[Public]

FaresV2 base implementation for the validator

https://bit.ly/faresv2-validator

Forewords

This document intends to identify how to update the Canonical GTFS Schedule Validator after the adoption of the Fares V2 base implementation.

Materials

- Fares v2 document (maintained by the spec team): [Public] Fares v2
- Pull Request on google/transit
- Schedule Validator Epic GitHub Issue
- First Pull Request by Brian Ferris: for new files that were added in Fares v2 base implementation)
    - New rules
        - `fare_transfer_rule_duration_limit_type_without_duration_limit` (ERROR)
        - `fare_transfer_rule_duration_limit_without_type` (ERROR)
        - `fare_transfer_rule_invalid_transfer_count` (ERROR)
        - `fare_transfer_rule_missing_transfer_count` (ERROR)
        - `fare_transfer_rule_with_forbidden_transfer_count` (ERROR)
        - `invalid_currency_amount` (ERROR)
    - Affected existing rules
        - all rules that check for files and field type compliance (`missing_required_field, invalid_interger, number_out_of_range`, etc)
        - `invalid_enum_value` (WARNING)
            - Because there are new enum values to gtfs duration limit and fare_transfer_type)
    - This Pull Request has been merged
- Second Pull Request by Brian Ferris (for changes to the pre-existing files that were added in the Fares v1 extension)
    - New rules
        - `forbidden_file` (ERROR)

- Affected existing rules
  - `missing_required_file` (ERROR)
- This Pull Request is currently paused, and an issue has been opened on google/transit in an attempt to unblock it.

Impact on production data

Who is producing fares v2 data?

This list comes from the Mobility Database, by filtering `feature` field with `fares-v2`. A red background indicates that the data contains validation errors. The number of red lines in this table (10) is smaller than the number of sources that were flagged by the validator acceptance test (73), and we believe that because the `fares-v2` field is added manually, we have sources that have fares v2 data but no `fares-v2` field associated.

Country	State	Municipality	Provider	Direct download URL
US	California	San Diego	San Diego International Airport, Metropolitan Transit System (MTS)	https://www.sdmts.com/google_transit_files/google_transit.zip
US	California		Palo Verde Valley Transit Agency	http://data.trilliumtransit.com/gtfs/paloverde_valley-ca-us/paloverde_valley-ca-us.zip
US	California	Arcata	Arcata & Mad River Transit System	http://data.trilliumtransit.com/gtfs/humboldtcounty-ca-us/humboldtcounty-ca-us.zip
US	California	Crescent City	Redwood Coast Transit (RCT)	http://data.trilliumtransit.com/gtfs/delnorte-ca-us/delnorte-ca-us.zip
US	California	Bishop	Eastern Sierra Transit	http://data.trilliumtransit.com/gtfs/easternsierra-ca-us/easternsierra-ca-us.zip
US	California	Santa Maria	Santa Maria Area Transit	http://data.trilliumtransit.com/gtfs/smat-ca-us/smat-ca-us.zip
US	California	Santa Barbara	Ventura County Transportation Commission	http://data.trilliumtransit.com/gtfs/vctc-ca-us/vctc-ca-us.zip
US	California		Palos Verdes Peninsula Transit Authority (PVPTA)	http://data.trilliumtransit.com/gtfs/pvpta-ca-us/pvpta-ca-us.zip
US	California	Simi Valley	Simi Valley Transit	http://data.trilliumtransit.com/gtfs/simivalley-ca-us/simivalley-ca-us.zip
US	California	Bakersfield	Kern Transit	http://data.trilliumtransit.com/gtfs/kerncounty-ca-us/kerncounty-ca-us.zip
US	California	Tulare	Porterville Transit	http://data.trilliumtransit.com/gtfs/porterville-ca-us/porterville-ca-us.zip
US	California	Hollister	San Benito County Express (SBCE)	http://data.trilliumtransit.com/gtfs/sanbenitocounty-ca-us/sanbenitocounty-ca-us.zip
US	California		Lake Transit Authority (LTA)	http://data.trilliumtransit.com/gtfs/laketransit-ca-us/laketransit-ca-us.zip
US	California	Ukiah	Mendocino Transit Authority	http://data.trilliumtransit.com/gtfs/mendocino-ca-us/mendocino-ca-us.zip
US	Colorado		El Dorado Transit	http://data.trilliumtransit.com/gtfs/eldoradotransit-ca-us/eldoradotransit-ca-us.zip

US	California	Sacremento	Placer County Transit	http://data.trilliumtransit.com/gtfs/placercounty-ca-us/placercounty-ca-us.zip
US	California	Yuba City	Yuba-Sutter Transit	http://data.trilliumtransit.com/gtfs/yubasutter-ca-us/yubasutter-ca-us.zip
US	California	Davis	Unitrans	https://unitrans.ucdavis.edu/media/gtfs/Unitrans_GTFS.zip
US	California	Nevada County	Nevada County Connects (Gold Country Stage)	http://data.trilliumtransit.com/gtfs/goldcountrystage-ca-us/goldcountrystage-ca-us.zip
US	California	Merced	Merced County Transit (The Bus)	http://data.trilliumtransit.com/gtfs/mercedthebus-ca-us/mercedthebus-ca-us.zip
US	California	Jackson	Amador Transit	http://data.trilliumtransit.com/gtfs/amador-ca-us/amador-ca-us.zip
US	California	Hesperia	Victor Valley Transit Authority (VVTA)	http://data.trilliumtransit.com/gtfs/victorville-ca-us/victorville-ca-us.zip
US	California	Los Angeles	Norwalk Transit System (NTS)	http://data.trilliumtransit.com/gtfs/nts-ca-us/nts-ca-us.zip
US	California	Corona	Corona Cruiser	http://data.trilliumtransit.com/gtfs/corona-ca-us/corona-ca-us.zip
US	California	Ridgecrest	Ridgerunner	http://data.trilliumtransit.com/gtfs/cityofridgecrest-ca-us/cityofridgecrest-ca-us.zip
US	California	Compton	Compton Renaissance	http://data.trilliumtransit.com/gtfs/compton-ca-us/compton-ca-us.zip
US	California		DowneyLINK	http://data.trilliumtransit.com/gtfs/downey-ca-us/downey-ca-us.zip
US	Maryland		Maryland Transit Administration Metro Subway	https://mdotmta-gtfs.s3.amazonaws.com/mdotmta_gtfs_metro.zip
US	California	Huntington Park	Huntington Park Express	http://data.trilliumtransit.com/gtfs/huntingtonpark-ca-us/huntingtonpark-ca-us.zip
US	California		Glendora Transportation Division	http://data.trilliumtransit.com/gtfs/glendora-ca-us/glendora-ca-us.zip
US	California	Sacremento	South County Transit Link	http://data.trilliumtransit.com/gtfs/southcountytransitlink-ca-us/southcountytransitlink-ca-us.zip

Impact of the spec addition in the validator (first Pull Request, new files)

On the data available from the Mobility Database, here is the impact we see of this addition in the validator. More details about this analysis can be found in this Google colab.

- What new errors appear in the data, and for how many sources?
    - `duplicate_key` : 12 sources
    - `fare_transfer_rule_missing_transfer_count` : 8 sources
    - `fare_transfer_rule_with_forbidden_transfer_count` : 5 sources
    - `foreign_key_violation` : 18 sources
    - `invalid_currency_amount` : 66 sources
    - `missing_required_column` : 1 sources
    - `missing_required_field` : 24 sources

- How many sources in total are affected?
    - 73 sources: some of them contain more than one new error.

- **fare_transfer_rule_missing_transfer_count**

    - What are the sources affected?

        1. `us-california-norwalk-transit-system-nts-gtfs-102`
        2. `us-california-city-of-tracy-tracer-gtfs-877`
        3. `us-california-downeylink-gtfs-383`
        4. `us-california-lake-transit-authority-lta-gtfs-68`
        5. `us-california-auburn-transit-gtfs-217`
        6. `us-california-ridgerunner-gtfs-311`
        7. `us-california-san-diego-international-airport-metropolitan-transit-system-mts-gtfs-13`
        8. `Us-california-corona-cruiser-gtfs-104`

These source Ids can then be matched with the Mobility Database to find the providers.

Country	State	Municipality	Provider	Direct download URL
US	California	Los Angeles	Norwalk Transit System (NTS)	http://data.trilliumtransit.com/gtfs/nts-ca-us/nts-ca-us.zip
US	California	Tracy	City of Tracy (TRACER)	http://data.trilliumtransit.com/gtfs/tracy-ca-us/tracy-ca-us.zip
US	California		DowneyLINK	http://data.trilliumtransit.com/gtfs/downey-ca-us/downey-ca-us.zip
US	California		Lake Transit Authority (LTA)	http://data.trilliumtransit.com/gtfs/laketransit-ca-us/laketransit-ca-us.zip
US	California	Auburn	Auburn Transit	http://data.trilliumtransit.com/gtfs/auburntransit-ca-us/auburntransit-ca-us.zip
US	California	Ridgecrest	Ridgerunner	http://data.trilliumtransit.com/gtfs/cityofridgecrest-ca-us/cityofridgecrest-ca-us.zip
US	California	San Diego	San Diego International Airport, Metropolitan Transit System (MTS)	https://www.sdmts.com/google_transit_files/google_transit.zip
US	California	Corona	Corona Cruiser	http://data.trilliumtransit.com/gtfs/corona-ca-us/corona-ca-us.zip

    - What seems to be the reason?
        - `transfer_count` is missing in the whole file. Other fields that aren't part of the adopted spec are included & have values: `is_symetrical`, `spanning_limit`, and other fields related to price information

- **fare_transfer_rule_with_forbidden_transfer_count**

- What are the sources affected?

1. `us-california-gtrans-gtfs-616`
2. `us-california-ventura-county-transportation-commission-gtfs-31`
3. `us-california-nevada-county-connects-gold-country-stage-gtfs-84`
4. `us-california-san-benito-county-express-sbce-gtfs-58`
5. `us-california-redwood-coast-transit-rct-gtfs-20`

These source Ids can then be matched with the Mobility Database to find the providers.

Country	State	Municipality	Provider	Direct download URL
US	California	Gardena	GTrans	http://data.trilliumtransit.com/gtfs/gtrans-ca-us/gtrans-ca-us.zip
US	California	Santa Barbara	Ventura County Transportation Commission	http://data.trilliumtransit.com/gtfs/vctc-ca-us/vctc-ca-us.zip
US	California	Nevada County	Nevada County Connects (Gold Country Stage)	http://data.trilliumtransit.com/gtfs/goldcountrystage-ca-us/goldcountrystage-ca-us.zip
US	California	Hollister	San Benito County Express (SBCE)	http://data.trilliumtransit.com/gtfs/sanbenitocounty-ca-us/sanbenitocounty-ca-us.zip
US	California	Crescent City	Redwood Coast Transit (RCT)	http://data.trilliumtransit.com/gtfs/delnorte-ca-us/delnorte-ca-us.zip

- What seems to be the reason?
  - The spec mentions: *Forbidden if fare_transfer_rules.from_leg_group_id does not equal fare_transfer_rules.to_leg_group_id.*
  - `transfer_count` =1 for every line, including cases where `from_leg_group_id` and `to_leg_group_id` are equal, and cases where they aren't

- **Invalid_currency_amount**
  - What seems to be the reason?
    - Most currency amounts do not specify the number of decimal places to match the ISO 4217 spec.

- **Missing_required_field**
  - What seems to be the reason?
    - `fare_product_id` is missing

        - `fare_transfer_type` & `fare_product_id` are missing

- **Missing_required_column**
    - What is the column?
        - Unknown, this notice wasn't there when I ran the validator on the feed that was affected (`us-california-unitrans-gtfs-82`)

Impact of the spec addition in the validator (second Pull Request, changes to Fares v1 files)

- PAUSED, this spec addition is being re-considered (see the issue on google/transit) -

Overview of what has changed in the spec

Additional files

- Fare_products.txt (optional)
- Fare_leg_rules.txt (optional)
- Fare_transfer_rules.txt (optional)
- Areas.txt (optional)
- Stop_areas.txt (optional)

Changes to existing files

- routes.txt: additional *network_id* field
- fare_rules.txt: the file becomes *conditionally required*

    Conditionally Required:

    Required if fare_attributes.txt is defined.

    Forbidden otherwise.

New requirement concept

Conditionally forbidden: The field or file must not be included under conditions outlined in the field or file description.

New field type

Currency amount: A decimal value indicating a currency amount. The number of decimal places is specified by ISO 4217 for the accompanying Currency code. All financial calculations should be processed as decimal, currency, or another equivalent type suitable for financial calculations depending on the programming language used to consume data. Processing currency amounts as float is discouraged due to gains or losses of money during calculations.

Copy of new and/or modified specification files

Routes.txt

`network_id`	ID	Optional	Identifies a group of routes. Multiple rows in routes.txt may have the same `network_id`.

Fare_products.txt

File: Optional

Primary Key (`fare_product_id`)

To describe the different types of tickets or fares that can be purchased by riders.

Field Name	Type	Presence	Description
`fare_product_id`	ID	Required	Identifies a fare product.
`fare_product_name`	Text	Optional	The name of the fare product as displayed to riders.
`amount`	Currency amount	Required	The cost of the fare product. May be negative to represent transfer discounts. May be zero to represent a fare product that is free.
`currency`	Currency code	Required	The currency of the cost of the fare product.

Fare_leg_rules.txt

File: Optional

Primary Key (`network_id, from_area_id, to_area_id, fare_product_id`)

Fare rules for individual legs of travel.

Fares in `fare_leg_rules.txt` must be queried by filtering all the records in the file to find rules that match the leg to be traveled by the rider.

To process the cost of a leg:

1. The file `fare_leg_rules.txt` must be filtered by the fields that define the characteristics of travel, these fields are:
    - `fare_leg_rules.network_id`
    - `fare_leg_rules.from_area_id`
    - `fare_leg_rules.to_area_id`

2. If the leg exactly matches a record in `fare_leg_rules.txt` based on the characteristics of travel, that record must be processed to determine the cost of the leg.
3. If no exact matches are found, then empty entries in `fare_leg_rules.network_id`, `fare_leg_rules.from_area_id`, and `fare_leg_rules.to_area_id` must be checked to process the cost of the leg:
    - An empty entry in `fare_leg_rules.network_id` corresponds to all networks defined in `routes.txt` excluding the ones listed under `fare_leg_rules.network_id`
    - An empty entry in `fare_leg_rules.from_area_id` corresponds to all areas defined in `areas.area_id` excluding the ones listed under `fare_leg_rules.from_area_id`
    - An empty entry in `fare_leg_rules.to_area_id` corresponds to all areas defined in `areas.area_id` excluding the ones listed under `fare_leg_rules.to_area_id`

4. If the leg does not match any of the rules described above, then the fare is unknown.

Field Name	Type	Presence	Description

`leg_group_id`	ID	Optional	Identifies a group of entries in fare_leg_rules.txt. Used to describe fare transfer rules between `fare_transfer_rules.from_leg_group_id` and `fare_transfer_rules.to_leg_group_id`. Multiple entries in fare_leg_rules.txt may belong to the same `fare_leg_rules.leg_group_id`. The same entry in fare_leg_rules.txt (not including `fare_leg_rules.leg_group_id`) must not belong to multiple `fare_leg_rules.leg_group_id`.
`network_id`	Foreign ID referencing `routes.network_id`	Optional	Identifies a route network that applies for the fare leg rule. If there are no matching `fare_leg_rules.network_id` values to the `network_id` being filtered, empty `fare_leg_rules.network_id` will be matched by default. An empty entry in `fare_leg_rules.network_id` corresponds to all networks defined in `routes.txt` excluding the ones listed under `fare_leg_rules.network_id`
`from_area_id`	Foreign ID referencing `areas.area_id`	Optional	Identifies a departure area. If there are no matching `fare_leg_rules.from_area_id` values to the `area_id`

			being filtered, empty `fare_leg_rules.from_area_id` will be matched by default. An empty entry in `fare_leg_rules.from_area_id` corresponds to all areas defined in `areas.area_id` excluding the ones listed under `fare_leg_rules.from_area_id`
`to_area_id`	Foreign ID referencing `areas.area_id`	Optional	Identifies an arrival area. If there are no matching `fare_leg_rules.to_area_id` values to the `area_id` being filtered, empty `fare_leg_rules.to_area_id` will be matched by default. An empty entry in `fare_leg_rules.to_area_id` corresponds to all areas defined in `areas.area_id` excluding the ones listed under `fare_leg_rules.to_area_id`
`fare_product_id`	Foreign ID referencing `fare_products.fare_product_id`	Required	The fare product required to travel the leg.

Fare_transfer_rules.txt

File: Optional

Primary Key (`from_leg_group_id, to_leg_group_id, fare_product_id, transfer_count, duration_limit`)

Fare rules for transfers between legs of travel defined in `fare_leg_rules.txt`.

To process the cost of a multi-leg journey:

1. The applicable fare leg groups defined in `fare_leg_rules.txt` should be determined for all individual legs of travel based on the rider's journey.
2. The file `fare_transfer_rules.txt` must be filtered by the fields that define the characteristics of the transfer, these fields are:
    - `fare_transfer_rules.from_leg_group_id`
    - `fare_transfer_rules.to_leg_group_id`
3. If the transfer exactly matches a record in `fare_transfer_rules.txt` based on the characteristics of the transfer, then that record must be processed to determine the transfer cost.
4. If no exact matches are found, then empty entries in `from_leg_group_id` or in `to_leg_group_id` must be checked to process the transfer cost:
    - An empty entry in `fare_transfer_rules.from_leg_group_id` corresponds to all leg groups defined under `fare_leg_rules.leg_group_id` excluding the ones listed under `fare_transfer_rules.from_leg_group_id`
    - An empty entry in `fare_transfer_rules.to_leg_group_id` corresponds to all leg groups defined under `fare_leg_rules.leg_group_id` excluding the ones listed under `fare_transfer_rules.to_leg_group_id`
5. If the transfer does not match any of the rules described above, then there is no transfer arrangement and the legs are considered separate.

Field Name	Type	Presence	Description
`from_leg_group_id`	Foreign ID referencing `fare_leg_rules.leg_group_id`	Optional	Identifies a group of pre-transfer fare leg rules. If there are no matching `fare_transfer_rules.from_leg_group_id` values to the `leg_group_id` being filtered, empty `fare_transfer_rules.from_leg_group_id` will be matched by default. An empty entry in `fare_transfer_rules.from_leg_group_id` corresponds to all leg groups defined under `fare_leg_rules.leg_group_id` excluding the ones listed under `fare_transfer_rules.from_leg_group_id`

`to_leg_group_id`	Foreign ID referencing `fare_leg_rules.leg_group_id`	Optional	Identifies a group of post-transfer fare leg rules. If there are no matching `fare_transfer_rules.to_leg_group_id` values to the `leg_group_id` being filtered, empty `fare_transfer_rules.to_leg_group_id` will be matched by default. An empty entry in `fare_transfer_rules.to_leg_group_id` corresponds to all leg groups defined under `fare_leg_rules.leg_group_id` excluding the ones listed under `fare_transfer_rules.to_leg_group_id`
`transfer_count`	Non-zero integer	Conditionally Forbidden	Defines how many consecutive transfers the transfer rule may be applied to. Valid options are: `-1` - No limit. `1` or more - Defines how many transfers the transfer rule may span. If a sub-journey matches multiple records with different `transfer_count`s, then the rule with the minimum `transfer_count` that is greater than or equal to the current transfer count of the sub-journey is to be selected. Conditionally Forbidden: - Forbidden if `fare_transfer_rules.from_leg_group_id` does not equal `fare_transfer_rules.to_leg_group_id`.

			- Required if `fare_transfer_rules.from_leg_group_id` equals `fare_transfer_rules.to_leg_group_id`.
`duration_limit`	Positive integer	Optional	Defines the duration limit of the transfer. Must be expressed in integer increments of seconds. If there is no duration limit, `fare_transfer_rules.duration_limit` must be empty.
`duration_limit_type`	Enum	Conditionally Required	Defines the relative start and end of `fare_transfer_rules.duration_limit`. Valid options are: `0` - Between the departure fare validation of the current leg and the arrival fare validation of the next leg. `1` - Between the departure fare validation of the current leg and the departure fare validation of the next leg. `2` - Between the arrival fare validation of the current leg and the departure fare validation of the next leg. `3` - Between the arrival fare validation of the current leg and the arrival fare validation of the next leg. Conditionally Required: - Required if `fare_transfer_rules.duration_limit` is defined. - Forbidden if `fare_transfer_rules.duration_limit` is empty.

`fare_transfer_type`	Enum	Required	Indicates the cost processing method of transferring between legs in a journey: —image– Valid options are: `0` - From-leg `fare_leg_rules.fare_product_id` plus `fare_transfer_rules.fare_product_id`; A + AB. `1` - From-leg `fare_leg_rules.fare_product_id` plus `fare_transfer_rules.fare_product_id` plus to-leg `fare_leg_rules.fare_product_id`; A + AB + B. `2` - `fare_transfer_rules.fare_product_id`; AB. Cost processing interactions between multiple transfers in a journey: —image– Where S indicates the total processed cost of the preceding leg(s) and transfer(s).
`fare_product_id`	Foreign ID referencing `fare_products.fare_product_id`	Optional	The fare product required to transfer between two fare legs. If empty, the cost of the transfer rule is 0.

Areas.txt

File: Optional

Primary key (`area_id`)

Defines area identifiers.

Field Name	Type	Presence	Description

area_id	Unique ID	Required	Identifies an area. Must be unique in areas.txt.
area_name	Text	Optional	The name of the area as displayed to the rider.

Stop_areas.txt

File: Optional

Primary key (*)

Assigns stops from stops.txt to areas.

Field Name	Type	Presence	Description
area_id	Foreign ID referencing areas.area_id	Required	Identifies a an area to which one or multiple stop_ids belong. The same stop_id may be defined in many area_ids.
stop_id	Foreign ID referencing stops.stop_id	Required	Identifies a stop. If a station (i.e. a stop with stops.location_type=1) is defined in this field, it is assumed that all of its platforms (i.e. all stops with stops.location_type=0 that have this station defined as stops.parent_station) are part of the same area. This behavior can be overridden by assigning platforms to other areas.