

# OGD - Linz

## EFA XML Schnittstelle

# Übersicht

## EFA XML Schnittstelle

1. Allgemeines
2. Grundlagen
3. Stop-Finder-Request - Haltestellensuche
4. Trip-Request – Fahrplananfrage
5. DM-Request – Abfahrtsmonitor

# Allgemeines - Hinweise

## EFA XML Schnittstelle

Die im Rahmen der Open Government Data Initiative zur Verfügung gestellte XML Schnittstelle zur Fahrplanauskunft EFA der Linz AG wurde im Dezember 2015 aktualisiert. Dabei gilt es folgendes zu beachten:

- XML Dokumente stehen wie bisher nur im Format ISO-8859-1 zur Verfügung. In der früheren Version wurde hier fälschlicherweise UTF-8 im Encoding der XML Antworten angeführt.
- Die XML Schnittstelle stellt mit der Aktualisierung keine Störungs- oder Echtzeitmeldungen mehr zur Verfügung
- Anfragen an der Server sollen nur mehr stateless erfolgen. Der EFA Server wurde entsprechend konfiguriert. In den Anfragen ist der Parameter stateless=1 zu berücksichtigen.

# 1. Grundlagen

1. Eingabe und Ausgabe
2. HTTP-Requests
3. Umwandlung der Ausgabe
4. Spracheinstellung
5. Ausführungsanweisung
6. Generische Parameter für das User-Interface („Layout-Parameter“)

# 1.1 Eingabe und Ausgabe

## Eingabe

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?](http://www.linzag.at/static/XML_TRIP_REQUEST2?)

## Ausgabe

XML-Struktur:

```
- <itdRequest version="9.16.28.8" language="de" lengthUnit="METER" sessionID="0" client="Firefox"
  <clientHeaderLines/>
  + <itdInfoLinkList></itdInfoLinkList>
  - <itdTripRequest requestID="0">
    <itdAddress name="" street="" place="" addressExt1="" addressExt2=""/>
    + <itdOdv type="any" usage="origin" anyObjFilter="0"></itdOdv>
    + <itdOdv type="stop" usage="destination" anyObjFilter="0"></itdOdv>
    + <itdOdv type="unknown" usage="via" anyObjFilter="0"></itdOdv>
    + <itdTripDateTime depar="dep" ttpFrom="20120709" ttpTo="20121231"></itdTripDateTime>
    + <itdTripOptions userDefined="0"></itdTripOptions>
    + <itdItinerary></itdItinerary>
  </itdTripRequest>
</itdRequest>
```

## 1.2. HTTP-Requests

Verschiedene Anfragearten (HTTP-Requests) stellen verschiedene Funktionalitäten zur Verfügung.

- Stop-Finder-Request - Haltestellensuche
- Trip-Request – Fahrtanfrage
- DM-Request – Abfahrtsmonitor

## 1.2 HTTP-Requests (2)

Alle Anfragearten leiten sich von einem gemeinsamen Anfrage-Objekt „*itdRequest*“ ab. Dieses stellt Grundfunktionalität wie

- die Umwandlung der Ausgabe (1.3)
- die Spracheinstellung (1.4)
- die Ausführungsanweisung (1.5)
- generische Parameter für das User-Interface (1.6)

zur Verfügung.

## 1.4 Spracheinstellung

### **language**

Kontrolle sprachspezifischer Ausgabeinformation.

Wert: Die durch den Standard ISO 638:1988 (E/F) definierten Sprachkürzel.

### Beispiele:

[http://www.linzag.at/static/XSLT\\_TRIP\\_REQUEST2?language=de](http://www.linzag.at/static/XSLT_TRIP_REQUEST2?language=de)

[http://www.linzag.at/static/XSLT\\_TRIP\\_REQUEST2?language=en](http://www.linzag.at/static/XSLT_TRIP_REQUEST2?language=en)



## 1.5 Ausführungsanweisung

### **execInst**

Beeinflusst die Ausführung der Anweisung.

Werte:

- **normal** – Standardmäßige Ausführung.
- **verifyOnly** – Punkte werden verifiziert und Parameter in die XML-Ausgabe übernommen, es findet jedoch keine Berechnung von Fahrten, Abfahrten, etc. statt.
- **readOnly** – Die Eingaben durch Parameter werden in die XML-Ausgabe übernommen, es findet jedoch weder eine Punktverifikation noch eine Berechnung von Fahrten, Abfahrten, etc. statt.

# 1.5 Ausführungsanweisung (2)

## Beispiel:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?type\\_origin=stopID&name\\_origin=60501200&type\\_destination=stopID&name\\_destinati](http://www.linzag.at/static/XML_TRIP_REQUEST2?type_origin=stopID&name_origin=60501200&type_destination=stopID&name_destinati)

```
-<itdRequest version="9.16.27.3" language="de" lengthUnit="METER" sessionID="0" client="Firefox" clientIP="127.0.0.1" serverID="localhost" virtDir="linz_neu" now="2011-09-13T07:49:21" nowWD="3">
- <itdTripRequest requestID="0" execInst="verifyOnly">
  <itdAddress name="" street="" place="" addressExt1="" addressExt2=""/>
- <itdOdv type="stop" usage="origin">
  - <itdOdvPlace state="identified" method="itp">
    <odvPlaceElem omc="44001001" placeID="4" value="44001001:4" span="0" type="remote" mainPlace="1">Linz/Donau</odvPlaceElem>
    <odvPlaceInput/>
  </itdOdvPlace>
  - <itdOdvName state="identified" method="itp">
    <odvNameElem x="5448851" y="810583" mapName="NAV5" stopID="60501200" value="60501200:1" isTransferStop="0" matchQuality="100000">WIFI/LINZ. AG</odvNameElem>
    <odvNameInput>60501200</odvNameInput>
  </itdOdvName>
</itdOdv>
- <itdOdv type="stop" usage="destination">
  - <itdOdvPlace state="identified" method="itp">
    <odvPlaceElem omc="44001001" placeID="4" value="44001001:4" span="0" type="remote" mainPlace="1">Linz/Donau</odvPlaceElem>
    <odvPlaceInput/>
  </itdOdvPlace>
  - <itdOdvName state="identified" method="itp">
    <odvNameElem x="5449235" y="811418" mapName="NAV5" stopID="60501070" value="60501070:1" isTransferStop="0" matchQuality="100000">Neue Welt</odvNameElem>
    <odvNameInput>60501070</odvNameInput>
  </itdOdvName>
</itdOdv>
```

## 1.6 Generische Parameter für das User-Interface

### itdLPxx\_paramName

Die generischen Parameter werden durchgereicht, ohne das Systemverhalten zu beeinflussen. Sie werden auch „Layout-Parameter“ genannt, da sie dazu dienen sich einen bestimmten Zustand der Programmoberfläche zu merken.

Sie beginnen mit „itdLPxx\_“, das nur zur Kennzeichnung dient und weggeschnitten wird. Der Name *paramName* kann ebenso wie der Wert frei gewählt werden.

Beispiel:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?itdLPxx\\_paramName=paramValue](http://www.linzag.at/static/XML_TRIP_REQUEST2?itdLPxx_paramName=paramValue)

## 2. Exkurs: Analysetools

Folgende Tools sind für die Analyse der Anfrageparameter hilfreich:

- Fiddler
- Firebug

### **Fiddler** (Freeware)

Web Debugging Proxi, das den HTTP(S)-Verkehr loggt.

<http://www.fiddler2.com>

### **Firebug** (Addon für Firefox)

- HTML/CSS analysieren und in Echtzeit ändern
- JavaScript Debugger
- Analyse von Performance, Headern, Requests,...

### 3. Grundfunktionalität

Einige Funktionalitäten werden von mehreren HTTP-Requests (Trip-Request, DM-Request,...) benötigt, z.B.:

1. Datum und Uhrzeit
2. Verifikation von Punkten

## 3.1 Datum und Uhrzeit

XML-Schema: *itdDateTime* beinhaltet *itdDate* (Datum) und *itdTime* (Uhrzeit).

### Parameter zur Wahl des Datums (*itdDate*)

Default: Anfragedatum

Parametername	Bedeutung	Format
itdDate	Jahr, Monat, Tag	JJJJMMTT   JJMMTT

## 3.1 Datum und Uhrzeit (2)

Anfragebeispiel:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?itdDate=20110225](http://www.linzag.at/static/XML_TRIP_REQUEST2?itdDate=20110225)

```
- <itdTripDateTime deparr="dep" ttpFrom="20110110" ttpTo="20111031">
  - <itdDateTime ttpFrom="20110110" ttpTo="20111031">
    <itdDate weekday="6" year="2011" month="2" day="25"/>
    <itdTime hour="7" minute="52"/>
  </itdDateTime>
</itdTripDateTime>
```

Wird die Zeit nicht angegeben (siehe später) so wird die aktuelle Systemzeit übernommen.

## 3.1 Datum und Uhrzeit (3)

### Fehlercodes für das Datum (*itdMessage*)

Code	Bedeutung
-1	Ungültiger Datumswert
-10	Jahr außerhalb des Wertebereichs
-20	Monat außerhalb des Wertebereichs
-30	Tag außerhalb des Wertebereichs
-4001	Datum außerhalb der Fahrplanperiode

Beispiel: Fehler -1 und -30

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?itdDate=20110299](http://www.linzag.at/static/XML_TRIP_REQUEST2?itdDate=20110299)



## Datum und Uhrzeit (3a)

### Beispiel – fehlerhafte Eingabe

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?itdDate=20110299](http://www.linzag.at/static/XML_TRIP_REQUEST2?itdDate=20110299)

```
- <itdTripDateTime deparr="dep" ttpFrom="20110110" ttpTo="20111031">
- <itdDateTime ttpFrom="20110110" ttpTo="20111031">
- <itdDate weekday="3" year="2011" month="02" day="99">
  <itdMessage type="error" module="BROKER" code="-1">invalid date</itdMessage>
  <itdMessage type="error" module="BROKER" code="-30">day value out of range</itdMessage>
</itdDate>
  <itdTime hour="7" minute="54"/>
</itdDateTime>
</itdTripDateTime>
```

## 3.1 Datum und Uhrzeit (4)

### Parameter zur Wahl der Uhrzeit (*itdTime*)

Default: Anfragezeit

Parametername	Bedeutung	Format
itdTime	Stunde, Minute	HHMM
timeOffset	Offset von der aktuellen Zeit (in Minuten)	MM
itdTimeAMPM	Zeit ist AM oder PM	am pm

## 3.1 Datum und Uhrzeit (5)

Anfragebeispiel:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?itdTime=1654](http://www.linzag.at/static/XML_TRIP_REQUEST2?itdTime=1654)

```
- <itdTripDateTime deparr="dep" ttpFrom="20110110" ttpTo="20111031">  
- <itdDateTime ttpFrom="20110110" ttpTo="20111031">  
  <itdDate weekday="3" year="2011" month="9" day="13"/>  
  <itdTime hour="16" minute="54"/>  
</itdDateTime>
```

Wird das Datum nicht angegeben, so wird das aktuelle Systemdatum übernommen.

## 3.1 Datum und Uhrzeit (6)

### Fehlercodes für die Uhrzeit (*itdMessage*)

Code	Bedeutung
-1	Ungültige Uhrzeit
-10	Stunde außerhalb des Wertebereichs
-20	Minute außerhalb des Wertebereichs

### Beispiel: Fehler -1 und -20

```
http://www.lin - <itdTripDateTime deparr="dep" ttpFrom="20110110" ttpTo="20111031">
- <itdDateTime ttpFrom="20110110" ttpTo="20111031">
  <itdDate weekday="3" year="2011" month="9" day="13"/>
  - <itdTime hour="16" minute="99">
    <itdMessage type="error" module="BROKER" code="-1">invalid time</itdMessage>
    <itdMessage type="error" module="BROKER" code="-20">minute value out of range</itdMessage>
  </itdTime>
</itdDateTime>
</itdTripDateTime>
```

## 3.2 Verifikation von Punkten

Die Punktverifikation wird von nahezu allen Requests benötigt. Sie kann auf viele verschiedene Arten erfolgen. Punkte können beispielsweise über Koordinaten, über Haltestellen-Identifikationsnummern, POI (Point of Interest) oder durch freien Suchtext ermittelt werden. Im Folgenden werden die HTTP-Parameter vorgestellt, durch die die genannten und weitere Eingabearten realisiert werden können. Zusätzliche Parameter dienen dazu das Suchverhalten zu beeinflussen und zu optimieren. Andere wirken als Filter und schränken den Suchraum ein oder erwirken eine Reduzierung der Trefferlisten.

## 3.2 Verifikation von Punkten

Die im Zusammenhang der Punktverifikation häufig gebrauchte Abkürzung ODV steht für Origin-Destination-Via. Sie bezieht sich auf die Funktion, die der Punkt in-nerhalb eines Requests inne hat. Die Funktion eines Punktes wird durch die Ergänzung `_<usage>` am Ende eines Parameters bestimmt. Parameter, die eine solche Ergänzung benötigen, werden für jeden Punkt gemäß seiner Funktionalität separat definiert.

Folgende Werte kann `<usage>` annehmen:

- origin, destination (für Trip-Request)
- dm (für DM-Request)
- sf (für StopFinder-Request)

## 3.2 Verifikation von Punkten

### Beispiel

#### XML-Schema: *itdOdv*

Odv ist eine Abkürzung für „origin destination via“.

Das Element *itdOdv* enthält den Ort *itdOdvPlace* und den Namen *itdOdvName* eines Punktes.

In der Beispielausgabe wurde noch keine Anfrage nach einem Punkt per XML Request gestellt. Ein Punkt kann eine Haltestelle, eine Adresse oder ein besonderer Punkt (POI) sein. Unter Punktverifikation wird hier die Suche nach Punkten im Auskunftssystem über Text / Strings verstanden.

```
- <itdOdv type="unknown" usage="origin">  
  - <itdOdvPlace state="empty">  
    <odvPlaceElem/>  
  </itdOdvPlace>  
  - <itdOdvName state="empty" method="itp">  
    <odvNameElem/>  
  </itdOdvName>  
</itdOdv>
```

## 3.2 Verifikation von Punkten (2)

Die Punkte werden durch eine Erweiterung des Parameternamens „usage“ unterschieden:

**origin** – Startpunkt (Trip-Request, PS-Request,...)

**destination** – Zielpunkt (Trip-Request, PS-Request,...)

**via** – Zwischenhalt (Trip-Request)

**dm** – Abfahrtshaltestelle (DM-Request)

**sf** – Haltestellensuche (Stop-Finder-Request)

...



Ein Punkt kann eine Haltestelle, eine Adresse, ein POI, usw. sein. Der Wert des Parameters ist abhängig vom Status der Punktverifikation. Der Status des Punktes ist dem Attribut state des Elements `itdOdvName` zu entnehmen:

`state=empty` oder `state=notidentified`: `name_<usage>` enthält die Eingabe des Anwenders.

`state=list`: `name_<usage>` enthält den Wert (Attribut value aus XML-Element `odvNameElem`) des ausgewählten Elements. Dieser Fall tritt ein, wenn kein Volltreffer ermittelt werden konnte und in der vorherigen Anfrage eine Liste zurückgegeben wurde.

`state=identified`: `name_<usage>` enthält den Wert des Attributs `statless` (siehe Parameter `stateless=1`) aus dem XML-Element `odvNameElem`.

## 3.2 Verifikation von Punkten (3)

### Obligatorische Parameter:

#### **locationServerActive = 1**

Aktiviert den EFA Location Server.

#### **name\_<usage>**

Start-, Ziel- oder Via-Punkt. Entspricht dem Eingabefeld (siehe auch Erklärung S. 22).

#### **type\_<usage> = any**

Dieser Parameter beschreibt den Typ des Punktes. Mögliche Werte sind:

- poiID
- stopID
- coord
- any

#### **stateless = 1**

Fügt identifizierten Punkten im Element in der XML-Ausgabe das Attribut stateless hinzu. Der Wert des Attributs ergibt bei der Punktsuche immer einen Volltreffer, wenn er als Wert des Parameters name\_<usage> übergeben wird.

## 3.2 Verifikation von Punkten (4)

Aufgabe: Verifizieren sie den „Hauptbahnhof Linz“ als Startpunkt.

### Lösung:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?locationServerActive=1&stateless=1&type\\_origin=any&name\\_origin=Hauptbahnhof](http://www.linzag.at/static/XML_TRIP_REQUEST2?locationServerActive=1&stateless=1&type_origin=any&name_origin=Hauptbahnhof)

Das Ergebnis ist kein einzelner Punkt sondern eine Liste.

Eine Meldung itdMessage weist auch darauf hin.

Die Listenelemente können über die **stateless Attribute** zugegriffen werden.

```
<?xml version="1.0" encoding="UTF-8"?>
- <itdRequest language="de" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" lengthUnit="METER"
nowWD="5" now="2015-12-17T11:04:44" virtDir="static" version="9.16.33.3" clientIP="10.250.39.16"
serverID="lagaspefaocri.linzag.at" client="NC6" sessionID="0"
xsi:noNamespaceSchemaLocation="http://www.linzag.at/static/Schema/itd.xsd">
  <clientHeaderLines/>
  - <itdTripRequest requestID="0">
    <itdAddress addressExt2="" addressExt1="" street="" place="" name=""/>
    - <itdOdv type="any" anyObjFilter="0" usage="origin">
      - <itdOdvPlace state="empty">
        <odvPlaceElem/>
      </itdOdvPlace>
      - <itdOdvName state="list" method="itp">
        <itdMessage type="error" code="-8011" module="BROKER"/>
        <odvNameElem id="60501720" value="0:1" posttown="Linz/Donau" matchQuality="1000"
        buildingNumber="" buildingName="" objectName="Hauptbahnhof" postCode=""
        locality="Linz/Donau" nameKey="" anyTypeSort="2" anyType="stop" mapName="NAV5"
        y="809422" x="5447580" placeID="4" omc="44001001" streetName="" listIndex="0" selected="1"
        stateless="60501720">Linz/Donau, Hauptbahnhof</odvNameElem>
        <odvNameElem id="1" value="1:2" posttown="" matchQuality="965" buildingNumber="" buildingName=""
        objectName="Hauptbahnhof Linz" postCode="" locality="Linz/Donau" nameKey="" anyTypeSort="4"
        anyType="poi" mapName="NAV5" y="809391" x="5447564" placeID="-1" omc="44001001"
        streetName="" listIndex="1" selected="0" stateless="poiID:1:44001001:-1">Linz/Donau,
        Hauptbahnhof Linz</odvNameElem>
        <odvNameElem id="45" value="2:3" posttown="" matchQuality="932" buildingNumber="" buildingName=""
        objectName="Touristeninformation am HBF" postCode="" locality="Linz/Donau" nameKey=""
        anyTypeSort="4" anyType="poi" mapName="NAV5" y="809550" x="5447498" placeID="-1"
        omc="44001001" streetName="" listIndex="2" selected="0"
        stateless="poiID:45:44001001:-1">Linz/Donau, Touristeninformation am HBF</odvNameElem>
        <odvNameInput>Hauptbahnhof</odvNameInput>
      </itdOdvName>
    </itdOdv>
  </itdTripRequest>
</itdRequest>
```

Mit der folgenden Anfrage identifiziert man den Hauptbahnhof:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?locationServerActive=1&stateless=1&type\\_origin=any&name\\_origin=60501720](http://www.linzag.at/static/XML_TRIP_REQUEST2?locationServerActive=1&stateless=1&type_origin=any&name_origin=60501720)

## 3.2 Verifikation von Punkten (5)

Ein Punkt kann eine Haltestelle, eine Adresse, ein POI, usw. sein. Der Wert des Parameters ist abhängig vom Status der Punktverifikation. Der Status des Punktes ist dem Attribut state des Elements itdOdvName zu entnehmen:

- empty – es hat keine Verifikation stattgefunden
- identified – Volltreffer
- list – Trefferliste
- notidentified – Verifikation fehlgeschlagen

## 3.2 Verifikation von Punkten (7)

### Aufgabe:

Wählen Sie die Touristeninformation am Hauptbahnhof aus.

### Hinweis:

Der Index des zu wählenden Listenelements *odvNameElem* befindet sich im Attribut *stateless* und wird durch den Parameter *name\_<usage>* übergeben.

Bei POIs wird der Suffix *poilD*: aus dem *stateless* Attribut der XML Antwort weggelassen. Der *type\_<usage>* Parameter ist mit *poilD* zu belegen.

## 3.2 Verifikation von Punkten (8)

Lösung:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?locationServerActive=1&stateless=1&type\\_origin=poilD&name\\_origin=45:44001001:-1](http://www.linzag.at/static/XML_TRIP_REQUEST2?locationServerActive=1&stateless=1&type_origin=poilD&name_origin=45:44001001:-1)

Das stateless Attribut stammt aus dem vorherigen Beispiel.

## 3.2 Verifikation von Punkten (9)

Die Parameter *type\_<usage>* und *name\_<usage>* können durch die Parameter

**typeInfo\_<usage>**  
**nameInfo\_<usage>**

übersteuert werden. Wird der Parameter übergeben, soll aber deaktiviert sein, ist sein Wert *invalid*.

Beispiel:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?locationServerActive=1&stateless=1&type\\_origin=any&name\\_origin=HauptbahnhofLinz&nameInfo\\_origin=Neues Rathaus](http://www.linzag.at/static/XML_TRIP_REQUEST2?locationServerActive=1&stateless=1&type_origin=any&name_origin=HauptbahnhofLinz&nameInfo_origin=Neues Rathaus)

## 3.2 Verifikation von Punkten (10)

**Das Eingabefeld mit einem Hinweistext vorbelegen:**

### **nameDefaultText\_<usage>**

Der Wert des Parameters ist ein Text, der bei der Verifikation des durch name\_<usage> angegebenen Punktes nicht berücksichtigt werden soll. Auf diese Weise kann ein Hinweistext (z.B. „Bitte Haltestelle, Adresse oder POI eingeben!“) im Eingabefeld angezeigt werden.



## 3.2 Verifikation von Punkten (11)

Beispiel:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?locationServerActive=1&stateless=1&type\\_origin=any&name\\_origin=Haltestelle, Adresse oder wichtiger Punkt](http://www.linzag.at/static/XML_TRIP_REQUEST2?locationServerActive=1&stateless=1&type_origin=any&name_origin=Haltestelle, Adresse oder wichtiger Punkt)

Im Vergleich zu:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?locationServerActive=1&stateless=1&type\\_origin=any&name\\_origin=Haltestelle, Adresse oder wichtiger Punkt&nameDefault\\_origin=Haltestelle, Adresse oder wichtiger Punkt](http://www.linzag.at/static/XML_TRIP_REQUEST2?locationServerActive=1&stateless=1&type_origin=any&name_origin=Haltestelle, Adresse oder wichtiger Punkt&nameDefault_origin=Haltestelle, Adresse oder wichtiger Punkt)

## 3.2 Verifikation von Punkten (12)

### Einschränkung des Suchraums:

#### **anyObjFilter\_<usage>**

Der Suchraum kann auf bestimmte Objekttypen eingegrenzt werden. Der Wert ist eine Bitmaske mittels der die Objekttypen miteinander kombiniert werden können.

#### Beispiel:

anyObjFilter\_<usage>=34 → (2 + 32)  
Haltestellen und wichtige Punkte.

Wert	Bedeutung
0	Kompletter Suchraum
1	Orte
2	Haltestellen
4	Straßen
8	Adressen
16	Kreuzungen
32	Wichtige Punkte
64	Postleitzahlen

## 3.2 Verifikation von Punkten (13)

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?locationServerActive=1&stateless=1&type\\_origin=any&name\\_origin=LinzWifi&anyObjFilter\\_origin=2](http://www.linzag.at/static/XML_TRIP_REQUEST2?locationServerActive=1&stateless=1&type_origin=any&name_origin=LinzWifi&anyObjFilter_origin=2)

```
- <itdRequest language="de" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" lengthUnit="METER"
nowWD="5" now="2015-12-17T12:01:07" virtDir="static" version="9.16.33.3" clientIP="10.250.39.16"
serverID="lagaspefaocr.linzag.at" client="NC6" sessionID="0"
xsi:noNamespaceSchemaLocation="http://www.linzag.at/static/Schema/itd.xsd">
  <clientHeaderLines/>
  - <itdTripRequest requestID="0">
    <itdAddress addressExt2="" addressExt1="" street="" place="" name=""/>
    - <itdOdv type="any" anyObjFilter="2" usage="origin">
      - <itdOdvPlace state="identified" method="itp">
        <odvPlaceElem type="remote" value="44001001:4" placeID="4" omc="44001001"
stateless="placeID:44001001:4" span="0" mainPlace="1">Linz/Donau</odvPlaceElem>
        <odvPlaceInput/>
      </itdOdvPlace>
      - <itdOdvName state="identified" method="itp">
        <odvNameElem id="60501200" posttown="Linz/Donau" buildingNumber="" buildingName=""
objectName="WIFI/LINZ AG" postCode="" locality="Linz/Donau" nameKey="" anyType="stop"
mapName="NAV5" y="810583" x="5448851" placeID="4" omc="44001001" streetName=""
stateless="60501200">Linz/Donau, WIFI/LINZ AG</odvNameElem>
        <odvNameInput>LinzWifi</odvNameInput>
      </itdOdvName>
      - <itdOdvAssignedStops select="0">
        <itdOdvAssignedStop value="60501200:WIFI/LINZ AG" place="Linz/Donau" mapName="NAV5"
y="810583" x="5448851" isTransferStop="0" stopID="60501200" distanceTime="0"
nameWithPlace="Linz WIFI/LINZ AG">WIFI/LINZ AG</itdOdvAssignedStop>
      </itdOdvAssignedStops>
      - <genAttrList>
        - <genAttrElem>
          <name>anyObjFilter</name>
          <value>STOP</value>
        </genAttrElem>
      </genAttrList>
    </itdOdv>
  </itdTripRequest>
</itdRequest>
```

## 3.2 Verifikation von Punkten (14)

Soll ein Punkt nicht verifiziert werden, sondern ein bekannter Punkt verwendet werden, kann dieser über seine ID oder Koordinate identifiziert werden.

### **Beispiel Haltestelle:**

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?type\\_origin=stopID&name\\_origin=60501170](http://www.linzag.at/static/XML_TRIP_REQUEST2?type_origin=stopID&name_origin=60501170)

### **Beispiel POI:**

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?type\\_origin=poiID&name\\_origin=45:44010021:-1](http://www.linzag.at/static/XML_TRIP_REQUEST2?type_origin=poiID&name_origin=45:44010021:-1)

Hinweis: Zusätzlich zur ID des POIs müssen die GKZ und die ID des Ortes angegeben werden (Bsp. Touristeninfo am HBF).

## 3.2 Verifikation von Punkten (15)

### Beispiel Koordinate:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?type\\_origin=coord  
&name\\_origin=5448851:810583:NAV5](http://www.linzag.at/static/XML_TRIP_REQUEST2?type_origin=coord&name_origin=5448851:810583:NAV5) → WIFI, schnappt auf die  
nächste Kante

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?type\\_origin=coord  
&name\\_origin=5443851:812583:NAV5:Ein schöner Ort](http://www.linzag.at/static/XML_TRIP_REQUEST2?type_origin=coord&name_origin=5443851:812583:NAV5:Ein%20sch%C3%B6ner%20Ort)  
→ Beliebige Koordinate mit Freitext (z.B. „Ein schöner Ort“)

Hinweis: Um eine Koordinate identifizieren zu können, müssen  
(mindestens) x, y und das Koordinatensystem angegeben werden.

## 3.2 Verifikation von Punkten (16)

### Fehlerbehandlung:

Analog zum Datums- und Uhrzeitelement werden im Element *itdOdv* Fehlermeldungen *itdMessage* ausgegeben. Die Fehlerbeschreibungen sind dem **Dokument EFA9\_Errorcodes\_V1.0** zu entnehmen.

## 4. Stop-Finder-Request

Die Verifikation eines Punktes kann auch zur Suche von Haltestellen verwendet werden. Dies wird mit folgender Anfrage erreicht:

[http://www.linzag.at/static/XML\\_STOPFINDER\\_REQUEST?](http://www.linzag.at/static/XML_STOPFINDER_REQUEST?)

Teil der Haltestellensuche:

- Verifikation von Punkten (3.2)

Punktverifikation: Der Parameter-Suffix *<usage>* nimmt den Wert *sf* an.

## 4. Stop-Finder-Request (2)

Der Stop-Finder-Request nutzt die Parameter zur Punktverifikation, hat aber keine eigenen Parameter.

Anwendungsbeispiel Vorschlagsliste bei der Punkt-Eingabe

[http://www.linzag.at/static/XML\\_STOPFINDER\\_REQUEST?locationServerActive=1&outputFormat=XML&type\\_sf=any&name\\_sf=Hauptbahnhof](http://www.linzag.at/static/XML_STOPFINDER_REQUEST?locationServerActive=1&outputFormat=XML&type_sf=any&name_sf=Hauptbahnhof)

```
- <itdStopFinderRequest requestID="0">
- <itdOdv type="any" usage="sf" anyObjFilter="0">
- <itdOdvPlace state="empty">
  <odvPlaceElem/>
</itdOdvPlace>
- <itdOdvName state="list">
  <itdMessage type="error" module="BROKER" code="-8011"/>
  <odvNameElem listIndex="0" selected="1" matchQuality="1000" x="5447580" y="809422"
  mapName="NAV5" id="60501720" omc="44001001" placeID="4" anyType="stop" anyTypeSort="2"
  locality="Linz/Donau" objectName="Hauptbahnhof" buildingName="" buildingNumber="" postCode=""
  streetName="" nameKey="" posttown="Linz/Donau" value="0:1">Linz/Donau,
  Hauptbahnhof</odvNameElem>
  <odvNameElem listIndex="1" selected="0" matchQuality="965" x="5447564" y="809391"
  mapName="NAV5" id="1" omc="44001001" placeID="-1" anyType="poi" anyTypeSort="4"
  locality="Linz/Donau" objectName="Hauptbahnhof Linz" buildingName="" buildingNumber=""
  postCode="" streetName="" nameKey="" posttown="" value="1:2">Linz/Donau, Hauptbahnhof
  Linz</odvNameElem>
```



# 4.1 Anwendungsbeispiel Vorschlagsliste

Von **Haupt**

Nach

- Linz/Donau, Brucknerhaus Haupteingang
- Linz/Donau, Hauptbahnhof
- Linz/Donau, Hauptplatz
- Linz/Donau, Anton-Bruckner-Privatuniversität/Hauptgebäude
- Linz/Donau, Hauptbahnhof Linz
- Linz/Donau, Hauptbücherei der Stadt Linz/Mediathek
- Linz/Donau, Hauptplatz
- Linz/Donau, Touristeninformation am HBF

+ We...

Fa...

Früh...

1. Ansfelden (OÖ) Hauptplatz

2. Linz/Donau Hauptplatz

08:52 Linz/Donau Hauptstraße

08:58 Traun (OÖ) Hauptplatz

an Linz WIFI/LINZ AG

+ 3. Fahrt	08:57	09:03	00:06	0x
+ 4. Fahrt	09:02	09:08	00:06	0x

Später  Letzte Fahrt

▸ [Details für alle Fahrten anzeigen](#) ▸ [Rückfahrt](#) ▸ [Weiterfahrt](#) ▸ [Neue Fahrt](#)

**Karte**

**Linieninformation**

**Drucken**

<input checked="" type="checkbox"/> 1.	08:47 - 08:53 Uhr	2
<input checked="" type="checkbox"/> 2.	08:52 - 08:58 Uhr	1
<input checked="" type="checkbox"/> 3.	08:57 - 09:03 Uhr	2
<input checked="" type="checkbox"/> 4.	09:02 - 09:08 Uhr	1

## 4.1 Anwendungsbeispiel Vorschlagsliste (2)

### **outputFormat**

Bestimmt das Format der Ausgabe.

Werte: HTML, XML, JSON

Hinweis: Der Parameter gilt auch bei verschiedenen anderen Requests.

Für eine mittels AJAX angefragte Vorschlagsliste empfiehlt sich die Ausgabe im JSON-Format.

### Beispiel:

[http://www.linzag.at/static/XML\\_STOPFINDER\\_REQUEST?locationServerActive=1&outputFormat=JSON&type\\_sf=any&name\\_sf=Hauptbahnhof](http://www.linzag.at/static/XML_STOPFINDER_REQUEST?locationServerActive=1&outputFormat=JSON&type_sf=any&name_sf=Hauptbahnhof)

# Ausgabebeispiel – JSON

[http://www.linzag.at/static/XML\\_STOPFINDER\\_REQUEST?locationServerActive=1&outputFormat=JSON&type\\_sf=any&name\\_sf=Hauptbahnhof](http://www.linzag.at/static/XML_STOPFINDER_REQUEST?locationServerActive=1&outputFormat=JSON&type_sf=any&name_sf=Hauptbahnhof)

```
{ "parameters": [ { "name": "requestID", "value": "0" }, { "name": "sessionID", "value": "0" } ], "message": [ { "name": "code", "value": "-8011" }, { "name": "error", "value": "" } ], "input": [ { "name": "input", "value": "Hauptbahnhof" } ], "stopFinder": [ { "usage": "sf", "type": "any", "name": "Linz/Donau, Hauptbahnhof", "stateless": "60501720", "anyType": "stop", "sort": "2", "quality": "1000", "object": "Hauptbahnhof", "posttown": "Linz/Donau", "ref": { "id": "60501720", "omc": "44001001", "placeID": "4", "place": "Linz/Donau", "coords": "5447580,809422" } }, { "usage": "sf", "type": "any", "name": "Linz/Donau, Hauptbahnhof Linz", "stateless": "poiID:1:44001001:-1", "anyType": "poi", "sort": "4", "quality": "965", "object": "Hauptbahnhof Linz", "ref": { "id": "1", "omc": "44001001", "placeID": "-1", "place": "Linz/Donau", "coords": "5447564,809391" } }, { "usage": "sf", "type": "any", "name": "Linz/Donau, Touristeninformation am HBF", "stateless": "poiID:45:44001001:-1", "anyType": "poi", "sort": "4", "quality": "932", "object": "Touristeninformation am HBF", "ref": { "id": "45", "omc": "44001001", "placeID": "-1", "place": "Linz/Donau", "coords": "5447498,809550" } } ] ] }
```

## 5. Trip-Request – Fahrtanfrage

Berechnung von Fahrten (ÖV) von A nach B (über C) zu einem bestimmten Datum und einer bestimmten Uhrzeit. Die Anpassung an individuelle Bedürfnisse erfolgt über die Fahrtoptionen (4.2).

Anfrage:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?](http://www.linzag.at/static/XML_TRIP_REQUEST2?)

Infolge werden benötigt:

Teil der Fahrtanfrage:

- Datum und Uhrzeit (3.1)
- Verifikation von Punkten (3.2)

Punktverifikation: Der Parameter-Suffix *<usage>* kann den Wert *origin*, *destination* oder *via* annehmen.

## 5. Trip-Request – Fahrtanfrage (2)

1. Ergänzung zu Datum und Uhrzeit
2. Fahrtoptionen
3. Fußwegbeschreibung
4. Exkurs: Koordinaten
5. Echtzeit

## 4.1 Ergänzung zu Datum und Uhrzeit

### **itdTripDateTimeDepArr**

Bestimmt, ob sich Datum und Uhrzeit einer Fahrt auf Abfahrt oder Ankunft beziehen.

Werte: **dep** | **arr**

Beispiel:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?itdDate=20110225&itdTime=16:45&itdTripDateTimeDepArr=arr](http://www.linzag.at/static/XML_TRIP_REQUEST2?itdDate=20110225&itdTime=16:45&itdTripDateTimeDepArr=arr)

## 4.2 Fahrtoptionen

Beeinflussung der Fahrtergebnisse, so dass sie den persönlichen Bedürfnissen entsprechen.

XML-Schema: *itdTripOptions*

## 4.2 Fahrtoptionen (2)

Die Optionen sind in verschiedenen Untergruppen organisiert:

### XMLSchema:

- *itdUsedOptions* – allgemeine Optionen
- *itdPtOptions* – Optionen für den ÖV
- *itdItOptions* – Optionen für den Individualtransport

### Parameter zur Aktivierung verschiedener Gruppen von Optionen:

- **ptOptionsActive=1** – Optionen für den ÖV
- **impairedOptionsActive=1** – Mobilitätseinschränkungen
- **itOptionsActive=1** – Optionen für den Individualtransport

Standardmäßig sind die Parameter deaktiviert (0).



## 4.2 Fahrtoptionen (3)

**Allgemeine Parameter (Auswahl):** (*itdUsedOptions*)

### **calcNumberOfTrips**

Anzahl der ÖV Fahrten, die berechnet werden (Alternativen nicht eingerechnet)

Hinweis: Mit *calcNumberOfTrips=0* ist Berechnung von ÖV-Fahrten deaktiviert.

## 4.2 Fahrtoptionen (4)

Parameter für ÖV Optionen (Auswahl): **ptOptionsActive = 1**

**useProxFootSearch = 1**

Berücksichtigung von nahegelegenen Haltestellen.

Zur Unterscheidung der Start- und Zielhaltestelle kann der Parametername durch die Erweiterung *Orig* bzw. *Dest* ergänzt werden.

## 4.2 Fahrtoptionen (5)

### maxChanges

Anzahl der Umstiege.

Werte:

- 0 – Direktfahrt
- 1 – 1 x umsteigen
- 2 – 2 x umsteigen
- **9** – 9 x umsteigen

### routeType

Optimierung der Fahrtauskunft:

Werte:

- **LEASTTIME**
- LEASTINTERCHANGE
- LEASTWALKING

Zeitkürzeste Verbindung

Verbindung mit wenig Umsteigen

Verbindung mit wenig Fußwegen

## 4.2 Fahrtoptionen (6)

### Parameter zum Ausschluss von Verkehrsmitteln:

#### **excludedMeans = checkbox**

Aktiviert den Verkehrsmittelausschluss mittels des Parameters `exclMOT_<ID> = 1`.

#### **exclMOT\_<ID> = 1**

Schließt das Verkehrsmittel mit der ID `<ID>` aus. Kann mehrfach verwendet werden.

#### Alternative:

#### **excludedMeans = <ID>**

Mehrfachverwendung möglich.

\*Die Pöstlingbergbahn wird unter der Verkehrsmittelklasse Seil-/Zahnradbahnen synonym mit Bergbahnen geführt.

<ID>	Verkehrsmitteltyp
0	Zug
1	S-Bahn
2	U-Bahn
3	Stadtbahn
4	Straßen-/Trambahn
5	Stadtbus
6	Regionalbus
7	Schnellbus
8	Seil-/Zahnradbahn*
9	Schiff
10	AST/Rufbus
11	Sonstige

## 4.2 Fahrtoptionen (7)

### Aufgabe:

Fahren Sie von WIFI Linz AG zur Johannes Kepler Universität ohne die Tram zu benutzen.

### Tipp:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?locationServerActive=1&stateless=1&type\\_origin=any&name\\_origin=WIFI Linz AG&type\\_destination=any&name\\_destination=Johannes Kepler Universität](http://www.linzag.at/static/XML_TRIP_REQUEST2?locationServerActive=1&stateless=1&type_origin=any&name_origin=WIFI%20Linz%20AG&type_destination=any&name_destination=Johannes%20Kepler%20Universit%C3%A4t)

## 4.2 Fahrtoptionen (8)

### Lösung:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?locationServerActive=1&stateles=1&type\\_origin=any&name\\_origin=WIFI Linz AG&type\\_destination=any&name\\_destination=Linz Johannes Kepler Universität&ptOptionsActive=1&excludedMeans=4](http://www.linzag.at/static/XML_TRIP_REQUEST2?locationServerActive=1&stateles=1&type_origin=any&name_origin=WIFI Linz AG&type_destination=any&name_destination=Linz Johannes Kepler Universität&ptOptionsActive=1&excludedMeans=4)

```

- <itdTripOptions userDefined="0">
- <itdPtOptions active="1" maxChanges="9" maxTime="360" maxWait="120" routeType="LEASTTIME"
  changeSpeed="normal" lineRestriction="400" useProxFootSearch="0" useProxFootSearchOrigin="0"
  useProxFootSearchDestination="0" bike="0" plane="0" noCrowded="0" noSolidStairs="0"
  noEscalators="0" noElevators="0" lowPlatformVhcl="0" wheelchair="0" SOSAvail="0"
  noLonelyTransfer="0" illumTransfer="0" overgroundTransfer="0" noInsecurePlaces="0"
  privateTransport="0">
- <excludedMeans>
  <meansElem value="0" selected="0">Zug</meansElem>
  <meansElem value="1" selected="0">S-Bahn</meansElem>
  <meansElem value="2" selected="0">U-Bahn</meansElem>
  <meansElem value="3" selected="0">Stadtbahn</meansElem>
  <meansElem value="4" selected="1">Straßen-/Trambahn</meansElem>
  <meansElem value="5" selected="0">Stadtbus</meansElem>
  <meansElem value="6" selected="0">Regionalbus</meansElem>
  <meansElem value="7" selected="0">Schnellbus</meansElem>
  <meansElem value="8" selected="0">Seil-/Zahnradbahn</meansElem>
  <meansElem value="9" selected="0">Schiff</meansElem>
  <meansElem value="10" selected="0">AST/Rufbus</meansElem>
  <meansElem value="11" selected="0">Sonstige</meansElem>
</excludedMeans>

```

## 4.2 Fahrtoptionen (9)

**Außerdem:**

### **changeSpeed**

Gehgeschwindigkeit beim Umsteigen bei *ptOptionsActive=1* und Geschwindigkeit für Weg zur Starthaltestelle bzw. von der Zielhaltestelle bei *itOptionsActive=1*.

Geschwindigkeit [km/h] = (100 x Standardgeschwindigkeit) / Wert  
Werte: **normal** (->100), **slow** (->50), **fast** (->200), [25..400]

## 4.2 Fahrtoptionen (10)

### Aufgabe 1: Menschen mit speziellen Bedürfnissen

Stellen Sie sich vor, Sie sind mit einer älteren Dame in Linz unterwegs. Sie wollen vom Hauptbahnhof zum Landeskulturzentrum fahren und haben noch ein wenig Zeit. Die ältere Dame möchte auf keinen Fall die Strassenbahn benutzen. Außerdem kann Sie nicht gut Treppen steigen und braucht etwas länger zum umsteigen. Welche Optionen würden Sie wählen?

Tipp:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?locationServerActive=1&stateless=1&type\\_origin=stopID&name\\_origin=LinzHauptbahnhof&type\\_destination=any&name\\_destination=LinzLandeskulturzentrum](http://www.linzag.at/static/XML_TRIP_REQUEST2?locationServerActive=1&stateless=1&type_origin=stopID&name_origin=LinzHauptbahnhof&type_destination=any&name_destination=LinzLandeskulturzentrum)



## 4.2 Fahrtoptionen (11)

### Lösung:

z.B:

[http://www.linzag.at/static/XML\\_TRIP\\_REQUEST2?locationServerActive=1&type\\_origin=stopID&name\\_origin=Linz%20Hauptbahnhof&type\\_destination=any&name\\_destination=Linz%20Landeskulturzentrum&ptOptionsActive=1&itOptionsActive=1&excludedMeans=4&changeSpeed=slow&routeType=LEASTINTERCHANGE&impairedOptionsActive=1&noSolidStairs=1&lowPlatformVhcl=1](http://www.linzag.at/static/XML_TRIP_REQUEST2?locationServerActive=1&type_origin=stopID&name_origin=Linz%20Hauptbahnhof&type_destination=any&name_destination=Linz%20Landeskulturzentrum&ptOptionsActive=1&itOptionsActive=1&excludedMeans=4&changeSpeed=slow&routeType=LEASTINTERCHANGE&impairedOptionsActive=1&noSolidStairs=1&lowPlatformVhcl=1)

```
- <itdItinerary>  
  - <itdRouteList>  
    - <itdRoute active="1" selected="1" changes="0" distance="167" routeIndex="1" routeTripIndex="1"  
      alternative="0" print="0" delete="0" searchMode="-1" cTime="20110913094632060" method="drs"  
      individualDuration="00:06" vehicleTime="6" publicDuration="00:12">
```

## 5. DM-Request - Abfahrtsmonitor

Abfahrtsmonitor einer Haltestelle.

Anfrage:

[http://www.linzag.at/static/XML\\_DM\\_REQUEST?](http://www.linzag.at/static/XML_DM_REQUEST?) Oder

Teil des Abfahrtsmonitors:

Datum und Uhrzeit (3.1)

Verifikation von Punkten (3.2)

Punktverifikation: Der Parameter-Suffix *<usage>* nimmt den Wert *dm* an.

## 5. DM-Request – Abfahrtsmonitor (2)

1. Ergänzung zu Datum und Uhrzeit
2. Ergänzung zur Verifikation von Punkten
3. DM-Optionen

## 5.1 Ergänzung zu Datum und Uhrzeit

### Parameter zur Unterscheidung von Abfahrt und Ankunft:

#### **itdDateTimeDepArr**

Unterscheidung zwischen Abfahrts- und Ankunftsmonitor

Werte: **dep** | arr

#### XML-Schema:

Die Ankünfte *itdDeparture* werden in einer Liste *itdDepartureList* gespeichert.  
Die Abfahrten *itdArrival* in einer List *itdArrivalList*.

#### **limit**

Dieser Parameter legt die maximale Anzahl der Abfahrten fest, die ausgegeben werden. Per Default werden 40 Fahrten innerhalb eines Zeitraums von maximal zwei Tagen ausgegeben.

## 5.2 Ergänzung zur Verifikation von Punkten

Soll der Abfahrtsmonitor nur für eine Halteselle angezeigt werden, muss gegebenenfalls die zugeordnete Halteselle in einem Zwischenschritt ermittelt werden.

Zugeordnete Haltestellen werden für Adressen und POIs ermittelt.

Aber auch komplexere Haltestellen, wie der Hauptbahnhof von Linz, können durch mehrere Haltestellen modelliert sein.

Sind zugeordnete Haltestellen vorhanden, sind diese als `itdAssignedStop` im Punkte `itdOdv` in der XML Antwort vorhanden. Die Wahl einer zugeordneten Haltestelle erfolgt über den Parameter `name_<usage>`. Als Wert wird der Wert des Attributs `stateless` übergeben (siehe Kapitel [Punktverifikation](#)).

## 5.3 DM-Optionen

### line

Für eine verifizierte Haltestelle kann der Abfahrts- bzw. Ankunftsmonitor für eine konkrete Linie erstellt werden. Diese wird durch eine durch Doppelpunkt getrennte Liste von Teilnetz, DIVA-Liniennummer, Ergänzung und Richtung <Netz>:<DIVA-Linie>:<Ergänzung>:<Richtung>:<Fahrplanprojekt> angegeben. Soll dabei einer der Parameter nicht berücksichtigt werden, wird der Wert leer gelassen. Zur Wahl mehrerer Linien kann der Parameter mehrfach verwendet werden.

**Beispiel:** esg:01001:E:H:f15 **oder** esg:01001:::

## 5.3 DM-Optionen (1)

Einige der Fahrtoptionen des Trip-Requests gelten auch für den Abfahrtsmonitor. Zum Beispiel der Ausschluss von Verkehrsmitteln (4.2). Weitere Parameter:

### **IsShowTrainsExplicit = 1**

Standardmäßig berücksichtigt der Abfahrtsmonitor keine Züge. Die Anzeige der Züge in der Linienauswahl muss explizit durch diesen Parameter aktiviert werden.

### **mode = direct**

Wenn keine Auswahl von Linien stattfinden soll und stattdessen alle Linien ausgegeben werden sollen

## 5.3 DM-Optionen (2)

### Aufgabe:

Zeigen Sie einen Abfahrtsmonitor mit den nächsten 5 Abfahrten für die Linz AG an. Eine Einschränkung auf bestimmte Linien soll nicht vorgenommen werden.



## 5.3 DM-Optionen (3)

Lösung:

[http://www.linzag.at/static/XML\\_DM\\_REQUEST?locationServerActive=1&stateless=1&type\\_dm=any&name\\_dm=60501200&limit=5&mode=direct](http://www.linzag.at/static/XML_DM_REQUEST?locationServerActive=1&stateless=1&type_dm=any&name_dm=60501200&limit=5&mode=direct)

## 5.3 DM-Optionen (4)

### Aufgabe:

Zeigen Sie einen Abfahrtsmonitor mit den Abfahrten der Linie 2 von WIFI , Linz AG an.

## 5.3 DM-Optionen (5)

Beispiel:

Schritt 1:

[http://www.linzag.at/static/XML\\_DM\\_REQUEST?&locationServerActive=1&stateless=1&type\\_dm=any&name\\_dm=WIFI](http://www.linzag.at/static/XML_DM_REQUEST?&locationServerActive=1&stateless=1&type_dm=any&name_dm=WIFI) Linz AG

→ **stateless=60501200, line=„,esg:01001:E:H:f15“**

Schritt 2:

[http://www.linzag.at/static/XML\\_DM\\_REQUEST?&locationServerActive=1&stateless=1&type\\_dm=any&name\\_dm=60501200&line=esg:01002:E:H:f15&mode=direct](http://www.linzag.at/static/XML_DM_REQUEST?&locationServerActive=1&stateless=1&type_dm=any&name_dm=60501200&line=esg:01002:E:H:f15&mode=direct)