Location Services In Action

Digital Platforms and Data Division
Data Systems and Services Branch

2020
1. Introduction to Location Services

2. BC Route Planner
   - Online resources
   - Making a request
   - Understanding the response
   - Python scripts

3. BC Address Geocoder
   - Common Features and Proximity Searches
   - Python script
BC Route Planner

Fastest/shortest route | Turn by turn directions
Nearest by road | Optimal stop order
API Overview

- **directions**: Directions, path, distance, travel time between a series of points

- **distance**: Distance and travel time between two points

- **route**: Path, distance and travel time between a series of points
Scope for today

- **directions:**
  - directions
  - optimalDirections
  - truck/directions

- **distance:**
  - distance/betweenPairs
Getting started

Location Services Homepage
https://data.gov.bc.ca/
  • -> Geographic Services
    • -> Location Services

Location Services in Action
• https://ols-demo.apps.gov.bc.ca/index.html
Getting started

Location Services Homepage
https://data.gov.bc.ca/

Location Services in Action
https://ols-demo.apps.gov.bc.ca/index.html
Open development

GitHub repo:
• [https://github.com/bcgov/ols-router](https://github.com/bcgov/ols-router)
  • Explore the ‘gh-pages’ branch for more documentation

Monitor progress:
• [https://github.com/bcgov/ols-router/issues](https://github.com/bcgov/ols-router/issues)
**GitHub repo:**

- [Open development GitHub repo](https://github.com/bcgov/ols-router)
- Explore the 'gh-pages' branch for more documentation.

**Monitor progress:**

- [GitHub Issues](https://github.com/bcgov/ols-router/issues)

---

<table>
<thead>
<tr>
<th>Commit</th>
<th>Message</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>324461</td>
<td>Add compliance audit file</td>
<td>4 months ago</td>
</tr>
<tr>
<td>324450</td>
<td>Update Dockerfile</td>
<td>9 months ago</td>
</tr>
<tr>
<td>324449</td>
<td>Change quotes</td>
<td>9 months ago</td>
</tr>
<tr>
<td>324448</td>
<td>Fix for infinite loop around internal turn restriction segments</td>
<td>2 months ago</td>
</tr>
<tr>
<td>324447</td>
<td>Create sonar-project-properties</td>
<td>10 months ago</td>
</tr>
</tbody>
</table>

---

Help people interested in this repository understand your project by adding a README.
Open development
GitHub repo:
• https://github.com/bcgov/ols-router

Explore the 'gh-pages' branch for more documentation

Monitor progress:
• https://github.com/bcgov/ols-router/issues

Introduction
Diving in

API Consoles

https://data.gov.bc.ca/
  • -> Geographic Services
    • -> Location Services
      • BC Route Planner
        • Launch REST API Console (API Key Required)
      • BC Address Geocoder
        • Launch REST API Console
### Diving in API Consoles

**Diving in API Consoles**

http://data.gov.bc.ca/

- **Geographic Services**
  - **Location Services**
  - **BC Route Planner**
  - **Launch REST API Console (API Key Required)**
  - **BC Address Geocoder**
    - **Launch REST API Console**

---

### API Endpoints

#### Directions

<table>
<thead>
<tr>
<th>HTTP Method</th>
<th>Endpoint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/directions</td>
<td>Get the directions, path, distance and travel time between a series of geographic points</td>
</tr>
<tr>
<td>POST</td>
<td>/directions</td>
<td>Get the directions, path, distance and travel time between a series of geographic points</td>
</tr>
<tr>
<td>GET</td>
<td>/optimalDirections</td>
<td>Get the directions, optimal path, distance and travel time between a start point and a series of end points which are reordered to minimize total distance or time.</td>
</tr>
<tr>
<td>POST</td>
<td>/optimalDirections</td>
<td>Get the directions, optimal path, distance and travel time between a start point and one or more end points which are reordered to minimize total distance or time.</td>
</tr>
<tr>
<td>GET</td>
<td>/truck/directions</td>
<td>Get the directions, path, distance and travel time between a series of geographic points for a commercial vehicle</td>
</tr>
<tr>
<td>POST</td>
<td>/truck/directions</td>
<td>Get the directions, path, distance and travel time between a series of geographic points</td>
</tr>
<tr>
<td>GET</td>
<td>/truck/optimalDirections</td>
<td>Get the directions, optimal path, distance and travel time between a start point and a series of end points which are reordered to minimize total distance or time for a commercial vehicle</td>
</tr>
<tr>
<td>POST</td>
<td>/truck/optimalDirections</td>
<td>Get the directions, optimal path, distance and travel time between a start point and one or more end points which are reordered to minimize total distance or time.</td>
</tr>
</tbody>
</table>

#### Distance

<table>
<thead>
<tr>
<th>HTTP Method</th>
<th>Endpoint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>/distance</td>
<td>Get distance and travel time between two geographic points</td>
</tr>
<tr>
<td>POST</td>
<td>/distance</td>
<td>Get distance and travel time between two geographic points</td>
</tr>
<tr>
<td>GET</td>
<td>/distance/betweenPairs</td>
<td>Get distance and travel time between each pair of geographic points</td>
</tr>
<tr>
<td>POST</td>
<td>/distance/betweenPairs</td>
<td>Get distance and travel time between each pair of geographic points</td>
</tr>
</tbody>
</table>
### Directions

**GET** `/directions`  
Get the directions, path, distance and travel time between a series of geographic points.

Represent the turn-by-turn directions, geometry, distance and time of the shortest path or fastest path between given start and end points.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>outputFormat</strong></td>
<td>Format of representation</td>
</tr>
<tr>
<td><strong>points</strong></td>
<td>A list of any number of route points in start to end order. See <a href="#">points</a></td>
</tr>
<tr>
<td><strong>outputSRS</strong></td>
<td>The EPSG code of the spatial reference system (SRS) to use for output geometries. See <a href="#">outputSRS</a></td>
</tr>
<tr>
<td><strong>criteria</strong></td>
<td>Routing criteria to optimize (e.g., shortest, fastest). Default is shortest.</td>
</tr>
<tr>
<td><strong>distanceUnit</strong></td>
<td>Distance unit of measure (e.g., km, mi). Default is km.</td>
</tr>
<tr>
<td><strong>roundTrip</strong></td>
<td>If true, route ends at start point. Default is false.</td>
</tr>
</tbody>
</table>
Diving in API Consoles

https://data.gov.bc.ca/

Geographic Services
Location Services
BC Route Planner
Launch REST API Console (API Key Required)
BC Address Geocoder
Launch REST API Console

Curl

eurl -X GET "https://router.map.gov.bc.ca/directions?json?points=153.7075483762,77.69561C-
153.7075483762,77.69561C&filterCriteria=false&distanceUnit=km&sendUrl=false&correct&hide=true" -H "accept:
/*" -H "apikey:"
Diving in

Developer Guides

https://data.gov.bc.ca/
  • -> Geographic Services
  • -> Location Services
    • BC Route Planner
    • BC Address Geocoder
Tools

• Chrome apps
  • ARC (Advanced REST client)

• Chrome extensions
  • JSONView
  • ZenHub

• Chrome developer console
  • CTRL + Shift + J

• Fiddler
BC Route Planner
Making a Request
outputFormat

• The file format for the response (*required*)

<table>
<thead>
<tr>
<th>Formats</th>
</tr>
</thead>
<tbody>
<tr>
<td>json</td>
</tr>
<tr>
<td>kml</td>
</tr>
<tr>
<td>html</td>
</tr>
</tbody>
</table>

https://router.api.gov.bc.ca/directions.json
points

• The points parameter *(required)* is a list of route points in order of start to finish.

• Listed as X1,Y1,...Xn,Yn in the projection specified by the 'outputSRS' parameter.

https://router.api.gov.bc.ca/directions.json?points=-123.1124558,49.2751466,-123.1168642,49.288516
outputSRS

- The EPSG code of the spatial reference system

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4326</td>
<td>WGS84</td>
</tr>
<tr>
<td>3005</td>
<td>BC Albers</td>
</tr>
<tr>
<td>26907, 26908, 26909, 269010, 26911</td>
<td>NAD83/UTM Zones 7N through 11N</td>
</tr>
</tbody>
</table>

https://router.api.gov.bc.ca/directions.json?points=-123.1124558,49.2751466,-123.1168642,49.288516

&outputSRS=4326

(default is -> 4326)
distanceUnit

- The distance unit of measure

<table>
<thead>
<tr>
<th>units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>km</td>
<td>Kilometres (default)</td>
</tr>
<tr>
<td>mi</td>
<td>Miles</td>
</tr>
</tbody>
</table>

https://router.api.gov.bc.ca/directions.json?points=-123.1124558,49.2751466,-123.1168642,49.288516&outputSRS=4326&distanceUnit=km

ℹ️ default is -> ‘km’
camelCase

- https://router.api.gov.bc.ca/directions.json?points=-123.33801,49.37924,-123.14850,49.33004 &distanceUnit=mi
  - Returns kilometers (default)

- https://router.api.gov.bc.ca/directions.json?points=-123.33801,49.37924,-123.14850,49.33004 &distanceUnit=mi
  - Returns miles
https://router.api.gov.bc.ca/directions.json?points=-123.1124558,49.2751466,-123.1168642,49.288516&outputSRS=4326&distanceUnit=km &criteria=shortest

default is -> ‘fastest’
https://router.api.gov.bc.ca/directions.json?points=-123.1124558,49.2751466,-123.1168642,49.288516&outputSRS=4326&distanceUnit=km &correctSide=true

default is -> ‘false’
roundTrip

= false

= true

correctSide routing
https://router.api.gov.bc.ca/directions.json?points=-123.1124558,49.2751466,-123.1168642,49.288516&outputSRS=4326&distanceUnit=km&correctSide=true&roundTrip=true

default is -> ‘false’
Turn Costs
# Crossing Costs

<table>
<thead>
<tr>
<th>Yield or Roundabout</th>
<th>vs.</th>
<th>Sign</th>
<th>vs.</th>
<th>Light</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="image1.jpg">Image</a></td>
<td></td>
<td><a href="image2.jpg">Image</a></td>
<td></td>
<td><a href="image3.jpg">Image</a></td>
</tr>
</tbody>
</table>

*Enabled by default*
Time dependency & Ferry Schedules

Enabled

Route travels 27.636 km in 2 hours 2 minutes
1 hr 18 min until next sailing

Disabled (default)

Route travels 27.636 km in 54 minutes 51 seconds
Defaults to a 10 minute wait

Depart at 10:37am
https://router.api.gov.bc.ca/directions.json?points=-123.3470,49.3763,-123.0814,49.3118
&enable=td,sc
&departure=2020-05-15T10:37:00-07:00
Adding many points (waypoints)

Route calculated in 0.06 seconds
Route travels 59.994 km in 1 hour 27 minutes
Route #1:
2.85 km in 5:26 min

Route #2:
1.06 km in 2:09 min
optimalDirections

Request:
• https://router.api.gov.bc.ca/optimalDirections.json?points=<LIST OF POINTS>

Response:
• Now includes ‘visitOrder’
Distortion Fields

- **Global Distortion Field**
  - Encourage vehicles to use more major roads

- **Local Distortion Field**
  - Special cases

Enabled by default
Municipality:
Discourage travel on Gatensbury Rd

Fastest route only

Fastest route & Global Distortion field

Port Moody, BC
Local Distortion Field

Fastest route & Global Distortion field & Local Distortion field
https://router.api.gov.bc.ca/truck/directions.json?points=-122.8544,49.2746,-122.8512,49.2633&outputSRS=4326&distanceUnit=km&correctSide=true
Truck Directions

Height (m)

Bridge (Max height = 3.77m)
Truck Directions

https://router.api.gov.bc.ca/truck/directions.json?
points=<LIST OF POINTS>...&height=3.8

You can also include – Length (m), Width (m), Weight (kg)

https://router.api.gov.bc.ca/truck/directions.json?
points=<LIST OF POINTS>...
&height=3.8&width=5&length=30&weight=12000
Truck Route Partitioning

```
partition: "isFerry,isTruckRoute,locality",
partitions: [
  - {
    index: 0,
    isFerry: false,
    isTruckRoute: false,
    locality: "Bowen Island"
  },
  - {
    index: 5,
    isFerry: true,
    isTruckRoute: false,
    locality: "Ferry"
  },
  - {
    index: 22,
    isFerry: false,
    isTruckRoute: false,
    locality: "West Vancouver"
  },
  - {
    index: 24,
    isFerry: false,
    isTruckRoute: true,
    locality: "West Vancouver"
  },
  - {
    index: 195,
    isFerry: false,
    isTruckRoute: false,
    locality: "West Vancouver"
  }
],
```
Truck Route Partitioning

https://router.api.gov.bc.ca/truck/directions.json?
points=<LIST OF POINTS>...
&partition=isTruckRoute,isFerry,locality
Truck Route Notifications
Truck Route Notifications

- {
  type: "TURN_RIGHT",
  name: "Robson St",
  distance: 0.631,
  time: 83,
  text: "Turn right onto Robson St for 700 m (1 minute 23 seconds)",
  point: [
    -125.12854,
    49.287
  ],
  notifications: [
    - {
      type: "TRUCK.Restriction",
      message: "Vehicles over 15.25 meters in length may only use this road between 7:00 am and 6:00 pm, Monday to Sunday"
    }
  ]
},

- {
  type: "TURN_LEFT",
  name: "Hornby St",
  distance: 0.179,
  time: 29,
  text: "Turn left onto Hornby St for 150 m (29 seconds)",
  point: [
    -125.12176,
    49.28054
  ],
  notifications: [
    - {
      type: "TRUCK.Restriction",
      message: "Vehicles over 15.25 meters in length may only use this road between 7:00 am and 6:00 pm, Monday to Sunday"
    }
  ]
}
Truck Route Notifications

- There is no ‘notification’ request parameter
- Provided automatically (where applicable) by requesting a truck route using `/truck/directions` or `/truck/optimalDirections`

https://router.api.gov.bc.ca/truck/directions.json
https://router.api.gov.bc.ca/truck/optimalDirections.json
HTTP Response

• Different for requests to /distance vs. /directions vs. /route

• Full list provided in the developer guide
  • https://bcgov.github.io/ols-router/router-developer-guide.html
<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>routeDescription</td>
<td>String</td>
</tr>
<tr>
<td>searchTimestamp</td>
<td>Datetime</td>
</tr>
<tr>
<td>executionTime</td>
<td>Real</td>
</tr>
<tr>
<td>version</td>
<td>String</td>
</tr>
<tr>
<td>disclaimer</td>
<td>String</td>
</tr>
<tr>
<td>privacyStatement</td>
<td>String</td>
</tr>
<tr>
<td>copyrightNotice</td>
<td>String</td>
</tr>
<tr>
<td>copyrightLicense</td>
<td>String</td>
</tr>
<tr>
<td>srsCode</td>
<td>Integer</td>
</tr>
<tr>
<td>criteria</td>
<td>String</td>
</tr>
<tr>
<td>enable</td>
<td>String</td>
</tr>
<tr>
<td>distanceUnit</td>
<td>String</td>
</tr>
<tr>
<td>points</td>
<td>List of Point</td>
</tr>
<tr>
<td>routeFound</td>
<td>Boolean</td>
</tr>
<tr>
<td>distance</td>
<td>String</td>
</tr>
<tr>
<td>time</td>
<td>Integer</td>
</tr>
<tr>
<td>timeText</td>
<td>String</td>
</tr>
<tr>
<td>route</td>
<td>List of Point</td>
</tr>
<tr>
<td>notifications</td>
<td>String</td>
</tr>
<tr>
<td>directions</td>
<td>String</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>routeDescription</td>
<td>String</td>
</tr>
<tr>
<td>searchTimestamp</td>
<td>Datetime</td>
</tr>
<tr>
<td>executionTime</td>
<td>Real</td>
</tr>
<tr>
<td>version</td>
<td>String</td>
</tr>
<tr>
<td>disclaimer</td>
<td>String</td>
</tr>
<tr>
<td>privacyStatement</td>
<td>String</td>
</tr>
<tr>
<td>copyrightNotice</td>
<td>String</td>
</tr>
<tr>
<td>copyrightLicense</td>
<td>String</td>
</tr>
<tr>
<td>srsCode</td>
<td>Integer</td>
</tr>
<tr>
<td>criteria</td>
<td>String</td>
</tr>
<tr>
<td>enable</td>
<td>String</td>
</tr>
<tr>
<td>distanceUnit</td>
<td>String</td>
</tr>
<tr>
<td>points</td>
<td>List of Point</td>
</tr>
<tr>
<td>routeFound</td>
<td>Boolean</td>
</tr>
<tr>
<td>distance</td>
<td>String</td>
</tr>
<tr>
<td>time</td>
<td>Integer</td>
</tr>
<tr>
<td>timeText</td>
<td>String</td>
</tr>
<tr>
<td>route</td>
<td>List of Point</td>
</tr>
<tr>
<td>notifications</td>
<td>String</td>
</tr>
<tr>
<td>directions</td>
<td>String</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>routeDescription</td>
<td>String</td>
</tr>
<tr>
<td>searchTimestamp</td>
<td>Datetime</td>
</tr>
<tr>
<td>executionTime</td>
<td>Real</td>
</tr>
<tr>
<td>version</td>
<td>String</td>
</tr>
<tr>
<td>disclaimer</td>
<td>String</td>
</tr>
<tr>
<td>privacyStatement</td>
<td>String</td>
</tr>
<tr>
<td>copyrightNotice</td>
<td>String</td>
</tr>
<tr>
<td>copyrightLicense</td>
<td>String</td>
</tr>
<tr>
<td>srsCode</td>
<td>Integer</td>
</tr>
<tr>
<td>criteria</td>
<td>String</td>
</tr>
<tr>
<td>enable</td>
<td>String</td>
</tr>
<tr>
<td>distanceUnit</td>
<td>String</td>
</tr>
<tr>
<td>points</td>
<td>List of Point</td>
</tr>
<tr>
<td>routeFound</td>
<td>Boolean</td>
</tr>
<tr>
<td>distance</td>
<td>String</td>
</tr>
<tr>
<td>time</td>
<td>Integer</td>
</tr>
<tr>
<td>timeText</td>
<td>String</td>
</tr>
<tr>
<td>route</td>
<td>List of Point</td>
</tr>
<tr>
<td>notifications</td>
<td>String</td>
</tr>
<tr>
<td>directions</td>
<td>String</td>
</tr>
</tbody>
</table>
{
"routeDescription": null,
"searchTimestamp": "2020-04-17 21:13:20",
"executionTime": 48,
"version": "2.1.0",
"disclaimer": "https://www2.gov.bc.ca/gov/content?id=79F93E018712422FBC8E674A67A70535",
"privacyStatement": "https://www2.gov.bc.ca/gov/content?id=9E809E16065E4FF4BFB38E07B472232",
"copyrightNotice": "Copyright 2020 Province of British Columbia - Open Government License",
"copyrightLicense": "https://www2.gov.bc.ca/gov/content?id=A519A56BC2BF44E4A808B33FCF527F61",
"rsrCode": 4328,
"criteria": "fastest",
"enable": "gdf,ldf,tc,tr,xc",
"distanceUnit": "km",
"points": [
    [ -123.70794, 48.77869 ],
    [ -123.53785, 48.382 ]
],
"routeFound": true,
"distance": 55.296,
"time": 3110.129311972462,
"timeText": "51 minutes 50 seconds"}
"routeDescription": null,
"searchTimestamp": "2020-04-17 20:53:51",
"executionTime": 49,
"version": "2.1.0",
"disclaimer": "https://www2.gov.bc.ca/gov/content?id=79F93E818712422FBC8E674A67A76535",
"privacyStatement": "https://www2.gov.bc.ca/gov/content?id=9E899E16955E4FF4BF3B0E07B4722932",
"copyrightNotice": "Copyright 2020 Province of British Columbia - Open Government License",
"copyrightLicense": "https://www2.gov.bc.ca/gov/content?id=A519A56BC2BF44E4A08B33FCF527F61",
"srsCode": 4326,
"criteria": "fastest",
"enable": "gdf,ldf,tc,tr,xc",
"distanceUnit": "km",
"points": [
  [-123.70794, 48.77869],
  [-123.53785, 48.382]
],
"routeFound": true,
"distance": 55.296,
"time": 31.10129311972462,
"timeText": "51 minutes 50 seconds",
"route": [
  [-123.70793, 48.7785],
  [-123.70719, 48.77851],
  [-123.7069, 48.77769],
  [-123.7066, 48.77685],
  [-123.7067, ...]
"directions": [ 
{
    "type": "START",
    "name": "Kenneth St",
    "distance": 0.055,
    "time": 17,
    "heading": "EAST",
    "text": "Head east on Kenneth St for 55 m (17 seconds)",
    "point": [ 
                 -123.70793,
                 48.7785
               ]
},
{
    "type": "TURN_RIGHT",
    "name": "Canada Ave",
    "distance": 0.189,
    "time": 25,
    "text": "Turn right onto Canada Ave for 200 m (25 seconds)",
    "point": [ 
                 -123.70719,
                 48.77851
               ]
} 
]
"routeDescription": null,
"searchTimestamp": "2020-04-17 20:53:51",
"executionTime": 49,
"version": "2.1.0",
"disclaimer": "https://www2.gov.bc.ca/gov/content?id=79F93E818712422FBC8E67A67A76535",
"privacyStatement": "https://www2.gov.bc.ca/gov/content?id=9E890E16955E4FF4BF3B0E07B4722932",
"copyrightNotice": "Copyright 2020 Province of British Columbia - Open Government License",
"copyrightLicense": "https://www2.gov.bc.ca/gov/content?id=A519A56BC2BF44E4A088B33FCF527F61",
"srsCode": 4326,
"criteria": "fastest",
"enable": "gdf,ldf,tc,tr,xc",
"distanceUnit": "km",
"points": [ [ -123.70794, 48.77869 ], [ -123.53785, 48.382 ] ],
"routeFound": true,
"distance": 55.296,
"time": 3110.129311972462,
"timeText": "51 minutes 50 seconds",
"route": [
  [[-123.70793, 48.7785],
   [-123.70793, 48.7785]],
  [[-123.7069, 48.7779],
   [-123.7069, 48.7779]],
  [[-123.7066, 48.7768],
   [-123.7067, 48.7768]]]
betweenPairs

- A sample betweenPairs script

- Input csv file

- Output csv file
betweenPairs - Request

• Similar to previous requests, with two notable differences

• https://router.api.gov.bc.ca/distance/betweenPairs.json?fromPoints=<LIST OF POINTS>&toPoints=<LIST OF POINTS>

• Or, https://router.api.gov.bc.ca/truck/distance/betweenPairs.json?
API Keys

• Web apps should pass the API key in the request header

• API keys can also be passed to a script as an argument
  • https://router.api.gov.bc.ca/optimalDirections.json?points=<LIST OF POINTS>&apikey=32kj3j3lk34n234lk234n4n3l3

• To acquire an API key
  • Developer / personal:
  • Production / organization:
    Email DataBC: Data@gov.bc.ca
Demo #1
Advanced REST Client

https://install.advancedrestclient.com
Demo #2
API Console

https://www2.gov.bc.ca/gov/content/data/geographic-data-services/location-services/route-planner
Python example #1

BC Address Geocoder

Cleaning | Correction | Completion | Geocoding | Reverse geocoding
<table>
<thead>
<tr>
<th>Application</th>
<th>Number of Requests</th>
<th>Viewer</th>
<th>Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address List Editor</td>
<td>1 – 1,000</td>
<td>Web browser</td>
<td>Public</td>
</tr>
<tr>
<td>Auto-completion Address Finder Widget</td>
<td>1</td>
<td>Web browser</td>
<td>Public</td>
</tr>
<tr>
<td>Batch Geocoder</td>
<td>1 - Millions</td>
<td>Web browser</td>
<td>IDIR &amp; BCeID</td>
</tr>
<tr>
<td>Batch Address List Submitter</td>
<td>1 – Millions</td>
<td>Python</td>
<td>‘cpf’ account</td>
</tr>
<tr>
<td>Location Services in Action</td>
<td>1, M (within)</td>
<td>Web browser (modern)</td>
<td>Public</td>
</tr>
<tr>
<td>Physical Address Viewer</td>
<td>1, M (within)</td>
<td>Google Earth</td>
<td>Public</td>
</tr>
<tr>
<td>REST API Console</td>
<td>1</td>
<td>Web browser</td>
<td>Public</td>
</tr>
</tbody>
</table>
Sample workflow

Addresses → BC Address Geocoder → Coordinates → BC Route Planner → Your route(s)

Existing coordinates → BC Route Planner → Your route(s)
BC Address Geocoder
Common Features and Proximity Searches
## Available formats

### BC Address Geocoder

<table>
<thead>
<tr>
<th>Output Formats</th>
</tr>
</thead>
<tbody>
<tr>
<td>csv</td>
</tr>
<tr>
<td>geojson</td>
</tr>
<tr>
<td>gml</td>
</tr>
<tr>
<td>json</td>
</tr>
<tr>
<td>kml</td>
</tr>
<tr>
<td>shpz</td>
</tr>
<tr>
<td>xhtml</td>
</tr>
</tbody>
</table>

### Batch Geocoder

<table>
<thead>
<tr>
<th>Input Formats</th>
<th>Output Formats</th>
</tr>
</thead>
<tbody>
<tr>
<td>csv</td>
<td>csv</td>
</tr>
<tr>
<td>GeoPackage</td>
<td>geojson</td>
</tr>
<tr>
<td>json</td>
<td>GeoPackage</td>
</tr>
<tr>
<td>shpz</td>
<td>gml</td>
</tr>
<tr>
<td>tsv</td>
<td>html</td>
</tr>
<tr>
<td></td>
<td>xhtml</td>
</tr>
<tr>
<td></td>
<td>json</td>
</tr>
<tr>
<td></td>
<td>kml</td>
</tr>
<tr>
<td></td>
<td>kmz</td>
</tr>
<tr>
<td></td>
<td>kmz</td>
</tr>
<tr>
<td></td>
<td>shpz</td>
</tr>
<tr>
<td></td>
<td>tsv</td>
</tr>
<tr>
<td></td>
<td>xml</td>
</tr>
</tbody>
</table>

[https://geocoder.api.gov.bc.ca/addresses.json](https://geocoder.api.gov.bc.ca/addresses.json)
- any (default)
- accessPoint
- frontDoorPoint
- parcelPoint
- rooftopPoint
- routingPoint
Intersections

Types:
- 2-way
- 3-way
- 4-way
- 5+
- Dead-ends
Standardize and Geocode

- Input (addressString) = 3400 Davidson Av, Victoria

- Output
  - (fullAddress) = 3400 Davidson Ave, Saanich, BC
  - Coordinates, faults etc. in the format of your choice
Sample Request

- https://geocoder.api.gov.bc.ca(addresses.json?addressString=3400 Davidson Av, Victoria&locationDescriptor=parcelPoint&maxResults=1&minScore=90

Saanich, BC
"features": [
{
"type": "Feature",
"geometry": {
"type": "Point",
"crs": {
"type": "EPSG",
"properties": {
"code": 4326
}
}
},
"coordinates": [
-123.3797449,
48.4529005
]
}]}
fullAddress vs. addressString

score
= (precisionPoints) – (faults)
= 100 – (4 + 2 + 1)
Unique and change monthly
Proximity
Nearest Site

- https://geocoder.api.gov.bc.ca/sites/nearest.kml?point=-118.19724,50.99595&maxDistance=250&locationDescriptor=accessPoint
Proximity
Near Sites

• https://geocoder.api.gov.bc.ca/sites/near.kml?point=-118.19724,50.99595&maxDistance=250&maxResults=20&locationDescriptor=accessPoint
Proximity
Near Sites

• Terrace, BC

https://geocoder.api.gov.bc.ca/sites/near.kml?point=-118.19724,50.99595&maxDistance=250&maxResults=20&locationDescriptor=accessPoint

• Saanich, BC
Proximity
Near Sites
83 Terrace, BC
Saanich, BC

https://geocoder.api.gov.bc.ca/sites/near.kml?point=-118.19724,50.99595&maxDistance=250&maxResults=20&locationDescriptor=accessPoint
Proximity
Sites Within a BBOX

• https://geocoder.api.gov.bc.ca/sites/within.kml?bbox=-119.51,49.48,-119.53,49.50&outputSRS=4326&maxResults=100&locationDescriptor=parcelPoint
BC Address Geocoder
Batch List Submitter
Batch Address List Submitter

Accessible from the DataBC website

Hosted on GitHub
- https://bcgov.github.io/ols-devkit/als/
Thank you

https://data.gov.bc.ca/
Email: data@gov.bc.ca