

Schnittstelle X-Sharing Router

Version 0.5.1

Max Wiederhold

28. Januar 2016

1 Endpunkte

Die Anfragen werden wie abgesprochen per Java Messaging Service (JMS) API in einem Request-Response Schema gestellt und beantwortet. Es ist pro Anfragetyp eine entsprechende Request-Queue vorgesehen, an welche die Anfragen im TextMessage-Format erfolgen. Dabei ist zu beachten, dass in den Nachrichten das Meta-Feld "ReplyToId" auf eine valide JMS-Destination gesetzt ist. Alternativ ist die Verwendung von CorrelationIds denkbar, allerdings bisher ungetestet.

Die jeweiligen aktuell verwendeten Definitionen für die Queues sind der folgenden Tabelle zu entnehmen:

Typ	JNDI-Name
Minimal	java:jboss/exported/jms/queue/xSharingMinimalRequestQueue
Compact	java:jboss/exported/jms/queue/xSharingCompactRequestQueue
Details	java:jboss/exported/jms/queue/xSharingDetailsRequestQueue

2 Anfragetypen & Parameter

2.1 Kommentare zur Schnittstelle

Die Anfragen an das System sind wie im Meeting am 18.06.15 besprochen (in Anlehnung an die URA3 Schnittstelle) aufgeteilt in drei Varianten. Hierbei ist insbesondere die MinimalRequest für die konkrete und hochfrequente Anfrage durch den intermodalen Router gedacht. Auf diese Art der Anfrage wird die kleinste nötige Repräsentation der berechneten Route ausgegeben. Mit der darin enthaltenen UUID lassen sich die Folgeanfragen Compact oder Details aufrufen, um weitere Informationen zur berechneten Route abzurufen. Die Antworten des Routers sind bis auf das Top-Level Element Datentypen aus der URA3 Spezifikation und daher größtenteils kompatibel zu bestehenden Systemen. Um den Overhead durch die hohe Anzahl der Anfragen zu reduzieren, erlaubt die Schnittstelle das Bündeln mehrerer Requests in einem Paket. Dazu wird die MinimalRequest wie folgt angepasst:

2.2 Minimal

Parameter Name	Typ	Beschreibung
startGeoCoordinates	List<GeoCoord>	Liste von Ursprungspunkten (lon/lat)
endGeoCoordinates	List<GeoCoord>	Liste von Zielpunkten (lon/lat)
departureTime	List<Timestamp>	Liste von Abfahrtszeiten (in milliseconds)
isArrivalTime	Boolean	verwende Ankunfts- statt Abfahrtszeit
Optionale Parameter		
isLastLeg	Boolean	legt fest ob Route direkt zum Ziel führen soll (\Rightarrow inkl. Auto), default: false
isWithReturn	Boolean	legt fest ob Route mit Rückreise geplant werden soll, default: false
maxWalkDistance	Double	die maximale Laufdistanz pro Weg (noch nicht implementiert)
stayTime	Integer	(nur wenn isWithReturn) Aufenthaltszeit am Zielort (in seconds), default: 0
modalTypes	String[]	Liste gewünschter Transportmodi (noch nicht implementiert)

Listing 1: Beispiel MinimalRequest.

```

1 {
2     "startGeoCoordinates": [{
3         "longitude": 6.062009,
4         "latitude": 50.768106
5     }],
6     "endGeoCoordinates": [{
7         "longitude": 6.1296,
8         "latitude": 50.774048
9     }],
10    {
11        "longitude": 6.1196,
12        "latitude": 50.764048
13    }],
14    "departureTime": [1439831784489],
15    "isArrivalTime": false,
16    "isLastLeg": true
17 }
```

Hier ist zu beachten, dass durch die Darstellung der Parameter in Listen vier verschiedene $m \times n$ Ausprägungen der Anfrage ermöglicht werden. Diese Ausprägungen sind eindeutig definiert durch die Kardinalitäten der jeweiligen Listen. Anfragen, die nicht entsprechend der erlaubten Belegungen validierbar sind, werden vom System verworfen. Die folgende Tabelle verdeutlicht die möglichen Kombinationen.

Tabelle 3: Erlaubte Kombinationen der Kardinalitäten

Start	Ziel	Zeit	isArrival
1	n	1	false
1	n	n	true
n	1	1	true
n	1	n	false

2.3 Compact

Parameter Name	Typ	Beschreibung
tripId	String	Routen-ID aus vorheriger Antwort auf MinimalRequest

2.4 Details

Parameter Name	Typ	Beschreibung
tripId	String	Routen-ID aus vorheriger Antwort auf MinimalRequest

3 Antwortparameter

3.1 SharingResponse

Als Antwort auf eine beliebige Request wird immer eine Liste von Elementen ausgegeben. Im Falle der Compact- und DetailsResponse wird die Liste exakt ein Element enthalten, bei MinimalRequests wird die Anzahl der Kardinalität der jeweiligen Anfragelisten entsprechen. In der Antwort werden zur Zuordnung der Listenelemente zu der entsprechenden Anfrage die Start- und Zielkoordinate eingebunden. Das minimale Top-Level Element ist wie folgt definiert:

Parameter Name	Typ	Beschreibung
tripId	String	ID der Route (für nachfolgende Anfragen)
startGeoCoordinates	GeoCoord	Ursprungspunkt der Route (lon/lat)
endGeoCoordinates	GeoCoord	Zielpunkt der Route (lon/lat)
totalDuration	Zahl	Gesamtdauer der Route in Sekunden
totalDistance	Zahl	Gesamtdistanz der Route in Metern
transfers	Zahl	Anzahl der Umstiege
route	List<IndividualTrip>	Liste von (URA3) Trip Elementen, Ausprägung je nach Anfragetyp. Nur in Compact und Details
routerError	RouterError	Typ für Error-Handling

Der RouterError-Typ ist folgendermaßen definiert:

Parameter Name	Typ	Beschreibung
errorMessage	String	Textuelle Beschreibung des Fehlers
errorCode	String	Enum, {NO_ROUTE_FOUND, INVALID_REQUEST, NOT_READY, UNKNOWN_ERROR}

Die folgenden Tabellen enthalten die jeweiligen (möglichen) Belegungen der Anfrage-resultate. Hierbei sind nur die verwendeten Attribute der Typen aufgezählt, alle weiter in URA3 spezifizierten Attribute werden nicht gesetzt. Das Top-Level Element "route" enthält in der minimalen Version ausschließlich die Kennzahlen der Route (Distanz, Dauer, Anzahl Umstiege) und in erweiterten Anfragen zusätzlich eine Liste von Routenabschnitten, welche jeweils einen Modus darstellen.

3.2 Compact

Parameter Name	Typ	Beschreibung
uuid	String	legId, interne ID (nicht geeignet für Folgeanfragen)
start	Prediction	Anfangsort des Abschnitts
end	Prediction	Zielort des Abschnitts
durationInS	Zahl	Dauer des Abschnitts
lengthInM	Zahl	Zurückgelegte Distanz innerhalb des Abschnitts
type	String	"IndividualTrip"
modalType	Verkehrsmittel	URA IndividualMode
pathDataSource	String	"xsharing-router", zur Identifikation

Der Typ Prediction ist folgendermaßen besetzt:

Parameter Name	Typ	Beschreibung
type	String	"Prediction"
location	Position/Station	Koordinaten des Ortes (lat/lon)
scheduledDepartureTime	DateTime	Abfahrtszeit von Ort
scheduledArrivalTime	DateTime	AnkunftsZeit am Ort

3.3 Details

Parameter Name	Typ	Beschreibung
uuid	String	legId, interne ID (nicht geeignet für Folgeanfragen)
start	Prediction	Anfangsort des Abschnitts (Location vom Typ Position/Station)
end	Prediction	Zielort des Abschnitts (Location vom Typ Station)
durationInS	Zahl	Dauer des Abschnitts
lengthInM	Zahl	Zurückgelegte Distanz innerhalb des Abschnitts
type	String	"IndividualTrip"
modalType	Verkehrsmittel	URA IndividualMode
pathDataSource	String	"xsharing-router", zur Identifikation
pathData	PathData	Konkrete Pfadinformationen (nur Feld "routeGeometry": Liste von lat/lon-Koordinaten)

Der Typ Station ist nun vorerst folgendermaßen besetzt:

Parameter Name	Typ	Beschreibung
type	String	"Station"
stationId	String	Globale ID der Station
provider	String	Betreiber der Station
totalSpaces	Integer	Gesamtanzahl Stellplätze in Station
availSpaces	Integer	Verfügbare Anzahl Stellplätze