

EFA-Schnittstelle mit JSON

1.1 Aufbau der Schnittstellen

http://`host`/`virtualDirectory`/`endPoint`

	Wert in der münster:app
host	app.vrr.de
virtualDirectory	msapp

Das `virtualDirectory` kann auf Seiten von MDV bereits weitere Makros und Einstellungen definieren die die Ergebnisse der Schnittstellen beeinflussen.

Alle Aufrufe der App finden mit `POST` statt, `GET` ist aber ebenfalls möglich.

1.2 Schnittstellen Übersicht

	Endpoint
StopFinder	XML_STOPFINDER_REQUEST
TripFinder	XML_TRIP_REQUEST2
Departure Monitor	XML_DM_REQUEST
Additional Info	XML_ADDINFO_REQUEST

1.3 Standard Request Parameter

Die folgenden Parameter werden immer mitgeschickt, unabhängig von der genutzten Schnittstelle.

Key	Value
outputFormat	JSON
stateless	1
locationServerActive	1
coordOutputFormat	WGS84[DD.DDDDD]
coordOutputFormatTail	5

outputFormat

Legt das Ausgabe-Format der Schnittstelle fest. In diesem Fall auf JSON. Diese benötigt weniger Datentransfer als eine XML Schnittstelle.

stateless

Aktiviert die Ausgabe des `stateless`-Wertes, ein eindeutiger Identifier für Haltestellen, Straßen, POI etc.

locationServerActive

Aktiviert den EFA Location Server und erlaubt eine Suche nach `stateless` Werten.

coordOutputFormat

Legt das Ausgabeformat der Geo-Coordinaten fest. WGS84 legt dabei die geläufigen Breiten- und Längengrade fest. Standardmäßig wird ein eigenes, von MDV entwickeltes Format, ausgegeben.

coordOutputFormatTail

Da entgegen der `coordOutputFormat` Annotation nicht ausreichend Nachkommastellen in den Koordinaten ausgegeben werden, wird hiermit der Wert auf 5 Nachkommastellen gesetzt.

2. Stop Finder

Der Stop Finder wird genutzt um Haltestellen (beinhaltet ggf auch POI, Straßen etc)

zu finden.

2.1 Parameter

Zusätzlich zu den Standardparametern werden folgende Werte übergeben

Key	Value
type_sf	any
name_sf	<input type="text" value="Eingabe des Nutzers"/>
anyMaxSizeHitList	64
anyObjFilter_sf	126
w_regPrefAl	2

Der Postfix bezeichnet den Einsatz für den StopFinder

type_sf

Beschreibt den Typ des Punktes.

Mögliche Werte sind: , , und

name_sf

Hier wird der vom Benutzer eingegebene Wert eingesetzt

anyMaxSizeHitList

Definiert das Limit der zurückgegebenen Werte da die Antwort sonst sehr groß wird.

anyObjFilter_sf

Filter für die Ergebnisse. Der Wert ist eine Kombination aus folgenden Werten

Type	Value	
Complete	0	
Place	1	
Stop	2	✓
Street	4	✓
Address	8	✓
Crossing	16	✓
POI	32	✓
Postcode	64	✓

Siehe EFA XML-Schnittstellen Dokumentation Seite 85 (Punkt 16.2)

w_regPrefAI

Dieser Wert passt die Gewichtung der Ergebnisse an.

Regionen (`reg`) werden immer (`AI`) priorisiert.

Siehe EFA XML-Schnittstellen Dokumentation Seite 29 (rMethod)

2.2 Auswertung der Suchergebnisse

Die Haltestellen befinden sich als `Array` im Keypath `stopFinder.points`.

ACHTUNG: Kommt nur ein Ergebnis von der Schnittstelle zurück, ist die Haltestelle als `Dictionary` im Keypath `stopFinder.points.point` zu finden.

Stop Object

Der angegebene `keypath` ist relativ zum Einzelnen Suchergebnis

keypath	Datentyp	
<code>anyType</code>	String	Siehe Sonstiges (<i>Punkt 8</i>)
<code>name</code>	String	
<code>object</code>	String	Bezeichner der Haltestelle - z.B. <i>Hauptbahnhof</i> oder <i>Alstadt / Bült</i>
<code>quality</code>	Integer	Von der Schnittstelle berechnete Qualität
<code>stateless</code>	String	Identifizier für die Haltestelle
<code>ref.place</code>	String	Ortsbezeichner - z.B. <i>Münster (Westf)</i>

Alle Ergebnisse werden anhand der `quality` absteigend (`DESC`) sortiert.

3. Verbindungsauskunft

3.1 Parameter

Zusätzlich zu den Standardparametern werden folgende Werte übergeben

Key	Value	
name_origin	Von <input type="text" value="stateless"/>	Siehe StopFinder (<i>Punkt 2.2</i>)
name_destination	Nach <input type="text" value="stateless"/>	Siehe StopFinder (<i>Punkt 2.2</i>)
type_origin	any / coord	Siehe Sonstiges (<i>Punkt 8</i>) - AnyType
type_destination	any / coord	Siehe Sonstiges (<i>Punkt 8</i>) - AnyType
useRealtime	1	
changeSpeed		Siehe Einstellungen (<i>Punkt 7</i>)
routeType		Siehe Einstellungen (<i>Punkt 7</i>)
useProxFootSearch	1 / 0	Siehe Einstellungen (<i>Punkt 7</i>)
trITMOTvalue100		Siehe Einstellungen (<i>Punkt 7</i>)
excludedMeans	checkbox	
exclMOT_<ID>	1	Siehe Einstellungen (<i>Punkt 7</i>)
impairedOptionsActive	1	
itOptionsActive	1	
ptOptionsActive	1	
itdTripDateTimeDepArr	dep / arr	
itdDate		optional
itdTime		optional
lineRestriction		optional - Siehe Einstellungen (<i>Punkt 7</i>)

useRealtime

Aktiviert die Echtzeitüberwachung inklusive Informationen zu möglichen

Verspätungen.

excludedMeans

Der Wert `checkbox` aktiviert die Optionen zum ausschließen von bestimmten Transportwegen. Wird in Kombination mit den `exclMOT_<ID>` Werten genutzt. *Siehe Einstellungen (Punkt 7)*

imparedOptionsActive

Aktiviert den Parameter für eingeschränkte Mobilität

itOptionsActive

Aktiviert die Verbindungssuche mit Individualtransport

ptOptionsActive

Aktiviert die Verbindungssuche mit öffentlichen Verkehrsmitteln

itdTripDateTimeDepArr

Legt fest ob die angegebene Zeit der Ankunft oder der Abfahrt dienen soll

itdDate

Abfahrt- oder Ankunftsdatum im Format `yyyyMMdd`

itdTime

Abfahrts- oder Ankunftszeitpunkt im Format `HHmm`

Weitere Einstellungen finden Sie in der EFA-XML Schnittstellen Dokumentation ab Seite 40 (Punkt 7.5)

3.2 Auswertung der Suchergebnisse

Die Verbindungen befinden sich als `Array` im Keypath `trips`.

ACHTUNG: Kommt nur ein Ergebnis von der Schnittstelle zurück, ist die Verbindung als `Dictionary` im Keypath `trips.trip` zu finden.

Trip Objekt

keypath	Datentyp	
<code>duration</code>	String	Dauer der Fahrt im Format <code>HH:mm</code>
<code>interchange</code>	Integer	Anzahl Umstiege
<code>legs</code>	Array	PartialRoute Objekte
<code>itdFare.fares.fare</code>	Dictionary mit Preisinformationen	

PartialRoute Objekt

Jede Partial Route beinhaltet eine Teil-Fahrt ohne Umstieg

keyPath	Datentyp	
<code>timeMinute</code>	Integer	
<code>controlled.delayMinutes</code>	Integer	
<code>controlled.delayMinutesArr</code>	Integer	
<code>path</code>	String	String mit Koordinaten
<code>mode</code>	Dictionary	Transport Objekt
<code>hints</code> bzw <code>hints.hint</code>	Array / Dictionary	Array bzw. einzelnes Hint-Objekt
<code>points</code>	Array	Array von Haltepunkten
<code>stopSeq</code>	Array	Array von Stop Sequence Objekten

path

Breiten- und Längengrade um die Route zu markieren. Wird in der `münster:app` genutzt um die Fahrt auf der Karte einzeichnen zu können

Bsp.: `7.63502,51.95622 7.63487,51.95573 7.63465,51.95520`

Stop Sequence Objekt

Einzelne Zwischenhaltestellen

keyPath	Datentyp	
<code>name</code>	String	Genauer Name der Haltestelle - z.B. <code>Dortmund Hbf</code>
<code>nameWO</code>	String	Name der Haltestelle - z.B. <code>Hauptbahnhof</code>
<code>place</code>	String	Ort der Haltestelle
<code>platformName</code>	String	Bahnsteig (Häufig nur Nummern, teilweise auch mit Text: <code>Gleis 4</code> - nicht immer gesetzt)
<code>ref.platform</code>	String	Bahnsteig - Gleiche Bedingungen wie <code>platformName</code> - tendenziell aber häufiger nur eine Nummer
<code>ref.arrDateTime</code>	String	Ankunfts-Zeit
<code>ref.arrDelay</code>	Integer	Verspätung der Ankunft in Minuten
<code>ref.depDateTime</code>	String	Abfahrts-Zeit
<code>ref.depDelay</code>	Integer	Verspätung der Abfahrt in Minuten

Point Objekt

keyPath	Datentyp	
<code>name</code>	String	<i>Siehe Stop Sequence</i>
<code>nameWO</code>	String	<i>Siehe Stop Sequence</i>
<code>place</code>	String	<i>Siehe Stop Sequence</i>
<code>usage</code>	String	<code>departure</code> oder <code>arrival</code>
<code>platformName</code>	String	<i>Siehe Stop Sequence</i>
<code>stamp.date</code>	String	Planmäßiges Datum der An-/Abreise im Format <code>yyyyMMdd</code>
<code>stamp.time</code>	String	Planmäßiger Zeitpunkt der An-/Abreise im Format <code>HHmm</code>
<code>stamp.rtDate</code>	String	Datum der An-/Abreise im Format <code>yyyyMMdd</code> inklusive möglicher Verspätung
<code>stamp.rtTime</code>	String	Zeitpunkt der An-/Abreise im Format <code>HHmm</code> inklusiver möglicher Verspätung

Transport Objekt

keyPath	Datentyp	
<code>name</code>	String	<code><product> - <number></code>
<code>number</code>	String	z.B: <code>RB50</code>
<code>product</code>	String	z.B. <code>Regionalbahn</code>
<code>type</code>	Integer	Transport Typ
<code>destination</code>	String	z.B. <code>Dortmund Hbf</code>

Weitere Informationen über Betreiber, Netzwerk etc sind ebenfalls verfügbar, werden aber von der `münster:app` aktuell nicht genutzt

Transport Typ

Wert	Bedeutung
1	U-Bahn
2	S-Bahn
3	Sprinterbus
4	Straßenbahn
5	StadtBus
6	R-Bahn
8	Taxibus
97	<i>nicht umsteigen</i>
98	<i>gesicherter Anschluss</i>
99, 100	Fußweg

Weitere Informationen zu Farben, Icons etc finden Sie am Ende unter Punkt 8 bei der Erläuterung der *TransportTypePill*

Hint Objekt

keyPath	Datentyp	
<code>infoText</code>	String	z.B. <code>Fahrradmitnahme reservierungspflichtig</code> aber auch Informationen zum TaxiBus

4. Abfahrtsmonitor

4.1 Parameter

Zusätzlich zu den Standardparametern werden folgende Werte übergeben

Key	Value	
<code>name_dm</code>	<code>stateless</code>	Siehe StopFinder (<i>Punkt 2.2</i>)
<code>type_dm</code>	<code>any / coord</code>	Siehe Sonstiges (<i>Punkt 8</i>) - AnyType

useRealtime	1	Siehe Parameter Verbindungsauskunft
mergeDep	1	
useAllStops	1	
mode	direct	
itOptionsActive	1	Siehe Parameter Verbindungsauskunft
ptOptionsActive	1	Siehe Parameter Verbindungsauskunft
imparedOptionsActive	1	Siehe Parameter Verbindungsauskunft
depType	stopEvents	
maxTimeLoop	2	
includeCompleteStopSeq	1	
changeSpeed		Siehe Einstellungen (<i>Punkt 7</i>)
useProxFootSearch	1 / 0	Siehe Einstellungen (<i>Punkt 7</i>)
excludedMeans	checkbox	Siehe Parameter Verbindungsauskunft
itdTripDateTimeDepArr	dep / arr	Siehe Parameter Verbindungsauskunft
itdDate		optional - Siehe Parameter Verbindungsauskunft
itdTime		optional - Siehe Parameter Verbindungsauskunft

mode

Siehe EFA-XML Schnittstellen Dokumentation Seite 58 (Punkt 9.5)

mergeDep, *useAllStops*, *mode*, *maxTimeLoop* basieren auf Auswertungen der

4.2 Auswertung der Suchergebnisse

Die Abfahrten befinden sich als `Array` im Keypath `departureList`.

Geokoordinaten für den Startpunkt finden sich als `String` im keyPath

`dm.points.point.ref.coords`. Unabhängig ob nach Aktuellem Standort oder anhand einer Haltestelle/POI gesucht wurde.

keyPath	Datentyp	
<code>stopName</code>	String	Name der Haltestelle - z.B. <i>Münster (Westf) Hauptbahnhof</i>
<code>x</code>	String	Längengrad (Latitude)
<code>y</code>	String	Breitengrad (Longitude)
<code>dateTime</code>	Dictionary	Planmäßige Ankunftszeit
<code>dateTimeReal</code>	Dictionary	Reale Ankunftszeit
<code>servingLine.delay</code>	Integer	Verspätung in Minuten - <code>-9999</code> bedeutet, dass diese Verbindung ausfällt
<code>servingLine.direction</code>	String	Angabe in welche Richtung die Linie fährt - z.B. <i>Münster Clemenshospital</i>
<code>servingLine.name</code>	String	Name der Linie - z.B. <i>StadtBus</i>
<code>servingLine.number</code>	String	Nummer der Linie - z.B. 2
<code>servingLine.itdNoTrain</code>	String	Beinhaltet einen Informationstext. z.B. Informationen, dass es sich um einen TaxiBus handelt
<code>lineInfos</code> / <code>lineInfos.lineInfo</code>	Array / Dictionary	
<code>stopInfos</code> / <code>stopInfos.stopInfo</code>	Array / Dictionary	
<code>onwardStopSeq</code> / <code>onwardStopSeq.lineInfo</code>	Array / Dictionary	StopSequence Objekt(e) - Siehe Stop Sequence Objekt im TripFinder (<i>Punkt 3.2</i>)

Die Ergebnisse werden nach `stopName` gruppiert.

`dateTime` Dictionaries beinhalten die Keys `year`, `month`, `day`, `hour` und `minute`

5. Störungen

5.1 Parameter

Zusätzlich zu den Standardparametern werden folgende Werte übergeben

Key	Value	
filterPublicationStatus	current	Aktuell aktive Störungen
filterOMC_PlaceID	5515000:29	Gemeindekennziffer für Münster und ID des Ortes

5.2 Auswertung der Suchergebnisse

Die Störungen befinden sich als Array im keyPath

```
additionalInformation.travelInformations.travelInformation
```

keyPath	Datentyp	
<code>infoLink.subject</code>	String	<i>z.B. Haltestelle bis Anfang April außer Betrieb</i>
<code>infoLink.subtitle</code>	String	Wird als Titel genutzt - z.B. <i>Straßenbauarbeiten an der Haltestelle Corrensstraße in Münster</i>
<code>infoLink.content</code>	String	Erweiterter Beschreibungstext mit HTML
<code>infoLink.concernedLines</code>	Array	Liste mit betroffenen Linien

concernedLines

Hier ist zu beachten, dass die selbe Linie mehrmals auftauchen kann, auf Grund der unterschiedlichen Richtungen in die sie fahren kann. In der münster:app wird von den Objekten nur das `name` Feld ausgewertet.

6. Folgeanfragen

6.1 Session basierte Requests

Über Sessions werden bei der münster:app bei einer Verbindungssuche die früheren und späteren Verbindungen gesucht.

Für die Kontinuität wird eine `sessionID` und eine `requestID` mitgesendet. Die Startwerte sind dabei bei beiden Werten jeweils `0` (null).

Von der EFA-Schnittstelle bekommt man darauf im Schlüssel `parameters` eine neue `sessionID` und eine passende `requestID` zurück.

Mit diesen neuen Werten können schließlich auf der EFA-Schnittstelle simple Befehle mitgeschickt werden und reduziert die Request Parameter auf `sessionID`, `requestID` und `command`.

6.2 Auszug der commands für die Verbindungsauskunft

<code>command</code>	
<code>tripPrev</code>	frühere Fahrt
<code>tripNext</code>	spätere Fahrt

Weitere Befehle finden sich in der von MDV bereitgestellten XML-Schnittstellen Dokumentation.

Verbindungsauskunft: Seite 47 (Punkt 7.10)

Abfahrtsmonitor: Seite 58 (Punkt 9.6)

7. Einstellungen

Die folgenden Einstellungen lassen sich durch den Nutzer verändern

Key	Optionen	Default
routeType	LEASTTIME / LEASTINTERCHANGE / LEASTWALKING	LEASTTIME
changeSpeed	fast / normal / slow	normal
trITMOTvalue100	5 / 10 / 15 / 20 / 30 / 45	15
excludedMOT	Array mit Werten von 0 bis 10	-
lineRestriction	400 / 401 / 402	400
useProxFootSearch	true / false	true

routeType

Wert	Bedeutung
LEASTTIME	Schnellste Verbindung
LEASTINTERCHANGE	Wenigste Umstiege
LEASTWALKING	Kürzeste Fußwege

changeSpeed

Modifiziert die für Umstiegsfußwege benötigte Zeit.

`fast`, `normal` und `slow` sind Verknüpfungen auf Werte die von der EFA Schnittstelle selbst verwaltet werden. Alternativ können auch eigene Integer Werte übergeben werden.

Weitere Informationen inklusive Formeln zum `changeSpeed` Wert finden Sie in der XML-Schnittstellen Dokumentation auf Seite 44 (Punkt 7.8)

trITMOTvalue100

Eine Variation des `trITMOT` Wertes. Die 100 steht für den Fußweg und der Wert definiert die maximale Fußwegzeit.

Weitere Informationen zum `trITMOT` Wert finden Sie in der XML-Schnittstellen Dokumentation auf Seite 46 (Punkt 7.9)

excludedMOT

Über die `excludedMOT` werden bestimmte Transportwege deaktiviert. Diese werden mit den angegebenen werten als Array gespeichert und in den Requests mit `exclMOT_<ID>` ausgeschlossen.

Wert	Bedeutung
0	Zug (Wird in Verbindung mit <code>lineRestriction</code> genutzt)
1	S-Bahn
2	U-Bahn
3	Stadtbahn
4	Straßenbahn
5	Bus
6	Regionalbus
7	Schnellbus
8	Seil-/Zahnradbahn
9	Schiff
10	Ruftaxi

Siehe EFA XML-Schnittstellen Dokumentation Seite 84 (Punkt 16.1)

lineRestriction

Wert	Bedeutung
400	Zug (alle)
401	Zug (alle ohne ICE)
402	Zug (nur regional)

useProxFootSearch

Legt fest ob auf Basis des `changeSpeed` Wertes, ein Umstieg mit Fußweg beachtet

werden soll, wenn diese Verbindung dadurch ggf schneller ist.

8. Sonstiges

AnyType

Gibt den Typ der Haltestelle an

Wert	Bisher in der münster:app genutzt
address	
crossing	
coord	✓
poi	✓
pp	
stop	✓
street	✓
singlehouse	✓

Stateless aus Geodaten

Für die Suchen nach z.B. dem aktuellen Standort wird eine bestimmte Formatierung voraus gesetzt, die als `stateless` Wert übergeben werden kann. Diese setzt sich wie folgt zusammen:

`x:y:type`

Als `type` setzen wir das Format `WGS84[DD.DDDDD]` ein, welches wir auch von den Schnittstellen bekommen (Siehe *Standard Request Parameter*).

Der Wert `x` ist der Längengrad (Longitude) und `y` der Breitengrad (Latitude).

`x` und `y` werden jeweils entgegen der Typ-Deklaration mit 5 Nachkommastellen und mit Punkt formatiert.

Beispiel: `7.63525:51.95693:WGS84[DD.DDDDD]`

TransportTypePill

Zur Kennzeichnung der in einer Verbindung enthaltenen Transportmittel werden in der App sogenannte TransportTypePills genutzt. Bei diesem handelt es sich um View-Elemente in Form einer Pille, welche anhand von der Hintergrundfarbe und eines Icons den Typ des Transportmittels darstellt.

Als Übergabeparameter erhält die TransportTypePill den in der entsprechenden Server-Response enthaltenen type

Die Icons stammen aus dem frei verfügbaren Material Design icon Set:

<https://design.google.com/icons>

type	Bezeichnung	Farbe	Icon
1	U-Bahn	#003399	ic_directions_subway
2, 4	S-Bahn	#007055	ic_directions_railway
3, 5	Sprinter-/Stadtbus	#00a0dd	ic_directions_bus
6	R-Bahn	#cc3333	ic_directions_railway
8	Taxi	#f9ad35	ic_local_taxi
97	"nicht umsteigen"	#7a7a7a	ic_airline_seat_recline_normal
98	"Anschlussgarantie"	#7a7a7a	ic_info_outline
99, 100	Fussweg	#7a7a7a	ic_directions_walk
	Mehr	#7a7a7a	ic_more_horiz
	default/fallback	#7a7a7a	ic_info_outline