265592

Aptean Ltd

Contents

1 265592

2 265592 BS-7SGKZF NAVTEQ Map

3 Functional Overview

3.1 Client Requirement

Map Generation and Print via Navteq

Required generation of map and printout with Trip visualization for Security procedures in Tobacco business. Sample printouts currently used in country attached (Appendix A). Mandatory for Tobacco goods transport

3.2 Solution

OBS will add a new button to the Trip Manipulation and Trip Planning Screens labelled ?Show Route? this button will be available based on a user access function i.e. only users with this promoted to their user group will be able to see the button.

When a user clicks on the new button a new browser window branded in DHL colours will be opened and the standard Navteq interactive map applet will be embedded and displayed. The Navteq applet will be configured to display a highlighted route for the trip selected in the MTS Trip screens. Including flags at start, end and via destination points. The user will then be able to use the standard applet functionality to zoom and print all or partial map areas.

Driving instructions can be included if required at no additional cost to this estimate.

- Due to space limitations on the trip forms then the new map functionality will be made available from the right click (from a trip) drop down menu.

3.3 Scope

This change will be applied to system version 10.6.

4 Functional Description

4.1 Trip Manipulation

The current Trip Manipulation form looks like :-

The right click drop down menu option for the trip will be changed to add the new ?Show Map? option.

4.2 Trip Planning

The current Trip Planning form looks like :-

The right click drop down menu option for the trip will be changed to add the new ?Show Map? option.

4.3 FORM and HTML

The code behind the new Show Map option will use the WebShowDocument command to call a new HTML page passing in the appropriate parameters.

This page will use Javascript commands (using the AJAX API provided by NAVTEQ) to produce the appropriate map.

4.3.1 Form changes

The new Show Map option in both forms will firstly need to use the standard code to check the function parameter ?GEO_SHOW_NAVTEQ_MAP? to ensure that the user belongs to a group that has this new function promoted.

Once this is confirmed it will then build a string to be used by the WebShowDocument command to call the new HTML page which in turn will produce the map.

The string will firstly contain the location and name of the HTML page for displaying the NAVTEQ map which will be extracted from the new system registry ?GEO_NAVTEQ_URL_MAP?.

It will then add all of the standard parameters to the string that are required by NAVTEQ to ensure that the route calculation for the map matches the original route calculation produced by the SOAP request to NAVTEQ when the trip was first planned.

These will include :-

GEO_NAVTEQ_MOTOR_SPEED GEO_NAVTEQ_COUNTRY_SPEED GEO_NAVTEQ_CITY_SPEED GEO_NAVTEQ_FAST_SHORT (Calculation Mode)

Finally it will need to pass the location information. The Start, Destination and up to 4 via locations can be provided. Each location will be provided as a longitude, latitude and description.

4.3.2 New HTML

The HTML will be written using Javascript to call NAVTEQ to calculate the route using the provided parameters and to display it on a Map with all of the locations marked and named (See example of expected layout in Appendix D).

NB) It will hard code the vehicle type of ?Truck? to match the original soap request.

Once the map is displayed then the user can navigate around the map (zoom in/out etc.) using the standard NAVTEQ commands.

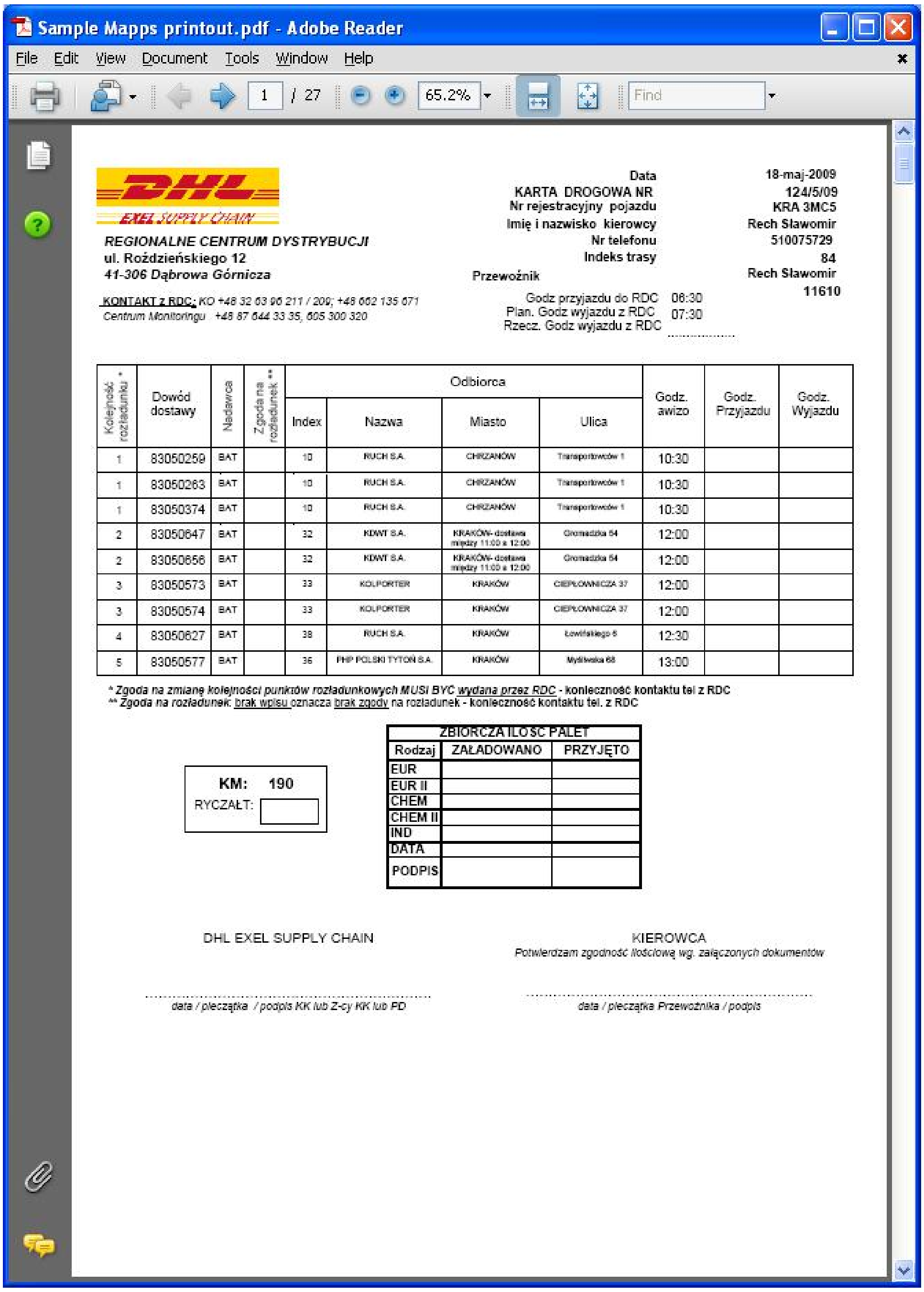
This will enable the user to get the level of detail they require before printing the map.

It will use the standard DHL Style sheet to format the output (See appendix C).

See appendix B for full list of parameters that can be passed to MAP24 Route

4.4 Appendix A

4.4.1 Sample Instructions


Sample Mapps printout.pdf - Adobe Reader
File Edit View Document Tools Window Help
1 / 27
65.2%
Find

REGIONALNE CENTRUM DYSTRYBUCJI
ul. Rożdzieńskiego 12
41-306 Dąbrowa Górnicza

KONTAKT z RDC: *KO +48 32 63 96 211 / 209; +48 662 135 671*
Centrum Monitoringu +48 87 644 33 35, 605 300 320

Data	**18-maj-2009**
KARTA DROGOWA NR	**124/5/09**
Nr rejestracyjny pojazdu	**KRA 3MC5**
Imię i nazwisko kierowcy	**Rech Sławomir**
Nr telefonu	**510075729**
Indeks trasy	**84**
Przewoźnik	**Rech Sławomir**
	11610
Godz przyjazdu do RDC	06:30
Plan. Godz wyjazdu z RDC	07:30
Rzecz. Godz wyjazdu z RDC	....

Kolejność rozładunku *	Dowód dostawy	Nadawca	Zgoda na rozładunek **	Odbiorca				Godz. awizo	Godz. Przyjazdu	Godz. Wyjazdu
				Index	Nazwa	Miasto	Ulica			
1	83050259	BAT		10	RUCH S.A.	CHRZANÓW	Transportowców 1	10:30		
1	83050263	BAT		10	RUCH S.A.	CHRZANÓW	Transportowców 1	10:30		
1	83050374	BAT		10	RUCH S.A.	CHRZANÓW	Transportowców 1	10:30		
2	83050647	BAT		32	KDWT S.A.	KRAKÓW- dostawa między 11:00 a 12:00	Gromadzka 54	12:00		
2	83050656	BAT		32	KDWT S.A.	KRAKÓW- dostawa między 11:00 a 12:00	Gromadzka 54	12:00		
3	83050573	BAT		33	KOLPORTER	KRAKÓW	CIEPŁOWNICZA 37	12:00		
3	83050574	BAT		33	KOLPORTER	KRAKÓW	CIEPŁOWNICZA 37	12:00		
4	83050627	BAT		38	RUCH S.A.	KRAKÓW	Łowińskiego 5	12:30		
5	83050577	BAT		36	PHP POLSKI TYTOŃ S.A.	KRAKÓW	Mydlnicka 68	13:00		

* *Zgoda na zmianę kolejności punktów rozładunkowych MUSI BYĆ wydana przez RDC* - konieczność kontaktu tel z RDC
** *Zgoda na rozładunek:* brak wpisu oznacza brak zgody na rozładunek - konieczność kontaktu tel. z RDC

KM: 190
RYCZAŁT: □

ZBIORCZA ILOŚĆ PALET		
Rodzaj	ZAŁADOWANO	PRZYJĘTO
EUR		
EUR II		
CHEM		
CHEM II		
IND		
DATA		
PODPIS		

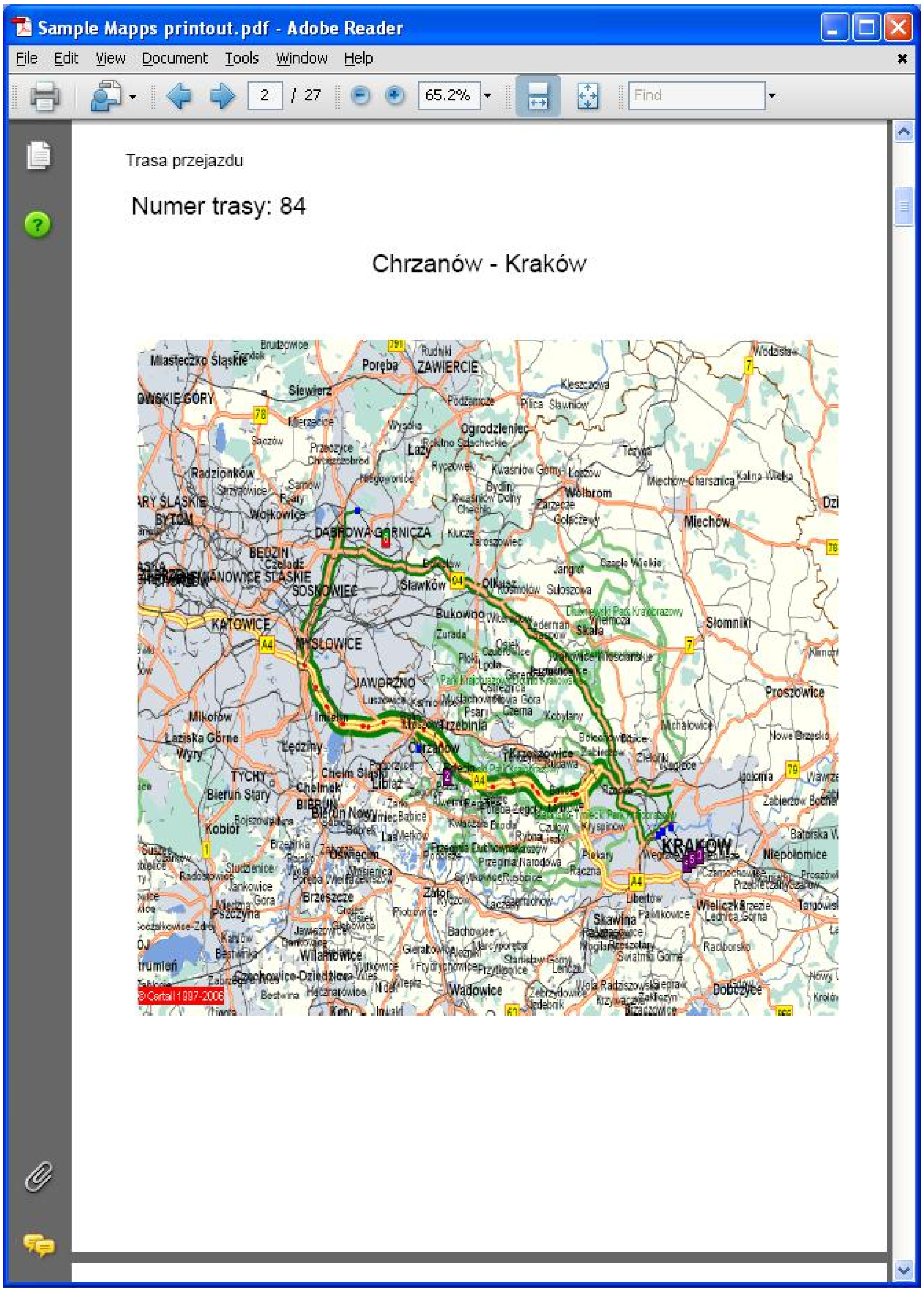
DHL EXEL SUPPLY CHAIN

....
data / pieczątka / podpis KK lub Z-cy KK lub PD

KIEROWCA
Potwierdzam zgodność ilościową wg. załączonych dokumentów

....
data / pieczątka Przewoźnika / podpis

4.4.2 Sample Map



Trasa przejazdu

Numer trasy: 84

Chrzanów - Kraków

4.5 Appendix B

4.5.1 LASS MAP24.ROUTINGSERVICESTUB

4.5.1.1 Object

```
Map24.RoutingServiceStub
```

This class provides methods for calculating and displaying a route. The calculateRoute function sends a request to the routing service to calculate a route between two locations and, optionally, via additional stops. The generated route information consists of a graphical representation (polyline) of the route and a turn by turn detailed route description. The mapping client can render the geographical representation of the route together with the map data so that the route is displayed on the map. The Map24 Routing Service is flexible in parameterization. The application developer can choose a preconfigured set of parameters for speed and other driver preferences. In addition, a sophisticated lower level access to the input parameters is provided that allows to fine tune the routing algorithm and to add application specific routing behavior. The routing attributes that are considered by the routing service depend in some respect on the underlying digital map content. The following routing attributes are usually supported from all digital map providers and are therefore considered by the routing service: functional road class, form of way, route numbers, direction of traffic flow (one-way, two-way, divided highway), road condition, network classifications, blocked passages, special restrictions, restricted maneuvers, toll roads, etc.

Copyright: © NAVTEQ 2009 Defined in RoutingServiceStub.js

4.5.1.2 Constructor Summary

```
Map24.RoutingServiceStub ( <Hashtable> _params )
```

Creates a new Map24.RoutingServiceStub object.

4.5.1.3 Method Summary

Type	Method	Summary
<void>	calculateRoute (<Hashtable> _params)	Calculates a route using given Map24.Coordinate objects or Map24.Location objects as start and destination points. ...
<void>	hideRoute (<Hashtable> _params)	Hides the specified route. ...
<void>	removeRoute (<Hashtable> _params)	Workaround (see mantis 0004126) Send the remove route command directly to the webervices ...
<void>	setDefaultDescriptionLanguage (<String> language)	This method sets the default language used for the textual description of the route. ...
<void>	setDefaultSpeedClassification (<Hashtable> _params)	Sets the default values for the assumed average speed on different road types. ...
<void>	showRoute (<Hashtable> _params)	Shows the specified route on the map. ...

4.5.1.4 Constructor Detail

```
Map24.RoutingServiceStub
Map24.RoutingServiceStub (<Hashtable> _params)
```

Creates a new Map24.RoutingServiceStub object.

Parameters:

```
<Hashtable> _params
```

Details see below:

Key	Type	Description
SpeedClassification (optional)	Hashtable	-
Map24.RoutingServiceStub.setDefaultSpeedClassification().	DescriptionLanguage (optional) String	The default language for the textual route description. If not set, the value "en" (English) is used.

4.5.1.5 Method Detail

4.5.1.5.1 calculateRoute

```
<void> calculateRoute (<Hashtable> _params)
```

Calculates a route using given Map24.Coordinate objects or Map24.Location objects as start and destination points.

This method calculates a route, if the start, destination, and optional via points are passed as map coordinates. With input parameters such as CalculationMode it is possible to fine tune the routing algorithm.

For retrieving the result a callback function must be passed in the CallbackFunction parameter. The result of type Map24.WebServices.Route is stored in the first parameter of the callback function.

The response contains the RouteID to identify the route on the server in subsequest requests. Furthermore, the response contains the geometry of the route and a detailed route description. It is possible to specify the language of the route description with the DescriptionLanguage parameter. Parameters:

```
<Hashtable> _params
```

A hashtable, details see below:

Key	Type	Description
Start (mandatory)	Map24.Location	Start point of the route.Alternatively, the start point of the route can be of type Map24.Coordinate.
Destination (mandatory)	Map24.Location	End point of the route.
Destination	Map24.Coordinate	Alternatively, the destination point of the route can be of type Map24.Coordinate.
ViaPoints (optional)	Array	Array of Map24.Location objects. Via points can be used to define the course of the route. The route will be calculated from its start to its destination point along the specified via points. It is possible to specify that the route either directly goes through a specified via point or passes the via point within a certain distance. The distance can be specified with the TransitRadius field of the Map24.Location object. For example, if you set the via point in a city center, you can set the radius in a way that the route goes along a highway nearby the city. Alternatively, the ViaPoints can be passed in an array of Map24.Coordinate objects.
DescriptionLanguage (optional)	String	The language for the textual route description. The following languages are supported: Czech (cs), German (de), Danish (da), English (en), Spanish (es), Finnish (fi), French (fr), Croatian (hr), Hungarian (hu), Icelandic (is), Italian (it), Dutch (nl), Norwegian (no), Polish (pl), Portuguese (pt), Romanian (ro), Russian (ru), Swedish (sv), Turkish (tr). It is possible to change the measuring unit in the route description to miles and yards by setting the DescriptionLanguage to British English (en_gb) or American English (us). The default value is "en", if not overwritten with the setDefaultDescriptionLanguage method.
CalculationMode (optional)	String	The mode for the route calculation. This may be either "Shortest" or "Fastest" and is "Fastest" by default. Be aware that the shortest route must not necessarily be the fastest.
VehicleType (optional)	String	Type of the vehicle with which the route is traversed. The value can be either "Car" or "Pedestrian". It is "Car" by default. In case of "Pedestrian" highways are not used. Furthermore, one-way rules and other direction instructions are ignored in this case.
SpeedClassification (optional)	Hashtable	An optional field for specifying the assumed average speed on different road types. The SpeedClassification settings can influence the calculation of the time for traversing the route (TotalTime field in the route description), but they do not have any effect on the course of the route. It is useful to change the default values if you know that the assumed average speed is different than

		the default value. For example, this can be the case for trucks if there is a speed limit on highways. Note: If you do not set the speed for all speed classifications, the setting will be ignored and the default values are used.	
CallbackFunction (mandatory)	Function	Function that is called after the client has received the routing response. The first parameter is of type Map24.WebServices.Route. The properties of Map24.WebServices.Route represent the result of the route calculation. It contains the RouteID, TotalLength, Segments and many other result values After the calculateRouteAddr was processed successfully, this parameter will contain the calculated route as result of the operation. The second parameter is the CallbackParameters hashtable that may contain further parameters. Alternatively, the callback function can be specified within an array.	
CallbackParameters (optional)	Hashtable	User-defined parameters that are passed to the callback function.	
ErrorHandler (optional)	function	A custom error handler can be optionally defined. The handler receives three parameters: the connection object, the request message, and the response.	
TimeoutHandler (optional)	function	A custom timeout handler can be optionally defined. The handler receives three parameters: The connection object, the request message, and the response.	
ShowRoute (optional)	Boolean	If true (default), the route is shown automatically on the map after the client has received the result from the routing service. If false, the route is not shown automatically. Instead, it is possible to show the route with the Map24.RoutingServiceStub.showRoute function. This way it is possible to change the default attributes of the route, for example the color of the route.	
IgnoreWaypoints (optional)	Boolean	The coordinates that describe the geometry of the route are called way points. If this field is set to FALSE (the default), all way points of the route are returned in the response. When setting this field to TRUE, only one coordinate per route segment is returned. It is possible to use this coordinate e.g. to center on a route segment or to show a logo for each route segment. This option can be used to reduce network traffic.	
IgnoreDescription (optional)	Boolean	If the parameter is set to true, only the RouteID is returned. If it is false (the default), the RouteID and a detailed route description is returned. The RouteID is needed to identify the route information on the server in subsequent requests.	
LoadTMCs (optional)	Boolean	If you have a license from the Mapsolute GmbH for retrieving traffic information, you can receive traffic information along the calculated route by setting the LoadTMCs flag to true.	String|If you have specified the start, destination, and via points with an address, the Map24 Geocoder will geocode these addresses. With the MinQualityOfGeocodeResults field you can specify the quality that

			the geocoded addresses should have at least. The default value is "Street" which means that the street contained in the geocoded address must match the street that was provided in the request. Otherwise, the route is not calculated. The possible values are: "Mismatch", "Exact", "Street", "ZIP", "District", "City", "State" and "Country".

The keys of the SpeedClassification hashtable are described below:

Key	Type	Description
Motorway	Integer	The assumed average speed on highways (default: 120 km/h)
CountryRoad	Integer	The assumed average speed on countryside roads (default: 60 km/h)
CityRoad	Integer	The assumed average speed on inner city roads (default: 40 km/h)
Ferry	Integer	The assumed average speed of ferries (default: 20 km/h).

4.5.1.5.2 hideRoute

```
<void> hideRoute (<Hashtable> _params)Hides the specified route.
```

Parameters:

```
<Hashtable> _params
```

Details see below:

Key	Type	Description
RouteId (mandatory)	String	The ID of a route.

4.5.1.5.3 removeRoute

```
<void> removeRoute (<Hashtable> _params)Workaround (see mantis 0004126) Send the remove route command direc
```

Parameters:

```
<Hashtable> _params
```

Details see below:

Key	Type	Description
RouteId (mandatory)	String	The ID of a route

4.5.1.5.4 setDefaultDescriptionLanguage

```
<void> setDefaultDescriptionLanguage (<String> language)This method sets the default language used for the
```

Parameters:

```
<String> language
```

- The following languages are supported:

Language	ISO3166-1	Alpha-2
Czech	cs	
German		de
Danish		da
English		en
Spanish		es

Finnish		fi
French		fr
Croatian		
Hungarian		
Icelandic		is
Italian		it
Dutch	nl	
Norwegian	no	
Polish	pl	
Portuguese	pt	
Romanian		ro
Russian		ru
Swedish		sv
Turkish		tr

Note: For the English language it is possible to change the measuring unit in the route description to miles and yards by setting the DescriptionLanguage to British English (en_gb) or American English (us).

4.5.1.5.5 setDefaultSpeedClassification

```
<void> setDefaultSpeedClassification (<Hashtable> _params)Sets the default values for the assumed average s
```

Parameters:

```
<Hashtable> _params
```

A hashtable for specifying the assumed average speed on different road types:

Key	Type	Description
CityRoad	String	The assumed average speed on inner city roads (initial default: 40 km/h)
CountryRoad	String	The assumed average speed on countryside roads (initial default: 60 km/h)
Ferry	String	The assumed average speed of ferries (initial default: 20 km/h)
Motorway	String	The assumed average speed on motorways (initial default: 120 km/h)

4.5.1.5.6 showRoute

```
<void> showRoute (<Hashtable> _params)Shows the specified route on the map.
```

This method can be used to show a previously calculated route in a mapping client. The route information is cached on the server for 45 minutes after a route was calculated. It is necessary to pass the RouteId of the calculated route. The RouteId is returned with the result of a calculateRoute function.

This method performs the following actions: ? At first, this function creates a Map24 Route object used for showing the specified route on the map. ? Then it sets the route object visible on the map. ? Finally, it sets the map view to the route object so that it is shown in the mapping client. Parameters:

```
<Hashtable> _params
```

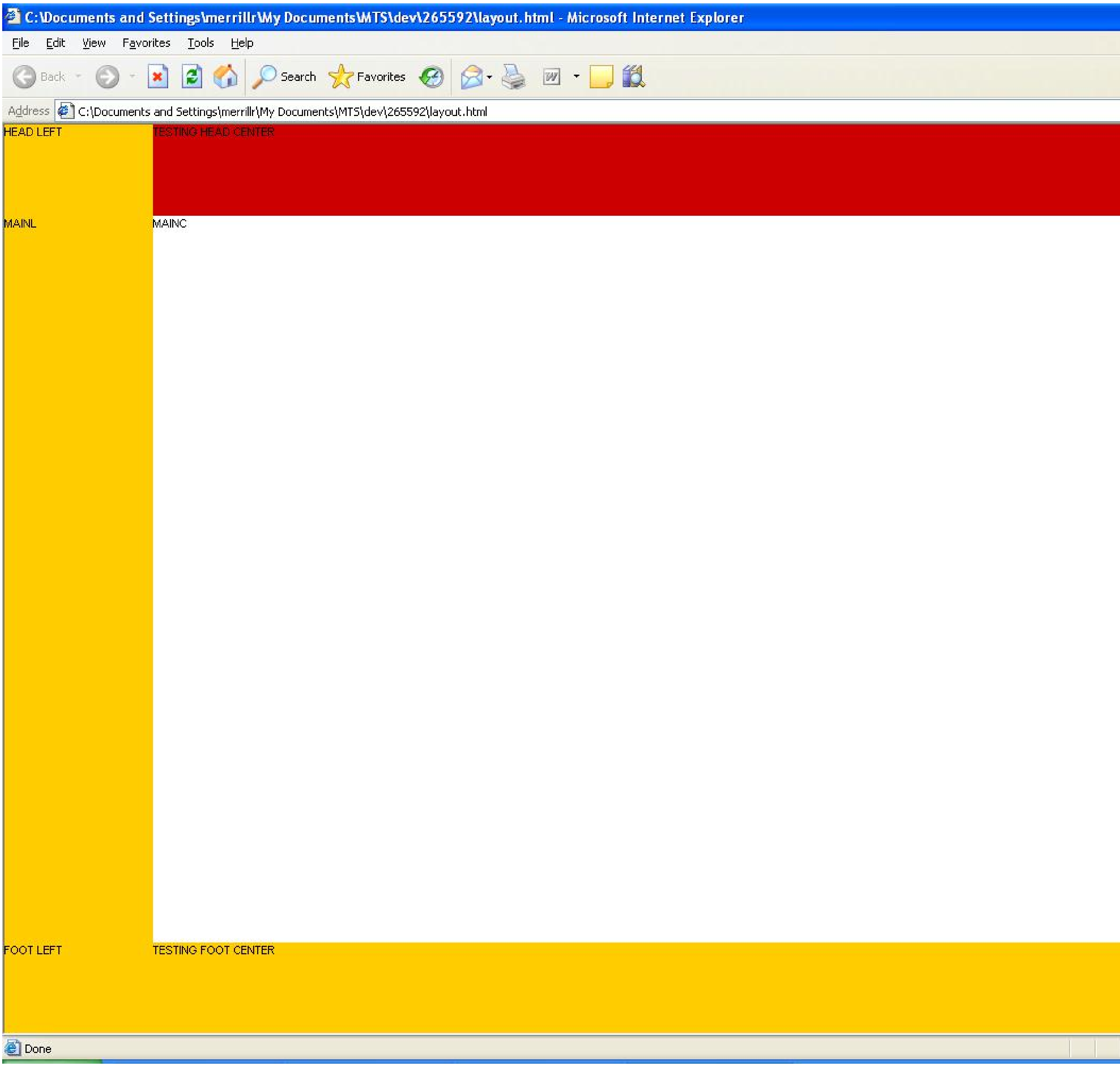
Details see below:

Key	Type	Description
RouteId (mandatory)	String	The ID of a route that was calculated.
ViewPercentage (optional)	Integer	whole route becomes visible.
Color (optional)	String	The color of the route as rgb value (without transparency). The color value can be specified either with #rrggbb or #rgb. The default color is blue.
	Array	Alternatively, the color can be specified in an array. Index 0 contains the color value (#rrggbb or #rgb). Index 1 contains the alpha value in the

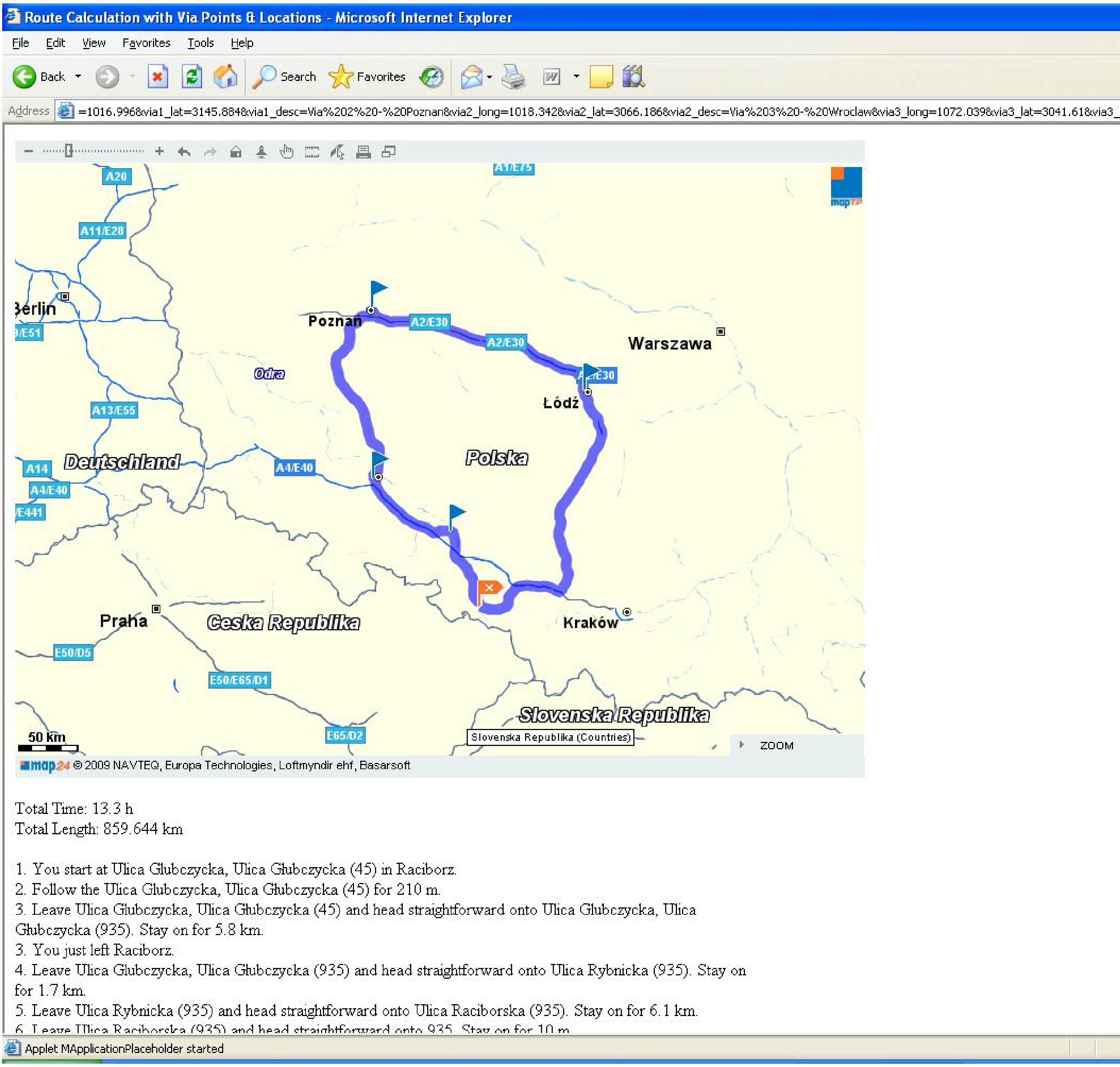
	range 0 to 255 that defines the transparency. Example: ['#00F', 150] defines a partly transparent blue route.
Map24.WebServices.Color	Alternatively, the color can be specified with a Map24.WebServices.Color object.

4.6 Appendix C

4.6.1 Style Sheet


C:\Documents and Settings\merrillr\My Documents\MTS\dev\265592\layout.html - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Search
Favorites
Address
C:\Documents and Settings\merrillr\My Documents\MTS\dev\265592\layout.html
HEAD LEFT
TESTING HEAD CENTER
MAINL
MAINC
FOOT LEFT
TESTING FOOT CENTER
Done

4.6.2 Sample NavTeq Map


Route Calculation with Via Points & Locations - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Search
Favorites
Address
=1016.996&via1_lat=3145.884&via1_desc=Via%202%20-%20Poznan&via2_long=1018.342&via2_lat=3066.186&via2_desc=Via%203%20-%20Wroclaw&via3_long=1072.039&via3_lat=3041.61&via3_
A20
A1/E75
A11/E28
Berlin
/E51
Poznan
A2/E30
A2/E30
Warszawa
Odra
A2/E30
Łódź
A13/E55
Polska
A14
Deutschland
A4/E40
A4/E40
E441
Praha
Ceska Republika
Kraków
E50/D5
E50/E65/D1
Slovenska Republika
Slovenska Republika (Countries)
50 km
E65/D2
ZOOM
map24 © 2009 NAVTEQ, Europa Technologies, Loftmyndir ehf, Basarsoft

Total Time: 13.3 h
Total Length: 859.644 km

1. You start at Ulica Glubczycka, Ulica Glubczycka (45) in Raciborz.
2. Follow the Ulica Glubczycka, Ulica Glubczycka (45) for 210 m.
3. Leave Ulica Glubczycka, Ulica Glubczycka (45) and head straightforward onto Ulica Glubczycka, Ulica Glubczycka (935). Stay on for 5.8 km.
3. You just left Raciborz.
4. Leave Ulica Glubczycka, Ulica Glubczycka (935) and head straightforward onto Ulica Rybnicka (935). Stay on for 1.7 km.
5. Leave Ulica Rybnicka (935) and head straightforward onto Ulica Raciborska (935). Stay on for 6.1 km.
6. Leave Ulica Raciborska (935) and head straightforward onto 935. Stay on for 10 m.

5 References

Ref. No.	Document Title & ID	Version	Date

6 Document History

Version	Date	Status	Reason	Initials
0.1	19/06/09	Draft	Initial version	DRM
0.2	19/06/09	Draft	Redrafted	MJC
0.3	23/06/09	Draft	Redrafted	DRM
2.0	23/06/09	Issues	Review and issue	PC

7 Authorised By

Dave Meir	Development Manager
Suk Sandhu	TMSCC MTS Product Manager