

SCHEDULE MASTERS, INC.

TMS2GT VERSION 4.0

EDWARD SITARSKI
VICE-PRESIDENT OF TECHNOLOGY
SCHEDULE MASTERS, INC.

April 2010

INTRODUCTION	1
HOW TO USE TMS2GT	1
Installation	1
Running TMS2GT	1
GOOGLE TRANSIT FILES	2
agency.txt (Required)	2
stops.txt (Required)	3
routes.txt (Required)	3
trips.txt (Required)	3
stop_times.txt (Required)	3
calendar.txt (Required)	3
calendar_dates.txt (Optional)	4
shapes.txt (Optional)	4
fare_attributes.txt (Optional)	4
fare_rules.txt (Optional)	4
frequencies.txt (Optional)	4
transfers.txt (Optional)	4

TMS2GT

INTRODUCTION

TMS2GT is a script that transforms the TMSweb files into Google Transit (GT) format. The TMSweb files are generated from the TMS (The Master Scheduler) application.

GT organizes its data somewhat differently than TMS, and requires some extra information. As far as possible, TMS2GT transforms the data from TMS into correct GT format.

This document outlines how to use TMS2GT as well as how the TMS data is mapped to the Google Transit representation. Complete details on the Google Transit format can be found at http://code.google.com/transit/spec/transit_feed_specification.html.

TMS2GT is capable of creating the “shapes.txt” file in GT. This file is used to draw the lines on the map showing the road paths of the vehicles. This must be set up using the TMSwebBusPathEditor or TMSwebBusPatternsEditor beforehand.

HOW TO USE TMS2GT

INSTALLATION

Run “setup_TMS2GT.exe”.

RUNNING TMS2GT

TMS2GT transforms TMSweb .txt files into Google Transit format. The first step is to create the TMSweb .txt files from TMS in the usual way (normally these get written to a folder called “Web-Based Trip Planner Data Files”, but you can rename or move them to another folder).

Now you can convert the files to Google Transit format with TMS2GT.

Run TMS2GT.exe.

TMS2GT itself is pretty self-explanatory. First, you specify the folder where the TMSweb .txt files are (these are in a “Web-Based Trip Planner Data Files” folder). If you used the TMSwebBusPathEditor or the TMSwebBusPatternsEditor, there will be a file in this folder containing the paths called TMSweb-BusPaths.shv. This file allows TMS2GT to create the “shapes.txt” file which is used by Google Transit to draw the road routes taken by the vehicles. Then you specify the folder where you want the Google Transit files to go. TMS2GT will default to a folder called “Google Transit Data Files” and will try to create a new folder for the GT files if it does not exist.

The “Duplicate Nodes Options” section gives you control over how you want to deal with duplicate nodes (stops). TMS Nodes are converted to GT stops, and the GT upload will create

warnings if the GPS coordinates (latitude/longitude) between stops are sufficiently close together. These are just warnings only – they do not prevent it from working.

Nodes with the same latitude/longitude can legitimately happen in TMS as they are required to differentiate between the first departure and last arrival on a loop route. However, they can also be caused by data input errors that should be corrected.

TMS2GT provides three options for dealing with this:

1. Allow duplicate nodes (do nothing)
2. Remove duplicates based on TMS Node Equivalences
3. Remove duplicates based on matching GPS coordinates

The first option does not remove duplicates. Choosing this option will still create valid GT files, but the GT upload will create warnings if there are duplicate stops. The second option removes duplicates based on the Node Equivalences information in TMS. The third option removes duplicates based on whether they have the same latitude longitude.

When nodes are removed, the first occurrence of the node in the TMS extracts will be used as the “standard” node, and will become the GT stop name. All subsequent matching nodes will not be included in the GT files, and all references to the duplicates will be changed to the “standard” node. From the perspective of the GT files created, it will appear as if the duplicates never existed.

TMS2GT writes its actions to a “TMS2GTmessages.txt” file (written in the folder with the TMS .txt files). You may want to open this file with Excel for better readability, or a text editor (Notepad, Word, etc.).

Now, press the “Do Conversion” button. Some status messages will appear as TMS2GT does its work. After a short while, you should see a “Files converted successfully” message.

TMS2GT conveniently remembers its parameters from the last run. If you want to re-run TMS2GT, you do not need to set anything the second time.

To reduce GT warnings, TMS2GT removes unused stops before creating the GT files. An unused stop is a stop that does not have a bus arriving or departing from it. The removed stops appear as messages in the “TMS2GTmessages.txt” file (written in the folder with the TMS .txt files). Unused stops play no part in GT.

After running successfully, TMS2GT creates a “GoogleTransitUpload.zip” file in your Google Transit folder. This is what you need to send to Google Transit for the upload.

GOOGLE TRANSIT FILES

This section discusses the data files and fields created for Google Transit in detail.

AGENCY.TXT (REQUIRED)

If this file is not present, TMS2GT will automatically generate a template file that you will need edit manually. After editing it manually, run TMS2GT again. If this file exists, TMS2GT will not overwrite it.

See http://code.google.com/transit/spec/transit_feed_specification.html for details.

STOPS.TXT (REQUIRED)

stop_id	Internally generated by TMS2GT
stop_name	Long name of stop
stop_lat	Latitude of stop
stop_lon	Longitude of stop
location_type	0 = regular stop

ROUTES.TXT (REQUIRED)

route_id	Internally generated by TMS2GT
route_short	Short name of route (the route number from TMS)
route_long	Long name of route (long name from TMS)
route_type	3 (GT code for a bus)

TRIPS.TXT (REQUIRED)

trip_id	Internally generated by TMS2GT
route_id	Refers to corresponding route_id in GT routes.txt file
service_id	Refers to corresponding service_id in GT services.txt file
trip_headsign	Direction of route (from TMS)
block_id	Vehicle block id (from TMS)
*shape_id	Shape id in shapes.txt file – only defined if a .shv file is present

STOP_TIMES.TXT (REQUIRED)

trip_id	Internally generated by TMS2GT
stop_id	Refers to corresponding stop_id in GT stops.txt file
arrival_time	Arrival time of the vehicle at this from TMS
departure_time	Departure time of the vehicle at this stop from TMS
stop_sequence	Sequence of this stop in the pattern (generated by TMS2GT)
*shape_dist_traveled	Distance along shape – only defined if a .shv file is present.

The difference between “arrival_time” and “departure_time” is the dwell time of the vehicle at the stop.

CALENDAR.TXT (REQUIRED)

This file indicates which days of the week a particular service is available. This is set up from the ServiceCalendarDefaults.txt file in TMSweb, which comes from the TMS.

service_id	Service id (in services.txt).
monday	0=unavailable, 1=available
tuesday	0=unavailable, 1=available
wednesday	0=unavailable, 1=available
thursday	0=unavailable, 1=available
friday	0=unavailable, 1=available

saturday	0=unavailable, 1=available
sunday	0=unavailable, 1=available
start_date	Start date
end_date	End date

CALENDAR_DATES.TXT (OPTIONAL)

This file indicates exceptions to the defaults defined in the calendar.txt file. This is set up from the ServiceCalendarExceptions.txt file, which comes from TMS.

date	Date of the exception
service_id	Service ID (in services.txt)
exception_type	1=service is added, 2=service is removed

SHAPES.TXT (OPTIONAL)

Generated by TMS2GT if a TMSweb-BusPaths.shv file is present in the TMS file folder. To generate this file, see the documentation on the TMSwebBusPathEditor.

It is important that the bus paths are defined properly, that is, each path starts and ends from the stops it connects. This is critical to create proper files for Google Transit as each stop is positioned along the path by distance.

TMS2GT ensures that it gets the right distances by internally adding the “from” stop and “to” stop to the beginning and end of the path. This allows it to make valid Google Transit files, however, it does not make the paths right.

shape_id	ID of this shape.
shape_pt_lat	Latitude of this shape point (from .shv file)
shape_pt_lng	Longitude of this shape point (from .shv file)
shape_pt_sequence	Sequence number of this point along the path
shape_dist_traveled	Distance (km) of this point from the beginning of path.

FARE_ATTRIBUTES.TXT (OPTIONAL)

Not currently generated by TMS2GT.

FARE_RULES.TXT (OPTIONAL)

Not currently generated by TMS2GT.

FREQUENCIES.TXT (OPTIONAL)

Not currently generated by TMS2GT.

TRANSFERS.TXT (OPTIONAL)

Generated by TMS2GT if a Transfers.txt file is present (must be created manually). The TMS Transfers.txt supports a more powerful way to maintain transfers than the Google Transit approach. See the TMSwebService documentation for details.