkehr der Linie 475.", "language": "de" }, { "text": "Bus services of line 475 are running at irregular intervals.", "language": "en" }, { "text": "Circolazione degli autobus irregolare sulla linea 475.", "language": "it" }, { "text": "Circulation irrégulière des bus sur la ligne 475.", "language": "fr" }] }</pre>
--

Table 13: TranslatedString: Example OpenTransportData

11.2.5.10 Message Translation

	General information in terms of Service Alerts can be found here: Service Alerts Realtime Transit Google for Developers Specific information to section can be found here: https://developers.google.com/transit/gtfs-real-time/reference#fields_15
--	---

Field Name	Required	Cardinality	Description Used in Swiss implementation Not used in Swiss implementation
text	Required	One	 A UTF-8 string containing the message.
language	Conditionally required	One	 BCP-47 language code. Can be omitted if the language is unknown or if no internationalization is done at all for the feed. At most one translation is allowed to have an unspecified language tag - if there is more than one translation, the language must be provided.

Table 14: Field definition – Translation

```

"headerText": {
  "translation": [
    {
      "text": "Unregelmässiger Busverkehr der Linie 475.",
      "language": "de"
    },
    {
      "text": "Bus services of line 475 are running at irregular intervals.",
      "language": "en"
    },
    {
      "text": "Circolazione degli autobus irregolare sulla linea 475.",
      "language": "it"
    },
    {
      "text": "Circulation irrégulière des bus sur la ligne 475.",
      "language": "fr"
    }
  ]
}

```

Table 15: Translation: Example OpenTransportData

11.2.6 Vehicle Position

 Not used in Swiss Implementation

11.2.6.1 Introduction

Not used in Swiss Profile

11.2.6.2 Elements

- [Alert](#)
 - [TimeRange](#)
 - [EntitySelector](#)
 - [TripDescriptor](#)
 - [ScheduleRelationship](#)
 - [Cause](#)
 - [Effect](#)
 - [TranslatedString](#)
 - [Translation](#)

11.2.7 Trip Updates (Provided by SBB)

	Updates zu Fahrten Realtime Transit Google Developers (Deutsch)
	Trip Updates Realtime Transit Google Developers (English)

The only HTTP stream currently available in Switzerland in the GTFS-RT area.

11.2.7.1 Introduction

A trip update message is used to report the progress of a vehicle along its trip. Each trip may only have one trip update message in a GTFS-Realtime feed.

A trip update can report that a trip has been cancelled, or it can update the progress of any number of stops on the trip. For example, a trip update may contain an arrival estimate only for the vehicle's next stop, or it may contain estimates for every remaining stop on the trip.

If a trip does not have a trip update message, this should be interpreted as there being no real-time information available; not that it is necessarily progressing as scheduled.

For GTFS-RT there is an update rule for delays. If, for example, a whole trip is delayed by 5 minutes, this is only shown on the first stop. For all further stops the delay must be updated during import. The update must be done for arrival and departure times.

GTFS-RT only provides new data if something has changed. Only the departure forecast is considered by our system. If the departure forecast remains and only the arrival forecast changes, no GTFS-RT message is generated for this trip.

Useful links:

1. [FAQ to the Swiss Implementation](#)
2. [Using HRDF timetable together with GTFS](#)

11.2.7.2 Elements

Full element index can be found here (official Google documentation):

GTFS Realtime Reference Realtime Transit Google Developers	
	• TripUpdate
	• TripDescriptor
	• ScheduleRelationship
	• VehicleDescriptor
	• StopTimeUpdate
	• StopTimeEvent
	• ScheduleRelationship

11.2.7.3 Message TripUpdate

	General information in terms of TripUpdates can be found here: Trip Updates Realtime Transit Google Developers
	Specific information to section can be found here: https://developers.google.com/transit/gtfs-real-time/reference#message-tripupdate Best Practices - General Transit Feed Specification (gtfs.org)

Field Name	Required	Cardinality	Description
			✓ Used in Swiss implementation ✗ Not used in Swiss implementation
<u>trip</u>	Required	One	✓ This element is used to match the referenced trip to trips.txt file from the corresponding GTFS feed
<u>vehicle</u>	Optional	One	✗ This element provides information that can be used to identify a particular vehicle
<u>stop_time_update</u>	Conditionally required	Many	✓ This element contains one or more instances of StopTimeUpdate . Each occurrence represents a prediction for a single stop. They must be in the order of their stop sequence
timestamp	Optional	One	✗ This value refers to the moment at which the real-time progress was measured, specified in number of seconds since 1-Jan-1970 00:00:00 UTC
delay	Optional	One	✗ This value is only experimental at time of writing. It is used to indicate the number of seconds the vehicle is either early (negative number) or late (positive number). Estimates specified within StopTimeUpdate elements take precedence over this value

Figure 29: Field definition - TripUpdate

```

{
  "Id": "42.TA.91-71A-j23-1.16.H",
  "IsDeleted": false,
  "TripUpdate": {
    "Trip": {
      "TripId": "42.TA.91-71A-j23-1.16.H",
      "RouteId": "91-71A-j23-1",
      "StartTime": "07:00:00",
      "StartDate": "20230403",
      "ScheduleRelationship": "Scheduled"
    },
    "StopTimeUpdate": [
      {
        "StopSequence": 1,
        "StopId": "8775605",
        "Departure": {
          "Delay": 0
        },
        "ScheduleRelationship": "Scheduled"
      }
    ]
  }
}

```

Figure 30: TripUpdate- Example OpenTransportData

11.2.7.4 Message TripDescription

This element is used to match the referenced trip to **trips.txt** file from the corresponding GTFS feed. Since the schedule_relationshipvalue is “SCHEDULED“, this trip corresponds to a trip in the Static File (**trips.txt**)

	Specific Information to this section could be found here: https://developers.google.com/transit/gtfs-real-time/reference#message-tripdescriptor Best Practices - General Transit Feed Specification (gtfs.org)
--	---

<i>Field Name</i>	<i>Required</i>	<i>Cardinality</i>	<i>Description</i> ☒ Used in Swiss Implementation ☐ Not used in Swiss Implementation
trip_id	Conditionally required	One	☒ This is the ID of a trip as it appears in the trips.txt of the corresponding GTFS feed. Alternatively, this value may refer to a trip that has been added via a TripUpdate message and does not exist in the GTFS feed
route_id	Conditionally required	One	☒ If this value is specified, it should match the route ID for the trip specified in trip_id . If the route_id is specified but no trip_id is specified, then this trip descriptor references all trips for the given route.
direction_id	Conditionally required	One	☐ This value corresponds to the direction_id value as specified in the trips.txt file of the corresponding GTFS feed. At time of writing this is an experimental field in the GTFS-Realtime specification GTFS-Flex already provides this element. Will be added in GTFS-RT (next Version)
start_time	Conditionally required	One	☒ If the specified trip in trip_id is a frequency-expanded trip, this value must be specified to determine which instance of a trip this selector refers to. Its value is in the format HH:MM:SS ,as in the stop_times.txt and frequencies.txt files
start_date	Conditionally required	One	☒ It is possible that knowing the trip_id may not be enough to determine a specific trip. For instance, if a train is scheduled to depart at 11:30 PM but is running 40 minutes late, then you would need to know its date to match up with the original trip (40 minutes late), and not the next day's instance of the trip (23 hours 20 minutes early). This field helps to avoid this ambiguity. The date is specified in YYYYMMDD format.
schedule_relationship	Optional	One	☒ This value indicates the relationship between the trip(s) specified in this selector and its regular schedule

modified_trip	Optional	One	 Linkage to any modifications done to this trip (shape changes, removal or addition of stops). If this field is provided, the trip_id, route_id, direction_id, start_time, start_date fields of the TripDescriptor MUST be left empty, to avoid confusion by consumers that aren't looking for the ModifiedTripSelector value.
original_trip_id	Optional	One	 This field contains the SJYID, which has a N:1 relationship with the trip_id. SJYID is unique over a day. Additionally, for those entries where VDV454 does not deliver SJYID, we use the VDV454-FahrtBezeichner as original_trip_id. This is Swiss only (until further notice).

Figure 31: Field definition – TripDescription



Specific information to this section can be found here:

<https://developers.google.com/transit/gtfs-real-time/reference#enum-schedulerelease-2>

<i>Value</i>	<i>Comment</i>
SCHEDULED	✓ Trip that is running in accordance with its GTFS schedule or is close enough to the scheduled trip to be associated with it.
ADDED	✓ An extra trip that was added in addition to a running schedule, for example, to replace a broken vehicle or to respond to sudden passenger load.
UNSCHEDULED	A trip that is running with no schedule associated to it - this value is used to identify trips defined in GTFS frequencies.txt with exact_times = 0. It should not be used to describe trips not defined in GTFS frequencies.txt, or trips in GTFS frequencies.txt with exact_times = 1.
CANCELED	✓ A trip that existed in the schedule but was removed.

Figure 32: Value definition: Enum ScheduleRelationship

```
"Id": "42.TA.91-71A-j23-1.16.H",  
"IsDeleted": false,  
"TripUpdate": {  
  "Trip": {  
    "TripId": "42.TA.91-71A-j23-1.16.H",  
    "RouteId": "91-71A-j23-1",  
    "StartTime": "07:00:00",  
    "StartDate": "20230330",  
    "ScheduleRelationship": "Scheduled"  
  }  
}
```

Figure 33: TripDescription - Example OpenTransportData

11.2.7.5 Message VehicleDescription

 Not used in Swiss implementation

This element provides information that can be used to identify a particular vehicle.



Specific information to this section could be found here:

<https://developers.google.com/transit/gtfs-real-time/reference#message-vehicledescriptor>
[Best Practices - General Transit Feed Specification \(gtfs.org\)](#)

11.2.7.6 Message StopTimeUpdate

This element contains one or more instances of StopTimeUpdate. Each occurrence represents a prediction for a single stop. They must be in order of their stop sequence.

The **stop_time_update** elements contain information specific to a stop on the trip. It is repeated for each stop for which information is available. If the trip has been cancelled (indicated by a schedule_relationship value of CANCELED) then there will no stop_time_update elements.

	Specific information to this section could be found here: https://developers.google.com/transit/gtfs-real-time/reference#message-stoptimeupdate Best Practices - General Transit Feed Specification (gtfs.org)
---	---

Field Name	Required	Cardinality	Description ✔ Used in Swiss implementation ✘ Not used in Swiss implementation
stop_sequence	Conditionally required	One	✔ In GTFS feeds, the order of stops in a trip is indicated by the stop_sequence value in stop_times.txt. If specified, the value specified in the StopTimeUpdate must match the value from the GTFS feed. It is possible for a single trip to make multiple visits to a single stop (for example, if it's a loop service), so this value is important.
stop_id	Conditionally required	One	✔ This value corresponds to a single stop from the associated GTFS feed. Using this value and the stop_sequence value, it is possible to pinpoint a specific record from stop_times.txt that this StopTimeUpdate element alters
arrival	Conditionally required	One	✔ Specifies the updated arrival time. If the schedule_relationship is SCHEDULED, then this field and/or departure must be specified. = StopTimeEvent
departure	Conditionally required	One	✔ Specifies the updated departure time. If the schedule_relationship is SCHEDULED, then this field and/or arrival must be specified. = StopTimeEvent
schedule_relationship	Optional	One	✔ If no value is specified, this defaults to SCHEDULED. Other possible values and their meanings are as described below

Figure 34: Field definition – StopTimeUpdate



Specific information to this section can be found here:

<https://developers.google.com/transit/gtfs-real-time/reference#enum-schedulereationship>

<i>Value</i>	<i>Comment</i>
SCHED- ULED	The vehicle is proceeding in accordance with its static schedule of stops, although not necessarily according to the times of the schedule. This is the default behaviour. At least one of arrival and departure must be provided.
SKIPPED	The stop is skipped, i.e., the vehicle will not stop at this stop. The arrival and departure fields are optional.
NO_DATA	No data is given for this stop. It indicates that there is no real-time information available. When set NO_DATA is propagated through subsequent stops, so this is the recommended way of specifying from which stop you do not have real-time information. When NO_DATA is set neither arrival nor departure should be supplied.

Figure 35: Value definition: Enum ScheduleRelationship

11.2.7.7 Message StopTimeEvent



Specific information to this section could be found here:

<https://developers.google.com/transit/gtfs-real-time/reference#message-stoptimeevent>
[Best Practices - General Transit Feed Specification \(gtfs.org\)](#)

Field Name	Required	Cardinality	Description ✓ Used in Swiss implementation ✗ Not used in Swiss implementation
delay	Conditionally required	One	✓ The number of seconds that a vehicle is early (a negative value) or late (a positive value). A value of 0 indicates the vehicle is exactly on time
time	Conditionally required	One	The time of the arrival or departure, specified in number of seconds since 1-Jan-1970 00:00:00 UTC
uncertainty	Optional	One	Represents the level of uncertainty attached to this prediction in seconds. A value of 0 means is it completely certain, while an omitted value means an unknown level of uncertainty.

```

"StopTimeUpdate": [
  {
    "StopSequence": 1,
    "StopId": "8775605",
    "Departure": {
      "Delay": 0
    },
    "ScheduleRelationship": "Scheduled"
  },
  {
    "StopSequence": 7,
    "StopId": "8775100",
    "Arrival": {
      "Delay": 4200
    },
    "Departure": {
      "Delay": 4200
    },
    "ScheduleRelationship": "Scheduled"
  },
  {
    "StopSequence": 8,
    "StopId": "8731901",
    "Arrival": {
      "Delay": 4800
    },
    "Departure": {
      "Delay": 4800
    },
    "ScheduleRelationship": "Scheduled"
  },
  {
    "StopSequence": 10,
    "StopId": "8772568",
    "Arrival": {
      "Delay": 4200
    },
    "Departure": {
      "Delay": 4200
    },
    "ScheduleRelationship": "Scheduled"
  },
  {
    "StopSequence": 11,
    "StopId": "8772500",
    "Arrival": {
      "Delay": 4800
    }
  }
]

```

```
        "Departure": {  
            "Delay": 4800  
        },  
        "ScheduleRelationship": "Scheduled"  
    }  
]
```

Figure 36: StopTimeUpdate - Example OpenTransportData

11.2.7.8 Samples from OpenTransportData

Complete sample form OpenTransportData platform:

```
"Entité": [
  {
    "Id": "350.TA.91-K4-j23-1.25.R",
    "Is Deleted": false,
    "Trip Update": {
      "Trip": {
        "Thripid": "350.TA.91-K4-j23-1.25.R",
        "RouteId": "91-K4-j23-1",
        "StartTime": "06:42:00",
        "StartDate": "20230329",
        "ScheduleRelationship": "Scheduled"
      },
      "StopTimeUpdate": [
        {
          "StopSequence": 1,
          "StopId": "8711300",
          "Departure": {
            "Delay": 0
          },
          "ScheduleRelationship": "Scheduled"
        },
        {
          "StopSequence": 2,
          "StopId": "8711819",
          "Arrival": {
            "Delay": 600
          },
          "Departure": {
            "Delay": 600
          },
          "ScheduleRelationship": "Scheduled"
        },
        {
          "StopSequence": 5,
          "StopId": "8711825",
          "Arrival": {
            "Delay": 300
          },
          "Departure": {
            "Delay": 300
          },
          "ScheduleRelationship": "Scheduled"
        },
        {
          "StopSequence": 7,
          "StopId": "8714200",
          "Arrival": {
            "Delay": 0
          },
          "Departure": {
            "Delay": 0
          },
          "ScheduleRelationship": "Scheduled"
        },
        {
          "StopSequence": 9,
          "StopId": "8714212",
          "Arrival": {
            "Delay": 0
          },
          "Departure": {
            "Delay": -120
          },
          "ScheduleRelationship": "Scheduled"
        },
        {
          "StopSequence": 10,
          "StopId": "8718500",
          "Arrival": {
            "Delay": 0
          },
          "Departure": {
```

```
        "Delay": 0
      },
      "ScheduleRelationship": "Scheduled"
    }
  ]
},
```

Full Trip Update example from Google: [Full Trip Update example | Realtime Transit | Google Developers](#)

12 Extension under Development: GTFS-Flex



The real content may change as development progresses. We will update this document then.

Section 12.1 describes the GTFS-Flex extension as it is defined in its current version 2, re-citing some of the existing documentation. Section 12.2 describes the adaptations and considerations relevant for Switzerland, which we call the Swiss Profile.

12.1 GTFS-Flex

12.1.1 Links

This and the following sections quote and use figures that can also be found in one or more of the following sources. For simplicity we do not cite them per sentence/figure. Additional sources will be given where appropriate.

No	Description	Link
1	General Description	https://gtfs.org/extensions/flex/
2	GTFS Flex examples	https://github.com/MobilityData/gtfs.org/pull/301

12.1.2 Description

GTFS-Flex stands for "General Transit Feed Specification - Flexible". It is an extension of the GTFS format developed by TriMet & Google to standardize and simplify information about public transportation.

In contrast to the original GTFS format, which contains static information about public transportation such as schedules and routes, GTFS-Flex provides additional flexibility by adding dynamic or flexible route information.

With GTFS-Flex, transit operators can respond to ride requests from passengers in real-time by providing information about alternative routes, stops, and departure times.

An example of this is a bus that normally has a fixed schedule but can also deviate from certain stops at certain times upon request to pick up or drop off passengers. With GTFS-Flex, this flexible route information can be integrated into the GTFS format to make public transportation even more user-friendly and efficient.

12.1.2.1 Schematic Extensions

An initial schematic representation of GTFS-Flex as provided directly by MobilityData on their GitHub page became later deprecated (which is why we do not include a link). Nonetheless, the following representation orients itself by that representation:

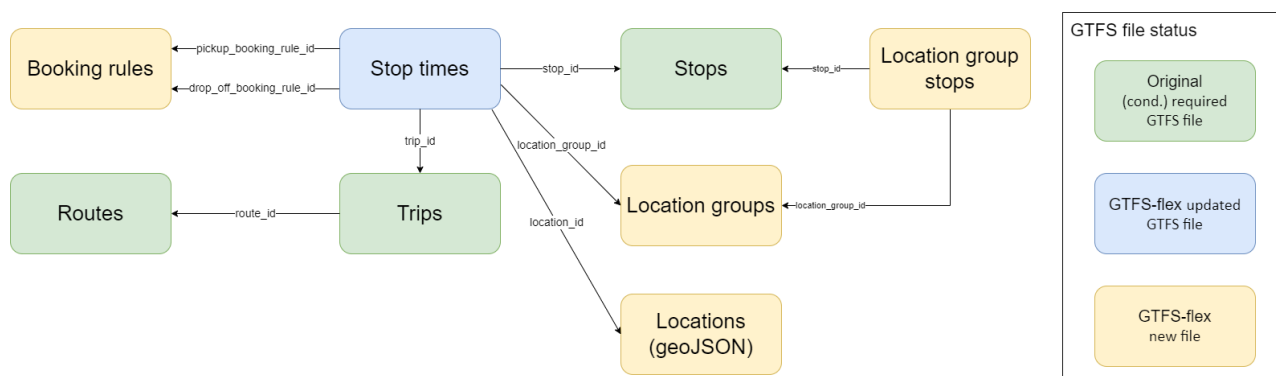


Figure 37: Updated schematic representation of the GTFS-Flex extension based on a deprecated official documentation. Not shown are required GTFS-Schedule files.

12.1.2.2 General Extensions

GTFS-Flex (in v2) comprises two extensions:

Extension Name	Short Description
GTFS- FlexibleTrips	Flexible services that operate according to some schedule but are responsive to on-demand requests of individual riders.
GTFS BookingRules	Booking information for rider-requested services using GTFS-FlexibleTrips , such as how far in advance booking should occur or a phone number that should be called.

GTFS-FlexibleTrips supports the following 5 use cases:

1. Dial-a-ride service: the vehicle serves a zone where pickups and drop offs are allowed during certain service hours (see section 12.2).
2. Route deviation services: the vehicle serves a fixed route and ordered set of stops and may detour to pick up or drop off a passenger between stops.
3. Point-to-zone service: the rider can board at a fixed stop such as a train station, and then alight anywhere within an area, or vice versa. Departures from some locations are scheduled or timed with other services.
4. Point deviation or checkpoint service: the rider can board at a fixed stop, and then alight anywhere among an unordered list of stops, or the opposite. The driver only serves stops at which a request is made.
5. Hail-and-ride services: the vehicle runs along a fixed path, but the rider can request a stop anywhere along the path to board or alight.

To integrate GTFS-Flex into a GTFS implementation, several files that in the GTFS feed are optional files become required, are modified, or new files are added as described in the following section.

12.1.3 Data-Set Files

The following table provides an overview of all GTFS files with a color indication for:

1. Blue = Files (conditionally) required from existing GTFS-Schedule

2. Green = Originally optional files from GTFS-Schedule, now (conditionally) required
3. Orange = Changed files from GTFS-Schedule
4. Red = New file for GTFS-Flex
5. White = Not used for GTFS-Flex

No	Filename	Required	Definition
			 Used in Swiss implementation  Not used in Swiss implementation
1	agency.txt	Required	 Transit agencies with service represented in this dataset
2	stops.txt	Required	 Stops where vehicles pick up or drop off riders. Also defines stations and station entrances.
3	routes.txt	Required	 Transit routes. A route is a group of trips that are displayed to riders as a single service
4	trips.txt	Required	 Trips for each route. A trip is a sequence of two or more stops that occur during a specific time period.
5	stop_times.txt	Required	 Extended and modified file to include time-windows that a vehicle picks up and drops off at stops or location groups/areas.
6	calendar.txt	Conditionally Required	 Service dates specified using a weekly schedule with start and end dates. This file is required unless all dates of service are defined in calendar.txt
7	calendar_dates.txt	Conditionally Required	 Exceptions for the services defined in the calendar.txt . If calendar.txt is omitted, then calendar_dates.txt is required and must contain all dates of service.
8	fare_attributes.txt	Optional	 Not used: Current prices and fares are not yet provided

9	fare_rules.txt	Optional	✗ Not used: Current prices and fares are not yet provided
10	shapes.txt	Optional	✗ Not used because the locations.geojson is used.
11	frequencies.txt	Optional	✗ Not used
12	transfers.txt	Optional	✓ When calculating an itinerary, GTFS-consuming applications interpolate transfers based on allowable time and stop proximity
13	pathways.txt	Optional	✗ We do not yet map multi-level stops
14	levels.txt	Optional	✗ We do not yet map multi-level stops
15	feed_info.txt	Optional	✓ This file contains information about the dataset itself, rather than the service the dataset describes
16	translations.txt	Optional	✗ Not used
17	attributions.txt	Optional	✗ Not used
18	areas.txt	Conditionally required	✗ Not used
19	stop_areas.txt	Conditionally required	✗ Not used
20	booking_rules.txt	Required	✓ New file that defines the booking rules / requirements that must be met for a successful booking

21	locations.geojson	Required	 New GeoJSON (and non-CSV) formatted file that allows defining Zones using Polygons and MultiPolygons (for Zones with “holes”).
22	location_groups	Required	 New files to allow grouping of stops which allow predetermined groups of these features to be specified on individual rows of stop_times.txt. This file specifically defines the group (similar to areas.txt).
23	location_group_stops	Required	 New files to allow grouping of stops which allow predetermined groups of these features to be specified on individual rows of stop_times.txt. This file specifically defines the mapping of the groups from location_groups to stops (similar to stop_areas.txt).

The following sections describe the changed, new and recently required files.

12.1.3.1 location_groups.txt

Field Name	Presence	Description
		 Used in Swiss implementation  Not used in Swiss implementation
location_group_id	Required	 Identifies a location group. ID must be unique across all stops.stop_id, locations.geojson id, and location_groups.location_group_id values. A location group is a group of stops that together indicate locations where a rider may request pickup or drop off.
location_group_name	Optional	 The name of the location group as displayed to the rider.

12.1.3.2 location_group_stops.txt

Field Name	Required	Description
		 Used in Swiss implementation  Not used in Swiss implementation
location_group_id	Required	

		Identifies a location group to which one or multiple stop_ids belong. The same stop_id may be defined in many location_group_ids.
stop_id	Required	 Identifies a stop belonging to the location group.

12.1.3.3 Booking_rules.txt

Field Name	Required	Description  Used in Swiss implementation  Not used in Swiss implementation
booking_rule_id	Required	 Identifies the rule.
booking_type	Required	 Indicates how far in advance booking can be made. Valid options are: 0 - Real time booking. 1 - Up to same-day booking with advance notice. 2 - Up to prior day(s) booking.
prior_notice_duration_min	Conditionally Required	 Minimum number of minutes before travel to make the request. Conditionally Required: - Required for booking_type=1. - Forbidden otherwise.
prior_notice_duration_max	Conditionally Forbidden	 Maximum number of minutes before travel to make the booking request. Conditionally Forbidden: - Forbidden for booking_type=0 and booking_type=2. - Optional for booking_type=1.
prior_notice_last_day	Conditionally Required	 Last day before travel to make the booking request. Example: "Ride must be booked 1 day in advance before 5PM" will be encoded as prior_notice_last_day=1. Conditionally Required: - Required for booking_type=2. - Forbidden otherwise.
prior_notice_last_time	Conditionally Required	 Last time on the last day before travel to make the booking request. Example: "Ride must be booked 1 day in advance before 5PM" will be encoded as prior_notice_last_time=17:00:00.

		Conditionally Required: - Required if prior_notice_last_day is defined. - Forbidden otherwise.
prior_no- tice_start_day	Conditionally Forbidden	 Earliest day before travel to make the booking request. Example: "Ride can be booked at the earliest one week in advance at midnight" will be encoded as prior_no- tice_start_day=7. Conditionally Forbidden: - Forbidden for booking_type=0. - Forbidden for booking_type=1 if prior_notice_dura- tion_max is defined. - Optional otherwise.
prior_no- tice_start_time	Conditionally Required	 Earliest time on the earliest day before travel to make the booking request. Example: "Ride can be booked at the earliest one week in advance at midnight" will be encoded as prior_no- tice_start_time=00:00:00. Conditionally Required: - Required if prior_notice_start_day is defined. - Forbidden otherwise.
prior_no- tice_service_id	Conditionally Forbidden	 Indicates the service days on which prior_no- tice_last_day or prior_notice_start_day are counted. Example: If empty, prior_notice_start_day=2 will be two calendar days in advance. If defined as a service_id con- taining only business days (weekdays without holi- days), prior_notice_start_day=2 will be two business days in advance. Conditionally Forbidden: - Optional if booking_type=2. - Forbidden otherwise.
message	Optional	 Message to riders utilizing service at a stop_time when booking on-demand pickup and drop off. Meant to pro- vide minimal information to be transmitted within a user interface about the action a rider must take to utilize the service.
pickup_mes- sage	Optional	 Functions in the same way as message but used when riders have on-demand pickup only.
drop_off_mes- sage	Optional	

		Functions in the same way as message but used when riders have on-demand drop off only.
phone_number	Optional	 Phone number to call to make the booking request.
info_url	Optional	 URL providing information about the booking rule.
booking_url	Optional	 URL to an online interface or app where the booking request can be made.

Table 1: Field definitions booking_rules.txt

12.1.3.4 Stop_times.txt (File extensions)

Field Name	Presence	Description  Used in Swiss implementation  Not used in Swiss implementation
stop_id	Required	 Identifies the serviced stop. All stops serviced during a trip must have a record in stop_times.txt. Referenced locations must be stops, not stations or station entrances. A stop may be serviced multiple times in the same trip, and multiple trips and routes may service the same stop.
location_group_id	Conditionally Required	 For referencing a location group in the location_groups file.
location_id	Conditionally Required	 For referencing a GeoJSON location. id from locations.geojson - stop_areas.area_id
stop_sequence	Required	 Order of stops for a particular trip. The values must increase along the trip but do not need to be consecutive. <i>Example: The first location on the trip could have a stop_sequence=1, the second location on the trip could have a stop_sequence=23, the third location could have a stop_sequence=40, and so on.</i> Travel within the same stop area or GeoJSON location requires two records in stop_times.txt with the same stop_id and consecutive values of stop_sequence.
arrival_time	Conditionally Required	

		Arrival time at a specific stop for a specific trip on a route. If there are not separate times for arrival and departure at a stop, enter the same value for arrival_time and departure_time. Scheduled stops where the vehicle strictly adheres to the specified arrival and departure times are timepoints. If this stop is not a timepoint, it is recommended to provide an estimated or interpolated time. If this is not available, arrival_time can be left empty. Further, indicate that interpolated times are provided with timepoint=0. If interpolated times are indicated with timepoint=0, then time points must be indicated with timepoint=1. Provide arrival times for all stops that are timepoints. Conditionally Required: - Required for the first and the last stop in a trip. - Forbidden when stop_times.start_pickup_drop_off_window or stop_times.end_pickup_drop_off_window are defined.
departure_time	Conditionally Required	✓ Departure time from a specific stop for a specific trip on a route. If there are not separate times for arrival and departure at a stop, enter the same value for arrival_time and departure_time. See the arrival_time description for more details about using timepoints correctly. The departure_time field should specify time values whenever possible, including non-binding estimated or interpolated times between timepoints. Conditionally Required: - Required for the first and the last stop in a trip. - Forbidden when stop_times.start_pickup_drop_off_window or stop_times.end_pickup_drop_off_window are defined.
start_pickup_drop_off_window	Conditionally Required	✓ Time that on-demand service becomes available in a GeoJSON location, stop area or stop. Conditionally Required: - Required if stop_times.stop_id refers to stop_areas.area_id or id from locations.geojson. - Forbidden if stop_times.arrival_time or stop_times.departure_time are defined.
end_pickup_drop_off_window	Conditionally Required	✓ Time that on-demand service ends in a GeoJSON location, stop area or stop. Conditionally Required: - Required if stop_times.stop_id refers to stop_areas.area_id or id from locations.geojson.

		- Forbidden if stop_times.arrival_time or stop_times.departure_time are defined.
pickup_type	Conditionally Forbidden	✓ Indicates pickup method. Valid options are: 0 or empty - Regularly scheduled pickup. 1 - No pickup available. 2 - Must phone agency to arrange pickup. 3 - Must coordinate with driver to arrange pickup. Conditionally Forbidden: - pickup_type=0 forbidden for stop_times.stop_id referring to stop_areas.area_id or id from locations.geojson. - pickup_type=3 forbidden for stop_areas.area_id or locations.geojson. - Optional otherwise.
drop_off_type	Conditionally Forbidden	✓ Indicates drop off method. Valid options are: 0 or empty - Regularly scheduled drop off. 1 - No drop off available. 2 - Must phone agency to arrange drop off. 3 - Must coordinate with driver to arrange drop off. Conditionally Forbidden: - drop_off_type=0 forbidden for stop_times.stop_id referring to stop_areas.area_id or id from locations.geojson. - Optional otherwise.
mean_duration_factor and mean_duration_offset	Conditionally Forbidden	✗ Together, mean_duration_factor and mean_duration_offset allow an estimation of the duration a rider's trip will take, in minutes, using the on-demand service in a GeoJSON location or stop area. Data consumers are expected to use mean_duration_factor and mean_duration_offset to make the following calculation: Conditionally Forbidden: - Forbidden if stop_times.stop_id does not refer to a stop_areas.area_id or an id from locations.geojson. - Optional otherwise.
safe_duration_factor and safe_duration_offset	Conditionally Forbidden	✗ Together, safe_duration_factor and safe_duration_offset allow an estimation of the longest amount of time a rider can expect the on-demand service in a GeoJSON location or stop areas may require, in minutes, for 95% of trips. Data consumers are expected to use safe_duration_factor and safe_duration_offset to make the following calculation: Conditionally Forbidden: - Forbidden if stop_times.stop_id does not refer to

		a stop_areas.area_id or an id from locations.geojson. - Optional otherwise
pickup_booking_rule_id	Optional	✗ Identifies the boarding booking rule at this stop time. Recommended when pickup_type=2.
drop_off_booking_rule_id	Optional	✗ Identifies the boarding booking rule at this stop time. Recommended when pickup_type=2.

Tabelle 2: Field definitions stop_times.txt

12.1.3.5 Locations.geojson

General Information:

- This file uses a subset of the GeoJSON format, described in [RFC 7946](#).
- The locations.geojson file must contain a FeatureCollection.
- A FeatureCollection defines various stop locations where riders may request pickup or drop off.
- Every GeoJSON Feature must have an id. The id belongs to the same namespace as stop_id in stops.txt and area_id in stop_areas.txt, called "stop locations".
- Every GeoJSON Feature should have objects and associated keys according to the table below:

Field Name	Presence	Description ✓ Used in Swiss implementation ✗ Not used in Swiss implementation
- type	Required	✓ "FeatureCollection" of locations.
- features	Required	✓ Collection of "Feature" objects describing the locations.
- type	Required	✓ "Feature"
- id	Required	✓ Location ID belonging to the same namespace as stops.stop_id. It is forbidden to define an id from locations.geojson with the same value as a stops.stop_id.
- properties	Required	✓ Location property keys.
- stop_name	Optional	✓ Indicates the name of the location as displayed to riders.
- stop_desc	Optional	✗ Meaningful description of the location to help orient riders.
- zone_id	Conditionally Required	✗ Identifies the fare zone for a stop. Conditionally required: - Required if fare_rules.txt is defined. - Optional otherwise.
- stop_url	Optional	URL of a web page about the location. If provided, the URL should be different from the agency.agency_url and the routes.route_url field values.

- geometry	Required	✓ Geometry of the location.
- type	Required	✓ Must be of type: - "Polygon" - "MultiPolygon"
- coordinates	Required	✓ Geographic coordinates (latitude and longitude) defining the geometry of the location.

12.2 Extension under Development: GTFS-Flex (Swiss Profile)

In Switzerland GTFS-Flex is currently mainly planned to be used to represent on-demand offers as described in this document [LINK](#).

In **GTFS-Flex terminology** (as mentioned in section 12.1.2.2) the **on-demand offers constitute** as ‘dial-a-ride services’ (called “On-Demand Flächenverkehr”), ‘route deviation services’ (called “On-Demand Korridorverkehr”), and ‘point deviation or checkpoint services’ (called “On-Demand Linienverkehr”). The ‘point-to-zone service’ is considered a specific variant of the ‘dial-ride-service’ by us. The ‘hail-and-ride service’ is treated separately as part of vehicle pooling (the concept is not yet published), for which we suggest the introduction of a GTFS-Pool extension.

Note that we will not offer the “On-Demand Linienverkehr” with GTFS Flex, because this line-based on-demand offers are represented using the regular timetable formats.

Based on the on-demand offers in Switzerland we will be using the data-set files as described in section 0. The official schematic representation will change as shown in Figure 38.

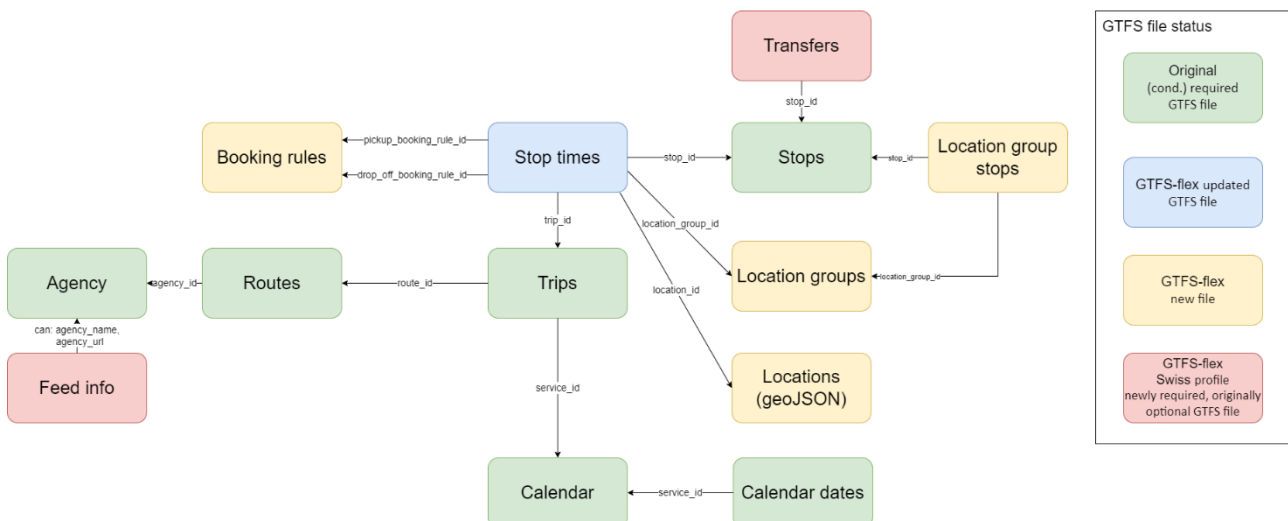


Figure 38: Schematic representation of GTFS-Flex - Swiss Profile

In summary, we will not be using all fields in the existing GTFS-Flex extension and extend it by including feed_info.txt and transfers.txt. This is required to ensure a proper integration with the existing transportation infrastructure.

13 Possible other Extensions in the near Future

GTFS Occupancies in GTFS-RT	Realtime information about occupancy	Will be studied
GTFS-Pathways	Transfer paths	Will be studied
GTFS Shapes (file, not an extension)	The path public transport takes	Will be studied
GTFS-Pool	Pooling service	Will be studied