Routing Reparto

API
SERVIDOR ROUTING REPARTO

Company: Nexus Geografics SL and NEWRONIA SL
Date: 6/10/2014
Document: API_Routing_Reparto 3.3.doc
Version: 3.3

nexusgeografics

NEWRONIA
OPTIMIZING TECHNOLOGIES

Index

1. Introduction

Routing Reparto is an online application for route planning and optimization for business fleets. Is helps to save costs and to improve customer satisfaction.

Routing Reparto has two modules:

- ✓ Planning: Route planning and optimization using the online application, from the office
- ✓ Monitoring: reporting, manage visits and real time monitoring, based on the driver's APP

This document describe the API for Planning, revising calls and parameters accepted by Routing Reparto server: http://www.routingreparto.com/planificador/server?

For servi**ce requests** use HTTP protocol: GET (limited to 1024 bytes) or POST with enctype **multipart/form-data** or **application/x-www-form-urlencoded**. The **service response** is in XML format.

The entire service request must include the Cercalia client ID: this code is provided by Nexus Geografics.

2. Planning

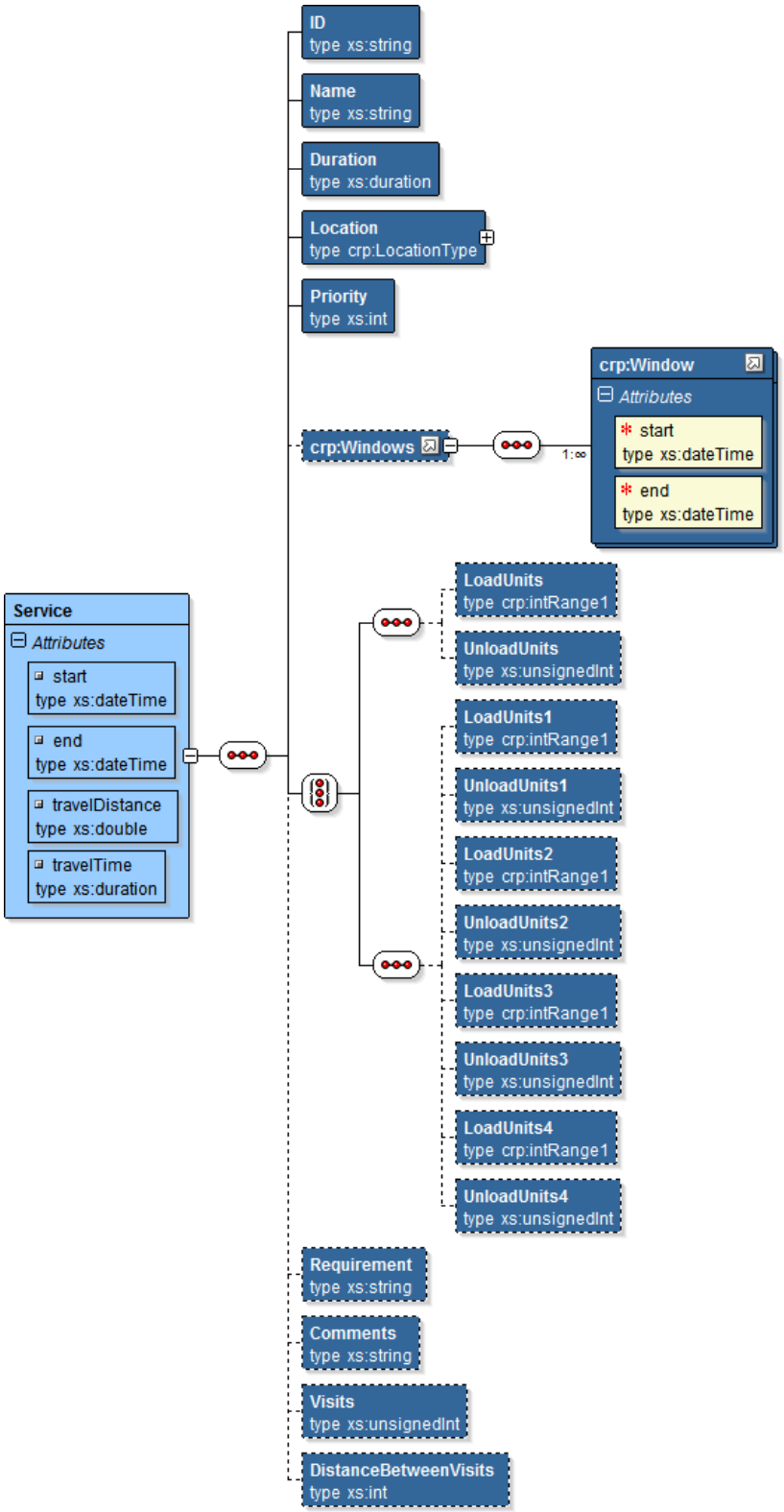
4.1 Planning request

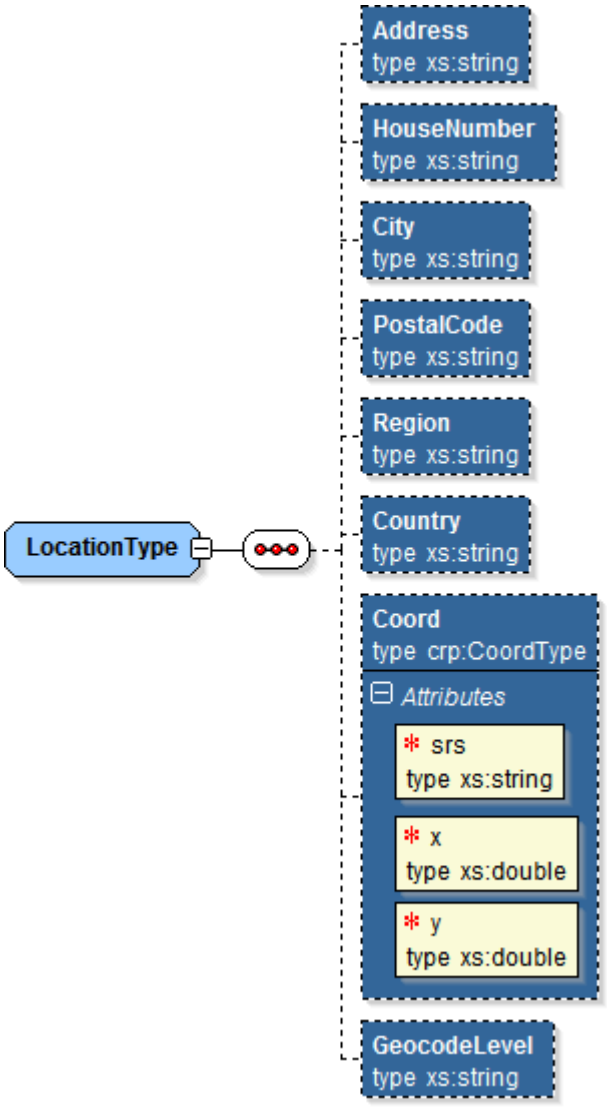
Planning request parameters:

cmd	Always use **start**.
clientid	Cercalia client ID. (Mandatory)
client	Client code. (Optional)
request	XML with this format: <Request> <Vehicles> ... </Vehicles> <Services> ... </Services> <Settings> ... </Settings> </Request> Optional: not include one or more above XML sections, AND use next parameters:
services	Xml with the services to include in the planning. <Services> ... </Services>
vehicles	Xml with the vehicles to include in the planning. <Vehicles> ... </Vehicles>
settings	Xml with the planning parameters. <Settings> ... </Settings>

The services and vehicles list must be correctly geocoded (with XY available).

2.1.1 Services specification (customers)





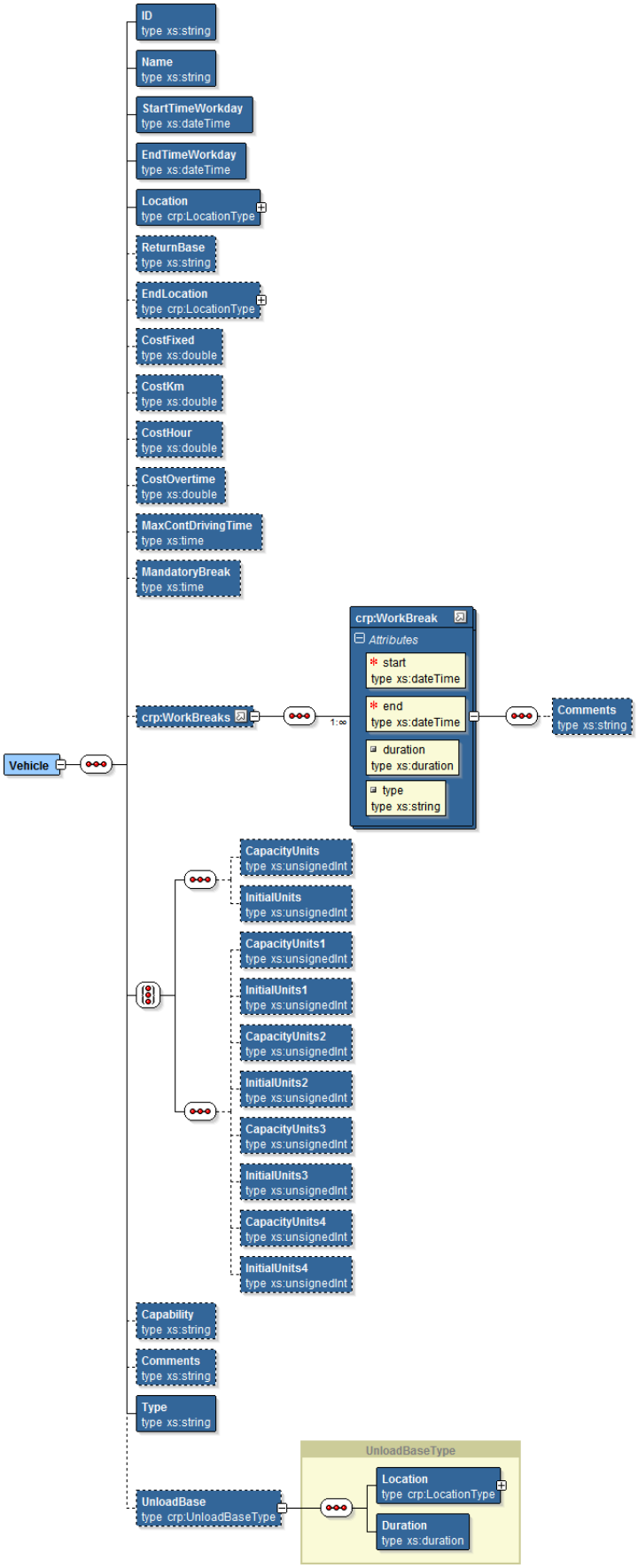
Data Service description:

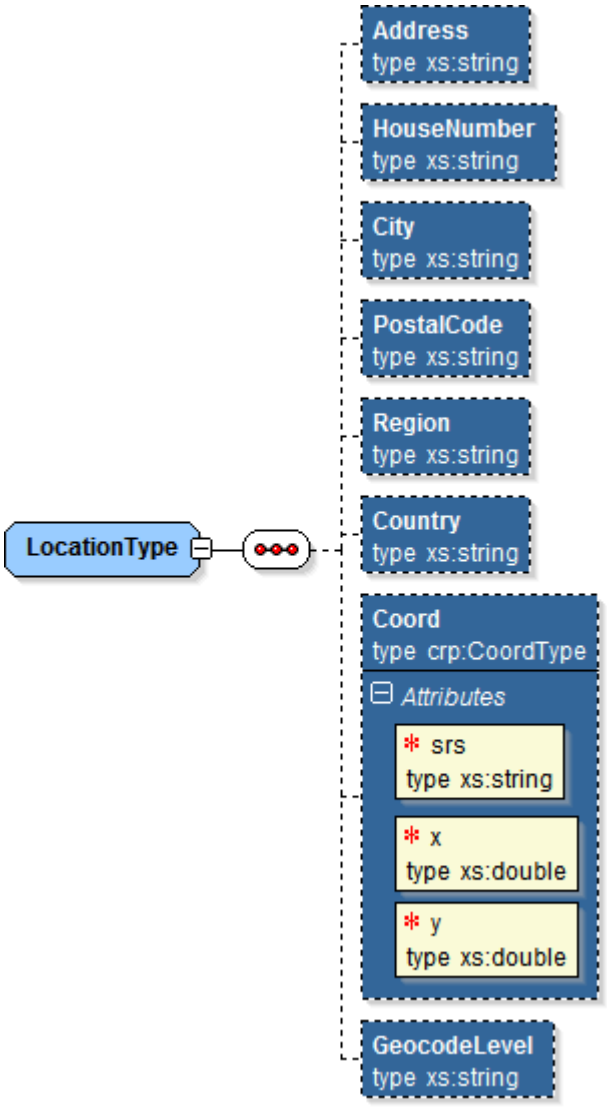
ID	Unique service ID.
Name	Name or service description.
Duration	Service duration.
Location	Service address Location.
Priority	Service priority (higher number = more priority)
Windows	Preferred time windows for each service. You can use several time windows.
Requirement	Service Requirement (hard restriction). Use a numerical values (with “,” break) matching with the vehicle capability value.
Comments	Service comments.
LoadUnits<X>	Units of type <X> loaded on this service. Where <X> is a value between 1 and 4. If only exists one type of units. you can also use the parameter **LoadUnits.**
UnloadUnits<X>	Units of type <X> unloaded. Where <X> is a value between 1 and 4. If only exists one type of units, you can also use the parameter **UnloadUnits**

Example:

```
<Service>
        <ID>1</ID>
        <Name>1</Name>
        <Duration>00:30:00</Duration>
        <Location>
                <Address></Address>
                <HouseNumber></HouseNumber>
                <City></City>
                <PostalCode></PostalCode>
                <Region></Region>
                <Country></Country>
                <Coord x=”-3.5433333” y=”40.5894444” srs=”EPSG:4326”/>
        </Location>
        <Priority>1</Priority>
        <Windows>
                <Window start=”2011-02-01T09:00:00” end=”2011-02-01T14:00:00” />
                <Window start=”2011-02-01T16:00:00” end=”2011-02-01T17:00:00” />
        <Windows>
        <Requirement>1,2</Requirement>
        <Comments>Ask for Peter</Comments>
<Service>
```

2.1.2 Vehicles specification





Data Vehicles description:

ID	Unique Vehicle ID
Name	Name or vehicle description
StartTimeWorkday	Starting workday time
EndTimeWorkday	End workday time
Location	Location address starting point.
ReturnBase	After last service, return to base? (Y – Yes, N - No)
EndLocation	Location address end point. (If null, use Location address starting point),
CostFixed	Optional. Fixed cost for using every vehicle.
CostKm	Optional. Vehicle cost per km
CostHour	Optional. Vehicle cost per hour
CostOvertime	Optional. Vehicle cost per overtime.
MaxContDrivingTime	Optional. Maximum continuous driving time, with any stop. Default value 23:59:55 hours.
MandatoryBreak	Optional. Mandatory break after maximum continuous driving time. Default value: 0.
WorkBreaks	Optional. List of break times (lunch, dinner, etc)
Capability	Optional. Vehicle capabilities to solve service requirements. Use a numerical value (with “,” break) matching with the service requirements.
Type	Vehicle type: A – Vehicle B – Light truck C – Heavy truck

Comments	Optional. Vehicle comments.
CapacityUnits<X>	Optional. Maximum number of units of type <X> that this vehicle can carry. Every type is defined by an integer value (values 1 to 4) If there is only one type of units, you can use **CapacityUnits**.
InitialUnits<X>	Optional. Number of units of type <X> that the vehicle carries when leaves the base. If there is only one type of units, you can use **InitialUnits**.

Location description:

Address	Address
HouseNumber	House number
City	City
PostalCode	Postal Code
Region	Region name
Country	Country name
Coord	Location Coordinates.

Location coordinates description:

srs	Spatial reference system (coordinates system)
x	X Coordinate
y	Y Coordinate

Work break description:

start	Starting time work break
end	End time work break
duration	Work break duration
Comments	Work break comments

Load/Unload Base description:

Location	Location of Base where the vehicle goes to unload or reload
Duration	Duration of unloading/reloading

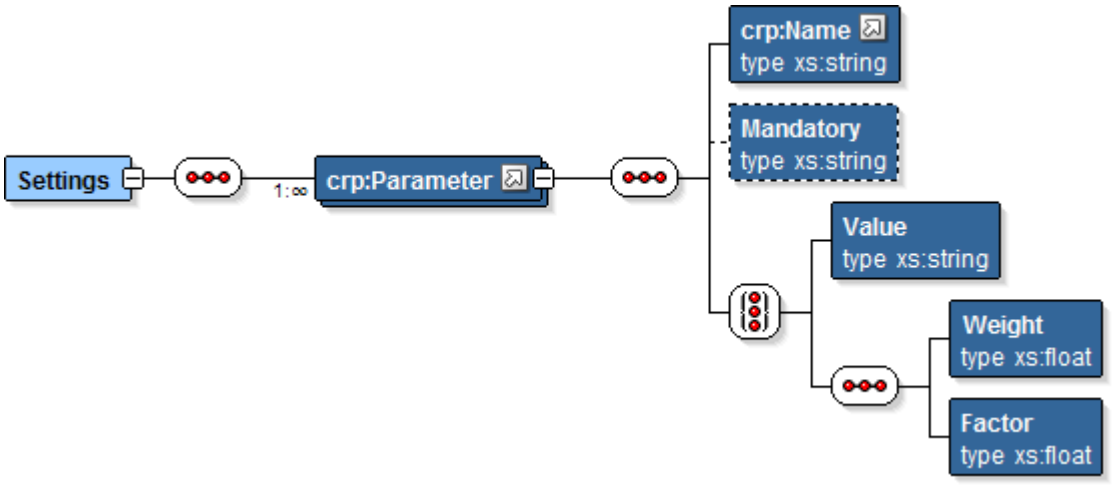
Example:

```
<Vehicle>
        <ID>1</ID>
        <Name>1</Name>
        <StartTimeWorkday>2011-02-01T08:00:00</StartTimeWorkday>      <EndTimeWorkday>2011-02-
01T16:00:00</EndTimeWorkday>
        <Location>
                <Address></Address>
                <HouseNumber ></ HouseNumber>
                <City></City>
                <PostalCode></PostalCode>
                <Region></Region>
                <Country></Country>
                <Coord x=”-3.5433333” y=”40.5894444” srs=”EPSG:4326”/>
        </Location>
        <ReturnBase>Y</ReturnBase>
        <EndLocation>
                <Address></Address>
                <HouseNumber ></ HouseNumber>
                <City></City>
                <PostalCode></PostalCode>
                <Region></Region>
                <Country></Country>
```

```
                <Coord x="-3.5433333" y="40.5894444" srs="EPSG:4326"/>
        </EndLocation>
        <CostFixed>20</CostFixed>
        <CostKm>0.2</CostKm>
        <CostHour>20</CostHour>
        <CostOvertime>50</CostOvertime>
        <MaxContDrivingTime>04:30:00</MaxContDrivingTime>
        <MandatoryBreak>00:45:00</MandatoryBreak>
        <WorkBreaks>
                <WorkBreak start="01/02/2011 13:00:00" end ="01/02/2011 14:00:00" />
        </WorkBreaks>
        <Capability>1,2,3,4</Capability>
        < Type >A</ Type >
        <Comments></Comments>
</Vehicle>
```

2.1.3 Configuration

A) Planning configuration:



Parameters list:

Priority	Service priority (higher number = more priority) **Weight:** Priority weight vs other planning factors. **Factor:** Economical value of priority factor.
Cost	**Weight:** Cost weight vs other planning factors **Factor:** Economical value of Cost factor.
Delay	**Mandatory**: Allow delays vs preferred time windows (if 'Y' = hard restriction). **Weight:** Delay weight vs other planning factors. **Factor:** Economical value of Delay factor.
AvoidOvertime	'Y' – Avoid extra hours 'N' - Allow extra hours

AvoidExtraDriving	'Y' - Maximum continuous driving time restriction 'N' - Allow overcome maximum continuous driving time.
Allocate	**Mandatory**: Distribute work among the vehicles **Weight:** Work distribution weight vs other planning factors. **Factor:** Economical value of Allocate factor
AllocateMode	Mode of vehicles effort balancing: • **NumberOfServices**: balancing the number of services. • **LoadDistribution**: balancing the load of vehicles. • **Work**: balancing the workload. Default: Work
GroupServices	Facilitates compacting services, especially in urban areas, to avoid wrong route-crossing. Tries to assign nearby services to the same vehicle • **Mandatory**: Enables Group Services • **Weight:** Group Services weight vs other planning factors • **Factor:** Economic value of Group Services functionallity.
GroupServicesMaxMeters	Maximum distance (in meters) between nearby services.
Network	Optional. Network to use.
RouteWeight	Optional. Calculate optimized routes for • **time** • **distance** Default: **time**.
Iterations	Number of iterations. Recommended use the default value
RestrictToServiceWindow	'Y' = the service duration must respect the preferred time windows 'N' = only the beginning of the service must respect the preferred time windows
MaxServiceDelay	Maximum service delay respect preferred time widows (in minutes).
OvertimeUnits	Time fraction, in minutes, for overtime. The overtime cost is calculated using this time fraction value
LimitOvertimeBefore	Maximum time, in minutes, before workday
LimitOvertimeAfter	Maximum time, in minutes, after workday
MaxDiffAllocate	Maximum difference between vehicles workload. Use this parameter to give an equal distribution of work among the vehicles.
AllVehicles	Optional. Default is 'N': try to minimize the number of vehicles. If enabled 'Y' try to use all the vehicles.

Example XML configuration:

```
<Settings>
	<Parameter>
		<Name>Priority</Name>
		<Weight>25</Weight>
		<Factor>1000</Factor>
	</Parameter>
	<Parameter>
		<Name>Cost</Name>
		<Weight>20</Weight>
		<Factor>100</Factor>
	</Parameter>
	<Parameter>
		<Name>Delay</Name>
		<Mandatory>Y/N</Mandatory>
		<Weight>10</Weight>
		<Factor>50</Factor>
	</Parameter>
	<Parameter>
		<Name>AvoidOvertime</Name>
		<Mandatory>Y/N</Mandatory>
	</Parameter>
	<Parameter>
		<Name>AvoidExtraDriving</Name>
		<Mandatory>Y/N</Mandatory>
	</Parameter>
	<Parameter>
		<Name>Allocate</Name>
		<Mandatory>Y</Mandatory>
		<Weight>15</Weight>
		<Factor>100</Factor>
	</Parameter>
	<Parameter>
		<Name>RouteWeight</Name>
		<Value>Time </Value>
	</Parameter>
	<Parameter>
		<Name>Iterations</Name>
		<Value>25</Value>
	</Parameter>
	<Parameter>
		<Name>Network</Name>
		<Value></Value>
	</Parameter>
	<Parameter>
		<Name>RestrictToServiceWindow</Name>
		<Value>Y</Value>
	</Parameter>
	<Parameter>
		<Name>MaxServiceDelay</Name>
		<Value>30</Value>
	</Parameter>
	<Parameter>
		<Name>OvertimeUnits</Name>
		<Value>30</Value>
	</Parameter>
	<Parameter>
		<Name>LimitOvertimeBefore</Name>
		<Value>30</Value>
	</Parameter>
	<Parameter>
		<Name>LimitOvertimeAfter</Name>
		<Value>30</Value>
	</Parameter>
	<Parameter>
		<Name> MaxDiffAllocate</Name>
		<Value>30</Value>
	</Parameter>
</Settings>
```

To consider:

***Priority**:* Must be a high value (ex: 1000), other values substract.

Factor: Indicates the conversion factor cost in euros.

***Weight**:* The sum of all weights should be 100.

The optimization criterion:

$$f(s) = P_P*F_P*\text{Priority} – (P_C*F_C * \text{Cost(s)} + P_T*F_T * \text{Delay} + P_R*F_R * \text{Allocate})$$

4.2 Planning response

The planning response is generated en XML format, including a job ID. With this job ID you can send request to know the planning progress status. Also you can receive an error response.

```
<Response>
    <Job id=”XXXX” />
</Response>
```

Error response:

```
<Response>
    <Errors>
        <Error code=”30002” desc=”Error interno del servidor” />
    </Errors>
</Response>
```

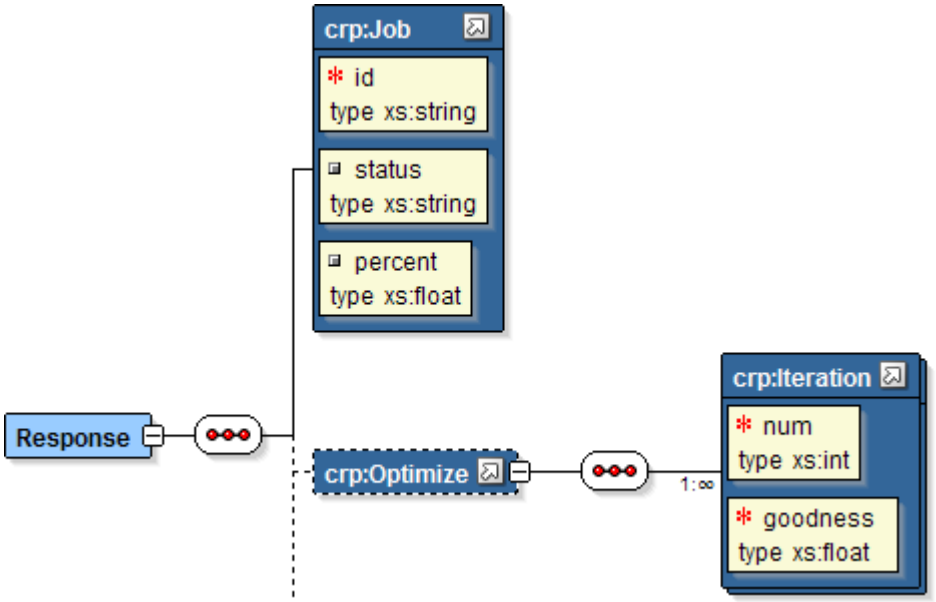
4.3 Planning process status

With this request you can know the status job phase, and the progress status.

cmd	Always use **status.**
clientid	Cercalia client ID. (Mandatory)
client	Client code. (Optional)
job	Job ID

4.4 Status response

Response structure:



Example:

```
<Response>
        <Job ID=”XXXX” status=” OPTIMIZE” percent =”50.0” />
        <Optimize maxgoodness=”122321.32”>
            <Iteration num=”1” goodness=”2331.34” />
            <Iteration num=”2” goodness=”3531.34” />
        </Optimize>
</Response>
```

Or in case of error:

```
<Response>
        <Errors>
                <Error code=”30002” desc=”Error interno del servidor” />
        </Errors>
</Response>
```

Status values:

QUEUE_MATRIX	In queue, waiting for distance & time matrix calculation.
BUILDING_MATRIX	Calculating distance & time matrix.
QUEUE_OPTIMIZE	In queue, waiting for planning
OPTIMIZE	Computing planning
BUILDING_RESPONSE	Building response
DONE	Planning finished.

For **OPTIMIZE** status, response include the tag **Optimize.** The **maxgoodness** attribute indicates the goodness of the theoretical perfect solution (for a distance cost = zero). For every status this attribute include the goodness of the best solution found.

If the status is **DONE** you need to do a new request for obtain this information (see “Planning results

request").

4.5 Cancel request

You can cancel a job at any time.

cmd	Always use **cancel.**
clientid	Cercalia client ID. (Mandatory)
client	Client code. (Optional)
job	Job ID

4.6 Cancel response

If the job status is **OPTIMIZE** wen the job is canceled, the response includes the planning result, in the tag **<Plan>**

```
<Response>
	<Job ID="XXXX" status="DONE" percent ="32" />
	<Plan> .... </Plan>
</Response>
```

In case of error:

```
<Response>
	<Errors>
		<Error code="30002" desc="Error interno del servidor" />
	</Errors>
</Response>
```

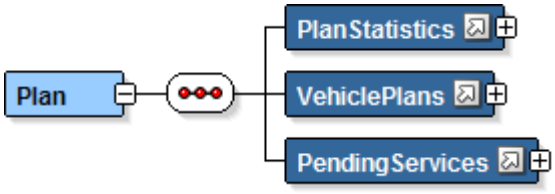
4.7 Planning results request

The planning results are only available during a few minutes in the service, once the planning is computed.

cmd	Always use **result.**
clientid	Cercalia client ID. (Mandatory)
client	Client code. (Optional)
job	Job ID

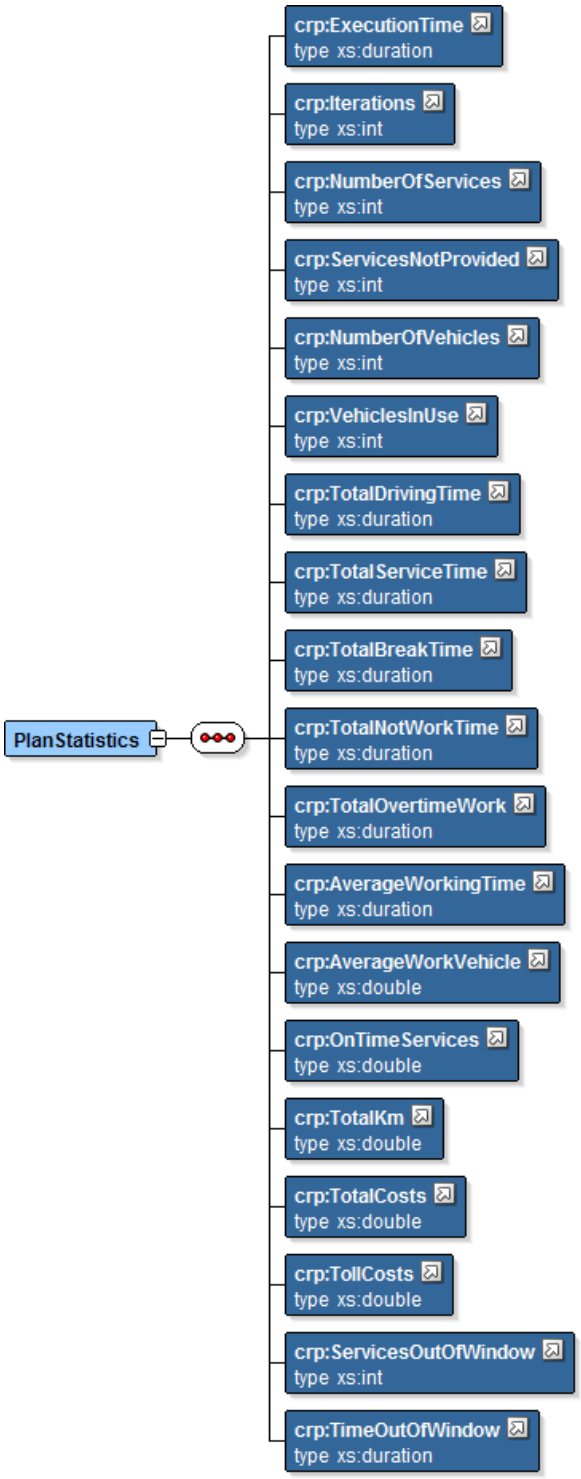
4.8 Planning results response

The XML response is a planning **<Plan>** composed by global statistics planning **<PlanStatistics>,** the vehicles used in the planning **<VehiclePlans>** and the services discarded **<PendingServices>**



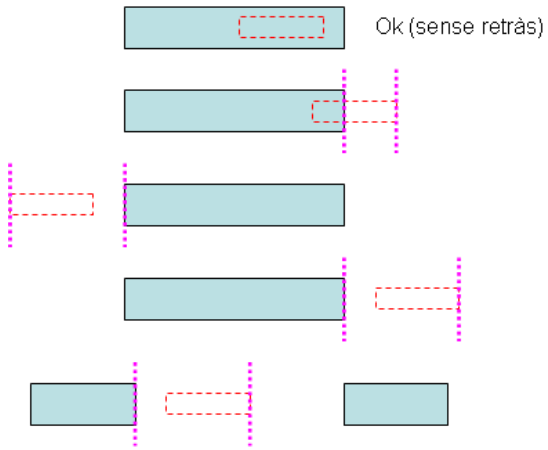

```
<Response>
        <Job ID=”XXXX” status=”DONE” percent =”100” />
        <Errors>
                <Error code=”30002” desc=”Error interno del servidor” />
        </Errors>
        <Plan> .... </Plan>
</Response>
```

The global statistics planning includes these elements:

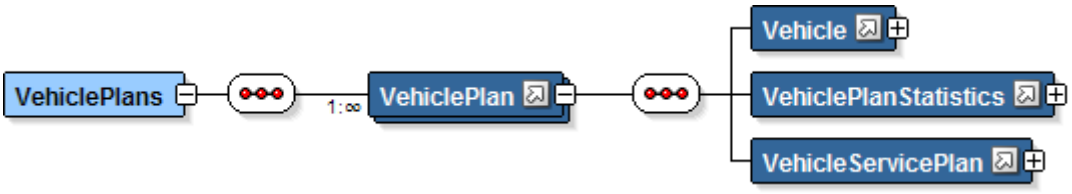


where

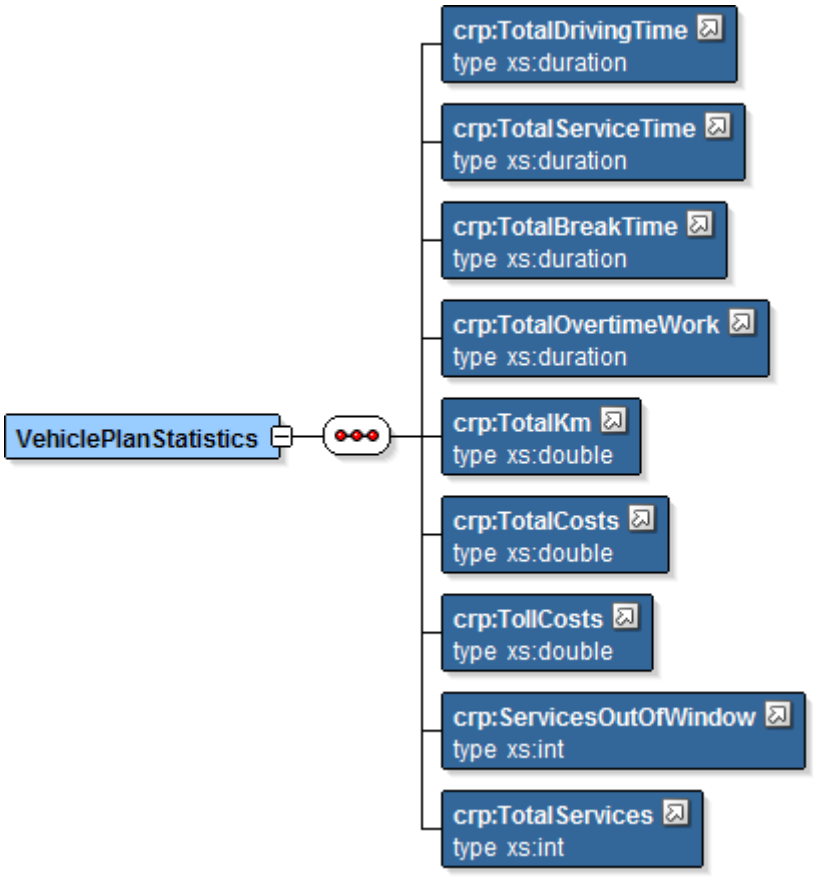
ExecutionTime	Total calculation time
Iterations	Number of iterations
NumberOfServices	Number of services
ServicesNotProvided	Number of discarded services
TotalServices	Number of planned services
NumberOfVehicles	Number of vehicles available
VehiclesInUse	Number of vehicles used in the planning

TotalDrivingTime	Global driving time
TotalServiceTime	Global service time
TotalBreakTime	Global rest time
TotalNotWorkTime	Global non-working time, considering the vehicle schedule
TotalOvertimeWork	Global overtime work
AverageWorkingTime	Average working time for vehicle (TotalDrivingTime + TotalServiceTime / VehiclesInUse)
AverageWorkVehicle	Average working time for vehicle, including rest time (TotalDrivingTime + TotalServiceTime + TotalBreakTime / VehiclesInUse)
OnTimeServices	Percentage of punctuality expected for services: ServicesOutOfWindow / (NumberOfServices – ServicesNotProvided)
TotalKm	Total Km
ServicesOutOfWindow	Number of services out of preferred time-window
TimeOutOfWindow	Total delay for the planned services: Casuistry: 
OrdinaryHoursCosts	Total ordinary cost per hour.
OvertimeCosts	Total overtime work cost per hour.
WorkingCosts	Global working costs (OrdinaryHoursCosts + OvertimeCosts).
KmCosts	Total cost per km
TollCosts	Toll costs (not available at this moment)
TotalCosts	Global planning cost

<VehiclePlans> is a list of **<VehiclePlan>**. Each **<VehiclePlan>** contains the vehicle data **<Vehicle>,** the vehicle statistics **<VehiclePlanStatistics>** and the vehicle service planning **<VehicleServicePlan>**.



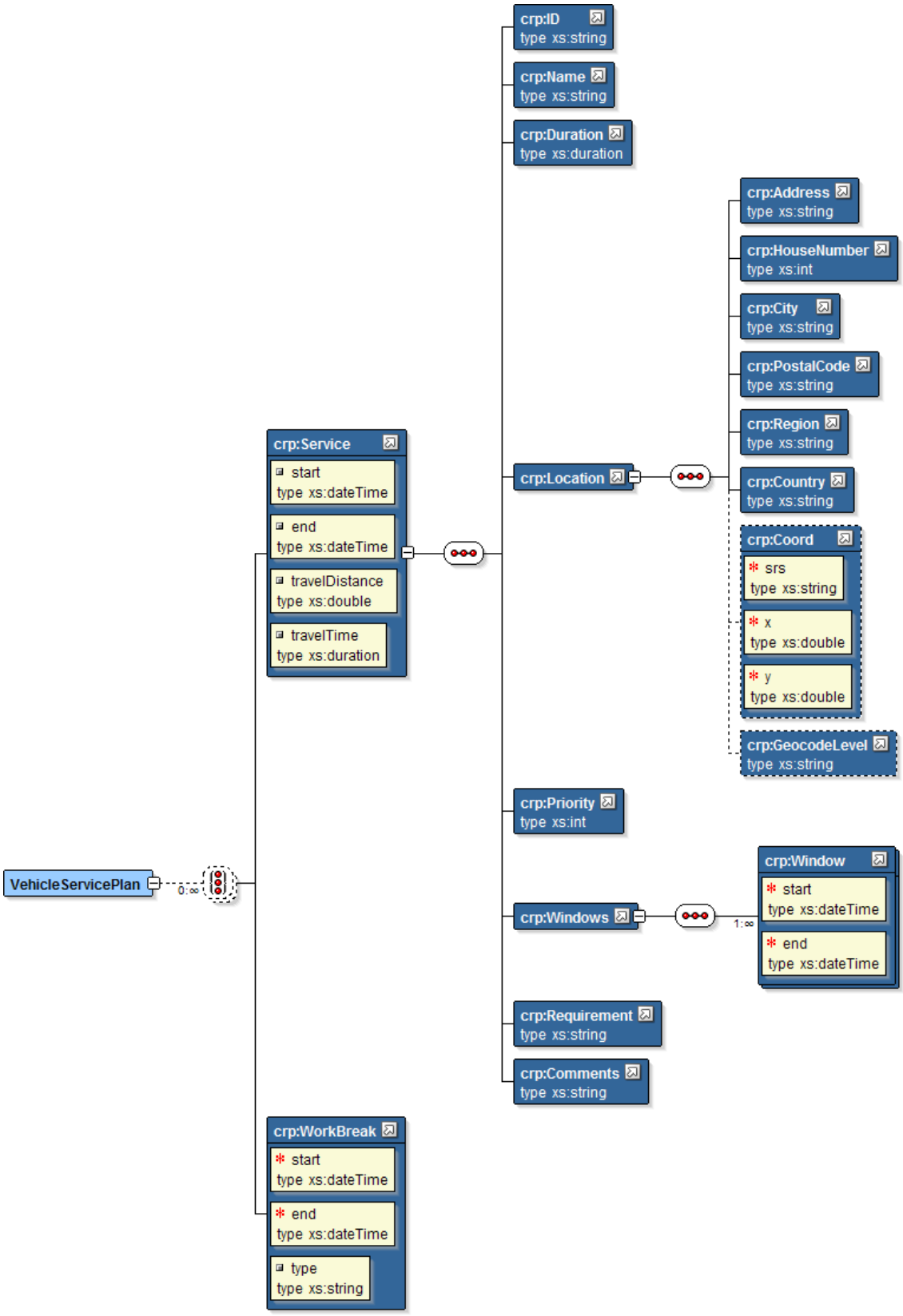
The vehicle statistics **<VehiclePlanStatistics>** are composed by (for every vehicle):



Where

TotalDrivingTime	Global driving time
TotalServiceTime	Global service time
TotalBreakTime	Global rest time
TotalOvertimeWork	Global overtime work
TotalKm	Total Km
TotalCosts	Global cost
TollCosts	Toll costs (not available at this moment)
ServicesOutOfWindow	Number of services out of preferred time-window
TotalServices	Total planned services

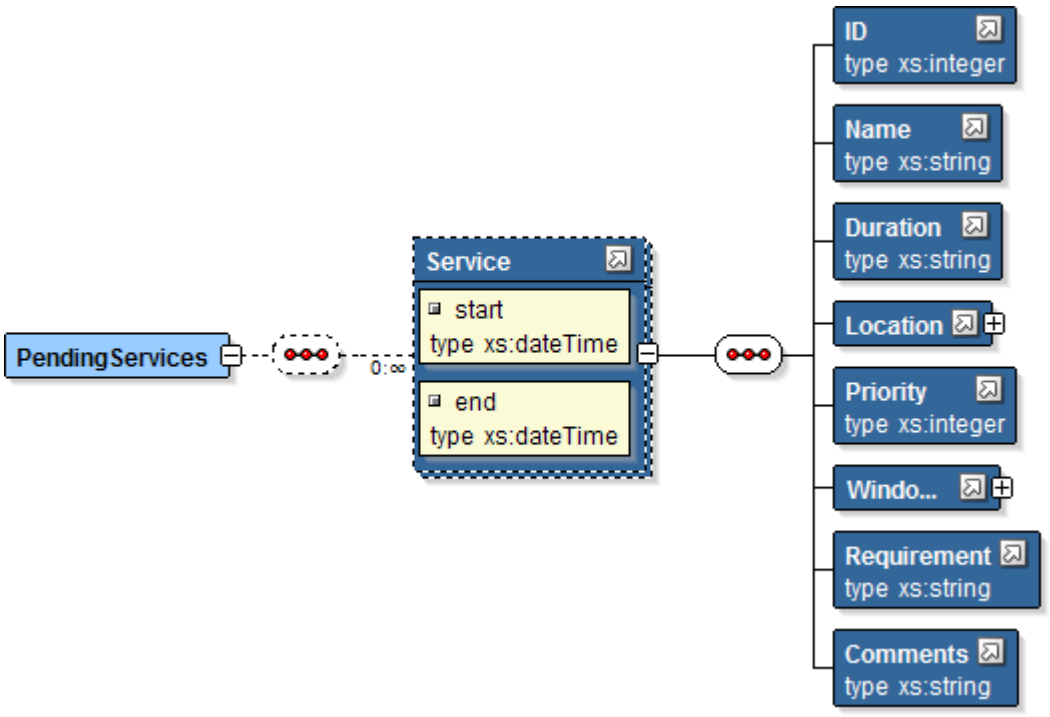
The vehicle service planning **<VehicleServicePlan>** is composed by a list of planned services **<Service>** and work breaks **<WorkBreak>** ordered by date and scheduling time. The **start** and **end** services attributes must be reported.



The WorkBreak types are:

DrivingBreak	Workbreak for driving time exceed
SpecifiedBreak	Specified break (dinner, etc)
BothBreak	Both kind of breaks.

<PendingServices> includes a not planned service list.



XML example:

```
<?xml version="1.0" encoding="windows-1252" ?>
<Response>
<Plan xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="RespostaOptimitzacio.xsd">
  <PlanStatistics>
    <!—Tiempo total de cálculo →
    <ExecutionTime>00:02:50</ExecutionTime>
    <!—Número iteraciones calculadas →
    <Iterations>12</Iterations>
    <!—Número total de servicios →
    <NumberOfServices>30</NumberOfServices>
    <!—Servicios descartados →
    <ServicesNotProvided>5</ServicesNotProvided>
    <!—Total vehículos disponibles →
    <NumberOfVehicles>5</NumberOfVehicles>
    <!—Total vehículos usados →
    <VehiclesInUse>3</VehiclesInUse>
    <!—Tiempo total conducción →
    <TotalDrivingTime>10:02:50</TotalDrivingTime>
    <!—Tiempo total en servicio →
    <TotalServiceTime>32:02:50</TotalServiceTime>
    <!—Tiempo total de descanso →
    <TotalBreakTime>10:02:50</TotalBreakTime>
    <!—Tiempo total sin ningún servicio ni descanso, según el horario de cada vehículo →
    <TotalNotWorkTime>30:02:50</TotalNotWorkTime>
    <!—Tiempo total de trabajo extra →
```

```
      <TotalOvertimeWork>00:00:00</TotalOvertimeWork>
      <!—Media de tiempo de trabajo y conducción por vehículo utilizado (TotalDrivingTime + TotalServiceTime / VehiclesInUse)
→
      <AverageWorkingTime>00:00:00</AverageWorkingTime>
      <!—Porcentaje medio de trabajo por vehículo, respecto a su horario, incluyendo tiempo de conducción, tiempo en servicio,
y descansos establecidos & obligatorios.→
      <AverageWorkVehicle>50.43</AverageWorkVehicle>
      <!—Porcentaje medio de puntualidad prevista en los servicios: ServicesOutOfWindow / (NumberOfServices –
ServicesNotProvided) →
      <OnTimeServices>23.45</OnTimeServices>
      <!—Km totales →
      <TotalKm>321.32</TotalKm>
      <!—Coste total →
      <TotalCosts>213</TotalCosts>
      <!—Costes de peajes →
      <TollCosts>12.32</TollCosts>
      <!—Servicios fuera de horario de entrega →
      <ServicesOutOfWindow>2</ServicesOutOfWindow>
      <!—Retraso total acumulado de todos los servicios planificados. Casuística: →
      <TimeOutOfWindow>00:00:00</TimeOutOfWindow>
   </PlanStatistics>
   <VehiclePlans>
      <VehiclePlan>
         <Vehicle>
            <ID>1</ID>
            <Name>1</Name>
            <StartTimeWorkday>2011-02-01T08:00:00</StartTimeWorkday>
            <EndTimeWorkday>2011-02-01T16:00:00</EndTimeWorkday>
            <Location>
               <Address>Botet I Sissó</Address>
               <HouseNumber>6</HouseNumber>
               <City>Girona</City>
               <PostalCode>17003</PostalCode>
               <Region>Girona</Region>
               <Country>España</Country>
               <Coord x=”-12.5433333” y=”40.5894444” srs=”EPSG:4326”/>
               <GeocodeLevel>ADR</GeocodeLevel>
            </Location>
            <CostKm>0.2</CostKm>
            <CostHour>20</CostHour>
            <MaxContDrivingTime>04:30:00</MaxContDrivingTime>
            <MandatoryBreak>00:45:00</MandatoryBreak>
            <Workbreaks>
               <WorkBreak start=”2011-02-01T09:00:00”
                     end=”2011-02-01T10:00:00”/>
               <WorkBreak start=”2011-02-01T13:00:00”
                     end=”2011-02-01T14:00:00”/>
            </WorkBreaks>
            <Capability>1,2,3,4</Capability>
            <Comments></Comments>
         </Vehicle>
         <VehiclePlanStatistics>
            <!—Tiempo total conducción →
            <TotalDrivingTime>10:02:50</TotalDrivingTime>
            <!—Tiempo total en servicio →
            <TotalServiceTime>32:02:50</TotalServiceTime>
            <!—Tiempo total de descanso →
            <TotalBreakTime>10:02:50</TotalBreakTime>
            <!—Tiempo total de trabajo extra →
            <TotalOvertimeWork>00:00:00</TotalOvertimeWork>
            <!—Km totales →
            <TotalKm>321.32</TotalKm>
            <!—Coste total →
            <TotalCosts>213</TotalCosts>
            <!—Costes de peajes →
            <TollCosts>12.32</TollCosts>
            <!—Servicios fuera de horario de entrega →
            <ServicesOutOfWindow>2</ServicesOutOfWindow>
            <!—Servicios realizados →
            <TotalServices>4</TotalServices>
         </VehiclePlanStatistics>
         <VehicleServicePlan>
            <Service start=”2011-02-01T09:00:00” end=”2011-02-01T09:30:00”>
               <ID>1</ID>
               <Name>1</Name>
               <Duration>00:30:00</Duration>
               <Location>
```

```
            <Address>Botet I Sissó</Address>
            <HouseNumber>6</HouseNumber>
            <City>Girona</City>
            <PostalCode>17003</PostalCode>
            <Region>Girona</Region>
            <Country>España</Country>
            <Coord x="-12.5433333" y="40.5894444" srs="EPSG:4326"/>
            <GeocodeLevel>ADR</GeocodeLevel>
          </Location>
          <Priority>1</Priority>
          <Windows>
            <Window start="2011-02-01T09:00:00"
                end="2011-02-01T14:00:00"/>
            <Window start="2011-02-01T16:00:00"
                end="2011-02-01T17:00:00"/>
          </Windows>
          <Requirement>1,2</Requirement>
          <Comments>Servicio ya cobrado.</Comments>
        </Service>
        <WorkBreak start="2011-02-01T13:00:00"
            end="2011-02-01T14:00:00"/>
        <Service start="2011-02-01T15:00:00" end="2011-02-01T16:30:00">
          <ID>2</ID>
          <Name>1</Name>
          <Duration>00:30:00</Duration>
          <Location>
            <Address>Botet I Sissó</Address>
            <HouseNumber>6</HouseNumber>
            <City>Girona</City>
            <PostalCode>17003</PostalCode>
            <Region>Girona</Region>
            <Country>España</Country>
            <Coord x="-12.5433333" y="40.5894444" srs="EPSG:4326"/>
            <GeocodeLevel>ADR</GeocodeLevel>
          </Location>
          <Priority>1</Priority>
          <Windows>
            <Window start="2011-02-01T09:00:00"
                end="2011-02-01T14:00:00"/>
            <Window start="2011-02-01T16:00:00"
                end="2011-02-01T17:00:00"/>
          </Windows>
          <Requirement>1,2</Requirement>
          <Comments>Servicio ya cobrado.</Comments>
        </Service>
      </VehicleServicePlan>
    </VehiclePlan>
    <VehiclePlan>
      <Vehicle>
        <ID>2</ID>
        <Name>1</Name>
        <StartTimeWorkday>2011-02-01T08:00:00</StartTimeWorkday>
        <EndTimeWorkday>2011-02-01T16:00:00</EndTimeWorkday>
        <Location>
          <Address>Botet I Sissó</Address>
          <HouseNumber>6</HouseNumber>
          <City>Girona</City>
          <PostalCode>17003</PostalCode>
          <Region>Girona</Region>
          <Country>España</Country>
          <Coord x="-12.5433333" y="40.5894444" srs="EPSG:4326"/>
          <GeocodeLevel>ADR</GeocodeLevel>
        </Location>
        <CostKm>0.2</CostKm>
        <CostHour>20</CostHour>
        <MaxContDrivingTime>04:30:00</MaxContDrivingTime>
        <MandatoryBreak>00:45:00</MandatoryBreak>
        <WorkBreaks>
          <WorkBreak start="2011-02-01T09:00:00"
              end="2011-02-01T10:00:00"/>
          <WorkBreak start="2011-02-01T13:00:00"
              end="2011-02-01T14:00:00"/>
        </WorkBreaks>
        <Capability>1,2,3,4</Capability>
        <Comments></Comments>
      </Vehicle>
      <VehiclePlanStatistics>
```

```
        <!—Tiempo total conducción →
        <TotalDrivingTime>10:02:50</TotalDrivingTime>
        <!—Tiempo total en servicio →
        <TotalServiceTime>32:02:50</TotalServiceTime>
        <!—Tiempo total de descanso →
        <TotalBreakTime>10:02:50</TotalBreakTime>
        <!—Tiempo total de trabajo extra →
        <TotalOvertimeWork>00:00:00</TotalOvertimeWork>
        <!—Km totales →
        <TotalKm>321.32</TotalKm>
        <!—Coste total →
        <TotalCosts>213</TotalCosts>
        <!—Costes de peajes →
        <TollCosts>12.32</TollCosts>
        <!—Servicios fuera de horario de entrega →
        <ServicesOutOfWindow>2</ServicesOutOfWindow>
        <!—Servicios realizados →
        <TotalServices>4</TotalServices>
      </VehiclePlanStatistics>
      <VehicleServicePlan>
          <Begin start="2011-02-01T09:00:00">
        <Service start="2011-02-01T09:00:00" end="2011-02-01T09:30:00">
          <ID>1</ID>
          <Name>1</Name>
          <Duration>00:30:00</Duration>
          <Location>
            <Address>Botet I Sissó</Address>
            <HouseNumber>6</HouseNumber>
            <City>Girona</City>
            <PostalCode>17003</PostalCode>
            <Region>Girona</Region>
            <Country>España</Country>
            <Coord x="-12.5433333" y="40.5894444" srs="EPSG:4326"/>
            <GeocodeLevel>ADR</GeocodeLevel>
          </Location>
          <Priority>1</Priority>
          <Windows>
            <Window start="2011-02-01T09:00:00"
                end="2011-02-01T14:00:00"/>
            <Window start="2011-02-01T16:00:00"
                end="2011-02-01T17:00:00"/>
          </Windows>
          <Requirement>1,2</Requirement>
          <Comments>Servicio ya cobrado.</Comments>
        </Service>
        <WorkBreak start="2011-02-01T10:00:00"
            end="2011-02-01T11:00:00"/>
        <Service start="2011-02-01T11:00:00" end="2011-02-01T11:30:00">
          <ID>2</ID>
          <Name>1</Name>
          <Duration>00:30:00</Duration>
          <Location>
            <Address>Botet I Sissó</Address>
            <HouseNumber>6</HouseNumber>
            <City>Girona</City>
            <PostalCode>17003</PostalCode>
            <Region>Girona</Region>
            <Country>España</Country>
            <Coord x="-12.5433333" y="40.5894444" srs="EPSG:4326"/>
            <GeocodeLevel>ADR</GeocodeLevel>
          </Location>
          <Priority>1</Priority>
          <Windows>
            <Window start="2011-02-01T09:00:00"
                end="2011-02-01T14:00:00"/>
            <Window start="2011-02-01T16:00:00"
                end="2011-02-01T17:00:00"/>
          </Windows>
          <Requirement>1,2</Requirement>
          <Comments>Servicio ya cobrado.</Comments>
        </Service>
      </VehicleServicePlan>
    </VehiclePlan>
  </VehiclePlans>
  <PendingServices>
    <Service>
      <ID>3</ID>
```

```
            <Name>1</Name>
            <Duration>00:30:00</Duration>
            <Location>
               <Address>Botet I Sissó</Address>
               <HouseNumber>6</HouseNumber>
               <City>Girona</City>
               <PostalCode>17003</PostalCode>
               <Region>Girona</Region>
               <Country>España</Country>
               <Coord x="-12.5433333" y="40.5894444" srs="EPSG:4326"/>
               <GeocodeLevel>ADR</GeocodeLevel>
            </Location>
            <Priority>1</Priority>
            <Windows>
               <Window start="2011-02-01T09:00:00" end="2011-02-01T14:00:00"/>
               <Window start="2011-02-01T16:00:00" end="2011-02-01T17:00:00"/>
            </Windows>
            <Requirement>1,2</Requirement>
            <Comments>Servicio ya cobrado.</Comments>
         </Service>
         <Service>
            <ID>4</ID>
            <Name>1</Name>
            <Duration>00:30:00</Duration>
            <Location>
               <Address>Botet I Sissó</Address>
               <HouseNumber>6</HouseNumber>
               <City>Girona</City>
               <PostalCode>17003</PostalCode>
               <Region>Girona</Region>
               <Country>España</Country>
               <Coord x="-12.5433333" y="40.5894444" srs="EPSG:4326"/>
               <GeocodeLevel>ADR</GeocodeLevel>
            </Location>
            <Priority>1</Priority>
            <Windows>
               <Window start="2011-02-01T09:00:00" end="2011-02-01T14:00:00"/>
               <Window start="2011-02-01T16:00:00" end="2011-02-01T17:00:00"/>
            </Windows>
            <Requirement>1,2</Requirement>
            <Comments>Servicio ya cobrado.</Comments>
         </Service>
         <End start="2011-02-01T09:00:00">
      </PendingServices>
   </Plan>
</Response>
```

5 Distance & time Matrix

5.1 Matrix request

Can use GET or POST, with these parameters:

cmd	Always use **matrix.**
clientid	Cercalia client ID. (Mandatory)
client	Client code. (Optional)
srs	Spatial reference system (coordinates system)
routeweight	Type of routing optimization (time, distance, ...)
network	Routing network (optional)
origs	Origin coordinates [X0,Y0],[X1,Y1], ... [Xn,Yn]
dests	Destination coordinates [X0,Y0],[X1,Y1], ... [Xn,Yn]

If dests is not used, matrix is calculated from all origins.

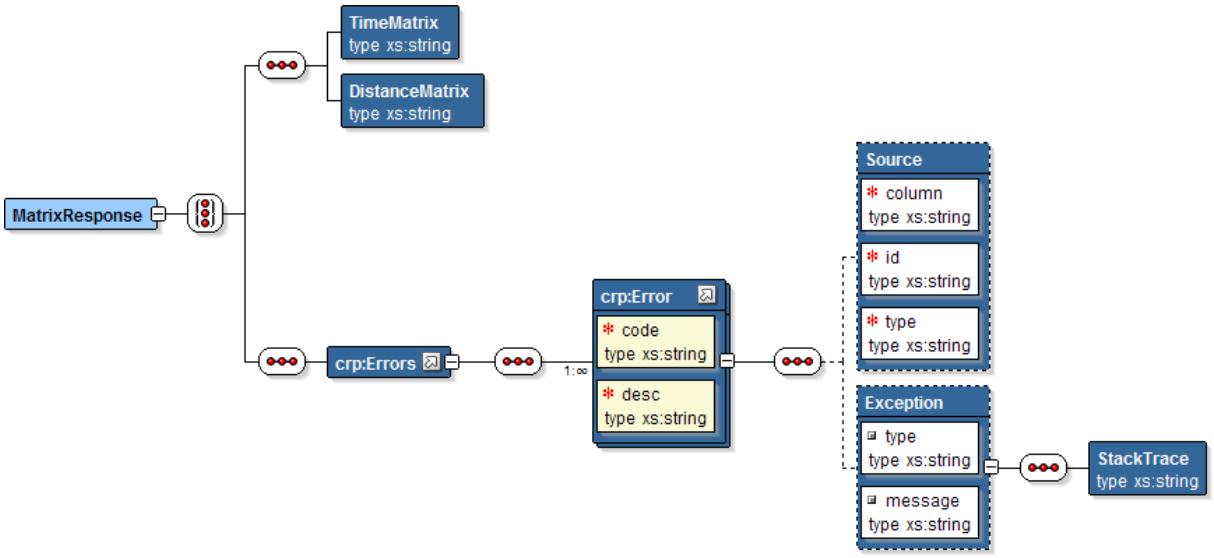
For requests larger than 256 characters, send it via POST.

Request example:

cmd=matrix&clientid=cli&srs=EPSG:4326&routeweight=time&origs=[2.00854781, 41.31808218],[1.69725889,41.34549167]

5.2 Matrix response

Matrix response is a XML with this structure:




```
<MatrixResponse>
        <TimeMatrix>0|232|321|0</TimeMatrix >
        <DistanceMatrix>0|5829227|5856304|0</ DistanceMatrix >
        or
        <Errors>
                <Error code="E10001" desc="Falta ID de distribuidor/cliente." />
        </Errors>
</MatrixResponse>
```

TimeMatrix contains the driving time to go from one coordinate to another expressed in seconds.

For calculating the driving time from Pi coordinate to Pj coordinate (where i and j indicate the coordinate order) apply the formula Pi * Number of coordinates + Pj.

DistanceMatrix contains the driving distange to go from one coordinate to another expressed in meters.

For calculating the driving distance from Pi coordinate to Pj coordinate (where i and j indicate the coordinate order) apply the formula Pi * Number of coordinates + Pj.

In case of errors, you will receive the **Errors** tag instead of **TimeMatrix** tag and **DistanceMatrix** tag.

Annex 1: Error codes

Error codes list:

Code	Description
E00000	No error
E10001	Need client ID
E10002	Unknown client ID
E20003	Insufficient points
E20004	Uncontracted functionality
E20005	Unauthorized mobile
E30001	Server not accessible
E30002	Internal server error.
E30003	Server busy
E30004	Cercalia connection error.
E30005	Resource unavailable
E30106	Error calculating matrix.
E40001	Cartography component not found
E40002	No way
E50001	Need required parameter.
E50002	Incorrect parameter value or out of range.
E50008	Wrong request format
E50009	XML validation error.
E60001	No services to plan
E60002	No vehicles for planning
E60003	There are no jobs with this ID