Package 'MBTAr'

September 18, 2015

Type Package

Title Access Data from the Massachusetts Bay Transit Authority (MBTA) Web API

Version 1.0.1

Depends R (>= 3.1.2)

LazyData true

Description Access to the MBTA API for R. Creates an easy-to-use bundle of functions to work with all the built-in calls to the MBTA API. Allows users to download realtime tracking data in dataframe format that is manipulable in standard R analytics functions.

License GPL-3

Imports jsonlite

NeedsCompilation no

Author Justin de Benedictis-Kessner [aut, cre]

Maintainer Justin de Benedictis-Kessner <justindbk@gmail.com>

Repository CRAN

Date/Publication 2015-09-18 22:02:21

R topics documented:

`Talertbyid` *Queries active and upcoming times for a particular alert.*

Description

Returns all information about a given alert. Some fields may be empty for an alert.

Usage

```
Talertbyid(alert_id,include_access_alerts = FALSE,
include_service_alerts = TRUE, api_key)
```

Arguments

`alert_id`	Unique identifier for the alert. Example: "781"
`include_access_alerts`	Whether or not alerts pertaining to accessibility (elevators, escalators) should be returned. Data type: logical. Defaults to FALSE.
`include_service_alerts`	Whether or not service alerts should be returned. Data type: logical. Defaults to TRUE.
`api_key`	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

`alert_id`	The unique identifier for the alert
`effect_name`	The human-readable name for the effect. Example: "Shuttle bus"
`effect`	The GTFS-realtime-compatible code for the effect. Example: "DETOUR"
`cause`	The human-readable name for the cause. Sometimes empty. Example: "maintenance"
`header_text`	A brief summary of the situation (GTFS-realtime-compatible). Example: "Shuttle buses replacing Red Line service from Sat Jun 01, 2013 to Sun Jun 30, 2013 every Saturday and Sunday from 09:00 PM to end of service due to tie replacement"

`short_header_text`
: A shortened version of header_text. Example: "Shuttle buses replacing Red Line service from Sat Apr 27 to Sun May 26 every Saturday and Sunday due to maintenance"

`description_text`
: Additional details (GTFS-realtime-compatible). Example: "Affected stops: Alewife Station Davis Station Porter Square Station Harvard Square Station"

`severity`
: Possible values: "Severe", "Moderate", "Minor"

`created_dt`
: Date and time the alert was created, in epoch time. Example: "1361395938"

`last_modified_dt`
: Date and time the alert was last modified, in epoch time. Example: "1361395938"

`service_effect_text`
: Summarizes the service and the impact to that service. Example: "Minor Route 1 delay"

`timeframe_text`
: Summarizes when an alert is in effect. Example: "starting Saturday"

`alert_lifecycle`
: Identifies whether alert is a new or old, in effect or upcoming. Not intended to be human-readable. Possible values: "Upcoming", "New", "Ongoing", "Ongoing-Upcoming." See notes.

`effect_start`
: Date and time of the start of the effect period, in epoch time. Example: "1367110800"

`effect_end`
: Date and time of the end of the effect period, in epoch time. Can be empty if effect end is not known. Example: "1367130600"

`affected_route_type`
: GTFS-compatible code for route type (i.e. mode). Appears for service alerts only (not access alerts). Example: "1"

`affected_mode_name`
: Human-readable name for the mode. Example: "Subway"

`affected_route_id`
: The unique GTFS-compatible identifier for the route. Example: "Red"

`affected_route_name`
: The human-readable name for the route. Example: "Red Line"

`affected_direction_id`
: The GTFS-compatible identifier for the direction. Example: "0"

`affected_direction_name`
: Human-readable direction name. Example: "Westbound"

`affected_trip_id`
: The GTFS-compatible unique identifier for the trip. Example: "CR-Newburyport-CR-Weekday-129"

`affected_trip_name`
: Human-readable trip name. Example: "129 (5:00 pm from North Station)"

`affected_stop_id`
: The GTFS-compatible unique identifier for the stop. Example: "70061"

`affected_stop_name`
: The GTFS-compatible name for the stop (not unique). Example: "Alewife Station Red Line"

affected_route_hide
: Whether this route should be hidden from users. See notes. Possible values: "true". Only included if "true."

affected_elev_id
: Unique identifier for the elevator/escalator. Example: "926"

affected_elev_name
: Human-readable name for the elevator/escalator. Example: "SOUTH STATION - Lobby to Street"

affected_elev_type
: Type of the elevator/escalator. Possible values: "Elevator", "Escalator", "Lift"

affected_elev_stop_id
: The GTFS-compatible unique identifier for the stop. Example: "70080"

affected_elev_stop_name
: The GTFS-compatible name for the stop (not unique). Example: "South Station - Inbound"

affected_elev_stop_parent_id
: The GTFS-compatible unique identifier for the larger station associated with the stop, if one exists. Can be empty if parent station does not exist. Example: "place_sstat"

See Also

Talerts

Talertheaders *Queries the header text for all active and upcoming alerts.*

Description

Returns alert id numbers and text descriptions of alerts.

Usage

```
 Talertheaders(include_access_alerts = FALSE,
include_service_alerts = TRUE, api_key)
```

Arguments

include_access_alerts
: Whether or not alerts pertaining to accessibility (elevators, escalators) should be returned. Data type: logical. Defaults to FALSE.

include_service_alerts
: Whether or not service alerts should be returned. Data type: logical. Defaults to TRUE.

api_key
: API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

alert_id	The unique identifier for the alert
header_text	A brief summary of the situation (GTFS-realtime-compatible) Example: "Shuttle buses replacing Red Line service from Sat Jun 01, 2013 to Sun Jun 30, 2013 every Saturday and Sunday from 09:00 PM to end of service due to tie replacement"

See Also

Talerts Talertheadersbyroute Talertheadersbystop

Talertheadersbyroute *Queries all active and upcoming alerts for a particular route.*

Description

Returns the alert id numbers and text descriptions of alerts.

Usage

```
Talertheadersbyroute(route_id, include_access_alerts = FALSE,
include_service_alerts = TRUE, api_key)
```

Arguments

route_id	GTFS-compatible route_id value for which alert headers should be returned. Data type: String. Example: "Red"
include_access_alerts	Whether or not alerts pertaining to accessibility (elevators, escalators) should be returned. Data type: logical. Defaults to FALSE.
include_service_alerts	Whether or not service alerts should be returned. Data type: logical. Defaults to TRUE.
api_key	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

route_id	The unique GTFS-compatible identifier for the route for which alert headers are returned. Data type: String. Example: "Red"
route_name	The human-readable name for the route for which alert headers are returned. Example: "Red Line"
alert_id	The unique identifier for the alert
header_text	A brief summary of the situation (GTFS-realtime-compatible) Example: "Shuttle buses replacing Red Line service from Sat Jun 01, 2013 to Sun Jun 30, 2013 every Saturday and Sunday from 09:00 PM to end of service due to tie replacement"

`Talertheadersbystop` *Queries alert header text by stop.*

Description

Returns text description headers for all active and upcoming alerts affecting a particular stop.

Usage

```
Talertheadersbystop(stop_id, include_access_alerts = FALSE,
include_service_alerts = TRUE, api_key)
```

Arguments

`stop_id`	GTFS-compatible stop_id value for which alert headers should be returned. Data type: String. Example: "place-portr"
`include_access_alerts`	Whether or not alerts pertaining to accessibility (elevators, escalators) should be returned. Data type: logical. Defaults to FALSE.
`include_service_alerts`	Whether or not service alerts should be returned. Data type: logical. Defaults to TRUE.
`api_key`	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

`stop_id`	String. The GTFS-compatible unique identifier for the stop for which alert headers are returned. Example: "place-portr"
`stop_name`	The GTFS-compatible name for the stop for which alert headers are returned. Example: "Porter Square Station"
`alert_id`	The unique identifier for the alert
`header_text`	A brief summary of the situation (GTFS-realtime-compatible) Example: "Shuttle buses replacing Red Line service from Sat Jun 01, 2013 to Sun Jun 30, 2013 every Saturday and Sunday from 09:00 PM to end of service due to tie replacement"

`Talerts` *Query all active and upcoming alerts.*

Description

Find current list of all MBTA alerts, including all routes and stops they affect, along with timeframes that the alert is in effect. Returns all possible information for a given alert, often with empty fields.

Usage

```
Talerts(include_access_alerts = FALSE, include_service_alerts = TRUE, api_key)
```

Arguments

`include_access_alerts`	Whether or not alerts pertaining to accessibility (elevators, escalators) should be returned. Data type: logical. Defaults to FALSE.
`include_service_alerts`	Whether or not service alerts should be returned. Data type: logical. Defaults to TRUE.
`api_key`	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

`alert_id`	The unique identifier for the alert
`effect_name`	The human-readable name for the effect. Example: "Shuttle bus"
`effect`	The GTFS-realtime-compatible code for the effect. Example: "DETOUR"
`cause`	The human-readable name for the cause. Seomtimes empty. Example: "maintenance"
`header_text`	A brief summary of the situation (GTFS-realtime-compatible). Example: "Shuttle buses replacing Red Line service from Sat Jun 01, 2013 to Sun Jun 30, 2013 every Saturday and Sunday from 09:00 PM to end of service due to tie replacement"
`short_header_text`	A shortened version of header_text. Example: "Shuttle buses replacing Red Line service from Sat Apr 27 to Sun May 26 every Saturday and Sunday due to maintenance"
`description_text`	Additional details (GTFS-realtime-compatible). Example: "Affected stops: Alewife Station Davis Station Porter Square Station Harvard Square Station"
`severity`	Possible values: "Severe", "Moderate", "Minor"
`created_dt`	Date and time the alert was created, in epoch time. Example: "1361395938"
`last_modified_dt`	Date and time the alert was last modified, in epoch time. Example: "1361395938"

`service_effect_text`
: Summarizes the service and the impact to that service. Example: "Minor Route 1 delay"

`timeframe_text`
: Summarizes when an alert is in effect. Example: "starting Saturday"

`alert_lifecycle`
: Identifies whether alert is a new or old, in effect or upcoming. Not intended to be human-readable. Possible values: "Upcoming", "New", "Ongoing", "Ongoing-Upcoming." See notes.

`effect_start`
: Date and time of the start of the effect period, in epoch time. Example: "1367110800"

`effect_end`
: Date and time of the end of the effect period, in epoch time. Can be empty if effect end is not known. Example: "1367130600"

`affected_route_type`
: GTFS-compatible code for route type (i.e. mode). Appears for service alerts only (not access alerts). Example: "1"

`affected_mode_name`
: Human-readable name for the mode. Example: "Subway"

`affected_route_id`
: The unique GTFS-compatible identifier for the route. Example: "Red"

`affected_route_name`
: The human-readable name for the route. Example: "Red Line"

`affected_direction_id`
: The GTFS-compatible identifier for the direction. Example: "0"

`affected_direction_name`
: Human-readable direction name. Example: "Westbound"

`affected_trip_id`
: The GTFS-compatible unique identifier for the trip. Example: "CR-Newburyport-CR-Weekday-129"

`affected_trip_name`
: Human-readable trip name. Example: "129 (5:00 pm from North Station)"

`affected_stop_id`
: The GTFS-compatible unique identifier for the stop. Example: "70061"

`affected_stop_name`
: The GTFS-compatible name for the stop (not unique). Example: "Alewife Station Red Line"

`affected_route_hide`
: Whether this route should be hidden from users. See notes. Possible values: "true". Only included if "true."

`affected_elev_id`
: Unique identifier for the elevator/escalator. Example: "926"

`affected_elev_name`
: Human-readable name for the elevator/escalator. Example: "SOUTH STATION - Lobby to Street"

`affected_elev_type`
: Type of the elevator/escalator. Possible values: "Elevator", "Escalator", "Lift"

`affected_elev_stop_id`
: The GTFS-compatible unique identifier for the stop. Example: "70080"

`affected_elev_stop_name`
: The GTFS-compatible name for the stop (not unique). Example: "South Station - Inbound"

`affected_elev_stop_parent_id`
: The GTFS-compatible unique identifier for the larger station associated with the stop, if one exists. Can be empty if parent station does not exist. Example: "place_sstat"

Note

Severity:

- "Severity" was created with the intent that it could drive presentation of alerts in a variety of ways - ordering, coloring, icons - and not with the intent that the words "severity," "mild," "moderate," or "severe" would necessarily be shown directly to customers.

Effect Periods:

- More than one 'effect_period' object can be present.
- 'effect_end' can be empty if the end time for an alert is not known.

Alert Lifecycle:

- "New" and "Ongoing" refer to alerts that are in effect now.
- "Upcoming" and "Ongoing-Upcoming" refer to alerts that will be in effect in the future.
- "Ongoing" and "Ongoing-Upcoming" refer to alerts that are "old news," like a station that is closed and has been for weeks.
- An example of an "Ongoing-Upcoming" alert would be a shuttle that has been happening every weekend for a month so far (if you retrieve the data on a weekday.)

Affected Services:

- The affected services for an alert can include either services or elevators/escalators but NOT both. If the 'services' object is empty (i.e. 'service' objects are not present) then the 'elevators' object will not be empty (i.e. 'elevator' objects will be present) and vice versa.
- More than one 'service' object can be present.
- Different service objects can have different combinations of attributes. They may have just a mode and route (affects an entire route), or mode and stop (affects all service at the stop.) Or be much more specific - including mode, route, direction, trip, and stop, indicating that it applies to one scheduled stop on one trip.
- Currently, the system does not allow creation of an alert that applies to multiple elevators/escalators. Therefore, only one 'elevator' object can be present. This may change in the future.
- For alerts that apply to elevators/escalators, 'parent_station' and 'parent_station_name' properties on the 'stop' object can be empty if parent station does not exist.

See Also

`Talertbyid` `Talertsbyroute` `Talertsbystop` `Talertheaders` `Talertheadersbyroute` `Talertheadersbystop`

`Talertsbyroute` *Queries all active and upcoming alerts by route*

Description

Returns all information about alerts affecting a given route. May include some empty fields.

Usage

```
Talertsbyroute(route_id, include_access_alerts = FALSE,
include_service_alerts = TRUE, api_key)
```

Arguments

`route_id`	GTFS-compatible route_id value for which alert information should be returned. Data type: String. Example: "Red"
`include_access_alerts`	Whether or not alerts pertaining to accessibility (elevators, escalators) should be returned. Data type: logical. Defaults to FALSE.
`include_service_alerts`	Whether or not service alerts should be returned. Data type: logical. Defaults to TRUE.
`api_key`	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

`route_id`	The unique GTFS-compatible identifier for the route for which alerts are returned. Example: "Red"
`route_name`	The human-readable name for the route for which alerts are returned. Example: "Red Line"
`alert_id`	The unique identifier for the alert
`effect_name`	The human-readable name for the effect. Example: "Shuttle bus"
`effect`	The GTFS-realtime-compatible code for the effect. Example: "DETOUR"
`cause`	The human-readable name for the cause. Seomtimes empty. Example: "maintenance"
`header_text`	A brief summary of the situation (GTFS-realtime-compatible). Example: "Shuttle buses replacing Red Line service from Sat Jun 01, 2013 to Sun Jun 30, 2013 every Saturday and Sunday from 09:00 PM to end of service due to tie replacement"
`short_header_text`	A shortened version of header_text. Example: "Shuttle buses replacing Red Line service from Sat Apr 27 to Sun May 26 every Saturday and Sunday due to maintenance"

`description_text`
: Additional details (GTFS-realtime-compatible). Example: "Affected stops: Alewife Station Davis Station Porter Square Station Harvard Square Station"

`severity`
: Possible values: "Severe", "Moderate", "Minor"

`created_dt`
: Date and time the alert was created, in epoch time. Example: "1361395938"

`last_modified_dt`
: Date and time the alert was last modified, in epoch time. Example: "1361395938"

`service_effect_text`
: Summarizes the service and the impact to that service. Example: "Minor Route 1 delay"

`timeframe_text`
: Summarizes when an alert is in effect. Example: "starting Saturday"

`alert_lifecycle`
: Identifies whether alert is a new or old, in effect or upcoming. Not intended to be human-readable. Possible values: "Upcoming", "New", "Ongoing", "Ongoing-Upcoming." See notes.

`effect_start`
: Date and time of the start of the effect period, in epoch time. Example: "1367110800"

`effect_end`
: Date and time of the end of the effect period, in epoch time. Can be empty if effect end is not known. Example: "1367130600"

`affected_route_type`
: GTFS-compatible code for route type (i.e. mode). Appears for service alerts only (not access alerts). Example: "1"

`affected_mode_name`
: Human-readable name for the mode. Example: "Subway"

`affected_direction_id`
: The GTFS-compatible identifier for the direction. Example: "0"

`affected_direction_name`
: Human-readable direction name. Example: "Westbound"

`affected_trip_id`
: The GTFS-compatible unique identifier for the trip. Example: "CR-Newburyport-CR-Weekday-129"

`affected_trip_name`
: Human-readable trip name. Example: "129 (5:00 pm from North Station)"

`affected_stop_id`
: The GTFS-compatible unique identifier for the stop. Example: "70061"

`affected_stop_name`
: The GTFS-compatible name for the stop (not unique). Example: "Alewife Station Red Line"

`affected_route_hide`
: Whether this route should be hidden from users. See notes. Possible values: "true". Only included if "true."

`affected_elev_id`
: Unique identifier for the elevator/escalator. Example: "926"

`affected_elev_name`
: Human-readable name for the elevator/escalator. Example: "SOUTH STATION - Lobby to Street"

affected_elev_type
: Type of the elevator/escalator. Possible values: "Elevator", "Escalator", "Lift"

affected_elev_stop_id
: The GTFS-compatible unique identifier for the stop. Example: "70080"

affected_elev_stop_name
: The GTFS-compatible name for the stop (not unique). Example: "South Station - Inbound"

affected_elev_stop_parent_id
: The GTFS-compatible unique identifier for the larger station associated with the stop, if one exists. Can be empty if parent station does not exist. Example: "place_sstat"

Talertsbystop *Queries all active and upcoming alerts for a particular stop.*

Description

Returns all information on alerts affecting a given stop. May include some empty fields.

Usage

```
Talertsbystop(stop_id, include_access_alerts = FALSE, include_service_alerts =
TRUE, api_key)
```

Arguments

stop_id
: GTFS-compatible stop_id value for which alert information should be returned. Data type: String. Example: "place-portr"

include_access_alerts
: Whether or not alerts pertaining to accessibility (elevators, escalators) should be returned. Data type: logical. Defaults to FALSE.

include_service_alerts
: Whether or not service alerts should be returned. Data type: logical. Defaults to TRUE.

api_key
: API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

stop_id
: The GTFS-compatible unique identifier for the stop for which alerts are returned. Data type: String. Example: "place-portr"

stop_name
: The GTFS-compatible name for the stop for which alerts are returned. Data type: String. Example: "Porter Square Station"

alert_id
: The unique identifier for the alert

effect_name
: The human-readable name for the effect. Example: "Shuttle bus"

`effect` The GTFS-realtime-compatible code for the effect. Example: "DETOUR"

`cause` The human-readable name for the cause. Seomtimes empty. Example: "maintenance"

`header_text` A brief summary of the situation (GTFS-realtime-compatible). Example: "Shuttle buses replacing Red Line service from Sat Jun 01, 2013 to Sun Jun 30, 2013 every Saturday and Sunday from 09:00 PM to end of service due to tie replacement"

`short_header_text` A shortened version of header_text. Example: "Shuttle buses replacing Red Line service from Sat Apr 27 to Sun May 26 every Saturday and Sunday due to maintenance"

`description_text` Additional details (GTFS-realtime-compatible). Example: "Affected stops: Alewife Station Davis Station Porter Square Station Harvard Square Station"

`severity` Possible values: "Severe", "Moderate", "Minor"

`created_dt` Date and time the alert was created, in epoch time. Example: "1361395938"

`last_modified_dt` Date and time the alert was last modified, in epoch time. Example: "1361395938"

`service_effect_text` Summarizes the service and the impact to that service. Example: "Minor Route 1 delay"

`timeframe_text` Summarizes when an alert is in effect. Example: "starting Saturday"

`alert_lifecycle` Identifies whether alert is a new or old, in effect or upcoming. Not intended to be human-readable. Possible values: "Upcoming", "New", "Ongoing", "Ongoing-Upcoming." See notes.

`effect_start` Date and time of the start of the effect period, in epoch time. Example: "1367110800"

`effect_end` Date and time of the end of the effect period, in epoch time. Can be empty if effect end is not known. Example: "1367130600"

`affected_route_type` GTFS-compatible code for route type (i.e. mode). Appears for service alerts only (not access alerts). Example: "1"

`affected_mode_name` Human-readable name for the mode. Example: "Subway"

`affected_route_id` The unique GTFS-compatible identifier for the route. Example: "Red"

`affected_route_name` The human-readable name for the route. Example: "Red Line"

`affected_direction_id` The GTFS-compatible identifier for the direction. Example: "0"

`affected_direction_name` Human-readable direction name. Example: "Westbound"

`affected_trip_id` The GTFS-compatible unique identifier for the trip. Example: "CR-Newburyport-CR-Weekday-129"

affected_trip_name
: Human-readable trip name. Example: "129 (5:00 pm from North Station)"

affected_route_hide
: Whether this route should be hidden from users. See notes. Possible values: "true". Only included if "true."

affected_elev_id
: Unique identifier for the elevator/escalator. Example: "926"

affected_elev_name
: Human-readable name for the elevator/escalator. Example: "SOUTH STATION - Lobby to Street"

affected_elev_type
: Type of the elevator/escalator. Possible values: "Elevator", "Escalator", "Lift"

affected_elev_stop_id
: The GTFS-compatible unique identifier for the stop. Example: "70080"

affected_elev_stop_name
: The GTFS-compatible name for the stop (not unique). Example: "South Station - Inbound"

affected_elev_stop_parent_id
: The GTFS-compatible unique identifier for the larger station associated with the stop, if one exists. Can be empty if parent station does not exist. Example: "place_sstat"

`Tpredictionsbyroute` *Query predictions by route*

Description

Returns predicted upcoming arrivals and departures in the next hour for a particular route.

Usage

```
 Tpredictionsbyroute(route_id,
include_access_alerts = FALSE, include_service_alerts = TRUE, api_key)
```

Arguments

route_id
: GTFS-compatible route_id value for which predictions should be returned. Example: "Orange"

include_access_alerts
: Whether or not alerts pertaining to accessibility (elevators, escalators) should be returned. Data type: logical. Defaults to FALSE.

include_service_alerts
: Whether or not service alerts should be returned. Data type: logical. Defaults to TRUE.

api_key
: API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

route_id	The unique GTFS-compatible identifier for the route for which predictions are returned. Example: "CR-Franklin"
route_name	The human-readable name for the route for which predictions are returned. Example: "Franklin Line"
route_type	The GTFS-compatible identifier for the type of service (mode). Example: "2"
mode_name	The human-readable name for the type of service (mode). Example: "Commuter Rail"
direction_id	The GTFS-compatible identifier for the direction. Example: "0"
direction_name	The human-readable name for the direction. Example: "Outbound"
trip_id	The unique GTFS-compatible identifier for the trip. Example: "CR-Providence-CR-Weekday-815"
trip_name	The human-readable name for the trip. Example: "815 (4:35 pm from South Station)"
trip_headsign	The text that identifies the trip's destination to passengers. Example: "North Station"
vehicle_id	The GTFS-compatible unique identifier for the vehicle. Example: "1531"
vehicle_lat	The GTFS-compatible latitude of the vehicle. Example: "42.08997"
vehicle_lon	The GTFS-compatible longitude of the vehicle. Example: "-71.4388"
vehicle_bearing	GTFS-compatable bearing of the vehicle. This can be the compass bearing, or the direction towards the next stop or intermediate location. May be empty. Example: "259"
vehicle_speed	Identifies the vehicle's momentary speed, in meters per second. Example: "21"
vehicle_timestamp	Identifies the moment when the content of this feed has been created, in epoch time. Example: "1400855704"
stop_sequence	Identifies where the stop comes in the sequence of stops for this trip. Example: "2"
stop_id	The GTFS-compatible unique identifier for the stop. Example: "Back Bay"
stop_name	The GTFS-compatible name for the stop. Example: "Back Bay"
sch_arr_dt	Scheduled arrival time at the stop for the trip, in epoch time. Example: "1361986080"
sch_dep_dt	Scheduled departure time at the stop for the trip, in epoch time. Example: "1361986080"
pre_dt	Predicted time at the stop - departure time for origin stop and arrival time for all other stops - for the trip, in epoch time. Example: "1400855700"
pre_away	Predicted amount of time until the vehicle arrives at the stop, in seconds. Example: "339"

`Tpredictionsbystop` *Query predictions by stop*

Description

Returns arrivals and departures in the next hour for a direction and route for a given stop.

Usage

```
 Tpredictionsbystop(stop_id,
include_access_alerts = FALSE, include_service_alerts = TRUE, api_key)
```

Arguments

`stop_id`	GTFS-compatible stop_id value for which predictions should be returned. Example: "Back Bay"
`include_access_alerts`	Whether or not alerts pertaining to accessibility (elevators, escalators) should be returned. Data type: logical. Defaults to FALSE.
`include_service_alerts`	Whether or not service alerts should be returned. Data type: logical. Defaults to TRUE.
`api_key`	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

`stop_id`	The GTFS-compatible unique identifier for the stop for which the predictions are returned. Example: "Back Bay"
`stop_name`	The GTFS-compatible name for the stop for which the predictions are returned. Example: "Back Bay"
`route_type`	The GTFS-compatible identifier for the type of service (mode). Example: "2"
`mode_name`	The human-readable name for the type of service (mode). Example: "Commuter Rail"
`route_id`	The unique GTFS-compatible identifier for the route. Example: "Red"
`route_name`	The human-readable name for the route. Example: "Red Line"
`direction_id`	The GTFS-compatible identifier for the direction. Example: "0"
`direction_name`	The human-readable name for the direction. Example: "Outbound"
`trip_id`	The unique GTFS-compatible identifier for the trip. Example: "CR-Providence-CR-Weekday-815"
`trip_name`	The human-readable name for the trip. Example: "815 (4:35 pm from South Station)"
`trip_headsign`	The text that identifies the trip's destination to passengers. Example: "North Station"

vehicle_id	The GTFS-compatible unique identifier for the vehicle. Example: "1531"
vehicle_lat	The GTFS-compatible latitude of the vehicle. Example: "42.08997"
vehicle_lon	The GTFS-compatible longitude of the vehicle. Example: "-71.4388"
vehicle_bearing	GTFS-compatable bearing of the vehicle. This can be the compass bearing, or the direction towards the next stop or intermediate location. May be empty. Example: "259"
vehicle_speed	Identifies the vehicle's momentary speed, in meters per second. Example: "21"
vehicle_timestamp	Identifies the moment when the content of this feed has been created, in epoch time. Example: "1400855704"
stop_sequence	Identifies where the stop comes in the sequence of stops for this trip. Example: "2"
stop_id	The GTFS-compatible unique identifier for the stop. Example: "Back Bay"
stop_name	The GTFS-compatible name for the stop. Example: "Back Bay"
sch_arr_dt	Scheduled arrival time at the stop for the trip, in epoch time. Example: "1361986080"
sch_dep_dt	Scheduled departure time at the stop for the trip, in epoch time. Example: "1361986080"
pre_dt	Predicted time at the stop - departure time for origin stop and arrival time for all other stops - for the trip, in epoch time. Example: "1400855700"
pre_away	Predicted amount of time until the vehicle arrives at the stop, in seconds. Example: "339"

See Also

Tpredictionsbyroute

Tpredictionsbytrip *Query predictions by trip*

Description

Returns the predicted arrival and departure times for a given trip.

Usage

```
Tpredictionsbytrip(trip_id, api_key)
```

Arguments

trip_id	GTFS-compatible trip_id value for which vehicle positions should be returned. Data type: String. Example: "CR-Providence-CR-Weekday-807"
api_key	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

route_id	The unique GTFS-compatible identifier for the route for which vehicle positions are returned. Example: "CR-Providence"
route_name	The human-readable name for the route for which vehicle positions are returned. Example: "Providence/Stoughton Line"
route_type	The GTFS-compatible identifier for the type of service (mode). Example: "2"
mode_name	The human-readable name for the type of service (mode). Example: "Commuter Rail"
trip_id	The unique GTFS-compatible identifier for the trip for which vehicle positions are returned. Example: "CR-Providence-CR-Weekday-815"
trip_name	The human-readable for the trip for which schedule is returned. Example: "815 (4:35 pm from South Station)"
trip_headsign	The text that identifies the trip's destination to passengers. Example: "North Station"
direction_id	The GTFS-compatible identifier for the direction. Example: "0"
direction_name	The human-readable name for the direction. Example: "Outbound"
vehicle_id	The GTFS-compatible unique identifier for the vehicle. Example: "1531"
vehicle_lat	The GTFS-compatible latitude of the vehicle. Example: "42.08997"
vehicle_lon	The GTFS-compatible longitude of the vehicle. Example: "-71.4388"
vehicle_bearing	GTFS-compatable bearing of the vehicle. This can be the compass bearing, or the direction towards the next stop or intermediate location. May be empty. Example: "259"
vehicle_speed	Identifies the vehicle's momentary speed, in meters per second. Example: "21"
vehicle_timestamp	Identifies the moment when the content of this feed has been created, in epoch time. Example: "1400855704"
stop_sequence	Identifies where the stop comes in the sequence of stops for this trip. Example: "2"
stop_id	The GTFS-compatible unique identifier for the stop. Example: "Back Bay"
stop_name	The GTFS-compatible name for the stop. Example: "Back Bay"
sch_arr_dt	Scheduled arrival time at the stop for the trip, in epoch time. Example: "1361986080"
sch_dep_dt	Scheduled departure time at the stop for the trip, in epoch time. Example: "1361986080"
pre_dt	Predicted time at the stop - departure time for origin stop and arrival time for all other stops - for the trip, in epoch time. Example: "1400855700"
pre_away	Predicted amount of time until the vehicle arrives at the stop, in seconds. Example: "339"

See Also

Tpredictionsbyroute Tpredictionsbystop

`Troutes` *Query all T routes*

Description

Returns information about all routes for which information can be requested.

Usage

```
Troutes(api_key)
```

Arguments

`api_key`	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

`route_type`	The GTFS-compatible identifier for the type of service (mode). Example: "2"
`mode_name`	The human-readable name for the type of service (mode). Example: "Commuter Rail"
`route_id`	The unique GTFS-compatible identifier for the route. Example: "CR-Providence"
`route_name`	The human-readable name for the route. Example: "Providence/Stoughton Line"

See Also

`Troutesbystop`

Examples

```
  ## Authenticate:

  mykey <- NULL
  mykey <- assign("mykey", value = test_key, envir = .GlobalEnv)
  ## use your own key from http://realtime.mbta.com/Portal/ instead of the test key

Troutes(api_key = mykey)

## returns:
    # route_type     mode_name         route_id                 route_name
# 1            0        Subway          Green-B                Green Line B
# 2            0        Subway          Green-C                Green Line C
# 3            0        Subway          Green-D                Green Line D
# 4            0        Subway          Green-E                Green Line E
# 5            0        Subway         Mattapan            Mattapan Trolley
# 6            1        Subway             Blue                   Blue Line
# 7            1        Subway           Orange                 Orange Line
# 8            1        Subway              Red                    Red Line
```

```
# 9           2 Commuter Rail     CR-Fairmount              Fairmount Line
# 10          2 Commuter Rail     CR-Fitchburg              Fitchburg Line
# ...
# 207         3           Bus             9702                        9702
# 208         3           Bus             9703                        9703
# 209         4          Boat          Boat-F4           Charlestown Ferry
# 210         4          Boat          Boat-F1               Hingham Ferry
# 211         4          Boat          Boat-F3                  Hull Ferry
```

`Troutesbystop` *Query all routes at a stop*

Description

Returns routes that serve a particular stop.

Usage

```
Troutesbystop(stop_id = NULL, api_key)
```

Arguments

`stop_id`	GTFS-compatible stop_id value for which routes should be returned. Example: "70065"
`api_key`	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

`stop_id`	The GTFS-compatible unique identifier for the stop for which routes are returned. Example: "70065"
`stop_name`	The GTFS-compatible name for the stop for which routes are returned. Example: "Porter Sq - Inbound"
`route_type`	The GTFS-compatible identifier for the type of service (mode). Example: "2"
`mode_name`	The human-readable name for the type of service (mode). Example: "Commuter Rail"
`route_id`	The unique GTFS-compatible identifier for the route. Example: "CR-Providence"
`route_name`	The human-readable name for the route. Example: "Providence/Stoughton Line"

See Also

`Troutes`

Tschedulebyroute *Query the schedule by route*

Description

Returns the scheduled arrivals and departures in a direction for a particular route.

Usage

```
 Tschedulebyroute(route_id, direction = NULL,
datetime = Sys.time(), max_time = 60, max_trips = 5, api_key)
```

Arguments

route_id	GTFS-compatible route_id value for which schedule should be returned. Example: "CR-Providence"
direction	GTFS-compatible direction_id value on route for which schedule should be returned. If not included then schedule for all directions of the route will be returned. Example: "0"
datetime	Epoch time after which schedule should be returned. If included then must be within the next seven (7) days. If not included then schedule starting from the current datetime will be returned. Example: "1361989200"
max_time	Defines maximum range of time (in minutes) within which trips will be returned. Integer between 1 and 1440 (24 hours). If not included defaults to 60. Example: "120"
max_trips	Defines number of trips to return. Integer between 1 and 100. If not included defaults to 5. Example: "100"
api_key	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

route_id	The unique GTFS-compatible identifier for the route for which the schedule is returned. Example: "CR-Providence"
route_name	The human-readable name for the route for which the schedule is returned. Example: "Providence/Stoughton Line"
direction_id	The GTFS-compatible identifier for the direction. Example: "0"
direction_name	The human-readable name for the direction. Example: "Outbound"
trip_id	The unique GTFS-compatible identifier for the trip. Example: "CR-Providence-CR-Weekday-815"
trip_name	The human-readable name for the trip. Example: "815 (4:35 pm from South Station)"
stop_sequence	Identifies where the stop comes in the sequence of stops for this trip. Example: "2"

stop_id	The GTFS-compatible unique identifier for the stop. Example: "Back Bay"
stop_name	The GTFS-compatible name for the stop. Example: "Back Bay"
sch_arr_dt	Scheduled arrival time at the stop for the trip, in epoch time. Example: "1361986080"
sch_dep_dt	Scheduled departure time at the stop for the trip, in epoch time. Example: "1361986080"

See Also

Tpredictionsbyroute Tschedulebystop Tschedulebytrip

Tschedulebystop *Query schedule by stop*

Description

Returns scheduled arrival and departure times for a direction and rotue for a particular stop.

Usage

```
 Tschedulebystop(stop_id, route_id,
direction = NULL, datetime = Sys.time(), max_time = 60, max_trips = 5,
api_key)
```

Arguments

stop_id	GTFS-compatible stop_id value for which schedule should be returned. Example: "Back Bay"
route_id	GTFS-compatible route_id value on the stop for which schedule should be returned. If not included then schedule for all routes serving the stop will be returned. Example: "CR-Providence"
direction	GTFS-compatible direction_id value on route of the stop for which schedule should be returned. Bit (0 or 1). If included then route must also be included. If not included then schedule for all directions of the route serving the stop will be returned. Example: "0"
datetime	Epoch time after which schedule should be returned. If included then must be within the next seven (7) days. If not included then schedule starting from the current datetime will be returned. Example: "1361989200"
max_time	Defines maximum range of time (in minutes) within which trips will be returned. Integer between 1 and 1440 (24 hours). If not included defaults to 60. Example: "120"
max_trips	Defines number of trips to return. Integer between 1 and 100. If not included defaults to 5. Example: "100"
api_key	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

stop_id	The GTFS-compatible unique identifier for the stop for which the schedule is returned. Example: "Back Bay"
stop_name	The GTFS-compatible name for the stop for which the schedule is returned. Example: "Back Bay"
route_type	The GTFS-compatible identifier for the type of service (mode). Example: "2"
mode_name	The human-readable name for the type of service (mode). Example: "Commuter Rail"
route_id	The unique GTFS-compatible identifier for the route. Example: "CR-Providence"
route_name	The human-readable name for the route. Example: "Providence/Stoughton Line"
direction_id	The GTFS-compatible identifier for the direction. Example: "0"
direction_name	The human-readable name for the direction. Example: "Outbound"
trip_id	The unique GTFS-compatible identifier for the trip. Example: "CR-Providence-CR-Weekday-815"
trip_name	The human-readable name for the trip. Example: "815 (4:35 pm from South Station)"
sch_arr_dt	Scheduled arrival time at the stop for the trip, in epoch time. Example: "1361986080"
sch_dep_dt	Scheduled departure time at the stop for the trip, in epoch time. Example: "1361986080"

Note

If the GTFS-compatible stop_id value in the stop parameter in the request is for a parent station then all routes that serve that parent station are returned.

See Also

Tpredictionsbystop Tschedulebyroute Tschedulebytrip

Tschedulebytrip *Query schedule by trip*

Description

Returns scheduled arrival and departure times for a particular trip.

Usage

```
Tschedulebytrip(trip_id, datetime = Sys.time(), api_key)
```

Arguments

trip_id	GTFS-compatible trip_id value for which schedule should be returned. Example: "CR-Providence-CR-Weekday-807"
datetime	Epoch time after which schedule should be returned. If included then must be within the next seven (7) days. If not included then schedule starting from the current datetime will be returned, using Sys.time converted to epoch time. Example: "1361989200"
api_key	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

route_id	The unique GTFS-compatible identifier for the route for which schedule is returned. Example: "CR-Providence"
route_name	The human-readable name for the route for which schedule is returned. Example: "Providence/Stoughton Line"
trip_id	The unique GTFS-compatible identifier for the trip for which schedule is returned. Example: "CR-Providence-CR-Weekday-815"
trip_name	The human-readable for the trip for which schedule is returned. Example: "815 (4:35 pm from South Station)"
direction_id	The GTFS-compatible identifier for the direction. Example: "0"
direction_name	The human-readable name for the direction. Example: "Outbound"
stop_sequence	Identifies where the stop comes in the sequence of stops for this trip. Example: "2"
stop_id	The GTFS-compatible unique identifier for the stop. Example: "Back Bay"
stop_name	The GTFS-compatible name for the stop. Example: "Back Bay"
sch_arr_dt	Scheduled arrival time at the stop for the trip, in epoch time. Example: "1361986080"
sch_dep_dt	Scheduled departure time at the stop for the trip, in epoch time. Example: "1361986080"

See Also

Tpredictionsbytrip Tschedulebystop Tschedulebyroute

Tservertime *Queries the current MBTA API server time.*

Description

Returns the server time. Useful if needing to sync user computer time with the epoch time of MBTA events.

Usage

```
Tservertime(api_key)
```

Arguments

api_key	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

server_dt	Server time, in epoch time. Example: "1361996667"

Tstopsbylocation *Query stops by goegraphic location*

Description

Returns information about the nearest stops to a particular location. Up to 15 are returned, within a 1-mile radius.

Usage

```
Tstopsbylocation(lat, lon, api_key)
```

Arguments

lat	The latitude for location near which stops should be returned. Example: "42.352913"
lon	The longitude for location near which stops should be returned. Example: "-71.064648"
api_key	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

stop_id	The GTFS-compatible unique identifier for the stop. Example: "70063"
stop_name	The GTFS-compatible name for the stop (not unique). Example: "Davis Sq - Inbound"
parent_station	The GTFS-compatible unique identifier for the station associated with the stop. (note: can be empty if stop does not have an associated station). Example: "place-davis"
parent_station_name	The human-readable name for the larger station associated with the stop. (note: can be empty if stop does not have an associated station). Example: "Davis Station"
stop_lat	The GTFS-compatible latitude of the station. Example: "42.3967399597168"
stop_lon	The GTFS-compatible longitude of the station. Example: "-71.1218185424805"
distance	The distance of the stop from the requested location in miles. Example: "0.00800655130296946"

`Tstopsbyroute` *Query stops by route*

Description

Returns the stops for a particular route, specified by id or by name.

Usage

```
Tstopsbyroute(route_id = NULL, route_name = NULL, api_key)
```

Arguments

`route_id`	GTFS-compatible route_id value for which stops should be returned. Example: "Red"
`route_name`	The full human-readable name for the route for which vehicle positions are returned. Example: "Red Line"
`api_key`	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

`route_id`	The unique GTFS-compatible identifier for the route for which stops are returned. Example: "Orange"
`route_name`	The human-readable name for the route for which stops are returned. Example: "Orange Line"
`direction_id`	The GTFS-compatible identifier for the direction. Example: "0"
`direction_name`	The human-readable name for the direction. Example: "Southbound"
`stop_order`	Identifies where the stop comes in the order of stops for this route and direction (note: not guaranteed to be unique). Example: "1"
`stop_id`	The GTFS-compatible unique identifier for the stop. Example: "70063"
`stop_name`	The GTFS-compatible name for the stop (not unique). Example: "Davis Sq - Inbound"
`parent_station`	The GTFS-compatible unique identifier for the station associated with the stop. (note: can be empty if stop does not have an associated station). Example: "place-davis"
`parent_station_name`	The human-readable name for the larger station associated with the stop. (note: can be empty if stop does not have an associated station). Example: "Davis Station"
`stop_lat`	The GTFS-compatible latitude of the station. Example: "42.3967399597168"
`stop_lon`	The GTFS-compatible longitude of the station. Example: "-71.1218185424805"

Tvehiclesbyroute *Query vehicles by a route*

Description

Returns vehicle positions for upcoming trips (including trips already underway) in a direction for a particular route.

Usage

```
Tvehiclesbyroute(route_id, api_key)
```

Arguments

route_id	GTFS-compatible route_id value for which vehicle positions should be returned. Example: "Red"
api_key	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

route_id	The unique GTFS-compatible identifier for the route for which vehicle positions are returned. Example: "CR-Franklin"
route_name	The human-readable name for the route for which vehicle positions are returned. Example: "Franklin Line"
route_type	The GTFS-compatible identifier for the type of service (mode). Example: "2"
mode_name	The human-readable name for the type of service (mode). Example: "Commuter Rail"
direction_id	The GTFS-compatible identifier for the direction. Example: "0"
direction_name	The human-readable name for the direction. Example: "Outbound"
trip_id	The unique GTFS-compatible identifier for the trip. Example: "CR-Providence-CR-Weekday-815"
trip_name	The human-readable name for the trip. Example: "815 (4:35 pm from South Station)"
trip_headsign	The text that identifies the trip's destination to passengers. Example: "North Station"
vehicle_id	The GTFS-compatible unique identifier for the vehicle. Example: "1531"
vehicle_lat	The GTFS-compatible latitude of the vehicle. Example: "42.08997"
vehicle_lon	The GTFS-compatible longitude of the vehicle. Example: "-71.4388"
vehicle_bearing	GTFS-compatable bearing of the vehicle. This can be the compass bearing, or the direction towards the next stop or intermediate location. May be empty. Example: "259"

vehicle_speed Identifies the vehicle's momentary speed, in meters per second. Example: "21"

vehicle_timestamp
Identifies the moment when the content of this feed has been created, in epoch time. Example: "1400855704"

See Also

Tvehiclesbytrip

Tvehiclesbytrip *Query vehicles by a particular trip.*

Description

Returns the predicted vehicle positions for a given trip.

Usage

```
Tvehiclesbytrip(trip_id, api_key)
```

Arguments

trip_id	GTFS-compatible trip_id value for which vehicle positions should be returned. Data type: String. Example: "CR-Providence-CR-Weekday-807"
api_key	API key for MBTA API. To obtain one, visit the MBTA Developer Portal (http://realtime.mbta.com/Portal/)

Value

route_id	The unique GTFS-compatible identifier for the route for which vehicle positions are returned. Example: "CR-Providence"
route_name	The human-readable name for the route for which vehicle positions are returned. Example: "Providence/Stoughton Line"
route_type	The GTFS-compatible identifier for the type of service (mode). Example: "2"
mode_name	The human-readable name for the type of service (mode). Example: "Commuter Rail"
trip_id	The unique GTFS-compatible identifier for the trip for which vehicle positions are returned. Example: "CR-Providence-CR-Weekday-815"
trip_name	The human-readable for the trip for which schedule is returned. Example: "815 (4:35 pm from South Station)"
trip_headsign	The text that identifies the trip's destination to passengers. Example: "North Station"
direction_id	The GTFS-compatible identifier for the direction. Example: "0"
direction_name	The human-readable name for the direction. Example: "Outbound"

`vehicle_id` The GTFS-compatible unique identifier for the vehicle. Example: "1531"

`vehicle_lat` The GTFS-compatible latitude of the vehicle. Example: "42.08997"

`vehicle_lon` The GTFS-compatible longitude of the vehicle. Example: "-71.4388"

`vehicle_bearing`
GTFS-compatable bearing of the vehicle. This can be the compass bearing, or the direction towards the next stop or intermediate location. May be empty. Example: "259"

`vehicle_speed` Identifies the vehicle's momentary speed, in meters per second. Example: "21"

`vehicle_timestamp`
Identifies the moment when the content of this feed has been created, in epoch time. Example: "1400855704"

See Also

`Tvehiclesbyroute`

Index