chicago transit authority
Bus Tracker API documentation

Introduction

The Bus Tracker API allows for querying data from the CTA Bus Tracker service. The system can provide estimated arrival times for buses as they approach bus stops, as well as data on routes, stops, patterns, detours, cancelations and more. Information about buses and their estimated arrivals is updated about once every 30 seconds.

Note that in using this API, you must agree to our License Agreement and Terms of Use.

Table of Contents

Overview

What is the Bus Tracker API?

The Bus Tracker API allows you to request and retrieve real-time data directly from BusTime (the system which produces estimated arrival times and which provides location and route information in real-time).

Registered third-party developers can make HTTP requests for data and receive XML or JSON responses from the Bus Tracker web server.

What data is available through the API?

Data available through the API includes:

- Vehicle locations
- Route data (route lists, stop lists geo-positional route definitions, etc.)
- Prediction Data

Will my application break if changes are made to the API?

No. The versioning of the API allows time for developers to upgrade their applications to make use of new API features. Note that occasionally new parameters may be added to an existing request or its response. However, existing parameters will never be removed or stop accepting previously legal values.

Continuing to work with a particular version of the API guarantees that an application will not break. When a new version is released, it will offer new features and fixes that would break compatibility if added to the previous version. Using this method allows developers to continue using the same version in their current applications while working to make use of the new features of the next version.

How does the Developer API work?

The developer API uses the same data from the BusTime system, which powers CTA Bus Tracker. Information about the location, direction and status of CTA buses is fed from each bus and delivered to the BusTime system, which then can show where buses are or estimate arrival times to stops ahead of a bus.

Data is updated about once every 30 seconds, and arrival estimations are based on how long it normally takes for a bus to get from one place to the next. Because traffic conditions and other unexpected delays occur, we can’t predict precisely when a bus will arrive—only estimate based on normal travel times during the time of day where an estimate is occurring.

In order to use the API, you must sign in to their CTA Bus Tracker account and request an API key. Only one key will be available per account. Once your request has been approved, an e-mail will be sent to you, containing the API key. After receiving the key, you will be able to make calls to the API, entering the key as part of the data request.

Is there a limit to the number of requests I can make to the Developer API?

Yes. By default, one API key can make a maximum of 100,000 requests. If you believe that you will require more than 100,000 daily requests, you must request that the cap on your key be raised to handle the additional traffic.

How are dynamic changes to schedule data handled?

Version 3 introduces some dynamic data which fundamentally changes the proper use of the API. Dynamic changes can be split into two categories: Detours and Disruption Management. Before these changes, it may have been sufficient for an application to request route data once during startup. If the API user wants to support dynamic changes, it is likely that the client will need to make repeated requests for route data such as stops and patterns.

Detours are temporary changes in pattern data. Detour patterns appear normally in the **getpatterns** call but have a **dtrid** identifying the detour. These patterns also come with a **dtrpt** array, which allows the application to show the *original* pattern that is no longer in effect (as a dashed line on a map, for example).

These new temporary patterns may add or remove stops from the original pattern that is being detoured. Stops (retrieved via **getstops**) which are affected by detours will have **dtradd** and/or **dtrrem** elements containing the identifier of the detour.

Scheduled arrival times may also be affected by detours, but there is no means of detecting this in **getpredictions** results. The client application should display the predictions normally even for detours.

The client application should rely on the new **getdetours** call to retrieve detour metadata and present this data to the end user when detour changes are encountered throughout the API.

Disruption Management is a suite of actions which can change the trip data of the schedule. Some examples are canceling or expressing arrivals and canceling, shifting, or creating trips. The API represents these changes in **getpredictions** using new elements for each prediction. Most rider-facing applications only need to be concerned about the **dyn** element, which may label a prediction as canceled or expressed (drop-off only).

Why we're providing this API

The hope is that, in places where travel information is either desired or deemed useful, that information about the state of CTA services can help people make empowered decisions about their trip. By providing an API into this information already published on our Web site, we hope to see applications of this information included in mobile device applications, public displays, Web sites and more.

Legal notice

By using this API, you agree to our Developer Terms of Use. It's important that you, the developer, understand that this service is provided on an as-is basis and without any guarantees as to availability or accuracy. You must read and agree to the full Developer Terms of Use to use this API.

Web Service

The Bus Tracker API is a web that uses HTTP/1.1 as its application protocol. Each type of call or request that can be made to the API is represented by a unique URL. Requests are made to the API using HTTP GET calls to the appropriate URL. Parameters are encoded in the HTTP GET request by following the URL with a "?" and "argument=value" pairs separated by "&".

A response is returned as a well-formed XML document with a Content-Type of "text/xml", or as a JSON document with a Content-Type of "application/json".

For example, to request the current system time through the developer API, a program or script will make a HTTP/1.1 GET request to the following URL with parameters:

https://www.ctabustracker.com/bustime/api/v3/gettime?key=89dj2he89d8j3j3ksjhdue93j

www.ctabustracker.com is the host on which the Developer API is servicing HTTP requests. The service is running on port 80.

The version of the API that is being accessed is built into the URL. In the above example, "**v3**" represents version 3.0 of the API.

The "**key**" parameter represents the API key assigned to the developer making the request. All requests to the API must be accompanied by a valid API key.

In Versions 2 and later, an optional "**format**" parameter can be included to specify the response type. XML is the default response format, and is used as the default if the "**format**" parameter is not included. JSON can be chosen by including "**format=json**".

This document's reference only details information about Version 3. For information about other versions of the API, review that version's document instead, as the request and response formats of different versions may not be compatible with one another.

Reference

This section describes all requests that can be made to the Bus Tracker API. For every request, a complete set of possible arguments is specified, along with the response. For XML responses, the schema is specified.

Definitions

- **Delayed Vehicle –** The state entered by a vehicle when it has been determined to be stationary for more than a pre-defined time period.
- **Direction –** Common direction of travel of a route.
- **Format –** The document type of the response. Currently XML and JSON are supported.
- **Locale –** A string that represents the language to be used for the request. A list of valid locales can be retrieved using getlocalelist. They are in ISO form, such as “en”, which would be English.
- **Off-route Vehicle** – State entered by a transit vehicle when it has strayed from its scheduled pattern.
- **Pattern** – A unique sequence of geo-positional points (waypoints and stops) that combine to form the path that a transit vehicle will repetitively travel. A route often has more than one possible pattern.
- **Route** – One or more set of patterns that together form a single service.
- **Stop –** Location where a transit vehicle can pick-up or drop-off passengers. Predictions are only generated at stops.
- **Waypoint –** A geo-positional point in a pattern used to define the travel path of a transit vehicle.

Common Parameters

All request URLs have these parameters in common:

Name	Supported Versions	Required?	Example	Description
version	All	Yes	/v3/	The version of the API being used. Legal values are v1, v2, and v3.
locale	All	No	locale=en	The language that the response should be in. See the reference for "Locale" for more details on how to use this field.
format	v2+	No	format=json	The format of the response. Legal values are "xml" and "json". XML is the default if no format is requested.

Time

Base URL: https://www.ctabustracker.com/bustime/api/v3/gettime

Parameters:

Name	Value	Description
key	string (required)	25-digit Bus Tracker API access key.
unixTime	boolean (optional)	If true, returns the number of milliseconds that have elapsed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970.

Response

A well-formed XML or JSON document containing the current system time will be returned as a response to **gettime**.

Response Fields:

Name	Description
bustime-response	Root element of the response document.
error	Child element of the root element. Contains a message if the processing of the request resulted in an error.
tm	Child element of the root element containing the current system date and time (local). Date and time are represented in the following format: YYYYMMDD HH:MM:SS. Month is represented as two digits where January is "01" and December is "12". Time is represented using a 24-hour clock. If the parameter unixTime=true is used, returns the number of milliseconds that have elapsed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970.

Remarks

Use the **gettime** request to retrieve the current system date and time. Since BusTime is a time-dependent system, it is important to synchronize your application with BusTime’s system date and time.

The time given in the schema below is the local time.

XML Schema

```
<?xml version="1.0" encoding="utf-8" ?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:element name="bustime-response" type=" bustime-response" />
      <xs:complexType name="bustime-response">
            <xs:sequence>
                  <xs:element name="error" type="error" minOccurs="0" maxOccurs="unbounded"/>
                  <xs:element name="tm" type="xs:string" minOccurs="0" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="error">
            <xs:sequence>
                  <xs:element name="msg" type="xs:string" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
</xs:schema>
```

Example

Request:

https://www.ctabustracker.com/bustime/api/v3/gettime?key=89dj2he89d8j3j3ksjhdue93j

Response:

```
<?xml version=”1.0”?>
<bustime-response>
      <tm>20160308 14:42:32</tm>
</bustime-response>
```

Request:

https://ctabustracker.com/bustime/api/v3/gettime?key=89dj2he89d8j3j3ksjhdue93j&format=json

Response:

```
{
      "bustime-response": {
            "tm": "20160308 14:11:54"
      }
}
```

Request:

https://ctabustracker.com/bustime/api/v3/gettime?key=89dj2he89d8j3j3ksjhdue93j&unixTime=true

Response:

```
<?xml version=”1.0”?>
<bustime-response>
      <tm>1531859957528</tm>
</bustime-response>
```

Request:

https://ctabustracker.com/bustime/api/v3/gettime?key=89dj2he89d8j3j3ksjhdue93j&unixTime=true&format=json

Response:

```
{
	"bustime-response": {
		"tm": "1531860021189"
	}
}
```

Vehicles

<u>Base URL</u>: https://www.ctabustracker.com/bustime/api/v3/getvehicles

Parameters:

Name	Value	Description
key	string (required)	25-digit Bus Tracker API access key.
vid	comma-delimited list of vehicle IDs (not available with **rt** parameter)	Set of one or more vehicle IDs whose location should be returned. For example: 509,392,201,4367 will return information for four vehicles (if available). A maximum of 10 identifiers can be specified.
rt	comma-delimited list of route designators (not available with **vid** parameter)	Set of one or more route designators for which matching vehicles should be returned. For example: X3,4,20 will return information for all vehicles currently running those three routes (if available). A maximum of 10 identifiers can be specified.
tmres	string (optional)	Resolution of time stamps. Set to “s” to get time resolution to the second. Set to “m” to get time resolution to the minute. If omitted, defaults to “m”. Date and time is represented in the following format: If specified as “s” *YYYYMMDD HH:MM:SS* If specified as “m” *YYYYMMDD HH:MM* Month is represented as two digits where January is equal to “01” and December is equal to “12”. Time is represented using a 24-hour clock.

Response

A well-formed XML or JSON document will be returned as a response to **getvehicles**. The response will include the most-recent status for each vehicle.

Response Fields:

Name	Description
bustime-response	Root element of the response document.
error	Child element of the root element. Message if the processing of the request resulted in an error.
vehicle	Child element of the root element. Encapsulates all information available for a single vehicle in the response.
vid	Child element of the **vehicle** element. Alphanumeric string representing the vehicle ID (ie. bus number)
tmstmp	Child element of the **vehicle** element. Date and local time of the last positional update of the vehicle. Date and time is represented in the following format: YYYYMMDD HH:MM. Month is represented as two digits where January is equal to “01” and December is equal to “12”. Time is represented using a 24-hour clock.
lat	Child element of the **vehicle** element. Latitude position of the vehicle in decimal degrees (WGS 84).
lon	Child element of the **vehicle** element. Longitude position of the vehicle in decimal degrees (WGS 84).
hdg	Child element of the **vehicle** element. Heading of vehicle as a 360º value, where 0º is North, 90º is East, 180º is South and 270º is West.
pid	Child element of the **vehicle** element. Pattern ID of trip currently being executed.
pdist	Child element of the **vehicle** element. Linear distance in feet that the vehicle has traveled into the pattern currently being executed.
rt	Child element of the **vehicle** element. Route that is currently being executed by the vehicle (ex. “20”).
des	Child element of the **vehicle** element. Destination of the trip being executed by the vehicle (ex. “Austin”).
dly	Child element of the **vehicle** element. The value is “true” if the vehicle is delayed. The **dly** element is only present if the vehicle is delayed.
tablockid	Child element of the **vehicle** element. TA’s version of the scheduled block identifier for the work currently being performed by the vehicle.
tatripid	Child element of the **vehicle** element. TA’s version of the scheduled trip identifier for the vehicle’s current trip.
origtatripno	Child element of the **vehicle** element. Trip ID defined by the TA scheduling system.
zone	Child element of the **prd** element. The zone name if the vehicle has entered a

	defined zone, otherwise blank.
mode	Child element of the **vehicle** element. Mode of transportation for the vehicle as a byte with range 0-4. 0 is None, 1 is Bus, 2 is Ferry, 3 is Rail, and 4 is People_Mover.
psgld	Child element of the **vehicle** element. String representing the ratio of the current passenger count to the vehicle's total capacity. Possible values include "FULL", "HALF_EMPTY", "EMPTY" and "N/A". Ratios for "FULL", "HALF_EMPTY" and "EMPTY" are determined by the transit agency. "N/A" indicates that the passenger load is unknown.
stst	Child element of the **vehicle** element. Contains the scheduled start time (in seconds past midnight) of the trip that the vehicle is running on.
stsd	Child element of the **vehicle** element. Contains the scheduled start date (in "yyyy-mm-dd" format) of the trip that the vehicle is running on.

Remarks

Use the **getvehicles** request to retrieve vehicle information (i.e., locations) of all or a subset of vehicles currently being tracked by BusTime.

Use the **vid** parameter to retrieve information for one or more vehicles currently being tracked.

Use the **rt** parameter to retrieve information for vehicles currently running one or more of the specified routes.

Note: The **vid** and **rt** parameters cannot be combined in one request. If both parameters are specified on a request to **getvehicles,** only the first parameter specified on the request will be processed.

XML Schema

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="bustime-response" type="bustime-response"/>
     <xs:complexType name="bustime-response">
          <xs:sequence>
               <xs:element name="error" type="error" minOccurs="0" maxOccurs="unbounded"/>
               <xs:element name="vehicle" type="vehicle" minOccurs="0" maxOccurs="unbounded"/>
          </xs:sequence>
     </xs:complexType>
     <xs:complexType name="error">
          <xs:sequence>
               <xs:element name="rtpidatafeed" type="xs:string" minOccurs="0" maxOccurs="1"/>
               <xs:element name="vid" type="xs:string" minOccurs="0" maxOccurs="1"/>
               <xs:element name="rt" type="xs:string" minOccurs="0" maxOccurs="1"/>
               <xs:element name="msg" type="xs:string" minOccurs="1" maxOccurs="1"/>
          </xs:sequence>
     </xs:complexType>
     <xs:complexType name="vehicle">
          <xs:sequence>
               <xs:element name="vid" type="xs:string" minOccurs="1" maxOccurs="1"/>
               <xs:element name="rtpidatafeed" type="xs:string" minOccurs="0" maxOccurs="1"/>
               <xs:element name="tmpstmp" type="xs:string" minOccurs="1" maxOccurs="1"/>
               <xs:element name="lat" type="xs:double" minOccurs="1" maxOccurs="1"/>
               <xs:element name="lon" type="xs:double" minOccurs="1" maxOccurs="1"/>
               <xs:element name="hdg" type="xs:int" minOccurs="1" maxOccurs="1"/>
               <xs:element name="pid" type="xs:int" minOccurs="1" maxOccurs="1"/>
               <xs:element name="rt" type="xs:string" minOccurs="1" maxOccurs="1"/>
               <xs:element name="des" type="xs:string" minOccurs="1" maxOccurs="1"/>
               <xs:element name="pdist" type="xs:int" minOccurs="1" maxOccurs="1"/>
```

```
                <xs:element name="stopstatus" type="xs:byte" minOccurs="0" maxOccurs="1"/>
                <xs:element name="timepointid" type="xs:int" minOccurs="0" maxOccurs="1"/>
                <xs:element name="stopid" type="xs:string" minOccurs="0" maxOccurs="1"/>
                <xs:element name="sequence" type="xs:int" minOccurs="0" maxOccurs="1"/>
                <xs:element name="gtfsseq" type="xs:int" minOccurs="0" maxOccurs="1"/>
                <xs:element name="dly" type="xs:boolean" minOccurs="1" maxOccurs="1"/>
                <xs:element name="srvtmstmp" type="xs:string" minOccurs="0" maxOccurs="1"/>
                <xs:element name="spd" type="xs:int" minOccurs="1" maxOccurs="1"/>
                <xs:element name="blk" type="xs:int" minOccurs="0" maxOccurs="1"/>
                <xs:element name="tablockid" type="xs:string" minOccurs="1" maxOccurs="1"/>
                <xs:element name="tatripid" type="xs:string" minOccurs="1" maxOccurs="1"/>
                <xs:element name="origtatripno" type="xs:string" minOccurs="1" maxOccurs="1"/>
                <xs:element name="zone" type="xs:string" minOccurs="1" maxOccurs="1"/>
                <xs:element name="mode" type="xs:byte" minOccurs="1" maxOccurs="1"/>
                <xs:element name="psgld" type="xs:string" minOccurs="1" maxOccurs="1"/>
                <xs:element name="stst" type="xs:int" minOccurs="0" maxOccurs="1"/>
                <xs:element name="stsd" type="xs:string" minOccurs="0" maxOccurs="1"/>
           </xs:sequence>
```

Example

Request:

https://www.ctabustracker.com/bustime/api/v3/getvehicles?key=89dj2he89d8j3j3ksjhdue93j&vid=509,392

Response:

```
<?xml version="1.0"?>
<bustime-response>
      <vehicle>
           <vid>509</vid>
           <tmstmp>20200308 10:28</tmstmp>
           <lat>41.92124938964844</lat>
           <lon>-87.64849853515625</lon>
           <hdg>358</hdg>
           <pid>3630</pid>
           <pdist>5678</pdist>
           <rt>8</rt>
           <des>Waveland/Broadway</des>
           <spd>27</spd>
           <tablockid>2 -701</tablockid>
           <tatripid>108</tatripid>
           <origtatripno>ME_ME403_V1_AA</origtatripno>
           <zone>Bay 1</zone>
           <mode>1</mode>
           <psgld>EMPTY</psgld>
           <stst>37560</stst>
           <stsd>2020-03-08</stsd>
      </vehicle>
      <vehicle>
           <vid>392</vid>
           <tmstmp>20200308 10:28</tmstmp>
           <lat>41.91095733642578</lat>
           <lon-87.64120713719782</lon>
           <hdg>88</hdg>
           <pid>1519</pid>
           <pdist>11203</pdist>
           <rt>72</rt>
           <des>Clark</des>
           <spd>36</spd>
           <tablockid>3 -703</tablockid>
           <tatripid>108156</tatripid>
           <origtatripno>ME_ME403_V1_AA</origtatripno>
           <zone>Bay 1</zone>
           <mode>1</mode>
           <psgld>FULL</psgld>
           <stst>36900</stst>
           <stsd>2020-03-08</stsd>
      </vehicle>
</bustime-response>
```

Request:

https://ctabustracker.com/bustime/api/v3/getvehicles?key=89dj2he89d8j3j3ksjhdue93j&vid=6438,1295&tmres=s&format=json

Response:

```
{
      "bustime-response": {
            "vehicle": [
                  {
                        "vid": "1",
                        "tmstmp": "20200307 13:14",
                        "lat": "37.54381",
                        "lon": "-77.43878166666667",
                        "hdg": "308",
                        "pid": 1689,
                        "rt": "6",
                        "des": "BROAD WILLOW LAWN",
                        "pdist": 3481,
                        "dly": false,
                        "spd": 3,
                        "tatripid": "12",
                        "tablockid": "6-05",
                        "origtatripno": "ME_ME403_V1_AA",
                        "zone": "",
                        "mode": 1,
                        "psgld": "EMPTY",
                        "stst": 47520,
                        "stsd": "2020-03-07”
                  },
                  {
                        "vid": "2",
                        "tmstmp": "20200307 13:14",
                        "lat": "37.55896532837857",
                        "lon": "-77.48567781754004",
                        "hdg": "294",
                        "pid": 1559,
                        "rt": "16",
                        "des": "GROVE BF",
                        "pdist": 20156,
                        "dly": false,
                        "spd": 5,
                        "tatripid": "12",
                        "tablockid": "16-02",
                        "origtatripno": "ME_ME403_V1_AA",
                        "zone": "",
                        "mode": 1,
                        "psgld": "FULL",
                        "stst": 46140,
                        "stsd": "2020-03-07"
                  }
            ]
      }
}
```

Routes

Base URL: https://www.ctabustracker.com/bustime/api/v3/getroutes

Parameters:

Name	Value	Description
key	string (required)	25-digit Bus Tracker API access key.

Response

A well-formed XML or JSON document will be returned as a response to **getroutes**.

Response Fields:

Name	Description
bustime-response	Root element of the response document.
error	Child element of the root element. Message if the processing of the request resulted in an error.
route JSON Array: **routes**	Child element of the root element. Encapsulates a route serviced by the system.
rt	Child element of the **route** element. Alphanumeric designator of a route (ex. “20” or “X20”).
rtnm	Child element of the **route** element. Common name of the route (ex. “Madison” for the 20 route).
rtclr	Child element of the **route** element. Color of the route line used in map (ex. "#ffffff")
rtdd	Child element of the **route** element. Language-specific route designator meant for display.

Remarks

Use the **getroutes** request to retrieve the set of routes serviced by the system.

XML Schema

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="bustime-response" type="bustime-response"/>
      <xs:complexType name="bustime-response">
            <xs:sequence>
                  <xs:element name="error" type="error" minOccurs="0" maxOccurs="unbounded"/>
                  <xs:element name="route" type="route" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="error">
            <xs:sequence>
                  <xs:element name="rtpidatafeed" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="msg" type="xs:string" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="route">
            <xs:sequence>
                  <xs:element name="rt" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="rtnm" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="rtclr" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="rtdd" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="rtpidatafeed" type="xs:string" minOccurs="0" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
</xs:schema>
```

Example

Request:

https://www.ctabustracker.com/bustime/api/v3/getroutes?key=89dj2he89d8j3j3ksjhdue93j

Response:

```
<?xml version=”1.0”?>
<bustime-response>
      <route>
            <rt>1</rt>
            <rtnm>Indiana/Hyde Park</rtnm>
            <rtclr>#000000</rtclr>
            <rtdd>1</rtdd>
      </route>
      <route>
            <rt>2</rt>
            <rtnm>Hyde Park Express</rtnm>
            <rtclr>#dc78af</rtclr>
            <rtdd>2</rtdd>
      </route>
      <route>
            <rt>3</rt>
            <rtnm>King Drive</rtnm>
            <rtclr>#ff0000</rtclr>
            <rtdd>3</rtdd>
      </route>
      <route>
            <rt>X3</rt>
            <rtnm>King Drive Express</rtnm>
            <rtclr>#ffffff</rtclr>
            <rtdd>X3</rtdd>
      </route>
      ...
</bustime-response>
```

Request

https://ctabustracker.com/bustime/api/v3/getroutes?key=89dj2he89d8j3j3ksjhdue93j&format=json

Response

```
{
      "bustime-response": {
            "routes": [
                  {
                        "rt": "1",
                        "rtnm": "Pontiac – Dhu Varren
                        "rtdd": "1",
                        "rtclr": "#ffffff"
                  },
                  {
                        "rt": "2",
                        "rtnm": "Pontiac - University",
                        "rtdd": "1",
                        "rtclr": "#dc78af"
                  },
                  ...
            ]
      }
}
```

Route Directions

Base URL: https://www.ctabustracker.com/bustime/api/v3/getdirections

Parameters:

Name	Value	Description
key	string (required)	25-digit Bus Tracker API access key.
rt	Single route designator (required)	Alphanumeric designator of a route (ex. “20” or “X20”) for which a list of available directions is to be returned.

Response

A well-formed XML or JSON document will be returned as a response to **getdirections**.

Response Fields:

Name	Description
bustime-response	Root element of the response document.
error	Child element of the root element. Message if the processing of the request resulted in an error.
dir JSON Array: **directions**	Child element of the root element. Encapsulates a route’s direction serviced by the system.
id	Child element of the **dir** element. This is the direction designator that should be used in other requests such as getpredictions.
name	Child element of the **dir** element. This is the human-readable, locale-dependent name of the direction.

Remarks

Use the **getdirections** request to retrieve the set of directions serviced by the specified route.

XML Schema

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
<xs:element name="bustime-response" type="bustime-response"/>
      <xs:complexType name="bustime-response">
            <xs:sequence>
```

```
                <xs:element name="error" type="error" minOccurs="0" maxOccurs="unbounded"/>
                <xs:element name="dir" type="dir" minOccurs="0" maxOccurs="unbounded"/>
     </xs:sequence>
     </xs:complexType>
     <xs:complexType name="error">
           <xs:sequence>
                <xs:element name="rtpidatafeed" type="xs:string" minOccurs="0" maxOccurs="1"/>
                <xs:element name="rt" type="xs:string" minOccurs="0" maxOccurs="1"/>
                <xs:element name="msg" type="xs:string" minOccurs="1" maxOccurs="1"/>
           </xs:sequence>
     </xs:complexType>
     <xs:complexType name="dir">
           <xs:sequence>
                <xs:element name="id" type="xs:string" minOccurs="1" maxOccurs="1"/>
                <xs:element name="name" type="xs:string" minOccurs="1" maxOccurs="1"/>
           </xs:sequence>
     </xs:complexType>
</xs:schema>
```

Example

Request

https://www.ctabustracker.com/bustime/api/v3/getdirections?key=89dj2he89d8j3j3ksjhdue93j&rt=20

Response

```
<?xml version="1.0"?>
<bustime-response>
    <dir>
        <id>Eastbound</id>
        <name>Eastbound</name>
    </dir>
    <dir>
        <id>Westbound</id>
        <name>Westbound</name>
    </dir>
</bustime-response>
```

Request

https://ctabustracker.com/bustime/api/v3/getdirections?key=89dj2he89d8j3j3ksjhdue93j&rt=20&format=json

Response

```
{
    "bustime-response": {
        "directions": [
            {
                "id": "Eastbound",
                "name": "Eastbound"
            },
            {
                "id": "Westbound",
                "name": "Westbound"
            }
        ]
    }
}
```

Stops

Base URL: https://www.ctabustracker.com/bustime/api/v3/getstops

Parameters:

Name	Value	Description
key	string (required)	25-digit Bus Tracker API access key.
rt	Single route designator (required if stpid is not provided)	Alphanumeric designator of the route (ex. “20” or “X20”) for which a list of available stops is to be returned.
dir	Single route direction (required if stpid is not provided)	Direction of the route (ex. “East Bound”) for which a list of available stops is to be returned. This needs to match the direction’s id in the **getdirections** call.
stpid	comma-delimited list of stop ids (required if rt and dir are not provided)	Numeric ID number for a specific stop (ex. "305") for which a single stop is to be returned. Can send up to 10 stop parameters.

Response

A well-formed XML or JSON document will be returned as a response to **getstops**.

Response Fields:

Name	Description
bustime-response	Root element of the response document.
error	Child element of the root element. Message if the processing of the request resulted in an error.
stop JSON Array: **stops**	Child element of the root element. Encapsulates all descriptive information about a particular stop.
stpid	Child element of the **stop** element. Unique identifier representing this stop.
stpnm	Child element of the **stop** element. Display name of this stop (ex. “Madison and Clark”)
lat	Child element of the **stop** element. Latitude position of the stop in decimal degrees (WGS 84).

lon	Child element of the **stop** element. Longitude position of the stop in decimal degrees (WGS 84).
dtradd	Child element of the **stop** element. A list of detour ids which add (temporarily service) this stop.
dtrrem	Child element of the **stop** element. A list of detour ids which remove (detour around) this stop.
ada	Child element of the **stop** element. Possible values are true or false, true indicating that the stop is ADA Accessible. Only included if supplied by the TA.

Remarks

Use the **getstops** request to retrieve the set of stops for the specified route and direction. A request must provide either a **rt & dir** or up to 10 **stpid**s, but not both.

Stop lists are only available for a valid route/direction pair. In other words, a list of all stops that service a particular route (regardless of direction) cannot be requested.

If a stop is affected by a detour, the detour's ID will appear in **dtradd** (if it was added to the pattern) or **dtrrem** (if it has been detoured around). The application can use the **getdetours** call to show relevant information about the detour to the end user.

XML Schema

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:element name="bustime-response" type="bustime-response"/>
      <xs:complexType name="bustime-response">
            <xs:sequence>
                  <xs:element name="error" type="xs:string" minOccurs="0" maxOccurs="unbounded"/>
                  <xs:element name="stop" type="stop" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="error">
            <xs:sequence>
                  <xs:element name="rtpidatafeed" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="rt" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="dir" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="msg" type="xs:string" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="stop">
            <xs:sequence>
                  <xs:element name="stpid" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="stpnm" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="lat" type="xs:double" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="lon" type="xs:double" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="dtradd" type="xs:int" minOccurs="0" maxOccurs="unbounded"/>
                  <xs:element name="dtrrem" type="xs:int" minOccurs="0" maxOccurs="unbounded"/>
                  <xs:element name="gtfsseq" type="xs:int" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="ada" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
</xs:schema>
```

Example

Request

https://www.ctabustracker.com/bustime/api/v3/getstops?key=89dj2he89d8j3j3ksjhdue93j&rt=20&dir=Eastbound

Response

```
<?xml version=”1.0”?>
<bustime-response>
      <stop>
            <stpid>4727</stpid>
            <stpnm>1633 W Madison</stpnm>
            <lat>41.881265</lat>
            <lon>-87.66849</lon>
      </stop>
      <stop>
            <stpid>9604</stpid>
            <stpnm>Austin & Pleasant/Fulton</stpnm>
            <lat>41.885206667</lat>
            <lon>-87.7748733333333</lon>
      </stop>
      <stop>
            <stpid>9605</stpid>
            <stpnm>Austin & Randolph/West End</stpnm>
            <lon>41.8838633333333</lon>
            <lat>-87.7748566666667</lat>
      </stop>
      <stop>
            <stpid>9603</stpid>
            <stpnm>Austin & South Blvd/Corcoran</stpnm>
            <lat>41.886908333</lat>
            <lon>-87.77493667</lon>
      </stop>
</bustime-response>
```

Request

https://ctabustracker.com/bustime/api/v3/getstops?key=89dj2he89d8j3j3ksjhdue93j&rt=20&dir=Eastbound&format=json

Response

```
{
      "bustime-response": {
            "stops": [
                  {
                        "stpid": "1577",
                        "stpnm": "1509 S Michigan",
                        "lat": 41.861706666665,
                        "lon": -87.623969999999
                  },
                  {
                        "stpid": "1564",
                        "stpnm": "3000 S Michigan",
                        "lat": 41.840606666667,
                        "lon": -87.623206666667
                        "dtrrem": [
                              "BFC46F62-990F-4AB4-A85C-3AF84574EC99",
                              "50C633C7-0891-4E5A-83A8-FF0C6214BF69"
                              ]
                  },
                  ...
            ]
      }
}
```

Patterns

Base URL: https://www.ctabustracker.com/bustime/api/v3/getpatterns

Parameters:

Name	Value	Description
key	string (required)	25-digit Bus Tracker API access key.
pid	comma-delimited list of pattern IDs (not available with **rt** parameter)	Set of one or more pattern IDs whose points should be returned. For example: 56,436,1221 will return points from three (3) patterns. A maximum of 10 identifiers can be specified.
rt	Single route designator (not available with **pid** parameter)	Route designator for which all active patterns should be returned.

Response

A well-formed XML or JSON document will be returned as a response to **getpatterns**.

Response Fields:

Name	Description
bustime-response	Root element of the response document.
error	Child element of the root element. Message if the processing of the request resulted in an error.
ptr	Child element of the root element. Encapsulates a set of points which define a pattern.
pid	Child element of the **ptr** element. ID of pattern.
ln	Child element of the **ptr** element. Length of the pattern in feet.
rtdir	Child element of the **ptr** element. Direction that is valid for the specified route designator. For example, “Eastbound”. This needs to match the direction id seen in the getdirections call.
pt	Child element of the **ptr** element. Child element of the root element. Encapsulates one a set of geo-positional points (including stops) that when connected define a pattern.

seq	Child element of the **pt** element. Position of this point in the overall sequence of points.
typ	Child element of the **pt** element. ‘S’ if the point represents a Stop, ‘W’ if the point represents a waypoint along the route.
stpid	Child element of the **pt** element. If the point represents a stop, the unique identifier of the stop.
stpnm	Child element of the **pt** element. If the point represents a stop, the display name of the stop.
pdist	Child element of the **pt** element. If the point represents a stop, the linear distance of this point (feet) into the requested pattern.
lat	Child element of the **pt** element. Latitude position of the point in decimal degrees (WGS 84).
lon	Child element of the **pt** element. Longitude position of the point in decimal degrees (WGS 84).
dtrid	Child element of the **ptr** element. If this pattern was created by a detour, contains the id of the detour. Does not appear for normal patterns.
dtrpt	Child element of the **ptr** element. If this pattern was created by a detour, encapsulates a set of geo-positional points that represent the *original* pattern. Useful for drawing dashed lines on a map.

Remarks

Use the **getpatterns** request to retrieve the set of geo-positional points and stops that when connected can be used to construct the geo-positional layout of a pattern (i.e., route variation).

Use **pid** to specify one or more identifiers of patterns whose points are to be returned. A maximum of 10 patterns can be specified.

Use **rt** to specify a route identifier where all active patterns are returned. The set of active patterns returned includes: one or more patterns marked as “default” patterns for the specified route and all patterns that are currently being executed by at least one vehicle on the specified route.

Note: The **pid** and **rt** parameters cannot be combined in one request. If both parameters are specified on a request to **getpatterns,** only the first parameter specified on the request will be processed.

XML Schema

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:element name="bustime-response" type="bustime-response"/>
      <xs:complexType name="bustime-response">
            <xs:sequence>
                  <xs:element name="error" type="error" minOccurs="0" maxOccurs="unbounded"/>
                  <xs:element name="ptr” type="ptr" minOccurs="0" maxOccurs="10"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="error">
            <xs:sequence>
                  <xs:element name="rtpidatafeed" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="pid" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="rt" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="msg" type="xs:string" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="ptr">
            <xs:element name="pid" type="xs:int" minOccurs="1" maxOccurs="1"/>
            <xs:element name="ln" type="xs:int" minOccurs="1" maxOccurs="1"/>
            <xs:element name="rtdir" type="xs:string" minOccurs="1" maxOccurs="1"/>
            <xs:element name="pt" type="pt" minOccurs="1" maxOccurs="unbounded"/>
            <xs:element name="dtrid" type="xs:string" minOccurs="0" maxOccurs="1"/>
            <xs:element name="dtrpt" type="pt" minOccurs="0" maxOccurs="unbounded"/>
      </xs:complexType>
      <xs:complexType name="pt">
            <xs:sequence>
                  <xs:element name="seq" type="xs:int" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="typ" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="stpid" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="stpnm" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="pdist" type="xs:float" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="lat" type="xs:double" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="lon" type="xs:double" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
</xs:schema>
```

Example

Request

https://www.ctabustracker.com/bustime/api/v3/getpatterns?key=89dj2he89d8j3j3ksjhdue93j&pid=954

Response

```
<?xml version=”1.0”?>
<bustime-response>
      <ptr>
            <pid>954</pid>
            <ln>41113.0</ln>
            <rtdir>Westbound</rtdir>
            <pt>
                  <seq>1</seq>
                  <lat>41.882155</lat>
                  <lon>-87.625147000001</lon>
                  <typ>W</typ>
            </pt>
            <pt>
                  <seq>2</seq>
                  <lat>41.88211</lat>
                  <lon>-87.62515</lon>
                  <typ>W</typ>
            </pt>
            <pt>
                  <seq>3</seq>
                  <lat>41.88211</lat>
                  <lon>-87.62544</lon>
                  <typ>W</typ>
            </pt>
            <pt>
                  <seq>4</seq>
                  <lat>41.8821</lat>
                  <lon>-87.62585</lon>
                  <typ>W</typ>
            </pt>
            <pt>
                  <seq>5</seq>
                  <lat>41.882129000001</lat>
                  <lon>-87.625855</lon>
                  <typ>S</typ>
                  <stpid>450</stpid>
                  <stpnm>Madison & Wabash</stpnm>
                  <pdist>206.0</pdist>
            </pt>
            ...
      </ptr>
</bustime-response>
```

Request

https://ctabustracker.com/bustime/api/v3/getpatterns?key=89dj2he89d8j3j3ksjhdue93j&pid=954&format=json

Response

```
{
      "bustime-response": {
            "ptr": [
                  {
                        "pid": 954,
                        "ln": 41113.0,
                        "rtdir": "Westbound",
                        "pt": [
                              {
                                    "seq": 1,
                                    "lat": 41.882155,
                                    "lon": -87.625147000001,
                                    "typ": "W",
                                    "pdist": 0.0
                              },
                              {
                                    "seq": 2,
                                    "lat": 41.88211,
                                    "lon": -87.62515,
                                    "typ": "W",
                                    "pdist": 0.0
                              },
                              {
                                    "seq": 3,
                                    "lat": 41.88211,
                                    "lon": -87.62544,
                                    "typ": "W",
                                    "pdist": 0.0
                              },
                              {
                                    "seq": 4,
                                    "lat": 41.8821,
                                    "lon": -87.62585,
                                    "typ": "W",
                                    "pdist": 0.0
                              },
                              {
                                    "seq": 5,
                                    "lat": 41.882129000001,
                                    "lon": -87.625855,
                                    "typ": "S",
                                    "stpid": "450",
                                    "stpnm": "Madison & Wabash",
                                    "pdist": 206.0
                              },
                              ...
                  ]
            }
      ]
}
```

Predictions

Base URL: https://www.ctabustracker.com/bustime/api/v3/getpredictions

Parameters:

Name	Value	Description
key	string (required)	25-digit Bus Tracker API access key.
stpid	comma-delimited list of stop IDs (not available with **vid** parameter)	Set of one or more stop IDs whose predictions are to be returned. For example: 5029,1392,2019,4367 will return predictions for the four stops. A maximum of 10 identifiers can be specified.
rt	comma-delimited list of route designators (optional, available with **stpid** parameter)	Set of one or more route designators for which matching predictions are to be returned.
vid	comma-delimited list of vehicle IDs (not available with **stpid** parameter)	Set of one or more vehicle IDs whose predictions should be returned. For example: 509,392,201,4367 will return predictions for four vehicles. A maximum of 10 identifiers can be specified.
top	number (optional)	Maximum number of predictions to be returned.
tmres	string(optional)	Resolution of time stamps. Set to “s” to get time resolution to the second. Set to “m” to get time resolution to the minute. If omitted, defaults to “m”. Date and time is represented in the following format: If specified as “s” YYYYMMDD HH:MM:SS If specified as “m” YYYYMMDD HH:MM Month is represented as two digits where January is equal to “01” and December is equal to “12”. Time is represented using a 24-hour clock.
unixTime	boolean (optional)	***True*** if timestamps should be provided as Unix times (milliseconds that have elapsed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970). Default is ***false***.

Response

A well-formed XML or JSON document will be returned as a response to **getpredictions**.

Response Fields:

Name	Description
bustime-response	Root element of the response document.
error	Child element of the root element. Message if the processing of the request resulted in an error.
prd	Child element of the root element. Encapsulates a predicted arrival or departure time for the specified set of stops or vehicles.
tmstmp	Child element of the **prd** element. Date and time (local) the prediction was generated. Date and time is represented based on the **tmres** parameter if the **unixTime** parameter is omitted or set to false. If the unixTime parameter is present and set to *true*, returns the number of milliseconds that have elapsed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970.
typ	Child element of the **prd** element. Type of prediction. 'A' for an arrival prediction (prediction of when the vehicle will arrive at this stop). 'D' for a departure prediction (prediction of when the vehicle will depart this stop, if applicable). Predictions made for first stops of a route or layovers are examples of departure predictions.
stpid	Child element of the **prd** element. Unique identifier representing the stop for which this prediction was generated.
stpnm	Child element of the **prd** element. Display name of the stop for which this prediction was generated.
vid	Child element of the **prd** element. Unique ID of the vehicle for which this prediction was generated.
dstp	Child element of the **prd** element. Linear distance (feet) left to be traveled by the vehicle before it reaches the stop associated with this prediction.
rt	Child element of the **prd** element. Alphanumeric designator of the route (ex. "20" or "X20") for which this prediction was generated.
rtdd	Child element of the **prd** element. Language-specific route designator meant for display.
rtdir	Child element of the **prd** element. Direction of travel of the route associated with this prediction (ex. "Eastbound").
des	Child element of the **prd** element. Final destination of the vehicle associated with this prediction.

prdtm	Child element of the **prd** element. Predicted date and time (local) of a vehicle's arrival or departure to the stop associated with this prediction. Date and time is represented based on the **tmres** parameter if the **unixTime** parameter is omitted or set to false. If the unixTime parameter is present and set to *true*, returns the number of milliseconds that have elapsed since 00:00:00 Coordinated Universal Time (UTC), Thursday, 1 January 1970.
dly	Child element of the **prd** element. "true" if the vehicle is delayed. In version 3 this element is always present.
dyn	Child element of the **prd** element. The dynamic action type affecting this prediction. See the "Dynamic Action Types" section for a list of possible identifiers.
tablockid	Child element of the **prd** element. TA's version of the scheduled block identifier for the work currently being performed by the vehicle.
tatripid	Child element of the **prd** element. TA's version of the scheduled trip identifier for the vehicle's current trip.
origtatripno	Child element of the **prd** element. Trip ID defined by the TA scheduling system.
prdctdn	Child element of the **prd** element. This is the time left, in minutes, until the bus arrives at this stop.
zone	Child element of the **prd** element. The zone name if the vehicle has entered a defined zones, otherwise blank.
psgld	Child element of the **prd** element. String representing the ratio of the current passenger count to the vehicle's total capacity. Possible values include "FULL", "HALF_EMPTY", "EMPTY" and "N/A". Ratios for "FULL", "HALF_EMPTY" and "EMPTY" are determined by the transit agency. "N/A" indicates that the passenger load is unknown.
stst	Child element of the prd element. Contains the time (in seconds past midnight) of the scheduled start of the trip.
stsd	Child element of the prd element. Contains the date (in "yyyy-mm-dd" format) of the scheduled start of the trip.
flagstop	Child element of the prd element. An integer code representing the flag-stop information for the prediction. -1 = UNDEFINED (no flag-stop information available) 0 = NORMAL (normal stop) 1 = PICKUP_AND_DISCHARGE (Flag stop for both pickup and discharge) 2 = ONLY_DISCHARGE (Flag stop for discharge only)

Remarks

Use the **getpredictions** request to retrieve predictions for one or more stops or one or more vehicles. Predictions are always returned in ascending order according to **prdtm**.

Use the **vid** parameter to retrieve predictions for one or more vehicles currently being tracked. A maximum of 10 vehicles can be specified.

Use the **stpid** parameter to retrieve predictions for one or more stops. A maximum of 10 stops can be specified.

> Note: The **vid** and **stpid** parameters cannot be combined in one request. If both parameters are specified on a request to **getpredictions,** only the first parameter specified on the request will be processed.

Calls to **getpredictions** without specifying the **vid** or **stpid** parameter is not allowed.

Use the **top** parameter to specify the maximum number of predictions to return. If **top** is not specified, then all predictions matching the specified parameters will be returned.

If canceled stops are not configured to be displayed to the public, predictions for them will not be included in the **getpredictions** response. If expressed stops are not configured to not be displayed to the public, predictions for them will not be included in the **getpredictions** response.

XML Schema

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:element name="bustime-response" type="bustime-response"/>
      <xs:complexType name="bustime-response">
            <xs:sequence>
                  <xs:element name="error" type="error" minOccurs="0" maxOccurs="unbounded"/>
                  <xs:element name="prd" type="prediction" minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="error">
            <xs:sequence>
                  <xs:element name="rtpidatafeed" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="stpid" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="vid" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="msg" type="xs:string" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="prediction">
            <xs:all>
                  <xs:element name="tmstmp" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="typ" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="stpid" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="stpnm" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="vid" type="xs:int" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="dstp" type="xs:int" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="rt" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="rtdd" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="rtdir" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="des" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="prdtm" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="dly" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="dyn" type="xs:byte" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="tablockid" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="tatripid" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="origtatripno" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="zone" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="psgld" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="gtfsseq" type="xs:int" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="nbus" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="stst" type="xs:int" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="stsd" type="xs:string" minOccurs="0" maxOccurs="1"/>
                  <xs:element name="flagstop" type="xs:int" minOccurs="1" maxOccurs="1"/>
```

```
          </xs:all>
     </xs:complexType>
</xs:schema>
```

Example

Request:

https://www.ctabustracker.com/bustime/api/v3/getpredictions?key=89dj2he89d8j3j3ksjhdue93j&rt=20&stpid=456

Response:

```
<?xml version=”1.0”?>
<bustime-response>
    <prd>
        <tmstmp>20250421 16:03</tmstmp>
        <typ>A</typ>
        <stpnm>Madison &amp; Jefferson</stpnm>
        <stpid>456</stpid>
        <vid>8184</vid>
        <dstp>1806</dstp>
        <rt>20</rt>
        <rtdd>20</rtdd>
        <rtdir>Westbound</rtdir>
        <des>Austin</des>
        <prdtm>20250421 16:07</prdtm>
        <dly>false</dly>
        <dyn>0</dyn>
        <tablockid>20 -803</tablockid>
        <tatripid>1040713</tatripid>
        <origtatripno>262522629</origtatripno>
        <prdctdn>3</prdctdn>
        <zone></zone>
        <psgld></psgld>
        <stst>57120</stst>
        <stsd>2025-04-21</stsd>
        <flagstop>0</flagstop>
    </prd>
    <prd>
        <tmstmp>20250421 16:03</tmstmp>
        <typ>A</typ>
        <stpnm>Madison &amp; Jefferson</stpnm>
        <stpid>456</stpid>
        <vid>8140</vid>
        <dstp>5703</dstp>
        <rt>20</rt>
        <rtdd>20</rtdd>
        <rtdir>Westbound</rtdir>
        <des>Austin</des>
        <prdtm>20250421 16:14</prdtm>
        <dly>false</dly>
        <dyn>0</dyn>
        <tablockid>20 -853</tablockid>
        <tatripid>1040714</tatripid>
        <origtatripno>262522630</origtatripno>
        <prdctdn>10</prdctdn>
        <zone></zone>
        <psgld></psgld>
        <stst>57540</stst>
        <stsd>2025-04-21</stsd>
        <flagstop>0</flagstop>
    </prd>
</bustime-response>
```

Request

https://ctabustracker.com/bustime/api/v3/getpredictions?key=89dj2he89d8j3j3ksjhdue93j&rt=20&stpid=456&format=json

Response

```
{
    "bustime-response": {
        "prd": [
            {
                "tmstmp": "20250421 16:04",
                "typ": "A",
                "stpnm": "Madison & Jefferson",
                "stpid": "456",
                "vid": "8184",
                "dstp": 686,
                "rt": "20",
                "rtdd": "20",
                "rtdir": "Westbound",
                "des": "Austin",
                "prdtm": "20250421 16:06",
                "tablockid": "20 -803",
                "tatripid": "1040713",
                "origtatripno": "262522629",
                "dly": false,
                "dyn": 0,
                "prdctdn": "DUE",
                "zone": "",
                "psgld": "",
                "stst": 57120,
                "stsd": "2025-04-21",
                "flagstop": 0
            },
            ...
        ]
    }
}
```

Locales

Base URL: https://www.ctabustracker.com/bustime/api/v3/getlocalelist

Parameters:

Name	Value	Description
key	string (required)	25-digit Bus Tracker API access key.
locale	string(optional)	The language to use for the response. Must match a supported locale id – See **localestring** below
inLocaleLanguage	boolean (optional)	Gets each locale with their display names in the native language of the locale when true. If omitted, defaults to false. When true, this parameter takes precedence over the setting of the 'locale' parameter.

Response

A well-formed XML or JSON document will be returned as a response to **getlocalelist**.

Response Fields:

Name	Description
bustime-response	Root element of the response document.
error	Child element of the root element. Message if the processing of the request resulted in an error.
locale	Child element of the root element. Encapsulates all data about a locale (language).
localestring	Child element of the **locale** element. Unique name/identifier of the locale. This is what is passed as the locale parameter in all API calls. The **localestring** contains an ISO 639 language code. Examples are "es".
displayname	Child element of the **locale** element. The name of the language. If the locale parameter was included, then this will be in that language. For human-readable use only. If the inLocaleLanguage parameter was true, then this will be in the language of the locale that it represents.

Remarks

Use the **getlocalelist** to get a list of what languages can be used as the locale parameter. It can be called a second time with a locale parameter that matches one of the previously returned localestrings to see the human-readable language names in that given language.

> Note: The locale parameter in all requests is meant to match values in this list, but it does support the inheritance model of Java Locale. If the given language is not supported then the default language of the Transit Authority is used. No indication of which language used is given in the response, so it is best to use a locale string out of the list returned by **getlocalelist**.

XML Schema

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:element name=”bustime-response” type=”bustime-response”/>
      <xs:complexType name="bustime-response">
            <xs:sequence>
                  <xs:element name="error" type="error" minOccurs=”0” maxOccurs=”unbounded”/>
                  <xs:element name="locale" type="locale" minOccurs="1" maxOccurs="unbounded"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="error">
            <xs:sequence>
                  <xs:element name="msg" type="xs:string" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="locale">
            <xs:sequence>
                  <xs:element name="localestring" type="xs:string" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="displayname" type="xs:string" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
</xs:schema>
```

Example

Request

https://www.ctabustracker.com/bustime/api/v3/getlocalelist?key=89dj2he89d8j3j3ksjhdue93j

Response

```
<?xml version=”1.0”?>
<bustime-response>
      <locale>
            <localestring>en</localestring>
            <displayname>English</displayname>
      </locale>
      <locale>
            <localestring>es</localestring>
            <displayname>Spanish</displayname>
      </locale>
</bustime-response>
```

Request

https://www.ctabustracker.com/bustime/api/v3/getlocalelist?key=89dj2he89d8j3j3ksjhdue93j&locale=es

Response

```
<?xml version=”1.0”?>
<bustime-response>
      <locale>
            <localestring>en</localestring>
            <displayname>inglés</displayname>
      </locale>
      <locale>
            <localestring>es</localestring>
            <displayname>español</displayname>
      </locale>
</bustime-response>
```

Request

https://www.ctabustracker.com/bustime/api/v3/getlocalelist?key=89dj2he89d8j3j3ksjhdue93j&inLocaleLanguage

Response

```
<?xml version=”1.0”?>
<bustime-response>
      <locale>
            <localestring>en</localestring>
            <displayname>English</displayname>
      </locale>
      <locale>
            <localestring>es</localestring>
            <displayname>español</displayname>
      </locale>
</bustime-response>
```

Detours

Base URL: https://www.ctabustracker.com/bustime/api/v3/getdetours

Parameters:

Name	Value	Description
key	string (required)	25-digit Bus Tracker API access key.
rt	route designator (optional)	Alphanumeric designator of the route (ex.“20” or “X20”) for which a list of detours is to be returned.
rtdir	route direction (optional)	Direction of travel of the route specified in the **rt** parameter. The **rt** parameter is required when using the **rtdir** parameter. This needs to match the direction id seen in the getdirections call.

Response

A well-formed XML or JSON document will be returned as a response to **getdetours**.

Response Fields:

Name	Description
bustime-response	Root element of the response document.
error	Child element of the root element. Message if the processing of the request resulted in an error.
dtr JSON Array: **dtrs**	Child element of the root element. Encapsulates data about a detour.
id	Child element of the **dtr** element. The unique id of the detour. Other API calls reference these identifiers.
ver	Child element of the **dtr** element. The version of this detour. Only the newest version of each detour is returned.
st	Child element of the **dtr** element. The state of the detour. A value of 1 indicates the detour is active; 0 indicates a canceled detour.
desc	Child element of the **dtr** element. Description of the detour.
rtdirs	Child element of the **dtr** element. Contains a series of rtdir elements.

rtdir	Child element of the **rtdirs** element. Contains a pair of the route and direction affected by the detour.
rt	Child element of the **rtdir** element. Alphanumeric designator of a route (ex. “20” or “X20”) affected by the detour.
dir	Child element of the **rtdir** element. The direction affected by the detour.
startdt	Child element of the **dtr** element. The start date and time of this detour.
enddt	Child element of the dtr element. The end date and time of this detour.

XML Schema

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
     <xs:element name=”bustime-response” type=”bustime-response”/>
     <xs:complexType name="bustime-response">
          <xs:sequence>
               <xs:element name="dtr" maxOccurs="unbounded" minOccurs="0">
                    <xs:complexType>
                         <xs:sequence>
                              <xs:element type="xs:string" name="id" minOccurs="1" maxOccurs="1"/>
                              <xs:element type="xs:int" name="ver" minOccurs="1" maxOccurs="1"/>
                              <xs:element type="xs:int" name="st" minOccurs="1" maxOccurs="1"/>
                              <xs:element type="xs:string" name="desc" minOccurs="1" maxOccurs="1"/>
                              <xs:element name="rtdirs" minOccurs="1" maxOccurs="1">
                                   <xs:complexType>
                                        <xs:sequence>
                                             <xs:element name="rtdir" minOccurs="0" maxOccurs="unbounded">
                                                  <xs:complexType>
                                                       <xs:sequence>
                                                            <xs:element type="xs:string" name="rt" minOccurs="1"
                                                       maxOccurs="1"/>
                                                            <xs:element type="xs:string" name="dir" minOccurs="1"
                                                       maxOccurs="1"/>
                                                       </xs:sequence>
                                                  </xs:complexType>
                                             </xs:element>
                                        </xs:sequence>
                                   </xs:complexType>
                              </xs:element>
                              <xs:element type="xs:string" name="startdt" minOccurs="1" maxOccurs="1"/>
                              <xs:element type="xs:string" name="enddt" minOccurs="1" maxOccurs="1"/>
                              <xs:element type="xs:string" name="rtpidatafeed" minOccurs="0" maxOccurs="1"/>
                         </xs:sequence>
               </xs:complexType>
          </xs:element>
     </xs:sequence>
     </xs:complexType>
     <xs:complexType name="error">
          <xs:sequence>
               <xs:element name="msg" type="xs:string" minOccurs="1" maxOccurs="1"/>
               <xs:element name="rt" type="xs:string" minOccurs="0" maxOccurs="1"/>
               <xs:element name="rtdir" type="xs:string" minOccurs="0" maxOccurs="1"/>
               <xs:element name="rtpidatafeed" type="xs:string" minOccurs="0" maxOccurs="1"/>
          </xs:sequence>
     </xs:complexType>
</xs:schema>
```

Example

Request

https://www.ctabustracker.com/bustime/api/v3/getdetours?key=89dj2he89d8j3j3ksjhdue93j

Response

```
<bustime-response>
      <dtr>
            <id>84A97FD3-0741-4004-884D-0ABB22DAFA28</id>
            <ver>2</ver>
            <st>0</st>
            <desc>IVD MultiRoute detour 47</desc>
            <rtdirs>
                  <rtdir>
                        <rt>72</rt>
                        <dir>NORTHBOUND</dir>
                  </rtdir>
            </rtdirs>
            <startdt>20180404 08:45</startdt>
            <enddt>20180430 03:00</enddt>
      </dtr>
      <dtr>
            <id>329E1F2D-A848-43E9-8F90-4FB00E643786</id>
            <ver>1</ver>
            <st>1</st>
            <desc>IVD Multiroute Detour S47/62</desc>
            <rtdirs>
                  <rtdir>
                        <rt>800</rt>
                        <dir>EASTBOUND</dir>
                  </rtdir>
                  <rtdir>
                        <rt>72M</rt>
                        <dir>NORTHBOUND</dir>
                  </rtdir>
            </rtdirs>
            <startdt>20180404 09:06</startdt>
            <enddt>20180430 03:00</enddt>
      </dtr>
</bustime-response>
```

Request

https://www.ctabustracker.com/bustime/api/v3/getdetours?key=89dj2he89d8j3j3ksjhdue93j&format=json

Response

```
{
      "bustime-response": {
            "dtrs": [
                  {
                        "id": "84A97FD3-0741-4004-884D-0ABB22DAFA28",
                        "ver": 2,
                        "st": 0,
                        "desc": "IVD MultiRoute detour 47",
                        "rtdirs": [
                              {
                                    "rt": "72",
                                    "dir": "NORTHBOUND"
                              }
                        ],
                        "startdt": "20180404 08:45",
                        "enddt": "20180430 03:00"
                  },
                  {
                        "id": "329E1F2D-A848-43E9-8F90-4FB00E643786",
                        "ver": 1,
                        "st": 1,
                        "desc": "IVD Multiroute Detour S47/62",
                        "rtdirs": [
                              {
                                    "rt": "800",
                                    "dir": "EASTBOUND"
                              },
                              {
                                    "rt": "72M",
                                    "dir": "NORTHBOUND"
                              }
                        ],
                        "startdt": "20180404 09:06",
                        "enddt": "20180430 03:00"
                  }
            ]
      }
}
```

Remarks

Use the **getdetours** request to retrieve a list of active detours in the system. Detours are considered “active” if they are currently affecting the current service day, even if the start time has not yet been reached or the end time has already passed.

The response only contains metadata about the detour. The pattern data for the detour can be displayed via the **getpatterns** request when an end user selects a route(s) affected by the detour.

If a detour is canceled or expired, it will still appear in this result. This is to handle cases where a vehicle is still running a canceled or expired detour and the developer wishes to alert users that the detour is technically still in effect.

If the client application is to support detours, it is recommended that detours are requested frequently in case a new version is added or a detour is canceled. If a current detour or new version is added (or removed), the client should consider requesting new stop and pattern data for the given route/direction combination in case data has been changed by the detour.

Enhanced Detours

Base URL: https://www.ctabustracker.com/bustime/api/v3/getenhanceddetours

Parameters:

Name	Value	Description
key	string (required)	25-digit Bus Tracker API access key.

Response

A well-formed XML or JSON document will be returned as a response to **getenhanceddetours**.

Response Fields:

Name	Description
bustime-response	Root element of the response document.
error	Child element of the root element. Message if the processing of the request resulted in an error.
dtr JSON Array: **dtrs**	Child element of the root element. Encapsulates data about a detour.
id	Child element of the **dtr** element. The unique id of the detour. Other API calls reference these identifiers.
ver	Child element of the **dtr** element. The version of this detour. Only the newest version of each detour is returned.
st	Child element of the **dtr** element. The state of the detour. A value of 1 indicates the detour is active; 0 indicates a canceled detour.
desc	Child element of the **dtr** element. Description of the detour.
rtdirs	Child element of the **dtr** element. Contains a series of **rtdir** elements.
rtdir	Child element of the **rtdirs** element. Contains a pair of the route and direction affected by the detour.
rt	Child element of the **rtdir** element. Alphanumeric designator of a route (ex. “20” or “X20”) affected by the detour.
dir	Child element of the **rtdir** element. The direction affected by the detour.
startdt	Child element of the **dtr** element. The start date and time of this detour

	represented in Epoch format.
enddt	Child element of the **dtr** element. The end date and time of this detour represented in Epoch format.
moddt	Child element of the **dtr** element. The last modified date and time of this detour represented in Epoch format.
ptrs	Child element of the **dtr** element. Encloses pattern details for all patterns affected by the detour.
ptr	Child element of the **ptrs** element. Provides pattern details affected by the detour.
origpid	Child element of the **ptr** element. Original pattern ID that the detour affects.
dtrpid	Child element of the **ptr** element. Detour pattern ID.
encpl	Child element of the **ptr** element. The detour's polyline information encoded using Google's Encoded Polyline algorithm.
trips	Child element of the **ptr** element. An array of trips affected by the detour.
trip	Child element of the **dtr** element. A trip affected by the detour.
tripid	Child element of the **trip** element. Trip identifier.
tatripid	Child element of the **trip** element. TA's version of the scheduled trip identifier.
origtatripno	Child element of the **trip** element. Trip ID defined by the TA scheduling system.
dates	Child element of the **trip** element. Dates on which the trip is active between the start and end dates of the detour.
date	Child element of the **dates** element. Date on which the trip is active, between the start and end dates of the detour. Format: YYYYMMDD
stst	Child element of the **trip** element. Start time of the trip in seconds from midnight.
dtrstartstop	Child element of the **ptr** element. The stop in the pattern at which the detour starts.
dtrendstop	Child element of the **ptr** element. The stop in the pattern where the detour ends.
delay	Child element of the **ptr** element. Delay time in seconds starting from the end of detour as compared to the original schedule.
repstops	Child element of the **dtr** element. An array of stops replaced in the detour pattern by the detour in comparison to the original pattern.
repstop	Child element of the **repstops** element. A stop replaced in the detour pattern by the detour in comparison to the original pattern.
geoid	Child element of the **dtrstartstop** / **dtrendstop** / **repstop** element. Time point identifier for the stop.

stpid	Child element of the **dtrstartstop** / **dtrendstop** / **repstop** element. Stop identifier for the stop.
seq	Child element of the **dtrstartstop** / **dtrendstop** / **repstop** element. Sequence of the stop in the trip.
stpnm	Child element of the **dtrstartstop** / **dtrendstop** / **repstop** element. Name of the stop as per the locale requested.
lat	Child element of the **repstop** element. Latitude of the stop location.
lon	Child element of the **repstop** element. Longitude of the stop location.
adhoc	Child element of the **repstop** element. Boolean value determining whether the stop was added adhoc or an existing stop.
relpasstime	Child element of the **repstop** element. Time in seconds when this stop occurs in the pattern after the previous stop.
sbnm	Child element of the **dtr** element. The unique name/identifier of the service bulletin, if present for the detour.

Remarks

The **getenhanceddetours** endpoint is implemented for use by GTFS-RT system. All the detours that are currently active or are going to be active in the future are retrieved. This API endpoint only returns data when detour support is enabled in BusTime.

XML Schema

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:element name=”bustime-response” type=”bustime-response”/>
      <xs:complexType name="bustime-response">
            <xs:sequence>
                  <xs:element name="dtr" maxOccurs="unbounded" minOccurs="0">
                        <xs:complexType>
                              <xs:sequence>
                                    <xs:element type="xs:string" name="id" minOccurs="1" maxOccurs="1"/>
                                    <xs:element type="xs:int" name="ver" minOccurs="1" maxOccurs="1"/>
                                    <xs:element type="xs:int" name="st" minOccurs="1" maxOccurs="1"/>
                                    <xs:element type="xs:string" name="desc" minOccurs="1" maxOccurs="1"/>
                                    <xs:element name="rtdirs" minOccurs="1" maxOccurs="1">
                                          <xs:complexType>
                                                <xs:sequence>
                                                      <xs:element name="rtdir" minOccurs="0" maxOccurs="unbounded">
                                                            <xs:complexType>
                                                                  <xs:sequence>
                                                                        <xs:element type="xs:string" name="rt" minOccurs="1"
                                                                  maxOccurs="1"/>
                                                                        <xs:element type="xs:string" name="dir" minOccurs="1"
                                                                  maxOccurs="1"/>
                                                                  </xs:sequence>
                                                            </xs:complexType>
                                                      </xs:element>
                                                </xs:sequence>
                                          </xs:complexType>
                                    </xs:element>
                                    <xs:element type="xs:string" name="startdt" minOccurs="1" maxOccurs="1"/>
                                    <xs:element type="xs:string" name="enddt" minOccurs="1" maxOccurs="1"/>
                                    <xs:element type="xs:string" name="moddt" minOccurs="1" maxOccurs="1"/>
                                    <xs:element type="xs:string" name="rtpidatafeed" minOccurs="0" maxOccurs="1"/>
                                    <xs:element name="ptrs" maxOccurs="1" minOccurs="0">
                                          <xs:complexType>
                                                <xs:element name="ptr" type="patterninfo" maxOccurs="unbounded" minOccurs="0"/>
                                          </xs:complexType>
                                    </xs:element>
                        <xs:element type="xs:string" name="sbnm" maxOccurs="1" minOccurs="0"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
        </xs:sequence>
    </xs:complexType>

    <xs:complexType name="patterninfo">
    <xs:sequence>
        <xs:element type="xs:int" name="origpid" minOccurs="1" maxOccurs="1"/>
        <xs:element type="xs:int" name="dtrpid" minOccurs="1" maxOccurs="1"/>
        <xs:element type="xs:string" name="encpl" minOccurs="0" maxOccurs="1"/>
        <xs:element name="trips" minOccurs="0" maxOccurs="1">
        <xs:element name="trip" minOccurs="0" maxOccurs="unbounded">
            <xs:complexType>
                <xs:sequence>
                    <xs:element type="xs:int" name="tripid" minOccurs="1" maxOccurs="1"/>
                    <xs:element type="xs:string" name="tatripid" minOccurs="1" maxOccurs="1"/>
                    <xs:element type="xs:string" name="origtatripno" minOccurs="1" maxOccurs="1"/>
                    <xs:element name="dates" minOccurs="0" maxOccurs="1">
                    <xs:element type="xs:string" name="date" minOccurs="0" maxOccurs="unbounded"/>
                    </xs:element>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        </xs:element>
        <xs:element name="dtrstartstop" type="stopinfo" minOccurs="1" maxOccurs="1"/>
        <xs:element name="dtrendstop" type="stopinfo" minOccurs="0" maxOccurs="1"/>
        <xs:element name="delay" type="xs:int" minOccurs="0" maxOccurs="1"/>
        <xs:element name="repstops" minOccurs="0" maxOccurs="1">
        <xs:element name="repstop" type="stopinfo" minOccurs="0" maxOccurs="unbounded"/>
    </xs:element>
    </xs:sequence>
    </xs:complexType>

    <xs:complexType name="stopinfo">
        <xs:sequence>
            <xs:element name="geoid" type="xs:int" minOccurs="1" maxOccurs="1"/>
            <xs:element name="stpid" type="xs:string" minOccurs="1" maxOccurs="1"/>
            <xs:element name="seq" type="xs:int" minOccurs="1" maxOccurs="1"/>
            <xs:element name="stpnm" type="xs:string" minOccurs="1" maxOccurs="1"/>
            <xs:element name="lat" type="xs:double" minOccurs="0" maxOccurs="1"/>
            <xs:element name="lon" type="xs:double" minOccurs="0" maxOccurs="1"/>
            <xs:element name="adhoc" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
            <xs:element name="relpasstime" type="xs:int" minOccurs="0" maxOccurs="1"/>
        </xs:sequence>
    </xs:complexType>

    <xs:complexType name="error">
        <xs:sequence>
            <xs:element name="msg" type="xs:string" minOccurs="1" maxOccurs="1"/>
            <xs:element name="rtpidatafeed" type="xs:string" minOccurs="0" maxOccurs="1"/>
        </xs:sequence>
    </xs:complexType>
</xs:schema>
```

Example

Request

https://www.ctabustracker.com/bustime/api/v3/getenhanceddetours?key=89dj2he89d8j3j3ksjhdue93j

Response

```
<bustime-response>
    <dtr>
        <id>766CFD24-111D-40F0-9216-6E67576E603F</id>
        <ver>2</ver>
        <st>1</st>
        <desc>Detour with stops added and removed in multiple segments</desc>
```

```
<rtdirs>
    <rtdir>
        <rt>7</rt>
        <dir>WEST</dir>
    </rtdir>
</rtdirs>
<startdt>1725595200000</startdt>
<enddt>1727409600000</enddt>
<moddt>1725631728000</moddt>
<ptrs>
    <ptr>
        <origpid>2</origpid>
        <dtrpid>500117</dtrpid>
        <encpl>}fjmErnypU???Ot@@?DBnAI?@@FANpG?FNdO?`@K???J?NvQ?XK???J?DhE?^HjI?`@@hA?x@@XAfBI???H??R@l@?nB@
    |AAtDBvC?b@G???F?@hA?PPrQ?j@@l@?V@`A?z@MpAe@zBIE??HDoMrb@IE??HDoGzTKG??JF_GvUKE??JDoA|EI`AqD`NWK??VJsE~PWK
    ??VJyErQY`@{@~CKE?AJFiKl`@g@xCDZw@nGSGA@TDuCzVWG??VFcB~MOTm@nFMA??L@sEr^MA??L@{@~HFRsCjVa@I?@`@FuEz_@_@I??
    ^HOx@wAGaEV}a@PAf@@^?bC@lH?|B@bCIN?lCI?AAJ@CrGBV?|DB~C?LM?A?N?A`Dg@f
    GYnECtDDnCdAI@N?AAM|QgB@T??AUxKqAt@RZd@A^K???J?]hLM??AL@KpBJZIvEkA~ZWA?@V?oAv^YA??X@w@nSUG??TFO
    vE_@bGQdBg@~BOPw@hCeBfDCE??FF[j@AP{I`MwAjBiAjA{BbBgAbAeDnEOO@ALPkFpHu@v@u@b@qA`@gCF?W???VwCIcA@
    eBf@_BHQjBOCA?NBc@fDg@`HOlEM???L??f@@R?J@Z?lCE???D?@pAAxBBdACj@Ol@g@pAo@x@c@v@Gr@H^?BZPX@\[Pm@E
    aCf@_ARs@Fw@JQ</encpl>
        <trips>
            <trip>
                <tripid>679020</tripid>
                <tatripid>274676</tatripid>
                <origtatripno>274676</origtatripno>
                <dates>
                    <date>20240917</date>
                    <date>20240918</date>
                </dates>
                <stst>43680</stst>
            </trip>
            <trip>
                <tripid>158020</tripid>
                <tatripid>274677</tatripid>
                <origtatripno>274677</origtatripno>
                <dates>
                    <date>20240917</date>
                    <date>20240918</date>
                </dates>
                <stst>56280</stst>
            </trip>
        </trips>
        <dtrstartstop>
            <geoid>611</geoid>
            <stpid>235053</stpid>
            <seq>3</seq>
            <stpnm>SEPULVEDA BL + DOLORES ST</stpnm>
        </dtrstartstop>
        <dtrendstop>
            <geoid>240</geoid>
            <stpid>203001</stpid>
            <seq>36</seq>
            <stpnm>REDONDO BEACH PIER</stpnm>
        </dtrendstop>
        <repstops>
            <repstop>
                <geoid>783</geoid>
                <stpid>734025</stpid>
                <seq>1</seq>
                <stpnm>Avalon Bl at Sepulveda Bl</stpnm>
                <lat>33.80863199999843</lat>
                <lon>-118.26425600000013</lon>
                <adhoc>false</adhoc>
                <relpasstime>null</relpasstime>
            </repstop>
            <repstop>
                <geoid>352</geoid>
                <stpid>215027</stpid>
                <seq>20</seq>
                <stpnm>CARSON ST + MADRONA AV</stpnm>
                <lat>33.83127499999939</lat>
                <lon>-118.34460600000085</lon>
                <adhoc>false</adhoc>
```

```
                                 <relpasstime>185</relpasstime>
                            </repstop>
                      </repstops>
                 </ptr>
           </ptrs>
           <sbnm>CWDetour-766cfd24-111d-40f0-9216-6e67576e603f</sbnm>
      </dtr>
</bustime-response>
```

Request

https://www.ctabustracker.com/bustime/api/v3/getenhanceddetours?key=89dj2he89d8j3j3ksjhdue93j&format=json

Response

```
{
    "bustime-response": {
        "dtrs": [
            {
                "id": "766CFD24-111D-40F0-9216-6E67576E603F",
                "ver": 2,
                "st": 1,
                "desc": "Detour with stops added and removed in multiple segments",
                "rtdirs": [
                    {
                        "rt": "7",
                        "dir": "WEST"
                    }
                ],
                "startdt": 1725595200000,
                "enddt": 1727409600000,
                "moddt": 1725631728000,
                "ptrs": [
                    {
                        "origpid": 2,
                        "dtrpid": 500117,
                        "encpl":
"}fjmErnypU???Ot@@?DBnAI?@@FANpG?FNdO?`@K???J?NvQ?XK???J?DhE?^HjI?`@@hA?x@@XAfBI???H??R@l@?nB@|AAtDBvC?b@G???F?@hA?PPrQ?j@
@l@?V@`A?z@MpAe@zBIE??HDoMrb@IE??HDoGzTKG??JF_GvUKE??JDoA|EI`AqD`NWK??VJsE~PWK??VJyErQY`@{@~CKE?AJFiKl`@g@xCDZw@nGSGA@TDuC
zVWG??VFcB~MOTm@nFMA??L@sEr^MA??L@{@~HFRsCjVa@I?@`@FuEz_@_@I??^HOx@wAGaEV}a@PAf@@^?bC@lH?|B@bCIN?lCI?AAJ@CrGBV?|DB~C?LM?A?
N?A`Dg@fGYnECtDDnCdAI@N?AAM|QgB@T??AUxKqAt@RZd@A^K???J?]hLM??AL@KpBJZIvEkA~ZWA?@V?oAv^YA??X@w@nSUG??TFOvE_@bGQdBg@~BOPw@hC
eBfDCE??FF[j@AP{I`MwAjBiAjA{BbBgAbAeDnEOO@ALPkFpHu@v@u@b@qA`@gCF?W???VwCIcA@eBf@_BHQjBOCA?NBc@fDg@`HOlEM???L??f@@R?J@Z?lCE
???D?@pAAxBBdACj@Ol@g@pAo@x@c@v@Gr@H^?BZPX@\\[Pm@EaCf@_ARs@Fw@JQ",
                        "trips": [
                            {
                                "tripid": 679020,
                                "tatripid": "274676",
                                "origtatripno": "274676",
                                "dates": [
                                    "20240917",
                                    "20240918"
                                ],
                                "stst": 43680
                            },
                            {
                                "tripid": 158020,
                                "tatripid": "274677",
                                "origtatripno": "274677",
                                "dates": [
                                    "20240917",
                                    "20240918"
                                ],
                                "stst": 56280
                            }
                        ],
                        "dtrstartstop": {
                            "geoid": 0,
                            "stpid": "235053",
                            "stpnm": "SEPULVEDA BL + DOLORES ST",
                            "seq": 3
                        },
                        "dtrendstop": {
                            "geoid": 0,
                            "stpid": "203001",
                            "stpnm": "REDONDO BEACH PIER",
                            "seq": 36
                        },
                        "delay": 0,
                        "repstops": [
                            {
                                "geoid": 783,
                                "stpid": "734025",
                                "stpnm": "Avalon Bl at Sepulveda Bl",
                                "seq": 1,
                                "lat": 33.80863199999843,
                                "lon": -118.26425600000013,
                                "adhoc": false
                            },
                            {
                                "geoid": 352,
                                "stpid": "215027",
                                "stpnm": "CARSON ST + MADRONA AV",
                                "seq": 20,
                                "lat": 33.83127499999939,
                                "lon": -118.34460600000085,
                                "adhoc": false,
                                "relpasstime": 185
                            }
                        ]
                    }
                ],
                "sbnm": "CWDetour-766cfd24-111d-40f0-9216-6e67576e603f"
            },
        ]
    }
}
```

Version 3 Release Notes

Version 3 of the Developer API contains a number of changes:

- The URL of the request changes.
- Most calls now support an **rtpidatafeed** parameter to query desired feeds of multi-feed systems
- In a multi-feed system, some calls now return an **rtpidatafeed** element in their results
- The results of some calls are now affected by detours which introduces a new **getdetours** call.
- The results of some calls are now affected by disruption management changes
- Standardization of format of the Route Directions call
- Changes to the Real Time Passenger Information call
- Miscellaneous fixes

Calling Version 3

Version 3 of the API is used by including “v3”in the request URL as follows:

https://www.ctabustracker.com/bustime/api/v3/getroutes?key=89dj2he89d8j3j3ksjhdue93j

Introduction of the Detours call

Some calls such as Stops and Patterns can now be affected by detours. These calls will reference detour ids which can be referenced in the new Detours call. See *How are dynamic changes to schedule data handled?* and the reference for the Detours call for more information.

Introduction of Disruption Management changes

Some calls are now affected by disruption management changes. For example, a prediction can now be marked as canceled if the vehicle will skip the associated stop. If the developer wishes to support disruption management, recurring requests to route data calls will be needed. See section 1.8 for more information.

Standardization of the Route Directions call

Version 2 introduced the multi-feed concept to the Route Directions call. In that version, the results for a multi-feed system had localization data but a single feed did not. In v3, the Route Directions call will always be formatted to show locale-specific data. See the reference for Route Directions for examples of this format.

Changes to Real Time Passenger Information call

Version 3 introduces some new elements and element name changes in order to provide developers with more valuable and more accurate information about RTPI feeds:

- The agency element is now displayname
- The call now returns disabled feeds in addition to enabled ones
- The enabled boolean element has been added

Miscellaneous Fixes

Predictions

- Using this call in a multi-feed system now appropriately returns “No Service Scheduled” for given **stpid**s
- All given **stpid**s/**vid**s are now in some way represented in the result, either with predictions or an error

Stops - Passing in a rtpidatafeed parameter along with stpid(s) no longer results in an empty response

Other Notes

- The **dir** and **rtdir** parameters now use the id of the direction instead of the localized name
- Much of the core API code has been optimized and primed for versioning, which should increase response time for users
- New error messages have been added to support new functionality in this version
- To allow easier transitioning from legacy versions to v3, the inconsistency of pluralizations of JSON arrays has not been changed

Dynamic Action Types

This section describes the dynamic action type identifiers available throughout the BusTime® Developer API's **dyn** elements.

ID	Name	Description
0	None	No change.
1	Canceled	The event or trip has been canceled.
2	Reassigned	The event or trip has been moved to a different work (to be handled by a different vehicle or operator).
3	Shifted	The time of this event, or the entire trip, has been moved.
4	Expressed	The event is "drop-off only" and will not stop to pick up passengers.
6	Stops Affected	This trip has events that are affected by Disruption Management changes, but the trip itself is not affected.
8	New Trip	This trip was created dynamically and does not appear in the TA schedule.
9	Partial Trip	This trip has been split, and this part of the split is using the original trip identifier(s). -or- The trip has been short-turned leading to the removal of shortturned stops from the trip resulting in the trip being partial.
10	Partial Trip New	This trip has been split, and this part of the split has been assigned a new trip identifier(s).
12	Delayed Cancel	This event or trip has been marked as canceled, but the cancellation should not be shown to the public.
13	Added Stop	This event has been added to the trip. It was not originally scheduled.
14	Unknown Delay	This trip has been affected by a delay.
15	Unknown Delay New	This trip, which was created dynamically, has been affected by a delay.
16	Invalidated Trip	This trip has been invalidated. Predictions for it should not be shown to the public.
17	Invalidated Trip New	This trip, which was created dynamically, has been invalidated. Predictions for it should not be shown to the public.
18	Cancelled Trip New	This trip, which was created dynamically, has been canceled.
19	Stops Affected New	This trip, which was created dynamically, has events that are affected by Disruption Management changes, but the trip itself is not affected.

Error codes

This section describes all possible error responses that can be received from the BusTime® Developer API.

Error Message	Related API Calls	Description
Internal server error - Unable to complete request at this time	All	The most general error message, given when we cannot find a more specific error message to send.
No API access permitted	All	The Developer API has been disabled by the Transit Authority.
No API access key supplied	All	The 'key=<DevKey>' parameter is missing from the API request.
Invalid API access key supplied	All	The given Developer key is not assigned to any users.
No version requested	All	The request URL is missing the version.
Unsupported version requested	All	The request URL contains an unsupported version.
Unsupported function	N/A	The request contains a function name that is not supported by the API.
Transaction limit for current day has been exceeded.	All	The user, identified by the Developer Key, has already reached the maximum number of API calls allowed for the day.
Invalid locale parameter	All	The requested locale string is not in a proper format. The proper format is "la" where la is a legal ISO 639 code.
Format parameter must be xml or json	All	The 'format' parameter is invalid. The value must be "xml" or "json".
No data found for parameter(s)	All except **gettime and getenhanceddetours**	No results were found that matched the given parameters.
No parameter provided	**getpattern**	Required 'rt' or 'pid' parameters are missing.
No parameter provided	**getpredictions**	The required 'stpid' or 'vid' parameters are missing.
No parameter provided	**getservicebulletins**	The required 'rt' or 'stpid' parameters are missing.
dir parameter missing	**getstops**	The required 'dir' parameter is missing.
rt parameter missing	**getdirections**, **getstops**, **getservicebulletins**	The required 'rt' parameter is missing.
Either rt or vid parameter must be specified	**getvehicles**	The request is required to contain either a 'rt' or 'vid' parameter.

Invalid parameter provided	**getpatterns**, **getpredictions, getdetours**	The listed parameter(s) does not match any known ID.
Maximum number of pid identifiers exceeded	**getpattern**	The 'pid' parameter contains more than 10 pattern IDs.
Invalid top parameter provided	**getpredictions**	The 'top' parameter is not an integer or contains extra characters. For instance "top=10" is legal but "top=10." is not.
Maximum number of <x> identifiers exceeded	**getpredictions**, **getvehicles**	The 'stpid' or 'vid' parameter contains too many IDs. <x> shows the maximum allowed in a single request.
No arrival times	**getpredictions**	The given stop has no scheduled arrival times.
No service scheduled	**getpredictions**	The given stop has no service scheduled.
Invalid RTPI Data Feed parameter	All except **gettime** and **getlocalelist**	The given 'rtpidatafeed' is an invalid or disabled feed.
No RTPI Data Feed parameter provided	**getdirections, getstops, getpatterns, getpredictions, getservicebulletins**	The required 'rtpidatafeed' parameter is missing.
The rtpidatafeed does not support this function	**getvehicles, getservicebulletins, getdetours, getenhanceddetours**	The given 'rtpidatafeed' is a valid feed but does not support the call's functionality.

Acknowledgements

The CTA Bus Tracker service is powered by BusTime, a product of Clever Devices, Inc. Both the API itself and the majority of this document were produced and developed by people at Clever Devices.

Special thanks go to Harper Reed and Dan O'Neil for their support and encouragement, and to the independent development community, for showing such great interest in developing applications with CTA data, leading to the creation of this official API.

Thank you.