NSW Trains 4Trak GTFS & GTFS-R Technical Documentation

September 2016 | Version: 1.0

Contents

Author: Transport for NSW

Date: September 2016

Version: 1

1 Overview

The GTFS (general transit feed specification) feed provides static timetabling information about a transit network, including stops, routes, trip schedules and other information like a trip's geometry and information about the transit agency running the trips. This data is suitable for planning trips as well as presenting trip information to a customer via a graphical user interface. The feed also contains real time information about the transit network, including which trips are currently running on the network, the position of each of the vehicles completing the trips, which of the scheduled stops have been made, the difference between the scheduled and actual departure times as well as the actual and scheduled arrival times.

The GTFS bundle feed is composed of a number of CSV files which are all contained within a single zip file. Each of the files contain scheduled information about a different aspect of the NSW trains transit network. For example stops_times.txt contains information about the scheduled time and order of stops for different trips.

This document specifies the files which are included in the GTFS bundle feed as well as the GTFS real-time feed. Each of the fields in the files are also documented and their meanings are explained.

The bundle and real time components of the feed both conform to the GTFS. There are some fields which are not requirements of the specification but are permissible extensions.

The GTFS real time portion of the feed has two component files a trip update file and a vehicle position file. Both files are protocol buffers. Protocol buffers are a mechanism developed by google for serialising data. Both protocol buffers contain trip descriptors which specify the scheduled trips that are running and are detailed with real time information.

The trip update file provides information about which stops have been completed by the running trip. It also contains information about the deviation between the actual arrival and scheduled arrival times as well as the actual and scheduled departure times for each of the stops.

The vehicle position file contains positional information for each of the currently running trips. The positional information includes a longitude, latitude and bearing.

2 Data Feed Access

The GTFS feed data can be accessed by sending a HTTP get request to the appropriate URL. The appropriate URL depends on which of GTFS files need to be accessed.

The GTFS feed data can be accessed by sending a HTTP get request to the appropriate URL. The appropriate URL depends on which of GTFS files need to be accessed.

These URLs and associated accounts can be provided via registration to the Transport for NSW Open Data program

https://opendata.transport.nsw.gov.au/site/en_us/home.html

3 General Transit Feed Specification

3.1.1 GTFS Compliance

The GTFS bundle is compliant with the specification reference published by google on 3 February 2016. The GTFS real time feed is also compliant with the GTFS reference published by google on 26 February 2015. The references for both feed components specifications can be found at the following URLs:

- GTFS Bundle reference: https://developers.google.com/transit/gtfs/reference
- GTFS Real time reference: https://developers.google.com/transit/gtfs-realtime/reference

3.1.2 Publication Schedule

The GTFS bundle and GTFS real time have two independent publication schedules. A new GTFS bundle is published every 24 hours at 1am Australian Eastern standard time. Both the trip update and vehicle position files in the GTFS real time are published every 30 seconds simultaneously.

3.1.3 Validity Period

The data provided in the GTFS bundle will be valid for a hundred days after the bundle is initially published unless changes are made to the timetable since the bundle was downloaded.

3.1.4 Value Quoting

All fields with in the GTFS bundle are enclosed in double quotes. This is not the case within the vehicle position or trip update files.

3.1.5 Feed Size

The size of the GTFS bundle will vary depending on how many trips are scheduled over the next 100 days and on how many stops have been scheduled to be part of the trips. The bundle will typically be in the order of 10 MB but this is subject to variability.

The protocol buffers which make up the real time feed vary much more in size than the bundle. The size of the vehicle position files depend on the number of trips currently running. Its size is typically 2-20 KB.

The size of the trip update is dependent on the number of currently running trips, the size of the currently running trips and how much progress the running trips have made. The trip update file size is typically 10-60KB. Both real time files are substantially smaller in the early morning.

3.2 Agency.txt

The agency.txt file specifies values relevant to the agencies whose timetabling information is included in the feed.

The following fields are populated within the agency.txt file:

Field name	Description	Example
agency_id *Referenced in*: trips.txt, routes.txt	The agency_id is an ID which uniquely identifies a transport agency in the feed.	711
agency_name	The agency name is the full name of the transit agency.	NSW TrainLink Train for Regional Trains
agency_url	This value is the URL of the transit agency.	http://transportnsw.info
agency_timezone	This value is the time zone the transit agency is located.	Australia/Sydney
agency_lang	This field contains the ISO 639-1 code for the language used by the transit agency.	en
agency_phone	This field contains a single telephone number for the transit agency.	131500

3.3 Calendar.txt

The Calendar.txt file specifies two dates between which the service is valid for as well as the days of the week that the service will run on.

The following fields are populated within the calendar.txt file:

Field name	Description	Example
service_id *Referenced in*: Trips.txt, calendar_dates.txt	The service_ID is a value which uniquely identifies a service with in the feed. It contains three fields. The first field contains the run number of the service. The second field represents the day of the week the service will run on. It is a seven digit binary number converted into a decimal number which indicates on which days of the week the service runs. Each of the digits in the binary value represents a day of the week with the left most value indicating a Monday and the right most value indicating a Sunday. For example 4 is 0000100 as a 7 digit binary number. This value indicates the service will run on a Friday only. The third field specifies which week of the year the service will run on.	166.4.24
monday	This field is 1 if the service runs on Monday. The field is 0 if it does not.	1
tuesday	This field is 1 if the service runs on Tuesday. The field is 0 if it does not.	0
wednesday	This field is 1 if the service runs on Wednesday. The field is 0 if it does not.	1

Field name	Description	Example
thursday	This field is 1 if the service runs on Thursday. The field is 0 if it does not.	0
friday	This field is 1 if the service runs on Friday. The field is 0 if it does not.	1
saturday	This field is 1 if the service runs on Saturday. The field is 0 if it does not.	0
sunday	This field is 1 if the service runs on Sunday. The field is 0 if it does not.	1
start_date	The start date specifies the Monday of the week this service is valid for. The date is in YYYYMMDD format.	20160606
end_date	The end date specifies the Sunday of the week this service is valid for. The date is in YYYYMMDD format.	20160612

3.4 Calendar_dates.txt

The calendar dates file can be used to specify any exceptions to the normal running of a service. This file is not populated in GTFS bundle V 1.5.0 but this may change.

Field name	Description	Example
service_id *Referenced in*: Calandar.txt, Trips.txt,	The service_ID is a value which uniquely identifies a service within the feed. It contains three fields. The first is the run number of the service. The second field represents the day of the week the service will run on. It is a seven digit binary number converted into a decimal number which indicates on which days of the week the service runs. Each of the digits in the binary value represents a day of the week with the left most value indicating a Monday and the right most value indicating a Sunday. For example 4 is 0000100 as a 7 digit binary number. This value indicates the service will run on a Friday only. The third field specifies which week of the year the service will run on.	166.4.24
date	The date specifies the day that this exception to the normal timetable applies to. This date is in YYYYMMDD format.	20160620
exception_type	The calendar_dates file can be used to add or remove services from the normal timetable. A value of 1 indicates that the service is an addition to the normal timetable. A value of 2 indicates that this service has been removed from the normal timetable.	1

3.5 Routes.txt

The routes.txt file contains all the transit routes relevant to the trips in the feed.

The following fields are populated within the routes.txt file:

Field name	Description	Example
route_id *Referenced in*: trips.txt, real time trip descriptors	The route_id is a unique identifier for a particular route. The ID is composed of three separate fields. The first field indicates it is a unique ID created by 4Tel. The second field is a T or C to indicate if it is a coach route or train route. The final field is the run number or region abbreviation which corresponds to this route.	4T.T.NT31
agency_id *Referenced in*: agency.txt	The agency is a value which uniquely identifies a transport agency. In the context of routes.txt it indicates that the route is associated with the specified agency.	711
route_short_name	This field contains a short abstract identifier for a route.	31
route_long_name	This field contains the full name of a particular route.	North Coast NSW Line (Service no. 031)
route_desc	This field specifies which transit network a particular route is on.	Regional Trains and Coaches Network
route_type	The route type describes the type of transport used on the route. The route type values used are 204 which specifies a regional coach service, 100 which specifies a railway service and 106 which specifies a regional railway service.	106
route_color	This field specifies which colour a route should be illustrated with. The colour must be a six-character hexadecimal number.	F6891F
route_text_color	This field specifies what colour should be used for any text which is used in the route illustration. It must be specified with a six-character hexadecimal number.	FFFFFF

3.6 Trips.txt

The trips.txt file is a list of all the trips which a timetable will run over the next 100 days.

The trips.txt file is composed of the fields set out in the table below:

Field name	Description	Example
route_id *Referenced in*: route.txt	The route_id is a unique identifier for a particular route.	4T.C.135

Field name	Description	Example
service_id *Referenced in*: calendar.txt	The service_ID is a value which uniquely identifies a service with in the feed. It contains three fields. The first is the run number of the service. The second field represents the day of the week the service will run on. It is a seven digit binary number converted into a decimal number which indicates on which days of the week the service runs. Each of the digits in the binary value represents a day of the week with the left most value indicating a Monday and the right most value indicating a Sunday. For example 4 is 0000100 as a 7 digit binary number. This value indicates the service will run on a Friday only. The third field specifies which week of the year the service will run on.	135.4.24
trip_id *Referenced in*: stop_times.txt, real time trip descriptors	The trip_id is the unique identifier for a particular trip. It is composed of two fields. The first is the run number of the trip. The second value is a unix timestamp which indicates the planed start of the trip.	135.14655 45600
trip_headsign	The trip headsign is a station name used to indicate the destination of the trip.	Taree
trip_short_name	This field contains the run number of the trip.	135
direction_id	This field indicates whether a trip is inbound or outbound relative to Central station. 0 indicates an outbound trip and 1 indicates an inbound trip.	0
block_id	The block id is not populated as of version 1.5.0 but will be used to group trips which are performed consecutively by the same vehicle.	
shape_id	This value is a unique identifier for a shape from the shapes.txt file and means this trip is expected to move through the geometry defined by that shape.	4T.C.135.1.H
wheelchair_accessible	This field is not populated as of the 1.5.0 version of this feed.	
bikes_allowed	This field is not populated as of the 1.5.0 version of this feed.	
trip_note	A trip note provides additional information about a trip which does not fit into the regular fields. This can be populated with an ID which will correspond to a notes.txt entry. Not all trips will have an accompanying note.	70005
route_direction	This field uses the first and last stop names in the trip to indicate the directionality of the trip.	Broadmeadow to Taree

3.7 Stops.txt

The stops.txt file contains all the locations that all the trips will stop at and the information relevant to them.

The trips.txt file is composed of the fields set out in the table below:

Field name	Description	Example
stop_id *Referenced in*: stop_times.txt	This field contains an ID that uniquely identifies a stop or station.	233621
stop_name	This field contains the full name of a particular stop or station.	Aberdeen Station Platform 1
stop_lat	The stop_lat field contains the latitude of a stop or station.	-32.166969
stop_lon	The stop_lon field contains the longitude of a stop or station.	150.892016
location_type	The location type indicates whether a stop is a physical structure containing multiple stops or it is a single stop. If the value is 0 it indicates a single stop. A value of 1 indicates it is a parent station which contains multiple stops.	0
parent_station	If the stop is a parent station then this field is blank. Otherwise it contains the stop_id of the location which contains the stop.	233610
wheelchair_boarding	This field is 0 by default unless the stop is specifically set to wheelchair accessible in 4Trip. If the stop is set to wheel chair accessible this field will be set to 1.	0
platform_code	The platform code indicates which platform a entries stop corresponds to. If the stop is a parent station or coach stop this value is blank.	1
stop_timezone	This field is not populated in the 1.5.0 version of this feed.	

3.8 Stop_times.txt

The stops_times.txt file contains a list of all the stops completed by all the trips and the time that the stop happens.

The stop_times.txt file is composed of the fields set out in the table below:

Field name	Description	Example
trip_id *Referenced in*: Trips.txt	The trip_id is the unique identifier for a particular trip. It is composed of two fields. The first is the run number for the trip. The second value is a unix timestamp which indicates the planed start of the trip.	135.1465545600

Field name	Description	Example
arrival_time	The arrival_time field specifies the time a trip arrives at a particular stop. If a trip runs over more than one day a value greater than the maximum usually allowed in 24 hour time will be used. For example if a trip runs from 11pm to 1.30am the arrival time for the final stop will be 25:30:00. Times in stop_times.txt are eight digit values in HH:MM:SS format.	18:00:00
departure_time	The departure_time field specifies the time a trip departs from a particular stop. If a trip runs over more than one day a value greater than 24 hour time can be used. For example if a trip runs from 11:00 pm to 01:30 am the departure time for the final stop will be 25:30:00. Times are eight digits in HH:MM:SS format.	25:30:00
stop_id	This field contains the unique ID for this entries stop.	229268
stop_sequence	The stop_sequence field identifies the order a stop is in the entries corresponding trip.	1
pickup_type	This field specifies whether a trip will include pick up at a particular stop or if special instruction will need to be given for pick up at this stop. 0-regular pick up 1-no pick up 2-Must phone agency for pick up. 3-Must coordinate with driver for pick up.	0
drop_off_type	This field specifies whether a trip will drop passengers at a particular stop or if special instruction will need to be given for passengers to be picked up at this stop. 0-regular pick up 1-no pick up 2-Must phone agency for drop off. 3-Must coordinate with driver for drop off.	0
timepoint	This field indicates whether a particular stop time should be considered approximate or exact. 0-indicates the time is approximate 1- indicates the time should be considered exact	1
stop_note *Referenced in*: Notes.txt	This field contains an ID that associates the stop with a note in notes.txt. Stop notes contain additional information about a stop which is not included in the stop entries normal fields.	70002
shape_dist_travelled *Referenced in*: Shapes.txt	This field specifies how far along a trip a particular stop is in meters.	24252.3918665822

3.9 Shapes.txt

The shapes.txt file contains a list of all the shapes relevant to the trips. A shape is a sequence of latitudes and longitudes combined with a sequence value which indicates there order. A shape defines the geometry of its corresponding trip.

The shapes.txt file is composed of the fields set out in the table below:

Field name	Description	Example
shape_id *Referenced in*: trips.txt	This field contains an ID which uniquely identifies a shape.	4T.T.SP41.2.H
shape_pt_lat	This field specifies the latitude of a single coordinate in a shape.	-34.7582496747171
shape_pt_lon	This field specifies the longitude of a single coordinate in a shape.	149.719581986594
shape_pt_sequence	This field specifies the order a particular coordinate is in within the shape.	1
shape_dist_traveled *Referenced in*: stop_times.txt	This field specifies the distance between a given point and the first point in the shape if you travel along the shape.	0

4 General Transit Feed Specification – Real time

4.1 Real time

The GTFS real time feed is composed of two protocol buffer files. A trip update file and a vehicle position file. The trip update and vehicle position files have common fields. One set of common fields is the header and the other is the trip descriptor.

4.1.1 Header

Field name	Description	Example
gtfs_realtime_version	This field contains the version or the GTFS realtime	1.0
timestamp	This timestamp indicates when the file was created.	1465534901
Incrementality	This field will always be populated with the full data set value. This indicates that the real time files contain a full snap shot of the running trips.	FULL_DATASET

4.1.2 Trip Descriptor

Field name	Description	Example
trip_id *Referenced in*: trips.txt	This value indicates which trip a real time entity is associated with.	725.1465534500

Field name	Description	Example
schedule_relationship	This value indicates whether the trip is scheduled in the timetable.	SCHEDULED
route_id *Referenced in*: routes.txt	This value indicates which route the vehicle is traveling along.	4T.C.725

4.2 Message Trip Update

Each trip update file contains a header and a feed entity for each of the currently running trips. Each feed entity contains a single trip descriptor and a stop time update for each of the stops that the vehicle completing the trip has arrived at.

Field name	Description	Example
id	This is just a single number used to differentiate the different entities in the feed.	1
stop_sequence	This value specifies the order of a stop is in for the trip.	4

arrival {
delay:
}
	This value specifies the difference between the expected arrival time and the actual arrival time in seconds.	284

departure {
delay:
}
	This value specifies the difference between the expected departure time and the actual departure time in seconds.	332

stop_id: *Referenced in*: stops.txt	This value is the id for a stop which the trip has completed.	24002

4.3 Message Vehicle Position

Each vehicle position file contains a header and a feed entity for each of the currently running trips. All the fields in the vehicle position trip entities can be seen in the table below.

Field name	Description	Example
id	This field contains a single number used to differentiate the entities in the feed.	1
latitude	This field specifies the latitude of the vehicle in decimal degrees.	-29.76868
longitude	This field specifies the longitude of the vehicle in decimal degrees.	151.49377
bearing	This field specifies the bearing of the vehicle measured in degrees from the horizontal.	59.0
timestamp	This timestamp is the time the last latitude, longitude and bearing for this vehicle were reported.	1465532033
congestion_level	This value is always populated with a default value which indicates an unknown level of congestion in the GTFS V 1.5.0	UNKNOWN_CONGESTION_LEVEL

Field name	Description	Example
stop_id *Referenced in*: stops.txt	The stop_id value indicates the last location the vehicle stopped at.	23604

vehicle {
id:
}
	The Vehicle ID is the run number of the trip the vehicle is currently completing.	142

vehicle {
label:
}
	The vehicle label is a value which customers can use to identify a vehicle.	11:05am (142) Moree Town - Grafton City

5 Settings

There are multiple settings which are configurable for both the GTFS Bundle and the GTFS real time components of the feed.

5.1 System Configuration

The GTFS feed is accompanied by a system configuration XML file which can be used to specify values for different settings in the GTFS feed. This file is named system_config.xml and is in the D:\Apps directory.

The system configuration can be used to specify the operator that the feed instance is being used by, the DoUpload setting flag value, the server credentials for the bundle as well as the real time files, the service area's which should be reported on and the valid run number prefix's.

The DoUpload attribute determines whether the feed will post the data to a server. If this value is set to false the feed will run as normal but will not upload any files.

The operator setting informs which stop codes are output by the feed. Different rail operators may use different sets of location codes. Setting the operator attribute to NSW Trains will mean that the feed output will contain only the NSW Trains stop codes.

There are a minimum of three server credential elements in the system configuration file. One for each file type in the GTFS feed. Each element contains three attributes. A URL attribute, as well as a username and password. The URL element contains the URL which that file will be posted to. The value of the username and password elements will be the username and password used when posting to the accompanying URL. Additional server credential elements can be added to the system configuration file if the feed needs to be posted to multiple servers.

The system configuration can be used to filter the feed output based on allowed run numbers. The child elements of element gtfsValidRunNumbers are called gtfsValidRunNumber. Each of them contain a value which specifies an allowable run number prefix. This list contains the allowable run number prefixes in both the bundle and real time components of the GTFS feed. If a trips run number does not contain one of the allowable prefix's then it will not be output in either the bundle or real time components of the feed.

The system configuration can also be used to filter the real time feed based upon region. The gtfsServiceAreas element in the system configuration file contains gtfsServiceArea child elements. The name attributes in these elements are what

control which regions are part of the real time feed. If a run is not in one of the service area's specified by these elements it will be excluded from the real time feed.

5.2 4Trak Configuration Variables

The GTFS settings that can be specified as 4Trak configuration variable's do not have to be set and will default to what will likely be suitable values.

The following table contains configuration variables which can be set in the 4Trak admin tool. The variables are accompanied by what the variables default to if these configuration values are not specified with the 4Trak admin tool in the table below.

Configuration Vehicle	Description	Default
GTFS_CREATE_DEBUG_FILES	This specifies whether or not copies of the feeds files should be stored. A value of true will cause the files to be stored and if the variable is set to false the files will not be stored. Debug files are stored in the following directory D:\Apps\GTFS\GTFS\output_files	true
GTFS_WS_RETRY_COUNT	This specifies how many times the feed should attempt to download the timetable information in the case of a network problem.	3
GTFS_WS_RETRY_WAIT	This specifies how long the feed should wait before retrying a web service call if the feed fails to successfully make a web service call.	30
GTFS_BUNDLE_UPLOAD_TIMEOUT	These values specify how long the feed upload connection can be idle for before the upload will timeout. One is applicable to the real time feed the other is applicable to the bundle.	10
GTFS_REALTIME_UPLOAD_TIMEOUT		10