PTV Timetable API – V2 to V3 Migration Guide

October 2016

Table of Contents

[Introduction 3](#_Toc463264004)

[Key improvements in Version 3 4](#_Toc463264005)

[APIs 5](#_Toc463264006)

[Version 2 APIs and Version 3 equivalents 5](#_Toc463264007)

[Response objects 7](#_Toc463264008)

[Version 2 objects and Version 3 equivalents 7](#_Toc463264009)

[Terminology 10](#_Toc463264010)

[Version 2 terms and Version 3 equivalents 10](#_Toc463264011)

[Version 3 terms and Version 2 equivalents 11](#_Toc463264012)

[Appendix 1 - Object diagrams 13](#_Toc463264013)

[Version 2 object structure 13](#_Toc463264014)

[Version 3 object structure 17](#_Toc463264015)

# Introduction

The PTV Timetable API exists in two distinct versions – Version 2 (first launched in March 2014 and subsequently updated), and a new Version 3 (launched in October 2016). The current versions of the API at the date of this document are:

* Version 2.3.0
* Version 3.0.0

This document has been created to assist users transitioning from Version 2 of the API to Version 3.

# Key improvements in Version 3

* Both calls in the Version 3 Departure API accept GTFS stop\_id as an input (using the optional “gtfs” parameter), effectively creating a Broad Next Departures (GTFS Input) API, in addition to Specific Next Departures (GTFS Input).
* The Version 3 Disruptions API allows you to search for disruptions by route\_id (and even by specific disruption id), so you can focus on the disruptions that are relevant to your query.
* The Version 3 Search API has been augmented with optional location inputs (i.e. latitude and longitude parameters), effectively merging the Version 2 Search and Transport POIs by Map APIs and broadening the search functionality.
* The Version 3 Departures API returns key data for each departure but makes stop, route, run, direction and even disruption information optional in your response; so you can choose how much data – or how little – is returned.
* Legacy object properties that were redundant have been removed.
* The Version 3 API structure is more streamlined.
* Terminology in the Version 3 API is consistent throughout calls.

# APIs

## Version 2 APIs and Version 3 equivalents

| **API V2** | **V2 objects returned** | **V3 equivalent API** | **v3 URL** | **v3 response** |
| --- | --- | --- | --- | --- |
| **Healthcheck** | indicators for: - security token - client clock - memory cache - database | **n/a - health of system returned as part of all APIs** | n/a | "health" property in the V3.Status object returned at the end of all responses |
| **Stops Nearby** | stop object returned within 'result' object | **Stops** | v3/stops/location/{latitude},{longitude} | stop objects |
| **Transport POIs by Map** | stop and outlet POI objects | **Search (plus optional parameters)** | v3/search/{search\_term} | stop, route and outlet objects |
| **Search** | stop and line objects | **Search (plus optional parameters)** | v3/search/{search\_term} | stop, route and outlet objects |
| **Lines by Mode** | line objects | **Routes API (plus optional parameters)** | v3/routes | route objects |
| **Stops on a Line** | stop objects | **Stops** | v3/stops/route/{route\_id}/route\_type/{route\_type} | stop objects |
| **Stop facilities** | stop facilities object | **Stops** | v3/stops/{stop\_id}/route\_type/{route\_type} | stop details object |
| **Stop facilities (GTFS input)** | stop facilities objects | **n/a – stop\_id obtained from Departures API can be used as input to Stops API** | n/a | stop details object |
| **Broad Next Departures\*** | departure time 'values' object and applicable disruption information | **Departures** | /v3/departures/route\_type/{route\_type}/stop/{stop\_id} | departure objects (and optional stop, route, run, direction and/or disruption objects) |
| **Specific Next Departures** | departure time 'values' object and applicable disruption information | **Departures** | /v3/departures/route\_type/{route\_type}/stop/{stop\_id}/route/{route\_id} | departure objects (and optional stop, route, run, direction and/or disruption objects) |
| **Specific Next Departures (GTFS input)** | departure time 'values' object and applicable disruption information | **Departures (plus optional parameter)** | /v3/departures/route\_type/{route\_type}/stop/{stop\_id}/route/{route\_id} | departure objects (and optional stop, route, run, direction and/or disruption objects) |
| **Stopping Pattern** | departure time 'values' object and applicable disruption information | **Patterns** | /v3/pattern/run/{run\_id}/route\_type/{route\_type} | departure and disruption objects |
| **Disruptions** | disruption information and applicable line and direction objects | **Disruptions (plus optional parameters)** | v3/disruptions | disruption objects and applicable route and direction objects |

# Response objects

## Version 2 objects and Version 3 equivalents

| **V2 object name** | **V2 object properties** | **V3 object name** | **V3 object properties** |
| --- | --- | --- | --- |
| **values (i.e. timetable)** | time\_timetable\_utc,time\_realtime\_utc,flags,***platform object,run object,disruptions object*** | V3.Departures | stop\_id,route\_id,run\_id,direction\_id,disruption\_ids,scheduled\_departure\_utc,estimated\_departure\_utc,at\_platform,platform\_number,flags |
| **platform** | platform\_number,at\_platform\_now,realtime\_id,***stop object,direction object*** | see V3.Departures | n/a |
| **run** | transport\_typeroute\_type,run\_id,num\_skipped,destination\_id,destination\_name,status | V3.Run | run\_id,route\_type,final\_stop\_id,destination\_name,status |
| **stop** | distance,suburb,transport\_typeroute\_type,stop\_id,location\_name,lat,lon | V3.StopOnRoute, V3.StopGeosearch, V3.ResultStop | stop\_distance[[1]](#footnote-1),stop\_name,stop\_id,route\_type,stop\_latitude,stop\_longitude |
| **direction** | linedir\_id,direction\_id,direction\_name,***line object*** | V3.Direction | direction\_id,direction\_name,route\_id,route\_type |
| **line** | transport\_typeroute\_type,line\_id,line\_name,line\_number,line\_name\_short,line-\_number\_long | V3.Route | route\_type,route\_id,route\_name,route\_number |
| **outlet** | outlet\_type,suburb,business\_name,distance,location\_name,lat,lon | V3.ResultOutlet | outlet\_distance,outlet\_name,outlet\_business,outlet\_latitude,outlet\_longitude |
| **disruptions** | disruption\_id,title,url,description,status,type,publishedOn,lastUpdated,fromDate,toDate,service\_time,***line object(s)*** | V3.Disruption | disruption\_id,title,url,description,disruption\_status,disruption\_type,published\_on,last\_updated,from\_date,to\_date,***routes array*** |
| **line (relating to disruptions)** | transport\_typeroute\_type,line\_id,line\_name,line\_number,line\_name\_short,line-\_number\_long,***direction object*** | V3.DisruptionRoute | route\_type,route\_id,route\_name,route\_number,***direction object*** |
| **direction (relating to disruption)** | linedir\_id,direction\_id,direction\_name,service\_time | V3.DisruptionDirection | route\_direction\_id,direction\_id,direction\_name,service\_time |
| **stop facilities** | stop\_id,stop\_mode\_id,stop\_type,stop\_type\_description,***location object,amenity object,accessibility object*** | V3.StopDetails | station\_type,station\_description,route\_type,***stop\_location object,stop\_amenities object,stop\_accessibility object***,stop\_id,stop\_name |
| **location** | suburb,***gps object***,postcode,municipality,municipality\_id,primary\_stop\_name,road\_type\_primary,second\_stop\_name,road\_type\_second,bay\_nbr | V3.StopLocation | ***gps object*** |
| **gps** | latitude,longitude | V3.StopGps | latitude,longitude |
| **amenity** | toilet,taxi\_rank,car\_parking,cctv | V3.StopAmenityDetails | toilet,taxi\_rank,car\_parking,cctv |
| **accessibility** | lighting,stairs,escalator,lifts,hearing\_loop,tactile\_tiles,***wheelchair object*** | V3.StopAccessibility | lighting,stairs,escalator,lifts,hearing\_loop,tactile\_tiles,***wheelchair object*** |
| **wheelchair** | accessible\_ramp,accessible\_parking,accessible\_phone,accessible\_toilet | V3.StopAccessibilityWheelchair | accessible\_ramp,accessible\_parking,accessible\_phone,accessible\_toilet |

# Terminology

## Version 2 terms and Version 3 equivalents

(list ordered alphabetically)

**business\_name** *context:**outlet object property*
V3 equivalent: **outlet\_business** (see V3 Swagger #Search)

**destination\_id** *context:**property in run object*
V3 equivalent: **final\_stop\_id** (see V3 Swagger #Runs)

**line\_id** *context:**line object property*
V3 equivalent: **route\_id** (see V3 Swagger #Routes)

**line\_name** *line object property*
V3 equivalent: **route\_name** (see V3 Swagger #Routes)

**line\_name\_short** *line object property*
V3 equivalent: **n/a** (see V3 Swagger #Routes for more information about route objects)

**line\_number** *line object property*
V3 equivalent: **route\_number** (see V3 Swagger #Routes)

**line\_number\_long** *line object property*
V3 equivalent: **n/a** (see V3 Swagger #Routes for more information about route objects)

**linedir\_id** *direction object property*
V3 equivalent: **route\_direction\_id** (see V3 Swagger #Disruptions - V3.DisruptionDirection)

**location\_name** *stop object property*
V3 equivalent: **stop\_name** (see V3 Swagger #Stops and #Search)

**location\_name** *outlet object*
V3 equivalent: **outlet\_name** (see V3 Swagger #Search)

**mode** *parameter (multiple APIs)*
V3 equivalent: **route\_type**

**nightrider** *a value of transport\_type, mode and poi*V3 equivalent: **Night Bus**

**num\_skipped** *run object property*
V3 equivalent: **n/a** (see V3 Swagger #Runs for more information about run objects)

**outlet\_type** *outlet object property*
V3 equivalent: **n/a** (see V3 Swagger #Search for more information about outlet objects)

**poi** *parameter (Transport POIs by Map API)*
V3 equivalent: **route\_type parameter** (note: outlets are not included in V3 route\_type; see V3 Swagger #Search **include\_outlets parameter**)

**realtime\_id** *timetable value object property*
V3 equivalent: **n/a** (see V3 Swagger #Departures and #Patterns for more information about departures objects)

**stop\_point\_id** *parameter (Stop Facilities (GTFS Input) API)*
V3 equivalent: **n/a** (see V3 Swagger #Stops v3/stops/{stop\_id}/route\_type/{route\_type} for more information about stop objects)

**time\_realtime\_utc** *timetable value object property*
V3 equivalent: **estimated\_departure\_utc** (see V3 Swagger #Departures and #Patterns)

**time\_timetable\_utc** *timetable value object property*
V3 equivalent: **scheduled\_departure\_utc** (see V3 Swagger #Departures and #Patterns)

**transport\_type** *stop/line/run object property*
V3 equivalent: n/a (this attribute was deprecated in V3 in favour of route\_type)

## Version 3 terms and Version 2 equivalents

(list ordered alphabetically)

**estimated\_departure\_utc** *V3.Departure object property*
V2 equivalent: **time\_realtime\_utc** (see timetable value object)

**final\_stop\_id** *V3.Runs object property*
V2 equivalent: **destination\_id** (see run object)

**outlet\_business** *V3.ResultOutlet object property*
V2 equivalent: **business\_name** (see outlet object)

**outlet\_name** *V3.ResultOutlet object property*
V2 equivalent: **location\_name** (see outlet object)

**route\_direction\_id** *V3.DisruptionDirection object property*
V2 equivalent: **linedir\_id** (see direction object)

**route\_id** *V3.Route object property*
V2 equivalent: **line\_id** (seeline object)

**route\_name** *V3.Route object property*
V2 equivalent: **line\_name** (seeline object)

**route\_number** *V3.Route object property*
V2 equivalent: **line\_number** (seeline object)

**route\_type** *parameter (multiple V3 APIs)* [[2]](#footnote-2)V2 equivalent: **mode or POI parameter** (multiple V2 APIs)

**scheduled\_departure\_utc** *V3.Departure object property*
V2 equivalent: **time\_timetable\_utc** (see timetable value object)

**stop\_name** *V3.StopDetails, V3.ResultStop, V3.Stop and V3.StopGeosearch object property*
V2 equivalent: **location\_name** (seestop object)

# Appendix 1 - Object diagrams

## Version 2 object structure



**“values” (i.e. timetable) object** - Returned by **Broad Next Departures, Specific Next Departures, Specific Next Departures (GTFS Input)** and **Stopping Pattern** API

“platform” object, “run” object, time\_timetable\_utc, time\_realtime\_utc, flags, “disruptions” object

**“platform” object**

platform\_number, at\_platform\_now, realtime\_id, “stop” object, “direction” object

**"run" object**

transport\_type, route\_type, run\_id, num\_skipped, destination\_id, destination\_name, status

**"stop" object** - Also returned by **Stops Nearby** and **Search** (as a “result” object - with extra attribute 'type'), **Transport POIs by Map** (as a “locations” object) and **Stops on a Line**

distance, suburb, transport\_type, route\_type, stop\_id, location\_name, lat, lon

**"direction" object**

line, dir\_id, direction\_id , direction\_name, “line” object

**"line" object** - Also returned by **Search** (as a "result" object - with extra attribute 'type') and **Lines by Mode**

transport\_type, route\_type, line\_id, line\_name, line\_number, line\_name\_short, line\_number\_long

**"locations" object** - Returned by **Transport POIs by Map**

Contains either a "stop" object or an "outlet" object

**"outlet" object**

outlet\_type, suburb, business\_name, distance, location\_name, lat, lon

**"result" object** - Returned by **Stops Nearby** and **Search**

Contains either a "stop" object or a "line" object, together with extra attribute 'type'

**“disruptions” object –** Returned by **Broad Next Departures** (as a list after each applicable timetable value), and by **Specific Next Departures**, **Specific Next Departures (GTFS Input)** and **Stopping Pattern** (if applicable, as a separate array after the timetable values)

disruption\_id, title, url, description, status, type, publishedOn, lastUpdated, fromDate, toDate, service\_time

**“stop facilities” object** – Returned by **Stop Facilities** and **Stop Facilities (GTFS Input)**

stop\_id, stop\_mode\_id, stop\_type, stop\_type\_description, “location” object, “amenity” object, “accessibility” object

**“location” object**

suburb, “gps” object, postcode, municipality, municipality\_id, primary\_stop\_name, road\_type\_primary, second\_stop\_name, road\_type\_second, bay\_nbr

**"gps" object**

longitude, latitude

**“accessibility” object**

lighting, stairs, escalator, lifts, hearing\_loop, tactile\_tiles, “wheelchair” object

**"wheelchair" object**

accessible\_ramp, accessible\_parking, accessible\_phone, accessible\_toilet

**“amenity” object**

toilet, taxi\_rank, car\_parking, cctv

*mode object* - Returned by **Disruptions**

Contains “disruption information” object

**“disruption information” object**

disruption\_id, title, url, description, status, type, publishedOn, lastUpdated, fromDate, toDate, “lines” object

**"lines" object**

transport\_type, route\_type, line\_id, line\_name, line\_number, line\_name\_short, line\_number\_long, “direction” object

**"direction" object**

linedir\_id, direction\_id, direction\_name, service\_time

## Version 3 object structure



**V3.DeparturesResponse** – returned by **Departures API** (both calls)

departures – i.e. V3.Departure (array), stops objects (optional), routes objects (optional), runs objects (optional), directions objects (optional), disruptions objects – i.e. V3.Disruption (optional), status object

**V3.Departure**

stop\_id, route\_id, run\_id, direction\_id, disruption\_ids (array), scheduled\_departure\_utc, estimated\_departure\_utc, at\_platform, platform\_number, flags

**V3.Status**

version, health

*The following objects are returned if included within Expand parameter:*

**stops (optional)**

stop\_distance, stop\_name, stop\_id, route\_type, stop\_latitude, stop\_longitude

**routes (optional)**

route\_type, route\_id, route\_name, route\_number

**runs (optional)**

run\_id, route\_type, final\_stop\_id, destination\_name, status

**directions (optional)**

direction\_id, direction\_name, route\_id, route\_type

**V3.Disruption (optional)**

disruption\_id, title, url, description, disruption\_status, published\_on, last\_updated, from\_date, to\_date, routes (array)

**V3.DisruptionRoute**

route\_type, route\_id, route\_name, route\_number, direction (array)

**V3.DisruptionDirection**

route\_direction\_id, direction\_id, direction\_name, service\_time

**V3.DisruptionsResponse** – returned by **Disruptions API /v3/disruptions** and **/v3/disruptions/route/{route\_id}**

disruptions (array), status

**V3.Disruptions**

general (array), metro\_train (array), metro\_tram (array), metro\_bus (array), regional\_train (array), regional\_coach (array), regional\_bus (array)

**V3.Disruption**

disruption\_id, title, url, description, disruption\_status, published\_on, last\_updated, from\_date, to\_date, routes (array)

**V3.DisruptionRoute**

route\_type, route\_id, route\_name, route\_number, direction (array)

**V3.DisruptionDirection**

route\_direction\_id, direction\_id, direction\_name, service\_time

**V3.Status**

version, health

**V3.DisruptionResponse** – returned by **Disruptions API /v3/disruptions/{disruption\_id}**

disruption, status

**V3.Disruption**

disruption\_id, title, url, description, disruption\_status, published\_on, last\_updated, from\_date, to\_date, routes (array)

**V3.DisruptionRoute**

route\_type, route\_id, route\_name, route\_number, direction (array)

**V3.DisruptionDirection**

route\_direction\_id, direction\_id, direction\_name, service\_time

**V3.Status**

version, health

**V3.StoppingPattern** – returned by **Patterns API**

departures – i.e. V3.Departure (array), disruptions – i.e. V3.Disruption (array), status

**V3.Departure**

stop\_id, route\_id, run\_id, direction\_id, disruption\_ids (array), scheduled\_departure\_utc, estimated\_departure\_utc, at\_platform, platform\_number, flags

**V3.Disruption** (only returned if any applicable disruptions exist)

disruption\_id, title, url, description, disruption\_status, published\_on, last\_updated, from\_date, to\_date, routes (array)

**V3.DisruptionRoute**

route\_type, route\_id, route\_name, route\_number, direction (array)

**V3.DisruptionDirection**

route\_direction\_id, direction\_id, direction\_name, service\_time

**V3.Status**

version, health

**V3.RouteResponse** – returned by **Routes API** (all calls)

route, status

**V3.Route**

route\_type, route\_id, route\_name, route\_number

**V3.Status**

version, health

**V3.DirectionsResponse** – returned by **Directions API** (all calls)

directions (array), status

**V3.Direction**

direction\_id, direction\_name, route\_id, route\_type

**V3.Status**

version, health

**V3.RunsResponse** – returned by **Runs API /v3/runs/route/{route\_id} and /v3/runs/{run\_id}**

runs (array), status

**V3.Run**

run\_id, route\_type, final\_stop\_id, destination\_name, status

**V3.Status**

version, health

**V3.RunResponse** – returned by **Runs API /v3/runs/route\_type/{route\_type}**

run, status

**V3.Run**

run\_id, route\_type, final\_stop\_id, destination\_name, status

**V3.Status**

version, health

**V3.SearchResult** – returned by **Search API**

stops (array), routes (array), outlets (array), status

**V3.ResultStop**

stop\_distance, stop\_name, stop\_id, route\_type, stop\_latitude, stop\_longitude

**V3.ResultRoute**

route\_name, route\_number, route\_type, route\_id

**V3.ResultOutlet**

outlet\_distance, outlet\_name, outlet\_business, outlet\_latitude, outlet\_longitude

**V3.Status**

version, health

**V3.StopsOnRouteResponse** – returned by **Stops API /v3/stops/route/{route\_id}/route\_type/{route\_type}**

stops (array), status

**V3.StopOnRoute**

stop\_name, stop\_id, route\_type, stop\_latitude, stop\_longitude

**V3.Status**

version, health

**V3.StopsByDistanceResponse** – returned by **Stops API /v3/stops/location/{latitude},{longitude}**

stops (array), status

**V3.StopGeosearch**

stop\_distance, stop\_name, stop\_id, route\_type, stop\_latitude, stop\_longitude

**V3.Status**

version, health

**V3.StopResponse** – returned by **Stops API /v3/stops/{stop\_id}/route\_type/{route\_type}**

**V3.StopDetails**

station\_type, station\_description, route\_type, stop\_location, stop\_amenities, stop\_accessibility, stop\_id, stop\_name

**V3.StopLocation**

gps

**V3.StopGps**

latitude, longitude

**V3.StopAmenityDetails**

toilet, taxi\_rank, car\_parking, cctv

**V3.StopAccessibility**

lighting, stairs, escalator, lifts, hearing\_loop, tactile\_tiles, wheelchair

V3.StopAccessibilityWheelchair

accessible\_ramp, accessible\_parking, accessible\_phone, accessible\_ramp

**V3.Status**

version, health

1. stop\_distance only returned in V3.StopGeosearch and V3.ResultStop objects; not relevant to V3.StopOnRoute object. [↑](#footnote-ref-1)
2. route\_type is also used as an object property for multiple objects in both V3 (and V2); here it is only referenced in its role as a parameter in V3, and the V2 equivalent. [↑](#footnote-ref-2)