Omnicomm Online

Omnicomm Online permite al usuario controlar el funcionamiento de los vehículos y conductores por medio de los informes que provienen de ellos. Para acceder a Omnicomm Online, solamente se necesita un ordenador personal conectado a Internet.

Omnicomm Online cuenta con herramientas integradas con funciones especializadas para recopilar los datos procesados y usarlos en los documentos de contabilidad y sistemas de monitoreo de flota.

Este manual describe el funcionamiento de las herramientas integradas y la integración con sistemas de terceros.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api

Last update: **2026/11/16 13:41**

Conexión

Para obtener la dirección de conexión a los servicios en línea, el cliente debe comunicarse con el equipo de soporte técnico de Omnicomm.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/connection

Last update: **2026/11/16 13:41**

SOAP

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/definition

Last update: **2026/11/16 13:41**

Errores

Lista de errores devueltos:

0: No errors – no hay errores

1: Signing in failed – usuario o contraseña introducida incorrecta

2: Authorization required – se require una autorización para acceder a los datos

3: Dead session number – se ha expirado la sesión, se necesita volver a dar autorización

4: Bad interval – se ha introducido un intervalo de tiempo inadecuado

5: Bad object – no existe objeto con este identificador

6: Admin login – alguien está tratando de iniciar sesión como Usuario Admin

7: Unusable object – no se puede calcular el valor para el objeto con este identificador

8: Bad event type – no existe tipo de evento con este identificador

9: Access denied – no hay autorización para accesar este objeto

10: Data not found – no existen datos para los valores de entrada correspondientes

11: Blocked interval – el intervalo solicitado contiene periodos de bloqueo de datos

12: Bad object type – el tipo de objeto especificado no existe

13: Invalid format – el formato es incorrecto

14: Undefined error – el error no está especificado

15: 404 – página no encontrada

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/definition/error

Last update: **2026/11/16 13:41**

Ejemplo del cliente de trabajo

Importe de Interfaces

wsimport -d bin -s src http://demo.omnicomm.ru:8000/AnalyticalServer/ws?wsdl

Código Java:

Es necesario cambiar las líneas «usuario» y «contraseña» por los valores verdadores.

```
package ru.omnicomm.test.client;

import ru.omnicomm.analyticalserver.*;

import java.net.MalformedURLException;

import java.net.URL;

import java.util.List;

public class  ExampleClient {
    public static void main(String[] args) throws MalformedURLException {
        AnalyticalServer = new AnalyticalServer(new
URL("http://demo.omnicomm.ru:8000/AnalyticalServer/ws?wsdl"));

        AnalyticalServerWS port =
AnalyticalServer.getAnalyticalServerPort();

        AuthResponseEntry auth = port.signIn("user", "pass");

        String sessionId = auth.getsessionId ();

        System.out.println("auth sessionId: " + sessionId);

        ObjectSetResponseEntry objects = port.getObjectSet(sessionId);
        List<Vehicle> vehicles = objects.getVehicleList();

        for (Vehicle vehicle : vehicles) {
            System.out.printf("vehicle: %d / %s\n", vehicle.getVehicleID(),
vehicle.getRegNmb());
        }

    }
}
```

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/definition/example

Last update: **2026/11/16 13:41**

Lista de métodos SOAP

Se utiliza el formato UNIXTIME (en segundos) para la fecha y el tiempo en que los datos fueron transmitidos. A continuación se enumeran las unidades de los demás parámetros.

signIn – autorización

Valores de entrada
String login – nombre de usuario en el sistema **String password** – contraseña en el sistema

Valores de retorno
Boolean **status** – verdadero/falso verdadero en el caso de una autorización exitosa String **sessionId** – en caso de una autenticación exitosa, el identificador de sesión (mínimo de 16 carácteres) Unixtimestamp **dateTimeEnd** – en caso de una autenticación exitosa, el tiempo de la terminación de sesión (el tiempo después de que debería iniciar sesión de nuevo) String **error** – mensaje de error en caso de autorización inadecuada (usuario y contraseña introducida incorrectamente, o formato de datos incorrecto)

getObjectSet – lista de objetos

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización

Valores de retorno
Boolean **status** – verdadero/falso verdadero en el caso de una autorización exitosa String **error** – mensaje de error en caso de que ocurra un error Dataset **objects** – lista de vehículos disponibles al usuario - dueño de la sesión registrada (es decir, solo aquellos objetos a los que el usuario tiene derecho ver): Integer **id** – identificador del objeto (la misma que el identificador de la unidad) String **objectName** – nombre del vehículo String **objectType** – tipo de vehículo String **GarageNumber** – número de garage

getSmoothedFuel – nivel de combustible suavizado para el periodo

Valores de entrada
String **sessionId** – identificador de sesión obtenido durante la autorización Integer **objectId** – identificador del vehículo/unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo

Valores de retorno
Boolean **status** – verdadero/falso verdadero en el caso de una operación exitosa Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Dataset **fuel** – conjunto de datos (todos los datos del archivo para el periodo seleccionado): Unixtimestamp **timeStamp** – tiempo del nivel de combustible registrado Double **smothedFuel** – valor suavizado de combustible, litros, precisión de hasta 0.1 l String **error** – mensaje de error en caso de ocurrir un error

getFuelConsumption – consumo de combustible para el periodo

Valores de entrada
String **sessionId** - identificador de la sesión obtenido durante la autorización Integer **objectId** - identificador del vehículo / unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp ** timeEnd** (segundos) - tiempo final del intervalo

Valores de retorno
Boolean **status** - verdadero/falso verdadero en el caso de una autorización exitosa. Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp ** timeEnd** (segundos) – tiempo final del intervalo Double **fuelConsumption** – consumo de combustible para el periodo, litros, precisión de hasta 0.1 l String **error** – mensaje de error en caso de ocurrir un error

getEvents – lista de eventos

Valores de entrada
String sessionId – identificación de sesión obtenida durante la autorización **Integer objectId** – identificador del vehículo/unidad. Parámetro opcional, si el identificador no está presente, devuelve los datos para todos los vehículos **Integer type** – tipo de evento, requerido **Unixtimestamp timeBegin** (segundos) – tiempo de inicio del intervalo **Unixtimestamp timeEnd** (segundos) - tiempo final del intervalo

Valores de retorno
Boolean **status** – verdadero/falso verdadero en el caso de una autorización exitosa. String **error** - mensaje de error en caso de que ocurra un error Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Dataset **objectEvents** – conjunto de datos para el evento: Unixtimestamp **timeStamp** – fecha y tiempo del evento Integer **objectId** – identificador del vehículo / unidad String **type** – tipo de evento String **parameters** – parámetros del evento String **eventAddress** – difrección del evento, si está disponible String **iButton** – código iButton en HEX. Solamente para eventos tales como 'Conductor' String **name** – nombre de la geocerca. Solamente para la entrada y salida de los eventos de Geocerca

getMileage – kilometraje del periodo

Valores de entrada
String **sessionId** – identificador de sesión durante la autorización Integer **objectId** – identificador del vehículo/unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) - tiempo final del vehículo

Valores de retorno
Boolean **status** - – verdadero/falso verdadero en el caso de una operación exitosa. Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Double **Mileage** – kilometraje en km para el intervalo en específico, precisión de 0.1 km String **error** – mensaje de error en caso de que ocurra un error

getEngineOnTime – tiempo de funcionamiento del motor para el periodo

Valores de entrada
String **sessionId** - identificador de sesión obtenido durante la autorización Integer **objectId** - identificador del vehículo/unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) - tiempo final del intervalo

Valores de retorno
Boolean **status** - verdadero/falso verdadero en el caso de una operación exitosa Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Double **engineOnTime** – tiempo de operacion total del motor, en segundos String **error** – mensaje de error en caso de que ocurra un error

getVehiclesState – estado actual del vehículo

Valores de entrada
String **sessionId** – identificador de sesión obtenida durante la autorización VehiclesType **vehicles** – lista de identificación del vehículo

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurra un error vehicleStatesType **states** – lista de parámetros que describen cada estado del vehículo

getMileageSpeedExcess – kilometraje con exceso de velocidad para el periodo

Valores de entrada
String **sessionId** – identificador de la sesión obtenido durante la autorización Integer **objectId** – identificador del vehículo / unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo

Valores de retorno
Boolean **status** - verdadero/falso verdadero en el caso de una autorización exitosa. Unixtimestamp ** timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Double **mileageSpeedExcess** – kilometraje con velocidad en km para el intervalo en específico, precisión 0.1 km String **error** – mensaje de error en caso de que ocurra un error

getMovementTime – tiempo en movimiento para el periodo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Integer **objectId** – identificador del vehículo/unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo

Valores de retorno
Boolean **status** - verdadero/falso verdadero en el caso de una autorización exitosa. Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp ** timeEnd** (segundos) – tiempo final del intervalo Double **movementTime** – tiempo del movimiento, segundos String **error** – mensaje de error en caso de que ocurra un error

getEngineOnTimeInMovement – tiempo de funcionamiento del motor en movimiento para el periodo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Integer **objectId** – identificador del vehículo/unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo

Valores de retorno
Boolean **status** - verdadero/falso verdadero en el caso de una operación exitosa Unixtimestamp ** timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp ** timeEnd** (segundos) – tiempo final del intervalo Double **engineOnTimeInMovement** – tiempo de funcionamiento del motor durante el movimiento, segundos String **error** – mensaje de error en caso de ocurrir un error

getEngineOnTimeWithoutMovement – tiempo de funcionamiento sin movimiento para el periodo

Valores de entrada
String **sessionId** – identificador de la sesión obtenido durante la autorización Integer **objectId** – identificador del vehículo / unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp ** timeEnd** (segundos) – tiempo final del intervalo

Valores de retorno
Boolean **status** – verdadero/falso verdadero en el caso de una operación exitosa. Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Double **engineOnTimeWithoutMovement** – tiempo de funcionamiento del motor sin movimiento, segundos String **error** – mensaje de error en caso de que ocurra un error

getEngineOffTime – tiempo de inactividad del motor para el periodo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Integer **objectId** – identificador del vehículo/unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo
Valores de retorno
Boolean **status** - verdadero/falso verdadero en el caso de una operación exitosa Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Double **engineOffTime** – tiempo de apagado del motor durante el periodo, segundos String **error** – mensaje de error en caso de que ocurra un error

getFuelConsumptionInMovement – consumo de combustible durante el movimiento del periodo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Integer **objectId** – identificador del vehículo/unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo
Valores de retorno
Boolean **status** - verdadero/falso verdadero en el caso de una operación exitosa Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Double **fuelConsumptionInMovement** – consumo de combustible durante el movimiento para el periodo, litros, precisión 0.1 litros String **error** – mensaje de error cuando ocurra un error

getFuelConsumptionWithoutMovement – consumo de combustible sin movimiento para el periodo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Integer **objectId** – identificador del vehículo/unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo
Valores de retorno
Boolean **status** – verdadero/falso verdadero en el caso de una operación exitosa. Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Double **fuelConsumptionWithout Movement** – consumo de combustible sin movimiento para el periodo, litros, precisión 0.1 l String **error** – mensaje de error cuando ocurra un error

getFuelConsumptionInMotohour – consumo de combustible por hora de funcionamiento del motor

Valores de entrada
String **sessionId** – identificador de sesión obtenida durante la autorización Integer **objectId** – identificador del vehículo/unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo

Valores de retorno
Boolean **status** - verdadero/falso verdadero en el caso de una operación exitosa Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Double **fuelConsumptionIn Motohour** – consumo de combustible promedio para la hora de motor del periodo, litros, precisión 0.1 l String **error** – mensaje de error cuando ocurra un error

getFuelAtTime – nivel de combustible en un momento determinado

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Integer **objectId** – identificador del vehículo/unidad Unixtimestamp **time** (segundos) – momento del tiempo

Valores de retorno
Boolean **status** - verdadero/falso verdadero en caso de una operación exitosa Unixtimestamp **time** (segundos) – momento del tiempo Double **fuelAtTime** – nivel de combustible en un momento dado, litros, precisión 0.1 l String **error** – mensaje de error en caso de que ocurriera un error

getUserNotificationsByPeriod – notificaciones de usuario acumuladas hasta el inicio del periodo

Valores de entrada
String **sessionId** – identificador de sesión obtenido durante la autorización Unixtimestamp **timeBegin** – tiempo de inicio del intervalo (UTC), segundos Unixtimestamp **timeEnd** – tiempo final del intervalo (UTC), segundos. Si no está definido, tiempo final = tiempo del sistema de ejecución de solicitud por el servidor Integer **page** – número de página solicitado. Si no está definido, se retorna la primera página que contenga los registros perPage (por página) Integer **perPage** – cantidad de registros por página, si no está definido, la cantidad no está limitada

Valores de retorno
Boolean **status** – estado de funcionamiento. Verdadero en caso de una operación exitosa. String **error** – mensaje de error en caso de que ocurra un error Unixtimestamp **timeBegin** – tiempo de inicio del intervalo (UTC), segundos Unixtimestamp **timeEnd** – tiempo final del intervalo (UTC), segundos userNotificationsType **userNotifications** – serie de conjuntos de parámetros de retorno Integer **notificationsCount** – número total de notificaciones para todas las páginas. Si no se encuentra notificación, se devuelve el código de error 10

getVisitedGeozonesByPeriod – geocercas visitadas hasta el inicio del periodo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Unixtimestamp **timeBegin** – tiempo de inicio del intervalo (UTC), segundos Unixtimestamp **timeEnd** – tiempo final del intervalo (UTC), segundos vehicleIdsType **vehicleId** – lista de identificaciones del vehículo. Si no existe, se utilizan todos los vehículos disponibles para la solicitud

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurra un error Unixtimestamp **timeBegin** – tiempo de inicio del intervalo (UTC), segundos Unixtimestamp **timeEnd** – tiempo final del intervalo (UTC), segundos visitedGeozone **geozoneVisits** – serie de conjunto de parámetros de geocercas visitadas

getVehiclesParams – lista de parámetros disponibles al usuario

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurra un error Vehicle **vehicles** – parámetros de vehículo que el usuario actual tiene derecho a consultar

signOut – término de la sesión

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurra un error String **sessionId** – identificación de sesión terminada

getActiveNotificationRules – perfiles de notificaciones activas

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurriera un error rulesType **rules** – parámetros para cada perfil de notificación

setDeviceIdToNotificationRules – asignación de perfiles de notificación al vehículo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización String **deviceId** – identificación del dispositivo String **deviceTypeId** – identificador del tipo de dispositivo rulesType **rules** – identificadores de perfiles de notificación que son necesarios asignar al vehículo

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurra un error

getFuelLevelsByTimeMoment – nivel de combustible en un tiempo determinado

Valores de entrada
String **sessionId** – identificador de sesión obtenido durante la autorización Unixtimestamp **timeMoment** – momento del tiempo (UTC), segundos vehicleAndTankIdsType **vehicleAndTankIds** – llista de identicaciones de vehículos y tanques de combustible. Si no hay una lista, se ejecutará la solicitud para todos los vehículos y tanques disponibles al usuario

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurriera un error Unixtimestamp **timeMoment** – momento del tiempo (UTC), segundos fuelData **fuelDataSet** – lista de parámetros para cada vehículo: int **vehicleId** – identificador del vehículo int **tankNumber** – número de tanque de combustible fuelLevelsType **fuelLevels** – datos de nivel de combustible activityPeriodsType **activityPeriods** – datos de funcionamiento del motor ignitionOffListType **ignitionOffList** – datos de apagado de ignición ignitionOnListType **ignitionOnList** – datos de encendido de ignición llsFailurePeriodsType **llsFailurePeriods** – datos de fallos del sensor de nivel de combustible

getFuelLevelsByPeriod – nivel de combustible registrado hasta el inicio del periodo de tiempo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Unixtimestamp **timeBegin** – tiempo de inicio del intervalo (UTC), segundos Unixtimestamp **timeEnd** – tiempo final del intervalo (UTC), segundos. Si no está indicado el tiempo final, el tiempo final del periodo = el tiempo de inicio del sistema del proceso de solicitud de parte del Servidor VehicleAndTankIdsType **vehicleAndTankIds** – lista de identicaciones de vehículos y tanques de combustible. Si no hay una lista, se ejecutará la solicitud para todos los vehículos y tanques disponibles al usuario Int **reduce** – raleo: 0 = no requerido 1 = requerido

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurra un error Unixtimestamp **timeBegin** – tiempo de inicio del periodo (UTC), segundos Unixtimestamp **timeEnd** – tiempo final del periodo (UTC), segundos fuelData **fuelDataSet** – lista de parámetros para cada vehículo: int **vehicleId** – identificador del vehículo int **tankNumber** – número de tanque de combustible fuelLevelsType **fuelLevels** – datos de nivel de combustible activityPeriodsType **activityPeriods** – datos de funcionamiento del motor ignitionOffListType **ignitionOffList** – datos de apagado de ignición ignitionOnListType **ignitionOnList** – datos de encendido de ignición llsFailurePeriodsType **llsFailurePeriods** – datos de fallos del sensor de nivel de combustible

getSmoothedFuelLevelsByPeriod – niveles de combustible suavizados por periodo de tiempo

Valores de entrada
String **sessionId** – identificador de sesión obtenida durante la autorización Integer **objectId** – identificador del objeto Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo vehicleAndTankIdsType **vehicleAndTankIds** – lista de identificaciones de los vehículos y tanques de combustible. Si no hay una lista, se ejecutará la solicitud para todos los vehículos y tanques disponibles al usuario Int **reduce** – raleo: 0 = no requerido 1 = requerido

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurra un error Unixtimestamp **timeBegin** – tiempo de inicio del periodo (UTC), segundos Unixtimestamp **timeEnd** – tiempo final del periodo (UTC), segundos fuelData **fuelDataSet** – lista de parámetros para cada vehículo: int **vehicleId** – identificador del vehículo int **tankNumber** – número de tanque de combustible fuelLevelsType **fuelLevels** – datos de nivel de combustible activityPeriodsType **activityPeriods** – datos de funcionamiento del motor ignitionOffListType **ignitionOffList** – datos de apagado de ignición ignitionOnListType **ignitionOnList** – datos de encendido de ignición llsFailurePeriodsType **llsFailurePeriods** – datos de fallos del sensor de nivel de combustible

getRefuelingsAndDrainsByPeriod – operaciones de drenaje/recarga de combustible hasta le inicio del periodo de tiempo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo vehicleAndTankIdsType **vehicleAndTankIds** – lista de identificaciones de los vehículos y tanques de combustible. Si no hay una lista, se ejecutará la solicitud para todos los vehículos y tanques disponibles al usuario Integer **page** – número de la página solicitada con datos Integer **perPage** – número de entradas por página; si no está presente, se tomará como ilimitada String **sortname** – campo donde se debe de acomodar los parámetros de retorno String **sortorder** – orden de calificación: **asc** – ascendente **desc** – decreciente

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurriera un error Unixtimestamp **timeBegin** – tiempo de inicio del intervalo (UTC), segundos Unixtimestamp **timeEnd** – tiempo final del intervalo (UTC), segundos Integer **entriesCounter** – número total de entradas por periodo de tiempo RefuelingsAndDrainsType **RefuelingsAndDrains** – lista de parámetros para cada vehículo

getVehiclesProfiles – perfiles de vehículo coincidentes con los identificadores de vehículo

Valores de entrada
String **sessionId** – identificador de sesión obtenida durante la autorización VehiclesType **vehicles** – lista de identificación del vehículo

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurra un error **Vehicles** – lista de parámetros para cada vehículo

getCurrentObjectState – estado actual del vehículo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Integer **objectId** – identificador del objeto

Valores de retorno
Boolean **status** – verdadero/falso verdadero en caso de una operación exitosa. String **error** - mensaje de error en caso de que ocurra un error String **lastGPS** – últimas coordinadas válidas. Contiene valores de latitud y longitud separados por un punto y coma Integer **lastGPSDir** – dirección de movimiento, grados de 0 a 359 Double **currentSpeed** – velocidad actual en un momento dado, en kph, precisión 0.1 kph Double **currentFuel** – nivel de combustible actual, en litros, precisión 0.1 l Boolean **currentIgn** – estado de ignición. Verdadero si la ignición está ENCENDIDA Boolean **speedExceed** – exceso del límite de velocidad. Verdadero en caso de límite de velocidad excedida Integer **lastGPSSat** – número de satélites con las últimas coordinadas válidas Double **currentInputValue** – valor actual de entrada universal. Atributos: Integer number – número UI, String name - nombre UI

getReportData – informe de equipo auxiliar durante el periodo, TPMS, IQFreeze

Valores de entrada
String **sessionId** – identificador de sesión obtenida durante la autorización Integer **objectId** – identificador del objeto Long **timeBegin** – tiempo de inicio del intervalo (UTC), segundos Long **timeEnd** – tiempo final del intervalo (UTC), segundos String **reportTemplateID** – identificador de plantilla de informe en Omnicomm Online. Posibles valores: addEquipment, TPMS, refState, refWork

Valores de retorno
Boolean **status** – verdadero / falso verdadero en caso de una operación exitosa. String **error** – mensaje de error en caso de que ocurra un error ReportDataType **reportData** – serie que incluye los datos de informe

getSEOnTime – tiempo de funcionamiento del equipo auxiliar correspondiente al periodo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Integer **objectId** – identificador del vehículo/unidad Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) - tiempo final del vehículo

Valores de retorno
Boolean **status** - verdadero/falso verdadero en el caso de una operación exitosa Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Double **sEOnTime** (segundos) – tiempo de ejecución del equipo auxiliar para cada UI conectado. Atributos: Integer number – número UI, String name - nombre UI

getStatisticsByPeriod – datos estadísticos del periodo

Valores de entrada
String **sessionId** – identificador de sesión obtenido durante la autorización Unixtimestamp **timeBegin** – tiempo de inicio del intervalo (UTC), segundos Unixtimestamp **timeEnd** – tiempo final del intervalo (UTC), segundos. Si no está definido, tiempo final = tiempo del sistema de ejecución de solicitud por el servidor. int **objectType** – tipo de objeto: 0=vehículo 1=conductor Si el tipo no existe, se devuelve el código de error 12. objectIdsType **objectIds** – conjunto del tipo objectIdsType, que contiene lista de parámetros objectId del tipo int. En caso de no existir, se realiza la solicitud para todos los objectId del tipo correspondiente, disponible al usuario. requiredStatParamsType **requiredStatParams** – lista de todos los subgrupos requeridos para los parámetros del informe de 'estadística'. Si la lista está vacía, se devuelven todos los subgrupos con todos los parámetros

Valores de retorno
movingAndWorkingParamsType **movementAndWorkingParams** – subgrupo de los parámetros de funcionamiento y movimiento del vehículo. Si la lista está vacía, se devuelven todos los parámetros del subgrupo, sino, solamente se devuelven los parámetros que están enumerados. Si no se transmite el subgrupo, no se devolverá el subgrupo. fuelParamsType **fuelParams** – subgrupo de parámetros para combustible, drenaje y recargas de combustible. Si la lista está vacía, se devuelven todos los parámetros del subgrupo, sino, solamente se devuelven los parámetros que están enumerados. Si no se transmite el subgrupo, no se devolverá el subgrupo. optionalEquipmentParamsType **optionalEquipmentParams** – subgrupo de parámetros del equipo auxiliar. Si la lista está vacía, se devuelven todos los parámetros del subgrupo, sino, solamente se devuelven los parámetros que están enumerados. Si no se transmite el subgrupo, no se devolverá el subgrupo. CANDataParamsType** CANDataParams** – subgrupo de parámetros CAN. Si la lista está vacía, se devuelven todos los parámetros del subgrupo, sino, solamente se devuelven los parámetros que están enumerados. Si el subgrupo no está transmitido, se devolverá el subgrupo entero. addDataParamsType **addDataParams** – subgrupo de parámetros adicionales estadísticos: TPMS, iQFreeze, etc.

getTracksByPeriod – rastreo del vehículo correspondiente al periodo

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Unixtimestamp **timeBegin** – tiempo de inicio del intervalo (UTC), segundos Unixtimestamp **timeEnd** – tiempo final del intervalo (UTC), segundos. Si no está indicado el tiempo final, el tiempo final del periodo = el tiempo de inicio del sistema del proceso de solicitud de parte del Servidor VehiclesType **vehicles** – lista de la identificación de vehículo Int **reduce** – raleo: 0 = no requerido 1 = requerido

Valores de retorno
Boolean **status** – estado de funcionamiento String **error** – mensaje de error en caso de que ocurra un error Unixtimestamp **timeBegin** – tiempo de inicio del intervalo (UTC), segundos Unixtimestamp **timeEnd** – tiempo final del intervalo (UTC), segundos trackDataSetType **trackDataSet** – parámetros de rastreo de cada vehículo: **trackPoint** – parámetros de rastreo por un solo vehículo unixtimestamp **timestamp** – tiempo en el evento cuando se corrigen las coordinadas Integer **latitude** – latitud con grado de precisión de 0.0000001 Integer **longitude** – longitud con grado de precisión de 0.0000001 Integer **direction** – dirección, grados Integer **sattelitesCount** – número de satélites Double **speed** (km/hour) – velocidad Long **timeStamp** (segundos) – tiempo del evento (UTC)

getTrack – rastrear

Valores de entrada
String **sessionId** – identificación de sesión obtenida durante la autorización Integer **objectId** – identificador de objeto Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalol Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo

Valores de retorno
Boolean **status** – verdadero/falso verdadero en el caso de una operación exitosa. String **error** – mensaje de error en caso de que ocurra un error Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo final del intervalo Dataset **trackEvents** – conjunto de puntos de rastreo: String **gpsPos** – coordinadas del evento. Contiene valores de latitud y longitud separados por un punto y coma Integer **gpsDir** – dirección de movimiento, grados de 0 a 359 Integer **sattelitesCount** – número de satélites Double **speed** – velocidad, en km/hora con una precisión de hasta 0.1 km/hora Unixtimestamp **timeStamp** – punto de fecha y tiempo

getEngineStatisticsByPeriod – estadísticas de funcionamiento del motor durante un período

Valores de entrada
String **sessionId** – identificador de sesión recibido durante la autorización Integer **vehicleId** – identificador del vehículo Unixtimestamp **timeBegin** (segundos) – hora de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – hora de término del intervalo

Valores de retorno
Boolean **status** – verdadero/falso. Devuelve «verdadero» si la operación fue satisfactoria String **error** – texto del mensaje de error que aparece cuando se produce un error Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo de finalización del intervalo Unixtimestamp **lastDataTimestamp** (segundos) – indicación de tiempo de los últimos datos procesados (UTC) Unixtimestamp **operationStartDate** (segundos) – tiempo de inicio de la operación para el período (UTC) Unixtimestamp **operationEndDate** (segundos) – tiempo de finalización de la operación para el período (UTC) Integer **operationTime** (segundos) – tiempo de funcionamiento para el período Integer **engineOffTime** (segundos) – tempo de inactividad del motor para el período Integer **engineOnTime** (segundos) – tiempo de funcionamiento del motor para el período Integer **engineIdlingTime** (segundos) – tiempo de funcionamiento en reposo del motor para el período Integer **engineOperationTimeNormalSpeed** (segundos) – tiempo de funcionamiento del motor bajo carga normal para el período Integer **engineOperationTimeMaxSpeed** (segundos) – tiempo de funcionamiento del motor a carga máxima para el período Integer **engineLoadTime** (segundos) – tiempo de funcionamiento del motor bajo carga durante el período Integer **dataAbsenceTime** (segundos) – tiempo de ausencia de datos para el periodo

getShiftDataByPeriod – información acerca de los turnos

Valores de entrada
String **sessionId** – identificador de sesión recibido durante la autorización Unixtimestamp **timeBegin** (segundos) – tiempo de inicio de turno programado Unixtimestamp **timeEnd** (segundos) – tiempo de finalización de turno programado Integer **devBeforeBegin** (segundos) – desviación permitida desde la hora de inicio del turno antes del tiempo programado Integer **devAfterBegin** (segundos) – desviación permitida desde la hora de inicio del turno retrasado Integer **devBeforeEnd** (segundos) – desviación permitida desde la hora de final de turno antes del tiempo programado Integer **devAfterEnd** (segundos) – desviación permitida desde la hora de finalización del turno retrasado Integer **vehicleId** – identificador del vehículo
Valores de retorno
Boolean **status** – verdadero/falso. Devuelve «verdadero» si la operación fue satisfactoria String **error** – texto del mensaje de error que aparece cuando se produce el error Integer **vehicleId** – identificador del vehículo Integer **vehicleType** – tipo de vehículo (0 - 1 - camión de combustible) **shiftData**: Unixtimestamp **pointDate** (segundos) – hora real de inicio/final del turno (UTC) Integer **engineOperationTime** (segundos) – tiempo de funcionamiento del motor en el momento del inicio del turno/final del turno Integer **shiftEngineOperationTime** (segundos) – tiempo de funcionamiento del motor correspondiente al turno Double **mileage** (km) – tiempo de funcionamiento del motor en el momento del inicio del turno/final del turno Double **shiftMileage** (km) – kilometraje correspondiente al turno **fuelData** – grupo de parámetros de combustible: Double **fuelVolume** (l) – el volumen de combustible en el momento del inicio del turno/final del turno Double **fuelConsumption** (l) – el consumo real en el momento de final del turno

getIntervalsInfo – información dividida en intervalos

Valores de entrada
String **sessionId** – identificador de sesión recibido durante la autorización Integer **vehicleId** – identificador del vehículo Unixtimestamp **timeBegin** (segundos) – tiempo de inicio del intervalo Unixtimestamp **timeEnd** (segundos) – tiempo de finalización del intervalo Boolean **partsFlag** – división en partes (verdadero o falso) Integer **interval** (min) – longitud del intervalo Boolean **geocodingFlag** – resolución de dirección basándose en las coordenadas (verdadero o falso) Boolean **additionalTankFlag** – devuelve un conjunto de datos en el nivel de combustible en el tanque adicional (verdadero o falso)

Valores de retorno
Boolean **status** – verdadero/falso. Devuelve «verdadero» si la operación fue satisfactoria Integer **errorCode** – código de error. Los códigos de error posibles son: 2: Authorization required - se require una autorización para accesar los datos 3: Dead session number – se ha expirado la sesión, se necesita volver a dar autorización 4: Bad interval – se ha introducido un intervalo de tiempo inadecuado 5: Bad object – no existe objeto con este identificador 7: Unusable object– no se puede calcular el valor para un objeto con este identificador. 9: Access denied – no hay autorización para accesar el objeto 10: Data not found – no hay datos para los valores de entrada correspondientes (no hay datos crudos para [período de solicitud + un evento antes de que inicie el periodo]) 11:Blocked interval – el intervalo solicitado contiene períodos de bloqueo de datos 13: Invalid format – el formato es incorrecto 14: Undefined error – el error no está especificado 19: actual N, permitido M - el número de intervalos por período (N) supera el límite (M) String **erorrDescription** – opcional Integer **vehicleId** – identificador del vehículo Integer **vehicleType** – tipo de vehículo (0 - 1 - camión de combustible) Integer **tanksNumber** – el número de tanques de combustible Integer **numberOfIntervals** – el número de intervalos en el período de solicitud **periodData** – el conjunto de datos correspondiente al período de solicitud **intervalsData** – los datos correspondientes intervalo: Integer **intervalNumber** – número de secuencia del intervalo Unixtimestamp **startTime** – fecha de inicio del intervalo Unixtimestamp **endTime** – fecha de final de intervalo **statData** – datos estadísticos establecidos **gpsData** – datos GPS establecidos **engineData** – conjunto de datos sobre el funcionamiento del motor **fuelData** – conjunto de datos sobre el nivel de combustible **seData** – conjunto de datos sobre el funcionamiento de equipos auxiliares **canData** – datos de bus CAN

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/definition/method

Last update: **2026/11/16 13:41**

Tipos de eventos

Número del tipo del evento	Significado	Valores del parámetro, notas
1	Inicio de recarga de combustible (para tanques de recarga de combustible - recarga de combustible)	Valor de recarga de combustible, precisión 0.1 litros
2	Final de recarga de combustible (para tanques de recarga de combustible - recarga de combustible)	Valor de recarga de combustible, precisión 0.1 litros
3	Inicio de drenaje	Valor de drenaje, precisión 0.1 litros
4	Fin de drenaje	Valor de drenaje, precisión 0.1 litros
5	Ignición ENCENDIDA	Tiempo de la última vez que se apagó la ignición, minutos
6	Ignición APAGADA	
7	Fuente de alimentación externa ENCENDIDA	
8	Alimentación de batería ENCENDIDA	
9	Autorización de conductor	Código de clave iButton en HEX
10	Transición a roaming	
11	Salir de roaming	
12	Velocidad instantánea	Valor de velocidad máxima, precisión 0.1 kph
13	Tiempo de inactividad	
14	Inicio de exceso de velocidad	
15	Inicio del grupo de transacciones (para tanques de recarga de combustible)	Volumen de combustible antes del inicio de grupos de transacción, precisión 0.1 litros
16	Fin del grupo de transacciones (para tanques de recarga de combustible)	Volumen de combustible después de finalizar el grupo de transacciones, precisión 0.1 litros
17	Inicio de recarga de combustible durante grupos de transacciones (para tanques de recarga de combustible)	
18	Fin de recarga durante el grupo de transacciones (para tanques de recarga de combustible)	
19	Inicio de drenaje durante el grupo de transacciones (para tanques de recarga de combustible)	
20	Fin de drenaje durante el grupo de transacciones (para tanques de recarga de combustible)	
21	Conexión establecida	
22	Equipo adicional ENCENDIDO	
23	Equipo adicional APAGADO	
24	Exceder los límites máximos del equipo auxiliar	

Número del tipo del evento	Significado	Valores del parámetro, notas
25	Devolver a valores normales del equipo auxiliar	
31	Presionar el botón de pánico	
32	Sobrecarga del equipo auxiliar	
33	ENCENDIDO	
34	Exceso instantáneo de revoluciones del motor RPM	
35	Ingresando la geocerca	
36	Abandonar la geocerca	
38	APAGADO	
42	Inicio de parada	
43	Fin de parada	
44	Inicio de aceleración	
45	Fin de aceleración	
46	Entrada digital ENCENDIDA	
47	Entrada digital APAGADA	
48	Aceleración instantánea	
49	Conductor desconocido	
52	Manipulación del dispositivo	
53	Autorización de conductor finalizada	
54	iButton aplicado	
55	Registro de conductor eliminado	

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/definition/type

Last update: **2026/11/16 13:41**

Acceso a la demostración de los servicios en línea

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/demo

Last update: **2026/11/16 13:41**

REST API

Utilice lo siguiente para acceder la demonstración:

dirección: http://online.omnicomm.ru

usuario: rudemoru

contraseña: rudemo123456

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/demo/rest

Last update: **2026/11/16 13:41**

SOAP

Si es necesario, para probar la conexión a los servicios en línea (para verificar la aplicación sin usar una cuenta real de Omnicomm Online, o si hay dudas sobre las configuración de la red), se puede utilizar el servidor de demostración de los servicios en línea de Omnicomm.

Dirección de conexión de servicios en línea (no apto para el sistema de contabilidad 1C):

http://demo.omnicomm.ru:8000/AnalyticalServer/v2/ws?wsdl

Dirección de conexión de servicios en línea para el sistema 1C:

http://demo.omnicomm.ru:8001/AnalyticalServer/v2/ws?wsdl

Usuario: rudemoru Contraseña: rudemo123456

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/demo/soap

Last update: **2026/11/16 13:41**

Información general

La integración con Omnicomm Online se utiliza para expandir la funcionalidad de los sistemas de terceros y automatizar la entrada de datos obtenidos de Omnicomm Online a los sistemas de contabilidad.

Los siguientes métodos de integración están disponibles:

- REST API
- usando el protocolo SOAP

Le recomendamos usar REST API para la integración. No es posible expandir las capacidades de integración usando el protocolo SOAP.

El módulo de integración para el sistema de Omnicomm se debe de desarrollar y liberado por los implementadores del sistema de gestión de cuentas de otros fabricante.

Omnicomm no se compromete a delegar especialistas o a desarrollar el módulo para los sistemas de otros fabricantes.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/general-information

Last update: **2026/11/16 13:41**

Características principales

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/opportunity

Last update: **2026/11/16 13:41**

REST API

REST API facilita la automatización de:

- gestión de usuario
- gestión de perfil de vehículo
- gestión de geocerca
- recepción de informes
- recepción de notificaciones
- usando el sub-sistema OVMS (servicio de vídeo)

REST API permite empezar la recolección, el procesamiento, y el monitoreo de datos rápidamente desde el vehículo.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/opportunity/rest

Last update: **2026/11/16 13:41**

SOAP

Los datos que son descargados del sistema Omnicomm Online permiten, al iniciar la adecuada funcionalidad de los sistemas de terceros, las siguientes tareas realizadas de manera automática:

- Introducción de datos sobre el kilometraje y consumo de combustible en las hojas de ruta
- Control del kilometraje y tiempo de uso para calcular la cantidad de trabajo, salario del conductor, etc.
- Control del equipo adjunto para calcular el alcance del trabajo realizado
- Control del kilometraje, horas de funcionamiento del motor y equipo auxiliar para planear su mantenimiento
- Amortización de combustible consumido
- Comparación de documentados llenados con los verdaderos para prevenir el robo de combustible
- Generación de informes de infracciones
- Plan versus análisis actual sobre el trabajo realizado, combustible consumido, etc.
- Uso de la ubicación actual a fin de seleccionar el vehículo adecuado para realizar la orden
- Visualización de rastreo sobre el mapa en software de otras compañias

Se puede generar varios informes usando los parámetros del vehículo descargados de Omnicomm Online.

Los datos de Omnicomm Online habilitan un cálculo rápido de la exactitud del sistema de planificación aceptado y un trato adecuado con los posibles planes de los empleados de flota asociados al robo de productos de petróleo y el uso no autorizado de los vehículos.

Los objetos principales para la integración de los sistemas son los sistemas de contabilidad, sistemas de gestión de datos (ERP), y sistemas de contabilidad de sucursal.

El protocolo SOAP no soporta el uso del sub-sistema OVMS (servicio de vídeo). Utilice REST API para una integración rápida con el sub-sistema OVMS.

Los servicios de Internet recopilan los siguientes parámetros:

- Kilometraje
- Consumo de combustible
- Tiempo de funcionamiento de motor
- Tiempo de funcionamiento del equipo auxiliar
- Gráfico de volumen de combustible
- Kilometraje con velocidad excedida*
- Duración del movimiento*
- Tiempo de funcionamiento del motor en movimiento*
- Funcionamiento del motor sin movimiento*
- Tiempo de «Apagado de motor»*
- Nivel de combustible en un momento dado*
- Consumo de combustible en movimiento*
- Consumo de combustible sin movimiento*

- Tasa de consumo de combustible por una hora de funcionamiento del motor*
- Rastreo*
- Ubicación actual*

* – Los parámetros marcados están disponibles en la versión 2.4.2.

Además de estos parámetros, es posible recibir todos los eventos mostrados en el informe de «Eventos» en Omnicomm Online.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/opportunity/soap

Last update: **2026/11/16 13:41**

REST API

Usted puede encontrar la especificación del método API en https://developers.omnicomm-world.com

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/rest_api

Last update: **2026/11/16 13:41**

Obtener los derechos para utilizar REST API

Comuníquese con el soporte técnico de Omnicomm por correo electrónico a support@omnicomm-world.com para obtener los derechos para utilizar REST API

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/rest_api/access

Last update: **2026/11/16 13:41**

Gestión de vehículo

Es necesario agregar un perfil del vehículo para activar el procesamiento de datos del vehículo en Omnicomm Online.

Un vehículo se identifica con el identificador de cadena único (UUID), el cual es asignado al agregar un vehículo nuevo a Omnicomm Online o con el número de identificación de la terminal.

Se pueden agregar vehículos a los grupos, y dicho vehículo puede pertenecer a más de un grupo. También se puede configurar el acceso a los grupos de vehículos para los usuarios. Los grupos de vehículos se crean al crear un usuario, así como en la interfaz Omnicomm Online.

Descripción de los métodos de gestión de vehículo:
https://developers.omnicomm-world.com/#/Vehicles/

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/rest_api/add_vehicle

Last update: **2026/11/16 13:41**

Gestión de usuario

La gestión de usuario cubre la adición, el borrado, y el bloqueo de usuarios en Omnicomm Online, así como obtener un listado de todos los usuarios de Omnicomm Online.

Un usuario autorizado puede acceder los datos del vehículo.

Un sistema externo puede realizarle las tareas al usuario después de iniciar sesión bajo su cuenta (después de obtener el JWT con el derecho del usuario).

Descripción de los métodos de gestión de usuario: https://developers.omnicomm-world.com/#/Users/

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/rest_api/ass_user

Last update: **2026/11/16 13:41**

Autorización

Un token JWT debe estar indicado en el encabezado de Autorización al acceder los métodos REST API (excepto para los métodos de autorización). Este token JWT le otorga el derecho a usar REST API.

Formato JWT: JWT<espacio><el JWT recibido del método de autorización>

Ejemplo:

```
Authorization: JWT
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MTU3MDM1OTEsImxvZ2luIjoiYWR0...
4ifQ.0I0CXcwWWxZWARE0eUEPOAvKd0prW_Uf0jbOMLnd5SI
```

La fecha de expiración de JWT se indica en el atributo de carga útil exp en el formato Unix Time Stamp en UTC. Después de la expiración, aparecerá Error 401 No autorizado al acceder el método.

Se puede obtener el JWT usando el siguiente método POST /auth/login?jwt=1 o, al expirar, post /auth/refresh al indicar en el encabezado de actualizar de la Autorización, el JWT obtenido durante la autorización del post /auth/login?jwt=1.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/rest_api/authorization

Last update: **2026/11/16 13:41**

Gestión de geocerca

Las geocercas son áreas virtuales en el mapa creadas por los usuarios en

Omnicomm Online. Al crear una geocerca, especifique su forma (un polígono, un círculo, o una línea) y sus coordenadas geográficas.

Las geocercas se usan para monitorear la ubicación de un vehículo (entrando/saliendo una geocerca) y otros parámetros de funcionamiento, así como la velocidad del vehículo.

Descripción de los métodos de gestión de geocerca:
https://developers.omnicomm-world.com/#/Geozones/

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/rest_api/geofence

Last update: **2026/11/16 13:41**

Manejo de notificaciones

Las notificaciones se usan para notificar a los usuarios de inmediato sobre los eventos registrados.

Descripción de métodos para el manejo de notificaciones:
https://developers.omnicomm-world.com/#/Notifications/

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/rest_api/notifications

Last update: **2026/11/16 13:41**

Informes

Los informes se usan para obtener información variada sobre el funcionamiento del vehículo.

Descripción de los métodos de obtención de informes:
https://developers.omnicomm-world.com/#/Reports/

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/rest_api/report

Last update: **2026/11/16 13:41**

Restricciones

Hay restricciones sobre la intensidad de las solicitudes que se envían a Omnicomm Online REST API para protegerlo contra los ataques DoS y errores de los sistemas de otros fabricantes.

Cuando estas restricciones son excedidas, se bloqueará cualquier solicitud de la dirrección IP o usuario correspondiente al REST API.

Intentos de autorización fallidos

No mayor a 10 en 1 minuto desde la misma dirección IP.

Llamadas autorizadas

No mayor a 180 en 1 minuto por cada usuario.

Llamadas no autorizadas

No mayor a 60 en un minuto desde la misma dirección IP.

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/rest_api/restriction

Last update: **2026/11/16 13:41**

Servicio de vídeo Omnicomm

El servicio de vídeo de Omnicomm se encarga de la gestión de la terminal de vídeo y brinda material de vídeo al usuario.

Características principales del servicio:

- recepción, almacenaje, modificación, y provisión de datos de los perfiles de la terminal de vídeo
- recepción, almacenaje, modificación, y provisión de los parámetros de tareas para la descarga de archivos de vídeo
- ejecución de las tareas de descarga de archivos de vídeo

Descripción de métodos para el manejo del servicio de vídeo:
https://developers.omnicomm-world.com/#/VideoService/

Obtención de un fragmento de vídeo

Esta selección describe cómo usar el servicio de vídeo en una situación típica.

Para obtener un fragmento de vídeo:

1. Inicie sesión a la cuenta de un distribuidor o un usuario que tenga derechos al vehículo o usar el servicio de vídeo: POST /auth/login?jwt=1

2. Obtenga el perfil de vídeo del identificador de la terminal del vehículo: GET /service/ovms/api/profiles

3. Obtenga el archivo de vídeo:

- Crea una tarea para obtener el fragmento de vídeo basado en el perfil de vídeo recibido: POST /service/ovms/api/tasks
- Verifique de manera periódica el estado de la tarea: GET /service/ovms/api/tasks/{task_id}
- Después de recibir el estado de «Hecho» de la tarea, solicite el archivo de vídeo:GET /service/ovms/api/tasks/file/{task_id}

From:
https://doc.omnicomm.ru/ - **OmniDoc**

Permanent link:
https://doc.omnicomm.ru/es/omnicomm_online-integration_api/rest_api/video

Last update: **2026/11/16 13:41**