

GTFS Realtime Reference

A GTFS Realtime feed lets transit agencies provide consumers with realtime information about disruptions to their service (stations closed, lines not operating, important delays, etc.) location of their vehicles, and expected arrival times.

Version 2.0 of the feed specification is discussed and documented on this site. Valid versions are "2.0", "1.0".

Term Definitions

Required

In GTFS-realtime v2.0 and higher, the *Required* column describes what fields must be provided by a producer in order for the transit data to be valid and make sense to a consuming application.

The following values are used in the *Required* field:

- **Required:** This field must be provided by a GTFS-realtime feed producer.
- **Conditionally required:** This field is required under certain conditions, which are outlined in the field *Description*. Outside of these conditions, the field is optional.
- **Conditionally forbidden:** This field is forbidden under certain conditions, which are outlined in the field *Description*. Outside of these conditions, the field is optional.
- **Optional:** This field is optional and is not required to be implemented by producers. However, if the data is available in the underlying automatic vehicle location systems (e.g., `VehiclePosition` `timestamp`) it is recommended that producers provide these optional fields when possible.

Note that semantic requirements were not defined in GTFS-realtime version 1.0, and therefore feeds with `gtfs_realtime_version` of `1` may not meet these requirements (see [the proposal for semantic requirements](#) for details).

Cardinality

Cardinality represents the number of elements that may be provided for a particular field, with the following values:

- **One** - A single one element may be provided for this field. This maps to the [Protocol Buffer required and optional cardinalities](#).
- **Many** - Many elements (0, 1, or more) may be provided for this field. This maps to the [Protocol Buffer repeated cardinality](#).

Always reference the *Required* and *Description* fields to see when a field is required, conditionally required, or optional. Please reference [gtfs-realtime.proto](#) for Protocol Buffer cardinality.

Protocol Buffer data types

The following protocol buffer data types are used to describe feed elements:

- **message:** Complex type
- **enum:** List of fixed values

Experimental fields

Fields labeled as **experimental** are subject to change and not yet formally adopted into the specification. An **experimental** field may be formally adopted in the future.

Element Index

- [FeedMessage](#)

- [FeedHeader](#)
 - [Incrementality](#)
- [FeedEntity](#)
 - [TripUpdate](#)
 - [TripDescriptor](#)
 - [ScheduleRelationship](#)
 - [VehicleDescriptor](#)
 - [WheelchairAccessible](#)
 - [StopTimeUpdate](#)
 - [StopTimeEvent](#)
 - [ScheduleRelationship](#)
 - [StopTimeProperties](#)
 - [TripProperties](#)
 - [VehiclePosition](#)
 - [TripDescriptor](#)
 - [ScheduleRelationship](#)
 - [ModifiedTripSelector](#)
 - [VehicleDescriptor](#)
 - [WheelchairAccessible](#)
 - [Position](#)
 - [VehicleStopStatus](#)
 - [CongestionLevel](#)
 - [OccupancyStatus](#)
 - [CarriageDetails](#)
 - [Alert](#)
 - [TimeRange](#)
 - [EntitySelector](#)
 - [TripDescriptor](#)
 - [ScheduleRelationship](#)
 - [Cause](#)
 - [Effect](#)
 - [TranslatedString](#)
 - [Translation](#)
 - [SeverityLevel](#)
 - [Shape](#)
 - [Stop](#)
 - [WheelchairBoarding](#)
 - [TripModifications](#)
 - [Modification](#)
 - [ReplacementStop](#)

Elements

message FeedMessage

The contents of a feed message. Each message in the stream is obtained as a response to an appropriate HTTP GET request. A realtime feed is always defined with relation to an existing GTFS feed. All the entity ids are resolved with respect to the GTFS feed.

Fields

Field Name	Type	Required	Cardinality	Description
------------	------	----------	-------------	-------------

Field Name	Type	Required	Cardinality	Description
header	FeedHeader	Required	One	Metadata about this feed and feed message.
entity	FeedEntity	Conditionally required	Many	Contents of the feed. If there is real-time information available for the transit system, this field must be provided. If this field is empty, consumers should assume there is no real-time information available for the system.

message FeedHeader

Metadata about a feed, included in feed messages.

Fields

Field Name	Type	Required	Cardinality	Description
gtfs_realtime_version	string	Required	One	Version of the feed specification. The current version is 2.0.
incrementality	Incrementality	Required	One	
timestamp	uint64	Required	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). To avoid time skew between systems producing and consuming realtime information it is strongly advised to derive timestamp from a time server. It is completely acceptable to use Stratum 3 or even lower strata servers since time differences up to a couple of seconds are tolerable.
feed_version	string	Optional	One	String that matches the <code>feed_info.feed_version</code> from the GTFS feed that the realtime data is based on. Consumers can use this to identify which GTFS feed is currently active or when a new one is available to download.

enum Incrementality

Determines whether the current fetch is incremental.

- FULL_DATASET**: this feed update will overwrite all preceding realtime information for the feed. Thus this update is expected to provide a full snapshot of all known realtime information.
- DIFFERENTIAL**: currently, this mode is **unsupported** and behavior is **unspecified** for feeds that use this mode. There are discussions on the [GTFS Realtime mailing list](#) around fully specifying the behavior of DIFFERENTIAL mode and the documentation will be updated when those discussions are finalized.

Values

Value
FULL_DATASET
DIFFERENTIAL

message FeedEntity

A definition (or update) of an entity in the transit feed. If the entity is not being deleted, exactly one of 'trip_update', 'vehicle', 'alert', 'shape', 'stop' or 'trip_modification' fields should be populated.

Fields

Field Name	Type	Required	Cardinality	Description
id	string	Required	One	Feed-unique identifier for this entity. The ids are used only to provide incrementality support. The actual entities referenced by the feed must be specified by explicit selectors (see EntitySelector below for more info).
is_deleted	bool	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.
trip_update	TripUpdate	Conditionally required	One	Data about the realtime departure delays of a trip. At least one of the fields trip_update, vehicle, alert, or shape must be provided - all these fields cannot be empty.
vehicle	VehiclePosition	Conditionally required	One	Data about the realtime position of a vehicle. At least one of the fields trip_update, vehicle, alert, or shape must be provided - all these fields cannot be empty.
alert	Alert	Conditionally required	One	Data about the realtime alert. At least one of the fields trip_update, vehicle, alert, or shape must be provided - all these fields cannot be empty.
shape	Shape	Conditionally required	One	Data about the realtime added shapes, such as for a detour. At least one of the fields trip_update, vehicle, alert, or shape must be provided - all these fields cannot be empty. Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.
stop	Stop	Conditionally required	One	A new stop added to the feed dynamically. Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.
trip_modifications	[TripModifications) (#message-tripmodifications)	Conditionally required	One	List of trips affected by a particular modifications, such as a detour. Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.

message TripUpdate

Realtime update on the progress of a vehicle along a trip. Please also refer to the general discussion of the [trip updates entities](#).

Depending on the value of ScheduleRelationship, a TripUpdate can specify:

- A trip that proceeds along the schedule.
- A trip that proceeds along a route but has no fixed schedule.
- A trip that has been added or removed with regard to schedule.
- A trip that replaces an existing trip in static GTFS.
- A new trip that is a copy of an existing trip in static GTFS. It will run at the service date and time specified in TripProperties.

The updates can be for future, predicted arrival/departure events, or for past events that already occurred. In most cases information about past events is a measured value thus its uncertainty value is recommended to be 0. Although there could be cases when this does not hold so it is allowed to have uncertainty value different from 0 for past events. If an update's uncertainty is not 0, either the update is an approximate prediction for a trip that has not completed or the measurement is not precise or the update was a prediction for the past that has not been verified after the event occurred.

If a vehicle is serving multiple trips within the same block (for more information about trips and blocks, please refer to [GTFS trips.txt](#)):

- the feed should include a TripUpdate for the trip currently being served by the vehicle. Producers are encouraged to include TripUpdates for one or more trips after the current trip in this vehicle's block if the producer is confident in the quality of the predictions for these future trip(s). Including multiple TripUpdates for the same vehicle avoids prediction "pop-in" for riders as the

vehicle transitions from one trip to another and also gives riders advance notice of delays that impact downstream trips (e.g., when the known delay exceeds planned layover times between trips).

- the respective TripUpdate entities are not required to be added to the feed in the same order that they are scheduled in the block. For example, if there are trips with `trip_ids` 1, 2, and 3 that all belong to one block, and the vehicle travels trip 1, then trip 2, and then trip 3, the `trip_update` entities may appear in any order - for example, adding trip 2, then trip 1, and then trip 3 is allowed.

Note that the update can describe a trip that has already completed. To this end, it is enough to provide an update for the last stop of the trip. If the time of arrival at the last stop is in the past, the client will conclude that the whole trip is in the past (it is possible, although inconsequential, to also provide updates for preceding stops). This option is most relevant for a trip that has completed ahead of schedule, but according to the schedule, the trip is still proceeding at the current time. Removing the updates for this trip could make the client assume that the trip is still proceeding. Note that the feed provider is allowed, but not required, to purge past updates - this is one case where this would be practically useful.

Fields

Field Name	Type	Required	Cardinality	Description
trip	TripDescriptor	Required	One	The Trip that this message applies to. There can be at most one TripUpdate entity for each actual trip instance. If there is none, that means there is no prediction information available. It does <i>not</i> mean that the trip is progressing according to schedule.
vehicle	VehicleDescriptor	Optional	One	Additional information on the vehicle that is serving this trip.
stop_time_update	StopTimeUpdate	Conditionally required	Many	Updates to StopTimes for the trip (both future, i.e., predictions, and in some cases, past ones, i.e., those that already happened). The updates must be sorted by stop_sequence, and apply for all the following stops of the trip up to the next specified stop_time_update. If trip.schedule_relationship is SCHEDULED or UNSCHEDULED, at least one stop_time_update must be provided for the trip. If trip.schedule_relationship is NEW or REPLACEMENT, stop_time_updates must be provided for all stops in the new or replacement trip, including stops with times in the past, and the stop times in the static GTFS are not used. If the trip is canceled or deleted, no stop_time_updates need to be provided. If stop_time_updates are provided for a canceled or deleted trip then the trip.schedule_relationship takes precedence over any stop_time_updates and their associated schedule_relationship. If the trip is duplicated, stop_time_updates may be provided to indicate real-time information for the new trip.
timestamp	uint64	Optional	One	The most recent moment at which the vehicle's real-time progress was measured to estimate StopTimes in the future. When StopTimes in the past are provided, arrival/departure times may be earlier than this value. In POSIX time (i.e., the number of seconds since January 1st 1970 00:00:00 UTC).

Field Name	Type	Required	Cardinality	Description
delay	int32	Optional	One	<p>The current schedule deviation for the trip. Delay should only be specified when the prediction is given relative to some existing schedule in GTFS.</p> <p>Delay (in seconds) can be positive (meaning that the vehicle is late) or negative (meaning that the vehicle is ahead of schedule). Delay of 0 means that the vehicle is exactly on time.</p> <p>Delay information in StopTimeUpdates take precedent of trip-level delay information, such that trip-level delay is only propagated until the next stop along the trip with a StopTimeUpdate delay value specified.</p> <p>Feed providers are strongly encouraged to provide a TripUpdate.timestamp value indicating when the delay value was last updated, in order to evaluate the freshness of the data.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
trip_properties	TripProperties	Optional	One	<p>Provides the updated properties for the trip.</p> <p>Caution: this message is still experimental, and subject to change. It may be formally adopted in the future.</p>

message StopTimeEvent

Timing information for a single predicted event (either arrival or departure). Timing consists of delay and/or estimated time, and uncertainty. A scheduled time can also be added for NEW, REPLACEMENT, or DUPLICATED trips.

- delay should be used when the prediction is given relative to some existing schedule in GTFS.
- time should be given whether there is a predicted schedule or not, and must be given for new or replacement trips. If both time and delay are specified, time will take precedence (although normally, time, if given for a scheduled trip, should be equal to scheduled time in GTFS + delay).
- scheduled time may be given if the trip is a new, replacement or duplicated trip.

Uncertainty applies equally to both time and delay. The uncertainty roughly specifies the expected error in true delay (but note, we don't yet define its precise statistical meaning). It's possible for the uncertainty to be 0, for example for trains that are driven under computer timing control.

Fields

Field Name	Type	Required	Cardinality	Description
delay	int32	Conditionally required	One	<p>Delay (in seconds) can be positive (meaning that the vehicle is late) or negative (meaning that the vehicle is ahead of schedule). Delay of 0 means that the vehicle is exactly on time.</p> <p>Forbidden if StopTimeUpdate.schedule_relationship is NO_DATA. Required otherwise if time is not given.</p>
time	int64	Conditionally required	One	<p>Estimated or actual event as absolute time. In POSIX time (i.e., number of seconds since January 1st 1970 00:00:00 UTC).</p> <p>Forbidden if StopTimeUpdate.schedule_relationship is NO_DATA. Required otherwise if delay is not given.</p>
scheduled_time	int64	Conditionally forbidden	One	<p>Scheduled time. In POSIX time (i.e., number of seconds since January 1st 1970 00:00:00 UTC).</p> <p>Optional if TripUpdate.schedule_relationship is NEW, REPLACEMENT or DUPLICATED, forbidden otherwise.</p>
uncertainty	int32	Optional	One	<p>If uncertainty is omitted, it is interpreted as unknown. To specify a completely certain prediction, set its uncertainty to 0.</p> <p>Forbidden if StopTimeUpdate.schedule_relationship is NO_DATA.</p>

message StopTimeUpdate

Realtime update for arrival and/or departure events for a given stop on a trip. Please also refer to the general discussion of stop time updates in the [TripDescriptor](#) and [trip updates entities](#) documentation.

Updates can be supplied for both past and future events. The producer is allowed, although not required, to drop past events, unless if `TripUpdate.schedule_relationship` is NEW or REPLACEMENT, in such case past stops must not be dropped as they define the trip the vehicle is on, until the whole trip has been finished.

The update is linked to a specific stop either through `stop_sequence` or `stop_id`, so one of these fields must necessarily be set. If the same `stop_id` is visited more than once in a trip, then `stop_sequence` should be provided in all StopTimeUpdates for that `stop_id` on that trip.

In new or replacement trips, updates are used to specify the stops visited by the trip without referring to an existing trip in the GTFS Static. In such trips, `stop_id`, `stop_sequence`, `departure` and `arrival` must all be set.

Fields

Field Name	Type	Required	Cardinality	Description
stop_sequence	uint32	Conditionally required	One	Must be the same as in stop_times.txt in the corresponding GTFS feed. Either <code>stop_sequence</code> or <code>stop_id</code> must be provided within a StopTimeUpdate - both fields cannot be empty. <code>stop_sequence</code> is required for trips that visit the same <code>stop_id</code> more than once (e.g., a loop) to disambiguate which stop the prediction is for. If <code>StopTimeProperties.assigned_stop_id</code> is populated, then <code>stop_sequence</code> must be populated. Required if <code>TripUpdate.schedule_relationship</code> is NEW or REPLACEMENT, and the value must be increasing along the trip.
stop_id	string	Conditionally required	One	Must be the same as in stops.txt in the corresponding GTFS feed. Either <code>stop_sequence</code> or <code>stop_id</code> must be provided within a StopTimeUpdate - both fields cannot be empty. If <code>StopTimeProperties.assigned_stop_id</code> is populated, it is preferred to omit <code>stop_id</code> and use only <code>stop_sequence</code> . If <code>StopTimeProperties.assigned_stop_id</code> and <code>stop_id</code> are populated, <code>stop_id</code> must match <code>assigned stop id</code> . Required if <code>TripUpdate.schedule_relationship</code> is NEW or REPLACEMENT.
arrival	StopTimeEvent	Conditionally required	One	If <code>schedule_relationship</code> is empty or SCHEDULED, either arrival or departure must be provided within a StopTimeUpdate - both fields cannot be empty. arrival and departure may both be empty when <code>schedule_relationship</code> is SKIPPED. Required if <code>TripUpdate.schedule_relationship</code> is NEW or REPLACEMENT.

Field Name	Type	Required	Cardinality	Description
departure	StopTimeEvent	Conditionally required	One	If schedule_relationship is empty or SCHEDULED, either arrival or departure must be provided within a StopTimeUpdate - both fields cannot be empty. arrival and departure may both be empty when schedule_relationship is SKIPPED. Required if <code>TripUpdate.schedule_relationship</code> is NEW or REPLACEMENT.
departure_occupancy_status	OccupancyStatus	Optional	One	The predicted state of passenger occupancy for the vehicle immediately after departure from the given stop. If provided, stop_sequence must be provided. To provide departure_occupancy_status without providing any real-time arrival or departure predictions, populate this field and set StopTimeUpdate.schedule_relationship = NO_DATA. Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.
schedule_relationship	ScheduleRelationship	Optional	One	The default relationship is SCHEDULED.
stop_time_properties	StopTimeProperties	Optional	One	Realtime updates for certain properties defined within GTFS stop_times.txt Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.

enum ScheduleRelationship

The relation between this StopTime and the static schedule.

Values

Value	Comment
SCHEDULED	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 behavior. At least one of arrival and departure must be provided. Frequency-based trips (GTFS frequencies.txt with exact_times = 0) should not have a SCHEDULED value and should use UNSCHEDULED instead.
SKIPPED	The stop is skipped, i.e., the vehicle will not stop at this stop. Arrival and departure are optional. When set <code>SKIPPED</code> is not propagated to subsequent stops in the same trip (i.e., the vehicle will stop at subsequent stops in the trip unless those stops also have a <code>stop_time_update</code> with <code>schedule_relationship: SKIPPED</code>). Delay from a previous stop in the trip <i>does</i> propagate over the <code>SKIPPED</code> stop. In other words, if a <code>stop_time_update</code> with an <code>arrival</code> or <code>departure</code> prediction is not set for a stop after the <code>SKIPPED</code> stop, the prediction upstream of the <code>SKIPPED</code> stop will be propagated to the stop after the <code>SKIPPED</code> stop and subsequent stops in the trip until a <code>stop_time_update</code> for a subsequent stop is provided.
NO_DATA	No real-time data is given for this stop. It indicates that there is no realtime timing 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 realtime timing information. When NO_DATA is set, arrival or departure must not be supplied, unless <code>TripDescriptor.schedule_relationship</code> is <code>NEW</code> or <code>REPLACEMENT</code> , in such case only the scheduled time, but not predictions, must be supplied. When <code>TripDescriptor.schedule_relationship</code> is <code>NEW</code> or <code>REPLACEMENT</code> , <code>arrival</code> and <code>departure</code> must still be given with scheduled times, as the StopTimeUpdate defines the stop list of the trip. In this case it indicates that the schedule is unrelated to the static GTFS, but real-time prediction is not available yet.

Value	Comment
UNSCHEDULED	<p>The vehicle is operating a frequency-based trip (GTFS frequencies.txt with exact_times = 0). This value should not be used for trips that are not defined in GTFS frequencies.txt, or trips in GTFS frequencies.txt with exact_times = 1. Trips containing <code>stop_time_updates</code> with <code>schedule_relationship: UNSCHEDULED</code> must also set the TripDescriptor <code>schedule_relationship: UNSCHEDULED</code></p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>

message StopTimeProperties

Realtime update for certain properties defined within GTFS stop_times.txt.

Caution: this message is still **experimental**, and subject to change. It may be formally adopted in the future.

Fields

Field Name	Type	Required	Cardinality	Description
assigned_stop_id	string	Optional	One	<p>Supports real-time stop assignments. Refers to a <code>stop_id</code> defined in the GTFS <code>stops.txt</code>.</p> <p>The new <code>assigned_stop_id</code> should not result in a significantly different trip experience for the end user than the <code>stop_id</code> defined in GTFS <code>stop_times.txt</code>. In other words, the end user should not view this new <code>stop_id</code> as an "unusual change" if the new stop was presented within an app without any additional context. For example, this field is intended to be used for platform assignments by using a <code>stop_id</code> that belongs to the same station as the stop originally defined in GTFS <code>stop_times.txt</code>.</p> <p>To assign a stop without providing any real-time arrival or departure predictions, populate this field and set <code>StopTimeUpdate.schedule_relationship = NO_DATA</code>.</p> <p>If this field is populated, <code>StopTimeUpdate.stop_sequence</code> must be populated and <code>StopTimeUpdate.stop_id</code> should not be populated. Stop assignments should be reflected in other GTFS-realtime fields as well (e.g., <code>VehiclePosition.stop_id</code>).</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
stop_headsign	string	Optional	One	<p>The updated headsign of the vehicle at the stop.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
drop_off_type	DropOffPickupType	Optional	One	<p>The updated drop off of the vehicle at the stop.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
pickup_type	DropOffPickupType	Optional	One	<p>The updated pickup of the vehicle at the stop.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>

enum DropOffPickupType

Values

Value	Comment
-------	---------

Value	Comment
REGULAR	Regularly scheduled pickup/dropoff.
NONE	No pickup/dropoff available.
PHONE_AGENCY	Must phone agency to arrange pickup/dropoff.
COORDINATE_WITH_DRIVER	Must coordinate with driver to arrange pickup/dropoff.

message TripProperties

Defines updated properties of the trip

Caution: this message is still **experimental**, and subject to change. It may be formally adopted in the future.

.

Fields

Field Name	Type	Required	Cardinality	Description
trip_id	string	Conditionally required	One	<p>Defines the identifier of a new trip that is a duplicate of an existing trip defined in (CSV) GTFS trips.txt but will start at a different service date and/or time (defined using <code>TripProperties.start_date</code> and <code>TripProperties.start_time</code>). See definition of <code>trips.trip_id</code> in (CSV) GTFS. Its value must be different than the ones used in the (CSV) GTFS. This field is required if <code>schedule_relationship</code> is <code>DUPLICATED</code> , otherwise this field must not be populated and will be ignored by consumers.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
start_date	string	Conditionally required	One	<p>Service date on which the duplicated trip will be run. Must be provided in YYYYMMDD format. This field is required if <code>schedule_relationship</code> is <code>DUPLICATED</code> , otherwise this field must not be populated and will be ignored by consumers.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
start_time	string	Conditionally required	One	<p>Defines the departure start time of the trip when it's duplicated. See definition of <code>stop_times.departure_time</code> in (CSV) GTFS. Scheduled arrival and departure times for the duplicated trip are calculated based on the offset between the original trip <code>departure_time</code> and this field. For example, if a GTFS trip has stop A with a <code>departure_time</code> of <code>10:00:00</code> and stop B with <code>departure_time</code> of <code>10:01:00</code> , and this field is populated with the value of <code>10:30:00</code> , stop B on the duplicated trip will have a scheduled <code>departure_time</code> of <code>10:31:00</code> . Real-time prediction <code>delay</code> values are applied to this calculated schedule time to determine the predicted time. For example, if a departure <code>delay</code> of <code>30</code> is provided for stop B, then the predicted departure time is <code>10:31:30</code> . Real-time prediction <code>time</code> values do not have any offset applied to them and indicate the predicted time as provided. For example, if a departure <code>time</code> representing 10:31:30 is provided for stop B, then the predicted departure time is <code>10:31:30</code> . This field is required if <code>schedule_relationship</code> is <code>DUPLICATED</code> , otherwise this field must not be populated and will be ignored by consumers.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>

Field Name	Type	Required	Cardinality	Description
shape_id	string	Optional	One	<p>Specifies the identifier of the shape of the vehicle travel path when the trip shape differs from the shape specified in (CSV) GTFS or to specify it in real-time when it's not provided by (CSV) GTFS, such as a vehicle that takes differing paths based on rider demand. See definition of <code>trips.shape_id</code> in (CSV) GTFS.</p> <p>If a shape is neither defined in (CSV) GTFS nor in real-time, the shape is considered unknown. This field can refer to a shape defined in the (CSV) GTFS in <code>shapes.txt</code> or a <code>Shape</code> in the same (protobuf) real-time feed. The order of stops (stop sequences) for this trip must remain the same as (CSV) GTFS. If it refers to a <code>Shape</code> entity in the same real-time feed, the value of this field should be the one of the <code>shape_id</code> inside the entity, and not the <code>id</code> of <code>FeedEntity</code>.</p> <p>Stops that are a part of the original trip but will no longer be made, such as when a detour occurs, should be marked as <code>schedule_relationship=SKIPPED</code> or more details can be provided via a <code>TripModifications</code> message.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
trip_headsign	string	Optional	One	<p>Specifies the headsign for this trip when it differs from the original.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
trip_short_name	string	Optional	One	<p>Specifies the name for this trip when it differs from the original.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>

message VehiclePosition

Realtime positioning information for a given vehicle.

Fields

Field Name	Type	Required	Cardinality	Description
trip	TripDescriptor	Optional	One	The Trip that this vehicle is serving. Can be empty or partial if the vehicle can not be identified with a given trip instance.
vehicle	VehicleDescriptor	Optional	One	Additional information on the vehicle that is serving this trip. Each entry should have a unique vehicle id.
position	Position	Optional	One	Current position of this vehicle.
current_stop_sequence	uint32	Optional	One	The stop sequence index of the current stop. The meaning of <code>current_stop_sequence</code> (i.e., the stop that it refers to) is determined by <code>current_status</code> . If <code>current_status</code> is missing <code>IN_TRANSIT_TO</code> is assumed.
stop_id	string	Optional	One	Identifies the current stop. The value must be the same as in <code>stops.txt</code> in the corresponding GTFS feed. If <code>StopTimeProperties.assigned_stop_id</code> is used to assign a <code>stop_id</code> , this field should also reflect the change in <code>stop_id</code> .
current_status	VehicleStopStatus	Optional	One	The exact status of the vehicle with respect to the current stop. Ignored if <code>current_stop_sequence</code> is missing.
timestamp	uint64	Optional	One	Moment at which the vehicle's position was measured. In POSIX time (i.e., number of seconds since January 1st 1970 00:00:00 UTC).
congestion_level	CongestionLevel	Optional	One	

Field Name	Type	Required	Cardinality	Description
occupancy_status	OccupancyStatus	Optional	One	<p>The state of passenger occupancy for the vehicle or carriage. If multi_carriage_details is populated with per-carriage OccupancyStatus, then this field should describe the entire vehicle with all carriages accepting passengers considered.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
occupancy_percentage	uint32	Optional	One	<p>A percentage value indicating the degree of passenger occupancy in the vehicle. The value 100 should represent the total maximum occupancy the vehicle was designed for, including both seating and standing capacity, and current operating regulations allow. The value may exceed 100 if there are more passengers than the maximum designed capacity. The precision of occupancy_percentage should be low enough that individual passengers cannot be tracked boarding or alighting the vehicle. If multi_carriage_details is populated with per-carriage occupancy_percentage, then this field should describe the entire vehicle with all carriages accepting passengers considered.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
multi_carriage_details	CarriageDetails	Optional	Many	<p>Details of the multiple carriages of this given vehicle. The first occurrence represents the first carriage of the vehicle, given the current direction of travel. The number of occurrences of the multi_carriage_details field represents the number of carriages of the vehicle. It also includes non boardable carriages, like engines, maintenance carriages, etc... as they provide valuable information to passengers about where to stand on a platform.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>

enum VehicleStopStatus

Values

Value	Comment
INCOMING_AT	The vehicle is just about to arrive at the stop (on a stop display, the vehicle symbol typically flashes).
STOPPED_AT	The vehicle is standing at the stop.
IN_TRANSIT_TO	The vehicle has departed the previous stop and is in transit.

enum CongestionLevel

Congestion level that is affecting this vehicle.

Values

Value
UNKNOWN_CONGESTION_LEVEL
RUNNING_SMOOTHLY
STOP_AND_GO

Value
CONGESTION
SEVERE_CONGESTION

enum OccupancyStatus

The state of passenger occupancy for the vehicle or carriage.

Individual producers may not publish all OccupancyStatus values. Therefore, consumers must not assume that the OccupancyStatus values follow a linear scale. Consumers should represent OccupancyStatus values as the state indicated and intended by the producer. Likewise, producers must use OccupancyStatus values that correspond to actual vehicle occupancy states.

For describing passenger occupancy levels on a linear scale, see [occupancy_percentage](#) .

Caution: this field is still **experimental**, and subject to change. It may be formally adopted in the future.

Values

Value	Comment
EMPTY	The vehicle is considered empty by most measures, and has few or no passengers onboard, but is still accepting passengers.
MANY_SEATS_AVAILABLE	The vehicle or carriage has a large number of seats available. The amount of free seats out of the total seats available to be considered large enough to fall into this category is determined at the discretion of the producer.
FEW_SEATS_AVAILABLE	The vehicle or carriage has a small number of seats available. The amount of free seats out of the total seats available to be considered small enough to fall into this category is determined at the discretion of the producer.
STANDING_ROOM_ONLY	The vehicle or carriage can currently accommodate only standing passengers.
CRUSHED_STANDING_ROOM_ONLY	The vehicle or carriage can currently accommodate only standing passengers and has limited space for them.
FULL	The vehicle is considered full by most measures, but may still be allowing passengers to board.
NOT_ACCEPTING_PASSENGERS	The vehicle or carriage is not accepting passengers. The vehicle or carriage usually accepts passengers for boarding.
NO_DATA_AVAILABLE	The vehicle or carriage doesn't have any occupancy data available at that time.
NOT_BOARDABLE	The vehicle or carriage is not boardable and never accepts passengers. Useful for special vehicles or carriages (engine, maintenance carriage, etc...).

message CarriageDetails

Carriage specific details, used for vehicles composed of several carriages.

Caution: this message is still **experimental**, and subject to change. It may be formally adopted in the future.

Fields

Field Name	Type	Required	Cardinality	Description
id	string	Optional	One	<p>Identification of the carriage. Should be unique per vehicle.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>

Field Name	Type	Required	Cardinality	Description
label	string	Optional	One	User visible label that may be shown to the passenger to help identify the carriage. Example: "7712", "Car ABC-32", etc... Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.
occupancy_status	OccupancyStatus	Optional	One	Occupancy status for this given carriage, in this vehicle. Default is set to <code>NO_DATA_AVAILABLE</code> . Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.
occupancy_percentage	int32	Optional	One	Occupancy percentage for this given carriage, in this vehicle. Follows the same rules as "VehiclePosition.occupancy_percentage". Use -1 in case data is not available for this given carriage. Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.
carriage_sequence	uint32	Required	One	Identifies the order of this carriage with respect to the other carriages in the vehicle's list of CarriageStatus. The first carriage in the direction of travel must have a value of 1. The second value corresponds to the second carriage in the direction of travel and must have a value of 2, and so forth. For example, the first carriage in the direction of travel has a value of 1. If the second carriage in the direction of travel has a value of 3, consumers will discard data for all carriages (i.e., the multi_carriage_details field). Carriages without data must be represented with a valid carriage_sequence number and the fields without data should be omitted (alternately, those fields could also be included and set to the "no data" values). Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.

message Alert

An alert, indicating some sort of incident in the public transit network.

Fields

Field Name	Type	Required	Cardinality	Description
active_period	TimeRange	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	EntitySelector	Required	Many	Entities whose users we should notify of this alert. At least one informed_entity must be provided.
cause	Cause	Conditionally Required	One	If cause_detail is included, then Cause must also be included.
cause_detail	TranslatedString	Optional	One	Description of the cause of the alert that allows for agency-specific language; more specific than the Cause. If cause_detail is included, then Cause must also be included. Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.
effect	Effect	Conditionally Required	One	If effect_detail is included, then Effect must also be included.

Field Name	Type	Required	Cardinality	Description
effect_detail	TranslatedString	Optional	One	Description of the effect of the alert that allows for agency-specific language; more specific than the Effect. If effect_detail is included, then Effect must also be included. Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.
url	TranslatedString	Optional	One	The URL which provides additional information about the alert.
header_text	TranslatedString	Required	One	Header for the alert. This plain-text string will be highlighted, for example in boldface.
description_text	TranslatedString	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.
tts_header_text	TranslatedString	Optional	One	Text containing the alert's header to be used for text-to-speech implementations. This field is the text-to-speech version of header_text. It should contain the same information as header_text but formatted such that it can read as text-to-speech (for example, abbreviations removed, numbers spelled out, etc.)
tts_description_text	TranslatedString	Optional	One	Text containing a description for the alert to be used for text-to-speech implementations. This field is the text-to-speech version of description_text. It should contain the same information as description_text but formatted such that it can be read as text-to-speech (for example, abbreviations removed, numbers spelled out, etc.)
severity_level	SeverityLevel	Optional	One	Severity of the alert.
image	TranslatedImage	Optional	One	TranslatedImage to be displayed along the alert text. Used to explain visually the alert effect of a detour, station closure, etc. The image should enhance the understanding of the alert and must not be the only location of essential information. The following types of images are discouraged : image containing mainly text, marketing or branded images that add no additional information. Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.
image_alternative_text	TranslatedString	Optional	One	Text describing the appearance of the linked image in the <code>image</code> field (e.g., in case the image can't be displayed or the user can't see the image for accessibility reasons). See the HTML spec for alt image text . Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.

enum Cause

Cause of this alert.

Values

Value
UNKNOWN_CAUSE
OTHER_CAUSE

<i>Value</i>
TECHNICAL_PROBLEM
STRIKE
DEMONSTRATION
ACCIDENT
HOLIDAY
WEATHER
MAINTENANCE
CONSTRUCTION
POLICE_ACTIVITY
MEDICAL_EMERGENCY

enum Effect

The effect of this problem on the affected entity.

Values

<i>Value</i>
NO_SERVICE
REDUCED_SERVICE
SIGNIFICANT_DELAYS
DETOUR
ADDITIONAL_SERVICE
MODIFIED_SERVICE
OTHER_EFFECT
UNKNOWN_EFFECT
STOP_MOVED
NO_EFFECT
ACCESSIBILITY_ISSUE

enum SeverityLevel

The severity of the alert.

Caution: this field is still **experimental**, and subject to change. It may be formally adopted in the future.

Values

<i>Value</i>
UNKNOWN_SEVERITY
INFO
WARNING

Value

SEVERE

message TimeRange

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.

Fields

Field Name	Type	Required	Cardinality	Description
start	uint64	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	uint64	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.

message Position

A geographic position of a vehicle.

Fields

Field Name	Type	Required	Cardinality	Description
latitude	float	Required	One	Degrees North, in the WGS-84 coordinate system.
longitude	float	Required	One	Degrees East, in the WGS-84 coordinate system.
bearing	float	Optional	One	Bearing, in degrees, clockwise from True North, i.e., 0 is North and 90 is East. This can be the compass bearing, or the direction towards the next stop or intermediate location. This should not be deduced from the sequence of previous positions, which clients can compute from previous data.
odometer	double	Optional	One	Odometer value, in meters.
speed	float	Optional	One	Momentary speed measured by the vehicle, in meters per second.

message TripDescriptor

A descriptor that identifies a single instance of a GTFS trip, unless `schedule_relationship` is `NEW`, in such case, it specifies a new instance of trip to be added.

To specify a single trip instance, in many cases a `trip_id` by itself is sufficient. However, the following cases require additional information to resolve to a single trip instance:

- For trips defined in `frequencies.txt`, `start_date` and `start_time` are required in addition to `trip_id`
- If the trip lasts for more than 24 hours, or is delayed such that it would collide with a scheduled trip on the following day, then `start_date` is required in addition to `trip_id`
- If the `trip_id` field can't be provided, then `route_id`, `direction_id`, `start_date`, and `start_time` must all be provided

In all cases, if `route_id` is provided in addition to `trip_id`, then the `route_id` must be the same `route_id` as assigned to the given trip in `GTFS trips.txt`.

The `trip_id` field cannot, by itself or in combination with other TripDescriptor fields, be used to identify multiple trip instances. For example, a TripDescriptor should never specify trip_id by itself for GTFS frequencies.txt exact_times=0 trips because start_time is also required to resolve to a single trip instance starting at a specific time of the day. If the TripDescriptor does not resolve to a single trip instance (i.e., it resolves to zero or multiple trip instances), it is considered an error and the entity containing the erroneous TripDescriptor may be discarded by consumers.

Note that if the trip_id is not known, then station sequence ids in TripUpdate are not sufficient, and stop_ids must be provided as well. In addition, absolute arrival/departure times must be provided.

TripDescriptor.route_id cannot be used within an Alert EntitySelector to specify a route-wide alert that affects all trips for a route - use EntitySelector.route_id instead.

If `schedule_relationship` is `NEW`, `trip_id` must be set to a value not listed in the GTFS feed, and `route_id` must be set to a value listed in `routes.txt` in the GTFS static, to associate the trip to a route. `start_date` should be set, and `direction_id` may be set for the new trip.

Fields

Field Name	Type	Required	Cardinality	Description
trip_id	string	Conditionally required	One	The trip_id from the GTFS feed that this selector refers to. For non frequency-based trips (trips not defined in GTFS frequencies.txt), this field is enough to uniquely identify the trip. For frequency-based trips defined in GTFS frequencies.txt, trip_id, start_time, and start_date are all required. For scheduled-based trips (trips not defined in GTFS frequencies.txt), trip_id can only be omitted if the trip can be uniquely identified by a combination of route_id, direction_id, start_time, and start_date, and all those fields are provided. When schedule_relationship is NEW, it must be specified with a unique value not defined in the GTFS static. When schedule_relationship is REPLACEMENT, the trip_id identifies the trip from static GTFS to be replaced. When schedule_relationship is DUPLICATED within a TripUpdate, the trip_id identifies the trip from static GTFS to be duplicated. When schedule_relationship is DUPLICATED within a VehiclePosition, the trip_id identifies the new duplicate trip and must contain the value for the corresponding TripUpdate.TripProperties.trip_id.
route_id	string	Conditionally required	One	The route_id from the GTFS that this selector refers to. If trip_id is omitted, route_id, direction_id, start_time, and schedule_relationship=SCHEDULED must all be set to identify a trip instance. TripDescriptor.route_id should not be used within an Alert EntitySelector to specify a route-wide alert that affects all trips for a route - use EntitySelector.route_id instead. When schedule_relationship is NEW, route_id must be specified for route which the new trip belongs to.
direction_id	uint32	Conditionally required	One	The direction_id from the GTFS feed trips.txt file, indicating the direction of travel for trips this selector refers to. If trip_id is omitted, direction_id must be provided. Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.

Field Name	Type	Required	Cardinality	Description
start_time	string	Conditionally required	One	The initially scheduled start time of this trip instance. When the trip_id corresponds to a non-frequency-based trip, this field should either be omitted or be equal to the value in the GTFS feed. When the trip_id corresponds to a frequency-based trip defined in GTFS frequencies.txt, start_time is required and must be specified for trip updates and vehicle positions. If the trip corresponds to exact_times=1 GTFS record, then start_time must be some multiple (including zero) of headway_secs later than frequencies.txt start_time for the corresponding time period. If the trip corresponds to exact_times=0, then its start_time may be arbitrary, and is initially expected to be the first departure of the trip. Once established, the start_time of this frequency-based exact_times=0 trip should be considered immutable, even if the first departure time changes – that time change may instead be reflected in a StopTimeUpdate. If trip_id is omitted, start_time must be provided. Format and semantics of the field is same as that of GTFS/frequencies.txt/start_time, e.g., 11:15:35 or 25:15:35.
start_date	string	Conditionally required	One	The start date of this trip instance in YYYYMMDD format. For scheduled trips (trips not defined in GTFS frequencies.txt), this field must be provided to disambiguate trips that are so late as to collide with a scheduled trip on a next day. For example, for a train that departs 8:00 and 20:00 every day, and is 12 hours late, there would be two distinct trips on the same time. This field can be provided but is not mandatory for schedules in which such collisions are impossible - for example, a service running on hourly schedule where a vehicle that is one hour late is not considered to be related to schedule anymore. This field is required for frequency-based trips defined in GTFS frequencies.txt. If trip_id is omitted, start_date must be provided.
schedule_relationship	ScheduleRelationship	Optional	One	The relation between this trip and the static schedule. If TripDescriptor is provided in an Alert EntitySelector , the schedule_relationship field is ignored by consumers when identifying the matching trip instance.
modified_trip	ModifiedTripSelector	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.

enum ScheduleRelationship

The relation between this trip and the static schedule. If a new trip is done in accordance with temporary schedule, not reflected in GTFS, then it shouldn't be marked as SCHEDULED, but marked as NEW. If a trip is done in accordance with a modified schedule, not reflected in GTFS, then it shouldn't be marked as SCHEDULED, but marked as REPLACEMENT.

Values

Value	Comment
-------	---------

Value	Comment
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	<i>NOTE: This value has been deprecated as the behavior was unspecified. Use DUPLICATED for an extra trip that is the same as a scheduled trip except the start date or time, or NEW for an extra trip that is unrelated to an existing trip.</i>
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. Trips with <code>schedule_relationship: UNSCHEDULED</code> must also set all StopTimeUpdates <code>schedule_relationship: UNSCHEDULED</code>
CANCELED	A trip that existed in the schedule but was removed.
REPLACEMENT	<p>A trip that replaces an existing scheduled trip, for example, with a changed schedule or a diverted routing. The complete journey of the replacement trip must be specified via <code>StopTimeUpdate</code> s, and the original schedule from the GTFS static isn't used for the replaced instance.</p> <p>REPLACEMENT can be used if the trip is operating on a revised schedule, but must not be used to communicate real-time schedule deviations (predictions) if the vehicle is aimed to follow the schedule listed in <code>stop_times.txt</code> the static GTFS.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
DUPLICATED	<p>A new trip that is the same as an existing scheduled trip except for service start date and time. Used with <code>TripUpdate.TripProperties.trip_id</code>, <code>TripUpdate.TripProperties.start_date</code>, and <code>TripUpdate.TripProperties.start_time</code> to copy an existing trip from static GTFS but start at a different service date and/or time. Duplicating a trip is allowed if the service related to the original trip in (CSV) GTFS (in <code>calendar.txt</code> or <code>calendar_dates.txt</code>) is operating within the next 30 days. The trip to be duplicated is identified via <code>TripUpdate.TripDescriptor.trip_id</code>.</p> <p>This enumeration does not modify the existing trip referenced by <code>TripUpdate.TripDescriptor.trip_id</code> - if a producer wants to cancel the original trip, it must publish a separate <code>TripUpdate</code> with the value of CANCELED. Trips defined in GTFS <code>frequencies.txt</code> with <code>exact_times</code> that is empty or equal to 0 cannot be duplicated. The <code>VehiclePosition.TripDescriptor.trip_id</code> for the new trip must contain the matching value from <code>TripUpdate.TripProperties.trip_id</code> and <code>VehiclePosition.TripDescriptor.ScheduleRelationship</code> must also be set to DUPLICATED.</p> <p><i>Existing producers and consumers that were using the ADDED enumeration to represent duplicated trips must follow the migration guide to transition to the DUPLICATED enumeration.</i></p>
NEW	<p>An extra trip unrelated to any existing trips, for example, to respond to sudden passenger load. The complete journey of the new trip, including all stops and times, must be specified via <code>StopTimeUpdate</code> s.</p> <p><i>Existing producers and consumers that were using the ADDED enumeration to represent new trips unrelated to the static GTFS must follow the migration guide to transition to the NEW enumeration.</i></p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>
DELETED	<p>A trip that existed in the schedule but was removed that must not be shown to users.</p> <p>DELETED should be used instead of CANCELED to indicate that a transit provider would like to entirely remove information about the corresponding trip from consuming applications, so the trip is not shown as cancelled to riders, e.g. a trip that is entirely being replaced by another trip. This designation becomes particularly important if several trips are cancelled and replaced with substitute service. If consumers were to show explicit information about the cancellations it would distract from the more important real-time predictions.</p> <p>Caution: this field is still experimental, and subject to change. It may be formally adopted in the future.</p>

message ModifiedTripSelector

When a service is affected by a trip modification, `ModifiedTripSelector` is used to select some trip. More detail in the [Trip Modification](#) specification.

Values

| **Field Name** | **Type** | **Required** | **Cardinality** | **Description** |

| **modifications_id** | [string](#) | Required | One | The `id` of the `FeedEntity` in which the contained `TripModifications` object affects this trip.

| **affected_trip_id** | [string](#) | Required | One | The `trip_id` from the GTFS feed that is modified by the `modifications_id`

| **start_time** | [string](#) | Optional | One | The initially scheduled start time of this trip instance, applied to the frequency based modified trip. Same definition as **start_time** in [TripDescriptor](#).

| **start_date** | [string](#) | Optional | One | The start date of this trip instance in YYYYMMDD format, applied to the modified trip. Same definition as **start_date** in [TripDescriptor](#).

message VehicleDescriptor

Identification information for the vehicle performing the trip.

Fields

Field Name	Type	Required	Cardinality	Description
id	string	Optional	One	Internal system identification of the vehicle. Should be unique per vehicle, and is used for tracking the vehicle as it proceeds through the system. This id should not be made visible to the end-user; for that purpose use the label field
label	string	Optional	One	User visible label, i.e., something that must be shown to the passenger to help identify the correct vehicle.
license_plate	string	Optional	One	The license plate of the vehicle.
wheelchair_accessible	WheelchairAccessible	Optional	One	If provided, can overwrite the <code>wheelchair_accessible</code> value from the static GTFS.

enum WheelchairAccessible

If a particular trip is accessible to wheelchair. When available, this value should overwrite the `wheelchair_accessible` value from the static GTFS.

Values

Value	Comment
NO_VALUE	The trip doesn't have information about wheelchair accessibility. This is the default behavior. If the static GTFS contains a <code>wheelchair_accessible</code> value, it won't be overwritten.
UNKNOWN	The trip has no accessibility value present. This value will overwrite the value from the GTFS.
WHEELCHAIR_ACCESSIBLE	The trip is wheelchair accessible. This value will overwrite the value from the GTFS.
WHEELCHAIR_INACCESSIBLE	The trip is not wheelchair accessible. This value will overwrite the value from the GTFS.

message EntitySelector

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, in order for an alert to apply to an entity in GTFS it must match all of the 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 EntitySelectors, 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.

Fields

Field Name	Type	Required	Cardinality	Description
agency_id	string	Conditionally required	One	The agency_id from the GTFS feed that this selector refers to.
route_id	string	Conditionally required	One	The route_id from the GTFS that this selector refers to. If direction_id is provided, route_id must also be provided.
route_type	int32	Conditionally required	One	The route_type from the GTFS that this selector refers to.
direction_id	uint32	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. Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.
trip	TripDescriptor	Conditionally required	One	The trip instance from the GTFS 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	string	Conditionally required	One	The stop_id from the GTFS feed that this selector refers to.

message TranslatedString

An internationalized message containing per-language versions of a snippet of text or a URL. One of the strings from a message will be picked up. The resolution proceeds as follows: If the UI language matches the language code of a translation, the first matching translation is picked. If a default UI language (e.g., English) matches the language code of a translation, the first matching translation is picked. If some translation has an unspecified language code, that translation is picked.

Fields

Field Name	Type	Required	Cardinality	Description
translation	Translation	Required	Many	At least one translation must be provided.

message Translation

A localized string mapped to a language.

Field Name	Type	Required	Cardinality	Description
text	string	Required	One	A UTF-8 string containing the message.
language	string	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.

message TranslatedImage

An internationalized message containing per-language versions of an image. One of the images from a message will be picked up. The resolution proceeds as follows: If the UI language matches the language code of a translation, the first matching translation is picked. If a default UI language (e.g., English) matches the language code of a translation, the first matching translation is picked. If some translation has an unspecified language code, that translation is picked.

Caution: this message is still **experimental**, and subject to change. It may be formally adopted in the future.

Fields

Field Name	Type	Required	Cardinality	Description
localized_image	LocalizedImage	Required	Many	At least one localized image must be provided.

message LocalizedImage

A localized image URL mapped to a language.

Field Name	Type	Required	Cardinality	Description
url	string	Required	One	String containing an URL linking to an image. The image linked must less than 2MB. If an image changes in a significant enough way that an update is required on the consumer side, the producer must update the URL to a new one. The URL should be a fully qualified URL that includes http:// or https://, and any special characters in the URL must be correctly escaped. See the following http://www.w3.org/Addressing/URL/4_URL_Recommendations.html for a description of how to create fully qualified URL values.
media_type	string	Required	One	IANA media type as to specify the type of image to be displayed. The type must start with "image/"
language	string	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.

message Shape

Describes the physical path that a vehicle takes when the shape is not part of the (CSV) GTFS, such as for an ad-hoc detour. Shapes belong to Trips and consist of an encoded polyline for more efficient transmission. Shapes do not need to intercept the location of Stops exactly, but all Stops on a trip should lie within a small distance of the shape for that trip, i.e. close to straight line segments connecting the shape points.

Caution: this message is still **experimental**, and subject to change. It may be formally adopted in the future.

Fields

Field Name	Type	Required	Cardinality	Description
shape_id	string	Required	One	Identifier of the shape. Must be different than any <code>shape_id</code> defined in the (CSV) GTFS. Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.

Field Name	Type	Required	Cardinality	Description
encoded_polyline	string	Required	One	Encoded polyline representation of the shape. This polyline must contain at least two points and represent the full shape of the trip where it's used. For more information about encoded polylines, see https://developers.google.com/maps/documentation/utilities/polylinealgorithm Caution: this field is still experimental , and subject to change. It may be formally adopted in the future.

message Stop

Represents a new Stop added to the feed dynamically. All fields are as described in the (CSV) GTFS specification. The location type of the new stop is 0 (routable stop).

Caution: this field is still **experimental**, and subject to change. It may be formally adopted in the future.

Fields

Field Name	Type	Required	Cardinality	Description
stop_id	string	Required	One	Identifier of the stop. Must be different than any <code>stop_id</code> defined in the (CSV) GTFS.
stop_code	TranslatedString	Optional	One	See definition of stops.stop_code in (CSV) GTFS.
stop_name	TranslatedString	Required	One	See definition of stops.stop_name in (CSV) GTFS.
tts_stop_name	TranslatedString	Optional	One	See definition of stops.tts_stop_name in (CSV) GTFS.
stop_desc	TranslatedString	Optional	One	See definition of stops.stop_desc in (CSV) GTFS.
stop_lat	float	Required	One	See definition of stops.stop_lat in (CSV) GTFS.
stop_lon	float	Required	One	See definition of stops.stop_lon in (CSV) GTFS.
zone_id	string	Optional	One	See definition of stops.zone_id in (CSV) GTFS.
stop_url	TranslatedString	Optional	One	See definition of stops.stop_url in (CSV) GTFS.
parent_station	string	Optional	One	See definition of stops.parent_station in (CSV) GTFS.
stop_timezone	string	Optional	One	See definition of stops.stop_timezone in (CSV) GTFS.
wheelchair_boarding	WheelchairBoarding	Optional	One	See definition of stops.wheelchair_boarding in (CSV) GTFS.
level_id	string	Optional	One	See definition of stops.level_id in (CSV) GTFS.
platform_code	TranslatedString	Optional	One	See definition of stops.platform_code in (CSV) GTFS.

enum WheelchairBoarding

Values

Value	Comment
UNKNOWN	No accessibility information for the stop.
AVAILABLE	Some vehicles at this stop can be boarded by a rider in a wheelchair.
NOT_AVAILABLE	Wheelchair boarding is not possible at this stop.

message TripModifications

A `TripModifications` message identifies a list of similar trips which are all affected by particular modifications, such as a detour.

Caution: this field is still **experimental**, and subject to change. It may be formally adopted in the future.

[More about Trip Modifications...](#)

Fields

Field Name	Type	Required	Cardinality	Description
selected_trips	SelectedTrips	Required	Many	A list of selected trips affected by this TripModifications. Needs to contain at least one <code>SelectedTrips</code> . If the value <code>start_times</code> is set, a maximum of one <code>SelectedTrips</code> with one trip_id can be listed.
start_times	string	Optional	Many	A list of start times in the real-time trip descriptor for the trip_id defined in trip_ids. Useful to target multiple departures of a trip_id in a frequency-based trip.
service_dates	string	Required	Many	Dates on which the modification occurs, in the YYYYMMDD format. A trip_id will only be modified if it runs on a given service date; the trip IS NOT required to run on all of the service dates. Producers SHOULD only transmit detours occurring within the next week. The dates provided should not be used as user-facing information, if a user-facing start and end date need to be provided, they can be provided in the linked service alert with <code>service_alert_id</code>
modifications	Modification	Required	Many	A list of modifications to apply to the affected trips.

message Modification

A `Modification` message describes changes to each affected trip starting at `start_stop_selector` .

Caution: this field is still **experimental**, and subject to change. It may be formally adopted in the future.

An example showing the effect of a modification on a particular trip. This modification may also be applied to several other trips.

Propagated detour delays affect all stops following the end of a modification. If a trip has multiple modifications, the delays are accumulated.

Fields

Field Name	Type	Required	Cardinality	Description
start_stop_selector	StopSelector	Required	One	The stop selector of the first stop of the original trip that is to be affected by this modification. Used in conjunction with <code>end_stop_selector</code> . <code>start_stop_selector</code> is required and is used to define the reference stop used with <code>travel_time_to_stop</code> . See travel_time_to_stop for details

Field Name	Type	Required	Cardinality	Description
end_stop_selector	StopSelector	Conditionally required	One	The stop selector of the last stop of the original trip that is to be affected by this modification. The selection is inclusive, so if only one stop_time is replaced by that modification, <code>start_stop_selector</code> and <code>end_stop_selector</code> must be equivalent. If no stop_time is replaced, <code>end_stop_selector</code> must not be provided. It's otherwise required.
propagated_modification_delay	int32	Optional	One	The number of seconds of delay to add to all departure and arrival times subsequent to the last stop inserted by a modification. If a modification affects only the shape (i.e. neither <code>end_stop_selector</code> nor <code>replacement_stops</code> are provided), then the delay propagation begins at the subsequent stop after <code>start_stop_selector</code> . Can be a positive or negative number. If multiple modifications apply to the same trip, the delays accumulate as the trip advances. If the value is not supplied, consumers MAY interpolate or infer the <code>propagated_modification_delay</code> based on other data.
replacement_stops	ReplacementStop	Optional	Many	A list of replacement stops, replacing those of the original trip. The length of the new stop times may be less, the same, or greater than the number of replaced stop times.
service_alert_id	string	Optional	One	An <code>id</code> value from the <code>FeedEntity</code> message that contains the <code>Alert</code> describing this Modification for user-facing communication.
last_modified_time	uint64	Optional	One	This timestamp identifies the moment when the modification has last been changed. In POSIX time (i.e., number of seconds since January 1st 1970 00:00:00 UTC).

message StopSelector

Selector for a stop. Either by `stop_id` or `stop_sequence`. At least one of the two values must be provided.

Caution: this field is still **experimental**, and subject to change. It may be formally adopted in the future.

Fields

Field Name	Type	Required	Cardinality	Description
stop_sequence	uint32	Conditionally Required	One	Must be the same as in stop_times.txt in the corresponding GTFS feed. Either <code>stop_sequence</code> or <code>stop_id</code> must be provided within a <code>StopSelector</code> - both fields cannot be empty. <code>stop_sequence</code> is required for trips that visit the same stop_id more than once (e.g., a loop) to disambiguate which stop the prediction is for.
stop_id	string	Conditionally Required	One	Must be the same as in stops.txt in the corresponding GTFS feed. Either <code>stop_sequence</code> or <code>stop_id</code> must be provided within a <code>StopSelector</code> - both fields cannot be empty.

message SelectedTrips

List of selected trips with an associated shape.

Caution: this field is still **experimental**, and subject to change. It may be formally adopted in the future.

Fields

Field Name	Type	Required	Cardinality	Description
trip_ids	uint32	Many	One	A list of trip_id from the original (CSV) GTFS that are affected by the containing replacement. Need to contain at least one trip_id. A TripUpdate with schedule_relationship=REPLACEMENT must not already exist for the trip.
shape_id	string	Required	One	The ID of the new shape for the modified trips in this SelectedTrips. May refer to a new shape added using a Shape message in the same GTFS-RT feed, or to an existing shape defined in the GTFS-Static feed's shapes.txt. If it refers to a Shape entity in the real-time feed, the value of this field should be the one of the shape_id inside the entity, and <i>not</i> the id of FeedEntity .

message ReplacementStop

Each [ReplacementStop](#) message defines a stop that will now be visited by the trip, and optionally specifies the estimated travel time to the stop.

Caution: this field is still **experimental**, and subject to change. It may be formally adopted in the future.

If a modification affects the first stop of the trip, that stop also serves as the reference stop of the modification.

Fields

Field Name	Type	Required	Cardinality	Description
stop_id	string	Required	One	The replacement stop ID which will now be visited by the trip. May refer to a new stop added using a GTFS-RT Stop message in the same GTFS-RT feed, or to an existing stop defined in the (CSV) GTFS feed's stops.txt . If it refers to a Shape entity in the real-time feed, the value of this field should be the one of the stop_id inside the entity, and <i>not</i> the id of FeedEntity . The stop MUST have location_type=0 (routable stops).
travel_time_to_stop	int32	Optional	One	<p>The difference in seconds between the arrival time at this stop and the arrival time at the reference stop. The reference stop is the stop prior to start_stop_selector. If the modification begins at the first stop of the trip, then the first stop of the trip is the reference stop.</p> <p>This value MUST be monotonically increasing and may only be a negative number if the first stop of the original trip is the reference stop.</p> <p>If the value is not supplied, consumers MAY interpolate or infer the travel_time_to_stop based on other data.</p>