Mobikat Webservices

Content

Part I

Introduction

1 Introduction

MobiKat Web Services

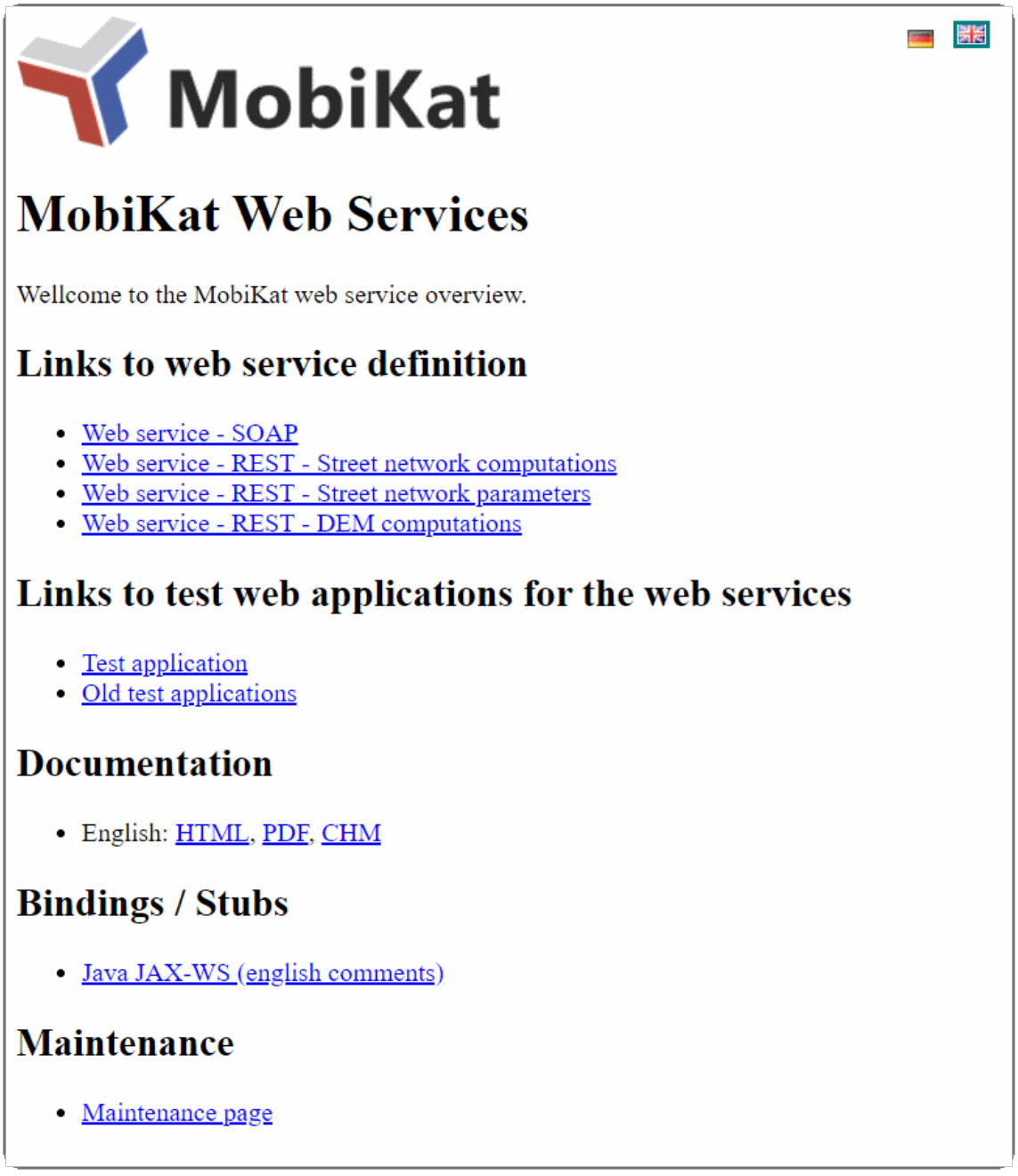
This document describes the structure and usage of the web services provided by MobiKat.

Last update: 2020-05-25

Structure

1.1 Web application

Welcome page



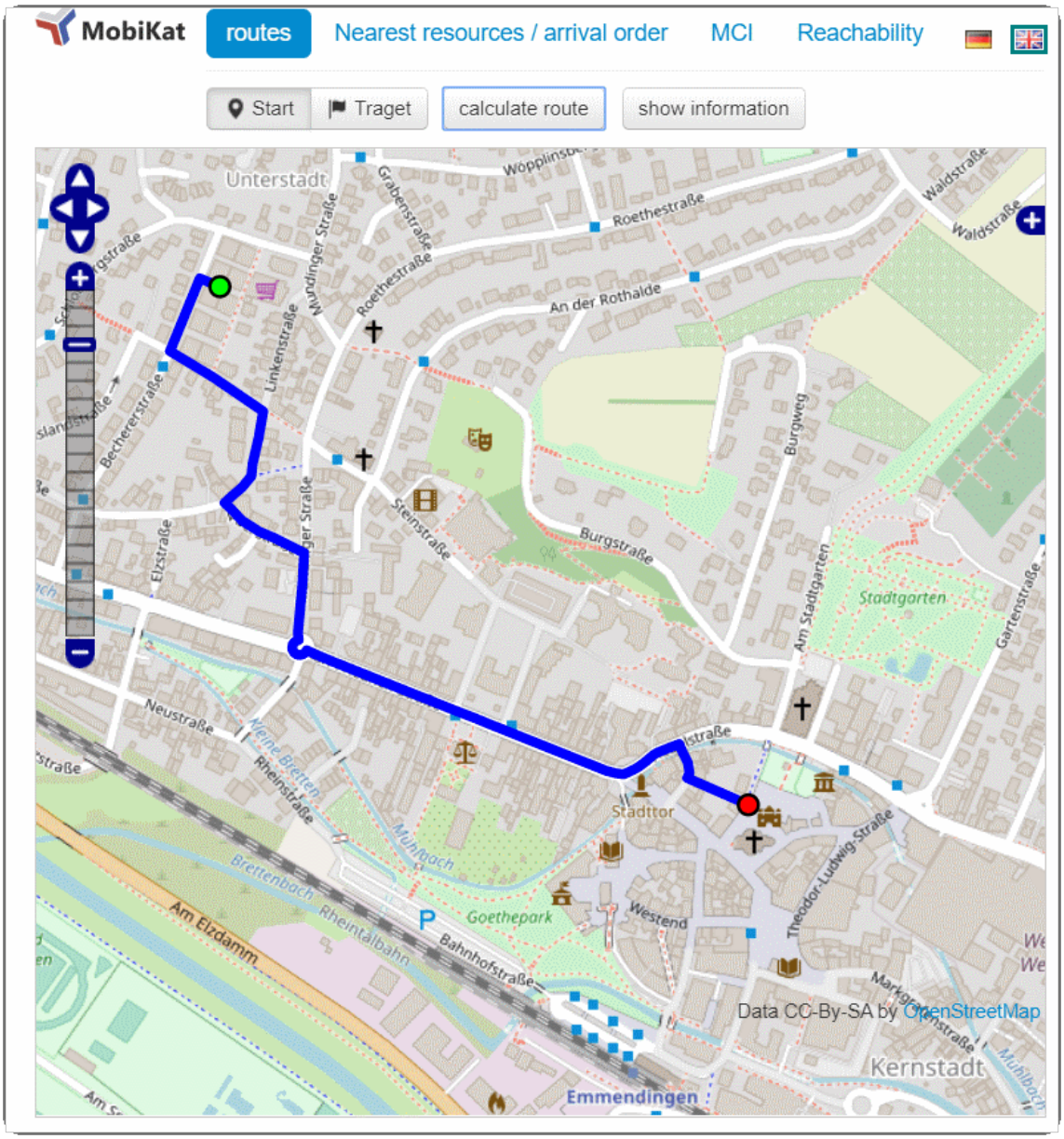
Index Page

On the welcome page several useful links could be found. See the following text

Service definition links

This section contains a list of links to the different supported protocols of the services. See also Protocols: SOAP & REST

Test Application



Test application

A web application is linked here, that allows some simple request to the web services.

Documentation

Links to this documentation in different formats.

Bindings / Stubs

Here you can find links to some packages with source code to access the web services.This allows a easy integration in your applications.

1.2 Protocols: SOAP & REST

The services can be accessed via SOAP or a more REST-like interface. But always the exchange format is XML. The object structure is described in the rest of the document.

SOAP

For SOAP two binding types are supported: *rpc* and *document*

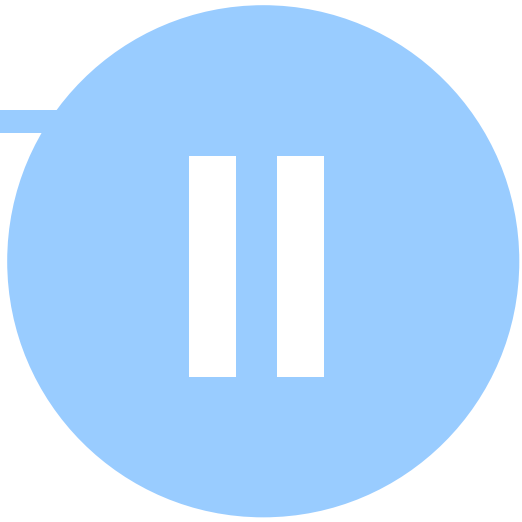
The *rpc* bindings don't have a special suffix (e.g. *MobikatStreetNetworkComputationsWS*) and the *document* bindings have the suffix "Doc" (e.g, *MobikatStreetNetworkComputationsWSDoc*)

REST

For each service port a separate sub-URL has been defined for the REST services. This sub-URL display some help information on a GET request.

The methods are again separate sub-URLs. They also display some help information on a GET request. To call a method, do a POST request to this method containing an approbate XML document.

Part



General Data Types

2 General Data Types

In this chapter the general data types of the MobiKat Web Services are described.

Request and Response data types

• RequestBase • ResponseBase

Other complex data types

• Position

Error data types

• MobikatServiceFault • MobikatServiceFaultCode

2.1 RequestBase

Base type for all requests.

Field name	Type	Default value	Comments
requestId	String	*generated*	Internal ID of this request. Must be unique! If not supplied an internal ID is generated.
user	String	*null*	User name. Can be null if no authentication is needed
password	String	*null*	User password. Can be null if no authentication is needed
coordSystem	String	EPSG:4326	Coordinate system ID of the coordinates in this request. E.g. "EPSG:4326"
outCoordSystem	String	EPSG:4326	Requested coordinate ID system in this response.E.g. "EPSG:4326"
outLanguage	String	en	Language for test outputs as ISO 639-1. (http://www.loc.gov/standards/iso639-2/php/English_list.php)
timeout	String	10000	Hint how long a computation should take at maximum in milliseconds. An error may be returned if a time-out occurs.
pointInTime	DateTime	*now*	Calculations that are based on time-dependent conditions (e.g. time-limited blocking of a road) use this time for the calculation. If not specified (null), the current time is used. Parameter is ignored if no time-dependent conditions are present.

Field name	Type	Default value	Comments
resultUrl	String	*null*	Optional URL where the result of the request is send (HTTP POST). Content will be an XML file containing a subtype of ResponseBase. This is an optional addition to the return value of the service call
statusChanged Url	String	*null*	Optional URL where status changes of the request are send (HTTP POST). Content will be an XML file containing a StatusResponse. This is an optional addition to the return value of the service call
errorUrl	String	*null*	Optional URL where errors occurring during the processing of the request are send (HTTP POST). Content will be an XML file containing a MobikatServiceFault. This is an optional addition to the return value of the service call

2.2 ResponseBase

Base type for all responses.

Field name	Type	Comments
requestId	String	Internal ID of the request
coordSystem	String	Coordinate system ID of the coordinates in this response. EPSG code, e.g. "EPSG:4326"
language	String	Language for text outputs as ISO 639-1 (if text is available in multiple languages) http://www.loc.gov/standards/iso639-2/php/English_list.php

2.3 MobikatServiceFault

Errors that are thrown by the service.

Field name	Type	Comments
requestId	String	Internal ID of the request
code	MobikatServiceFaultCode	Error code
description	String	Description of the error (optional)
cause	String	Additional technical descriptions of the error (optional)

2.4 MobikatServiceFaultCode

Error codes for different errors of the services.

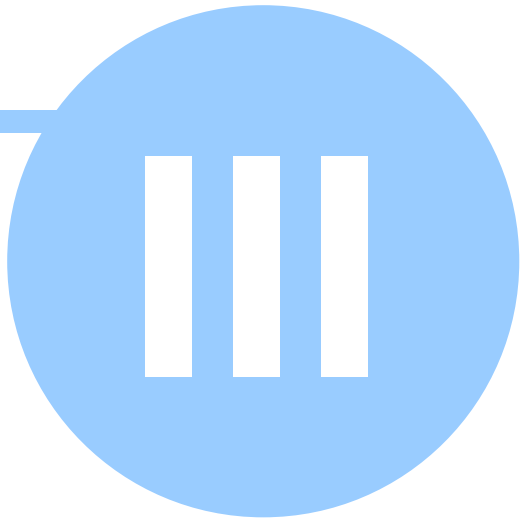
Enumeration value	Comments
Internal	Unknown internal error
AccessDenied	The given user does not hat the right to access the service
Timeout	Timeout error
OutOfMemory	Out of memory error: Request is to expensive!
IllegalArgument	Illegal argument: The arguments of a request are invalid!
NoResult	For the given parameters, the computation could not be finish with the given parameters. This is e.g. the case for a route computation with start and end point on two roads that are not connected.

2.5 Position

A 2D position, point or coordinate.

Field name	Type	Comments
x	Float	X-Value
y	Float	Y-Value

Part



Asynchronous calls

3 Asynchronous calls

It is possible to call services asynchronously.

Each request has a unique ID. Via this ID the request is identified during the asynchronous call handling. In RequestBase a custom ID could be set. If not, a new ID will be generated.

To do a asynchronous call, the methods with the "Async" suffix need to be called. They only deliver a StatusResponse with the ID of the request. The status can be requested by calling getStatus. The final result can be retrieved (when the computation has finished) by calling getResult. (Results are kept one hour before they are deleted).

Also notifications about the state and result of the computation are supported. They are sent via HTTP POST to a given URL. See *statusChangedUrl*, *resultUrl*, *errorUrl* in RequestBase (all three URL could be equal).

Functions

• getRequest • getResult • getStatus

Request and Response data types

• StatusRequest • StatusResponse • RequestDataRequest • ResultRequest

Enumeration data types

• ProcessingState

Error data types

• AsyncCallFault • AsyncCallFaultCode

3.1 getRequest

Returns the request data (parameter of the request) of the given request ID.

Input value	Output value
RequestDataRequest	subtype of RequestBase

3.2 getStatus

Returns the state of the request with the given request ID.

Input value	Output value
StatusRequest	StatusResponse

3.3 getResult

Returns the result of the request with the given request ID.

If there was an error during the computation, this error thrown "as result".

Throws an AsyncCallFault if the computation has not be finished!

Input value	Output value
ResultRequest	subtype of ResponseBase

3.4 StatusRequest

Ask for the status of a request (getStatus)

Field name	Type	Default value	Comments
requestId	String		Internal ID of the request

3.5 StatusResponse

Returns the status of a request (getStatus)

Field name	Type	Comments
requestId	String	Internal ID of the request
state	String	State of the request
info	String	Optional additional information Will contain the error if the request failed

3.6 RequestDataRequest

Ask for the parameter data of a request (getRequest)

Field name	Type	Default value	Comments
requestId	String		Internal ID of the request

3.7 ResultRequest

Ask for the result of a request (getResult)

Field name	Type	Default value	Comments
requestId	String		Internal ID of the request

3.8 AsyncCallFault

Errors that are thrown by asynchronous calls

Field name	Type	Comments
requestId	String	Internal ID of the request
code	AsyncCallFaultCode	Error code
description	String	Description of the error (optional)

3.9 AsyncCallFaultCode

Error codes for different errors of asynchronous calls.

Enumeration value	Comments
NotFound	No request with the given ID found (it may be to old)
NotFinished	The request with the given ID has not finished until now
AccessDenied	The given user does not hat the right to access the service

3.10 ProcessingState

Status of request.

Enumeration value	Comments
Waiting	Calculation has not yet started running (is still pending)
Running	Calculation is running, but has not completed
Finished	Calculation did successfully finish
NoResult	For the given parameters, the computation could not be finish with the given parameters. This is e.g. the case for a route computation with start and end point on two roads that are not connected.

Enumeration value	Comments
ParameterError	There was some problem with input or result request data. The web-service framework determined that one or more parameters were invalid.
RunError	The calculation was started, but failed before completion.
NotFound	The given request ID was not found.

Part IV

Computation Service

4 Computation Service

The Mobikat Webservice offers different functions for calculations on a street network.

Overview of the functions

Overview of the data types

4.1 Functions

The Mobikat Webservice offers following functions:

- Route computation (calculateRoute)
- Retrieval nearby resources (calculateNearestResources)
- Computation of the reachability (calculateReachability)
- Distributing injured at a MCI on hospitals(calculateMci)

4.1.1 calculateRoute

Computation of a route.

Calculate an optimal route from a starting point to a destination point, depending the current traffic network situation.

Input value	Output value
RouteRequest	RouteResponse

4.1.2 calculateNearestResources

Searches the nearest resources.

Calculation of optimal routes between a given point and the resource positions. Can be used for an operating resources proposal.

It is possible to compute the optimale routes from the point to the resources or from the resources to the point. (One-way streets and other non-symmetrical road parameters can causes different results for both variants.)

Input value	Output value
NearestResourcesRequest	NearestResourcesResponse

4.1.3 calculateReachability

Determines the reachable area/streets depending of predefined travel time or maximum distance.

Based on given station locations, it calculates the reachable area and streets of the given stations depending of predefined travel time or maximum distance.

This method can be applied in order to find some gaps in the coverage of an area.

Input value	Output value
ReachabilityRequest	ReachabilityResponse

4.1.4 calculateMci

In case of an Mass Casualty Incident (MCI), this method will return a proposal where to transport how many casualties depending of the capacities of the hospitals.

In case when many casualties need to be transported to hospitals, this functionally will help the responsible person to decide the distribution of the casualties over the hospitals.

Input value	Output value
MciRequest	MciResponse

4.2 Data types

In this chapter all data types of the Mobikat Webservices are described.

Request and Response data types

• RouteRequest • RouteResponse
• ReachabilityRequest • ReachabilityResponse
• MciRequest • MciResponse

Other complex data types

• AirVehicleResource • AirwaySystemParameters • HospitalResource • MciDispositionItem • Reachability • Resource • RoadwaySystemParameters • Route • RouteDescribtionElement • RouteOfResource • VehicleResource •

Enumeration data types

• RoutingOptimization • LegalVehicleClass • ObeyTurnRestrictions • Trafficability • RouteDescribtionElementType

4.2.1 RouteRequest

Parameters of a route computation (calculateRoute).

Inherits from RequestBase.

Field name	Type	Default value	Comments
start	Position		Starting point
target	Position		Target point

Field name	Type	Default value	Comments
optimization	RoutingOptimization	Time	Optimization (minimale time or way), used for routing
roadwaySystemParameters	RoadwaySystemParameters	*default values*	Vehicle and street parameters (can be null)
includeRouteCoordinates	Boolean	false	Include the route coordinates (line string) in response.
includeRouteDescription	Boolean	false	Include the route description (verbal description of the route) in response.
includeStructuredRouteDescription	Boolean	false	Include route description (structured) in response
includeStreetSectionsIds	Boolean	false	Include the street sections ID of the computed route in response

4.2.2 RouteResponse

Results of a route computation (calculateRoute).

Inherits from ResponseBase.

Field name	Type	Comments
route	Route	Computed route

4.2.3 NearestResourcesRequest

Parameters of a nearest resource computation (calculateNearestResources).

Inherits from RequestBase.

Field name	Type	Default value	Comments
position	Position		Position, used for distance computation to the resources
resources	List of Resource	*null*	Additional resources, used for computations. If the resources have a special speed profile, use VehicleResource.
useResourcesFromDB	Boolean	true	True, if the resources from the database should be loaded and used. Only available if database resources are defined.
resourceTypes	Set of String	*null*	Filter: String representations of the searched resources. To include unknown types add an empty string. Empty list for all.
resourceStatuses	Set of String	*null*	Filter: String representations of the statuses of the searched resources. To include unknown statuses add an empty string. Empty list for all.

Field name	Type	Default value	Comments
optimization	RoutingOptimization	Time	Optimization (minimal time or way), used for routing
inverseRouting	Boolean	true	Routing direction: • false - from position to resources • true - from resources to position (One-way streets and other non-symmetrical street parameters causes different results for the two variants.)
maxTravelTime	Double	0	Maximum travel time to resource (in seconds). 0 for unlimited.
maxDistance	Double	0	Maximum distance to resource (in meters). 0 for unlimited.
maxCount	Integer	0	Maximum count of returned resources. 0 for unlimited.
defaultRoadwaySystemParameters	RoadwaySystemParameters	*default values*	Vehicle and street parameters (VehicleResource used, if no special parameters are given for the resource.)
useStraightLineDistance	Boolean	false	No routing, using the straight-line distance. Parameters inverseRouting, roadwaySystemParameters, optimization, includeRouteCoordinates and maxTravelTime are ignored.
includeRouteCoordinates	Boolean	true	Include route coordinates in response
includeRouteDescription	Boolean	false	Include route description (verbal description of the route) in response
includeStructuredRouteDescription	Boolean	false	Include route description (structured) in response
includeStreetSectionsIds	Boolean	false	Include the street sections ID of the computed route in response

4.2.4 NearestResourcesResponse

Results of a nearest resources computation (calculateNearestResources).

Inherits from ResponseBase.

Field name	Type	Comments
resourceRoutes	List of RouteOfResource	List of a nearest resources computation. Sorted by distance / travel time (depending on optimization).

4.2.5 ReachabilityRequest

Parameters of a reachability computation (calculateReachability).

Inherits from RequestBase.

Field name	Type	Default value	Comments
resources	Resource		Resources with their coordinates used for computations. If the resources have a special speed profile, use VehicleResource. Also supports AirVehicleResource, to compute a beeline reachability (e.g. of a helicopter).
limit	Double		Max. distance (in meters) or travel time (in seconds) (depending on optimization), used for reachability.
optimization	RoutingOptimization	Time	Optimization (minimal time or way), used for routing
defaultRoadwaySystemParameters	RoadwaySystemParameters	*default values*	Vehicle and street parameters (Ignored for resources of type VehicleResource.with valid *roadwaySystemParameters* field)
defaultAirwaySystemParameters	AirwaySystemParameters	*default values*	Air way speed parameters for AirVehicleResources without a valid air*waySystemParameters* field.
isocroneAreaResolution	Integer	10000	Resolution of the reached area raster in raster cells. 0 for no area computations.
interspaceVelocitiy	Double	2.0	Velocity that is assume between streets when calculating the reached area
includeReachedRoadsCoordinates	Boolean	true	Include coordinates of the reached roads in response.
investigationAreaWkt	String	*null*	Area (polygon or multi-polygon) of the study area as a WKT (Well Known Text). When this area is passed, an inverse reachability is calculated: the unreachable areas and road sections within the given area. If not specified (null), the normal availability is calculated.

4.2.6 ReachabilityResponse

Results of a reachability computation (calculateReachability).

Inherits from ResponseBase.

Field name	Type	Comments
reachability	Reachability	Computed reachability

4.2.7 MciRequest

Parameters of an MCI (Mass Casualty Incident) computation (calculateMci).

Inherits from RequestBase.

Field name	Type	Default value	Comments
position	Position		Starting point
hospitals	List of HospitalResource	*null*	Additional hospitals, used for computations
useHospitalsFromDB	Boolean	true	Truth, if the hospitals from the database should be loaded and used. Only available if database resources are defined.
resourceTypes	Set of String	*null*	Filter: String representations of the searched hospitals. To include an unknown type add an empty string. Empty list for all.
resourceStatuses	Set of String	*null*	Filter: String representations of the statuses of the searched hospitals. To include an unknown status add an empty string. Empty list for all.
t1	Integer	0	Casualties in category 1
t2	Integer	0	Casualties in category 2
t3	Integer	0	Casualties in category 3
limit	Double	*null*	Maximum distance or travel time (depending on optimization) to an hospital. null or 0 for unlimited.
optimization	RoutingOptimization	Time	Optimization (minimal time or way), used for routing
roadwaySystemParameters	RoadwaySystemParameters	*default values*	Vehicle and street parameters
includeRouteCoordinates	Boolean	true	Include route coordinates (line string) in response.
includeRouteDescription	Boolean	false	Include route description (verbal description of the route) in response.
includeStructuredRouteDescription	Boolean	false	Include route description (structured) in response
includeStreetSectionsIds	Boolean	false	Include the street sections ID of the computed route in response

4.2.8 MciResponse

Parameters of an MCI (Mass Casualty Incident) computation (calculateMci).

Inherits from ResponeBase.

Field name	Type	Comments
items	List of MciDispositionItem	Allocation of casualties to hospitals.

4.2.9 RoadwaySystemParameters

Velocity (in km/h), vehicle and street options.

Field name	Type	Default value	Comments
motorwayVelocitiy	Double	100	Velocity on motorways
roadAVelocitiyInTown	Double	45	Velocity on A class roads inside towns
roadAVelocitiyOutOfTown	Double	60	Velocity on A class roads outside towns
roadBVelocitiyInTown	Double	40	Velocity on B class roads inside towns
roadBVelocitiyOutOfTown	Double	60	Velocity on B class roads outside towns
roadCVelocitiyInTown	Double	40	Velocity on C class roads inside towns
roadCVelocitiyOutOfTown	Double	60	Velocity on C class roads outside towns
roadDVelocitiyInTown	Double	30	Velocity on D class roads inside towns
roadDVelocitiyOutOfTown	Double	60	Velocity on D class roads outside towns
roadEVelocitiyInTown	Double	10	Velocity on E class roads inside towns
roadEVelocitiyOutOfTown	Double	10	Velocity on E class roads outside towns
perpendicularVelocitiy	Double	5	Velocity on to the perpendicular to nearest road. Zero for ignoring the perpendicular in time calculations (infinite velocity)
~~trafficability~~	~~Trafficability~~	~~null~~	DEPRICATED ~~Trafficability of the vehicle~~. ~~Only roads with this trafficability or better~~ are ~~used.~~ ~~Only applies if trafficability values of the~~ roads ~~are given~~.

Field name	Type	Default value	Comments
legalVehicleClass	LegalVehicleClass	*null*	Legal class of the vehicle. Null to ignore the property in the computation. Roads that are prohibited for this class are avoided.
height	Integer	0	Height of the vehicle in [cm]. 0 to ignore the property in the computation.
width	Integer	0	Width of the vehicle in [cm] 0 to ignore the property in the computation.
length	Integer	0	Length of the vehicle in [cm] 0 to ignore the property in the computation.
weight	Integer	0	Weight of the vehicle in [kg] 0 to ignore the property in the computation.
weightPerAxle	Integer	0	Weight per axle of the vehicle in [kg] 0 to ignore the property in the computation.
ignoreUnpavedRoads	Boolean	false	Don't use unpaved roads
ignoreFourWheelDriveRoads	Boolean	false	Don't use four wheel drive (4WD, 4x4) roads
ignoreOneWayRoads	Boolean	false	Ignore one-way streets!
ignoreMotorwayDirection	Boolean	false	Ignore motorway directions!
ignoreRoadsE	Boolean	true	Do not use roads of class E!
ignorePedestrianAreas	Boolean	false	Ignore pedestrian areas (also use "no drive through" roads)
ignoreSpeedLimits	Boolean	true	Ignore legal speed limits
obeyTurnRestrictions	ObeyTurnRestrictions	OnlyPhysical	Obey turn bans / turn restrictions.
vehicleType	String	*null*	Vehicle type for special street parameters. Used for the statistical velocity parameters if available. Only if supported by the current system and database (optional).

4.2.10 AirwaySystemParameters

Velocity (in km/h), vehicle and other options.

Field name	Type	Default value	Comments
beelineVelocity	Double	200	Velocity when moving on direct beeline (in km/h).

Field name	Type	Default value	Comments
timeBeforeStart	Integer	0	Amount of time (in seconds) that is needed to make a vehicle ready to start.
timeAfterArrive	Integer	0	Amount of time (in seconds) that is needed after the vehicle has reached the target point, but before anything else can be done.

4.2.11 Route

Computed route from starting point to the target point (calculateRoute, calculateNearestResources, calculateMci).

Field name	Type	Comments
duration	Float	Duration / travel time of the route in seconds.
length	Float	Length / distance of the route in meters.
routeCoordinatesWkt	String	Coordinates of the points of the route as Well Known Text (WKT). (optional)
descriptionList	List of String	List of travel instructions for the route (optional).
structuredDescriptionList	List of RouteDescribtionElement	List of travel instructions for the route in a structured way (optional)
streetSectionId	List of Long	IDs of the street sections of this route (optional)

4.2.12 RouteDescribtionElement

A manoeuvre on the Route

Field name	Type	Comments
type	RouteDescribtionElementType	Type of the manoeuvre on the route
counter	Integer	If there are several turns in nearly the same direction at nearly the same position, this number says which particular (order from the nearest) turn should be taken. In a roundabout this is the number of the exit (base index is 1) CURRENTLY NOT IMPLEMENTED!
sourceStreetId	Integer	Street ID at the beginning of the manoeuvre
destinationStreetId	Integer	Street ID at the end of the manoeuvre
sourceStreetName	String	Street name at the beginning of the manoeuvre
destinationStreetName	String	Street name at the end of the manoeuvre

length	Float	Length in meters from the start of the route to this manoeuvre
duration	Float	Time in seconds from the start to this manoeuvre

4.2.13 RouteDescribtionElementType

Type of the manoeuvre on the Route

Enumeration value	Comments
Roundabout	Drive throw roundabout (currently not implemented)
StraightOn	straight on, on change of road type / name
TurnLeftSlightly	left turn (slightly)
TurnLeft	left turn (normal)
TurnLeftSharp	left turn (sharp)
TurnRightSlightly	right turn (slightly)
TurnRight	right turn (normal)
TurnRightSharp	right turn (sharp)

4.2.14 Resource

A resource. Needed for different computations.

Field name	Type	Comments
id	String	ID of the resource
name	String	Name of the resource (optional)
type	String	Type of the resource (optional)
status	String	Status of the resource (optional)
position	Position	Coordinates of the resource

4.2.15 VehicleResource

A vehicle represented as resource with its own vehicle and street parameters.

Inherits from Resource.

Field name	Type	Comments
roadwaySystemParameters	RoadwaySystemParameters	Vehicle and street parameters. If not specified (null), the default parameters are used.

4.2.16 AirVehicleResource

An air vehicle represented as resource with its own air way speed parameters.

Inherits from Resource.

Field name	Type	Comments
airwaySystemParameters	AirwaySystemParameters	Air way speed parameters. If not specified (null), the default parameters are used.

4.2.17 RouteOfResource

Route to / from a resource (calculateNearestResources).

Inherits from Resource.

Field name	Type	Comments
resource	Resource	The resource
route	Route	The route

4.2.18 HospitalResource

Resource representing as a hospital (calculateMci).

Inherits from Resource.

Field name	Type	Comments
t1	Integer	Count of current available beds in category 1
t2	Integer	Count of current available beds in category 2
t3	Integer	Count of current available beds in category 3

4.2.19 Reachability

Reached routes and area (calculateReachability).

Field name	Type	Comments
reachedRoadsWkt	String	Coordinates of the reached roads as WKT (well known text). Can be null!
isochroneWkt	String	Coordinates of the reached area as WKT (well known text). Can be null!

4.2.20 MciDispositionItem

Contain the count assigned patients to a hospital and the route to the hospital (calculateMci).

Field name	Type	Comments
hospital	HospitalResource	Hospital that is used
t1	Integer	Count of assigned casualties in category 1
t2	Integer	Count of assigned casualties in category 2
t3	Integer	Count of assigned casualties in category 3
route	Route	Route to the hospital

4.2.21 RoutingOptimization

Optimization criteria for the routing inside the street network.

Enumeration value	Comments
Time	Route with shortest time (in seconds)
Way	Route with shortest way (in meters)

4.2.22 Trafficability

Minimum vehicle type needed to pass a road. (The further above in the table, the better!). Used in RoadwaySystemParameters

Enumeration value	Comments
Off-Road Vehicle	Cross-country vehicle
Car	Car
Transporter	Van
Truck	Heavy goods vehicle
Trailer Truck	Trailer Truck
Heavy Duty Truck	Heavy Duty Truck

4.2.23 ObeyTurnRestrictions

Way of obeying turn restrictions. Used in RoadwaySystemParameters

Enumeration value	Comments
All	Obey all turn restrictions of the given vehicle class
OnlyPhysical	Obey only physical (like barriers or dividers) turn restrictions of the given vehicle class
No	Obey no turn restrictions

4.2.24 LegalVehicleClass

The legal (by law) class of a vehicle. Used in RoadwaySystemParameters

Enumeration value	Comments
Pedestrian	Pedestrian
Bicycle	Bicycle
Motorcycle	Motorcycle
Car	Car
Taxi	Taxi
Bus	Bus
Truck	Truck
EmergencyVehicle	Emergency Vehicle

Part V

Parameter Service

5 Parameter Service

For the street network computation the Webservice "MobikatStreetnetworkParameterEditing" is made.

Overview of the functions

Overview of the data types

5.1 Functions

The Mobikat Webservice for street network computation offers following functions:

- Specify and edit a traffic situation (defineTrafficSituation)
- Activation/deactivation of a traffic situation (enableTrafficSituation)
- Delete a traffic situation (deleteTrafficSituation)
- Enumerate existing traffic situations (listTrafficSituation)
- Determine street section in an area (streetSectionInArea)
- Determine street section via ID (listStreetSections)

5.1.1 defineTrafficSituation

Definition of a traffic situation.

This functionally can be used for creating or editing/updating of traffic situations.

Input value	Output value
DefineTrafficSituationRequest	

5.1.2 enableTrafficSituation

Activation or deactivation of traffic situations.

One or more traffic situations can be activate or deactivate by this functionally.

When a traffic situation is disabled, it is not considered in calculations.

Input value	Output value
EnableTrafficSituationRequest	

5.1.3 deleteTrafficSituation

Deletion of traffic situations.

One or more traffic situations can be deleted by this functionally.

Input value	Output value
DeleteTrafficSituationRequest	

5.1.4 listTrafficSituation

Enumeration of all existing traffic situations.

All founded traffic situations are listed by the system.

Input value	Output value
ListTrafficSituationsRequest	ListTrafficSituationsResponse

5.1.5 streetSectionsInArea

Determine street section in an area.

The street sections can be determined by the intersection with the area.

Input value	Output value
StreetSectionsInAreaRequest	StreetSectionsInAreaResponse

5.1.6 listStreetSections

Returns all street sections within the given IDs

Input value	Output value
ListStreetSectionsRequest	ListStreetSectionsResponse

5.2 Data types

In this chapter all data types of the Mobikat Webservices for street network computation are described.

Request and Response data types

- DefineTrafficSituationRequest
- EnableTrafficSituationRequest
- deleteTrafficSituation
- ListTrafficSituationRequest • ListTrafficSituationResponse
- StreetSectionsInAreaRequest • StreetSectionsInAreaResponse
- ListStreetSectionsRequest • ListStreetSectionsResponse

Other complex data types

• TrafficSituationFilter • TrafficSituation • TimePeriode • Limitation • StreetSectionPart • StreetSection

Enumeration data types

• LimitationType

5.2.1 DefineTrafficSituationRequest

Parameters for a "Define Traffic Situation" action (defineTrafficSituation).

Inherits from RequestBase.

Field name	Type	Default value	Comments
trafficSituation	TrafficSituation		Traffic situation, which is to be created or updated.
overwrite	Boolean	true	Applies only when a traffic situation with the same ID exists: If true, it will be overwritten / updated. If false, an error is thrown.
checkLastUpdate	Boolean	false	Check the date of the last update of the traffic situation (detect parallel editing). Applies only when a traffic situation with the same ID exists and overwrite is true: If true and if "last update" of "trafficSituation" differs from the value in the system, an exception is thrown.

5.2.2 EnableTrafficSituationRequest

Parameters for an "Enable Traffic Situation" action (enableTrafficSituation).

Inherits from RequestBase.

Field name	Type	Default value	Comments
trafficSituationIDs	List of String		IDs of the traffic situation, which should be activated or deactivated.
disabled	Boolean		Should the traffic situations be activated (false) or deactivated (true)?

5.2.3 DeleteTrafficSituationRequest

Parameters for a "Delete Traffic Situation" action. (deleteTrafficSituation)

Inherits from RequestBase.

Field name	Type	Default value	Comments
trafficSituationIDs	List of String		IDs of the traffic situation to be deleted.

5.2.4 ListTrafficSituationsRequest

Parameters for a "List Traffic Situation" action (listTrafficSituation).

Inherits from RequestBase.

Field name	Type	Default value	Comments
filter	TrafficSituationFilter		Filter to list only certain traffic situations.
includeLimitations	Boolean	false	Should the limitation of the traffic situations be returned? (Reduced response time if set to false.)
includeValidityPeriods	Boolean	faöse	Should the validity periods of the traffic situations be returned? (Reduced response time if set to false.)

5.2.5 ListTrafficSituationsResponse

Results of a "List Traffic Situation" action (listTrafficSituation).

Inherits from ResponseBase.

Field name	Type	Comments
trafficSituations	List of TrafficSituation	List of the founded traffic situations

5.2.6 StreetSectionsInAreaRequest

Parameters for a "Street Section in Area" computation (streetSectionsInArea).

Inherits from RequestBase.

Field name	Type	Default value	Comments
area	String		Coordinates of the area (polygon or multi-polygon) for the intersection as a WKT (Well Known Text).

Field name	Type	Default value	Comments
fullyInside	Boolean	true	If true, only street sections are returned that are completely within the given area. If false, all street sections are returned lying wholly or partly in the area.
includeCoordinates	Boolean	true	If true, the coordinates of the street sections are included in the response. If false, the coordinate are left out in the response (less size).

5.2.7 StreetSectionsInAreaResponse

Results of a "Street Sections in Area" computation (streetSectionsInArea).

Inherits from ResponseBase.

Field name	Type	Comments
streetSections	List of StreetSection	List of the found street sections

5.2.8 ListStreetSectionsResponse

Parameters for a "Street Section in Area" computation (listStreetSections).

Inherits from RequestBase.

Field name	Type	Default value	Comments
ids	String		IDs of the street sections to be loaded
includeCoordinates	Boolean	true	If true, the coordinates of the street sections are included in the response. If false, the coordinate are left out in the response (less size).

5.2.9 ListStreetSectionsRequest

Result of a "list street sections" request (listStreetSections)

Inherits from ResponseBase.

Field name	Type	Comments
streetSections	List of StreetSection	List of the found street sections

5.2.10 TrafficSituationFilter

Filter for selected traffic situations.

Field name	Type	Default value	Comments
trafficSituationIDs	List of String	*null*	Only traffic situations that have one of these ID´s. Null / empty list for all.
categories	List of String	*null*	Only traffic situations that have one of these categories. Null / empty list for all.
disabled	Boolean	*null*	Only traffic situations that have activation status. Null / empty list for all.
valid	Boolean	*null*	Only traffic situations that have this validity status. (Valid means that a least one validity period is active. As time parameter "pointInTime" from "RequestBase" is used.) Null for all.

5.2.11 TrafficSituation

Representation of the data of a traffic situation, which can be used for street network calculations.

Field name	Type	Comments
id	String	Unique ID or name of the traffic situation
comment	String	any comment (optional)
categories	List of String	Categories that are assigned to this traffic situation. Example: "flood" for traffic situations that represent certain flood situations.
disabled	Boolean	Whether this traffic situation is currently turned off or disabled. When disabled, it is not considered in calculations.
validityPeriods	List of TimePeriod	Validity periods for this traffic situation. Outside of this periods, the traffic situation is not considered in calculations. If no periods are specified (null or empty list), the traffic situation is always valid.
limitations	List of Limitation	List of associated restrictions on street sections. When a street section has several limitations, the limitation with the lowest resulting velocity applies.
lastUpdate	DateTime	Time of the last update or creation time. Will be used to recognised the parallel edits.

Field name	Type	Comments
		Is set by the system and should not be changed by the user. Set to zero if the traffic situation should be created.

5.2.12 TimePeriode

Represents a time period.

Field name	Type	Comments
name	String	Name (optional)
comment	String	any comment (optional)
dateFrom	DateTime	Begin of the period (null for unlimited)
dateTo	DateTime	End of the period (null for unlimited)

5.2.13 Limitation

Describes a street section that is between two nodes in the road network.

Field name	Type	Comments
id	String	Name (unique inside the traffic situation)
comment	String	any comment (optional)
type	LimitationTyps	Type of the limitation
value	Double	Value of the limitation
streetSectionParts	List of StreetSectionPart	List of street sections for which this limitation is applies.

5.2.14 LimitationType

List of available restriction types for the street.

Enumeration value	Comments
closed	The street is closed / blocked.
velocityFactor	Default velocity is multiplied with this factor. Value of less than or equal 0 blocks the street sections. Value 1 will have no effect. Value higher than 1 will increase the velocity.
maxVelocity	Maximum used speed in km/h. A value equal to 0 blocks the street sections.
velocity	Fixed used speed in km/h. A value equal to 0 blocks the street sections.
minTime	Time in seconds, which is at least required to pass through the street sections. (It corresponds to a maximum speed, depending on route length.)

Enumeration value	Comments
time	Fixed time in seconds, which is required to pass through the street sections. (It corresponds to a fixed speed, depending on route length.)
timePenalty	Time penalty in seconds, that is added when traveling the street section.

5.2.15 StreetSectionPart

Definition of a part of a street section.

A restriction applying to a certain part of the street section is possible by specifying factors for the start and end of the part (factor 0 is the beginning of the directional street section, 1 for the end, 0.5 for the middle). Limiting the direction of travel is also possible (forward or reverse direction).

Field name	Type	Comments
streetSectionId	Long	ID of the associated street section
start	Double	Start factor of used part
end	Double	End factor of used part
direction1	Boolean	Should it apply to the forward direction (along the direction of the directional street section)?
direction2	Boolean	Should it apply to the reverse direction (against the direction of the directional street section)?

5.2.16 StreetSection

Describes a street section that is between two nodes in the road network.

Field name	Type	Comments
id	Long	ID of this street section
name	String	Street name (may be null if not available)
coordinates	String	Line coordinates (LineString) of the section as a WKT (Well Known Text). The defined forward direction of the section corresponds to the direction / orientation of the polyline. (Null if coordinates where not requested.)
direction1	Boolean	Is the forward direction (along the direction of the directional street section) passable?
direction2	Boolean	Is the reverse direction (against to the direction of the directional street section) passable?

Index

- O -

- P -

- R -

- S -

- T -

- U -

- V -

- W -