



Bundesamt für  
Kartographie und Geodäsie

# Routingdienste des BKG

---

## Schnittstellenbeschreibung

Version 1.0

10.11.2015

Dienstleistungszentrum  
Bundesamt für Kartographie und Geodäsie

## Inhalt

Abkürzungsverzeichnis.....	4
1 Überblick über den Dienst .....	5
2 Verwendete Software, Daten und Lizenzen.....	6
3 Datenmodell.....	7
3.1 Projektion.....	7
3.2 Ortsangaben .....	7
3.2.1 Breite (lat) .....	7
3.2.2 Länge (lon) .....	7
3.2.3 Geocodierung von Ortsangaben .....	7
4 Funktionen des Routingdienstes.....	8
4.1 Routingserver-API .....	8
4.2 Implementierte Dienste (Services) und Funktionen.....	9
4.3 Service locate (Knotenpunkt) .....	10
4.3.1 Anfrage.....	10
4.3.2 Antwort.....	10
4.3.3 Beispiel.....	10
4.4 Service nearest (Straßensegment).....	10
4.4.1 Anfrage.....	11
4.4.2 Antwort.....	11
4.4.3 Beispiel.....	11
4.5 Service viaroute (Standardroute) .....	11
4.5.1 Anfrage.....	12
4.5.2 Antwort.....	12
4.5.3 Beispiel.....	13
4.6 Service fr (schnellste Route) .....	15
4.6.1 Anfrage.....	15
4.6.2 Antwort.....	15
4.6.3 Beispiel.....	15
4.7 Erreichbarkeitsanalyse.....	16
4.7.1 Anfrage.....	16
4.7.2 Antwort.....	16
4.7.3 Beispiel.....	16
5 Allgemeines .....	18
5.1 Anwendungshinweise .....	18

5.2	Fehlermeldungen.....	18
6	Datenschutz.....	19
7	Quellenverzeichnis .....	20
8	Kontakt .....	21
8.1	Freischaltung.....	21
8.2	Technische Anfragen.....	21

## Abkürzungsverzeichnis

EPSG	European Petroleum Survey Group
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation
geoJSON	JavaScript Object Notation für geographische Objekte
pbf	Protocolbuffer Binary Format

## 1 Überblick über den Dienst

Der Routingdienst für die Ermittlung von Routen ist ein Projekt des Bundesamtes für Kartographie und Geodäsie (BKG) für die Bundesverwaltung. Als Datengrundlage für die Routenberechnung werden die frei verfügbaren Daten des Projektes OpenStreetMap verwendet. Das BKG bereitet die Daten vor und betreibt die im BKG entwickelten Webdienste.

Das vorliegende Dokument beschreibt die Schnittstellen des Routingdienstes und gibt Hinweise zu dessen Anwendung.

Ein *Routingdienst* ist ein Webdienst, der Wegberechnungen (Routen) zwischen mit Koordinaten versehenen Objekten unter Berücksichtigung von Geschwindigkeitsprofilen umsetzt und für eine Weiterverarbeitung zugänglich macht.

Die vorliegende Serverimplementierung stellt die folgenden Diensttypen bereit:

- *Route* (viaroute und fr)  
Der Routingdienst unterstützt die Ermittlung einer schnellsten und einer kürzesten Route für verschiedene Geschwindigkeitsprofile einschließlich der Ausgabe von Routinganweisungen.
- *Nächstsgelegenes navigierbares Straßensegment*  
Zu einer gegebenen Ortsangabe wird das am nächsten gelegene navigierbare Straßensegment ermittelt.
- *Umkreisanalysen auf Basis von Zeit- oder Entfernungsangaben*  
Zu einer definierten Ortsangabe wird eine Umkreisanalyse (Erreichbarkeitsanalyse) in Abhängigkeit von einer vorgegebenen Zeit oder vorgegebenen Entfernung eine Fläche rund um die Ortsangabe berechnet, welche die Erreichbarkeit darstellt.

Es findet keine persistente Speicherung von Daten statt.

Datengrundlage des Dienstes sind die Datenbestände des OpenSource-Projektes „OpenStreetMap“, da zum heutigen Zeitpunkt existierende amtliche Geobasisdaten keine Routingdaten enthalten und somit nicht routingfähig sind.

Der Routingdienst unterstützt die Schnittstelle geoJSON und wird bereitgestellt über die folgende URL

[http://sg.geodatenzentrum.de/web\\_routing](http://sg.geodatenzentrum.de/web_routing)

Im vorliegenden Dokument werden die Nutzung der Schnittstelle und das Ausgabeformat beschrieben.

## 2 Verwendete Software, Daten und Lizenzen

In der bereitgestellten Implementierung werden als Routing-Engine die Softwarepakete OpenStreetMapRoutingMachine v. 0.3.7 [1] und OSM2PO v. 5.0.0 [2] unter Anwendung und Beachtung ihrer jeweiligen Nutzungsbedingungen verwendet und zur Nutzung bereitgestellt.

Die Serverimplementierung ist eine Eigenentwicklung des Betreibers [3] (s. Kap 8).

Die Berechnung der Routeninformation erfolgt auf der Basis von OpenStreetMap-Daten, welche regelmäßig von einem OpenStreetMap-Daten-Spiegelserver [4] im Format pbf<sup>1</sup> für das Gebiet der Bundesrepublik Deutschland geladen werden. Deren Nutzung erfolgt unter Beachtung und Anwendung der Lizenz „Open Data Commons Open Database License (ODbL)“ [5]

---

<sup>1</sup> [http://wiki.openstreetmap.org/wiki/DE:PBF\\_Format](http://wiki.openstreetmap.org/wiki/DE:PBF_Format) (letzter Aufruf am 09.11.2015)

### 3 Datenmodell

Für die Berechnung der Routingfunktionen werden lediglich Koordinatenangaben benötigt. Diese können mit Hilfe eines Geocodierungsdienstes wie dem Geocoder des BKG ermittelt werden.

#### 3.1 Projektion

Derzeit wird nur die Projektion WGS84 geographisch unterstützt.

#### 3.2 Ortsangaben

##### 3.2.1 Breite (lat)

Die Angabe der geographischen Breite erfolgt in Dezimalschreibweise wie folgt

50.19991683959961

##### 3.2.2 Länge (lon)

Die Angabe der geographischen Länge erfolgt in Dezimalschreibweise wie folgt

8.500038146972656

##### 3.2.3 Geocodierung von Ortsangaben

Die für die Routenberechnung notwendigen Koordinatenwerte der Länge und Breite können z.B. von einem Geocodierungsdienst bereitgestellt werden.

Ortsangabe

→

Breite, Länge

Beispiel

Frankfurt/Main, Richard-Strauss-Alle-11 → 50.19991683, 8.50003814

## 4 Funktionen des Routingdienstes

### 4.1 Routingserver-API

Die Serveranfrage erfolgt in der folgenden Form:

```
http://{server-address}/{profile}/{service}?{parameter}={value}
```

Url des Routingserver

```
server-address = sg.geodatenzentrum.de/web_routing
```

Geschwindigkeitsprofile

```
Auto           profile = {car}
Fahrrad        profile = {bic}
Fußgänger      profile = {foot}
```

Routingfunktionen

```
Nächstgelegenes routingfähiges Straßensegment
service = {nearest}
```

```
Nächstgelegener routingfähiger Knotenpunkt
service = {locate}
```

Routenberechnung

```
Service = {viaroute}
Service = http://{server-address}/{o2/car/Osm2poService?cmd=fr}
```

Erreichbarkeitsanalysen

```
car.php        {dist|time}
bic.php        {dist|time}
foot.php       {dist|time}
```

Beispiele:

```
locate:
http://sg.geodatenzentrum.de/web_routing/car/locate?loc=52.4224,13.333086

Nearest:
http://sg.geodatenzentrum.de/web_routing/car/nearest?loc=52.4224,13.333086

Routeberechnung:
http://sg.geodatenzentrum.de/web_routing/bic/viaroute?loc=52.503033,13.420526&loc=52.516582,13.429290&loc=52.554070,13.160621&instructions=true

http://sg.geodatenzentrum.de/web_routing/o2/car/Osm2poService?cmd=fr&source=50.10014343261719,8.500038146972656&target=50.19991683959961,8.601035118103027&format=geojson

Erreichbarkeitsanalysen:
http://sg.geodatenzentrum.de/web_routing/bic.php?lat=49.8993&lon=8.84&time=0.2
```



```
http://sg.geodatenzentrum.de/web_routing/bic.php?lat=49.8993&lon=8.84&dist=10
```

Als Standardeinstellung liefert der Routingserver JSON-Objekte aus. Jedes JSON-Objekt hat den folgenden Aufbau:

```
{
  status: 0,
  status_message: "Message text",
  .....
}
```

Wobei

<b>status</b>	einen Statuscode liefert. (0 ... erfolgreiche Anfrage, 207 ... keine Route gefunden)
<b>status_message</b>	eine optionale Statusmeldung wie "Found route between points" oder "Cannot find route between points" enthalten kann.

## 4.2 Implementierte Dienste (Services) und Funktionen

In der vorliegenden Implementierung sind folgende Dienste und Funktionen enthalten:

Service (Dienst)	Beschreibung
<b>viaroute</b>	kürzeste Route zwischen zwei oder mehr mit Koordinaten beschriebenen Punkten
<b>fr</b>	schnellste Route zwischen zwei oder mehr mit Koordinaten beschriebenen Punkten
<b>locate</b>	Liefert den nächstgelegenen navigierbaren Knotenpunkt für einen mit Koordinaten beschriebenen Punkt
<b>nearest</b>	Liefert das nächstgelegene navigierbare Straßensegment für einen mit Koordinaten beschriebenen Punkt

Funktionen	Beschreibung
<b>car.php</b>	Erreichbarkeitsanalyse (Zeit oder Entfernung) für das Geschwindigkeitsprofil „car“ für einen mit Koordinaten beschriebenen Punkt (Auto)
<b>bic.php</b>	Erreichbarkeitsanalyse (Zeit oder Entfernung) für das Geschwindigkeitsprofil „bic“ für einen mit Koordinaten beschriebenen Punkt (Fahrrad)
<b>foot.php</b>	Erreichbarkeitsanalyse (Zeit oder Entfernung) für das Geschwindigkeitsprofil „foot“ für einen mit Koordinaten beschriebenen Punkt (Fußgänger)

### 4.3 Service locate (Knotenpunkt)

Der Dienst ermittelt den nächstgelegenen navigierbaren Knotenpunkt für einen mit Koordinaten beschriebenen Punkt.

#### 4.3.1 Anfrage

```
http://{server-address}/{profile}/locate?loc={lat,lon}
```

Parameter	Wert	Beschreibung
<b>loc</b>	lat,lon	Ein mit Koordinaten (Breite, Länge) beschriebener Punkt, Die Koordinatenangaben erfolgen im Koordinatensystem WGS84 in Dezimalschreibweise getrennt durch ein Komma (,)
<b>profile</b>	car, bic, foot	Geschwindigkeitsprofile Auto, Fahrrad oder Fußgänger

#### 4.3.2 Antwort

Parameter	Beschreibung
<b>status</b>	Statuscode
<b>mapped_coordinate</b>	Ein Array, welches die ermittelten Koordinaten enthält [ lat , lon ]

#### 4.3.3 Beispiel

Anfrage:

```
http://sg.geodatenzentrum.de/web_routing/car/locate?loc=52.4224,13.333086
```

Antwort:

```
{
  "version":0.3,
  "status":0,
  "mapped_coordinate":
  [52.422442,13.332101],
  "transactionId":
  "OSRM Routing Engine JSON Locate (v0.3)"
}
```

### 4.4 Service nearest (Straßensegment)

Der Dienst ermittelt das nächstgelegene navigierbare Straßensegment für einen mit Koordinaten beschriebenen Punkt.

---

### 4.4.1 Anfrage

```
http://{server-address}/{profile}/nearest?loc={lat,lon}
```

Parameter	Wert	Beschreibung
<b>loc</b>	lat,lon	Ein mit Koordinaten (Breite, Länge) beschriebener Punkt, Die Koordinatenangaben erfolgen im Koordinatensystem WGS84 in Dezimalschreibweise getrennt durch ein Komma (.)
<b>profile</b>	car, bic, foot	Geschwindigkeitsprofile Auto, Fahrrad oder Fußgänger

### 4.4.2 Antwort

Parameter	Beschreibung
<b>status</b>	Statuscode
<b>name</b>	Name des nächstgelegenen navigierbaren Straßensegments
<b>mapped_coordinate</b>	Ein Array, welches die ermittelten Koordinaten enthält [ lat , lon ]

### 4.4.3 Beispiel

Anfrage:

```
http://sg.geodatenzentrum.de/web_routing/car/nearest?loc=52.4224,13.333086
```

Antwort:

```
{
  "version":0.3,
  "status":0,
  "mapped_coordinate":[
    52.422809,13.333689
  ],
  "name":"Mariannenstraße",
  "transactionId":
  "OSRM Routing Engine JSON Nearest (v0.3)"
}
```

## 4.5 Service viaroute (Standardroute)

Der Dienst ermittelt die kürzeste Route zwischen einem Start- und einem Zielpunkt sowie ggf. eingefügten Zwischenpunkten. Unterstützt werden die Berechnung einer Alternativroute und die Ausgabe der dazugehörigen Routinginstruktionen.

### 4.5.1 Anfrage

```
http://sg.geodatenzentrum.de/web_routing/{profile}/viaroute?loc={lat,lon}&loc={lat,lon}<&loc={lat,lon} ...>
```

Parameter	Wert	Beschreibung
<b>profile</b>	car, bic, foot	Geschwindigkeitsprofile Auto, Fahrrad oder Fußgänger
<b>loc</b>	lat,lon	Ein mit Koordinaten (Länge, Breite) beschriebener Punkt, Die Koordinatenangaben erfolgen im System WGS84 in Dezimalschreibweise getrennt durch ein Komma (.)
<b>output</b>	json (default), gpx	Ausgabeformat der Antwort
<b>instructions</b>	true, false (default)	Routinginstruktionen der Route
<b>alt</b>	true, false (default)	Alternativeroute ein/aus
<b>geometry</b>	true (default), false	Ausgabe der Routengeometrie

### 4.5.2 Antwort

Parameter	Beschreibung
<b>route_geometry</b>	Routengeometrie in komprimierter Form als Polyline [6] auf 6 Nachkommastellen genau. Alternativ kann das Modul [7] npm zum decodieren verwendet werden.
<b>route_instructions</b>	ein Array, welches die Routinganweisungen für jedes Routensegment in der nachfolgenden Form enthält. Jeder Eintrag hat die Array-Elemente: [{drive instruction code}, {street name}, {length}, {location index}, {time}, {formatted length}, {direction}, {azimuth}] <ul style="list-style-type: none"> <li>• driving directions code – Integer-Code der Abbiegerichtung</li> <li>• street name – Strassenname als string</li> <li>• length – Strassenlänge in Meter als Integer</li> <li>• time – Fahrzeit in Sekunden als float</li> <li>• formatted length – Streckenlänge mit Einheitenangabe als string</li> <li>• direction – Richtungswert: N: Nord, S: Süd, E: Ost, W:</li> </ul>

	West, NW: Nord-West, ... als string <ul style="list-style-type: none"> <li>• azimuth – Azimutangabe als float</li> </ul>
<b>route_summary</b>	Zusammenfassung der gelieferten Route <ul style="list-style-type: none"> <li>• total_distance – Gesamtstrecke in Metern als integer</li> <li>• total_time – Gesamtreisezeit in Sekunden als integer</li> <li>• start_point – Name des Startpunktes als string</li> <li>• end_point – Name des Routenziels als string</li> </ul>
<b>route_name</b>	Routenname ("most prominent streets") als string - Array
<b>via_points</b>	Index aller Zwischenpunkte. Jeder Zwischenpunkt ist als Koordinatenpaar der Werte [lat, lon] angegeben
<b>found_alternative</b>	Alternativroute gefunden ( true oder false )

Wenn eine Alternativeroute angefragt war (alt=true), sind in der Serverantwort zusätzlich die folgenden Elemente für jede Route enthalten:

- alternative\_geometries - Array von route\_geometry
- alternative\_instructions - Array von route\_instructions
- alternative\_summaries - Array von route\_summary
- alternative\_names - Array von route\_name

Sonderzeichen in String-Werten sind Url-codiert.

### 4.5.3 Beispiel

Anfrage:

```
http://sg.geodatenzentrum.de/web_routing/foot/viaroute?loc=52.503033,13.420526&loc=52.516582,13.429290&loc=52.554070,13.160621&instructions=true
```

Antwort:

```
{
  "alternative_geometries": [
    "q~occB{ }brX}KaHyY{QaW}OoN{IoHuEcIcFeM}HoKyGuBsA}U}OcvA} }@yxBgwAgT`kAyAjH_CvLyDvS{Mhv@aDbS{E`Vky{QofAur@eTkN{QsLkFcF_b@q\\_IqImKkJwNyMaGqFwD{KeEkGad@aqq@eReTyPgOk_A_z@fE_q@^_GnIqkAkTeGcg@oMgCq@"
  ],
  "alternative_indices": [
    0,
    42
  ],
  "alternative_instructions": [
    [
      "10",
      "Adalbertstra\u00dfe",
      701,
    ]
  ]
}
```

```

        0,
        71,
        "701m",
        "NE",
        23,
        1
    ],
    [
        "7",
        "L 1066",
        260,
        12,
        21,
        "260m",
        "NW",
        295,
        1
    ],
    [
        0,
        69,
        0,
        "0m",
        "N",
        0
    ]
],
"route_name": [
    "Bethaniendamm",
    "Andreasstra\u00dfe"
],
"route_summary": {
    "end_point": "Krautstra\u00dfe",
    "start_point": "Adalbertstra\u00dfe",
    "total_distance": 2034,
    "total_time": 164
},
"status": 0,
"status_message": "Found route between points",
"via_indices": [
    0,
    70
],
"via_points": [
    [
        52.503033,
        13.420526
    ],
    [
        52.516582,
        13.42929
    ]
]
}

```

## 4.6 Service fr (schnellste Route)

Der Dienst ermittelt die schnellste Route zwischen einem Start- und einem Zielpunkt ohne Zwischenpunkte. In der vorliegenden Implementierung nur verfügbar für das Geschwindigkeitsprofil Auto.

### 4.6.1 Anfrage

```
http://sg.geodatenzentrum.de/web_routing/
o2/car/Osm2poService?cmd=fr?source={lat,lon}&target={lat.lon}&format=geojson
```

Parameter	Wert	Beschreibung
<b>source</b>	lat,lon	mit Koordinaten (Breite, Länge) beschriebene Punkte. Die Koordinatenangaben erfolgen im System WGS84 in Dezimalschreibweise getrennt durch ein Komma (,)
<b>target</b>	lat,lon	
<b>format</b>	geojson, text(default)	Ausgabeformat der Antwort

### 4.6.2 Antwort

Als Antwort liefert der Server die Routingdaten im Format „geoJSON“ des Typs „FeatureCollection“ aus.

### 4.6.3 Beispiel

Anfrage:

```
http://sg.geodatenzentrum.de/web_routing/o2/car/Osm2poService?cmd=fr&source=50.10014343261719,8.500038146972656&target=50.19991683959961,8.601035118103027&format=geojson
```

Antwort:

```
{ "type": "FeatureCollection", "features": [
  { "type": "Feature", "geometry": { "type": "LineString", "coordinates": [[ [ 8.501403, 50.1006422 ], [ 8.5017413, 50.100312 ], [ 8.5018018, 50.1002869 ], [ 8.5018834, 50.1002992 ], [ 8.5019066, 50.1003086 ], [ 8.5023464, 50.1004864 ], [ 8.5028522, 50.1006908 ] ] }, "properties": { "id": 902860, "name": "Rombergstraße", "flags": 16973856, "noturns": "00000000", "length": 0.13672815, "time": 0.004557605, "source": 657937, "target": 657928, "oneway": true, "reverse": false, "wrongway": false, "ratio": 1.0 } },
  .
  .
  .
  { "type": "Feature", "geometry": { "type": "LineString", "coordinates": [[ [ 8.6003469, 50.2005857 ], [ 8.6012003, 50.2002385 ], [ 8.6014523, 50.2001599 ] ] }, "properties": { "id": 11383889, "name": "Burgstraße", "flags": 196640, "noturns": "00000000", "length": 0.09192518, "time": 0.0018385036, "source": 8558669, "target": 27283359, "oneway": false, "reverse": true, "wrongway": false, "ratio": 1.0 } }
  ] }
```

## 4.7 Erreichbarkeitsanalyse

Die Funktionen car.php/bic.php/foot.php erlauben die Berechnung von Erreichbarkeiten für einen mit Koordinaten beschriebenen Punkt mit den Parametern Zeit und Entfernung.

### 4.7.1 Anfrage

Erreichbarkeitsanalyse für Entfernung

```
http://{server-address}/{car|bic|foot}.php?loc={lat,lon}&dist=yyy
```

Erreichbarkeitsanalyse für Zeit

```
http://{server-address}/{car|bic|foot}.php?loc={lat,lon}&time=xxx
```

Parameter	Wert	Beschreibung
<b>profile</b>	car, bic, foot	Geschwindigkeitsprofile Auto, Fahrrad oder Fußgänger
<b>loc</b>	lat,lon	Ein mit Koordinaten (Breite, Länge) beschriebener Punkt, Die Koordinatenangaben erfolgen im Koordinatensystem WGS84 in Dezimalschreibweise getrennt durch ein Komma (,)
<b>dist</b>	km	Entfernungsangabe in km (max. 60)
<b>time</b>	Stunden	Zeitangabe in Stunden als Dezimalwert (max. 0,5)

**Anmerkung:** Die Berechnungszeit ist abhängig von der Serverlast und nimmt mit Erhöhung der angefragten Werte zu. (s. Kap.5)

### 4.7.2 Antwort

Parameter	Beschreibung
<b>type</b>	geoJSON-Objekt vom Typ „Polygon“
<b>coordinates</b>	Liste Koordinaten aller Polygonknotenpunkte

### 4.7.3 Beispiel

Anfrage:

```
http://sg.geodatenzentrum.de/web_routing/foot.php?lat=49.8993&lon=8.84&dist=10
```

Antwort:

```
{ "type"
: "Polygon", "coordinates": [[ [8.83685054283331, 49.8972413834284], [8.836894339
87258, 49.8974313715083], [8.83690990591129, 49.8976932893354], [8.836899858284
```



```
27,49.8980082885078],[8.83689984331333,49.8980098817868],[8.83690550302003,
49.8984420790928],[8.83683764639326,49.8985811802555],[8.8368284011929,49.8
99541653849],[8.836825,49.899895],[8.837052,49.900406],[8.83746738574273,49
.9006995487582],[8.837635,49.900818],[8.83785792601441,49.9007062533719],[8
.83786620363283,49.9006098011225],[8.83786626059138,49.9006090417001],[8.83
786917157876,49.9005637002594],[8.83786869593574,49.9005552211878],[8.83785
689454529,49.900488934009],[8.83815162241995,49.8999176504543],[8.838195412
11187,49.8998751890288],[8.83866232715259,49.8997096435321],[8.838749297401
52,49.899700889134],[8.838766445872,49.8996943928395],[8.83882600473395,49.
8996515197287],[8.83883643476225,49.8996440117317],[8.83933601560159,49.899
4792857876],[8.83941883114127,49.8995014761446],[8.83943716885873,49.899501
4761446],[8.83957264020105,49.8994651767078],[8.83961429760769,49.899454014
6393],[8.83963017853686,49.8994448457806],[8.83975771952927,49.899317304788
2],[8.83975984578058,49.8993151785369],[8.83976901463932,49.8992992976077],
[8.8397941093296,49.8992056429486],[8.84002068309646,49.8991474210117],[8.8
4018041712246,49.899190221615],[8.84041924072299,49.8993583416675],[8.84043
247186064,49.8994077209454],[8.84043723551004,49.8994133170599],[8.840779,4
9.899242],[8.84107226565644,49.8989705796425],[8.84110521263933,49.89876038
15704],[8.84145308325206,49.8985242304184],[8.8415296063547,49.898537122584
1],[8.84154386010073,49.8985341141721],[8.841759,49.898335],[8.841666188644
64,49.8981180249075],[8.84161623336789,49.8981157627817],[8.84154085860609,
49.8978250273901],[8.841312,49.89729],[8.84042744318424,49.8971493709355],[
8.83794300258831,49.8967543883288],[8.837538,49.89669],[8.83709470763602,49
.8968384956269],[8.836932,49.896893],[8.836852,49.89709],[8.83685054283331,
49.8972413834284],[[8.83907097046627,49.8985232168843],[8.83895796782284,4
9.8984894976541],[8.83889377865979,49.8984303972594],[8.83886104822561,49.8
98351816257],[8.83892259889635,49.8983650947849],[8.83907097046627,49.89852
32168843]]]}
```

## 5 Allgemeines

### 5.1 Anwendungshinweise

Die Berechnung der Routen und insbesondere der Erreichbarkeitsanalysen sind sehr rechenintensiv. Die Berechnungszeiten nehmen mit der Größe der angefragten Routen zu und sind abhängig von der aktuellen Serverlast. Im Normalfall liegt die Bearbeitungszeit einer Route unter 10 Sekunden. Die Berechnung einer Erreichbarkeitsanalyse kann bis 45 Sekunden andauern.

Sonderzeichen in String-Werten werden Url-codiert ausgeliefert.

### 5.2 Fehlermeldungen

Fehler werden durch einen HTTP-Status-Code gekennzeichnet. Dabei werden in clientseitige Fehler (4xx) und serverseitige Fehler (5xx) unterschieden.

Code	Name	Details	Beschreibung
<b>500</b>	Internal Server Error		Standardfehler für serverseitige Fehler
<b>503</b>	Service Unavailable		Server ist temporär nicht verfügbar (z.B. wegen Überlastung)
<b>504</b>	Gateway Timeout		Die internen Suchdienste im Backend antworten nicht innerhalb des geforderten Timeout-Intervall.
<b>400</b>	Bad Request	Invalide Format	Fehlender Parameter in der Anfrage

## **6 Datenschutz**

Der Routingdienst stellt Routinginformationen ohne jeden Personenbezug bereit und kann nur entsprechende Anfragen ohne jeglichen Personenbezug verarbeiten.

Der Routingdienst und ebenso die vorgelagerte IT-Infrastruktur des BKG speichern grundsätzlich keine an ihn gerichteten Anfragen mit ihren semantischen Parametern, auch nicht temporär. Die Suchparameter und die Ergebnisse der Suche werden nur flüchtig verarbeitet, um die Antwort an den Client zu senden.

Es findet ausschließlich eine synchrone Verarbeitung der Routeninformation ohne jede Zwischenspeicherung statt. Eine asynchrone Massenverarbeitung mit dem Erfordernis einer Zwischenspeicherung auf Serverseite ist nicht realisiert.

## 7 Quellenverzeichnis

1	<a href="https://github.com/Project-OSRM/osrm-backend">https://github.com/Project-OSRM/osrm-backend</a> (Letzter Aufruf am 08.11.2015)
2	<a href="http://osm2po.de">http://osm2po.de</a> (Letzter Aufruf am 08.11.2015)
3	Bundesamt für Kartographie und Geodäsie, Richard-Strauss-Allee 11, 60598 Frankfurt/Main
4	<a href="https://github.com/Project-OSRM/osrm-backend">https://github.com/Project-OSRM/osrm-backend</a> (Letzter Aufruf am 08.11.2015)
5	z.Zt. <a href="http://download.geofabrik.de/">http://download.geofabrik.de/</a> (Letzter Aufruf am 08.11.2015)
6	<a href="http://opendatacommons.org/licenses/odbl/">http://opendatacommons.org/licenses/odbl/</a> (Letzter Aufruf am 08.11.2015)
7	<a href="https://developers.google.com/maps/documentation/utilities/polylinealgorithm?csw=1">https://developers.google.com/maps/documentation/utilities/polylinealgorithm?csw=1</a> (Letzter Aufruf am 08.11.2015)
8	<a href="https://www.npmjs.com/package/polyline">https://www.npmjs.com/package/polyline</a> (Letzter Aufruf am 08.11.2015)
9	<i>GeoJSON</i> . <a href="http://geojson.org/">http://geojson.org/</a> (letzter Aufruf am 09.11.2015).

## 8 Kontakt

### 8.1 Freischaltung

Für die Freischaltung des Dienstes richten Sie sich bitte an:

Bundesamt für Kartographie und Geodäsie  
Dienstleistungszentrum  
Karl-Rothe-Straße 10-14  
04105 Leipzig  
E-Mail: [dlz@bkg.bund.de](mailto:dlz@bkg.bund.de)  
Tel.: 0341/5634-333  
Fax: 0341/5634-415

### 8.2 Technische Anfragen

Technische Anfragen richten Sie bitte an:

Dienstleistungszentrum  
E-Mail: [dlz@bkg.bund.de](mailto:dlz@bkg.bund.de)  
Tel.: 0341/5634-333  
Fax: 0341/5634-415