API Сириус Навигатора

Web-сервис для интеграции со сторонними приложениями обеспечивает возможность передачи обработанных данных в произвольные приложения. Сервис использует протокол SOAP. Для подключения к Web-сервису используйте адрес: http://_your_IP_:8000/SiriusAPI, где _your_IP_ - адрес, по которому запущен сервис. Описание сервиса в стандарте WSDL доступно по адресу: http://_your_IP_:8000/SiriusAPI?wsdl

Описание методов сервиса:

GetVehicles — получение списка ТС.
Входные данные: нет.
Выходные данные:

- vehicles - массив, содержащий список объектов.

Структура vehicles:
Каждый элемент массива(vehicle) имеет следующие поля:

- vehicleId — идентификатор ТС, тип - int;
- model – марка ТС, тип - string;
- vehicleRegistrationNumber — государственный номер ТС, тип - string;
- fleetNumber — гаражный номер ТС, тип - string.

GetZones — получение списка географических зон.
Входные данные: нет.
Выходные данные:

- zones — массив, содержащий список геозон.

Структура zones:
Каждый элемент массива (zone) имеет следующие поля:

- zoneId — идентификатор геозоны, тип - int;
- name – название геозоны, тип — string.

GetDrivers — получение списка водителей.
Входные данные: нет.
Выходные данные:

- drivers — массив, содержащий список водителей.

Структура drivers:
Каждый элемент массива (driver) имеет следующие поля:

- driverId — идентификатор водителя, тип — int;
- lastName — фамилия, тип — string;
- firstName – имя, тип — string;
- patronymic - отчество, тип — string.

GetGroupsOfDigitalSensors — получение списка групп дискретных датчиков.
Входные данные: нет.
Выходные данные:

- groups — массив, содержащий список групп.

Структура groups:
Каждый элемент массива (group) имеет следующие поля:

- groupId — идентификатор группы, тип — int;
- name — наименование группы, тип — string.

GetStatesOfDigitalSensor – получение периодов активности дискретных датчиков, входящих в указанные группы.

Входные данные:

- vehicleId – идентификатор ТС, тип - int;
- groupIds – идентификаторы групп датчиков, тип - массив int;
- from – время начала периода запроса, тип - DataTime;
- till – время конца периода запроса, тип - DataTime.

Выходные данные:

- vehicleId – идентификатор ТС, тип - int;
- states - массив, содержащий список активностей.

Структура states:

Каждый элемент массива(state) имеет следующие поля:

- groupId – идентификатор группы датчиков, тип — int;
- sensorName – наименование датчика, тип — string;
- onTimeOOR – валидность поля onTime (поле не валидно при начале активности ранее времени начала запроса), тип — bool;
- onTime – время начала активности, тип — DataTime;
- offTimeOOR – валидность поля offTime (поле не валидно при окончании активности позднее времени окончания запроса запроса), тип — bool;
- offTime – время окончания активности, тип — DataTime.

GetVisits – получение посещений геозон.

Входные данные:

- vehicleId – идентификатор ТС, тип - int;
- zoneIds – идентификаторы геозон, тип - массив int;
- from – время начала периода запроса, тип - DataTime;
- till – время конца периода запроса, тип - DataTime.

Выходные данные:

- vehicleId – идентификатор ТС, тип - int;
- visits - массив, содержащий список посещений геозон.

Структура visit:

Каждый элемент массива(visit) имеет следующие поля:

- zoneId – идентификатор геозоны, тип — int;
- arrivalOOR – валидность поля arrival (поле не валидно при въезде ранее времени начала запроса), тип — bool;
- arrival – время въезда, тип — DataTime;
- departureOOR – валидность поля departureOOR (поле не валидно при выезде позднее времени окончания запроса запроса), тип — bool;
- departure – время выезда, тип — DataTime.

GetFuelEvents — получение событий по топливу ТС.

Входные данные:

- vehicleId – идентификатор ТС, тип - int;
- beginTime – дата начала периода запроса, тип - DataTime;
- endTime – дата конца периода запроса, тип - DataTime.

Выходные данные:

- vehicleId – идентификатор ТС, тип - int;
- events - массив, содержащий список событий.

Структура events:

Каждый элемент массива(fuelEvent) имеет следующие поля:

- time – время совершения события, тип - DataTime;
- volume - объём заправленного (volume > 0) или слитого (volume < 0) топлива, тип — double;
- address - адрес места заправки (слива), тип - string.

GetConvergences – получение списка сближений объектов.

Входные данные:

- vehicleId – идентификатор объекта, тип - int;
- unitIds – идентификаторы субъектов, тип - массив int;
- distance – максимальное расстояние, тип - int;
- from – время начала периода запроса, тип - DataTime;
- till – время конца периода запроса, тип - DataTime.

Выходные данные:

- vehicleId – идентификатор объекта, тип - int;
- convergences - массив, содержащий список сближений.

Структура convergence:

Каждый элемент массива(convergence) имеет следующие поля:

- unitId – идентификатор субъекта, тип — int;
- arrivalOOR – валидность поля arrival (поле не валидно при сближении ранее времени начала запроса), тип — bool;
- arrival – время сближения, тип — DataTime;
- departureOOR – валидность поля departure (поле не валидно при удалении позднее времени окончания запроса запроса), тип — bool;
- departure – время удаления, тип — DataTime;
- minDistance – минимальная дистанция сближения, тип — int.

GetDriverBindings – получение списка регистраций водителей на ТС.

Входные данные:

- vehicleId – идентификатор ТС, тип - int;
- driverId – идентификатор водителя или 0, если водитель не известен, тип - int;
- from – время начала периода запроса, тип - DataTime;
- till – время конца периода запроса, тип - DataTime.

Выходные данные:

- bindings - массив, содержащий список регистраций.

Структура binding:

Каждый элемент массива(binding) имеет следующие поля:

- vehicleId – идентификатор ТС, тип — int;
- driverId – идентификатор водителя, тип — int;
- bindingTimeOOR – валидность поля bindingTime (поле не валидно при регистрации ранее времени начала запроса), тип — bool;
- bindingTime – время регистрации, тип — DataTime;
- unbindingTimeOOR – валидность поля unbindingTime (поле не валидно при разрегистрации позднее времени окончания запроса запроса), тип — bool;
- unbindingTime – время разрегистрации, тип — DataTime.

GetMileage – получение пробега по одному ТС.

Входные данные:

- vehicleId – идентификатор ТС, тип - int;
- beginTime – дата начала периода запроса, тип - DataTime;
- endTime – дата конца периода запроса, тип - DataTime.

Выходные данные:

- mileage — пробег ТС за запрашиваемый период, тип - double.

GetHistoryOfMovement — получение истории перемещения ТС.

Входные данные:

- vehicleId – идентификатор ТС, тип - int;
- beginTime – дата начала периода запроса, тип - DataTime;
- endTime – дата конца периода запроса, тип - DataTime;
- groupInterval – интервал группировки данных в секундах, тип - int;
- minParkingDuration – минимальная длительность стоянки в секундах, тип - int.

Выходные данные:

- vehicleId – идентификатор ТС, тип - int;
- periods – массив, передающий состояние ТС.

Структура period:

Каждый элемент массива (period) имеет следующие поля:

- address – адрес местоположения ТС на момент времени, определённый в поле beginTime, тип — string;
- longitude – долгота, тип — double;
- latitude – широта, тип — double;
- beginTime — время начала периода, тип - DataTime;
- course – курс ТС, тип - ushort;
- courseSpecified – флаг, показывающий есть ли данные в поле course, тип - bool;
- duration – содержит продолжительность стоянки, если type = "Parking", иначе равно 0, тип - int;
- durationSpecified - флаг, показывающий есть ли данные в поле duration, тип - bool;
- race – пробег ТС на момент времени, определённый в поле beginTime, тип - double;
- raceSpecified - флаг, показывающий есть ли данные в поле race, тип - bool;
- speed – скорость ТС, тип - float;
- speedSpecified - флаг, показывающий есть ли данные в поле speed, тип - bool;
- type – статус ТС, может иметь следующие значения - "Parking", "Moving", "NoData", тип — string.

GetFuelConsumption – получение расхода топлива по ТС.

Входные данные:

- vehicleId – идентификатор ТС, тип - int;
- beginTime – дата начала периода запроса, тип - DataTime;
- endTime – дата конца периода запроса, тип - DataTime.

Выходные данные:

- fuelConsumption — расход топлива за запрашиваемый период, тип - double;
- fuelConsumptionSpecified – флаг, показывающий определено ли поле fuelConsumption, тип - bool.

Возможны следующие варианты работы метода:

- если не установлены датчики топлива, то значение fuelConsumption равно 0, а поле fuelConsumptionSpecified принимает значение false;
- если на ТС установлен или датчик расхода топлива или датчик объёма топлива (датчик суммарного объёма топлива в ТС), то fuelConsumption отображает показание этого датчика, а поле fuelConsumptionSpecified принимает значение true;
- если установлены и датчик расхода топлива и датчик объёма топлива (датчик суммарного объёма топлива в ТС), то fuelConsumption отображает показание датчика расхода топлива как наиболее точного. Поле fuelConsumptionSpecified при этом принимает значение true;

GetTripReport — получение отчёта о поездках ТС.

Под поездкой понимается движение ТС от момента выезда из базы до момента возвращения на базу. При указании нескольких баз поездка может начаться в одной базе и завершиться в другой. Поездки, начавшиеся до начала периода запроса и/или закончившиеся после в отчёт не включаются.

Входные данные:

- vehicleId – идентификатор ТС, тип — int;
- idsOfBases — список идентификаторов геозон, рассматриваемых как базы, тип — массив int.
- idsOfCheckpoints — список идентификаторов геозон, рассматриваемых как контрольные точки, тип — массив int.
- beginTime – дата начала периода запроса, тип - DataTime;
- endTime – дата конца периода запроса, тип - DataTime;

Выходные данные:

- vehicleId – идентификатор ТС, тип - int;
- events – массив, содержащий события поездок.

Структура events:

Каждый элемент массива (tripEvent) имеет следующие поля:

- tripId – номер поездки, тип — int.
- type – событие, может иметь следующие значения: "Departure" – выезд из базы, "Entry" – въезд в контрольную точку, "Exit" – выезд из контрольной точки, "Arrival" – прибытие на базу, тип - string.
- checkpointId – идентификатор геозоны, связанной с событием, тип - int;
- time — время события, тип - DataTime;
- mileage – пробег ТС на момент времени, определённый в поле time, тип — double.

GetFuelingOperations — получение списка топливораздач по объекту.

Входные данные:

- vehicleId – идентификатор ТС, тип - int;
- beginTime – дата начала периода запроса, тип - DataTime;
- endTime – дата конца периода запроса, тип - DataTime.

Выходные данные:

- controllerId – идентификатор ТС, тип - int;
- operations - массив, содержащий список топливораздач.

Структура operations:

Каждый элемент массива(operation) имеет следующие поля:

- time – время совершения события, тип - DataTime;
- volume - объём выданного топлива, тип — double.