Intermodal Park and Ride

Developer's Guide

Version 1.1.0

Important Information
Notices

Topics:

This section contains document notices.

Legal Notices

Trademark Acknowledgements

HERE is trademark or registered trademark of HERE Global B.V.

Other product and company names mentioned herein may be trademarks or trade names of their respective owners.

Disclaimer

This content is provided "as-is" and without warranties of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability, fitness for a particular purpose, satisfactory quality and non-infringement. HERE does not warrant that the content is error free and HERE does not warrant or make any representations regarding the quality, correctness, accuracy, or reliability of the content. You should therefore verify any information contained in the content before acting on it.

To the furthest extent permitted by law, under no circumstances, including without limitation the negligence of HERE, shall HERE be liable for any damages, including, without limitation, direct, special, indirect, punitive, consequential, exemplary and/ or incidental damages that result from the use or application of this content, even if HERE or an authorized representative has been advised of the possibility of such damages.

Document Information

Product	
Name:	Intermodal Park and Ride
Version:	Version 1.1.0
Document	
Name:	Intermodal Park and Ride Developer's Guide
ID:	1038008-1513273072-8b2c170e
Status:	FINAL
Date:	2017-Dec-14, 17:40 (GMT)

Contents

Chapter 1
Introduction

Topics:

This document introduces Intermodal Park and Ride and:

- explains key concepts
- provides examples
- documents resources and query parameters
- documents response structures and data types

What Is Intermodal Park and Ride?

HERE Intermodal Park and Ride is a REST API that allows you to plan a trip using different modes of transportation in the same journey to deliver optimized routes that include driving via car, parking at a dedicated parking lot, and continuing to your final destination via public transit. Ultimately, Intermodal Park and Ride assists vehicle drivers to use Park and Ride services to reduce congestion in Urban Areas and lower emissions; resulting in an environmentally friendly solution.

Note: Intermodal Park and Ride is only available for selected cities. See *Coverage Information* for more information. Currently, the city coverage does not include real-time coverage.

Why Use Intermodal Park and Ride?

Intermodal Park and Ride provides resources to address the following high-level use cases:

Table 1: Main features of HERE Intermodal Park and Ride

Feature	Description
Park late	Drive to the limits of your destination city and find a parking lot as soon as possible upon entering the city. This way, vehicle drivers can plan their trips or commute, to shorten the amount of time it takes to find a car park as close to their destination as possible.
Park early	Find a parking lot and switch to public transportation as quickly as possible. This way, vehicle drivers can avoid routes with congestion or construction, to ensure a more reliable Estimated Time of Arrival (ETA).

Chapter 2
Quick Start

Topics:

This section illustrates how a basic request with Intermodal Park and Ride works.

Routes for Park And Ride

To request routes for Park and Ride modes departing from the vicinity of Chicago into the inner city of Chicago at 7.30am, on 2017-12-18T, send the following GET request.

```
https://mobility.cit.api.here.com/v1/route
    ?app_id={YOUR_APP_ID}
    &app_code={YOUR_APP_CODE}
    &profile=parkandride
    &dep=42.59452409,-87.825373057
    &arr=41.86851845,-87.657831553
    &time=2017-12-18T07:30:00
```

The response is according to the following structure. For a full response, see *Park and Ride Routing Example*.

```
{
 "Res": {
  "serviceUrl": "https://mobility.cit.api.here.com/smartmob/v1/route.json",
  "Connections": {
   "context": "",
   "Connection": [{
    "id": "R000015-C2",
    "duration": "PT2H11M0S",
    "transfers": 2,
    "Dep": {
     "time": "2017-12-18T07:30:00",
     "Addr": {
      "x": -87.8253731,
      "y": 42.5944448
     }
    },
    "Arr": {
     "time": "2017-12-18T09:41:00",
     "Addr": {
      "x": -87.657824,
      "y": 41.868518
     }
    },
    "Sections": {
     "Sec": [{
      "mode": 21,
      "id": "R000015-C2-S0",
      "Dep": {
       "time": "2017-12-18T07:30:00",
       "Addr": {
        "x": -87.8253731,
        "y": 42.5944448
       },
       "Transport": {
        "mode": 21
       }
      },
      "Journey": {
       "distance": 80383,
       "duration": "PT1H3M9S"
      },
      "Arr": {
       "time": "2017-12-18T08:33:09",
       "RT": {
        "arr": "2017-12-18T08:38:27"
       },
```

```
        "Addr": {
         "x": -87.7967599,
         "y": 42.05892,
         "name": "Metra Golf Station",
         "At": {
          "category": "parking",
          "parkingId": "840bma4v-b05ca11a8a7f0b43e397a72721e958aa",
          "PnR": 1,
          "openingHours": "Mo-Su 00:00-24:00",
          "spaces": 110
         }
        },
        "Activities": {
         "Act": [{
          "duration": "PT5M",
          "type": "parking"
         }]
        }
       }
      }, {
       "mode": 20,
       "id": "R000015-C2-S1",
       "Dep": {
        "time": "2017-12-18T08:38:09",
        "Addr": {
         "x": -87.796752,
         "y": 42.058918
        },
        "Transport": {
         "mode": 20
        }
       },
       "Journey": {
        "duration": "PT3M",
        "distance": 92
       },
       "Arr": {
        "time": "2017-12-18T08:41:09",
        "Stn": {
         "id": "718820327#100",
         "name": "Golf",
         "x": -87.796941,
         "y": 42.058334
        },
        "AP": {
         "x": -87.796941,
         "y": 42.058334,
         "id": "718820819#100",
         "name": "Golf"
        }
       }
      }, {
       "mode": 3,
       "id": "R000015-C2-S2",
       "Dep": {
        "time": "2017-12-18T08:43:00",
        "Stn": {
         "id": "718820327#100",
         "name": "Golf",
         "x": -87.796941,
         "y": 42.058334
        },
        "Transport": {
         "mode": 3,
         "dir": "Chicago Union Station",
         "name": "Milwaukee North",
         "At": {
```

```
          "operator": "43AMET00",
          "color": "#CC5500",
          "textColor": "#FFFFFF",
          "category": "Regional Train"
         }
        },
        "Freq": {
         "min": 59,
         "max": 60,
         "AltDep": [{
          "time": "2017-12-18T09:43:00",
          "Transport": {
           "mode": 3,
           "dir": "Chicago Union Station",
           "name": "Milwaukee North",
           "At": {
            "operator": "43AMET00",
            "color": "#CC5500",
            "textColor": "#FFFFFF",
            "category": "Regional Train"
           }
          }
         }, {
          "time": "2017-12-18T10:42:00",
          "Transport": {
           "mode": 3,
           "dir": "Chicago Union Station",
           "name": "Milwaukee North",
           "At": {
            "operator": "43AMET00",
            "color": "#CC5500",
            "textColor": "#FFFFFF",
            "category": "Regional Train"
           }
          }
         }, {
          "time": "2017-12-18T11:42:00",
          "Transport": {
           "mode": 3,
           "dir": "Chicago Union Station",
           "name": "Milwaukee North",
           "At": {
            "operator": "43AMET00",
            "color": "#CC5500",
            "textColor": "#FFFFFF",
            "category": "Regional Train"
           }
          }
         }]
        }
       },
       "Journey": {
        "duration": "PT39M",
        "Stop": [{
         "dep": "2017-12-18T08:43:00",
         "Stn": {
          "id": "718820327#100",
          "name": "Golf",
          "x": -87.796941,
          "y": 42.058334
         }
        }, {
         "dep": "2017-12-18T08:46:00",
         "Stn": {
          "id": "718820389#100",
          "name": "Morton Grove",
          "x": -87.785273,
```

```
          "y": 42.034998
         }
        }, {
         "dep": "2017-12-18T08:51:00",
         "Stn": {
          "id": "718820307#100",
          "name": "Edgebrook",
          "x": -87.76555,
          "y": 41.997774
         }
        }, {
         "dep": "2017-12-18T08:54:00",
         "Stn": {
          "id": "718820316#100",
          "name": "Forest Glen",
          "x": -87.755554,
          "y": 41.978051
         }
        }, {
         "dep": "2017-12-18T08:57:00",
         "Stn": {
          "id": "718820379#100",
          "name": "Mayfair",
          "x": -87.745828,
          "y": 41.959722
         }
        }, {
         "dep": "2017-12-18T08:59:00",
         "Stn": {
          "id": "718820329#100",
          "name": "Grayland",
          "x": -87.740273,
          "y": 41.94889
         }
        }, {
         "dep": "2017-12-18T09:02:00",
         "Stn": {
          "id": "718820340#100",
          "name": "Healy",
          "x": -87.727778,
          "y": 41.924718
         }
        }, {
         "dep": "2017-12-18T09:08:00",
         "Stn": {
          "id": "718820448#100",
          "name": "Western Ave",
          "x": -87.688054,
          "y": 41.889166
         }
        }, {
         "arr": "2017-12-18T09:22:00",
         "Stn": {
          "id": "718820303#100",
          "name": "Chicago Union Station",
          "x": -87.638883,
          "y": 41.878891
         }
        }]
       },
       "Arr": {
        "time": "2017-12-18T09:22:00",
        "Stn": {
         "id": "718820303#100",
         "name": "Chicago Union Station",
         "x": -87.638883,
         "y": 41.878891
```

```
        }
       }
      }, {
       "mode": 20,
       "id": "R000015-C2-S3",
       "Dep": {
        "time": "2017-12-18T09:22:00",
        "Stn": {
         "id": "718820303#100",
         "name": "Chicago Union Station",
         "x": -87.638883,
         "y": 41.878891
        },
        "Transport": {
         "mode": 20
        },
        "AP": {
         "x": -87.641041,
         "y": 41.878361,
         "id": "718820617#100",
         "name": "Chicago Union Station"
        }
       },
       "Journey": {
        "duration": "PT6M",
        "distance": 214,
        "_guide": 0
       },
       "Arr": {
        "time": "2017-12-18T09:28:00",
        "Stn": {
         "id": "418804488#100",
         "name": "Clinton + Jackson",
         "x": -87.641149,
         "y": 41.878136
        }
       }
      }, {
       "mode": 5,
       "id": "R000015-C2-S4",
       "Dep": {
        "time": "2017-12-18T09:28:00",
        "Stn": {
         "id": "418804488#100",
         "name": "Clinton + Jackson",
         "x": -87.641149,
         "y": 41.878136
        },
        "Transport": {
         "mode": 5,
         "dir": "Cicero/24th Pl",
         "name": "60",
         "At": {
          "operator": "1gA48700",
          "category": "Bus"
         }
        },
        "Freq": {
         "min": 8,
         "max": 15,
         "AltDep": [{
          "time": "2017-12-18T09:36:00",
          "Transport": {
           "mode": 5,
           "dir": "Roosevelt/Racine",
           "name": "60",
           "At": {
```

```
            "operator": "1gA48700",
            "category": "Bus"
           }
          }
         }, {
          "time": "2017-12-18T09:45:00",
          "Transport": {
           "mode": 5,
           "dir": "Cicero/24th Pl",
           "name": "60",
           "At": {
            "operator": "1gA48700",
            "category": "Bus"
           }
          }
         }, {
          "time": "2017-12-18T09:54:00",
          "Transport": {
           "mode": 5,
           "dir": "Roosevelt/Racine",
           "name": "60",
           "At": {
            "operator": "1gA48700",
            "category": "Bus"
           }
          }
         }]
        }
       },
       "Journey": {
        "duration": "PT8M",
        "Stop": [{
         "dep": "2017-12-18T09:28:00",
         "Stn": {
          "id": "418804488#100",
          "name": "Clinton + Jackson",
          "x": -87.641149,
          "y": 41.878136
         }
        }, {
         "dep": "2017-12-18T09:29:00",
         "Stn": {
          "id": "418804490#100",
          "name": "Clinton Blue Line Station",
          "x": -87.641077,
          "y": 41.875592
         }
        }, {
         "dep": "2017-12-18T09:30:00",
         "Stn": {
          "id": "418804491#100",
          "name": "Clinton + Harrison",
          "x": -87.641041,
          "y": 41.874514
         }
        }, {
         "dep": "2017-12-18T09:30:00",
         "Stn": {
          "id": "418804492#100",
          "name": "Harrison + Jefferson",
          "x": -87.642245,
          "y": 41.874424
         }
        }, {
         "dep": "2017-12-18T09:31:00",
         "Stn": {
          "id": "418800165#100",
```

```
          "name": "Harrison + Halsted",
          "x": -87.647747,
          "y": 41.874442
         }
        }, {
         "dep": "2017-12-18T09:32:00",
         "Stn": {
          "id": "418800166#100",
          "name": "900 W Harrison",
          "x": -87.649859,
          "y": 41.874478
         }
        }, {
         "dep": "2017-12-18T09:32:00",
         "Stn": {
          "id": "418800167#100",
          "name": "Harrison + Morgan",
          "x": -87.651585,
          "y": 41.87446
         }
        }, {
         "dep": "2017-12-18T09:34:00",
         "Stn": {
          "id": "418804493#100",
          "name": "Racine + Harrison",
          "x": -87.656889,
          "y": 41.873938
         }
        }, {
         "dep": "2017-12-18T09:35:00",
         "Stn": {
          "id": "418810056#100",
          "name": "Racine + Lexington",
          "x": -87.656925,
          "y": 41.872347
         }
        }, {
         "dep": "2017-12-18T09:35:00",
         "Stn": {
          "id": "418810057#100",
          "name": "Racine + Cabrini",
          "x": -87.65688,
          "y": 41.871179
         }
        }, {
         "dep": "2017-12-18T09:36:00",
         "Stn": {
          "id": "418809979#100",
          "name": "Racine + Taylor",
          "x": -87.656835,
          "y": 41.869165
         }
        }]
       },
       "Arr": {
        "time": "2017-12-18T09:36:00",
        "Stn": {
         "id": "418809979#100",
         "name": "Racine + Taylor",
         "x": -87.656835,
         "y": 41.869165
        }
       }
      }, {
       "mode": 20,
       "id": "R000015-C2-S5",
       "Dep": {
```

```
        "time": "2017-12-18T09:36:00",
        "Stn": {
         "id": "418809979#100",
         "name": "Racine + Taylor",
         "x": -87.656835,
         "y": 41.869165
        },
        "Transport": {
         "mode": 20
        }
       },
       "Journey": {
        "duration": "PT5M",
        "distance": 229
       },
       "Arr": {
        "time": "2017-12-18T09:41:00",
        "Addr": {
         "x": -87.657824,
         "y": 41.868518
        }
       }
      }]
     },
     "Tariff": {
      "Fares": [{
       "Fare": [{
        "name": "Parking costs",
        "reason": "parking",
        "estimated": 1,
        "price": 2.0,
        "type": "daily",
        "currency": "USD",
        "sec_ids": "R000015-C2-S0"
       }, {
        "name": "Single Ride, with no transfer, Metra",
        "currency": "USD",
        "price": 6.0,
        "sec_ids": "R000015-C2-S2"
       }]
      }]
     }
    },
    ... more options ...
    ],
    "Operators": {
     "Op": [{
      "code": "43AMET00",
      "name": "Metra",
      "type": "RT",
      "Link": [{
       "type": "website",
       "href": "http://www.metrarail.com/",
       "text": "Metra"
      }]
     }, {
      "code": "1gA48700",
      "name": "Chicago Transit Authority",
      "type": "TT",
      "Link": [{
       "type": "website",
       "href": "http://transitchicago.com",
       "text": "Chicago Transit Authority"
      }],
      "At": {
       "phone": "1-888-YOURCTA"
      }
```

```
      }]
    }
   }
  }
 }
```

Related Information

- *Response Data Types*
- *Park and Ride Routing*

Chapter 3
Guide

Topics:

The articles in this section provide a guide to using Intermodal Park and Ride. The guide describes a common use case for the API and illustrates it with a practical example.

Key Concepts

This section provides information intended to help you understand and use Intermodal Park and Ride.

Transit Coverage Types

Intermodal Park and Ride offers two different levels of coverage for public transit: real-time (with limited geographic coverage) and timetable.

- Real-Time

  Real-time information ensures that the service considers the vehicle's current location or any sudden service disruptions, to provide an up to date Expected Time of Arrival (ETA) for the Transit services at a particular station. In the context of routing, real-time information is considered within the route and journey calculation process. However, the vehicle's current location messages alone cannot provide enough information to the user; this information needs to be matched with relevant journey information to be useful.
- Timetable

  The Park and Ride service sources detailed transit schedules from transit agencies. This information may come in the form of actual times, such as: the train leaves the station at 09:18, or scheduled frequencies, such as: the bus runs every 20 minutes between the hours of 08:00 and 10:40 AM. In addition, timetable transit coverage areas provide a more complete representation of the public transit networks available in a given city or metropolitan area.

Copyright Notices

In certain cases, data provided by Transit Agencies cannot be used without displaying the correct copyright notices to the end user. If the routing response contains this type of data, the service includes operator attribution information to the route response.

Operator attribution must be displayed together with a route. This requirement forms part of the terms and conditions of the API.

Example: Operator Attribution in Route Response

If copyright information is available, the response contains an `Attributions` element, with a ready-to-display string, which must be displayed in full to end users. The string is in the `Link` element identified with the attribute key/value pair `type="agency"`. In addition, possible tariff information is in the `Link` element identified with the attribute key/value pair `type="tariff"`. For more information, refer to the *Attributions* type. The `sec_ids` attributes represent the relevant route segment(s) of the trip for which the copyright notices apply.

The example below shows a response with copyright information.

```
<Attributions>
 <Link type="agency" href="http://example-company-a.com" sec_ids="R000010-C2-S3 R000010-C2-S1
 R000010-C0-S1">Company A</Link>
```

```
  <Link type="tariff" href="http://example-company-a.com/tariff" sec_ids="R000010-C2-S3 R000010-C2-S1
  R000010-C0-S1">Tariff</Link>
 </Attributions>
```

Acquiring Credentials

All users of HERE APIs must obtain authentication and authorization credentials and provide them as values for the parameters `app_id` and `app_code`. The credentials are assigned per application.

This document uses the placeholder text `{YOUR_APP_ID}` and `{YOUR_APP_CODE}` as placeholders for access and authorization credentials. Replace these placeholders with your own unique application-specific credentials to access the API resources.

To obtain the credentials for an application, please visit *http://developer.here.com/plans* to register with HERE.

If you wish to explore the API, use the API Explorer at *https://developer.here.com/api-explorer*.

Constructing a Request

A request to the Intermodal Park and Ride includes the basic elements shown in the following table and may contain resource-specific parameters or data.

Table 2: Basic Request Elements

Element	Value/Example	Description
Base URL	`https://mobility.api.here.com`	Production environment
	`https://mobility.cit.api.here.com`	CIT environment
		For information about the available evironments, see *HERE Server Environments*.
Path	`/v1/`	
Resource	*route*	
Application Code	`&app_code={YOUR_APP_CODE}`	Substitute your own unique `app_code`
Application ID	`&app_id={YOUR_APP_ID}`	Substitute your own unique `app_id`

HERE Server Environments

HERE provides two server environments for handling your requests: a Production environment and a Customer Integration Testing (CIT) environment.

You are required to use the CIT Environment when evaluating our products via our 90-day free trial.

To access CIT for REST APIs, amend the base URL to include an additional `cit` segment. For example, the CIT URL for this API is `https://mobility.cit.api.here.com`.

HERE examples and demos use this CIT environment only to provide an illustration of how the service operates.

You are required to use the Production environment for general production use. The CIT environment must **not** be used for production.

High Loads and Performance Testing

Neither standard server environment (CIT and Production) is designed to support high loads or performance testing. You must not conduct performance tests against the CIT or Production environments. If you need to do performance testing, *contact HERE* to discuss options.

Example

This section provides examples of requests reflecting typical use cases for Intermodal Park and Ride, along with the corresponding results.

For readability, in some responses, line breaks are used and long strings are shortened with ellipsis (...).

Park and Ride Routing Example

User Story

The user wants to search for a park and ride route departing from 63, Glenmore Drive, West Vancouver (49.36446767,-123.119102027) and arriving at 2 SW Marine Drive, Vancouver (49.21229508,-123.106722459).

Request

```
https://mobility.cit.api.here.com/v1/route.json
    ?app_id={YOUR_APP_ID}
    &app_code={YOUR_APP_CODE}
    &profile=parkandride
    &dep=49.36446767,-123.119102027
    &arr=49.21229508,-123.106722459
    &time=2017-12-18T07%3A30%3A00
    &details=0
```

Response

The response to the request contains the following information blocks:

- A connections or journey directing the vehicle driver to a parking lot.
- The journey starts with a driving route segment and ends with a public transit route, making use of an appropriate parking lot close to a public transit station.
- Parking related information:
  - Estimated time it takes to park the car
  - Estimated time to walk from the parking lot to a nearby public transit station
- Parking lot specific information:

    - Location
    - Whether the parking lot is designated specifically for Park and Ride (optional)
- A list of operators who run the public transit services included in the journeys, such as the operator code, operator name, and links to operator websites, if available.


```
{
 "Res": {
  "serviceUrl": "https://mobility.cit.api.here.com/smartmob/v1/route.json",
  "Connections": {
   "context": "",
   "Connection": [{
    "id": "R000023-C0",
    "duration": "PT0H58M0S",
    "transfers": 1,
    "Dep": {
     "time": "2017-12-18T07:30:00",
     "Addr": {
      "x": -123.1192379,
      "y": 49.3642824
     }
    },
    "Arr": {
     "time": "2017-12-18T08:28:00",
     "Addr": {
      "x": -123.106713,
      "y": 49.212291
     }
    },
    "Sections": {
     "Sec": [{
      "mode": 21,
      "id": "R000023-C0-S0",
      "Dep": {
       "time": "2017-12-18T07:30:00",
       "Addr": {
        "x": -123.1192379,
        "y": 49.3642824
       },
       "Transport": {
        "mode": 21
       }
      },
      "Journey": {
       "distance": 11944,
       "duration": "PT0H19M59S"
      },
      "Arr": {
       "time": "2017-12-18T07:49:59",
       "RT": {
        "arr": "2017-12-18T07:52:39"
       },
       "Addr": {
        "x": -123.11751,
        "y": 49.28161,
        "name": "Vancouver Centre",
        "At": {
         "category": "parking",
         "parkingId": "124bma4v-9fbb59b615320f52562db71267c44e64",
         "openingHours": "Mo-Su 00:00-24:00",
         "spaces": 550
        }
       },
       "Activities": {
        "Act": [{
```

```
          "duration": "PT5M",
          "type": "parking"
         }]
        }
       }
      }, {
       "mode": 20,
       "id": "R000023-C0-S1",
       "Dep": {
        "time": "2017-12-18T07:54:59",
        "Addr": {
         "x": -123.1175,
         "y": 49.281607
        },
        "Transport": {
         "mode": 20
        }
       },
       "Journey": {
        "duration": "PT4M",
        "distance": 156
       },
       "Arr": {
        "time": "2017-12-18T07:58:59",
        "Stn": {
         "id": "414508954#100",
         "name": "Vancouver City Centre Station",
         "x": -123.118408,
         "y": 49.282488
        }
       }
      }, {
       "mode": 3,
       "id": "R000023-C0-S2",
       "Dep": {
        "time": "2017-12-18T07:58:00",
        "Stn": {
         "id": "414508954#100",
         "name": "Vancouver City Centre Station",
         "x": -123.118408,
         "y": 49.282488
        },
        "Transport": {
         "mode": 3,
         "dir": "Canada Line TO Yvr-Airport",
         "name": "Canada Line Skytrain",
         "At": {
          "operator": "6kySKY00",
          "color": "#005CA9",
          "textColor": "#FFFFFF",
          "category": "Regional Train"
         }
        },
        "Freq": {
         "min": 1,
         "max": 4,
         "AltDep": [{
          "time": "2017-12-18T08:01:00",
          "Transport": {
           "mode": 3,
           "dir": "Canada Line TO Richmond-Brighouse",
           "name": "Canada Line Skytrain",
           "At": {
            "operator": "6kySKY00",
            "color": "#005CA9",
            "textColor": "#FFFFFF",
            "category": "Regional Train"
```

```
          }
         }
        }, {
         "time": "2017-12-18T08:03:00",
         "Transport": {
          "mode": 3,
          "dir": "Canada Line TO Yvr-Airport",
          "name": "Canada Line Skytrain",
          "At": {
           "operator": "6kySKY00",
           "color": "#005CA9",
           "textColor": "#FFFFFF",
           "category": "Regional Train"
          }
         }
        }, {
         "time": "2017-12-18T08:05:00",
         "Transport": {
          "mode": 3,
          "dir": "Canada Line TO Yvr-Airport",
          "name": "Canada Line Skytrain",
          "At": {
           "operator": "6kySKY00",
           "color": "#005CA9",
           "textColor": "#FFFFFF",
           "category": "Regional Train"
          }
         }
        }]
       }
      },
      "Journey": {
       "duration": "PT15M"
      },
      "Arr": {
       "time": "2017-12-18T08:13:00",
       "Stn": {
        "id": "414508965#100",
        "name": "Marine Drive Station",
        "x": -123.117033,
        "y": 49.209801
       }
      }
     }, {
      "mode": 20,
      "id": "R000023-C0-S3",
      "Dep": {
       "time": "2017-12-18T08:13:00",
       "Stn": {
        "id": "414508965#100",
        "name": "Marine Drive Station",
        "x": -123.117033,
        "y": 49.209801
       },
       "Transport": {
        "mode": 20
       }
      },
      "Journey": {
       "duration": "PT15M",
       "distance": 857
      },
      "Arr": {
       "time": "2017-12-18T08:28:00",
       "Addr": {
        "x": -123.106713,
        "y": 49.212291
```

```
       }
      }
     }]
    },
    "Tariff": {
     "Fares": [{
      "Fare": [{
       "name": "Parking costs",
       "reason": "parking",
       "estimated": 1,
       "price": 7.5,
       "type": "hourly",
       "currency": "CAD",
       "sec_ids": "R000023-C0-S0"
      }]
     }, {
      "Fare": [{
       "name": "Parking costs",
       "reason": "parking",
       "estimated": 1,
       "price": 22.5,
       "type": "daily",
       "currency": "CAD",
       "sec_ids": "R000023-C0-S0"
      }]
     }]
    }
   },
   ... more options ...
   ],
   "Operators": {
    "Op": [{
     "code": "6kySKY00",
     "name": "British Columbia Rapid Transit Company",
     "type": "TT",
     "Link": [{
      "type": "website",
      "href": "http://www.translink.ca/en/About-Us/Corporate-Overview/Operating-Companies/
BCRTC.aspx",
      "text": "British Columbia Rapid Transit Company"
     }]
    }, {
     "code": "6kyCMB00",
     "name": "TransLink",
     "type": "TT",
     "Link": [{
      "type": "website",
      "href": "http://www.translink.ca",
      "text": "TransLink"
     }]
    }]
   },
   "Attributions": {
    "Link": [{
     "type": "agency",
     "href": "http://www.translink.ca",
     "sec_ids": "R000023-C0-S2 R000023-C1-S2 R000023-C2-S2 R000023-C2-S4",
     "text": "TransLink"
    }]
   }
  }
 }
}
```

Related Information

- *Response Data Types*
- *Park and Ride Routing*

Service Support

If you need assistance with this or any other HERE product, select one of the following options.

- If you have a HERE representative, contact them when you have questions/issues.
- If you manage your applications and accounts through *developer.here.com*, log into your account and check the pages on the SLA report or API Health. If this does not clarify the issue, then check *stackoverflow.com/questions/tagged/here-api*.
- If you have an evaluation plan, check *stackoverflow.com/questions/tagged/here-api*.
- If you have questions about billing or your account, *Contact Us*.
- If you have purchased your plan/product from a HERE reseller, contact your reseller.

Chapter 4
API Reference

Topics:

This section provides descriptions of the resources, parameters, return types and error codes of HERE Intermodal Park and Ride.

Park and Ride Routing

Use the `route` resource to request routes between two sets of WGS 84-compliant geocoordinates using a `GET` request.

```
https://mobility.cit.api.here.com/v1/route
    ?parameter=value
    &...
```

Table 3: Mandatory Parameters

Parameter	Data Type	Description
profile	String	Set this parameter to *parkandride* to enable Park and Ride routing.
app_id	String	A 20 bytes Base64 URL-safe encoded string used for the authentication of the client application. You must include an `app_id` and `app_code` with every request. To get an `app_code` assigned to you, please see *Acquiring Credentials* on page 22.
app_code	String	A 20 bytes Base64 URL-safe encoded string used for the authentication of the client application. You must include an `app_id` and `app_code` with every request. To get an `app_code` assigned to you, please see *Acquiring Credentials* on page 22.
dep	String	Specifies the latitude, longitude and (optional) descriptive text for the location of the starting point of your route. Latitude and longitude are given in WGS 84-compliant degrees. Latitude values are between -90 and 90, longitude values are between -180 and 180. Examples: • *with* descriptive text: 40.751036894211815,-73.99162351623576,NY Penn Station • *without* descriptive text: 40.751036894211815,-73.99162351623576
arr	String	Specifies the latitude, longitude and (optional) descriptive text for the location of the end point of your route. Latitude and longitude are given in WGS 84-compliant degrees. Latitude values are between -90 and 90, longitude values are between -180 and 180. Examples: • *with* descriptive text: 40.751036894211815,-73.99162351623576,NY Penn Station • *without* descriptive text: 40.751036894211815,-73.99162351623576
time	DateTime	Specifies the time in ISO 8601 (for example, 2016-06-22T06:36:40) format.

Table 4: Optional parameters for *Park and Ride* mode

Parameter	Data Type	Description
car_change_strategy	String	Strategy to apply for finding Park and Ride routes. Possible **String** values are: • *park_late* (default) - switch from car to public transit as soon as possible after entering the destination city's limits • *park_early* - try to switch from car to public transit as early as possible The default is *park_late*.
modes	String	Specifies the transport mode filter used to determine which types of transport should be excluded in the response. It is a string of the following format: • **-code_1,-code_2,...,-code_N** - disable specified vehicle types and enable everything else See *List of Transport Codes in Request and Response* for a complete list of supported vehicle types.
details	Integer	Requests the inclusion of a list of intermediate stops within the the transit segment part of the routes in the response. If enabled, the response includes stop/station names, WGS 84 geocoordinates, and the departure/arrival times at the stops. 1 (enabled), 0 (disabled) The default is 1.
graph	Integer	Requests coordinate pairs to allow the drawing of a polyline for the route. The possible values are: 1 (enabled), 0 (disabled) The default is 0.
maneuvers	Integer	Requests turn-by-turn walk maneuvers information in the response. 1 (enabled), 0 (disabled) The default is 0.
units	String	Units of measurement used, for example, in guidance instructions. The possible values are: • metric • imperial The default is *metric*.
lang	String	Specifies the language of the the maneuvers elements in the response. The value complies with the ISO 639-1 standard and defaults to en.

Transport Codes in Park and Ride Requests and Responses

The following table lists transport codes that you can use to filter which vehicle types to include in the Transit segment of a Park and Ride route.

Table 5: List of Transport Codes in Park and Ride Requests and Responses

Request Code	Response Code	Vehicle Type Description
high_speed_train	0	High-speed Trains
intercity_train	1	Intercity/EuroCity Trains
inter_regional_train	2	Inter-regional and fast trains
regional_train	3	Regional and other trains
city_train	4	City trains
bus	5	Buses
ferry	6	Boats/Ferries
subway	7	Metros/Subways/Underground
light_rail	8	Trams
private_bus	9	Ordered services/Private Buses
inclined	10	Inclined/Funiculars
aerial	11	Aerials/Cable Cars
bus_rapid	12	Rapid Buses
monorail	13	Monorails
flight	14	Airplanes

The following transport codes occur in the drive and walk segments of Park and Ride routing response. They cannot be used to filter vehicle types in the Transit segment.

Table 6: List of Transport Codes in Park and Ride response only

Response Code	Vehicle Type Description
20	Walk
21	Car

Response Data Types

This section documents the response data types supported by Intermodal Park and Ride resources.

AP

AP on page 32 contains all available information about an individual transit access point (i.e. entrance/exit of a tranist station).

Table 7: AP Type Attributes

Attribute	Always Included	Type	Description
x	Yes	*Coordinate* on page 54	Longitude coordinate
y	Yes	*Coordinate* on page 54	Latitude coordinate

Attribute	Always Included	Type	Description
name	-	String	Name of the access point
id	-	String	Id of the access point

Act

Act on page 33 specifies a particular activity related to a particular mode of transport.

Table 8: Act Type Attributes

Attribute	Always Included	Type	Description
type	Yes	*ActIds* on page 53	Type of activity to perform before or after the journey.
duration	Yes	*TimeDelta* on page 57	Estimated time required for the activity.

Activities

Activities on page 33 specifies a list of activities related to the journey.

Table 9: Activities Type Elements

Element	Always Included	Type	Description
Act	Yes	List of *Act* on page 33	Specifies a particular activity and the duration related to a mode of transport.

Addr

Addr on page 33 specifies an address using coordinates.

Table 10: Addr Type Attributes

Attribute	Always Included	Type	Description
x	Yes	*Coordinate* on page 54	Longitude coordinate.
y	Yes	*Coordinate* on page 54	Latitude coordinate.
name	-	String	Name of the place, if present.
country	-	String	Name of the country where the coordinates are located.
ccode	-	String	3 letter ISO 3166-1 country code.
state	-	String	State or region within the country.
city	-	String	Name of the city.
postal	-	String	Postal code.
district	-	String	District (example: 'Manhattan' in New York City).

Attribute	Always Included	Type	Description
street	-	String	Name of the street, square, boulevard, etc.
number	-	String	Civic number.

Table 11: Addr Type Elements

Element	Always Included	Type	Description
Transports	-	*Transports* on page 53	List of mobility transports that are available at this address.
Info	-	*Info* on page 43	Generic text content that can be shown to the final user.
At	-	List of *At* on page 37	Specifies a key value pair with further information about the address.

Alert

Alert on page 34 contains all available information about an individual transit alert or an announcement.

Note: The Alerts are sourced from transit agencies' API feeds and/or Twitter accounts (where available). If the alerts' origin is from Twitter, you are required to follow Twitter's general principles for presenting alerts correctly. We do not provide any data for Twitter's logos and links for Twitter's actions. The full guidance on the display requirements can be found on *Twitter Display Requirements*.

Table 12: Alert Type Attributes

Attribute	Always Included	Type	Description
origin	Yes	String	Source of the alert. Available values are: • `API` - received from API • `RSS` - received from RSS • `WEB` - obtained from WEB • `TWITTER` - received from Twitter • `INVITRO` - alert issued internally
severity	-	String	The severity value The following values are possible: • `DISRUPT` - The disruption exists but the severity is unknown • `INFO` - No disruption, just information • `LOW` - Low severity disruption • `MEDIUM` - Medium severity disruption • `HIGH` - High severity disruption

Attribute	Always Included	Type	Description
operator	Yes	String	Name of the operator of the public transit line.
id	Yes	String	Identifier for the alert. The `id` is a unique value.
valid_till	-	*DateTime* on page 54	Time in UTC until which the alert is valid.
valid_from	-	*DateTime* on page 54	Time in UTC from which the alert is valid.
url	-	String	URL which can be used by end-users to access the original alert. For example, for Twitter it would be an URL of the tweet.
sec_ids	-	IDREFS	Whitespace-separated list of section Ids of a connection relevant to the given Alert.

Table 13: Alert Type Elements

Element	Always Included	Type	Description
Info	Yes	*Info* on page 43	Contains information about the public transit line or lines, such as expected closures or special requirements.
Link	-	*Link* on page 44	Link to a web resource with additional information. *Link* on page 44 nodes do not contain any text if inside an *Alert* on page 34 node.
Transports	-	*Transports* on page 53	List of transports affected by the alert.
Branding	-	*Branding* on page 38	Some alert sources (like Twitter) have branding requirement and clients must comply to these, for example by showing branding info to user.

Alerts

Alerts on page 35 contains a list of alerts containing information about possible public transit alerts or other announcements.

Table 14: Alerts Type Elements

Element	Always Included	Type	Description
Alert	-	List of *Alert* on page 34	Contains information about an individual alert.

AltDep

AltDep on page 36 contains information about an alternative departure, including time. *Transport* on page 52 contains the service name of the alternative departure. If available, real time information is provided.

Table 15: AltDep Type Attributes

Attribute	Always Included	Type	Description
time	-	*DateTime* on page 54	Departure time

Table 16: AltDep Type Elements

Element	Always Included	Type	Description
RT	-	*RT* on page 47	Specifies the expected arrival or departure time and the platform.
Transport	-	*Transport* on page 52	List of attributes related to the transport.

Arr

Arr on page 36 contains destination of the current section/connection. Include an Address or a Station. If available real time information will also be provided.

Table 17: Arr Type Attributes

Attribute	Always Included	Type	Description
platform	-	String	Name or number of the platform at which a line stops.
time	-	*DateTime* on page 54	Arrival time in ISO 8601 (for example, 2017-12-25T01:23:45) format.

Table 18: Arr Type Elements

Element	Always Included	Type	Description
RT	-	*RT* on page 47	*RT* on page 47, when present, contains the actual arrival or departure time, and also the platform information (if available) for transit stations.
Addr	-	*Addr* on page 33	Specifies an address using coordinates.
Stn	-	*Stn* on page 50	Specifies various information about a stop or a station.
AP	-	*AP* on page 32	Contains all available information about an individual transit access point (i.e. entrance/exit of a tranist station).

Element	Always Included	Type	Description
Activities	-	*Activities* on page 33	Information about activities.

At

At on page 37 is an attribute within an list of attributes. Each of these individual attributes contains an id and a value. The id specifies the type of information related to the value.

Note: The representation of a list of attributes in JSON differ from the XML representation since in JSON all attributes are collapsed as key values of a single object.

The example below demonstrates an AttributesIds list with various information about a transport such as category, name of the operator or whether bikes are allowed on this particular transport mode, in this case, the train.

Figure 1: List of attributes from an XML response:

```
<Transport name="Tram 12" code="8" dir="Mitte, Am Kupfergraben">
  <At id="category">Tram</At>
  <At id="operator">BVG</At>
  <At id="bikeAllowed">1</At>
  <At id="barrierFree">1</At>
  <At id="color">#CC0A22</At>
  <At id="textColor">#FFFFFF</At>
  <At id="iconShape">Rectangle</At>
<Transport>
```

Figure 2: List of attributes from a JSON response:

```
"At": {
  "category": "Tram",
  "operator": "BVG",
  "bikeAllowed": 1,
  "barrierFree": 1,
  "color": "#CC0A22",
  "textColor": "#FFFFFF",
  "iconShape": "Rectangle"
}
```

Table 19: At Type Attributes

Attribute	Always Included	Type	Description
id	Yes	*AttributesIds* on page 53	Specifies the type of information as an id and the actual information as a value associated with that id.

Content type: String

Attributions

Attributions on page 37 contains information that the application is required to display to the end user such as operator/provider disclaimers and tariffs. See *Copyright Notices* on page 21 for more details.

Note: Only links with id `agency` and `tariff` are used in the *Attributions* on page 37 element.

Table 20: Attributions Type Elements

Element	Always Included	Type	Description
Link	Yes	List of *Link* on page 44	Specifies a link to a web resource.

Branding

Branding on page 38 contains a list of attributes related to the branding.

Table 21: Branding Type Elements

Element	Always Included	Type	Description
At	-	List of *At* on page 37	Specifies a key value pair with further information about the brand. See `IdListBranding` for allowed keys.

Connection

Connection on page 38 contains all available information about a complete route between a departure and a destination.

Table 22: Connection Type Attributes

Attribute	Always Included	Type	Description
id	Yes	String	Identifier of the connection.
duration	Yes	*TimeDelta* on page 57	Expected duration of the journey.
transfers	Yes	Integer	Number of transport changes to reach the destination.
ridable	-	*Boolean* on page 54	If false, then the connection based on realtime situation is not possible. The default is `1`.
has_alt	-	*Boolean* on page 54	Returns 1 if a realtime alternative for this connection exists. The default is `0`.
alt	-	*Boolean* on page 54	Returns 1 if the connection is a real-time alternative to another connection that is not feasable due to delays. This connection will take into account real-time information. The default is `0`.

Attribute	Always Included	Type	Description
first_last_mile	-	*Boolean* on page 54	If 0 this connection does not include the first and/or the last part of the trip, but only the long distance sections in between. This normally happens for providers that do not support first and last mile (such as BlaBlaCar). Also it can be set to 0, if the user specified first_last_mile=0 in the request. The default is 1.

Table 23: Connection Type Elements

Element	Always Included	Type	Description
Dep	Yes	*Dep* on page 40	Contains information about a departure and includes time, station, address, platform and real time information if available. The *Transport* on page 52 attributes provide information about the transport that need to be used for this departure.
Arr	Yes	*Arr* on page 36	Contains destination of the current section/connection. Include an Address or a Station. If available real time information will also be provided.
Sections	Yes	*Sections* on page 50	Specifies a list of connection segments, using different transport modes.
Tariff	-	*Tariff* on page 52	Specifies a list of various possible fare combinations a user can buy/pay for the requested connection.

Connections

Connections on page 39 contains a list of possible connections.

Table 24: Connections Type Attributes

Attribute	Always Included	Type	Description
context	Yes	String	Session key to be used in subsequent calls that require it (deprecated).
valid_until	-	String	Datetime which says that dataset has expired but was auto-projected.

Attribute	Always Included	Type	Description
sup_changes	-	*Boolean* on page 54	1 if in this area changing "changes" parameter is supported (response is affected by it), 0 otherwise. The default is 1.
sup_speed	-	*Boolean* on page 54	1 if in this area changing the speed is supported (response is affected by it), 0 otherwise. The default is 1.
sup_max_dist	-	*Boolean* on page 54	1 if in this area changing the max walking distance is supported (response is affected by it), 0 otherwise. The default is 1.
sup_prod	-	*Boolean* on page 54	1 if in this area selecting the feature type is supported (response is affected by it), 0 otherwise. The default is 1.

Table 25: Connections Type Elements

Element	Always Included	Type	Description
Connection	-	List of *Connection* on page 38	Contains all available information about a complete route between a departure and a destination.
Operators	-	*Operators* on page 47	Contains all available information about operators referenced in the response.
Attributions	-	*Attributions* on page 37	Contains information that the application is required to display to the end user such as operator/ provider disclaimers and tariffs. See *Copyright Notices* on page 21 for more details.

Dep

Dep on page 40 contains information about a departure and includes time, station, address, platform and real time information if available. The *Transport* on page 52 attributes provide information about the transport that need to be used for this departure.

Table 26: Dep Type Attributes

Attribute	Always Included	Type	Description
platform	-	String	Platform name/number where the transport stop.
time	-	*DateTime* on page 54	Departure time.

Table 27: Dep Type Elements

Element	Always Included	Type	Description
RT	-	*RT* on page 47	*RT* on page 47, when present, contains the actual arrival or departure time, and also the platform information (if available) for transit stations.
Stn	-	*Stn* on page 50	Specifies various information about a stop or a station.
Addr	-	*Addr* on page 33	Specifies an address using coordinates.
Transport	-	*Transport* on page 52	Specifies a list of attributes related to a transport.
AP	-	*AP* on page 32	Contains all available information about an individual transit access point (i.e. entrance/exit of a tranist station).
Freq	-	*Freq* on page 42	Contains the frequency of alternative departures.
Activities	-	*Activities* on page 33	Specifies a list of activities related to the journey.

Fare

Fare on page 41 type contains information about a single fare needed for one or more sections of this connection.

Table 28: Fare Type Attributes

Attribute	Always Included	Type	Description
name	Yes	String	Name of the fare.
currency	Yes	String	Local currency of the price compliant to ISO 4217.
price	Yes	*RPrice* on page 57	Price of the fare. **Note:** If the fare type is `range` this attribute includes minimum and maximum price separated with the minus ('-') sign.
sec_ids	Yes	IDREFS	Whitespace-separated list of section ids of a *Connection* on page 38 covered by the given *Fare* on page 41.
estimated	-	*Boolean* on page 54	Attribute value is `1` if the fare price is estimated, `0` if it's an exact value. The default is `0`.

Attribute	Always Included	Type	Description
type	-	*FareTypes* on page 54	Type of fare represented by this object. The client application is responsible to correctly visualize the fare model to the user. Since the supported fare types can be extended in the future, we recommend to hide the fare information when you encounter a fare type unknown to you. See *FareTypes* on page 54 for more details. The default is `value`.
reason	-	*FareReason* on page 54	Reason for the cost described in this *Fare* on page 41 element. The default is `ride`.

Table 29: Fare Type Elements

Element	Always Included	Type	Description
Link	-	List of *Link* on page 44	Specifies a link to a web resource.

Fares

Fares on page 42 contains a list of fares that together are sufficient to cover the connection costs.

Table 30: Fares Type Elements

Element	Always Included	Type	Description
Fare	Yes	List of *Fare* on page 41	*Fare* on page 41 type contains information about a single fare needed for one or more sections of this connection.

Freq

Freq on page 42 contains the frequency of alternative departures.

The `min` and `max` attributes specify the time intervals between departures of a particular connection. The response only contains this information if it is likely that the user cannot reach a stop in time for the departure specified in the response and may need to wait for the next one.

Table 31: Freq Type Attributes

Attribute	Always Included	Type	Description
min	-	Integer	Minimal number of minutes between transport scheduled departures.

Attribute	Always Included	Type	Description
max	-	Integer	Maximal number of minutes between transport scheduled departures.
minRT	-	Integer	Minimal number of minutes between expected transport real-time departures.
maxRT	-	Integer	Maximal number of minutes between expected transport real-time departures.

Table 32: Freq Type Elements

Element	Always Included	Type	Description
AltDep	-	List of *AltDep* on page 36	Contains the service name of the alternative departure. If available, real time information is provided.

Graph

A polyline is a list of geographic coordinates. The edge between two subsequent coordinates is alway assumed to be the shortest way between the two.

Note: This element is represented in lowercase in the JSON response.

Content type: String

Guidance

Guidance on page 43 contains a group of maneuvers for one of the journey sections.

Table 33: Guidance Type Elements

Element	Always Included	Type	Description
Maneuvers	Yes	List of *Maneuvers* on page 45	Group of maneuvers to guide the user in one of the journey sections.

Info

Info on page 43 is a generic text content that can be showed to the final user. For example a station can require to call a number to request the bus/taxi.

Content type: String

Instruction

Verbal description of a maneuver.

Content type: String

Journey

Journey on page 44 type provides details for a journey. This include traveling duration, distance and a list of intermediate stops for transit modes.

Table 34: Journey Type Attributes

Attribute	Always Included	Type	Description
duration	Yes	*TimeDelta* on page 57	Expected time to cover that distance.
intermediate	-	Integer	This attribute is set to 0 if for a specific transit journey we are not able to provide intermediate stops. **Note:** For other modes of transport beside transit, this parameter is not provided. The default is 1.
distance	-	Integer	Distance to cover.

Table 35: Journey Type Elements

Element	Always Included	Type	Description
Stop	-	List of *Stop* on page 51	Specifies station where the line stops.

Link

Link on page 44 specifies a link to a web resource.

Table 36: Link Type Attributes

Attribute	Always Included	Type	Description
href	-	String	An URL address the link is pointing to. This attribute can be missing if the content we want to show is embedded in the text.
type	Yes	*LinkIds* on page 56	Type of the link.
sec_ids	-	IDREFS	When present, the IDs in the string represent one or more specific sections in the route response (space separated ids of sections).
href_text	-	String	When present, this attribute describes the part of the text that need to be linked (clickable) to a web resource. If not present (default), the full node text needs to be linked.

Content type: String

Maneuver

Maneuver on page 45 contains maneuver needed to take during the jounrney.

Table 37: Maneuver Type Attributes

Attribute	Always Included	Type	Description
direction	Yes	String	Maneuver direction hint. Can be used to display the appropriate arrow icon for the maneuver.
action	Yes	String	Code that identifies the action for this maneuver. Does not always indicate a direction.
duration	Yes	*TimeDelta* on page 57	Describes the amount of time needed for a single maneuver.
next_road	-	String	Name of the next road in the route that the maneuver is heading toward.
next_number	-	String	Number of the road (such as A5, B49, etc.) towards which the maneuver is heading.
distance	-	Integer	Distance to the next maneuver, in meters (default).
traffic	-	Float	How much traffic affects to travel time in this part of trip. Value can range from 0 to any value. 0 means that traffic has no impact and 1 stands for doubled travel time. Values close to 1 and above should be rare extreme cases.

Table 38: Maneuver Type Elements

Element	Always Included	Type	Description
Instruction	Yes	*Instruction* on page 43	Verbal description of a maneuver.
Graph	Yes	*Graph* on page 43	A polyline is a list of geographic coordinates. The edge between two subsequent coordinates is alway assumed to be the shortest way between the two.

Maneuvers

Maneuvers on page 45 contains a group of maneuvers to guide the user in one of the journey section.

Maneuvers are not available for transit sections.

Table 39: Maneuvers Type Attributes

Attribute	Always Included	Type	Description
sec_ids	Yes	IDREFS	Whitespace-separated list of section ids where this maneuvers are relevant.

Table 40: Maneuvers Type Elements

Element	Always Included	Type	Description
Maneuver	-	List of *Maneuver* on page 45	Contains maneuver needed to take during the jounrney.

Message

Message specifies a message from the server, pay attention if level is E (Error) or F (fatal).

Note: The message string is not meant to be presented to the user and can change at any time. Please use only the content in code and subcode to generate a message for the user.

Table 41: Message Type Attributes

Attribute	Always Included	Type	Description
code	Yes	String	Internal message code. See *Error Codes* on page 58.
subcode	-	String	Currently, the NO_COV value implies that the specified coordinates are not related to any available region.
level	Yes	*Level* on page 56	Specifies the error level of the message.

Content type: String

Op

Op on page 46 contains information about a particular operator.

An operator is the entity operating a transport service. For example, BVG operates some transit lines in Berlin.

Table 42: Op Type Attributes

Attribute	Always Included	Type	Description
code	-	String	Unique code of the operator. Match one of the Transport/ At[@id='operator'] on the response.
name	Yes	String	Human readable long operator name running the line

Attribute	Always Included	Type	Description
type	-	String	Describe the kind of coverage available for this operator. The following values are possible: • *RT* on page 47: real-time routing • TT: timetable routing • SR: estimated routing.
short_name	-	String	Short operator name when available
fare	-	*Boolean* on page 54	Returns 1 if fare information is supported by the operator, 0 otherwise. The default is 0.

Table 43: Op Type Elements

Element	Always Included	Type	Description
Link	-	List of *Link* on page 44	Specifies a link to a web resource.
At	-	List of *At* on page 37	*At* on page 37 is an attribute within an list of attributes. Each of these individual attributes contains an id and a value. The id specifies the type of information related to the value.

Operators

Operators on page 47 contains all available information about operators referenced in the response.

Table 44: Operators Type Elements

Element	Always Included	Type	Description
Op	Yes	List of *Op* on page 46	Contains information about a particular operator.

RT

RT on page 47, when present, contains the actual arrival or departure time, and also the platform information (if available) for transit stations.

Table 45: RT Type Attributes

Attribute	Always Included	Type	Description
has_arr	-	*Boolean* on page 54	If 0 the arrival is cancelled. The default is 1.

Attribute	Always Included	Type	Description
has_dep	-	*Boolean* on page 54	If 0 the departure is cancelled. The default is 1.
arr	-	*DateTime* on page 54	Time when the transport arrives to a location or a station.
dep	-	*DateTime* on page 54	Time when the transit leaves a location or a station.
new_stop	-	*Boolean* on page 54	The stop is a new stop in the service route. The default is 0.
platform	-	String	Platform name/number where the transit line stops.
status	-	String	If the attribute is not present it's considered that the train is scheduled at the given arrival/ departure time and at the given platform. Possible other values are: • `redirected`: The line is not following the normal stops sequence • `replaced`: User should take a replacement service for this transit • `cancelled`: The service has a permanent failure and will not arrive and depart • `additional`: This is an additional not planned service

Res

Res on page 48 is a root response node and contains response data generated as a result of a request to the Park and Ride request resources.

Table 46: Res Type Attributes

Attribute	Always Included	Type	Description
serviceUrl	-	String	URL of the backend server that produced the results.

Table 47: Res Type Elements

Element	Always Included	Type	Description
Message	-	*Message* on page 46	Message specifies a message from the server, pay attention if level is E (Error) or F (fatal).

Element	Always Included	Type	Description
Connections	-	*Connections* on page 39	Contains a list of possible connections.
Guidance	-	*Guidance* on page 43	Contains a group of maneuvers for one of the journey sections.
Alerts	-	*Alerts* on page 35	Contains a list of alerts containing information about possible public transit alerts or other announcements.

Sec

Sec on page 49 represents a section of the entire connection. It contains information about departure, arrival and journey.

Table 48: Sec Type Attributes

Attribute	Always Included	Type	Description
uncertainty	-	Integer	The times specified a number of minutes of uncertainly. If the attribute is not present this section is based on time table.
id	-	ID	Unique identifier of a section. Other elements in the response may reference to this section using this ID (examples are *Fare* on page 41, *Maneuvers* on page 45, *Link* on page 44).
mode	Yes	Integer	Numerical code of the transport mode that will be used on this trip section. See *Transport Modes*.

Table 49: Sec Type Elements

Element	Always Included	Type	Description
Dep	Yes	*Dep* on page 40	Contains information about a departure and includes time, station, address, platform and real time information if available. The *Transport* on page 52 attributes provide information about the transport that need to be used for this departure.
Journey	Yes	*Journey* on page 44	*Journey* on page 44 type provides details for a journey. This include traveling duration, distance and a list of intermediate stops for transit modes.

Element	Always Included	Type	Description
Arr	Yes	*Arr* on page 36	Contains destination of the current section/connection. Include an Address or a Station. If available real time information will also be provided.
Graph	-	*Graph* on page 43	A polyline is a list of geographic coordinates. The edge between two subsequent coordinates is alway assumed to be the shortest way between the two.

Sections

Sections on page 50 specifies a list of connection segments, using different transport modes.

Table 50: Sections Type Elements

Element	Always Included	Type	Description
Sec	Yes	List of *Sec* on page 49	*Sec* on page 49 represents a section of the entire connection. It contains information about departure, arrival and journey.

Stn

Stn on page 50 specifies various information about a stop or a station.

Table 51: Stn Type Attributes

Attribute	Always Included	Type	Description
x	Yes	*Coordinate* on page 54	Longitude coordinate
y	Yes	*Coordinate* on page 54	Latitude coordinate
name	Yes	String	Name of the station
id	-	*StationId* on page 57	Unique id of the station
country	-	String	Name of the country where the coordinates are into.
ccode	-	String	3 letter ISO 3166-1 country code
state	-	String	State of the country
city	-	String	Name of the city
postal	-	String	Postal code
district	-	String	District (e.g. 'Manhattan' in New York)
street	-	String	Name of the street, square, boulevard,..
number	-	String	Civic number

Attribute	Always Included	Type	Description
distance	-	Integer	Distance in meters from the station, available only when querying for stations around you.
duration	-	*TimeDelta* on page 57	Expected time to reach the station from your current location, available only when querying stations around you.
has_board	-	Integer	When this attribute is 1, a station board (next departure) request can be performed. Only regions where timetable or real-time data supports the station board request. The default is 1.

Table 52: Stn Type Elements

Element	Always Included	Type	Description
Transports	-	*Transports* on page 53	List of public transit transports that depart/arrive to this station.
Info	-	*Info* on page 43	*Info* on page 43 is a generic text content that can be showed to the final user. For example a station can require to call a number to request the bus/taxi.
At	-	List of *At* on page 37	*At* on page 37 is an attribute within an list of attributes. Each of these individual attributes contains an id and a value. The id specifies the type of information related to the value.

Stop

Stop on page 51 specifies station where the line stops.

Table 53: Stop Type Attributes

Attribute	Always Included	Type	Description
arr	-	*DateTime* on page 54	Time when the transit arrives to the station.
dep	-	*DateTime* on page 54	Time when the transit leaves the station.

Table 54: Stop Type Elements

Element	Always Included	Type	Description
RT	-	*RT* on page 47	*RT* on page 47, when present, contains the actual arrival or departure time, and also the platform information (if available) for transit stations.
Stn	Yes	*Stn* on page 50	Specifies various information about a stop or a station.

Tariff

Tariff on page 52 specifies a list of various possible fare combinations a user can buy/pay for the requested connection.

The first *Fares* on page 42 element determines the cheapest combination of one or multiple fares suitable for the standard single trip. Further occurrences of *Fares* on page 42 might offer other options, like discounts groups or different season tickets.

Table 55: Tariff Type Elements

Element	Always Included	Type	Description
Fares	Yes	List of *Fares* on page 42	Contains a list of fares that together are sufficient to cover the connection costs.

Transport

Transport on page 52 specifies a list of attributes related to a transport.

Table 56: Transport Type Attributes

Attribute	Always Included	Type	Description
name	-	String	Name of the line for public transit modes or of the veicle for some other mode of transport. Walk mode don't include this attribute.
mode	Yes	Integer	Numerical code of the transport mode. See *Transport Modes*.
dir	-	String	Direction of the line, usually specified as the destination station.

Table 57: Transport Type Elements

Element	Always Included	Type	Description
At	-	List of *At* on page 37	*At* on page 37 is an attribute within an list of attributes. Each of these individual attributes contains an id and a value. The id specifies the type of information related to the value.
Link	-	List of *Link* on page 44	Specifies a link to a web resource.

Transports

Transports on page 53 contains a list of transports.

Table 58: Transports Type Elements

Element	Always Included	Type	Description
Transport	Yes	List of *Transport* on page 52	Specifies a list of attributes related to a transport.

Simple Types

This section documents simple data types used by the Intermodal Park and Ride.

ActIds

Specifies the type of an activity.

Type: String

Table 59: ActIds allowed values

Value	Description
wait	Waiting for something. For example the user may need to wait for a taxi to arrive.
setup	User's action needed (e.g. logging in and setting up a shared car prior to departing).
parking	Parking the vehicle at the destination or at a parking lot.

AttributesIds

Possible IDs for the At element. Depending on the parent element where the attributes appears, different IDs are allowed.

Note: The list of defined IDs can grow over time. The client application should ignore attributes with unknown IDs.

Type: *IdListTransport* on page 56 | *IdListOp* on page 55 | *IdListBranding* on page 55 | *IdListAddr* on page 55

Boolean

Number between 0 (considered False) and 1 (considered True)

Type: Integer

Color

Represent a color in the format #RRGGBB.

Type: String

Coordinate

Coordinate simple type is a floating point value in format WGS84.

Latitude values range between -90.0 and 90.0 degrees. Longitude values range between -180.0 and 180.0 degrees.

Type: Float

DateTime

DateTime on page 54 represents a point in time. The format follows the ISO 8601 specification for combined date and time representations. If timezone is not provided, the time represents a local time in the location identified by the elements *Dep* on page 40 and *Arr* on page 36 or UTC if specifically defined in the documentation.

Format examples: `2016-06-22T06:36:40`, `2016-06-22T06:36:40Z`, `2016-06-22T06:36:40-02:00`

Type: String

FareReason

Specifies the fare reason.

Type: String

Table 60: FareReason allowed values

Value	Description
ride	If the cost refer to the ride fee. It can be a transit ticket or taxi/car sharing fees.

FareTypes

Specifies the possible fare types.

Type: String

Table 61: FareTypes allowed values

Value	Description
value	The indicated price is the value we expect to be paid for this Fare element. Please look at the *Fare* on page 41`@estimated` attribute to see if this value is just an estimation or an exact value.
hourly	The indicated price is the cost per hour. Example: 11$/hour
daily	The indicated price is the cost per day. Example: 62$/day
range	The price represent a range with minimum and maximum value. In this case the format is as follow: "min-max". See *RangePrice* on page 57 type.

IdListAddr

Possible attributes IDs that can appear under the *Addr* on page 33 element.

Type: String

Table 62: IdListAddr allowed values

Value	Content Type	Description
category	String	If the address is a POI, this attribute will contain the category: `parking` - parking lot POI.
parkingId	String	Parking ID to call the Automotive Cloud Service Off Street parking API.
PnR	*Boolean* on page 54	If 1 this POI is a parking of type "Park and Ride", i.e. is a parking specifically designed to allow transition between car and transit.

IdListBranding

Possible attributes IDs that can appear under the *Branding* on page 38 element.

Type: String

Table 63: IdListBranding allowed values

Value	Content Type	Description
tweetId	String	Id of the tweet.
tweetTime	*DateTime* on page 54	Time of the tweet.
tweetFullName	String	Full name of the tweet's author.
tweetUser	String	User name of the tweet's author.
tweetAvatar	String	Avatar (image) of the tweet's author.

IdListOp

Possible attributes IDs that can appear under the *Op* on page 46 element.

Type: String

Table 64: IdListOp allowed values

Value	Content Type	Description
phone	String	Phone number provided by the operator.
email	String	E-mail address provided by the operator.

IdListTransport

Possible attributes IDs that can appear under the *Transport* on page 52 element.

Type: String

Table 65: IdListTransport allowed values

Value	Content Type	Description
category	String	Human readable transport category: for example: Bus, Gondola, Zug, CarShare, Taxi...
operator	String	Unique code of the operator running the transport. This code can reference the code attribute in the *Op* on page 46 element.
bikeAllowed	*Boolean* on page 54	If = 1, bikes are allowed onboard.
barrierFree	*Boolean* on page 54	If = 1, the stop is accessible for people with disabilities.
escalator	*Boolean* on page 54	If = 1, escalator is present on the current station.
elevator	*Boolean* on page 54	If = 1, elevator is present on the current station.
blindGuide	*Boolean* on page 54	If = 1 blind guidelines are present on the current station.
color	*Color* on page 54	Color of the transport and background for the transport name.
textColor	*Color* on page 54	Color of the transport name.
outlineColor	*Color* on page 54	Color of the border around the transport name.

Level

Specifies the logging level.

Type: String

Table 66: Level allowed values

Value	Description
M	Message
W	Warning
E	Error
F	Fatal

LinkIds

Specifies the type of a link.

Type: String

Table 67: LinkIds allowed values

Value	Description
appStore	Link to download the operator application in the store. The resulting url depends on the request parameter `device`.
tariff	Tariff information URL.
booking	Booking URL.
agency	Operator information URL.
website	Operator website URL.
logo	Partner logo URL.
alert	Link to the original source of a transit alert (eg. Twitter page). There is not text in the *Link* on page 44 element when the type is alert.
opIcon	Url to the Operator icon.
productIcon	Url to the Transport product icon. For a transit type this is normally the icon associated with the system. For a mobility provider this represent the characteristic of the product. When present, the link text refer to the product name.
rideIcon	Url to the Transport ride icon. For a transit type this is normally the graphical representation of the line name. For a mobility provider this represent the transport model. When present, the link text refer to the ride name.

RPrice

RPrice on page 57 type can be either a `float` value (price) or a `string` representing a price range.

Type: Float | *RangePrice* on page 57

RangePrice

Represent minumum and maximum price concatenated with a '-' sign. For example: "11-15" means that the price can vary between 11 and 15.

Type: String

StationId

Represent a station identifier.

Type: String

TimeDelta

Represent a fixed point in time from a start date or a duration. The format follow the ISO 8601 specification for durations.

Format: -?P(\d+Y)?(\d+M)?(\d+D)?T((\d+H)?(\d+M)?(\d+S)?)?

Type: String

Response Status Codes

In response to a request, Intermodal Park and Ride can return several types of status codes that indicate an error.

The following sections describe the error codes and the message format.

Error Codes

Intermodal Park and Ride supports the following standard *HTTP status error codes*

If the HTTP Status Code is 200, the response body can contain the following error messages (see *Message* on page 46):

Error code	Description
I4	Incorrect `app_code` or `app_id` in the request. See *Acquiring Credentials* on page 22 for more information
SS0007	No results found
GW100	Wrong parameter type or parameter missing
GW0001	Routing was not possible The following `subcodes` are available: • `NO_COV`: There is no transit coverage in the area • `NO_STN_NEARBY`: Nearby to the given address stations could not be found • `DEP_ARR_TOO_CLOSE`: Departure/Arrival are too near
GW0002	Backend failed to provide a response
GW0006	Unexpected error
GW0007	No support for this API in this area
K9360	Invalid Period

Error message response examples:

```
<?xml version='1.0' encoding='utf-8'?>
<Res serviceUrl="">
 <Message code="GW100" level="E">
  Invalid request: 2 is greater than the maximum of 1 in request/details
 </Message>
</Res>
```

```
<?xml version='1.0' encoding='utf-8'?>
<Res serviceUrl="">
 <Message code="GW0001" level="E" subcode="DEP_ARR_TOO_CLOSE">
  K895 - Departure/Arrival are too near.
 </Message>
</Res>
```

Message

Message specifies a message from the server, pay attention if level is E (Error) or F (fatal).

Note: The message string is not meant to be presented to the user and can change at any time. Please use only the content in code and subcode to generate a message for the user.

Table 68: Message Type Attributes

Attribute	Always Included	Type	Description
code	Yes	String	Internal message code. See *Error Codes* on page 58.
subcode	-	String	Currently, the NO_COV value implies that the specified coordinates are not related to any available region.
level	Yes	*Level* on page 56	Specifies the error level of the message.

Content type: String

Chapter 5
Coverage Information

Park and Ride routing is available as a service in the following cities.

Park and Ride Routing

The table below lists the cities in which Park and Ride routing is currently available.

Table 69: Cities with Park and Ride Routing

Administrative Area	Cities
Australia	Sydney
Canada	Toronto
France	Paris
Germany	Frankfurt, Munich, Stuttgart
Italy	Milan, Rome
Netherlands	Amsterdam
Spain	Barcelona, Madrid, Valencia
United Kingdom	Birmingham, London
United States	Chicago, New York City, Seattle