

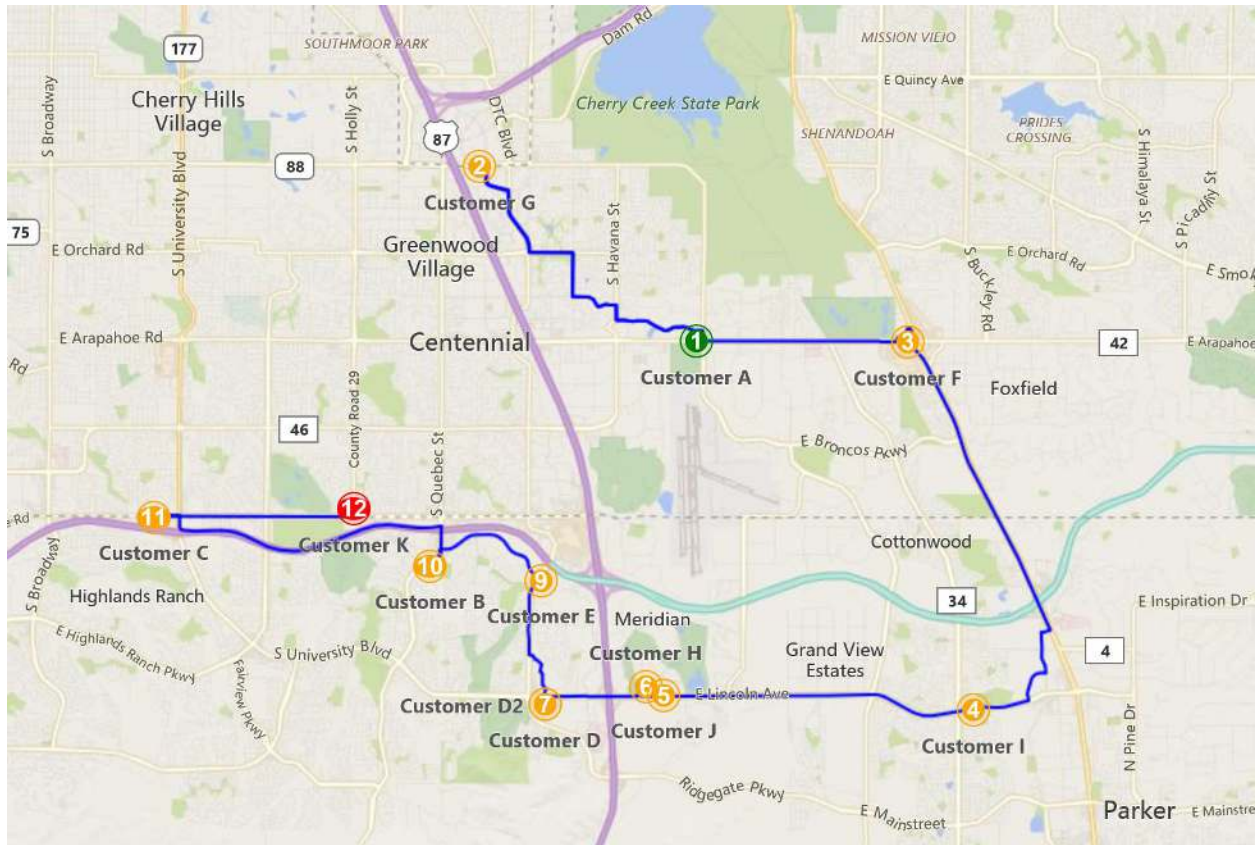


# RouteSavvy API

## Introduction

RouteSavvy API is an easy to use REST service for optimizing driving stop order. RSAPI.svc allows both GET and POST options. Requests use JSON formatted strings and results are returned in JSON format for optimal use in web applications.

The optimized results return stops ordered for the most efficient driving route. Results include turn by turn directions for reach route leg and a latitude, longitude polyline describing the route for use in map visualization.



Sample optimized Stops and RoutePath displayed over Bing Maps

**GET - RSAPI.svc/GetOptimize** requests are limited to the URL encoded parameter maximum length. Although this varies by browser, 2048 char is the typical suggested limit for encoding stops in the URL string. Because of the URL length limitation smaller numbers of stops can be optimized.

**POST - RSAPI.svc/PostOptimize** requests do not have URL length limitations and can accommodate large numbers of stops.

### URLs

Base Url: <http://optimizer2.routesavvy.com>

Service wsdl Url: <http://optimizer2.routesavvy.com/RSAPI.svc>

POST Url: <http://optimizer2.routesavvy.com/RSAPI.svc/PostOptimize>

GET Url: <http://optimizer2.routesavvy.com/RSAPI.svc/GetOptimize?query={jsonquery}>

### More information

General and purchase information is at <http://www.routesavvy.com/routesavvy-api>

Free trial <https://www.routesavvy.com/routesavvy-api-free-trial>

Online Documentation <https://www.routesavvy.com/routesavvy-api-docs>

### Predictive Traffic

Predictive Traffic is an advanced route optimization feature that allows fleet managers to plan a multi-stop route based on when the route will be driven. This is accomplished because historical data on known traffic patterns (such as at rush hour) are incorporated into the calculations for the most efficient route.

## Request JSON

### Example JSON Request:

```
{
  "Locations": [{
    "Name": "Customer A",
    "Latitude": 39.595140,
    "Longitude": -104.849620,
    "VisitDurationInMinutes": 10
  },
  {
    "Name": "Customer B",
    "Latitude": 39.558012,
    "Longitude": -104.906670,
    "VisitDurationInMinutes": 5
  },
  {
    "Name": "Customer C",
    "Latitude": 39.566128,
    "Longitude": -104.965901,
    "VisitDurationInMinutes": 10
  },
  {
    "Name": "Customer D",
    "Latitude": 39.535220,
    "Longitude": -104.882080,
    "VisitDurationInMinutes": 10
  },
  {
    "Name": "Customer E",
    "Latitude": 39.555669,
    "Longitude": -104.882878,
    "VisitDurationInMinutes": 5
  }
  ],
}
```

```

    ],
    "OptimizeParameters": {
      "AppId": "90c13fb711be4684bc724d306321a609",
      "OptimizeType": "distance",
      "RouteType": "realroadcar",
      "Avoid": "none",
      "Departure": "2020-05-23T17:30:00"
    }
  }
}

```

### Request components:

Array of Locations – minimum 3 locations (note that no optimization is required until four or more stops are requested.)

<pre> {   "Name": string,   "Latitude": float,   "Longitude": float,   "VisitDurationInMinutes": integer minutes } </pre>	<p>Example</p> <p>"Customer A"</p> <p>39.595140</p> <p>-104.906670</p> <p>10</p>
---	--

The first location is the start stop and the last location is the ending stop. If you wish to optimize a round trip route, duplicate the start stop as the end stop.

Optimize Parameters:

<pre> "OptimizeParameters": {   "AppId": string - user token   "OptimizeType": string - time or distance   "RouteType": string – basic ,realroadcar, realroadcarpredictive   "Avoid": string - none, tolls, or highways   "Departure": string GMT datetime } </pre>	<p>example:</p> <p>"90c13.....21a609"</p> <p>"distance"</p> <p>"realroadcar"</p> <p>"none"</p> <p>"2020-05-23T17:30:00"</p>
---	---

Notes:

RouteType *basic* is limited to 500 stops, *realroadcar* and *realroadcarpredictive* are limited to 300 stops.

RouteType *realroadcarpredictive* uses a future Departure date time for the start time, taking into account historical variations by day of week, holidays, and rush hour traffic.

Important: The Departure date time for *realroadcarpredictive* must be a valid UTC date time and must be in the future. For example 2019-07-12 8:00AM MST in UTC is 2019-07-12T14:00:00

RouteTypes, *basic* and *realroadcar* only use the departure date time for labelling directions and calculating stop arrival/departure times but do not take into account traffic variations.

Sample GET Optimize request which can run from a browser once your token has been pasted into the AppId.

```
http://optimizer2.routesavvy.com/RSAPI.svc/GETOptimize?query= { "Locations": \[{ "Name": "Customer A", "Latitude": 39.595140, "Longitude": -104.849620, "VisitDurationInMinutes":10}, { "Name": "Customer B", "Latitude": 39.558012, "Longitude": -104.906670, "VisitDurationInMinutes":5}, { "Name": "Customer C", "Latitude": 39.566128, "Longitude": -104.965901, "VisitDurationInMinutes":10}, { "Name": "Customer D", "Latitude": 39.535220, "Longitude": -104.882080, "VisitDurationInMinutes":10}, { "Name": "Customer D2", "Latitude": 39.535220, "Longitude": -104.882080, "VisitDurationInMinutes":10}, { "Name": "Customer E", "Latitude": 39.555669, "Longitude": -104.882878, "VisitDurationInMinutes":5}, { "Name": "Customer F", "Latitude": 39.595030, "Longitude": -104.804320, "VisitDurationInMinutes":5}, { "Name": "Customer G", "Latitude": 39.624000, "Longitude": -104.896010, "VisitDurationInMinutes":5}, { "Name": "Customer H", "Latitude": 39.538212, "Longitude": -104.860457, "VisitDurationInMinutes":10}, { "Name": "Customer I", "Latitude": 39.534298, "Longitude": -104.790182, "VisitDurationInMinutes":10}, { "Name": "Customer J", "Latitude": 39.536745, "Longitude": -104.856546, "VisitDurationInMinutes":5}, { "Name": "Customer K", "Latitude": 39.567508, "Longitude": -104.923069, "VisitDurationInMinutes":5}\], "OptimizeParameters": { "AppId": "<user token>", "OptimizeType": "distance", "RouteType": "realroadcar", "Avoid": "none", "Departure": "2020-05-23T17:30:00"}}
```

Sample POST optimize request

<http://optimizer2.routesavvy.com/RSAPI.svc/POSTOptimize>

```
{
  "Locations": [{
    "Name": "Customer A",
    "Latitude": 39.595140,
    "Longitude": -104.849620,
    "VisitDurationInMinutes": 10
  },
  {
    "Name": "Customer B",
    "Latitude": 39.558012,
    "Longitude": -104.906670,
    "VisitDurationInMinutes": 5
  },
  {
    "Name": "Customer C",
    "Latitude": 39.566128,
    "Longitude": -104.965901,
    "VisitDurationInMinutes": 10
  },
  {
    "Name": "Customer D",
```

```

        "Latitude": 39.535220,
        "Longitude": -104.882080,
        "VisitDurationInMinutes": 10
    },
    {
        "Name": "Customer E",
        "Latitude": 39.555669,
        "Longitude": -104.882878,
        "VisitDurationInMinutes": 5
    },
    "OptimizeParameters": {
        "Appld": "90c13fb711be4684bc724d306321a609",
        "OptimizeType": "distance",
        "RouteType": "realroadcar",
        "Avoid": "none",
        "Departure": "2020-05-23T17:30:00"
    }
}

```

## Result JSON

### Sample JSON Result:

```

{
    "Message": "Success",
    "OptimizedStops": [
        {
            "Arrival": null,
            "Departure": "9:25:00 3/29/2018",
            "Distance": null,
            "Duration": null,
            "IsDuplicate": false,
            "Name": "Customer A",
            "RouteLocation": {
                "Latitude": 39.59514,
                "Longitude": -104.84962
            },
            "StopTimeMinutes": 10
        },
        {
            "Arrival": "9:39:44 3/29/2018",
            "Departure": "9:44:44 3/29/2018",
            "Distance": "4.14 miles (6.67 km)",
            "Duration": "14 minutes, 44 seconds",
            "IsDuplicate": false,
            "Name": "Customer G",
            "RouteLocation": {
                "Latitude": 39.624,
                "Longitude": -104.89601
            }
        }
    ]
}

```

```

    },
    "StopTimeMinutes": 5
  },
  {
    "Arrival": "10:06:40 3/29/2018",
    "Departure": "10:11:40 3/29/2018",
    "Distance": "6.88 miles (11.07 km)",
    "Duration": "21 minutes, 56 seconds",
    "IsDuplicate": false,
    "Name": "Customer F",
    "RouteLocation": {
      "Latitude": 39.59503,
      "Longitude": -104.80432
    },
    "StopTimeMinutes": 5
  },
  .
  .
  .
},
{
  "Arrival": "12:03:39 3/29/2018",
  "Departure": "12:08:39 3/29/2018",
  "Distance": "2.38 miles (3.84 km)",
  "Duration": "4 minutes, 59 seconds",
  "IsDuplicate": false,
  "Name": "Customer K",
  "RouteLocation": {
    "Latitude": 39.567508,
    "Longitude": -104.923069
  },
  "StopTimeMinutes": 5
}
],
"Route": {
  "DriveDistance": 51.698,
  "DriveDistanceUnit": "Kilometer",
  "DriveTime": 5019,
  "DriveTimeUnit": "Second",
  "RouteLegs": [
    {
      "Directions": [
        "Depart CO-88 W / E Arapahoe Rd toward S Paris St",
        "Turn right onto S Paris St",
        "Turn left onto E Peakview Ave",
        "Turn right onto S Havana St",
        "Turn left onto E Caley Ave",
        "Turn right onto S Dayton St",
        "Turn left onto E Orchard Rd",
        "Turn right onto DTC Blvd",
        "Turn left onto E Prentice Ave",

```

```

        "Bear right onto S Ulster St",
        "Turn left onto E Belleview Ave",
        "Make a U-turn at Promenade Pl",
        "Arrive at E Belleview Ave"
    ],
    "DriveDistance": 6.668,
    "DriveTime": 884,
    "LegBegin": {
        "Name": "Customer A",
        "RouteLocation": {
            "Latitude": 39.59514,
            "Longitude": -104.84962
        }
    },
    "LegEnd": {
        "Name": "Customer G",
        "RouteLocation": {
            "Latitude": 39.624,
            "Longitude": -104.89601
        }
    }
},
{
    "Directions": [
        "Depart E Belleview Ave toward S Ulster St",
        "Turn right onto S Ulster St",
        "Turn left onto E Prentice Ave",
        "Turn right onto DTC Blvd",
        "Turn left onto E Orchard Rd",
        "Turn right onto S Dayton St",
        "Turn left onto E Caley Ave",
        "Turn right onto S Havana St",
        "Turn left onto E Peakview Ave",
        "At roundabout, take 1st exit onto S Peoria St",
        "Turn left onto CO-88 / E Arapahoe Rd",
        "Turn left onto S Helena St",
        "Turn right onto road",
        "Take ramp right",
        "Arrive at ramp"
    ],
    "DriveDistance": 11.065,
    "DriveTime": 1316,
    "LegBegin": {
        "Name": "Customer G",
        "RouteLocation": {
            "Latitude": 39.624,
            "Longitude": -104.89601
        }
    },
    "LegEnd": {
        "Name": "Customer F",

```



```

        "RouteLocation": {
            "Latitude": 39.59503,
            "Longitude": -104.8043
        }
    },
    .
    .
    {
        "Directions": [
            "Depart E County Line Rd toward E County Line Rd",
            "Turn left onto County Road 29 / S Holly St",
            "Arrive at County Road 29 / S Holly St"
        ],
        "DriveDistance": 3.838,
        "DriveTime": 299,
        "LegBegin": {
            "Name": "Customer C",
            "RouteLocation": {
                "Latitude": 39.566128,
                "Longitude": -104.965901
            }
        },
        "LegEnd": {
            "Name": "Customer K",
            "RouteLocation": {
                "Latitude": 39.56745,
                "Longitude": -104.92307
            }
        }
    }
},
"RoutePath": [
    [39.595139,-104.849619],
    [39.595139,-104.85009],
    [39.59552,-104.85009],
    .
    .
    .
    [39.565979,-104.92306],
    [39.56614,-104.923069],
    [39.56745,-104.923069]
]
}
}

```

**Result components:**

```
{
```

```

"Message": string - "Success" or error message
"OptimizedStops": Array of stops
"Route": route details
}

```

**OptimizedStops** Array of stops in the new optimized order

**Stops**

"Arrival": string date time	Example
"Departure": string date time	"9:39:44 3/29/2018",
"Distance": string distance mi and km	"9:44:44 3/29/2018",
"Duration": string drive time min and sec	"4.14 miles (6.67 km)",
"IsDuplicate": Boolean	"14 minutes, 44 seconds",
"Name": string	false,
"RouteLocation": {	"Customer G",
"Latitude": float	39.624,
"Longitude": float	-104.89601
},	
"StopTimeMinutes": integer	5

**OptimizedStops**

```

[
{
  "Arrival": null,
  "Departure": "9:25:00 3/29/2018",
  "Distance": null,
  "Duration": null,
  "IsDuplicate": false,
  "Name": "Customer A",
  "RouteLocation": {
    "Latitude": 39.59514,
    "Longitude": -104.84962
  },
  "StopTimeMinutes": 10
},
{
  "Arrival": "9:39:44 3/29/2018",
  "Departure": "9:44:44 3/29/2018",
  "Distance": "4.14 miles (6.67 km)",
  "Duration": "14 minutes, 44 seconds",
  "IsDuplicate": false,
  "Name": "Customer G",
  "RouteLocation": {
    "Latitude": 39.624,
    "Longitude": -104.89601
  },
  "StopTimeMinutes": 5
}
]

```

```
    },
```

### Route Components

```
{
  "DriveDistance": float total distance in DriveDistanceUnit      51.698,
  "DriveDistanceUnit": string                                     "Kilometer",
  "DriveTime": integer DriveTimeUnit                             5019,
  "DriveTimeUnit": string                                       "Second",
  "RouteLegs": Array of Route Legs                               [...],
  "RoutePath": Array of lat,lon                                  [...],
}
```

### RouteLegs

```
{
  "Directions": Array of strings
  Example
  [
    "Depart CO-88 W toward S Paris St",
    "Turn right onto S Paris St",
    "Turn left onto E Peakview Ave",
    "Turn right onto S Havana St",
    "Turn left onto E Caley Ave",
    "Turn right onto S Dayton St",
    "Turn left onto E Orchard Rd",
    .
    .
  ],
  "DriveDistance": float in DriveDistanceUnit                  6.668,
  "DriveTime": integer in DriveTimeUnit                         884,
  "LegBegin": {
    "Name": string                                              "Customer A",
    "RouteLocation": {
      "Latitude": float                                          39.59514,
      "Longitude": float                                         -104.84962
    }
  },
  "LegEnd": {
    "Name": string                                              "Customer G",
    "RouteLocation": {
      "Latitude": float                                          39.624,
      "Longitude": float                                         -104.89601
    }
  }
},
```

### RoutePath

```
[
```

```
[ 39.595139,-104.849619],  
[ 39.595139,-104.85009],  
[ 39.59552,-104.85009],
```

```
.  
. .  
. . .
```

### Errors results

If there is an error, Message returns information about the error. OptimizedStops and Route are null.

```
{  
  "Message": "Invalid Token. For assistance, please contact routesavvy@onterrasystems.com",  
  "OptimizedStops": null,  
  "Route": null  
}
```

### Support

Please contact [support@routesavvy.com](mailto:support@routesavvy.com) with details on your issue and we will respond as quickly as possible.

