



Web Service de Integración Sascar

MICHELIN Connected Fleet

2026

Índice

1. Términos de Confidencialidad	13
2. Introducción	14
2.1. Soporte	15
2.2. URL de conexión	16
2.3. Autenticación	17
3. Métodos – Comandos y Actuadores	18
3.1. AtualizarSenha	18
3.2. ObterAlertasAVDVinculados	20
3.3. ObterGrupoAtuadores	22
3.4. ObterCadastroAlertasAVD	27
3.5. ObterStatusComando	29
3.6. ObterStatusComandoTicketSascar	31
3.7. ObterTipoComando	33
3.8. ObterSequenciamentoEvento	35
3.9. ComandoEmbarquePontoDiario	37
4. Métodos – Vehículos y Clientes	39
4.1. ObterClientes	39
4.2. ObterClientesV2	41
4.3. ObterVeiculos	43
4.4. ObterVeiculosJSON	48
4.5. ObterVeiculosRFNacional	53
4.6. getVehiclesJSON	56
4.7. ObterDadosAdicionais	61
4.8. ObterDadosAdicionaisCliente	63
4.9. ObterPontosReferencia	65
4.10. ObterRotas	67
4.11. ObterEnderecoPosicao	69
5. Métodos – Posición	71
5.1. ObterPacotePosicoes	71
5.2. ObterPacotePosicoesMotorista	79
5.3. ObterPacotePosicoesMotoristaComPlaca	87
5.4. ObterPacotePosicoesMotoristaRestricao	95
5.5. ObterPacotePosicaoMotoristaPorRange	103
5.6. ObterPacotePosicaoMotoristaHistorico	111
5.7. ObterPacotePosicaoMotoristaPorRangeJSON	119
5.8. ObterPacotePosicoesMotoristaJSON	127
5.9. ObterPacotePosicoesRestricao	136
5.10. ObterPacotePosicaoPorRange	143
5.11. ObterPacoteLocalizacao	151
5.12. ObterPacotePosicaoHistorico	154
5.13. ObterPacotePosicoesJSON	162
5.14. ObterPacotePosicoesJSONComPlaca	170
5.15. ObterPacotePosicaoPorRangeJSON	178
5.16. ObterPacotePosicoesRFNacional	186
5.17. getPositionsPacketJSON	188
5.18. getDriverPositionPacketJSON	196

5.19. getPositionPacketByRangeJSON	205
5.20. getDriverPositionPacketByRangeJSON	214
5.21. getPositionPacketWithLicensePlateJSON	223
6. Métodos – Layout e Macros	231
6.1. ObterMacroTd50Tmcd	231
6.2. ObterMacroTd50TmcdDetalhado	233
6.3. ObterMascaraDispositivo	235
6.4. ObterMacroTD40	237
6.5. ObterLayout	239
6.6. ObterLayoutDetalhado	241
6.7. ObterLayoutAcaoEmbarcadaAVD	243
6.8. ObterLayoutAreaAvd	245
6.9. ObterLayoutData	247
6.10. ObterLayoutTecladoVeiculos	249
6.11. ObterLayoutGrupoPontos	251
7. Métodos – Telemetria e Câmeras	253
7.1. ObterEventosTempoDirecao	253
7.2. ObterEventosTempoDirecaoDataChegada	257
7.3. ObterDeltaTelemetriaIntegracaoInercia	261
7.4. ObterDeltaTelemetriaIntegracaoDataChegada	268
7.5. ObterDeltaTelemetriaIntegracaoInerciaDataChegada	271
7.6. ObterEventoTelemetriaIntegracaoDataChegada	277
7.7. getSmartCamerasEvents	280
7.8. SolicitarEventosCaixaPreta	284
7.9. RecuperarEventosCaixaPreta	286
8. Métodos – Motoristas	290
8.1. ObterMotoristas	290
8.2. ObterMotoristasJson	293
8.3. ObterMotoristasVeiculos	296
9. Referencia – Estructura de Eventos y Excepciones	298
9.1. Estructura Actual	298
9.2. Nueva Estructura de Eventos	300
9.3. Ejemplos de la Nueva Estructura de Eventos	301
9.4. Mensajes de Excepción	302
10. Anexo	303
10.1. Creación de Cliente para Conexión con el Web Service SasIntegra	303



Nueva versión disponible: Este documento corresponde a la versión **3.00** del Manual Técnico del Web Service SasIntegra. Consulte la versión más reciente del manual para obtener información actualizada sobre nuevos métodos y funcionalidades.

REVISION

Fecha	Versión	Autor	Descripción
2012-06-01	1.0	Desarrollo de Sistemas	Creación del documento.
2012-06-15	1.1	Desarrollo de Sistemas	Actualización de los elementos de retorno del paquete de posición.
2012-07-10	1.2	Desarrollo de Sistemas	Añadido tema sobre la nueva estructura de eventos; modificada la descripción del campo de eventos del paquete de posición.
2012-07-27	1.3	Desarrollo de Sistemas	Cambio del medio de autenticación.
2012-08-27	1.4	Desarrollo de Sistemas	Modificación del envío de paquetes de posición incluyendo datos de dirección y puntos de referencia en todos los paquetes.
2012-09-24	1.5	Desarrollo de Sistemas	Añadido ejemplo descriptivo de la nueva estructura de Eventos.
2013-01-08	1.6	Desarrollo de Sistemas	Inclusión de nuevos servicios de layouts y macros: • obterLayoutDetalhado • obterMacroTd50TmcdDetalhado
2013-02-08	1.7	Desarrollo de Sistemas	Modificación del método obterGrupoAtuador (envía sólo información de Eventos, Entradas y Salidas). Eliminación del campo tipoMensagem e inclusión del campo jamming en el método obterPacotePosições .
2013-03-25	1.8	Desarrollo de Sistemas	Inclusión de nuevos métodos: • Consultar Conductores • Consultar Conductores Vinculados a Vehículos • Consultar Layout de Teclado en Vehículo • Consultar Layout de Grupo de Puntos Inclusión de regla que permite retornar sólo el registro deseado mediante id para los métodos obterClientes, obterVeiculos e obterMotoristas.
2013-06-05	1.9	Desarrollo de Sistemas	Inclusión del Anexo A con ejemplo de creación de un cliente para el uso de los métodos del Web Service SasIntegra.

Fecha	Versión	Autor	Descripción
2013-08-01	1.10	Desarrollo de Sistemas	Actualización de nuevo campo en <code>obterMacroTd50Tmcd</code> y de elementos y ejemplos en diversos métodos.
2013-10-18	1.11	Desarrollo de Sistemas	Inclusión del nuevo método <code>obterLayoutAcaoEmbarcadaAVD</code> .
2014-01-31	1.12	Desarrollo de Sistemas	Cambio de la URL de llamada del servicio de integración. Inclusión de nuevos métodos y campos: - Identificador del paquete (<code>idPacote</code>) en <code>obterPacotePosicoes()</code> <code>atualizarSenha()</code> <code>obterPacotePosicaoPorRange()</code> - Campos <code>idMacro</code> y <code>layout</code> en el retorno de <code>ObterLayoutDetalhado</code>
2014-05-21	1.13	Desarrollo de Sistemas	Inclusión de nuevos métodos de Telemetría: <code>obterEventosTelemetria</code> <code>obterDeltasTelemetria</code> <code>enviarParametrizacaoTelemetria</code>
2014-10-23	1.14	Desarrollo de Sistemas	Inclusión del nuevo método <code>comandoEmbarquePontoDiario</code> .
2015-03-06	1.15	Desarrollo de Sistemas	Inclusión del nuevo método <code>obterStatusComandoTicketSascar</code> . Inclusión de dos nuevos estados (Comando Cancelado y Comando Pendiente) en los retornos de los métodos: <code>obterStatusComandoTicketSascar</code> <code>obterStatusComando</code>
2015-04-06	1.16	Desarrollo de Sistemas	Inclusión del nuevo método <code>obterPacoteLocalizacao</code> .
2016-01-14	1.17	Desarrollo de Sistemas	Inclusión del nuevo método <code>obterDadosAdicionais</code> .
2015-09-03	1.18	Desarrollo de Sistemas	Modificación en los elementos: <code>ObterGrupoAtuadores</code> <code>ObterSequenciamentoEvento</code>
2016-04-22	1.19	Desarrollo de Sistemas	Inclusión de mensajes de Excepciones.
2016-05-27	1.20	Desarrollo de Sistemas	Actualización del Manual de la V3.

Fecha	Versión	Autor	Descripción
2016-07-28	1.21	Desarrollo de Sistemas	Actualización del Manual de la V3 — Observación en Inhibir Comando en el método ObterStatusComando .
2016-08-03	1.22	Desarrollo de Sistemas	Inclusión de regla de bloqueo de clientes con restricción. Modificación en los elementos: • obterPacotePosicoes • obterPacotePosicoesJSON • obterPacotePosicaoPorRange • obterPacotePosicaoPorRangeJSON • obterPacotePosicaosHistorico • obterMotorista • obterVeiculos Inclusión del nuevo método obterPacotePosicoesRestricao.
2016-09-14	1.23	Desarrollo de Sistemas	Corrección del XML en el método obterPacotePosicaoHistorico y de la cantidad de registros en obterPacotePosicoes .
2016-10-03	1.24	Desarrollo de Sistemas	Inclusión de los IDs 674 y 675 (eventos del seguro de 5.ª rueda inteligente) en la lista de actuadores soportados por Sascar.
2017-02-06	1.25	Desarrollo de Sistemas	Inclusión de las Restricciones Financieras.
2017-02-09	1.26	Desarrollo de Sistemas	Actualización del manual en relación a los valores de retorno del tipoTeclado .
2017-04-19	1.27	Desarrollo de Sistemas	Inclusión del método ObterPacotePosicaoRFNacional .
2017-07-06	1.28	Desarrollo de Sistemas	Inclusión del nuevo valor 2 (Satelital Texto Libre) para el campo satelite. Impacta los métodos: • ObterPacotePosicoes • ObterPacotePosicoesRestricao • ObterPacotePosicaoPorRange • obterPacotePosicaoHistorico • obterPacotePosicoesJSON • obterPacotePosicaoPorRangeJSON
2017-08-03	1.29	Desarrollo de Sistemas	Actualización del método obterPacotePosicaoHistorico .
2017-09-18	1.30	Desarrollo de Sistemas	Ajuste en el número de consultas simultáneas.

Fecha	Versión	Autor	Descripción
2017-12-29	1.31	Desarrollo de Sistemas	Actualización en el retorno de <code>ObterGrupoAtuadores</code> con inclusión de nuevos teclados: • SASMDT • TMCD • TD50 • TD-40 Inclusión de los campos <code>velocidadeMaximaEvento</code> y <code>velocidadeReferencia</code> en el retorno de <code>obterEventoTelemetriaIntegracao</code>.
2018-02-01	1.32	Desarrollo de Sistemas	Creación de los nuevos métodos de posición con información de conductor y limpiaparabrisas: • <code>obterPacotePosicoesMotorista</code> • <code>obterPacotePosicoesMotoristaRestricao</code> • <code>obterPacotePosicoesMotoristaJSON</code> • <code>obterPacotePosicaoMotoristaPorRangeJSON</code> • <code>obterPacotePosicaoMotoristaPorRange</code> • <code>obterPacotePosicaoMotoristaHistorico</code>
2018-05-08	1.33	Desarrollo de Sistemas	Inclusión de la información de ID del evento, Velocidad de referencia, Velocidad máxima del evento y Duración del evento en los métodos de posición: • <code>obterPacotePosicoesMotorista</code> • <code>obterPacotePosicoesMotoristaJSON</code> • <code>ObterPacotePosicoesMotoristaRestricao</code> • <code>ObterPacotePosicaoMotoristaHistorico</code> • <code>ObterPacotePosicaoMotoristaPorRange</code> • <code>ObterPacotePosicaoMotoristaPorRangeJSON</code>
2018-05-09	1.34	Desarrollo de Sistemas	Creación del método <code>obterEventoTelemetriaDescricao</code>, que retorna una lista con todos los Eventos disponibles, sus descripciones, IDs y Tipos de Evento.
2018-05-22	1.35	Desarrollo de Sistemas	Inclusión de la información de Consumo de Combustible en el método <code>obterDeltaTelemetriaIntegracao</code>.
2019-04-11	1.36	Desarrollo de Sistemas	Actualización de la información de Velocidad y Odómetro en los métodos de posición: por defecto se envía la del GPS; si el vehículo posee Telemetría, se envían las de la Telemetría.

Fecha	Versión	Autor	Descripción
2019-07-18	1.37	Desarrollo de Sistemas	Inclusión de parámetros de frenado brusco en el método enviarParametrizacaoTelemetria : pedal de freno accionado y velocidad mínima al inicio del evento.
2019-07-20	1.38	CSA	Modificación en el método ObterVeiculos , retornando datos en el campo descricao e incluido el campo idEquipamentoDesc .
2019-07-20	1.39	CSA	Creado el método ObterVeiculosJson .
2019-07-25	1.40	CSA	Corrección en las descripciones de los elementos idReferencia , pontoReferencia , anguloReferencia y distanciaReferencia .
2019-07-06	1.41	CSA	Inclusión de los métodos de posiciones que retornan patente como atributo del response: • obterPacotePosicoesComPlaca • obterPacotePosicoesMotoristaComPlaca • obterPacotePosicoesJSONComPlaca Inclusión del método obterVeiculosRFNacional .
2019-09-02	1.42	Desarrollo de Sistemas	Inclusión del nuevo método ObterLayoutAreaAvd .
2019-10-09	1.43	Desarrollo de Sistemas	Inclusión del nuevo método obterEventosTempoDirecao .
2019-10-22	1.44	Desarrollo de Sistemas	Inclusión del nuevo método obterLayoutData .
2019-11-12	1.45	Desarrollo de Sistemas	Inclusión de los nuevos métodos en inglés: • getPositionPacketJSON • getDriverPositionPacketJSON • getPositionPacketByRangeJSON • getDriverPositionPacketByRangeJSON • getVehiclesJSON • getPositionPacketWithLicensePlateJSON
2020-04-14	1.46	Desarrollo de Sistemas	Inclusión de los campos aceleracaoLateralForcaG , direcaoForcaG y limiteMovLateralForcaG en el método obterEventoTelemetriaIntegracao para eventos de Fuerza G lateral (214, 215, 216).
2020-04-30	1.47	Desarrollo de Sistemas	Inclusión del nuevo método obterDadosAdicionaisCliente .

Fecha	Versión	Autor	Descripción
2020-08-25	1.48	Desarrollo de Sistemas	Inclusión del campo país en los métodos JSON: • obterPacotePosicaoMotoristaPorRangeJSON • obterPacotePosicaoPorRangeJSON • obterPacotePosicoesJSON • obterPacotePosicoesJSONComPlaca • obterPacotePosicoesMotoristaJSON • getPositionsPacketJSON • getDriverPositionPacketJSON • getPositionPacketByRangeJSON • getDriverPositionPacketByRangeJSON • getPositionPacketWithLicensePlateJSON Corrección de la solicitud y el retorno, y el formato de fecha y hora en el método obterEventoTelemetriaIntegracao.
2020-09-23	1.49	Desarrollo de Sistemas	Inclusión del nuevo método obterDeltaTelemetriaIntegracaoInercia. Descontinuación del método obterDeltaTelemetriaIntegracao.
2020-09-30	1.50	Desarrollo de Sistemas	Inclusión de los campos nivelCombustivel/fuelLevel en los métodos: • obterPacotePosicaoMotoristaPorRangeJSON • obterPacotePosicoesMotoristaJSON • obterPacotePosicoesJSON • obterPacotePosicoesJSONComPlaca • obterPacotePosicaoPorRangeJSON • getPositionsPacketJSON • getDriverPositionPacketJSON • getPositionPacketByRangeJSON • getDriverPositionPacketByRangeJSON • getPositionPacketWithLicensePlateJSON
2020-10-26	1.51	CSA	Actualización del período de datos disponibles para consumo: cambiado de 48 horas a D0 y D-1.
2020-11-11	1.52	Desarrollo de Sistemas	Inclusión de nuevos parámetros en el método obterDeltaTelemetriaIntegracaoInercia .

Fecha	Versión	Autor	Descripción
2021-01-02	1.53	Desarrollo de Sistemas	Inclusión del campo litrometro/lithometer (litros, 3 decimales) en los métodos JSON: • obterPacotePosicaoMotoristaPorRangeJSON • obterPacotePosicoesMotoristaJSON • obterPacotePosicoesJSON • obterPacotePosicoesJSONComPlaca • obterPacotePosicaoPorRangeJSON • getPositionsPacketJSON • getDriverPositionPacketJSON • getPositionPacketByRangeJSON • getDriverPositionPacketByRangeJSON • getPositionPacketWithLicensePlateJSON
2021-03-29	1.54	Desarrollo de Sistemas	Creación de los métodos: • ObterRotas • ObterAlertasAVDVinculados • ObterCadastroAlertasAVD
2021-05-11	1.55	Desarrollo de Sistemas	Inclusión de nuevos campos en el método obterDeltaTelemetriaIntegracaoInercia y nuevos eventos en el método obterEventoTelemetriaIntegracao.
2021-06-17	1.56	Desarrollo de Sistemas	Creación del método getSmartCamerasEvents.
2021-07-08	1.57	Desarrollo de Sistemas	Inclusión de nuevos campos y eventos en el método obterEventoTelemetriaIntegracao.
2021-08-27	1.58	Desarrollo de Sistemas	Inclusión de nuevos Eventos en el método obterGrupoAtuadores.
2021-11-16	1.59	CSA	Cambio de HTTP a HTTPS con TLSv1.2 (véase apartados "3.1" y "Anexo").

Fecha	Versión	Autor	Descripción
2022-01-03	1.60	CSA	Actualización del Manual sin inclusión de nuevos métodos o campos. Eliminado registro duplicado de "ObterPacotePosicoesMotorista". Movido el apartado "Soporte" al inicio del manual. Complemento a la revisión 1.28: comportamiento del campo satelite con valor 2 (Satelital Texto Libre) en los métodos: • ObterPacotePosicoes • ObterPacotePosicoesMotorista • ObterPacotePosicoesMotoristaComPlaca • ObterPacotePosicoesMotoristaRestricao • ObterPacotePosicaoMotoristaPorRange • obterPacotePosicaoMotoristaHistorico • obterPacotePosicaoMotoristaPorRangeJSON • obterPacotePosicoesMotoristaJSON • ObterPacotePosicoesRestricao • ObterPacotePosicaoPorRange • obterPacotePosicaoHistorico • obterPacotePosicoesJSON • obterPacotePosicoesJSONComPlaca • obterPacotePosicaoPorRangeJSON • getPositionsPacketJSON • getDriverPositionPacketJSON • getPositionPacketByRangeJSON • getDriverPositionPacketByRangeJSON • getPositionPacketWithLicensePlateJSON
2022-03-21	1.61	Desarrollo de Sistemas	Inclusión de los nuevos métodos: • obterDeltaTelemetriaIntegracaoDataChegada • obterDeltaTelemetriaIntegracaoInerciaDataChegada • obterEventoTelemetriaIntegracaoDataChegada
2022-06-18	1.62	Desarrollo de Sistemas	Inclusión del nuevo método obterEventosTempoDirecaoDataChegada.

Fecha	Versión	Autor	Descripción
2022-08-18	1.63	Desarrollo de Sistemas	Actualización del Manual con inclusión del campo statusAncora en los métodos: • obterPacotePosicaoMotoristaPorRangeJSON • obterPacotePosicaoPorRangeJSON • obterPacotePosicoesJSON • obterPacotePosicoesJSONComPlaca • obterPacotePosicoesMotoristaJSON El método obterSequenciamentoEvento pasará a retornar las informaciones del Ancora.
2024-07-25	1.64	Ingeniería	Inclusión de la información de Envío de Horómetro = 0 cuando el método de envío sea satelital para la tecnología MSC de rastreadores.
2024-07-25	1.65	Desarrollo de Sistemas	Ajustes de información de los métodos: • ObterLayoutAreaAvd • obterEventosTempoDirecaoDataChegada • ObterEventosTempoDirecao
2024-10-14	1.66	Desarrollo de Sistemas	Inclusión del campo acessórios en los métodos: • obterPacotePosicoesJSON • obterPacotePosicoesJSONComPlaca • obterPacotePosicoesMotoristaJSON • obterPacotePosicaoMotoristaPorRangeJSON • obterPacotePosicaoPorRangeJSON • getPositionsPacketJSON • getPositionPacketByRangeJSON • getDriverPositionPacketByRangeJSON • getPositionPacketWithLicensePlateJSON • getDriverPositionPacketJSON
2025-01-14	2.05	Desarrollo de Sistemas	Adaptación del layout al estándar institucional. Adecuación a la LGPD.
2026-04-07	2.06	Desarrollo de Sistemas	Inclusión del nuevo método para soporte de CNPJ alfanumérico.
2026-05-05	2.07	Desarrollo de Sistemas	Modificación del método ObterClientes para mantener compatibilidad del protocolo y funcionamiento en virtud del nuevo CNPJ alfanumérico.

Fecha	Versión	Autor	Descripción
2026-08-20	3.00	Desarrollo de Sistemas	Inclusión del nuevo método <code>obterMotoristasJson</code>: • Retorno equivalente al método <code>obterMotoristas</code> en formato JSON. • Inclusión de los campos <code>ativo</code> y <code>dataInativacao</code>. • El método <code>obterMotoristas</code> permanece sin cambios.

1. Términos de Confidencialidad

The information contained herein is confidential and the property of SASCAR Tecnologia e Segurança Automotiva S/A (SASCAR). No data be used for other purposes rather than development authorized by SASCAR, and can not be shared with others parties outside your company without previous written authorization from SASCAR. It is forbidden to generate photocopies of this document, along with any kind of reproduction or distribution, partially or in its integrity, by any means, including graphics, magnetic, photographic or electronic. SASCAR reserves its right to alter this document without prior notice and to require the devolution of this document at the end of development process.

La información contenida en este documento es confidencial y propiedad de SASCAR Tecnología y Seguridad Automotriz S/A (SASCAR). Ninguna información puede utilizarse para otros fines que no sean el desarrollo con autorización de SASCAR, y no puede compartirse con personas ajenas a su empresa sin previa autorización escrita de SASCAR. Queda prohibido realizar fotocopias de este documento, así como su reproducción o distribución, parcial o total, por cualquier medio, incluyendo gráficos, magnéticos, fotografías o medios electrónicos. SASCAR se reserva el derecho de modificar este documento sin previo aviso y de exigir la devolución del mismo al finalizar el proceso de desarrollo.

2. Introducción

Este documento tiene como finalidad documentar el uso del Web Service de Integración Sascar. La información aquí presentada será útil durante el desarrollo de aplicaciones que utilicen el Web Service de Integración como fuente de información. La integración de datos consiste en la disponibilización de información de rastreo de los equipos instalados en vehículos. Todos los datos, como mensajes, estados de sensores, actuadores e información de posicionamiento, estarán disponibles para consumo a través del Web Service. La información de este manual es exclusivamente de carácter técnico. Para comprender el material descrito a continuación, es necesario que el lector posea conocimientos sobre Web Service, comunicación de red TCP/IP, protocolo SOAP y XML. SASCAR se limita a disponibilizar la información en D0 y D-1, siendo el INTEGRADOR el responsable del consumo de dicha información dentro del plazo estipulado; por lo tanto, Sascar no se responsabiliza por la manera en que el INTEGRADOR utilice la información. La información de rastreo sólo se pone a disposición del INTEGRADOR mediante autorización expresa del cliente, siendo el INTEGRADOR responsable de cualquier uso indebido de información privilegiada del CLIENTE, así como de la custodia de la información de los clientes que le sean dirigidos.

2.1. Soporte

Esperamos que la información contenida en este documento sea suficiente para el uso del Web Service de Integración Sascar. Sin embargo, si surgieran dudas que no hayan sido abordadas en este documento, contáctese con nuestra central de atención a través de los teléfonos 4002-6004 (Capitales y Regiones Metropolitanas) o 0800-648-6004 (Demás Regiones) seleccionando la opción 2 del IVR. Las sugerencias para la mejora continua de nuestros servicios y comentarios positivos también pueden dirigirse a la central de atención. Desde ya agradecemos la preferencia por nuestra tecnología.



Este manual está en constante actualización; siempre verifique en el sitio de SASCAR si está utilizando la última versión.

2.2. URL de conexión

URL SasIntegra

<https://sasintegra.sascar.com.br/SasIntegra/SasIntegraWSService?wsdl>



Sascar, priorizando la seguridad de la información de sus clientes, está descontinuoando el protocolo HTTP para el servicio "SASINTEGRA.SASCAR.COM.BR". Siguiendo las recomendaciones del IETF (Internet Engineering Task Force), la comunicación se realizará mediante el protocolo HTTPS con TLS v1.2. Debido a este cambio, los clientes deberán validar si sus aplicaciones soportan la comunicación a través de este protocolo y, de ser necesario, realizar los ajustes para soportar HTTPS con TLS v1.2.

2.3. Autenticación

La autenticación con el Web Service se realizará mediante el envío del usuario y contraseña del integrador como parámetros de los métodos que sean consumidos.

Consultas Simultáneas

El Web Service de Integración cuenta con un control sobre la cantidad de consultas simultáneas que podrán realizarse.



El límite actual es de **1 solicitud simultánea por integradora**. Las consultas que excedan este límite serán rechazadas por el Web Service.

Los métodos sujetos a esta restricción son:

- [obterPacotePosicoes](#)
- [obterPacotePosicoesRestricao](#)
- [obterPacotePosicoesRFNacional](#)
- [obterPacotePosicoesJSON](#)
- [obterPacotePosicaoPorRangeJSON](#)
- [obterPacotePosicaoPorRange](#)
- [obterPacotePosicaoHistorico](#)
- [obterPacotePosicoesMotorista](#)
- [obterPacotePosicoesMotoristaRestricao](#)
- [obterPacotePosicoesMotoristaJSON](#)
- [obterPacotePosicaoMotoristaPorRangeJSON](#)
- [obterPacotePosicaoMotoristaPorRange](#)
- [obterPacotePosicaoMotoristaHistorico](#)
- [getPositionPacketJSON](#)
- [getDriverPositionPacketJSON](#)
- [getPositionPacketByRangeJSON](#)
- [getDriverPositionPacketByRangeJSON](#)
- [getPositionPacketWithLicensePlateJSON](#)

Para los demás métodos no hay límite de solicitudes simultáneas. El Web Service permite realizar una consulta por gestora, pudiendo retornar hasta 3.000 paquetes.

3. Métodos – Comandos y Actuadores

3.1. AtualizarSenha

Descripción

Método que permite al integrador modificar o actualizar su contraseña de acceso a los métodos de consumo de la integración web service.

Declaración del Método

```
String atualizarSenha(String usuario, String senhaAtual, String novaSenha)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senhaAtual	String	Sí	Contraseña actual del integrador para acceso al sistema.
novaSenha	String	Sí	Nueva contraseña para reemplazar la contraseña actual.

Retorno

En caso de éxito retorna un mensaje informando que la contraseña fue modificada: **"Login Atualizado com Sucesso! Agora o webservice já está pronto para utilização."**

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:atualizarSenha>
      <usuario>Usuário</usuario>
      <senhaAtual>ValorSenhaAtual</senhaAtual>
      <novaSenha>ValorSenhaNova</novaSenha>
    </web:atualizarSenha>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:atualizarSenhaResponse xmlns:ns2="http://webservice.web.integracao.sascar.com.br/">
      <return>
        Login Atualizado com Sucesso! Agora o webservice já esta pronto para utilizacao!
      </return>
    </ns2:atualizarSenhaResponse>
  </S:Body>
</S:Envelope>
```

```
</return>
</ns2:atualizarSenhaResponse>
</S:Body>
</S:Envelope>
```

3.2. ObterAlertasAVDVinculados

Descripción

Método que permite al integrador consultar qué alertas AVD están vinculados a los vehículos.

Declaración del Método

```
String obterAlertasAVDVinculados(String usuario, String senha, String veiplaca, String veioId)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuário	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña actual del integrador para acceso al sistema.
veiplaca	String	Sí	Patente a consultar.
veioId	String	Sí	Identificación de la patente a consultar.

Retorno

En caso de éxito retorna una lista de objetos del tipo "ObterAlertasAVDVinculados" con las propiedades (Acoes, idAlerta, Login, nomeAlerta) — como se muestra en la tabla a continuación:

Nombre	Tipo	Descripción
acoes	String	Lista de acciones asociadas a la alerta.
idAlerta	Integer	ID de la alerta vinculada.
login	String	Usuario responsable de la asociación de la alerta al vehículo.
nomeAlerta	String	Descripción de la alerta asociada.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:ObterAlertasAVDVinculados>
      <usuario>Usuário</usuario>
      <senha>ValorSenha</senha>
      <!--Optional:-->
      <veiplaca>ValorVeiculoPlaca</veiplaca>
      <!--Optional:-->
      <veioId>ValorVeiculoID</veioId>
    </web:ObterAlertasAVDVinculados>
  </soapenv:Body>
</soapenv:Envelope>
```

```
</web:ObterAlertasAVDVinculados>  
</soapenv:Body>  
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">  
  <S:Body>  
    <ns0:obterAlertasAVDVinculadosResponse  
      xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">  
      <return>  
        <acoes>513 I oi</acoes>  
        <idAlerta>36031</idAlerta>  
        <login>ADM</login>  
        <nomeAlerta>517 E 1 </nomeAlerta>  
      </return>  
    </ns0:obterAlertasAVDVinculadosResponse>  
  </S:Body>  
</S:Envelope>
```

3.3. ObterGrupoAtuadores

Descripción

Método para consultar la relación de sensores, actuadores y eventos disponibilizados por Sascar.

Declaración del Método

```
List obterGrupoAtuadores(String usuario, String senha)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuário	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.

Retorno

Retorna una lista de objetos del tipo 'GrupoAtuador' con las propiedades a continuación:

Nombre	Tipo	Descripción
idAtuador	Integer	Código del actuador.
descricao	String	Nombre del actuador, sensor, accesorio o evento.
tipoPorta	String	Tipo de actuador: • S — Salida • E — Entrada • V — Evento



La información obtenida puede utilizarse para alimentar la tabla 'atuador_grupo' de la base de datos de integración.

A continuación se detalla la lista de datos enviados por el método en cuestión, los cuales se utilizan para señalar la ocurrencia de eventos en el método ObterPacotePosicoes.

Código	Descripción	Tipo	Observación
251	Sensor Caja Trasera	Entrada	

Código	Descripción	Tipo	Observación
541	Falla Teclado TD50	Entrada	
249	Sensor Caja	Entrada	
246	Sensor Puertas	Entrada	
250	Sensor Caja Lateral	Entrada	
241	Desenganche	Entrada	
218	Sensor Puerta Intermedia	Entrada	
217	Sensor Limpiaparabrisas	Entrada	
216	Sensor Linterna	Entrada	
215	Sensor Cofre de Motor	Entrada	
227	Alimentación Semirremolque	Entrada	
214	Sensor de Hormigonera	Entrada	
206	Sensor Tapa de Combustible	Entrada	
207	Sensor Ventanas	Entrada	
229	Violación Sascarreta	Entrada	

Código	Descripción	Tipo	Observación
517	Pánico	Entrada	Indica accionamiento del botón de pánico.
248	Sensor Puerta Acompañante	Entrada	
231	Violación Panel	Entrada	
247	Sensor Puerta Conductor	Entrada	
232	Buzzer	Salida	
252	Seguro Caja	Salida	
253	Seguro Caja Lateral	Salida	
254	Seguro Caja Trasera	Salida	
211	Buzzer - Segunda Instalación	Salida	
245	Seguro 5.ª Rueda	Salida	
240	Sirena	Salida	
659	Pre Sleep	Evento	
658	Ancora	Evento	Indica entrada o salida de geocerca.
509	Exceso de Tiempo Detenido	Evento	
1002	Puntos de referencia	Evento	
563	Exceso Tiempo Detenido	Evento	
562	Conductor bajo Coacción	Evento	

Código	Descripción	Tipo	Observación
555	Grupo de Puntos	Evento	Indica entrada o salida de un grupo de puntos.
553	Área	Evento	Indica entrada o salida de área.
652	Llegada a Punto	Evento	
653	Salida de Punto	Evento	
662	Prórroga de exceso de tiempo detenido	Evento	
660	Contraseña a Conductor	Evento	
674	Falla en la activación automática del Seguro 5.ª Rueda	Evento	Indica que ocurrió una falla al intentar reactivar (trabar) el Seguro de 5.ª Rueda. Obs.: Evento enviado sólo por el Seguro de 5.ª Rueda Inteligente.
675	Violación 5.ª Rueda	Evento	Indica que hubo un intento de violación del Seguro de 5.ª Rueda. Obs.: Evento enviado sólo por el Seguro de 5.ª Rueda inteligente.
238	Teclado TMCD	Evento	Indica el tipo de teclado.
224	Teclado TD50	Evento	Indica el tipo de teclado.
550	Teclado TD50	Evento	Indica el tipo de teclado.
202	Teclado SasMDT	Evento	Indica el tipo de teclado.
235	Teclado TD-40	Evento	Indica el tipo de teclado.
200	Teclado SasMDT	Evento	Indica el tipo de teclado.
204	Sensor de humedad y temperatura	Entrada	

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ws="http://ws.integra.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:obterGrupoAtuadores>
      <usuario>teste</usuario>
      <senha>teste</senha>
    </ws:obterGrupoAtuadores>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:obterGrupoAtuadoresResponse xmlns:ns2="http://ws.integra.sascar.com.br/">
      <return>
        <descricao>Panico Desativado</descricao>
        <idAtuador>517</idAtuador>
        <tipoPorta>E</tipoPorta>
      </return>
    </ns2:obterGrupoAtuadoresResponse>
  </S:Body>
</S:Envelope>
```

3.4. ObterCadastroAlertasAVD

Descripción

Método de consulta que retorna el registro de alertas AVD de los vehículos.

Declaración del Método

```
String obterCadastroAlertasAVD(String usuario, String senha, String dataInicio)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
dataInicio	String	No	Fecha de inicio para la consulta.

Retorno

Retorna una lista de objetos del tipo "ObterCadastroAlertasAVD" con las propiedades a continuación:

Nombre	Tipo	Descripción
Evento	String	Descripción de la alerta vinculada.
Id	String	ID de la alerta vinculada.
Login	String	Login responsable de la asociación de la alerta al vehículo.
Timezone	String	Zona horaria de la alerta.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterCadastroAlertasAvd>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <dataInicio>2021-10-01 00:00:00</dataInicio>
    </web:obterCadastroAlertasAvd>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterCadastroAlertasAvdResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <evento>254 0 1</evento>
        <id>62228</id>
        <login>ADM</login>
        <timezone>America/Sao_Paulo</timezone>
      </return>
    </ns0:obterCadastroAlertasAvdResponse>
  </S:Body>
</S:Envelope>
```

3.5. ObterStatusComando

Descripción

Método para consultar el estado de los comandos enviados a través del Web Service XML-RPC.

Declaración del Método

```
StatusComando obterStatusComando(String usuario, String senha, Integer ticket)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
ticket	Integer	Sí	Número del ticket "EXTERNO" del comando enviado. Este ticket es informado por la integradora al momento del envío del comando.

Retorno

Retorna una lista de objetos del tipo 'StatusComando' con las propiedades a continuación:

Nombre	Tipo	Descripción
idStatusComando	Integer	ID del estado del comando. Este ID es el mismo registrado en la tabla 'status_comando' donde se encuentran los posibles estados de un comando: • 1 — Comando expirado • 2 — Error general • 100 — Ejecución del comando vía GPRS • 101 — Ejecución del comando vía SATELITAL • 102 — Comando cancelado • 103 — Comando pendiente
dataExec	Calendar	Fecha de ejecución/cancelación/expiración del comando.



La información obtenida puede utilizarse para alimentar la tabla 'comando' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterStatusComando>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <ticket>NumeroTicket</ticket>
    </web:obterStatusComando>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterStatusComandoResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <dataExec>2023-11-06T11:35:23.0</dataExec>
        <idStatusComando>100</idStatusComando>
      </return>
      <return>
        <dataExec>2023-11-06T11:34:09.0</dataExec>
        <idStatusComando>100</idStatusComando>
      </return>
      <return>
        <dataExec>2023-11-06T11:33:53.0</dataExec>
        <idStatusComando>100</idStatusComando>
      </return>
      <return>
        <dataExec>2023-11-06T11:33:27.0</dataExec>
        <idStatusComando>100</idStatusComando>
      </return>
      <return>
        <dataExec>2023-11-06T11:21:34.0</dataExec>
        <idStatusComando>100</idStatusComando>
      </return>
    </ns0:obterStatusComandoResponse>
  </S:Body>
</S:Envelope>
```



Cuando se envía el comando XML ***inibir_sensor*** no retornará el ***idStatusComando***, dado que el comando no va al servidor de comandos, ya que se trata de un estado de análisis de cuando la posición del equipo llega al BD. Este comando no se envía al equipo.

3.6. ObterStatusComandoTicketSascar

Descripción

Método para consultar el estado de los comandos enviados a través del Web Service XML-RPC.

Declaración del Método

```
StatusComando obterStatusComando(String usuario, String senha, Integer ticket)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
ticket	Integer	Sí	Número del ticket "INTERNO" del comando enviado. Este ticket es generado e informado por Sascar al momento del envío de un comando.

Retorno

Retorna una lista de objetos del tipo 'StatusComando' con las propiedades a continuación:

Nombre	Tipo	Descripción
idStatusComando	Integer	ID del estado del comando. Este ID es el mismo registrado en la tabla 'status_comando' donde se encuentran los posibles estados de un comando: • 1 — Comando expirado • 2 — Error general • 100 — Ejecución del comando vía GPRS • 101 — Ejecución del comando vía SATELITAL • 102 — Comando cancelado • 103 — Comando pendiente
dataExec	Calendar	Fecha de ejecución/cancelación/expiración del comando.



La información obtenida puede utilizarse para alimentar la tabla 'comando' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterStatusComandoTicketSascar>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <ticket>Ticket</ticket>
    </web:obterStatusComandoTicketSascar>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterStatusComandoTicketSascarResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      </S:Body>
</S:Envelope>
```

3.7. ObterTipoComando

Descripción

Método para consultar la relación de comandos que el INTEGRADOR puede enviar vía XMLRPC a sus vehículos.

Declaración del Método

```
List obterTipoComando(String usuario, String senha)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.

Retorno

Retorna una lista de objetos del tipo 'TipoComando' con las propiedades a continuación:

Nombre	Tipo	Descripción
idTipoComando	Integer	Código del comando.
nome	String	Nombre del comando.
descricao	String	Descripción resumida del comando.



La información obtenida puede utilizarse para alimentar la tabla 'comando_tipo' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterTipoComando>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
    </web:obterTipoComando>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterTipoComandoResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <descricao>Bloqueio de Veículo</descricao>
        <idTipoComando>2</idTipoComando>
        <nome>Bloqueio</nome>
      </return>
    </ns0:obterTipoComandoResponse>
  </S:Body>
</S:Envelope>
```

3.8. ObterSequenciamentoEvento

Descripción

Método para consultar la relación de los posibles eventos de secuenciamiento.

Declaración del Método

```
List obterSequenciamentoEvento(String usuario, String senha)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del usuario para acceso al sistema.
senha	String	Sí	Contraseña del usuario para acceso al sistema.

Retorno

Retorna una lista de objetos del tipo 'SequenciamentoEvento' con las propiedades a continuación:

Nombre	Tipo	Descripción
idSequenciamentoEvento	Integer	Identificación del evento de secuenciamiento de macros.
atuador	Integer	Actuador al que se refiere (cuando corresponda).
descricao	String	Descripción del evento de secuenciamiento de macros.



La información obtenida puede utilizarse para alimentar la tabla 'sequenciamento_evento' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterSequenciamentoEvento>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
    </web:obterSequenciamentoEvento>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterSequenciamentoEventoResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <atuador>248</atuador>
        <descricao>Sensor Porta Carona</descricao>
        <idSequenciamentoEvento>2031</idSequenciamentoEvento>
      </return>
    </ns0:obterSequenciamentoEventoResponse>
  </S:Body>
</S:Envelope>
```

3.9. ComandoEmbarquePontoDiario

Descripción

Método utilizado para registrar puntos de referencia, vincularlos con un layout de grupo de puntos y, a continuación, embarcar este nuevo layout en el vehículo.

Declaración del Método

```
LogComando comandoEmbarquePontoDiario(String usuario, String senha, Integer idVeiculo, String pontosRef)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idVeiculo	Integer	Sí	ID del vehículo al que se enviará el comando.
pontosRef	String	Sí	Puntos de referencia a registrar. Ver formato a continuación.



Formato del campo pontosRef

NOME DO PONTO|LAT|LONG|RAIO|GRUPO;NOME DO PONTO 2|LAT|LONG|RAIO|GRUPO



El Grupo debe ser 1 o 2.

- Grupo 1 = PUNTO PARADA CLIENTE.
- Grupo 2 = PUNTO DE CHEQUEO.

Retorno

Retorna un objeto del tipo 'LogComando' con las propiedades a continuación:

Nombre	Tipo	Descripción
mensagem	String	Mensaje de retorno del comando.
codigo	String	Identificador (Ticket de consulta) del comando enviado.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/" >
  <soapenv:Header/>
  <soapenv:Body>
    <web:comandoEmbarquePontoDiario>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <idVeiculo>IDVeiculo</idVeiculo>
      <pontosRef> nomeDoPonto | Latitude | Longitude | Raio |1 ou 2</pontosRef>
    </web:comandoEmbarquePontoDiario>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:comandoEmbarquePontoDiarioResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/" >
      <return>
        <codigo>2222222</codigo>
        <mensagem>Comando enviado para execucao!</mensagem>
      </return>
    </ns0:comandoEmbarquePontoDiarioResponse>
  </S:Body>
</S:Envelope>
```

4. Métodos – Vehículos y Clientes

4.1. ObterClientes

Descripción

Método para consultar los datos de registro de clientes pertenecientes al INTEGRADOR.

Declaración del Método

```
List obterClientes(String usuario, String senha, Integer quantidade, Integer idCliente)
```

Se implementó una adecuación con el objetivo de mantener el método obterClientes funcionando para que los clientes que lo utilizan no tengan problemas, y en caso de necesitar CNPJs alfanuméricos deberán utilizar obterClientesV2.

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	No	Cantidad máxima de registros de la consulta, siendo el límite de 1.000 registros. Para obtener la información pertinente a un único registro de cliente, completar el campo quantidade con el valor 1 (uno) e indicar el id del cliente deseado en el campo idCliente.
idCliente	Integer	No	El límite máximo de resultados del método es 1.000 registros. Para obtener el resto de los registros se implementó un sistema de paginación: enviar 0 (cero) para obtener la primera página de 1.000 registros; para obtener la siguiente página, enviar como parámetro el último id de cliente de la última página consultada. Para obtener la información pertinente a un único registro de cliente, enviar sólo el id del cliente deseado y en el campo quantidade el valor 1 (uno).

Retorno

Retorna una lista de objetos del tipo 'Cliente' con las propiedades a continuación:

Nombre	Tipo	Descripción
idCliente	Integer	Código del cliente.
nome	String	Nombre del cliente.
CPF	Long	CPF para clientes persona física.

Nombre	Tipo	Descripción
CNPJ	Long	CNPJ para clientes persona jurídica; retorna CNPJ sólo numérico. Si muestra valor 0, utilice el método obterClientesV2 para obtener el CNPJ en formato alfanumérico.



La información obtenida puede utilizarse para alimentar la tabla 'cliente' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterClientes>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>ValorNumérico</quantidade>
      <!--Optional:-->
      <idCliente>idCliente</idCliente>
    </web:obterClientes>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterClientesResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <cnpj>CNPJVALOR</cnpj>
        <cpf> CPFVALOR </cpf>
        <idCliente> IDCLIENTE </idCliente>
        <nome>NOMECLIENTE</nome>
      </return>
    </ns0:obterClientesResponse>
  </S:Body>
</S:Envelope>
```

4.2. ObterClientesV2

Descripción

Nuevo método para consultar los datos de registro de clientes, con soporte mejorado para CNPJ alfanumérico. Verifica dinámicamente la existencia de la columna en la base de datos. Si está presente y completa, prioriza ese campo para CNPJ; de lo contrario, utiliza el campo convertido a string. Mantiene compatibilidad con bases heredadas y retorna CNPJ como string.

Declaración del Método

```
List obterClientesV2(String usuario, String senha, Integer quantidade, Integer idCliente)
```

Parámetros

Mismos parámetros del método obterClientes (ver sección 4.5).

Retorno

Misma estructura de retorno del método obterClientes, con CNPJ siempre como string (soportando numérico y alfanumérico).



Recomendado para integraciones que necesitan CNPJ alfanumérico. El método obterClientes permanece disponible por compatibilidad.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterClientesV2>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>ValorNumérico</quantidade>
      <!--Optional:-->
      <idCliente>idCliente</idCliente>
    </web:obterClientesV2>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
```

```
</S:Header>
<S:Body>
  <ns0:obterClientesV2Response xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
</S:Body>
</S:Envelope>
```

4.3. ObterVeiculos

Descripción

Método para consultar la información relativa a los vehículos de cada INTEGRADOR.



Si el cliente tiene restricciones, no será posible consultar sus vehículos.

Declaración del Método

```
List obterVeiculos(String usuario, String senha, Integer quantidade, Integer idVeiculo)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	Sí	Cantidad máxima de registros de la consulta, siendo el límite de 1.000 registros incluyendo los que poseen restricción; es decir, pueden retornarse 940 vehículos y los 60 restantes podrían tener alguna restricción. Para obtener la información pertinente a un único registro de vehículo, completar el campo quantidade con el valor 1 (uno) e indicar el id del vehículo deseado en el campo idVeiculo.
idVeiculo	Integer	No	El límite máximo de resultados del método es 1.000 registros. Para obtener el resto de los registros se implementó un sistema de paginación: enviar 0 (cero) como parámetro para obtener la primera página; para obtener la siguiente página, enviar como parámetro el último id de vehículo de la última página consultada. Repetir este proceso hasta que no se retorne ningún vehículo más, ya que el retorno "vacío" indica el fin de la paginación. Para obtener la información pertinente a un único registro de vehículo, enviar sólo el id del vehículo deseado y en el campo quantidade el valor 1 (uno).

Retorno

Retorna una lista de objetos del tipo 'Veiculo' con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Identificación del vehículo.
placa	String	Patente del vehículo.
idCliente	Integer	Identificación del cliente al que pertenece el vehículo.

Nombre	Tipo	Descripción
descricao	String	Descripción del vehículo.
idEquipamento	String	Identificación del equipo instalado en el vehículo.
idSensor1	Integer	Código del sensor instalado en la entrada 1 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
idSensor2	Integer	Código del sensor instalado en la entrada 2 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
idSensor3	Integer	Código del sensor instalado en la entrada 3 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
idSensor4	Integer	Código del sensor instalado en la entrada 4 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
idAtuador1	Integer	Código del actuador instalado en la salida 1 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
idAtuador2	Integer	Código del actuador instalado en la salida 2 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
idAtuador3	Integer	Código del actuador instalado en la salida 3 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
idAtuador4	Integer	Código del actuador instalado en la salida 4 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
portaPanico	Integer	Identificación del puerto donde está instalado el botón de pánico.
portaBloqueio	Integer	Identificación del puerto donde está instalado el botón de bloqueo.
serial0	Integer	Código del periférico instalado en el vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de periféricos disponibles para instalación en los vehículos.

Nombre	Tipo	Descripción
<code>serial1</code>	Integer	Código del periférico instalado en el vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de periféricos disponibles para instalación en los vehículos.
<code>satelital</code>	boolean	Propiedad que indica si el vehículo posee antena satelital: • <code>false</code> — No posee antena satelital • <code>true</code> — Posee antena satelital
<code>idSensor5</code>	Integer	Código del sensor instalado en la entrada 5 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>idSensor6</code>	Integer	Código del sensor instalado en la entrada 6 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>idSensor7</code>	Integer	Código del sensor instalado en la entrada 7 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>idSensor8</code>	Integer	Código del sensor instalado en la entrada 8 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>idAtuador5</code>	Integer	Código del actuador instalado en la salida 5 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
<code>idAtuador6</code>	Integer	Código del actuador instalado en la salida 6 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
<code>idAtuador7</code>	Integer	Código del actuador instalado en la salida 7 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
<code>idAtuador8</code>	Integer	Código del actuador instalado en la salida 8 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
<code>idEquipamentoDesc</code>	String	Descripción del ID del equipo (MTC700, MTC600, MTC550, MXT140A, etc.)



La información obtenida puede utilizarse para alimentar la tabla 'veiculo' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterVeiculos>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>Quantidade</quantidade>
      <!--Optional:-->
      <idVeiculo>idVeiculo</idVeiculo>
    </web:obterVeiculos>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterVeiculosResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <descricao>GSM/GPS</descricao>
        <idAtuador1>0</idAtuador1>
        <idAtuador2>0</idAtuador2>
        <idAtuador3>0</idAtuador3>
        <idAtuador4>0</idAtuador4>
        <idAtuador5>0</idAtuador5>
        <idAtuador6>0</idAtuador6>
        <idAtuador7>0</idAtuador7>
        <idAtuador8>0</idAtuador8>
        <idCliente>114208</idCliente>
        <idEquipamento>75807</idEquipamento>
        <idEquipamentoDesc>MTC400</idEquipamentoDesc>
        <idSensor1>0</idSensor1>
        <idSensor2>0</idSensor2>
        <idSensor3>0</idSensor3>
        <idSensor4>0</idSensor4>
        <idSensor5>0</idSensor5>
        <idSensor6>0</idSensor6>
        <idSensor7>0</idSensor7>
        <idSensor8>0</idSensor8>
        <idSerial0>237</idSerial0>
        <idSerial1>0</idSerial1>
        <idVeiculo>53106</idVeiculo>
        <placa>DMH8074</placa>
        <portaBloqueio>1</portaBloqueio>
        <portaPanico>0</portaPanico>
        <satelital>false</satelital>
        <telemetria>false</telemetria>
      </return>
    </ns0:obterVeiculosResponse>
  </S:Body>
</S:Envelope>
```

4.4. ObterVeiculosJSON

Descripción

Método para consultar la información relativa a los vehículos de cada INTEGRADOR. Su retorno es en formato JSON.



Si el cliente tiene restricciones, no será posible consultar sus vehículos.

Declaración del Método

```
List obterVeiculos(String usuario, String senha, Integer quantidade, Integer idVeiculo)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	Sí	Cantidad máxima de registros de la consulta, siendo el límite de 1.000 registros incluyendo los que poseen restricción; es decir, pueden retornarse 940 vehículos y los 60 restantes podrían tener alguna restricción. Para obtener la información pertinente a un único registro de vehículo, completar el campo quantidade con el valor 1 (uno) e indicar el id del vehículo deseado en el campo idVeiculo.
idVeiculo	Integer	No	El límite máximo de resultados del método es 1.000 registros. Para obtener el resto de los registros se implementó un sistema de paginación: enviar 0 (cero) para obtener la primera página; para obtener la siguiente página, enviar como parámetro el último id de vehículo de la última página consultada. Repetir este proceso hasta que no se retorne ningún vehículo más, ya que el retorno "vacío" indica el fin de la paginación. Para obtener la información pertinente a un único registro de vehículo, enviar sólo el id del vehículo deseado y en el campo quantidade el valor 1 (uno).

Retorno

Retorna una lista de objetos del tipo 'Veiculo' con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Identificación del vehículo.
placa	String	Patente del vehículo.

Nombre	Tipo	Descripción
idCliente	Integer	Identificación del cliente al que pertenece el vehículo.
descricao	String	Descripción de la clase del vehículo. Ej.: SASTM FULL, SASTM COM TELEMETRIA, SASCARGA FULL SAT 200, SASCARGA FULL SAT I.
idEquipamento	String	Identificación del equipo instalado en el vehículo.
idSensor1	Integer	Código del sensor instalado en la entrada 1 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
idSensor2	Integer	Código del sensor instalado en la entrada 2 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
idSensor3	Integer	Código del sensor instalado en la entrada 3 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
idSensor4	Integer	Código del sensor instalado en la entrada 4 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
idAtuador1	Integer	Código del actuador instalado en la salida 1 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
idAtuador2	Integer	Código del actuador instalado en la salida 2 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
idAtuador3	Integer	Código del actuador instalado en la salida 3 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
idAtuador4	Integer	Código del actuador instalado en la salida 4 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
portaPanico	Integer	Identificación del puerto donde está instalado el botón de pánico.
portaBloqueio	Integer	Identificación del puerto donde está instalado el botón de bloqueo.

Nombre	Tipo	Descripción
<code>serial0</code>	Integer	Código del periférico instalado en el vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de periféricos disponibles para instalación en los vehículos.
<code>serial1</code>	Integer	Código del periférico instalado en el vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de periféricos disponibles para instalación en los vehículos.
<code>satelital</code>	boolean	Propiedad que indica si el vehículo posee antena satelital: • <code>false</code> — No posee antena satelital • <code>true</code> — Posee antena satelital
<code>idSensor5</code>	Integer	Código del sensor instalado en la entrada 5 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>idSensor6</code>	Integer	Código del sensor instalado en la entrada 6 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>idSensor7</code>	Integer	Código del sensor instalado en la entrada 7 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>idSensor8</code>	Integer	Código del sensor instalado en la entrada 8 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>idAtuador5</code>	Integer	Código del actuador instalado en la salida 5 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
<code>idAtuador6</code>	Integer	Código del actuador instalado en la salida 6 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
<code>idAtuador7</code>	Integer	Código del actuador instalado en la salida 7 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
<code>idAtuador8</code>	Integer	Código del actuador instalado en la salida 8 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.

Nombre	Tipo	Descripción
<code>idEquipamentoDesc</code>	String	Descripción del ID del equipo (MTC700, MTC600, MTC550, MXT140A, etc.)



La información obtenida puede utilizarse para alimentar la tabla 'veiculo' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterVeiculosJson>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>Quantidade</quantidade>
      <!--Optional:-->
      <idVeiculo>idVeiculo</idVeiculo>
    </web:obterVeiculosJson>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterVeiculosJsonResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        {
          "idVeiculo": 53106,
          "placa": "DMH8074",
          "idCliente": 114208,
          "descricao": "GSM/GPS",
          "idEquipamento": 75807,
          "idEquipamentoDesc": "MTC400",
          "idSensor1": 0,
          "idSensor2": 0,
          "idSensor3": 0,
          "idSensor4": 0,
          "idAtuador1": 0,
          "idAtuador2": 0,
          "idAtuador3": 0,
          "idAtuador4": 0,
          "portaPanico": 0,
          "portaBloqueio": 1,
          "idSerial0": 237,
          "idSerial1": null,
          "satelital": false,
          "idSensor5": 0,
          "idSensor6": 0,
        }
      </return>
    </ns0:obterVeiculosJsonResponse>
  </S:Body>
</S:Envelope>
```

```
        "idSensor7": 0,  
        "idSensor8": 0,  
        "idAtuador5": 0,  
        "idAtuador6": 0,  
        "idAtuador7": 0,  
        "idAtuador8": 0,  
        "esn": "null",  
        "idProjeto": null,  
        "isTelemetria": false  
    }  
    </return>  
</ns0:obterVeiculosJsonResponse>  
</S:Body>  
</S:Envelope>
```

4.5. ObterVeiculosRFNacional

Descripción

Método para consultar la información relativa a los vehículos de cada INTEGRADOR.



Si el cliente tiene restricciones, no será posible consultar sus vehículos.

Declaración del Método

```
List obterVeiculosRFNacional(String usuario, String senha, Integer quantidade, Integer idVeiculo)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	Sí	Cantidad máxima de registros de la consulta, siendo el límite de 1.000 registros incluyendo los que poseen restricción; es decir, pueden retornarse 940 vehículos y los 60 restantes podrían tener alguna restricción. Para obtener la información pertinente a un único registro de vehículo, completar el campo quantidade con el valor 1 (uno) e indicar el id del vehículo deseado en el campo idVeiculo.
idVeiculo	Integer	No	El límite máximo de resultados del método es 1.000 registros. Para obtener el resto de los registros se implementó un sistema de paginación: enviar 0 (cero) para obtener la primera página; para obtener la siguiente página, enviar como parámetro el último id de vehículo de la última página consultada. Repetir este proceso hasta que no se retorne ningún vehículo más, ya que el retorno "vacío" indica el fin de la paginación. Para obtener la información pertinente a un único registro de vehículo, enviar sólo el id del vehículo deseado y en el campo quantidade el valor 1 (uno).

Retorno

Retorna una lista de objetos del tipo 'Veiculo' con las propiedades a continuación:

Nombre	Tipo	Descripción
ccid	String	ID de la tarjeta SIM.

Nombre	Tipo	Descripción
descricao	String	Descripción de la clase del vehículo. Ej.: SASTM FULL, SASTM COM TELEMETRIA, SASCARGA FULL SAT 200, SASCARGA FULL SAT I.
idCliente	Integer	Identificación del cliente al que pertenece el vehículo.
idVeiculo	Integer	Identificación del vehículo.
placa	String	Patente del vehículo.
satelital	boolean	Indica si contiene posición satelital.
telemetria	boolean	Indica si contiene telemetría.



La información obtenida puede utilizarse para alimentar la tabla 'veiculo_rfnacional' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterVeiculosRFNacional>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade></quantidade>
      <!--Optional:-->
      <idVeiculo></idVeiculo>
    </web:obterVeiculosRFNacional>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterVeiculosRFNacionalResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <ccid>89550532420010181845</ccid>
        <descricao>RF-Nacional</descricao>
        <idCliente>68553</idCliente>
        <idVeiculo>15213</idVeiculo>
        <placa>MRB1053-1</placa>
        <satelital>false</satelital>
        <telemetria>false</telemetria>
      </return>
    </ns0:obterVeiculosRFNacionalResponse>
  </S:Body>
</S:Envelope>
```

4.6. getVehiclesJSON

Descripción

Método para consultar la información relativa a los vehículos de cada INTEGRADOR. Su retorno es en formato JSON. Retorno en inglés.



Si el cliente tiene restricciones, no será posible consultar sus vehículos.

Declaración del Método

```
List getVehiclesJSON(String user, String password, Integer quantity, Integer vehicleId)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
user	String	Sí	Login del integrador para acceso al sistema.
password	String	Sí	Contraseña del integrador para acceso al sistema.
quantity	Integer	Sí	Cantidad máxima de registros de la consulta, siendo el límite de 1.000 registros incluyendo los que poseen restricción; es decir, pueden retornarse 940 vehículos y los 60 restantes podrían tener alguna restricción. Para obtener la información pertinente a un único registro de vehículo, completar el campo quantity con el valor 1 (uno) e indicar el id del vehículo deseado en el campo vehicleId.
vehicleId	Integer	No	El límite máximo de resultados del método es 1.000 registros. Para obtener el resto de los registros se implementó un sistema de paginación: enviar 0 (cero) para obtener la primera página; para obtener la siguiente página, enviar como parámetro el último id de vehículo de la última página consultada. Repetir este proceso hasta que no se retorne ningún vehículo más, ya que el retorno "vacío" indica el fin de la paginación. Para obtener la información pertinente a un único registro de vehículo, enviar sólo el id del vehículo deseado y en el campo quantity el valor 1 (uno).

Retorno

Retorna una lista de objetos del tipo 'Veiculo' con las propiedades a continuación:

Nombre	Tipo	Descripción
vehicleId	Integer	Identificación del vehículo.
licensePlate	String	Patente del vehículo.

Nombre	Tipo	Descripción
clientId	Integer	Identificación del cliente al que pertenece el vehículo.
description	String	Descripción de la clase del vehículo. Ej.: SASTM FULL , SASTM COM TELEMETRIA , SASCARGA FULL SAT 200 , SASCARGA FULL SAT I .
deviceId	String	Identificación del equipo instalado en el vehículo.
deviceDescId	String	Descripción del ID del equipo (MTC700, MTC600, MTC550, MXT140A, etc.)
sensor1Id	Integer	Código del sensor instalado en la entrada 1 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
sensor2Id	Integer	Código del sensor instalado en la entrada 2 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
sensor3Id	Integer	Código del sensor instalado en la entrada 3 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
sensor4Id	Integer	Código del sensor instalado en la entrada 4 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
actuator1Id	Integer	Código del actuador instalado en la salida 1 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
actuator2Id	Integer	Código del actuador instalado en la salida 2 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
actuator3Id	Integer	Código del actuador instalado en la salida 3 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
actuator4Id	Integer	Código del actuador instalado en la salida 4 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
panicPort	Integer	Identificación del puerto donde está instalado el botón de pánico.
blockingPort	Integer	Identificación del puerto donde está instalado el botón de bloqueo.

Nombre	Tipo	Descripción
<code>serial0Id</code>	Integer	Código del periférico instalado en el vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de periféricos disponibles para instalación en los vehículos.
<code>serial1Id</code>	Integer	Código del periférico instalado en el vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de periféricos disponibles para instalación en los vehículos.
<code>layoutId</code>	Integer	Retorna null.
<code>satellite</code>	boolean	Propiedad que indica si el vehículo posee antena satelital: • <code>false</code> — No posee antena satelital • <code>true</code> — Posee antena satelital
<code>sensor5Id</code>	Integer	Código del sensor instalado en la entrada 5 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>sensor6Id</code>	Integer	Código del sensor instalado en la entrada 6 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>sensor7Id</code>	Integer	Código del sensor instalado en la entrada 7 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>sensor8Id</code>	Integer	Código del sensor instalado en la entrada 8 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de sensores disponibles para instalación en los vehículos.
<code>actuator5Id</code>	Integer	Código del actuador instalado en la salida 5 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
<code>actuator6Id</code>	Integer	Código del actuador instalado en la salida 6 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
<code>actuator7Id</code>	Integer	Código del actuador instalado en la salida 7 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método <code>ObterAtuadorGrupo()</code> , cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.

Nombre	Tipo	Descripción
actuator8Id	Integer	Código del actuador instalado en la salida 8 del vehículo. Este id está relacionado con la tabla 'atuador_grupo' o con el resultado del método ObterAtuadorGrupo(), cuyo retorno es la lista de actuadores disponibles para instalación en los vehículos.
esn	String	Retorna null.
projectId	Integer	Retorna null.
isTelemetry	String	Indica si el vehículo posee telemetría ("true", "false").



La información obtenida puede utilizarse para alimentar la tabla 'veiculo' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/" >
  <soapenv:Header/>
  <soapenv:Body>
    <web:getVehiclesJSON>
      <user>Usuário</user>
      <password>Senha</password>
      <!--Optional:-->
      <quantity>Quantidade</quantity>
      <!--Optional:-->
      <vehicleId>IDVEICULO</vehicleId>
    </web:getVehiclesJSON>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:getVehiclesJSONResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        {
          "vehicleId": 53106,
          "licensePlate": "DMH8074",
          "clientId": 114208,
          "description": "GSM/GPS",
          "deviceId": 75807,
          "deviceDescId": "MTC400",
          "sensor1Id": 0,
          "sensor2Id": 0,
          "sensor3Id": 0,
          "sensor4Id": 0,
          "actuator1Id": 0,
          "actuator2Id": 0,
```

```
        "actuator3Id": 0,  
        "actuator4Id": 0,  
        "panicPort": 0,  
        "blockingPort": 1,  
        "serial0Id": 237,  
        "serial1Id": 0,  
        "layoutId": null,  
        "satellite": false,  
        "sensor5Id": 0,  
        "sensor6Id": 0,  
        "sensor7Id": 0,  
        "sensor8Id": 0,  
        "actuator5Id": 0,  
        "actuator6Id": 0,  
        "actuator7Id": 0,  
        "actuator8Id": 0,  
        "esn": "null",  
        "projectId": null,  
        "isTelemetry": false  
    }  
    </return>  
</ns0:getVehiclesJSONResponse>  
</S:Body>  
</S:Envelope>
```

4.7. ObterDadosAdicionais

Descripción

Método para consultar la información sobre notas registradas en los vehículos.

Declaración del Método

```
List obterDadosAdicionais(String usuario, String senha, Integer idVeiculo)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idVeiculo	Integer	No	Para obtener la página con todos los vehículos que tienen notas registradas, enviar 0 (cero) como parámetro; para obtener la información pertinente a un único registro de vehículo, enviar sólo el id del vehículo deseado.

Retorno

Retorna una lista de objetos del tipo 'Veiculo' con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Identificación del vehículo.
placa	String	Patente del vehículo.
idCliente	Integer	Identificación del cliente al que pertenece el vehículo.
dataAlteracao	String	Fecha de la última modificación realizada.
notaUm	String	Información registrada en el sasgc para el campo nota 1.
notaDois	String	Información registrada en el sasgc para el campo nota 2.



La información sólo se obtendrá si existe al menos una nota registrada. La información debe ser registrada por la misma gestora de consulta para el vehículo deseado.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
```

```
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterDadosAdicionais>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <idVeiculo>IDVEICULO</idVeiculo>
    </web:obterDadosAdicionais>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterDadosAdicionaisResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <dataAlteracao>2023-08-18 05:50:50</dataAlteracao>
        <idCliente>265107</idCliente>
        <idVeiculo>1462395</idVeiculo>
        <notaDois>nota 2 teste OK</notaDois>
        <notaUm>teste bezerra 3</notaUm>
        <placa>0002MTN</placa>
      </return>
    </ns0:obterDadosAdicionaisResponse>
  </S:Body>
</S:Envelope>
```

4.8. ObterDadosAdicionaisCliente

Descripción

Método para consultar los campos editables por el cliente, para visualización de la gestora.

Declaración del Método

```
List obterDadosAdicionaisCliente(String usuario, String senha, Integer idVeiculo)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idVeiculo	Integer	No	Para obtener la página con todos los vehículos que tienen notas registradas, enviar 0 (cero) como parámetro; para obtener la información pertinente a un único registro de vehículo, enviar sólo el id del vehículo deseado.

Retorno

Retorna una lista de objetos del tipo 'DadosAdicionaisCliente' con las propiedades a continuación:

Nombre	Tipo	Descripción
dataAlteracao	String	Fecha de la última modificación realizada.
descricaoDois	String	Información registrada en el sasgc para el campo descripción 2.
descricaoUm	String	Información registrada en el sasgc para el campo descripción 1.
frota	String	Información registrada en el sasgc para el campo flota.
grupo	String	Información registrada en el sasgc para el campo grupo.
idCliente	Integer	Identificación del cliente al que pertenece el vehículo.
idVeiculo	Integer	Identificación del vehículo.
notaDois	String	Información registrada en el sasgc para el campo nota 2.
notaUm	String	Información registrada en el sasgc para el campo nota 1.
placa	String	Patente del vehículo.



La información sólo se obtendrá si existe al menos una nota registrada. La información debe ser registrada por la misma gestora de consulta para el vehículo deseado.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterDadosAdicionaisCliente>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <idVeiculo>881456</idVeiculo>
    </web:obterDadosAdicionaisCliente>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterDadosAdicionaisClienteResponse
      xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <dataAlteracao>2023-07-11 15:11:49</dataAlteracao>
        <descricaoDois/>
        <descricaoUm/>
        <frota/>
        <grupo/>
        <idCliente>50774</idCliente>
        <idVeiculo>881456</idVeiculo>
        <notaDois>03194998562</notaDois>
        <notaUm/>
        <placa>3002TCP</placa>
      </return>
    </ns0:obterDadosAdicionaisClienteResponse>
  </S:Body>
</S:Envelope>
```

4.9. ObterPontosReferencia

Descripción

Método para consultar los puntos de referencia creados en el SASGC y marcados como embarcables.

Declaración del Método

```
List obterPontosReferencia(String usuario, String senha)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del usuario para acceso al sistema.
senha	String	Sí	Contraseña del usuario para acceso al sistema.

Retorno

Retorna una lista de objetos del tipo 'PontoReferencia' con las propiedades a continuación:

Nombre	Tipo	Descripción
idPontoReferencia	Integer	Identificación del punto de referencia.
codigo	String	Código de Sascar para la visualización del punto.
descricao	String	Descripción del punto.
latitudes	Double	Latitud superior del punto.
longitudes	Double	Longitud superior del punto.
latitudei	Double	Latitud inferior del punto.
longitudei	Double	Longitud inferior del punto.
endereco	String	Dirección del punto creado.
data	Calendar	Fecha de creación del punto en el servidor de integración.
nome	String	Nombre del punto de referencia.



La información obtenida puede utilizarse para alimentar la tabla 'ponto_referencia' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPontosReferencia>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
    </web:obterPontosReferencia>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterPontosReferenciaResponse xmlns:ns0=
"http://webservice.web.integracao.sascar.com.br/">
      <return>
        <codigo>1568</codigo>
        <data>2008-01-27T00:00:00.0</data>
        <descricao>Teste de ponto de referencia para o MTC600.</descricao>
        <endereco/>
        <idPontoReferencia>4008</idPontoReferencia>
        <latitudeI>-25.539593</latitudeI>
        <latitudeS>-25.538414</latitudeS>
        <longitudeI>-49.199693</longitudeI>
        <longitudeS>-49.198265</longitudeS>
        <nome>SASCAR - LITORAL</nome>
      </return>
    </ns0:obterPontosReferenciaResponse>
  </S:Body>
</S:Envelope>
```

4.10. ObterRotas

Descripción

Método para consultar el registro de rutas asociadas a los vehículos.

Declaración del Método

```
String obterRotas(String usuario, String senha, String data)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña actual del integrador para acceso al sistema..
data	String	No	Fecha para la consulta (YYYY-MM-DD HH:MM:DD).

Retorno

Retorna una lista de objetos del tipo "ObterRotas" con las propiedades a continuación:

Nombre	Tipo	Descripción
Login	String	Identificación del login creador de la ruta.
Id	String	Identificación del id de la ruta.
Nome rota	String	Descripción de la ruta.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterRotas>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <dataInicio/>
    </web:obterRotas>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterRotasResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <idrota>992</idrota>
        <loginRota>KIELSE</loginRota>
        <nome>novoll</nome>
      </return>
    </ns0:obterRotasResponse>
  </S:Body>
</S:Envelope>
```

4.11. ObterEnderecoPosicao

Descripción

Método para consultar los datos de dirección mediante coordenadas de latitud y longitud retornadas en un paquete de posición.

Declaración del Método

```
List obterEnderecoPosicao(String usuario, String senha, String latitude, String longitude)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del usuario para acceso al sistema.
senha	String	Sí	Contraseña del usuario para acceso al sistema.
latitude	String	Sí	Coordenada de latitud.
longitude	String	Sí	Coordenada de longitud.

Retorno

Retorna una lista de objetos del tipo 'EnderecoPosicao' con las propiedades a continuación:

Nombre	Tipo	Descripción
cidade	String	Nombre de la ciudad de la dirección de la posición.
Rua	String	Nombre de la calle de la dirección de la posición.
uf	String	Provincia/Estado.



La información obtenida puede utilizarse para alimentar la tabla 'enderecoposicao' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterEnderecoPosicao>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <latitude>-6.91337333333333</latitude>
```

```
<longitude>-37.632421666666666</longitude>
</web:obterEnderecoPosicao>
</soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterEnderecoPosicaoResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <address>
          <city>
            <state>PB</state>
            <name>Condado</name>
          </city>
          <street>Rod BR-230</street>
        </address>
      </return>
    </ns0:obterEnderecoPosicaoResponse>
  </S:Body>
</S:Envelope>
```

5. Métodos – Posición

5.1. ObterPacotePosicoes

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos, cuyo contenido está compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma de Secuenciamiento.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicoes(String usuario, String senha, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	No	Límite de registros como resultado de la consulta. El sistema tiene un límite estándar de 3.000 registros, que se aplicará en los siguientes casos: cuando el cliente no informe este parámetro; cuando el cliente informe un número mayor que el límite estándar; cuando el cliente informe un valor no numérico o menor o igual a 0.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.
dataPosicao	String	Fecha de grabación de la posición en el servidor.
dataPacote	String	Fecha del GPS (GMT) en el momento del envío de la posición.
latitude	Double	Latitud en formato decimal.

Nombre	Tipo	Descripción
longitude	Double	Longitud en formato decimal.
direcao	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MS (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
velocidade	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
ignicao	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
horimetro	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satellite=0", informando que el mensaje fue enviado vía GPRS.
odometro	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
tensao	Integer	Valor de tensión de la batería.
saida1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.

MTC (BIT)	LMU/MS (Grados)	Dirección
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satelite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.

MTC (BIT)	LMU/MSC (Grados)	Dirección
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.

MTC (BIT)	LMU/MSC (Grados)	Dirección
textoMensagem	String	Contenido del mensaje tipo texto libre.
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamiento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:ws="http://webservice.web.integracao.sascar.com.br/"
    <soapenv:Header/>
    <soapenv:Body>
      <ws:obterPacotePosicoes>
        <usuario>Usuário</usuario>
        <senha>Senha</senha>
        <quantidade>valorNumérico</quantidade>
      </ws:obterPacotePosicoes>
    </soapenv:Body>
  </soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterPacotePosicoesResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <anguloReferencia>0</anguloReferencia>
        <bloqueio>0</bloqueio>
        <cidade>Curitiba</cidade>
        <codigoMacro>0</codigoMacro>
        <conteudoMensagem/>
        <dataPacote>2023-10-10T09:05:13.0</dataPacote>
        <dataPosicao>2023-10-10T09:05:14.0</dataPosicao>
        <direcao>189</direcao>
        <distanciaReferencia>0</distanciaReferencia>
        <entrada1>0</entrada1>
        <entrada2>-250</entrada2>
        <entrada3>-251</entrada3>
        <entrada4>-241</entrada4>
        <entrada5>-247</entrada5>
        <entrada6>-248</entrada6>
        <entrada7>-231</entrada7>
        <entrada8>0</entrada8>
        <eventoFormatado/>
        <eventoSeqFormatado/>
        <gps>1</gps>
        <horimetro>1496825</horimetro>
        <idPacote>81204594</idPacote>
        <idReferencia>0</idReferencia>
        <idVeiculo>608581</idVeiculo>
        <ignicao>0</ignicao>
        <integradoraId>80</integradoraId>
        <jamming>0</jamming>
        <latitude>-25.4516521</latitude>
        <longitude>-49.2460098</longitude>
        <memoria>0</memoria>
        <nomeMensagem/>
        <odometro>2662</odometro>
        <pontoEntrada>0</pontoEntrada>
        <pontoReferencia>novo</pontoReferencia>
        <pontoSaida>0</pontoSaida>
        <rpm>0</rpm>
        <rua>R Joao Marchesini</rua>
        <saida1>0</saida1>
        <saida2>-240</saida2>
        <saida3>-254</saida3>
        <saida4>-253</saida4>
        <saida5>0</saida5>
        <saida6>-232</saida6>
```

```
<saida7>-232</saida7>
<saida8>0</saida8>
<satelite>0</satelite>
<temperatura1>15</temperatura1>
<temperatura2>-125</temperatura2>
<temperatura3>-125</temperatura3>
<temperaturaSerial>-1250</temperaturaSerial>
<tensao>11</tensao>
<textoMensagem/>
<tipoTeclado>0</tipoTeclado>
<uf>PR</uf>
<umidadeSerial>-1250</umidadeSerial>
<velocidade>0</velocidade>
</return>
</ns0:obterPacotePosicoesResponse>
</S:Body>
</S:Envelope>
```

5.2. ObterPacotePosicoesMotorista

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos, cuyo contenido está compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma de Secuenciamiento.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicoesMotorista(String usuario, String senha, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	No	Límite de registros como resultado de la consulta. El sistema tiene un límite estándar de 3.000 registros, que se aplicará en los siguientes casos: cuando el cliente no informe este parámetro; cuando el cliente informe un número mayor que el límite estándar.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.
dataPosicao	String	Fecha de grabación de la posición en el servidor.
dataPacote	String	Fecha del GPS (GMT) en el momento del envío de la posición.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.
direcao	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MSC (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
velocidade	Integer	Velocidad obtenida del GPS.
ignicao	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
horimetro	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
odometro	Integer	Odómetro actual del vehículo.
tensao	Integer	Valor de tensión de la batería.
saida1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satelite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).

MTC (BIT)	LMU/MS (Grados)	Dirección
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pais	String	País.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (. <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
textoMensagem	String	Contenido del mensaje tipo texto libre.
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT

MTC (BIT)	LMU/MS (Grados)	Dirección
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicoesMotorista>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>ValorNumérico</quantidade>
    </web:obterPacotePosicoesMotorista>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterPacotePosicoesMotoristaResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <anguloReferencia>134</anguloReferencia>
        <bloqueio>0</bloqueio>
        <cidade>Campo Grande</cidade>
        <codigoMacro>0</codigoMacro>
        <conteudoMensagem/>
        <dataPacote>2023-10-09T11:02:21.0</dataPacote>
        <dataPosicao>2023-10-09T11:02:27.0</dataPosicao>
        <direcao>0</direcao>
        <distanciaReferencia>8563</distanciaReferencia>
        <entrada1>0</entrada1>
        <entrada2>0</entrada2>
        <entrada3>0</entrada3>
        <entrada4>0</entrada4>
        <entrada5>0</entrada5>
        <entrada6>0</entrada6>
        <entrada7>0</entrada7>
        <entrada8>0</entrada8>
        <estadoLimpadorParabrisa>0</estadoLimpadorParabrisa>
        <eventoFormatado/>
        <eventoSeqFormatado/>
        <eventosTelemetria/>
        <gps>1</gps>
        <horimetro>43807</horimetro>
        <idMotorista>0</idMotorista>
        <idPacote>81182489</idPacote>
        <idReferencia>0</idReferencia>
        <idVeiculo>1894048</idVeiculo>
        <ignicao>0</ignicao>
        <integradoraId>80</integradoraId>
        <jamming>0</jamming>
        <latitude>-20.5580288</latitude>
        <longitude>-54.6784256</longitude>
        <memoria>0</memoria>
        <nomeMensagem/>
        <nomeMotorista/>
        <odometro>35541</odometro>
        <pontoEntrada>0</pontoEntrada>
        <pontoReferencia>PE - Couro Azul - Campo Grande, MS</pontoReferencia>
        <pontoSaida>0</pontoSaida>
        <rpm>0</rpm>
        <rua>Sem nome</rua>
        <saida1>0</saida1>
        <saida2>0</saida2>
        <saida3>0</saida3>
        <saida4>0</saida4>
        <saida5>0</saida5>
        <saida6>0</saida6>
        <saida7>0</saida7>
        <saida8>0</saida8>
        <satelite>0</satelite>
        <temperatura1>-125</temperatura1>
        <temperatura2>-125</temperatura2>
        <temperatura3>-125</temperatura3>
        <tensao>25</tensao>
        <textoMensagem/>
      </return>
    </ns0:obterPacotePosicoesMotoristaResponse>
  </S:Body>
```

</S:Envelope>

5.3. ObterPacotePosicoesMotoristaComPlaca

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos. Además de traer la información del método ObterPacotePosicoes, este método también trae la información del Conductor (ID y Nombre) y el estado del Limpiaparabrisas, cuyo contenido está compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma de Secuenciamiento.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicoesMotoristaComPlaca(String usuario, String senha, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	No	Límite de registros como resultado de la consulta. El sistema tiene un límite estándar de 3.000 registros, que se aplicará en los siguientes casos: cuando el cliente no informe este parámetro; cuando el cliente informe un número mayor que el límite estándar.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.
dataPosicao	String	Fecha de grabación de la posición en el servidor.
dataPacote	String	Fecha del GPS (GMT) en el momento del envío de la posición.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.

Nombre	Tipo	Descripción
direcao	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MS (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
velocidade	Integer	Velocidad obtenida del GPS.
ignicao	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
horimetro	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
odometro	Integer	Odómetro actual del vehículo.
tensao	Integer	Valor de tensión de la batería.
saida1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo “2” (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satelite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria

MTC (BIT)	LMU/MSC (Grados)	Dirección
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pais	String	País.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
textoMensagem	String	Contenido del mensaje tipo texto libre.

MTC (BIT)	LMU/MS (Grados)	Dirección
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamiento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
idMotorista	Integer	Identificación del conductor. (0 si no hay conductor)
nomeMotorista	String	Nombre del conductor si está conectado.
limpadorParabrisa	Integer	Estado del limpiaparabrisas: • 0 — Limpiaparabrisas No Accionado • 1 — Limpiaparabrisas Accionado
eventosTelemetria	Array<int>	Lista de evento(s) de telemetría compuesta por los campos: • idEvento — Identificador del Evento • tempoDuracao — Duración del Evento en segundos • velocidadeMaximaEvento — Velocidad Máxima alcanzada en el Evento • velocidadeReferencia — Velocidad de Referencia
placa	String	Patente del vehículo.

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicoesMotoristaComPlaca>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>ValorNumérico</quantidade>
    </web:obterPacotePosicoesMotoristaComPlaca>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <work:WorkContext xmlns:work="http://oracle.com/weblogic/soap/workarea/"
XYZ</work:WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterPacotePosicoesMotoristaComPlacaResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <anguloReferencia>0</anguloReferencia>
        <bloqueio>0</bloqueio>
        <cidade>Curitiba</cidade>
        <codigoMacro>0</codigoMacro>
        <conteudoMensagem/>
        <dataPacote>2018-05-04T06:20:16.0</dataPacote>
        <dataPosicao>2018-05-04T06:20:37.0</dataPosicao>
        <direcao>102</direcao>
        <distanciaReferencia>0</distanciaReferencia>
        <entrada1>0</entrada1>
        <entrada2>0</entrada2>
        <entrada3>0</entrada3>
        <entrada4>-241</entrada4>
        <entrada5>-247</entrada5>
        <entrada6>-248</entrada6>
        <entrada7>-231</entrada7>
        <entrada8>0</entrada8>
        <estadoLimpadorParabrisa>0</estadoLimpadorParabrisa>
```

```

<eventoFormatado/>
<eventoSeqFormatado/>
<eventosTelemetria>
  <eventoTelemetria>
    <idEvento>245</idEvento>
    <tempoDuracao>100</tempoDuracao>
    <velocidadeMaximaEvento>100</velocidadeMaximaEvento>
    <velocidadeReferencia>23</velocidadeReferencia>
  </eventoTelemetria>
  <eventoTelemetria>
    <idEvento>24</idEvento>
    <tempoDuracao>100</tempoDuracao>
    <velocidadeMaximaEvento>100</velocidadeMaximaEvento>
    <velocidadeReferencia>40</velocidadeReferencia>
  </eventoTelemetria>
</eventosTelemetria>
<gps>0</gps>
<horimetro>123003</horimetro>
<idMotorista>0</idMotorista>
<idPacote>1315843</idPacote>
<idReferencia>0</idReferencia>
<idVeiculo>406575</idVeiculo>
<ignicao>1</ignicao>
<integradoraId>80</integradoraId>
<jamming>0</jamming>
<latitude>-25.451593</latitude>
<longitude>-49.2465991</longitude>
<memoria>0</memoria>
<nomeMensagem/>
<nomeMotorista/>
<odometro>50000</odometro>
<placa>ABCD1239</placa>
<pontoEntrada>0</pontoEntrada>
<pontoReferencia/>
<pontoSaida>0</pontoSaida>
<rpm>0</rpm>
<rua>Rua Joao Marchesini</rua>
<saida1>0</saida1>
<saida2>-240</saida2>
<saida3>0</saida3>
<saida4>0</saida4>
</return>
</ns0:obterPacotePosicoesMotoristaComPlacaResponse>
</S:Body>
</S:Envelope>

```

```
0 0 0 0 23 -125 2 24 0 PR 0 </ns0: obterPacotePosicoesMotoristaComPlacaResponse>
```

5.4. ObterPacotePosicoesMotoristaRestricao

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos en el momento en que el cliente tenía restricciones. Además de traer la información del método ObterPacotePosicoesRestricao, este método también trae la información del Conductor (ID y Nombre) y el estado del Limpiaparabrisas, cuyo contenido está compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma de Secuenciamiento. Método exclusivo para paquetes con restricción.

Declaración del Método

```
List obterPacotePosicoesMotoristaRestricao(String usuario, String senha, Integer quantidade, Integer idVeiculo)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	No	Límite de registros como resultado de la consulta. El sistema tiene un límite estándar de 300 registros, que se aplicará en los siguientes casos: cuando el cliente no informe este parámetro; cuando el cliente informe un número mayor que el límite estándar.
idVeiculo	Integer	Sí	Código del vehículo, que puede obtenerse mediante el método obterVeiculos.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.
dataPosicao	String	Fecha de grabación de la posición en el servidor.
dataPacote	Calendar	Fecha del GPS (GMT) en el momento del envío de la posición.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.

Nome	Tipo	Descrição
direcao	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo direcao:

MTC (BIT)	LMU/MSC (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
velocidade	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
ignicao	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
horimetro	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
odometro	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
tensao	Integer	Valor de tensión de la batería.
saida1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo “2” (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satelite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria

MTC (BIT)	LMU/MSC (Grados)	Dirección
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
textoMensagem	String	Contenido del mensaje tipo texto libre.

MTC (BIT)	LMU/MS (Grados)	Dirección
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
idMotorista	Integer	Identificación del conductor.
nomeMotorista	String	Nombre del conductor conectado.
limpadorParabrisa	Integer	Estado del limpiaparabrisas: • 0 — Limpiaparabrisas No Accionado • 1 — Limpiaparabrisas Accionado
eventosTelemetria	Array<int>	Lista de evento(s) de telemetría compuesta por los campos: • idEvento — Identificador del Evento • tempoDuracao — Duración del Evento en segundos • velocidadeMaximaEvento — Velocidad Máxima alcanzada en el Evento • velocidadeReferencia — Velocidad de Referencia



La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de

datos de integración:

- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicoesMotoristaRestricao>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>ValorNumérico</quantidade>
      <!--Optional:-->
      <idVeiculo>IdVeiculo</idVeiculo>
    </web:obterPacotePosicoesMotoristaRestricao>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <work:WorkContext xmlns:work="http://oracle.com/weblogic/soap/workarea/">
      XYZ</work:WorkContext>
    </S:Header>
  <S:Body>
    <ns0:obterPacotePosicoesMotoristaRestricaoResponse
      xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <anguloReferencia>0</anguloReferencia>
        <bloqueio>0</bloqueio>
        <cidade>Curitiba</cidade>
        <codigoMacro>0</codigoMacro>
        <conteudoMensagem/>
        <dataPacote>2018-05-02T13:42:22.0</dataPacote>
        <dataPosicao>2018-05-02T13:42:38.0</dataPosicao>
        <direcao>102</direcao>
        <entrada1>0</entrada1>
        <entrada2>0</entrada2>
        <entrada3>0</entrada3>
        <entrada4>-241</entrada4>
        <entrada5>-247</entrada5>
        <entrada6>-248</entrada6>
        <entrada7>-231</entrada7>
        <entrada8>0</entrada8>
        <estadoLimpadorParabrisa>0</estadoLimpadorParabrisa>
        <eventoFormatado/>
        <eventoSeqFormatado/>
        <eventosTelemetria>
          <eventoTelemetria>
            <idEvento>245</idEvento>
            <tempoDuracao>100</tempoDuracao>
          </eventoTelemetria>
          <eventoTelemetria>
            <idEvento>24</idEvento>
            <tempoDuracao>100</tempoDuracao>
            <velocidadeMaximaEvento>100</velocidadeMaximaEvento>
            <velocidadeReferencia>40</velocidadeReferencia>
          </eventoTelemetria>
        </eventosTelemetria>
        <gps>0</gps>
        <horimetro>120567</horimetro>
        <idMotorista>0</idMotorista>
        <idPacote>1310503</idPacote>
        <idReferencia>0</idReferencia>
        <idVeiculo>406575</idVeiculo>
        <ignicao>1</ignicao>
        <integradoraId>80</integradoraId>
        <jamming>0</jamming>
        <latitude>-25.451593</latitude>
        <longitude>-49.2465991</longitude>
      </return>
    </ns0:obterPacotePosicoesMotoristaRestricaoResponse>
  </S:Body>
</S:Envelope>
```

0 50000 0 0 400 Rua Joao Marchesini 0 -240 0 0 0 0 0 0 23 -125 2 24 0 PR 0

5.5. ObterPacotePosicaoMotoristaPorRange

Descripción

Método que permite obtener uno o más paquetes de posiciones específicos, a través del identificador único de cada paquete. Este método retorna la misma información que el método ObterPacotePosicoesMotorista.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicaoMotoristaPorRange(String usuario, String senha, Integer idInicio, Integer idFinal, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idInicio	Integer	No	Identificador del paquete de posiciones a partir del cual se desea que el web service reenvíe.
idFinal	Integer	Sí	Identificador del paquete de posiciones hasta el cual se desea que el web service reenvíe.
quantidade	Integer	Sí	Límite de registros como resultado de la consulta. El sistema posee un límite estándar de 3000 registros, que se aplicará en los siguientes casos: Cuando el cliente no informe este parámetro; Cuando el cliente informe un número mayor que el límite estándar.



Si se desea recibir solo un paquete de posiciones, se debe informar el mismo idInicio e idFinal.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.

Nombre	Tipo	Descripción
<code>dataPosicao</code>	String	Fecha de grabación de la posición en el servidor.
<code>dataPacote</code>	Calendar	Fecha del GPS (GMT) en el momento del envío de la posición.
<code>latitude</code>	Double	Latitud en formato decimal.
<code>longitude</code>	Double	Longitud en formato decimal.
<code>direcao</code>	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MS (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
<code>velocidade</code>	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
<code>ignicao</code>	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
<code>horimetro</code>	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
<code>odometro</code>	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
<code>tensao</code>	Integer	Valor de tensión de la batería.
<code>saida1</code>	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.

MTC (BIT)	LMU/MS (Grados)	Dirección
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satelite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.

MTC (BIT)	LMU/MS (Grados)	Dirección
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: <i>_João da Silva_</i> .

MTC (BIT)	LMU/MSC (Grados)	Dirección
textoMensagem	String	Contenido del mensaje tipo texto libre.
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamiento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
idMotorista	Integer	Identificación del conductor.
nomeMotorista	String	Nombre del conductor conectado.
limpadorParabrisa	Integer	Estado del limpiaparabrisas: • 0 — Limpiaparabrisas No Accionado • 1 — Limpiaparabrisas Accionado
eventosTelemetria	Array<int>	Lista de evento(s) de telemetría compuesta por los campos: • idEvento — Identificador del Evento • tempoDuracao — Duración del Evento en segundos • velocidadeMaximaEvento — Velocidad Máxima alcanzada en el Evento • velocidadeReferencia — Velocidad de Referencia

MTC (BIT)	LMU/MS (Grados)	Dirección
acessorios	Json	Este campo retorna la lista de accesorios vinculados al componente "Rede de Acessorios". La red de accesorios está preparada para recibir una cantidad indefinida de accesorios que se agregan al rastreador sin necesidad de ocupar los puertos físicos. Cada uno de los sensores envía los siguientes campos: Número de Serie * Tipo Índice Valor Local *El campo serial vendrá vacío y su implementación ocurrirá en un desarrollo futuro. El campo Tipo puede contener los siguientes valores: ENTRADA_DIGITAL 1 SAIDA_DIGITAL 2 TEMPERATURA 3 UMIDADE 4 TENSAO 5 ANALOGICO 6 INDEFINIDO 65.535

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicaoMotoristaPorRange>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <idInicio>PacoteInicial</idInicio>
      <idFinal>PacoteFinal</idFinal>
      <quantidade>ValorNumérico</quantidade>
    </web:obterPacotePosicaoMotoristaPorRange>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterPacotePosicaoMotoristaPorRangeResponse
      xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <idVeiculo>1186796</idVeiculo>
        <idPacote>81181262</idPacote>
        <dataPosicao>2023-10-09T10:08:43.0</dataPosicao>
        <dataPacote>2023-10-09T10:08:42.0</dataPacote>
        <latitude>-20.997711</latitude>
        <longitude>-51.800974</longitude>
        <direcao>176</direcao>
        <velocidade>59</velocidade>
        <ignicao>1</ignicao>
        <horimetro>1602858</horimetro>
        <odometro>464030</odometro>
        <tensao>25</tensao>
        <saida1>0</saida1>
        <saida2>0</saida2>
        <saida3>0</saida3>
        <saida4>0</saida4>
      </return>
    </ns0:obterPacotePosicaoMotoristaPorRangeResponse>
  </S:Body>
</S:Envelope>
```

```

<entrada1>0</entrada1>
<entrada2>0</entrada2>
<entrada3>0</entrada3>
<entrada4>0</entrada4>
<satelite>0</satelite>
<memoria>0</memoria>
<idReferencia>0</idReferencia>
<bloqueio>0</bloqueio>
<gps>1</gps>
<uf>MS</uf>
<cidade>Tres Lagoas</cidade>
<rua>Rod P/ Brasilandia</rua>
<pais/>
<pontoReferencia>PE - PRF. Tres lagoas/MS</pontoReferencia>
<anguloReferencia>176</anguloReferencia>
<distanciaReferencia>20052</distanciaReferencia>
<rpm>0</rpm>
<temperatura1>-125</temperatura1>
<temperatura2>-125</temperatura2>
<temperatura3>-125</temperatura3>
<saida5>0</saida5>
<saida6>0</saida6>
<saida7>0</saida7>
<saida8>0</saida8>
<entrada5>0</entrada5>
<entrada6>0</entrada6>
<entrada7>0</entrada7>
<entrada8>0</entrada8>
<pontoEntrada>0</pontoEntrada>
<pontoSaida>0</pontoSaida>
<codigoMacro>0</codigoMacro>
<nomeMensagem/>
<conteudoMensagem/>
<textoMensagem/>
<tipoTeclado>0</tipoTeclado>
<jamming>0</jamming>
<idMotorista>0</idMotorista>
<nomeMotorista/>
<integradoraId>80</integradoraId>
<eventoFormatado/>
<eventoSeqFormatado/>
<eventosTelemetria/>
</return>
</ns0:obterPacotePosicaoMotoristaPorRangeResponse>
</S:Body>
</S:Envelope>

```

5.6. ObterPacotePosicaoMotoristaHistorico

Descripción

Método para consultar el histórico de paquetes de posiciones enviados por los vehículos. Además de traer la información que el método ObterPacotePosicaoHistorico traía, este método trae también la información del Conductor (ID y Nombre) y el Estado del Limpiaparabrisas, siendo su contenido compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma Secuenciamiento.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicaoMotoristaHistorico(String usuario, String senha, String dataInicio, String dataFinal, Integer idVeiculo)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
dataInicio	String	Sí	Inicio del rango de fecha para la búsqueda.
dataFinal	String	Sí	Fin del rango de fecha para la búsqueda.
idVeiculo	Integer	No	Código del vehículo.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.
dataPosicao	String	Fecha de grabación de la posición en el servidor.
dataPacote	String	Fecha del GPS (GMT) en el momento del envío de la posición.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.

Nombre	Tipo	Descripción
<code>direcao</code>	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MSC (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
<code>velocidade</code>	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
<code>ignicao</code>	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
<code>horimetro</code>	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
<code>odometro</code>	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
<code>tensao</code>	Integer	Valor de tensión de la batería.
<code>saida1</code>	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
<code>saida2</code>	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo “2” (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satelite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria

MTC (BIT)	LMU/MSC (Grados)	Dirección
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
textoMensagem	String	Contenido del mensaje tipo texto libre.

MTC (BIT)	LMU/MS (Grados)	Dirección
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
eventoSequenciamiento	EventoSequenciamiento[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamiento relacionado con la tabla 'sequenciamiento_evento' o con el resultado del método ObterSequenciamientoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
idMotorista	Integer	Identificación del conductor.
nomeMotorista	String	Nombre del conductor conectado.
limpadorParabrisa	Integer	Estado del limpiaparabrisas: • 0 — Limpiaparabrisas No Accionado • 1 — Limpiaparabrisas Accionado
eventosTelemetria	Array<int>	Lista de evento(s) de telemetría compuesta por los campos: • idEvento — Identificador del Evento • tempoDuracao — Duración del Evento en segundos • velocidadeMaximaEvento — Velocidad Máxima alcanzada en el Evento • velocidadeReferencia — Velocidad de Referencia



La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de

datos de integración:

- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicaoMotoristaHistorico>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <dataInicio>YYYY-MM-DD HH:MM:SS</dataInicio>
      <dataFinal>2023-10-09 23:59:00</dataFinal>
      <!--Optional:-->
      <idVeiculo>IDVEICULO</idVeiculo>
    </web:obterPacotePosicaoMotoristaHistorico>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <work:WorkContext xmlns:work="http://oracle.com/weblogic/soap/workarea/">
      XYZ</work:WorkContext>
    </S:Header>
  <S:Body>
    <ns0:obterPacotePosicaoMotoristaHistoricoResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <anguloReferencia>0</anguloReferencia>
        <bloqueio>0</bloqueio>
        <cidade>Curitiba</cidade>
        <codigoMacro>0</codigoMacro>
        <conteudoMensagem/>
        <dataPacote>2018-05-02T13:41:21.0</dataPacote>
        <dataPosicao>2018-05-02T13:41:38.0</dataPosicao>
        <direcao>102</direcao>
        <entrada1>0</entrada1>
        <entrada2>0</entrada2>
        <entrada3>0</entrada3>
        <entrada4>-241</entrada4>
        <entrada5>-247</entrada5>
        <entrada6>-248</entrada6>
        <entrada7>-231</entrada7>
        <entrada8>0</entrada8>
        <estadoLimpadorParabrisa>0</estadoLimpadorParabrisa>
        <eventoFormatado/>
      </return>
    </ns0:obterPacotePosicaoMotoristaHistoricoResponse>
  </S:Body>
</S:Envelope>
```

```

<eventoSeqFormatado/>
<eventosTelemetria>
  <eventoTelemetria>
    <idEvento>245</idEvento>
    <tempoDuracao>100</tempoDuracao>
  </eventoTelemetria>
  <eventoTelemetria>
    <idEvento>24</idEvento>
    <tempoDuracao>100</tempoDuracao>
    <velocidadeMaximaEvento>100</velocidadeMaximaEvento>
    <velocidadeReferencia>40</velocidadeReferencia>
  </eventoTelemetria>
</eventosTelemetria>
<gps>0</gps>
<horimetro>120566</horimetro>
<idMotorista>0</idMotorista>
<idPacote>1310500</idPacote>
<idReferencia>0</idReferencia>
<idVeiculo>406575</idVeiculo>
<ignicao>1</ignicao>
<integradoraId>80</integradoraId>
<jamming>0</jamming>
<latitude>-25.451593</latitude>
<longitude>-49.2465991</longitude>
<memoria>0</memoria>
<nomeMensagem/>
<nomeMotorista/>
<odometro>50000</odometro>
<pontoEntrada>0</pontoEntrada>
<pontoReferencia/>
<pontoSaida>0</pontoSaida>
<rpm>400</rpm>
<rua>Rua Joao Marchesini</rua>
<saida1>0</saida1>
<saida2>-240</saida2>
<saida3>0</saida3>
<saida4>0</saida4>
<saida5>0</saida5>
<saida6>0</saida6>
<saida7>0</saida7>
<saida8>0</saida8>
<satelite>0</satelite>
<temperatura1>23</temperatura1>
<temperatura2>-125</temperatura2>
<temperatura3>2</temperatura3>
<tensao>24</tensao>
<textoMensagem/>
<tipoTeclado>0</tipoTeclado>
<uf>PR</uf>
<velocidade>0</velocidade>
</return>
</ns0:obterPacotePosicaoMotoristaHistoricoResponse>
</S:Body>
</S:Envelope>

```

5.7. ObterPacotePosicaoMotoristaPorRangeJSON

Descripción

Método que permite obtener uno o más paquetes de posiciones específicos, a través del identificador único de cada paquete. Este método retorna la misma información que el método ObterPacotePosicoesMotorista.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicaoMotoristaPorRangeJSON(String usuario, String senha, Integer idInicio, Integer idFinal, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idInicio	Integer	Sí	Identificador del paquete de posiciones a partir del cual se desea que el web service reenvíe.
idFinal	Integer	Sí	Identificador del paquete de posiciones hasta el cual se desea que el web service reenvíe.
quantidade	Integer	Sí	Límite de registros como resultado de la consulta. El sistema posee un límite estándar de 3000 registros, que se aplicará en los siguientes casos: Cuando el cliente no informe este parámetro; Cuando el cliente informe un número mayor que el límite estándar.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.
dataPosicao	String	Fecha de grabación de la posición en el servidor.
dataPacote	String	Fecha del GPS (GMT) en el momento del envío de la posición.

Nombre	Tipo	Descripción
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.
direcao	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MS (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
velocidade	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
ignicao	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
horimetro	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
odometro	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
tensao	Integer	Valor de tensión de la batería.
saida1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.

MTC (BIT)	LMU/MS (Grados)	Dirección
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satellite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.

MTC (BIT)	LMU/MSC (Grados)	Dirección
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
idReferencia	Integer	Identificação do ponto de referência. Esse id está relacionado com a tabela 'ponto_referencia' ou com o resultado do método ObterPontosReferencia(), cujo retorno é a lista de pontos de referência criados no SASGC e marcados como EMBARCÁVEL.
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pais	String	País.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.

MTC (BIT)	LMU/MSC (Grados)	Dirección
textoMensagem	String	Contenido del mensaje tipo texto libre.
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamiento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
idMotorista	Integer	Identificación del conductor.
nomeMotorista	String	Nombre del conductor.
nivelCombustivel	Integer	Porcentaje de llenado del tanque de combustible en el momento de la captura de la posición.
litrometro	Double	Cantidad de combustible en litros consumida durante la vida útil del vehículo (combustible total). Retorna "-1.0" si la información del litrómetro no está disponible en el vehículo.
limpadorParabrisa	Integer	Estado del limpiaparabrisas: • 0 — Limpiaparabrisas No Accionado • 1 — Limpiaparabrisas Accionado

MTC (BIT)	LMU/MSC (Grados)	Dirección
eventosTelemetria	Array[int]	Lista de evento(s) de telemetría compuesta por los campos: • idEvento — Identificador del Evento • tempoDuracao — Duración del Evento en segundos • velocidadeMaximaEvento — Velocidad Máxima alcanzada en el Evento • velocidadeReferencia — Velocidad de Referencia

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicaoMotoristaPorRangeJSON>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <idInicio>idPacoteInicio</idInicio>
      <idFinal>idPacoteFinal</idFinal>
      <quantidade>ValorNumérico</quantidade>
    </web:obterPacotePosicaoMotoristaPorRangeJSON>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <work:WorkContext xmlns:work="http://oracle.com/weblogic/soap/workarea/">
      XYZ </work:WorkContext>
    </S:Header>
  <S:Body>
    <ns0:obterPacotePosicoesJSONResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        {
          "idVeiculo": 1186796,
          "dataPosicao": "2023-10-09 10:08:43.0",
          "dataPacote": "2023-10-09 10:08:42.0",
          "latitude": -20.997711,
          "longitude": -51.800974,
          "direcao": 176,
          "velocidade": 59,
          "ignicao": 1,
          "odometro": 464030,
          "horimetro": 1602858,
          "tensao": 25,
          "saida1": 0,
          "saida2": 0,
          "saida3": 0,
          "saida4": 0,
          "entrada1": 0,
```

```

    "entrada2": 0,
    "entrada3": 0,
    "entrada4": 0,
    "satelite": 0,
    "memoria": 0,
    "idReferencia": 0,
    "bloqueio": 0,
    "gps": 1,
    "uf": "MS",
    "cidade": "Tres Lagoas",
    "rua": "Rod P/ Brasilandia",
    "pais": "BR",
    "pontoReferencia": "PE - PRF. Tres lagoas/MS",
    "anguloReferencia": 176,
    "distanciaReferencia": 20052,
    "rpm": 0,
    "temperatura1": -125,
    "temperatura2": -125,
    "temperatura3": -125,
    "saida5": 0,
    "saida6": 0,
    "saida7": 0,
    "saida8": 0,
    "entrada5": 0,
    "entrada6": 0,
    "entrada7": 0,
    "entrada8": 0,
    "pontoEntrada": 0,
    "pontoSaida": 0,
    "codigoMacro": 0,
    "nomeMensagem": "",
    "conteudoMensagem": "",
    "textoMensagem": "",
    "tipoTeclado": 0,
    "eventoSequenciamento": [],
    "eventos": [],
    "jamming": 0,
    "statusAncora": 0,
    "idPacote": 81181262,
    "integradoraId": 80,
    "idMotorista": 0,
    "nomeMotorista": "",
    "nivelCombustivel": "0",
    "litrometro": "0.0",
    "estadoLimpadorParabrisa": 0,
    "eventosTelemetria": []
  }
</return>
</ns0:obterPacotePosicoesJSONResponse>
</S:Body>
</S:Envelope>

```

5.8. ObterPacotePosicoesMotoristaJSON

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos. Además de traer la información que el método ObterPacotePosicoesJSON traía, este método trae también la información del Conductor (ID y Nombre) y el Estado del Limpiaparabrisas, siendo su contenido compuesto por los siguientes bloques de información:

- Pacote de posição;
- Pacote de mensagens (todos os tipos);
- Pacote de eventos;
- Pacote Alarme Sequenciamento.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicoesMotoristaJSON(String usuario, String senha, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	Sí	Límite de registros como resultado de la consulta. El sistema posee un límite estándar de 3000 registros, que se aplicará en los siguientes casos: Cuando el cliente no informe este parámetro; Cuando el cliente informe un número mayor que el límite estándar.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.

Nombre	Tipo	Descripción
<code>dataPosicao</code>	String	Fecha de grabación de la posición en el servidor.
<code>dataPacote</code>	String	Fecha del GPS (GMT) en el momento del envío de la posición.
<code>latitude</code>	Double	Latitud en formato decimal.
<code>longitude</code>	Double	Longitud en formato decimal.
<code>direcao</code>	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MS (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
<code>velocidade</code>	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
<code>ignicao</code>	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
<code>horimetro</code>	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
<code>odometro</code>	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
<code>tensao</code>	Integer	Valor de tensión de la batería.
<code>saida1</code>	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.

MTC (BIT)	LMU/MS (Grados)	Dirección
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satellite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.

MTC (BIT)	LMU/MSC (Grados)	Dirección
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pais	String	País.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.

MTC (BIT)	LMU/MSC (Grados)	Dirección
textoMensagem	String	Contenido del mensaje tipo texto libre.
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamiento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
idMotorista	Integer	Identificación del conductor.
nomeMotorista	String	Nombre del conductor conectado.
nivelCombustivel	Integer	Porcentaje de llenado del tanque de combustible en el momento de la captura de la posición.
litrometro	Double	Cantidad de combustible en litros consumida durante la vida útil del vehículo (combustible total). Retorna "-1.0" si la información del litrómetro no está disponible en el vehículo.
limpadorParabrisa	Integer	Estado del limpiaparabrisas: • 0 — Limpiaparabrisas No Accionado • 1 — Limpiaparabrisas Accionado

MTC (BIT)	LMU/MSC (Grados)	Dirección
eventosTelemetria	Array[int]	Lista de evento(s) de telemetría compuesta por los campos: • idEvento — Identificador del Evento • tempoDuracao — Duración del Evento en segundos • velocidadeMaximaEvento — Velocidad Máxima alcanzada en el Evento • velocidadeReferencia — Velocidad de Referencia
acessorios	Json	Este campo retorna la lista de accesorios vinculados al componente “Rede de Acessorios”. La red de accesorios está preparada para recibir una cantidad indefinida de accesorios que se agregan al rastreador sin necesidad de ocupar los puertos físicos. Cada uno de los sensores envía los siguientes campos: Número de Serie * Tipo Índice Valor Local *El campo serial vendrá vacío y su implementación ocurrirá en un desarrollo futuro. El campo Tipo puede contener los siguientes valores: ENTRADA_DIGITAL 1 SAIDA_DIGITAL 2 TEMPERATURA 3 UMIDADE 4 TENSÃO 5 ANALOGICO 6 INDEFINIDO 65.535

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicoesMotoristaJSON>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <quantidade>ValorNumérico</quantidade>
    </web:obterPacotePosicoesMotoristaJSON>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterPacotePosicoesMotoristaJSONResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        {
          "idVeiculo": 267254,
          "dataPosicao": "2023-10-10 08:54:51.0",
          "dataPacote": "2023-10-10 07:30:44.0",
          "latitude": -23.482824,
          "longitude": -46.8879424,
          "direcao": 0,
          "velocidade": 0,
          "ignicao": 0,
          "odometro": 0,
          "horimetro": 121299,
          "tensao": 11,
          "saida1": 0,
          "saida2": -240,
          "saida3": -254,
          "saida4": -253,
          "entrada1": 0,
          "entrada2": 0,
          "entrada3": -231,
          "entrada4": -247,
          "satelite": 0,
          "memoria": 1,
          "idReferencia": 2296820,
          "bloqueio": 0,
          "gps": 0,
          "uf": "SP",
          "cidade": "Barueri",
          "rua": "Sem nome",
          "pais": "BR",
          "pontoReferencia": "Ponto Barueri - Jd Graziela 3",
          "anguloReferencia": 0,
          "distanciaReferencia": 0,
          "rpm": 0,
          "temperatura1": -125,
          "temperatura2": -125,
          "temperatura3": -125,
          "saida5": 245,
          "saida6": 0,
          "saida7": 0,
          "saida8": -232,
          "entrada5": -248,
          "entrada6": -241,
          "entrada7": -250,
          "entrada8": -251,
          "pontoEntrada": 0,
          "pontoSaida": 0,
          "codigoMacro": 0,
          "nomeMensagem": "",
          "conteudoMensagem": "",
          "textoMensagem": "",
          "tipoTeclado": 0,
          "eventoSequenciamento": [],
          "eventos": [],
          "jamming": 0,
          "statusAncora": 0,
          "idPacote": 81204377,
          "integradoraId": 80,
          "idMotorista": 0,
          "nomeMotorista": "",

```

```
        "nivelCombustivel": "0",  
        "litrometro": "0.0",  
        "estadoLimpadorParabrisa": 0,  
        "eventosTelemetria": []  
    }  
    </return>  
</ns0:obterPacotePosicoesMotoristaJSONResponse>  
</S:Body>  
</S:Envelope>
```

5.9. ObterPacotePosicoesRestricao

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos en el momento en que el cliente tenía restricción, siendo su contenido compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma Secuenciamiento. Método exclusivo para paquetes con restricción.

Declaración del Método

```
List obterPacotePosicoesRestricao(String usuario, String senha, Integer quantidade, Integer idVeiculo)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	No	Límite de registros como resultado de la consulta. El sistema tiene un límite estándar de 300 registros, que se aplicará en los siguientes casos: cuando el cliente no informe este parámetro; cuando el cliente informe un número mayor que el límite estándar.
idVeiculo	Integer	Sí	Código del vehículo, que puede obtenerse mediante el método obterVeiculos.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.
dataPosicao	Calendar	Fecha de grabación de la posición en el servidor.
dataPacote	Calendar	Fecha del GPS (GMT) en el momento del envío de la posición.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.
direcao	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo direcao:

MTC (BIT)	LMU/MSC (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
velocidade	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
ignicao	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
horimetro	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
odometro	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
tensao	Integer	Valor de tensión de la batería.
saida1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satelite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).

MTC (BIT)	LMU/MS (Grados)	Dirección
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
textoMensagem	String	Contenido del mensaje tipo texto libre.
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT

MTC (BIT)	LMU/MS (Grados)	Dirección
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicoesRestricao>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <quantidade>ValorNumérico</quantidade>
      <idVeiculo>idVeiculo</idVeiculo>
    </web:obterPacotePosicoesRestricao>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:obterPacotePosicoesRestricaoResponse xmlns:ns2="http://ws.integra.sascar.com.br/">
      <return>
        <bloqueio>0</bloqueio>
        <cidade>São José dos Pinhais</cidade>
        <conteudoMensagem/>
        <dataPacote>2012-07-27T07:38:04-03:00</dataPacote>
        <dataPosicao>2012-07-27T10:53:17-03:00</dataPosicao>
        <direcao>4</direcao>
        <entrada1>0</entrada1>
        <entrada2>0</entrada2>
        <entrada3>0</entrada3>
        <entrada4>-241</entrada4>
        <entrada5>-247</entrada5>
        <entrada6>-248</entrada6>
        <entrada7>-231</entrada7>
        <entrada8>0</entrada8>
        <eventos>
          <codigo>501</codigo>
        </eventos>
        <eventos>
          <codigo>599</codigo>
        </eventos>
        <eventos>
          <codigo>653</codigo>
        </eventos>
        <eventos>
          <codigo>1002</codigo>
        </eventos>
        <eventos>
          <codigo>652</codigo>
        </eventos>
        <gps>1</gps>
        <idReferencia>0</idReferencia>
        <horimetro>94297</horimetro>
        <idPacote>6199608</idPacote>
        <idVeiculo>325900</idVeiculo>
        <ignicao>1</ignicao>
        <latitude>-25.540326666666665</latitude>
        <longitude>-49.200445</longitude>
        <memoria>1</memoria>
        <nomeMensagem/>
        <odometro>2028335</odometro>
        <pontoEntrada>0</pontoEntrada>
        <pontoReferencia/>
        <pontoSaida>0</pontoSaida>
        <rpm>1119</rpm>
        <rua>R Voluntários da Pátria</rua>
        <saida1>0</saida1>
        <saida2>-240</saida2>
        <saida3>0</saida3>
        <saida4>0</saida4>
      </return>
    </ns2:obterPacotePosicoesRestricaoResponse>
  </S:Body>
</S:Envelope>
```

```
0 0 0 0 -125 -125 -125 0 PR 24 <jamming>0</jamming> </ns2:obterPacotePosicoesRestricaoResponse>
```

5.10. ObterPacotePosicaoPorRange

Descripción

Método que permite obtener uno o más paquetes de posiciones específicos, a través del identificador único de cada paquete. Este método retorna la misma información que el método ObterPacotePosicoes.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicaoPorRange(String usuario, String senha, Integer idInicio, Integer idFinal, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idInicio	Integer	Sí	Identificador del paquete de posiciones a partir del cual se desea que el web service reenvíe.
idFinal	Integer	Sí	Identificador del paquete de posiciones hasta el cual se desea que el web service reenvíe.
quantidade	Integer	Sí	Límite de registros como resultado de la consulta. El sistema posee un límite estándar de 3000 registros, que se aplicará en los siguientes casos: Cuando el cliente no informe este parámetro; Cuando el cliente informe un número mayor que el límite estándar.



Si se desea recibir solo un paquete de posiciones, se debe informar el mismo idInicio e idFinal.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.

Nombre	Tipo	Descripción
<code>dataPosicao</code>	String	Fecha de grabación de la posición en el servidor.
<code>dataPacote</code>	String	Fecha del GPS (GMT) en el momento del envío de la posición.
<code>latitude</code>	Double	Latitud en formato decimal.
<code>longitude</code>	Double	Longitud en formato decimal.
<code>direcao</code>	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MS (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
<code>velocidade</code>	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
<code>ignicao</code>	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
<code>horimetro</code>	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
<code>odometro</code>	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
<code>tensao</code>	Integer	Valor de tensión de la batería.
<code>saida1</code>	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.

MTC (BIT)	LMU/MS (Grados)	Dirección
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satelite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.

MTC (BIT)	LMU/MSC (Grados)	Dirección
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>

MTC (BIT)	LMU/MSC (Grados)	Dirección
textoMensagem	String	Contenido del mensaje tipo texto libre.
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamiento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicaoPorRange>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <idInicio>ValorIDInicial</idInicio>
      <!--Optional:-->
      <idFinal>ValorIDFinal</idFinal>
      <quantidade>ValorNumérico</quantidade>
    </web:obterPacotePosicaoPorRange>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">
      r00ABXdRABt3ZWJsb2dpYy5hcHAubW9kdWxvLXd1Yi1lYXIAAADWAAAAI3d1YmxvZ2ljLndvcmtbcmVhLlN0cm1uZ1dvcmtDb
      250ZXh0AAV2XzQwMQAA</WorkContext>
    </S:Header>
  <S:Body>
    <ns0:obterPacotePosicaoPorRangeResponse
      xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <anguloReferencia>168</anguloReferencia>
        <bloqueio>0</bloqueio>
        <cidade>Cajuru</cidade>
        <codigoMacro>0</codigoMacro>
        <conteudoMensagem/>
        <dataPacote>2023-10-10T09:05:24.0</dataPacote>
        <dataPosicao>2023-10-10T09:05:39.0</dataPosicao>
        <direcao>114</direcao>
        <distanciaReferencia>13563</distanciaReferencia>
        <entrada1>0</entrada1>
        <entrada2>0</entrada2>
        <entrada3>0</entrada3>
        <entrada4>-241</entrada4>
        <entrada5>-247</entrada5>
        <entrada6>-248</entrada6>
        <entrada7>-231</entrada7>
        <entrada8>0</entrada8>
        <eventoFormatado/>
        <eventoSeqFormatado/>
        <gps>1</gps>
        <horimetro>92352</horimetro>
        <idPacote>81204600</idPacote>
        <idReferencia>0</idReferencia>
        <idVeiculo>1731832</idVeiculo>
        <ignicao>1</ignicao>
        <integradoraId>80</integradoraId>
        <jamming>0</jamming>
        <latitude>-21.154035</latitude>
        <longitude>-47.3294983</longitude>
        <memoria>1</memoria>
        <nomeMensagem/>
        <odometro>281582</odometro>
        <pontoEntrada>0</pontoEntrada>
        <pontoReferencia>PE - Cajuru-SP</pontoReferencia>
        <pontoSaida>0</pontoSaida>
        <rpm>601</rpm>
        <rua>Unnamed Road</rua>
        <saida1>0</saida1>
        <saida2>-240</saida2>
        <saida3>0</saida3>
        <saida4>0</saida4>
        <saida5>0</saida5>
        <saida6>-232</saida6>
        <saida7>0</saida7>
        <saida8>0</saida8>
        <satelite>0</satelite>
        <temperatura1>-125</temperatura1>
        <temperatura2>-125</temperatura2>
        <temperatura3>-125</temperatura3>
        <temperaturaSerial>-1250</temperaturaSerial>
        <tensao>28</tensao>
        <textoMensagem/>
        <tipoTeclado>0</tipoTeclado>
        <uf>SP</uf>
        <umidadeSerial>-1250</umidadeSerial>
      </return>
    </ns0:obterPacotePosicaoPorRangeResponse>
  </S:Body>
</S:Envelope>
```

```
<velocidade>0</velocidade>
</return>
</ns0:obterPacotePosicaoPorRangeResponse>
</S:Body>
</S:Envelope>
```

5.11. ObterPacoteLocalizacao

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos.

Declaración del Método

```
List obterPacoteLocalizacao(String usuario, String senha, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	Sí	Límite de registros como resultado de la consulta. El sistema posee un límite estándar de 2000 registros, que se aplicará en los siguientes casos: Cuando el cliente no informe este parámetro; Cuando el cliente informe un número mayor que el límite estándar.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
dataPacote	Calendar	Fecha del GPS (GMT) en el momento del envío de la posición.
direcao	Integer	Dirección del vehículo. Ver tabla a continuación.
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
idVeiculo	Integer	Código del vehículo.
ignicao	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.

Nome	Tipo	Descrição
velocidade	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MSC (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/" >
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacoteLocalizacao>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>Quantidade</quantidade>
    </web:obterPacoteLocalizacao>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns0:obterPacoteLocalizacaoResponse xmlns:ns0="http://webservice.web.integracao.sascar/" >
      <return>
        <dataPacote>2015-04-06T12:00:42.0</dataPacote>
        <direcao>4</direcao>
        <gps>0</gps>
        <idVeiculo>021456</idVeiculo>
        <ignicao>1</ignicao>
        <latitude>-23.62143</latitude>
        <longitude>-46.829361666666664</longitude>
        <velocidade>60</velocidade>
      </return>
    </ns0:obterPacoteLocalizacaoResponse>
  </S:Body>
</S:Envelope>
```

```
</return>  
</ns0:obterPacoteLocalizacaoResponse>  
</S:Body>  
</S:Envelope>
```

5.12. ObterPacotePosicaoHistorico

Descripción

Método para consultar el histórico de paquetes de posiciones enviados por los vehículos, siendo su contenido compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma Secuenciamiento.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicaoHistorico(String usuario, String senha, String dataInicio, String dataFinal, Integer idVeiculo)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
dataInicio	String	Sí	Inicio del rango de fecha para la búsqueda.. Formato: (YYYY-MM-DD HH:MM:SS).
dataFinal	String	Sí	Fin del rango de fecha para la búsqueda. Formato: (YYYY-MM-DD HH:MM:SS).
idVeiculo	Integer	No	Código del vehículo.

Retorno

El sistema posee un límite estándar de 3000 registros por solicitud. Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.
dataPosicao	String	Fecha de grabación de la posición en el servidor.
dataPacote	String	Fecha del GPS (GMT) en el momento del envío de la posición.
latitude	Double	Latitud en formato decimal.

Nombre	Tipo	Descripción
longitude	Double	Longitud en formato decimal.
direcao	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MS (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
velocidade	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
ignicao	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
horimetro	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satellite=0", informando que el mensaje fue enviado vía GPRS.
odometro	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
tensao	Integer	Valor de tensión de la batería.
saida1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.

MTC (BIT)	LMU/MS (Grados)	Dirección
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satelite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.

MTC (BIT)	LMU/MSC (Grados)	Dirección
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: <i>_João da Silva_</i> .

MTC (BIT)	LMU/MSC (Grados)	Dirección
textoMensagem	String	Contenido del mensaje tipo texto libre.
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamiento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicaoHistorico>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <dataInicio>2023-10-16 00:00:00</dataInicio>
      <dataFinal>2023-10-16 23:59:00</dataFinal>
      <!--Optional:-->
      <idVeiculo>900757</idVeiculo>
    </web:obterPacotePosicaoHistorico>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterPacotePosicaoHistoricoResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <anguloReferencia>0</anguloReferencia>
        <bloqueio>0</bloqueio>
        <cidade>Curitiba</cidade>
        <codigoMacro>0</codigoMacro>
        <conteudoMensagem/>
        <dataPacote>2023-10-16T00:00:03.0</dataPacote>
        <dataPosicao>2023-10-16T00:00:09.0</dataPosicao>
        <direcao>0</direcao>
        <distanciaReferencia>0</distanciaReferencia>
        <entrada1>0</entrada1>
        <entrada2>0</entrada2>
        <entrada3>-231</entrada3>
        <entrada4>-247</entrada4>
        <entrada5>-248</entrada5>
        <entrada6>-241</entrada6>
        <entrada7>-250</entrada7>
        <entrada8>-251</entrada8>
        <eventoFormatado/>
        <eventoSeqFormatado/>
        <gps>1</gps>
        <horimetro>261533</horimetro>
        <idPacote>81327633</idPacote>
        <idReferencia>2387650</idReferencia>
        <idVeiculo>900757</idVeiculo>
        <ignicao>1</ignicao>
        <integradoraId>80</integradoraId>
        <jamming>0</jamming>
        <latitude>-25.451672</latitude>
        <longitude>-49.2460896</longitude>
        <memoria>0</memoria>
        <nomeMensagem/>
        <odometro>559</odometro>
        <pontoEntrada>0</pontoEntrada>
        <pontoReferencia>JUNIOR</pontoReferencia>
        <pontoSaida>0</pontoSaida>
        <rpm>0</rpm>
        <rua>R Joao Marchesini</rua>
        <saida1>0</saida1>
        <saida2>-240</saida2>
        <saida3>254</saida3>
        <saida4>253</saida4>
        <saida5>245</saida5>
        <saida6>0</saida6>
        <saida7>-232</saida7>
        <saida8>0</saida8>
        <satelite>0</satelite>
        <temperatura1>-125</temperatura1>
        <temperatura2>-125</temperatura2>
        <temperatura3>-125</temperatura3>
        <tensao>11</tensao>
        <textoMensagem/>
        <tipoTeclado>0</tipoTeclado>
        <uf>PR</uf>
        <velocidade>0</velocidade>
      </return>
    </ns0:obterPacotePosicaoHistoricoResponse>
  </S:Body>
</S:Envelope>
```

5.13. ObterPacotePosicoesJSON

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos, cuyo contenido está compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma de Secuenciamiento.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicoesJSON(String usuario, String senha, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	No	Límite de registros como resultado de la consulta. El sistema tiene un límite estándar de 3.000 registros, que se aplicará en los siguientes casos: cuando el cliente no informe este parámetro; cuando el cliente informe un número mayor que el límite estándar.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.
dataPosicao	String	Fecha de grabación de la posición en el servidor.
dataPacote	String	Fecha del GPS (GMT) en el momento del envío de la posición.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.
direcao	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MSC (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
velocidade	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
ignicao	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
horimetro	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
odometro	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
tensao	Integer	Valor de tensión de la batería.
saida1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satelite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).

MTC (BIT)	LMU/MSC (Grados)	Dirección
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pais	String	País.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (. <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
textoMensagem	String	Contenido del mensaje tipo texto libre.
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT

MTC (BIT)	LMU/MSC (Grados)	Dirección
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
nivelCombustivel	Integer	Porcentaje de llenado del tanque de combustible en el momento de la captura de la posición.
litrometro	Double	Cantidad de combustible en litros consumida durante la vida útil del vehículo (combustible total). Retorna "-1.0" si la información del litrómetro no está disponible en el vehículo.
acessorios	Json	Este campo retorna la lista de accesorios vinculados al componente "Rede de Accesorios". La red de accesorios está preparada para recibir una cantidad indefinida de accesorios que se agregan al rastreador sin necesidad de ocupar los puertos físicos. Cada uno de los sensores envía los siguientes campos: Número de Serie * Tipo Índice Valor Local *El campo serial vendrá vacío y su implementación ocurrirá en un desarrollo futuro. El campo Tipo puede contener los siguientes valores: ENTRADA_DIGITAL 1 SAIDA_DIGITAL 2 TEMPERATURA 3 UMIDADE 4 TENSAO 5 ANALOGICO 6 INDEFINIDO 65.535

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicoesJSON>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>?</quantidade>
    </web:obterPacotePosicoesJSON>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterPacotePosicoesJSONResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        {
          "idVeiculo":1731832,
          "dataPosicao":"2023-10-15 00:00:24.0",
          "dataPacote":"2023-10-15 00:00:24.0",
          "latitude":-22.2847315,
          "longitude":-46.6652308,
          "direcao":212,
          "velocidade":0,
          "ignicao":1,
          "odometro":284463,
          "horimetro":97084,
          "tensao":28,
          "saida1":0,
          "saida2":-240,
          "saida3":0,
          "saida4":0,
          "entrada1":0,
          "entrada2":0,
          "entrada3":0,
          "entrada4":-241,
          "satelite":0,
          "memoria":0,
          "idReferencia":0,
          "bloqueio":0,
          "gps":1,
          "uf":"MG",
          "cidade":"Jacutinga",
          "rua":"Unnamed Road",
          "pais":"Brazil",
          "pontoReferencia":"PE - Jacutinga-MG",
          "anguloReferencia":270,
          "distanciaReferencia":5316,
          "rpm":1302,
          "temperatura1":-125,
          "temperatura2":-125,
          "temperatura3":-125,
          "saida5":0,
          "saida6":-232,
          "saida7":0,
          "saida8":0,
          "entrada5":-247,
```

```
"entrada6":-248,  
"entrada7":-231,  
"entrada8":0,  
"pontoEntrada":0,  
"pontoSaida":0,  
"codigoMacro":12,  
"nomeMensagem":"DIRECAO CARREGADO",  
"conteudoMensagem":"","  
"textoMensagem":"","  
"tipoTeclado":8,  
"eventoSequenciamento":[],  
"eventos":[{"codigo" : - 234}],  
"jamming":0,  
"statusAncora":0,  
"idPacote":81313330,  
"integradoraId":80,  
"idMotorista":2741963,  
"nomeMotorista":"GABRIEL APARECIDO ROMAO FILHO",  
"nivelCombustivel":"65",  
"litrometro":"171690.496",  
"estadoLimpadorParabrisa":null,  
"umidadeSerial":-1250,  
"temperaturaSerial":- 1250  
}  
</return>  
</ns0:obterPacotePosicoesJSONResponse>  
</S:Body>  
</S:Envelope>
```

5.14. ObterPacotePosicoesJSONComPlaca

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos, cuyo contenido está compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma de Secuenciamiento.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicoesJSONComPlaca(String usuario, String senha, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	No	Límite de registros como resultado de la consulta. El sistema tiene un límite estándar de 3.000 registros, que se aplicará en los siguientes casos: cuando el cliente no informe este parámetro; cuando el cliente informe un número mayor que el límite estándar.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.
dataPosicao	String	Fecha de grabación de la posición en el servidor.
dataPacote	String	Fecha del GPS (GMT) en el momento del envío de la posición.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.
direcao	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MSC (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
velocidade	Integer	Velocidad obtenida del GPS.
ignicao	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
horimetro	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
odometro	Integer	Odómetro actual del vehículo.
tensao	Integer	Valor de tensión de la batería.
saida1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satelite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).

MTC (BIT)	LMU/MSC (Grados)	Dirección
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pais	String	País.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
textoMensagem	String	Contenido del mensaje tipo texto libre.
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT

MTC (BIT)	LMU/MS (Grados)	Dirección
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
nivelCombustivel	Integer	Porcentaje de llenado del tanque de combustible en el momento de la captura de la posición.
litrometro	Double	Cantidad de combustible en litros consumida durante la vida útil del vehículo (combustible total). Retorna "-1.0" si la información del litrómetro no está disponible en el vehículo.
Placa	String	Patente del vehículo.
acessorios	Json	Este campo retorna la lista de accesorios vinculados al componente "Rede de Acessorios". La red de accesorios está preparada para recibir una cantidad indefinida de accesorios que se agregan al rastreador sin necesidad de ocupar los puertos físicos. Cada uno de los sensores envía los siguientes campos: Número de Serie * Tipo Índice Valor Local *El campo serial vendrá vacío y su implementación ocurrirá en un desarrollo futuro. El campo Tipo puede contener los siguientes valores: ENTRADA_DIGITAL 1 SAIDA_DIGITAL 2 TEMPERATURA 3 UMIDADE 4 TENSÃO 5 ANALOGICO 6 INDEFINIDO 65.535

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd

- teclado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicoesJSONComPlaca>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>?</quantidade>
    </web:obterPacotePosicoesJSONComPlaca>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterPacotePosicoesJSONComPlacaResponse
      xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        {
          "idVeiculo": 1575379,
          "dataPosicao": "2023-10-16 09:43:25.0",
          "dataPacote": "2023-10-16 09:43:25.0",
          "latitude": -22.6030336,
          "longitude": -47.365104,
          "direcao": 0,
          "velocidade": 0,
          "ignicao": 0,
          "odometro": 223892,
          "horimetro": 221115,
          "tensao": 25,
          "saida1": 0,
          "saida2": 0,
          "saida3": 0,
          "saida4": 0,
          "entrada1": -194,
          "entrada2": -194,
          "entrada3": -194,
          "entrada4": -194,
          "satelite": 0,
          "memoria": 0,
          "idReferencia": 0,
          "bloqueio": 0,
          "gps": 0,
          "uf": "SP",
          "cidade": "Limeira",
          "rua": "Sem nome",
          "pais": "BR",
          "pontoReferencia": "PE - Posto Topazio - Limeira-SP",
          "anguloReferencia": 141,
          "distanciaReferencia": 932,
```

```
        "rpm": 0,  
        "temperatura1": 36,  
        "temperatura2": 0,  
        "temperatura3": 0,  
        "saida5": 0,  
        "saida6": 0,  
        "saida7": 0,  
        "saida8": 0,  
        "entrada5": -194,  
        "entrada6": -194,  
        "entrada7": -194,  
        "entrada8": -194,  
        "pontoEntrada": 0,  
        "pontoSaida": 0,  
        "codigoMacro": 0,  
        "nomeMensagem": "",  
        "conteudoMensagem": "",  
        "textoMensagem": "",  
        "tipoTeclado": 0,  
        "eventoSequenciamento": [],  
        "eventos": [],  
        "jamming": 0,  
        "statusAncora": 0,  
        "idPacote": 81334476,  
        "integradoraId": 80,  
        "idMotorista": 0,  
        "nomeMotorista": "",  
        "nivelCombustivel": "0",  
        "litometro": "0.0",  
        "estadoLimpadorParabrisa": null,  
        "umidadeSerial": 0,  
        "temperaturaSerial": 0,  
        "placa": "MSC0032"  
    }  
    </return>  
</ns0:obterPacotePosicoesJSONComPlacaResponse>  
</S:Body>  
</S:Envelope>
```

5.15. ObterPacotePosicaoPorRangeJSON

Descripción

Método que permite obtener uno o más paquetes de posiciones específicos, a través del identificador único de cada paquete. Este método retorna la misma información que el método ObterPacotePosicoes.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List obterPacotePosicaoPorRangeJSON(String usuario, String senha, Integer idInicio, Integer idFinal, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idInicio	Integer	Sí	Identificador del paquete de posiciones a partir del cual se desea que el web service reenvíe.
idFinal	Integer	Sí	Identificador del paquete de posiciones hasta el cual se desea que el web service reenvíe.
quantidade	Integer	Sí	Límite de registros como resultado de la consulta. El sistema posee un límite estándar de 3000 registros, que se aplicará en los siguientes casos: Cuando el cliente no informe este parámetro; Cuando el cliente informe un número mayor que el límite estándar.



Si se desea recibir solo un paquete de posiciones, se debe informar el mismo idInicio e idFinal.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idPacote	Integer	Identificador único del paquete de posición.

Nombre	Tipo	Descripción
<code>dataPosicao</code>	String	Fecha de grabación de la posición en el servidor.
<code>dataPacote</code>	String	Fecha del GPS (GMT) en el momento del envío de la posición.
<code>latitude</code>	Double	Latitud en formato decimal.
<code>longitude</code>	Double	Longitud en formato decimal.
<code>direcao</code>	Integer	Dirección del vehículo. Ver tabla a continuación.

Valores para el campo **direcao**:

MTC (BIT)	LMU/MS (Grados)	Dirección
0	0 a 44	(N - Norte)
1	45 a 89	(NE - Nordeste)
2	90 a 134	(E - Este)
3	135 a 179	(SE - Sudeste)
4	180 a 224	(S - Sur)
5	225 a 269	(SO - Sudoeste)
6	270 a 314	(O - Oeste)
7	315 a 359	(NO - Noroeste)
<code>velocidade</code>	Integer	Velocidad obtenida del GPS; si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
<code>ignicao</code>	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
<code>horimetro</code>	Integer	Horómetro actual del vehículo. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
<code>odometro</code>	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
<code>tensao</code>	Integer	Valor de tensión de la batería.
<code>saida1</code>	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.

MTC (BIT)	LMU/MS (Grados)	Dirección
saida2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: • 240 (Código de sirena informando que está activada) • -240 (Código de sirena informando que está desactivada) • 0 (Ningún dispositivo).
saida3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satelite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIBRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satellite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.

MTC (BIT)	LMU/MS (Grados)	Dirección
memoria	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
idReferencia	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
bloqueio	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
uf	String	Provincia/Estado.
cidade	String	Ciudad.
rua	String	Dirección.
pais	String	País.
pontoReferencia	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
anguloReferencia	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
distanciaReferencia	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperatura1	Integer	Datos del sensor de temperatura 1.
temperatura2	Integer	Datos del sensor de temperatura 2.
temperatura3	Integer	Datos del sensor de temperatura 3.
saida5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

MTC (BIT)	LMU/MSC (Grados)	Dirección
saida6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
saida8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entrada8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado, o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
pontoEntrada	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
pontoSaida	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
codigoMacro	Integer	Código de la macro.
conteudoMensagem	String	Contenido del mensaje tipo macro, separado por guión bajo (). Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.

MTC (BIT)	LMU/MSC (Grados)	Dirección
textoMensagem	String	Contenido del mensaje tipo texto libre.
tipoTeclado	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
eventoSequenciamento	EventoSequenciamento[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: Id (Integer) (id del secuenciamiento relacionado con la tabla 'sequenciamento_evento' o con el resultado del método ObterSequenciamentoEvento()); tempo (Integer) (Duración de la alarma).
evento	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: codigo (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
nivelCombustivel	Integer	Porcentaje de llenado del tanque de combustible en el momento de la captura de la posición.
litometro	Double	Cantidad de combustible consumida durante la vida útil del vehículo.
acessorios	Json	Este campo retorna la lista de accesorios vinculados al componente "Rede de Acessorios". La red de accesorios está preparada para recibir una cantidad indefinida de accesorios que se agregan al rastreador sin necesidad de ocupar los puertos físicos. Cada uno de los sensores envía los siguientes campos: Número de Serie * Tipo Índice Valor Local *El campo serial vendrá vacío y su implementación ocurrirá en un desarrollo futuro. El campo Tipo puede contener los siguientes valores: ENTRADA_DIGITAL 1 SAIDA_DIGITAL 2 TEMPERATURA 3 UMIDADE 4 TENSÃO 5 ANALOGICO 6 INDEFINIDO 65.535

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicaoPorRangeJSON>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <idInicio>IDPACOTEINICIAL</idInicio>
      <idFinal>IDPACOTEFINAL</idFinal>
      <quantidade>QUANTIDADE</quantidade>
    </web:obterPacotePosicaoPorRangeJSON>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterPacotePosicaoPorRangeJSONResponse
      xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        {
          "idVeiculo": 900757,
          "dataPosicao": "2023-10-16 00:00:03.0",
          "dataPacote": "2023-10-16 00:00:09.0",
          "latitude": -25.451672,
          "longitude": -49.2460896,
          "direcao": 0,
          "velocidade": 0,
          "ignicao": 1,
          "odometro": 559,
          "horimetro": 261533,
          "tensao": 11,
          "saida1": 0,
          "saida2": -240,
          "saida3": 254,
          "saida4": 253,
          "entrada1": 0,
          "entrada2": 0,
          "entrada3": -231,
          "entrada4": -247,
          "satelite": 0,
          "memoria": 0,
          "idReferencia": 2387650,
          "bloqueio": 0,
          "gps": 1,
          "uf": "PR",
          "cidade": "Curitiba",
          "rua": "R Joao Marchesini",
          "pais": "BR",
          "pontoReferencia": "JUNIOR",
          "anguloReferencia": 0,
          "distanciaReferencia": 0,
          "rpm": 0,
          "temperatura1": -125,
          "temperatura2": -125,
          "temperatura3": -125,
```

```
"saida5": 245,  
"saida6": 0,  
"saida7": -232,  
"saida8": 0,  
"entrada5": -248,  
"entrada6": -241,  
"entrada7": -250,  
"entrada8": -251,  
"pontoEntrada": 0,  
"pontoSaida": 0,  
"codigoMacro": 0,  
"nomeMensagem": "",  
"conteudoMensagem": "",  
"textoMensagem": "",  
"tipoTeclado": 0,  
"eventoSequenciamento": [],  
"eventos": [],  
"jamming": 0,  
"statusAncora": 0,  
"idPacote": 81327633,  
"integradoraId": 80,  
"idMotorista": null,  
"nomeMotorista": "null",  
"nivelCombustivel": "0",  
"litrometro": "0.0",  
"estadoLimpadorParabrisa": null,  
"umidadeSerial": 0,  
"temperaturaSerial": 0  
}  
</return>  
</ns0:obterPacotePosicaoPorRangeJSONResponse>  
</S:Body>  
</S:Envelope>
```

5.16. ObterPacotePosicoesRFNacional

Descripción

Método para consultar la lista de paquetes de posiciones de rastreadores de carga.

Declaración del Método

```
List obterPacotePosicoesRFNacional(String usuario, String senha, Integer quantidade)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	No	Límite de registros como resultado de la consulta. El sistema tiene un límite estándar de 3.000 registros, que se aplicará en los siguientes casos: cuando el cliente no informe este parámetro; cuando el cliente informe un número mayor que el límite estándar.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
ccid	String	ID de la tarjeta SIM.
cidade	String	Ciudad.
dataPacote	Calendar	Fecha del GPS (GMT) en el momento del envío de la posición.
dataPosicao	Calendar	Fecha de grabación de la posición en el servidor.
idPacote	Integer	Identificador único del paquete de posición.
idVeiculo	Integer	Código del vehículo.
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: 0 — No hubo jamming 1 — Ocurrió jamming
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.

Nombre	Tipo	Descripción
rf	Integer	Informa si el RF del equipo está encendido: 0 — No está encendido 1 — Está encendido
rua	String	Dirección.
uf	String	Provincia/Estado.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterPacotePosicoesRFNacional>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade/>
    </web:obterPacotePosicoesRFNacional>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/"><S:Header><work:WorkContext
xmlns:work="http://oracle.com/weblogic/soap/workarear"
>
XYZ</work:WorkContext></S:Header><S:Body><ns0:obterPacotePosicoesRFNacionalResponse
xmlns:nse="http://webservice.web.integracao.sascar.com.br"
><return><ccid>89551020226000238751</ccid><cidade>Sao Bernardo do Campo</cidade><detaPacote>2017-03-
17 T 16:60:25</dataPacote><dataPosicao>2017-03-17T16:60:25</dataPosicao><idPacote>
9547614</idPacote><idVeiculo>622828</idVeiculo><jamming>0</jamming><latitude>-
23.6524171586162</latitude><longitude>-46.5734561983093</longitude><rf>1</rf><rua>Avenida Bispo Cesar
Dacorso Filho</rua><uf>
SP</uf></return></ns0:obterPacotePosicoesRFNacionalResponse></S:Body></S:Envelope></detaPacote>
```

5.17. getPositionsPacketJSON

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos, cuyo contenido está compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma de Secuenciamiento. La información retornada es la misma que la del método obterPacotePosicoesJSON, pero traducida al inglés y mostrando las fechas en UTC-0.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List getPositionsPacketJSON(String user, String password, Integer quantity)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
user	String	Sí	Login del integrador para acceso al sistema.
password	String	Sí	Contraseña del integrador para acceso al sistema.
quantity	Integer	No	Límite de registros como resultado de la consulta. El sistema posee un límite estándar de 3000 registros, que se aplicará en los siguientes casos: Cuando el cliente no informe este parámetro; Cuando el cliente informe un número mayor que el límite estándar.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
vehicleId	Integer	Código del vehículo.
positionDateUtc	Calendar	Fecha de grabación de la posición en el servidor, en UTC-0.
packetDateUtc	Calendar	Fecha del GPS (GMT) en el momento del envío de la posición, en UTC-0.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.

Nombre	Tipo	Descripción
direction	Integer	Dirección del vehículo: • 0 — N - Norte • 1 — NE - Nordeste • 2 — E - Este • 3 — SE - Sudeste • 4 — S - Sur • 5 — SO - Sudoeste • 6 — O - Oeste • 7 — NO - Noroeste
speed	Integer	Velocidad en km/h obtenida del GPS. Si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
ignition	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
odometer	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
horimeter	Integer	Horómetro actual del vehículo, en minutos. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
power	Integer	Valor de tensión de la batería, en Voltios.
output1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
output2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: 240 (Código de sirena informando que está activada); -240 (Código de sirena informando que está desactivada); 0 (Ningún dispositivo).
output3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

Nombre	Tipo	Descripción
input1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satellite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIVRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satellite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memory	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
referenceId	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
blocking	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido

Nombre	Tipo	Descripción
state	String	Provincia/Estado.
city	String	Ciudad.
street	String	Dirección.
country	String	País.
referencePoint	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
referenceAngle	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
referenceDistance	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperature1	Integer	Datos del sensor de temperatura 1, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
temperature2	Integer	Datos del sensor de temperatura 2, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
temperature3	Integer	Datos del sensor de temperatura 3, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
output5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

Nombre	Tipo	Descripción
input7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entryPoint	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
exitPoint	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
macroCode	Integer	Código de la macro.
messageName	String	Nombre del mensaje tipo macro.
messageContent	String	Contenido del mensaje tipo macro, separado por guión bajo (. <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
messageText	String	Contenido del mensaje tipo texto libre.
mdtType	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
sequencingEvent	EventoSecuenciamen to[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: eventId (Integer) (id del secuenciamiento relacionado con la tabla 'secuenciamiento_evento' o con el resultado del método ObterSequenciamientoEvento()); time (Integer) (Duración de la alarma).

Nombre	Tipo	Descripción
events	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: code (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
packetId	Integer	Identificador único del paquete de posición.
integratorId	Integer	Identificador único de la integradora.
driverId	Integer	Retorna siempre null.
driverName	String	Retorna siempre null.
fuelLevel	Integer	Porcentaje de llenado del tanque de combustible en el momento de la captura de la posición.
lithometer	Double	Cantidad de combustible en litros consumida durante la vida útil del vehículo (combustible total). Retorna "-1.0" si la información del litrómetro no está disponible en el vehículo.
windshieldWiperState	Integer	Retorna siempre null.
telemetryEvents	Array<int>	Lista de evento(s) de telemetría compuesta por los campos: • idEvent — Identificador del Evento • durationTime — Duración del Evento en segundos • maximumSpeedEvent — Velocidad Máxima alcanzada en el Evento • speedReference — Velocidad de Referencia
humiditySerial	Integer	Información del sensor de humedad en porcentaje multiplicado por 10. Ejemplo: 534 = 53,4%. No aparece si no está instalado.
temperatureSerial	Integer	Información del sensor de temperatura en grados Celsius multiplicado por 10. Ejemplo: 248 = 24,8°C. No aparece si no está instalado.

Nombre	Tipo	Descripción
acessorios	Json	Este campo retorna la lista de accesorios vinculados al componente "Rede de Acessorios". La red de accesorios está preparada para recibir una cantidad indefinida de accesorios que se agregan al rastreador sin necesidad de ocupar los puertos físicos. Cada uno de los sensores envía los siguientes campos: Número de Serie * Tipo Índice Valor Local *El campo serial vendrá vacío y su implementación ocurrirá en un desarrollo futuro. El campo Tipo puede contener los siguientes valores: ENTRADA_DIGITAL 1 SAIDA_DIGITAL 2 TEMPERATURA 3 UMIDADE 4 TENSAO 5 ANALOGICO 6 INDEFINIDO 65.535

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getPositionsPacketJSON>
      <user>Usuário</user>
      <password>Senha</password>
      <!--Optional:-->
      <quantity>?</quantity>
    </web:getPositionsPacketJSON>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:getPositionsPacketJSONResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        {
          "vehicleId": 886022,
```

```

    "positionDateUtc": "2023-10-16 13:14:50.0",
    "packetDateUtc": "2023-10-16 13:14:48.0",
    "latitude": -15.6679905,
    "longitude": -55.959408,
    "direction": 247,
    "speed": 0,
    "ignition": 0,
    "odometer": 106986,
    "horimeter": 142041,
    "power": 26,
    "output1": 0,
    "output2": 0,
    "output3": 0,
    "output4": 0,
    "input1": 0,
    "input2": 0,
    "input3": 0,
    "input4": 0,
    "satellite": 0,
    "memory": 0,
    "referenceId": 0,
    "blocking": 1,
    "gps": 1,
    "state": "MT",
    "city": "Cuiaba",
    "country": "BR",
    "street": "Sem nome",
    "referencePoint": "PE - BASE SHELL CUIABA",
    "referenceAngle": 0,
    "referenceDistance": 0,
    "rpm": 0,
    "temperature1": -125,
    "temperature2": -125,
    "temperature3": -125,
    "output5": 0,
    "output6": 0,
    "output7": 0,
    "output8": 0,
    "input5": 0,
    "input6": 0,
    "input7": 0,
    "input8": 0,
    "entryPoint": 0,
    "exitPoint": 0,
    "macroCode": 0,
    "messageName": "",
    "messageContent": "",
    "messageText": "",
    "mdtType": 0,
    "sequencingEvent": [],
    "events": [],
    "jamming": 0,
    "anchorStatus": 0,
    "packetId": 81335055,
    "integratorId": 80,
    "driverId": null,
    "driverName": "null",
    "fuelLevel": 0,
    "lithometer": 48686.762,
    "windshieldWiperState": null,
    "humiditySerial": -1250,
    "temperatureSerial": -1250
  }
</return>
</ns0:getPositionsPacketJSONResponse>
</S:Body>
</S:Envelope>

```

5.18. getDriverPositionPacketJSON

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos. Además de traer la información que el método ObterPacotePosicoesJSON traía, este método trae también la información del Conductor (ID y Nombre) y el Estado del Limpiaparabrisas, siendo su contenido compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma Secuenciamiento. La información retornada es la misma que la del método obterPacotePosicoesMotoristaJSON, pero traducida al inglés y mostrando las fechas en UTC-0.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List getDriverPositionPacketJSON(String user, String password, Integer quantity)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
user	String	Sí	Login del integrador para acceso al sistema.
password	String	Sí	Contraseña del integrador para acceso al sistema.
quantity	Integer	No	Límite de registros como resultado de la consulta. El sistema posee un límite estándar de 3000 registros, que se aplicará en los siguientes casos: Cuando el cliente no informe este parámetro; Cuando el cliente informe un número mayor que el límite estándar.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
vehicleId	Integer	Código del vehículo.
positionDateUtc	Calendar	Fecha de grabación de la posición en el servidor, en UTC-0.
packetDateUtc	Calendar	Fecha del GPS (GMT) en el momento del envío de la posición, en UTC-0.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.

Nombre	Tipo	Descripción
direction	Integer	Dirección del vehículo: • 0 — N - Norte • 1 — NE - Nordeste • 2 — E - Este • 3 — SE - Sudeste • 4 — S - Sur • 5 — SO - Sudoeste • 6 — O - Oeste • 7 — NO - Noroeste
speed	Integer	Velocidad en km/h obtenida del GPS. Si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
ignition	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
odometer	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
horimeter	Integer	Horómetro actual del vehículo, en minutos. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
power	Integer	Valor de tensión de la batería, en Voltios.
output1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
output2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: 240 (Código de sirena informando que está activada); -240 (Código de sirena informando que está desactivada); 0 (Ningún dispositivo).
output3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

Nombre	Tipo	Descripción
input1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satellite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIVRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satellite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memory	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
referenceId	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
blocking	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido

Nombre	Tipo	Descripción
state	String	Provincia/Estado.
city	String	Ciudad.
street	String	Dirección.
country	String	País.
referencePoint	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
referenceAngle	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
referenceDistance	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperature1	Integer	Datos del sensor de temperatura 1, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
temperature2	Integer	Datos del sensor de temperatura 2, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
temperature3	Integer	Datos del sensor de temperatura 3, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
output5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

Nombre	Tipo	Descripción
input7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entryPoint	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
exitPoint	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
macroCode	Integer	Código de la macro.
messageName	String	Nombre del mensaje tipo macro.
messageContent	String	Contenido del mensaje tipo macro, separado por guión bajo (. <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
messageText	String	Contenido del mensaje tipo texto libre.
mdtType	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
sequencingEvent	EventoSecuenciamen to[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: eventId (Integer) (id del secuenciamiento relacionado con la tabla 'secuenciamiento_evento' o con el resultado del método ObterSecuenciamientoEvento()); time (Integer) (Duración de la alarma).

Nombre	Tipo	Descripción
events	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: code (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
packetId	Integer	Identificador único del paquete de posición.
integratorId	Integer	Identificador único de la integradora.
driverId	Integer	Identificación del conductor.
driverName	String	Nombre del conductor conectado.
fuelLevel	Integer	Porcentaje de llenado del tanque de combustible en el momento de la captura de la posición.
lithometer	Double	Cantidad de combustible en litros consumida durante la vida útil del vehículo (combustible total). Retorna "-1.0" si la información del litrómetro no está disponible en el vehículo.
windshieldWiperState	Integer	Estado del limpiaparabrisas: • 0 — Limpiaparabrisas No Accionado • 1 — Limpiaparabrisas Accionado
telemetryEvents	Array<int>	Lista de evento(s) de telemetría compuesta por los campos: • idEvent — Identificador del Evento • durationTime — Duración del Evento en segundos • maximumSpeedEvent — Velocidad Máxima alcanzada en el Evento • speedReference — Velocidad de Referencia
humiditySerial	Integer	Información del sensor de humedad en porcentaje multiplicado por 10. Ejemplo: 534 = 53,4%. No aparece si no está instalado.
temperatureSerial	Integer	Información del sensor de temperatura en grados Celsius multiplicado por 10. Ejemplo: 248 = 24,8-C. No aparece si no está instalado.

Nombre	Tipo	Descripción
acessorios	Json	Este campo retorna la lista de accesorios vinculados al componente "Rede de Acessorios". La red de accesorios está preparada para recibir una cantidad indefinida de accesorios que se agregan al rastreador sin necesidad de ocupar los puertos físicos. Cada uno de los sensores envía los siguientes campos: Número de Serie * Tipo Índice Valor Local *El campo serial vendrá vacío y su implementación ocurrirá en un desarrollo futuro. El campo Tipo puede contener los siguientes valores: ENTRADA_DIGITAL 1 SAIDA_DIGITAL 2 TEMPERATURA 3 UMIDADE 4 TENSAO 5 ANALOGICO 6 INDEFINIDO 65.535

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getDriverPositionPacketJSON>
      <user>Usuário</user>
      <password>Senha</password>
      <!--Optional:-->
      <quantity>?</quantity>
    </web:getDriverPositionPacketJSON>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:getDriverPositionPacketJSONResponse
      xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        {
          "vehicleId": 1563022,
```

```

"positionDateUtc": "2023-10-16 14:06:57.0",
"packetDateUtc": "2023-10-16 14:06:56.0",
"latitude": -19.948707,
"longitude": -44.010242,
"direction": 67,
"speed": 11,
"ignition": 1,
"odometer": 323519,
"horimeter": 191239,
"power": 27,
"output1": 0,
"output2": 0,
"output3": 0,
"output4": 0,
"input1": 0,
"input2": 0,
"input3": 0,
"input4": 0,
"satellite": 0,
"memory": 0,
"referenceId": 0,
"blocking": 1,
"gps": 1,
"state": "MG",
"city": "Belo Horizonte",
"country": "BR",
"street": "Av Amazonas",
"referencePoint": "PE - Posto 040-MG (2)",
"referenceAngle": 12,
"referenceDistance": 1054,
"rpm": 956,
"temperature1": -125,
"temperature2": -125,
"temperature3": -125,
"output5": 0,
"output6": -232,
"output7": 0,
"output8": 0,
"input5": 0,
"input6": 0,
"input7": 0,
"input8": 0,
"entryPoint": 0,
"exitPoint": 0,
"macroCode": 0,
"messageName": "",
"messageContent": "",
"messageText": "",
"mdtType": 0,
"sequencingEvent": [],
"events": [],
"jamming": 0,
"anchorStatus": 0,
"packetId": 81336079,
"integratorId": 80,
"driverId": 0,
"driverName": "",
"fuelLevel": 99,
"lithometer": 182447.616,
"windshieldWiperState": 0,
"telemetryEvents": [
  {
    "eventId": 227,
    "durationTime": 11
  }
],
"humiditySerial": -1250,
"temperatureSerial": -1250

```

```
    }  
    </return>  
  </ns0:getDriverPositionPacketJSONResponse>  
</S:Body>  
</S:Envelope>
```

5.19. getPositionPacketByRangeJSON

Descripción

Método que permite obtener uno o más paquetes de posiciones específicos, a través del identificador único de cada paquete. Este método retorna la misma información que el método ObterPacotePosicoes, pero en inglés y con las fechas en UTC-0.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List getPositionPacketByRangeJSON(String user, String password, Integer startId, Integer endId, Integer quantity)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
user	String	Sí	Login del integrador para acceso al sistema.
password	String	Sí	Contraseña del integrador para acceso al sistema.
startId	Integer	Sí	Identificador del paquete de posiciones a partir del cual se desea que el web service reenvíe.
endId	Integer	Sí	Identificador del paquete de posiciones hasta el cual se desea que el web service reenvíe.
quantity	Integer	Sí	Límite de registros como resultado de la consulta. El sistema posee un límite estándar de 3000 registros, que se aplicará en los siguientes casos: Cuando el cliente no informe este parámetro; Cuando el cliente informe un número mayor que el límite estándar.



Si se desea recibir solo un paquete de posiciones, se debe informar el mismo startId e endId.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
vehicleId	Integer	Código del vehículo.
positionDateUtc	Calendar	Fecha de grabación de la posición en el servidor, en UTC-0.

Nombre	Tipo	Descripción
packetDateUtc	Calendar	Fecha del GPS (GMT) en el momento del envío de la posición, en UTC-0.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.
direction	Integer	Dirección del vehículo: • 0 — N - Norte • 1 — NE - Nordeste • 2 — E - Este • 3 — SE - Sudeste • 4 — S - Sur • 5 — SO - Sudoeste • 6 — O - Oeste • 7 — NO - Noroeste
speed	Integer	Velocidad en km/h obtenida del GPS. Si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
ignition	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
odometer	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
horimeter	Integer	Horómetro actual del vehículo, en minutos. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
power	Integer	Valor de tensión de la batería, en Voltios.
output1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
output2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: 240 (Código de sirena informando que está activada); -240 (Código de sirena informando que está desactivada); 0 (Ningún dispositivo).
output3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

Nombre	Tipo	Descripción
output4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satellite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIVRE IMPORTANTE: Si el paquete es del tipo “2” (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satellite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memory	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
referenceId	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
blocking	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado

Nombre	Tipo	Descripción
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
state	String	Provincia/Estado.
city	String	Ciudad.
street	String	Dirección.
country	String	País.
referencePoint	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
referenceAngle	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
referenceDistance	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperature1	Integer	Datos del sensor de temperatura 1, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
temperature2	Integer	Datos del sensor de temperatura 2, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
temperature3	Integer	Datos del sensor de temperatura 3, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
output5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

Nombre	Tipo	Descripción
input6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entryPoint	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
exitPoint	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
macroCode	Integer	Código de la macro.
messageName	String	Nombre del mensaje tipo macro.
messageContent	String	Contenido del mensaje tipo macro, separado por guión bajo (). <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
messageText	String	Contenido del mensaje tipo texto libre.
mdtType	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
sequencingEvent	EventoSequenciamen to[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: eventId (Integer) (id del secuenciamiento relacionado con la tabla 'sequenciamiento_evento' o con el resultado del método ObterSequenciamientoEvento()); time (Integer) (Duración de la alarma).

Nombre	Tipo	Descripción
events	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: code (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
packetId	Integer	Identificador único del paquete de posición.
integratorId	Integer	Identificador único de la integradora.
driverId	Integer	Retorna siempre null.
driverName	String	Retorna siempre null.
fuelLevel	Integer	Porcentaje de llenado del tanque de combustible en el momento de la captura de la posición.
lithometer	Double	Cantidad de combustible en litros consumida durante la vida útil del vehículo (combustible total). Retorna "-1.0" si la información del litrómetro no está disponible en el vehículo.
windshieldWiperState	Integer	Retorna siempre null.
telemetryEvents	Array<int>	Lista de evento(s) de telemetría compuesta por los campos: • idEvent — Identificador del Evento • durationTime — Duración del Evento en segundos • maximumSpeedEvent — Velocidad Máxima alcanzada en el Evento • speedReference — Velocidad de Referencia
humiditySerial	Integer	Información del sensor de humedad en porcentaje multiplicado por 10. Ejemplo: 534 = 53,4%. No aparece si no está instalado.
temperatureSerial	Integer	Información del sensor de temperatura en grados Celsius multiplicado por 10. Ejemplo: 248 = 24,8°C. No aparece si no está instalado.

Nombre	Tipo	Descripción
acessorios	Json	Este campo retorna la lista de accesorios vinculados al componente "Rede de Acessorios". La red de accesorios está preparada para recibir una cantidad indefinida de accesorios que se agregan al rastreador sin necesidad de ocupar los puertos físicos. Cada uno de los sensores envía los siguientes campos: Número de Serie * Tipo Índice Valor Local *El campo serial vendrá vacío y su implementación ocurrirá en un desarrollo futuro. El campo Tipo puede contener los siguientes valores: ENTRADA_DIGITAL 1 SAIDA_DIGITAL 2 TEMPERATURA 3 UMIDADE 4 TENSAO 5 ANALOGICO 6 INDEFINIDO 65.535

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getPositionPacketByRangeJSON>
      <user>Usuário</user>
      <password>Senha</password>
      <!--Optional:-->
      <startId>81336079</startId>
      <!--Optional:-->
      <endId/>
      <!--Optional:-->
      <quantity>Quantidade</quantity>
    </web:getPositionPacketByRangeJSON>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:getPositionPacketByRangeJSONResponse>
```

```

xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
  <return>
  {
    "vehicleId": 1563022,
    "positionDateUtc": "2023-10-16 14:06:57.0",
    "packetDateUtc": "2023-10-16 14:06:56.0",
    "latitude": -19.948707,
    "longitude": -44.010242,
    "direction": 67,
    "speed": 11,
    "ignition": 1,
    "odometer": 323519,
    "horimeter": 191239,
    "power": 27,
    "output1": 0,
    "output2": 0,
    "output3": 0,
    "output4": 0,
    "input1": 0,
    "input2": 0,
    "input3": 0,
    "input4": 0,
    "satellite": 0,
    "memory": 0,
    "referenceId": 0,
    "blocking": 1,
    "gps": 1,
    "state": "MG",
    "city": "Belo Horizonte",
    "country": "BR",
    "street": "Av Amazonas",
    "referencePoint": "PE - Posto 040-MG (2)",
    "referenceAngle": 12,
    "referenceDistance": 1054,
    "rpm": 956,
    "temperature1": -125,
    "temperature2": -125,
    "temperature3": -125,
    "output5": 0,
    "output6": -232,
    "output7": 0,
    "output8": 0,
    "input5": 0,
    "input6": 0,
    "input7": 0,
    "input8": 0,
    "entryPoint": 0,
    "exitPoint": 0,
    "macroCode": 0,
    "messageName": "",
    "messageContent": "",
    "messageText": "",
    "mdtType": 0,
    "sequencingEvent": [],
    "events": [],
    "jamming": 0,
    "anchorStatus": 0,
    "packetId": 81336079,
    "integratorId": 80,
    "driverId": 0,
    "driverName": "",
    "fuelLevel": 99,
    "lithometer": 182447.616,
    "windshieldWiperState": 0,
    "telemetryEvents": [
      {
        "eventId": 227,
        "durationTime": 11
      }
    ]
  }

```

```
        }  
      ],  
      "humiditySerial": -1250,  
      "temperatureSerial": -1250  
    }  
  </return>  
</ns0:getPositionPacketByRangeJSONResponse>  
</S:Body>  
</S:Envelope>
```

5.20. getDriverPositionPacketByRangeJSON

Descripción

Método que permite obtener uno o más paquetes de posiciones específicos, a través del identificador único de cada paquete. Este método retorna la misma información que el método ObterPacotePosicoesMotorista, en inglés y presentando las fechas en UTC-0.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List getDriverPositionPacketByRangeJSON(String user, String password, Integer startId, Integer endId, Integer quantity)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
user	String	Sí	Login del integrador para acceso al sistema.
password	String	Sí	Contraseña del integrador para acceso al sistema.
startId	Integer	Sí	Identificador del paquete de posiciones a partir del cual se desea que el web service reenvíe.
endId	Integer	Sí	Identificador del paquete de posiciones hasta el cual se desea que el web service reenvíe.
quantity	Integer	Sí	Límite de registros como resultado de la consulta. El sistema posee un límite estándar de 3000 registros, que se aplicará en los siguientes casos: Cuando el cliente no informe este parámetro; Cuando el cliente informe un número mayor que el límite estándar.



Si se desea recibir solo un paquete de posiciones, se debe informar el mismo startId e endId.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
vehicleId	Integer	Código del vehículo.
positionDateUtc	Calendar	Fecha de grabación de la posición en el servidor, en UTC-0.

Nombre	Tipo	Descripción
packetDateUtc	Calendar	Fecha del GPS (GMT) en el momento del envío de la posición, en UTC-0.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.
direction	Integer	Dirección del vehículo: • 0 — N - Norte • 1 — NE - Nordeste • 2 — E - Este • 3 — SE - Sudeste • 4 — S - Sur • 5 — SO - Sudoeste • 6 — O - Oeste • 7 — NO - Noroeste
speed	Integer	Velocidad en km/h obtenida del GPS. Si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
ignition	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
odometer	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
horimeter	Integer	Horómetro actual del vehículo, en minutos. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
power	Integer	Valor de tensión de la batería, en Voltios.
output1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
output2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: 240 (Código de sirena informando que está activada); -240 (Código de sirena informando que está desactivada); 0 (Ningún dispositivo).
output3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

Nombre	Tipo	Descripción
output4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satellite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIVRE IMPORTANTE: Si el paquete es del tipo “2” (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satellite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memory	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
referenceId	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
blocking	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado

Nombre	Tipo	Descripción
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido
state	String	Provincia/Estado.
city	String	Ciudad.
street	String	Dirección.
country	String	País.
referencePoint	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
referenceAngle	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
referenceDistance	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperature1	Integer	Datos del sensor de temperatura 1, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
temperature2	Integer	Datos del sensor de temperatura 2, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
temperature3	Integer	Datos del sensor de temperatura 3, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
output5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

Nombre	Tipo	Descripción
input6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entryPoint	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
exitPoint	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
macroCode	Integer	Código de la macro.
messageName	String	Nombre del mensaje tipo macro.
messageContent	String	Contenido del mensaje tipo macro, separado por guión bajo (). <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
messageText	String	Contenido del mensaje tipo texto libre.
mdtType	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
sequencingEvent	EventoSequenciamen to[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: eventId (Integer) (id del secuenciamiento relacionado con la tabla 'sequenciamiento_evento' o con el resultado del método ObterSequenciamientoEvento()); time (Integer) (Duración de la alarma).

Nombre	Tipo	Descripción
events	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: code (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
packetId	Integer	Identificador único del paquete de posición.
integratorId	Integer	Identificador único de la integradora.
driverId	Integer	Identificación del conductor.
driverName	String	Nombre del conductor conectado.
fuelLevel	Integer	Porcentaje de llenado del tanque de combustible en el momento de la captura de la posición.
lithometer	Double	Cantidad de combustible en litros consumida durante la vida útil del vehículo (combustible total). Retorna "-1.0" si la información del litrómetro no está disponible en el vehículo.
windshieldWiperState	Integer	Estado del limpiaparabrisas: • 0 — Limpiaparabrisas No Accionado • 1 — Limpiaparabrisas Accionado
telemetryEvents	Array<int>	Lista de evento(s) de telemetría compuesta por los campos: • idEvent — Identificador del Evento • durationTime — Duración del Evento en segundos • maximumSpeedEvent — Velocidad Máxima alcanzada en el Evento • speedReference — Velocidad de Referencia
humiditySerial	Integer	Información del sensor de humedad en porcentaje multiplicado por 10. Ejemplo: 534 = 53,4%. No aparece si no está instalado.
temperatureSerial	Integer	Información del sensor de temperatura en grados Celsius multiplicado por 10. Ejemplo: 248 = 24,8-C. No aparece si no está instalado.

Nombre	Tipo	Descripción
acessorios	Json	Este campo retorna la lista de accesorios vinculados al componente "Rede de Acessorios". La red de accesorios está preparada para recibir una cantidad indefinida de accesorios que se agregan al rastreador sin necesidad de ocupar los puertos físicos. Cada uno de los sensores envía los siguientes campos: Número de Serie * Tipo Índice Valor Local *El campo serial vendrá vacío y su implementación ocurrirá en un desarrollo futuro. El campo Tipo puede contener los siguientes valores: ENTRADA_DIGITAL 1 SAIDA_DIGITAL 2 TEMPERATURA 3 UMIDADE 4 TENSAO 5 ANALOGICO 6 INDEFINIDO 65.535

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getDriverPositionPacketByRangeJSON>
      <user>Usuário</user>
      <password>Senha</password>
      <!--Optional:-->
      <startId>81336079</startId>
      <!--Optional:-->
      <endId/>
      <!--Optional:-->
      <quantity?></quantity>
    </web:getDriverPositionPacketByRangeJSON>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:getDriverPositionPacketByRangeJSONResponse>
```

```

xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
  <return>
  {
    "vehicleId": 900757,
    "positionDateUtc": "2023-10-16 14:21:34.0",
    "packetDateUtc": "2023-10-16 14:21:41.0",
    "latitude": -25.4514864,
    "longitude": -49.2461824,
    "direction": 193,
    "speed": 2,
    "ignition": 1,
    "odometer": 561,
    "horimeter": 262214,
    "power": 11,
    "output1": 0,
    "output2": -240,
    "output3": 254,
    "output4": 253,
    "input1": 0,
    "input2": 0,
    "input3": -231,
    "input4": -247,
    "satellite": 0,
    "memory": 0,
    "referenceId": 2387650,
    "blocking": 0,
    "gps": 0,
    "state": "PR",
    "city": "Curitiba",
    "country": "BR",
    "street": "R Joao Marchesini",
    "referencePoint": "novo",
    "referenceAngle": 0,
    "referenceDistance": 0,
    "rpm": 0,
    "temperature1": -125,
    "temperature2": -125,
    "temperature3": -125,
    "output5": 245,
    "output6": 0,
    "output7": -232,
    "output8": 0,
    "input5": -248,
    "input6": -241,
    "input7": -250,
    "input8": -251,
    "entryPoint": 0,
    "exitPoint": 0,
    "macroCode": 0,
    "messageName": "",
    "messageContent": "",
    "messageText": "",
    "mdtType": 0,
    "sequencingEvent": [],
    "events": [],
    "jamming": 0,
    "anchorStatus": 0,
    "packetId": 81336320,
    "integratorId": 80,
    "driverId": 3539390,
    "driverName": "Jean Santos",
    "fuelLevel": 0,
    "lithometer": 0,
    "windshieldWiperState": 0,
    "telemetryEvents": [],
    "humiditySerial": 0,
    "temperatureSerial": 0
  }

```

```
</return>
</ns0:getDriverPositionPacketByRangeJSONResponse>
</S:Body>
</S:Envelope>
```

5.21. getPositionPacketWithLicensePlateJSON

Descripción

Método para consultar la lista de paquetes de posiciones enviados por los vehículos, cuyo contenido está compuesto por los siguientes bloques de información: Paquete de posición; Paquete de mensajes (todos los tipos); Paquete de eventos; Paquete Alarma de Secuenciamiento.

Basado en el método obterPacotePosicoesJSONComPlaca, presenta la información en inglés y con las fechas en UTC-0.



Si el cliente tiene restricciones, no será posible consumir paquetes de ese período; el mismo podrá obtenerse mediante el método obterPacotePosicoesRestricao.

Declaración del Método

```
List getPositionPacketWithLicensePlateJSON(String user, String password, Integer quantity)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
user	String	Sí	Login del integrador para acceso al sistema.
password	String	Sí	Contraseña del integrador para acceso al sistema.
quantity	Integer	No	Límite de registros como resultado de la consulta. El sistema posee un límite estándar de 3000 registros, que se aplicará en los siguientes casos: Cuando el cliente no informe este parámetro; Cuando el cliente informe un número mayor que el límite estándar.

Retorno

Retorna una lista de objetos del tipo 'PacotePosicao', con el retorno de los datos más antiguos a los más recientes (D-1 y día actual), con las propiedades a continuación:

Nombre	Tipo	Descripción
vehicleId	Integer	Código del vehículo.
positionDateUtc	Calendar	Fecha de grabación de la posición en el servidor, en UTC-0.
packetDateUtc	Calendar	Fecha del GPS (GMT) en el momento del envío de la posición, en UTC-0.
latitude	Double	Latitud en formato decimal.
longitude	Double	Longitud en formato decimal.

Nombre	Tipo	Descripción
direction	Integer	Dirección del vehículo: • 0 — N - Norte • 1 — NE - Nordeste • 2 — E - Este • 3 — SE - Sudeste • 4 — S - Sur • 5 — SO - Sudoeste • 6 — O - Oeste • 7 — NO - Noroeste
speed	Integer	Velocidad en km/h obtenida del GPS. Si el vehículo posee Telemetría, la velocidad será la de la Telemetría.
ignition	Integer	Estado del encendido: • 1 — Encendido • 0 — Apagado
odometer	Integer	Odómetro obtenido del GPS; si el vehículo posee Telemetría, se utilizará el de la Telemetría.
horimeter	Integer	Horómetro actual del vehículo, en minutos. Cuando el campo "SATELITE" sea igual a 1 o 2, este campo vendrá = 0 (cero) para tecnologías MSC. El valor actual del horómetro sólo se enviará cuando "Satelite=0", informando que el mensaje fue enviado vía GPRS.
power	Integer	Valor de tensión de la batería, en Voltios.
output1	Integer	Estado del dispositivo instalado en la salida 1, que por defecto de Sascar es el bloqueo del equipo.
output2	Integer	Estado del dispositivo instalado en la salida 2. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado, cuyo valor positivo indica que está activo y valor negativo indica que está inactivo. Por ejemplo: 240 (Código de sirena informando que está activada); -240 (Código de sirena informando que está desactivada); 0 (Ningún dispositivo).
output3	Integer	Estado del dispositivo instalado en la salida 3. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output4	Integer	Estado del dispositivo instalado en la salida 4. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

Nombre	Tipo	Descripción
input1	Integer	Estado del dispositivo instalado en la entrada 1. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input2	Integer	Estado del dispositivo instalado en la entrada 2. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input3	Integer	Estado del dispositivo instalado en la entrada 3. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input4	Integer	Estado del dispositivo instalado en la entrada 4. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
satellite	Integer	Informa el origen del paquete: • 0 — GPRS • 1 — SATELITAL • 2 — SATELITAL TEXTO LIVRE IMPORTANTE: Si el paquete es del tipo "2" (Satelital texto libre), considere sólo los campos siguientes: idVeiculo idPacote dataPosicao dataPacote latitude longitude textoMensagem satellite = 2. Los demás deben descartarse, ya que no se usan en la transmisión del mensaje satelital con texto libre.
memory	Integer	Informa el tipo del paquete: • 0 — Online • 1 — Memoria
referenceId	Integer	Identificación del punto de referencia. Este id está relacionado con la tabla 'ponto_referencia' o con el resultado del método ObterPontosReferencia(), cuyo retorno es la lista de puntos de referencia creados en el SASGC y marcados como EMBARCABLES. Este es el id del punto de referencia EMBARCADO (dentro del equipo; puede no estar embarcado por la gestora que está consumiendo el paquete).
blocking	Integer	Informa si el vehículo está bloqueado o no: • 0 — Desbloqueado • 1 — Bloqueado
gps	Integer	Estado de la señal: • 0 — GPS inválido • 1 — GPS válido

Nombre	Tipo	Descripción
state	String	Provincia/Estado.
city	String	Ciudad.
street	String	Dirección.
country	String	País.
referencePoint	String	Nombre del punto de referencia más cercano registrado por la gestora que está consumiendo el paquete.
referenceAngle	Integer	Ángulo en relación al punto de referencia (Pacote.pontoReferencia), registrado por la gestora que está consumiendo el paquete.
referenceDistance	Integer	Distancia en metros al punto de referencia más cercano, registrado por la gestora que está consumiendo el paquete.
rpm	Integer	Datos de RPM (disponible para vehículos con multisensor).
temperature1	Integer	Datos del sensor de temperatura 1, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
temperature2	Integer	Datos del sensor de temperatura 2, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
temperature3	Integer	Datos del sensor de temperatura 3, en grados Celsius. Si presenta el valor -125, indica que no hay valor disponible o válido para ese sensor.
output5	Integer	Estado del dispositivo instalado en la salida 5. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output6	Integer	Estado del dispositivo instalado en la salida 6. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output7	Integer	Estado del dispositivo instalado en la salida 7. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
output8	Integer	Estado del dispositivo instalado en la salida 8. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input5	Integer	Estado del dispositivo instalado en la entrada 5. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input6	Integer	Estado del dispositivo instalado en la entrada 6. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).

Nombre	Tipo	Descripción
input7	Integer	Estado del dispositivo instalado en la entrada 7. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
input8	Integer	Estado del dispositivo instalado en la entrada 8. Retorna 0 (cero) si no hay dispositivo instalado o el código del dispositivo instalado con valor positivo (dispositivo activo) o negativo (dispositivo inactivo).
entryPoint	Integer	Informa si el vehículo entró en un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Entró al punto
exitPoint	Integer	Informa si el vehículo salió de un punto embarcado en el vehículo: • 0 — No hubo cambio • 1 — Salida del punto
macroCode	Integer	Código de la macro.
messageName	String	Nombre del mensaje tipo macro.
messageContent	String	Contenido del mensaje tipo macro, separado por guión bajo (. <i>Por ejemplo, si la macro contiene los campos CONDUCTOR y CONTRASEÑA, el contenido del mensaje se presentará de la siguiente manera: _João da Silva_.</i>
messageText	String	Contenido del mensaje tipo texto libre.
mdtType	Integer	Identificación del tipo de teclado: • 0 — No contiene mensaje en el paquete • 1 — TD50 • 3 — TD40 • 5 — TMCD • 7 — TMS3 • 8 — SASMDT
sequencingEvent	EventoSecuenciamen to[]	Lista con las alarmas de secuenciamiento de macros. Cada objeto de la lista se formará con las siguientes propiedades: eventId (Integer) (id del secuenciamiento relacionado con la tabla 'secuenciamiento_evento' o con el resultado del método ObterSecuenciamientoEvento()); time (Integer) (Duración de la alarma).

Nombre	Tipo	Descripción
events	Evento[]	Lista de los eventos generados por el equipo. Cada objeto de la lista se formará con la siguiente propiedad: code (Integer) — Código del actuador que generó el evento, cuyo valor puede ser positivo indicando que fue activado o negativo indicando que fue desactivado. Este código está relacionado con la tabla atuador_grupo o con el resultado del método ObterGrupoAtuadores().
jamming	Integer	Informa la ocurrencia de un evento de jamming por el equipo: • 0 — No hubo jamming • 1 — Ocurrió jamming
packetId	Integer	Identificador único del paquete de posición.
integratorId	Integer	Identificador único de la integradora.
driverId	Integer	Retorna siempre null.
driverName	String	Retorna siempre null.
fuelLevel	Integer	Porcentaje de llenado del tanque de combustible en el momento de la captura de la posición.
lithometer	Double	Cantidad de combustible en litros consumida durante la vida útil del vehículo (combustible total). Retorna "-1.0" si la información del litrómetro no está disponible en el vehículo.
windshieldWiperState	Integer	Retorna siempre null.
telemetryEvents	Array<int>	Lista de evento(s) de telemetría compuesta por los campos: • idEvent — Identificador del Evento • durationTime — Duración del Evento en segundos • maximumSpeedEvent — Velocidad Máxima alcanzada en el Evento • speedReference — Velocidad de Referencia
humiditySerial	Integer	Información del sensor de humedad en porcentaje multiplicado por 10. Ejemplo: 534 = 53,4%. No aparece si no está instalado.
temperatureSerial	Integer	Información del sensor de temperatura en grados Celsius multiplicado por 10. Ejemplo: 248 = 24,8°C. No aparece si no está instalado.
licensePlate	String	Patente del vehículo.

Nombre	Tipo	Descripción
acessorios	Json	Este campo retorna la lista de accesorios vinculados al componente "Rede de Acessorios". La red de accesorios está preparada para recibir una cantidad indefinida de accesorios que se agregan al rastreador sin necesidad de ocupar los puertos físicos. Cada uno de los sensores envía los siguientes campos: Número de Serie * Tipo Índice Valor Local *El campo serial vendrá vacío y su implementación ocurrirá en un desarrollo futuro. El campo Tipo puede contener los siguientes valores: ENTRADA_DIGITAL 1 SAIDA_DIGITAL 2 TEMPERATURA 3 UMIDADE 4 TENSAO 5 ANALOGICO 6 INDEFINIDO 65.535

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- pacote
- evento
- mensagem_td50
- mensagem_tmcd
- teclado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getPositionPacketWithLicensePlateJSON>
      <user>Usuário</user>
      <password>Senha</password>
      <!--Optional:-->
      <quantity>aaa</quantity>
    </web:getPositionPacketWithLicensePlateJSON>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:getPositionPacketWithLicensePlateJSONResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return> { "vehicleId": 608581, "positionDateUtc": "2023-10-16 16:49:06.0",
"packetDateUtc": "2023-10-16 16:49:05.0", "latitude": -25.4516521, "longitude": -49.2460098,
"direction": 189, "speed": 0, "ignition": 0, "odometer": 2662, "horimeter": 1496825, "power": 11,
```

```
"output1": 0, "output2": -240, "output3": -254, "output4": -253, "input1": 0, "input2": -250,
"input3": -251, "input4": -241, "satellite": 0, "memory": 0, "referenceId": 0, "blocking": 0, "gps":
1, "state": "PR", "city": "Curitiba", "country": "BR", "street": "R Joao Marchesini",
"referencePoint": "novo", "referenceAngle": 0, "referenceDistance": 0, "rpm": 0, "temperature1": 23,
"temperature2": -125, "temperature3": -125, "output5": 0, "output6": -232, "output7": -232, "output8":
0, "input5": -247, "input6": -248, "input7": -231, "input8": 0, "entryPoint": 0, "exitPoint": 0,
"macroCode": 0, "messageName": "", "messageContent": "", "messageText": "", "mdtType": 0,
"sequencingEvent": [], "events": [], "jamming": 0, "anchorStatus": 0, "packetId": 81339207,
"integratorId": 80, "driverId": null, "driverName": "null", "fuelLevel": 0, "lithometer": 0,
"windshieldWiperState": null, "humiditySerial": -1250, "temperatureSerial": -1250, "licensePlate":
AND3007 } </return>
</ns0:getPositionPacketWithLicensePlateJSONResponse>
</S:Body>
</S:Envelope>
```

6. Métodos – Layout e Macros

6.1. ObterMacroTd50Tmcd

Descripción

Método para consultar el layout/formato de las macros de los teclados TD50 y TMCD que están cargadas en los vehículos de una gestora.

Declaración del Método

```
List obterMacroTd50Tmcd(String usuario, String senha, TipoTeclado teclado)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
teclado	TipoTeclado	Sí	Tipo de teclado a consultar. Este parámetro es un enumerador con los tipos de teclados admitidos por este método: - TECLADO_TD50 - TECLADO_TMCD

Retorno

Retorna una lista de objetos do tipo ' MacroTd50Tmcd' com as propriedades abaixo:

Nombre	Tipo	Descripción
idMacroTd50Tmcd	Integer	Código de la macro.
idVeiculo	Integer	Código del vehículo que tiene la macro.
nome	String	Nombre de la macro.
layout	String	Layout/formato de la macro.
layoutDetalhado	String	Retorna detalles de cada layout personalizado



La información obtenida puede utilizarse para alimentar las tablas 'macro_td50' o 'macro_tmcd' de la base de datos de integración de acuerdo con el tipo de teclado consultado. Si la macro está contenida en más de un vehículo, se devolverá varias veces cambiando únicamente el id del vehículo.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterMacroTd50Tmcd>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <tipoTeclado>TIPO_TECLADO</tipoTeclado>
    </web:obterMacroTd50Tmcd>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterMacroTd50TmcdResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <idMacroTd50Tmcd>2</idMacroTd50Tmcd>
        <idVeiculo>1363161</idVeiculo>
        <layout/>
        <layoutDetalhado/>
        <nome>BLOQUEIO IGNICAO</nome>
      </return>
    </ns0:obterMacroTd50TmcdResponse>
  </S:Body>
</S:Envelope>
```

6.2. ObterMacroTd50TmcdDetalhado

Descripción

Método para consultar el layout/formato de las macros de los teclados TD50 y TMCD que están cargadas en los vehículos de una gestora.

Declaración del Método

```
List obterMacroTd50Tmcd(String usuario, String senha, TipoTeclado teclado)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
teclado	TipoTeclado	Sí	Tipo de teclado a consultar. Este parámetro es un enumerador con los tipos de teclados admitidos por este método: • TECLADO_TD50 • TECLADO_TMCD

Retorno

Retorna una lista de objetos del tipo 'MacroTd50Tmcd', que tiene la lista de layouts de la macro y la lista de vehículos que tiene la macro, con las propiedades a continuación:

Nombre	Tipo	Descripción
idMacroTd50Tmcd	Integer	Código de la macro.
nome	String	Nombre de la macro.
listaLayout	Array	Lista de layouts de la macro embarcados en los vehículos de la gestora, compuesto por: • Descricao — descripción del layout • IdLayout — Identificador do layout
listaVeiculos	Array	Lista de todos los vehículos que tienen la macro en cuestión.



La información obtenida puede utilizarse para alimentar las tablas 'macro_td50' o 'macro_tmcd' de la base de datos de integración de acuerdo con el tipo de teclado consultado.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterMacroTd50TmcdDetalhado>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <tipoTeclado>TIPO_TECLADO</tipoTeclado>
    </web:obterMacroTd50TmcdDetalhado>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterMacroTd50TmcdDetalhadoResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <idMacroTd50Tmcd>5004</idMacroTd50Tmcd>
        <listaLayout>
          <descricao>PADRAO</descricao>
          <idLayout>1521</idLayout>
        </listaLayout>
        <listaVeiculos>1436376</listaVeiculos>
        <nome>REI</nome>
      </return>
    </ns0:obterMacroTd50TmcdDetalhadoResponse>
  </S:Body>
</S:Envelope>
```

6.3. ObterMascaraDispositivo

Descripción

Método para consultar la relación de sensores o actuadores inhibidos de un vehículo.

Declaración del Método

```
MascaraDispositivo obterMascaraDispositivo(String usuario, String senha, Integer idVeiculo)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idVeiculo	Integer	Sí	Código del vehículo a consultar.

Retorno

Retorna un objeto del tipo 'MascaraDispositivo' con la propiedad a continuación:

Nombre	Tipo	Descripción
atuadores	Integer[]	Lista con los actuadores o sensores inhibidos en el vehículo informado como parámetro.



La información obtenida puede utilizarse para alimentar la tabla 'mascara_dispositivo' de la base de datos de integración.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterMascaraDispositivos>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <idVeiculo>IDVeiculo</idVeiculo>
    </web:obterMascaraDispositivos>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns2:obterMascaraDispositivosResponse xmlns:ns2="http://ws.integracao.sascar.com/">
      <return>
        <atuadores>1</atuadores>
        <atuadores>2</atuadores>
        <atuadores>3</atuadores>
      </return>
    </ns2:obterMascaraDispositivosResponse>
  </S:Body>
</S:Envelope>
```

6.4. ObterMacroTD40

Descripción

Método para consultar las macros del teclado TD40 que están cargadas en los vehículos de una gestora.

Declaración del Método

```
List obterMacroTD40(String usuario, String senha, Integer idVeiculo, boolean satelital)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idVeiculo	Integer	Sí	Código del vehículo, que puede obtenerse mediante el método obterVeiculos.
satelital	Boolean	Sí	Parámetro para consulta de equipos con antena satelital, siendo sus valores: • <code>true</code> — Tienen antena satelital • <code>false</code> — No tienen antena satelital

Retorno

Retorna una lista de objetos del tipo 'MacroTd40' con las propiedades a continuación:

Nombre	Tipo	Descripción
idMacroTd40	Integer	Código de la macro enviada por el vehículo.
idVeiculo	Integer	Id del vehículo que tiene la macro.
Mensagem	String	Contenido de la macro.
tipoMensagem	Integer	Tipo del mensaje: • <code>0</code> — Mensaje • <code>1</code> — Defecto



La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:

- 'mensagem_teclado' para vehículos que no tienen antena satelital;

- 'mensagem_teclado_avl' para veículos que tienen antena satelital.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterMacroTd40>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <satelital>BOOLEAN</satelital>
    </web:obterMacroTd40>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns2:obterMacroTd40Response xmlns:ns2="http://ws.integracao.sascar.com/">
      <return>
        <idMacroTd40>1</idMacroTd40>
        <mensagem>Mensagem</mensagem>
        <tipoMensagem>1</tipoMensagem>
      </return>
    </ns2:obterMacroTd40Response>
  </S:Body>
</S:Envelope>
```

6.5. ObterLayout

Descripción

Método para consultar la relación de layouts disponibles para ser enviados a los teclados mediante comando en el Web Service XML-RPC.

Declaración del Método

```
List obterLayout(String usuario, String senha, TipoLayout layout)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
layout	TipoLayout	Sí	Tipo de layout a consultar. Este parámetro es un enumerador con los tipos de layouts existentes: • LAYOUT_TD40 • LAYOUT_TD50 • LAYOUT_TMCD • LAYOUT_SEQUENCIAMENTO_TD50 • LAYOUT_TMS3

Retorno

Retorna una lista de objetos del tipo 'Layout' con las propiedades a continuación:

Nombre	Tipo	Descripción
idLayout	Integer	Código interno Sascar del layout y permite el envío mediante el comando de integración.
descricao	String	Nombre del layout.
tipoTeclado	Integer	Tipo de teclado informado: • 0 — TD40 • 1 — TD50 • 2 — TMCD • 3 — Sequenciamento TD50 • 4 — TMS3

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- tipo_layout_teclado
- tipo_layout_teclado_td50
- tipo_layout_teclado_tmcd
- tipo_sequenciamento_macro_td50

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterLayout>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <layout>TIPO_LAYOUT</layout>
    </web:obterLayout>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterLayoutResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <descricao>Fitel</descricao>
        <idLayout>1826</idLayout>
        <tipoTeclado>0</tipoTeclado>
      </return>
    </ns0:obterLayoutResponse>
  </S:Body>
</S:Envelope>
```

6.6. ObterLayoutDetalhado

Descripción

Método para consultar la relación de layouts disponibles y sus respectivas macros para ser enviados a los teclados mediante comando en el Web Service XML-RPC.

Declaración del Método

```
List obterLayoutDetalhado(String usuario, String senha, TipoLayout layout, Integer idLayout, String dataReferencia)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
layout	TipoLayout	Sí	Tipo de layout a consultar. Este parámetro es un enumerador con los tipos de layouts existentes: • LAYOUT_TD40 • LAYOUT_TD50 • LAYOUT_TMCD • LAYOUT_SEQUENCIAMENTO_TD50 • LAYOUT_TMS3
idLayout	Integer	No	Para obtener la información pertinente solo a un registro de layout, enviar únicamente el id del layout deseado.
dataReferencia	String	No	Filtro utilizado para recuperar layouts que fueron creados o modificados después de la fecha informada. Formato fecha: YYYY-DD-MM HH:MM:SS



Para os teclados do tipo SASMDT, utilizar os os [LAYOUT_SEQUENCIAMENTO_TD50](#).

Retorno

Retorna una lista de objetos del tipo 'Layout' con las propiedades a continuación:

Nombre	Tipo	Descripción
codigo	Integer	Código interno Sascar del layout y permite el envío mediante el comando de integración.
descricao	String	Nombre del layout.

Nombre	Tipo	Descripción
macros	ArrayList	Lista de descripciones de macros (lista de macros). Tiene los campos: idMacro – Identificador de la macro descricao – Nombre de la macro layout – Formato de la Macro

La información obtenida puede utilizarse para alimentar las siguientes tablas de la base de datos de integración:



- tipo_layout_teclado
- tipo_layout_teclado_td50

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterLayoutDetalhado>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <layout>TIPO_LAYOUT</layout>
      <!--Optional:-->
      <idLayout/>
      <!--Optional:-->
      <dataReferencia/>
    </web:obterLayoutDetalhado>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XUZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterLayoutDetalhadoResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <descricao>Fitel</descricao>
        <idLayout>1826</idLayout>
        <macros>
          <descricao>INICIO DE VIAGEM FITEL</descricao>
        </macros>
      </return>
    </ns0:obterLayoutDetalhadoResponse>
  </S:Body>
</S:Envelope>
```

6.7. ObterLayoutAcaoEmbarcadaAVD

Descrição

Método para consultar los datos de registro de Layouts de Ação Embarcada AVD del INTEGRADOR.

Declaración del Método

```
List obterLayoutAcaoEmbarcadaAVD(String usuario, String senha)
```

Parâmetros

Nombre	Tipo	Obrigatorio	Descrição
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.

Retorno

Retorna uma lista de objetos do tipo ' LayoutAcaoEmbarcadaAVD ' com as propriedades abaixo:

Nombre	Tipo	Descrição
idLayoutAcaoEmbarcadaAVD	Integer	Id del layout de la Acción Embarcada AVD
nome	Integer	Nombre del layout de la Acción Embarcada AVD

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterLayoutAcaoEmbarcadaAVD>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
    </web:obterLayoutAcaoEmbarcadaAVD>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterLayoutAcaoEmbarcadaAVDResponse
```

```
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
  <return>
    <idLayoutAcaoEmbarcadaAVD>1229</idLayoutAcaoEmbarcadaAVD>
    <nome>TEMPO_PANICO</nome>
  </return>
</ns0:obterLayoutAcaoEmbarcadaAVDResponse>
</S:Body>
</S:Envelope>
```

6.8. ObterLayoutAreaAvd

Descripción

Método para consultar los datos de Id y nombre de los layouts de áreas registrados por el Integrador.

Declaración del Método

```
List obterLayoutAreaAvd(String usuario, String senha)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.

Retorno

Retorna una lista de objetos del tipo 'ObterLayoutAreaAvd' con las propiedades a continuación:

Nombre	Tipo	Descripción
id	Integer	Id del layout área avd
nome	String	Nombre del layout del área avd

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterLayoutAreaAvd>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
    </web:obterLayoutAreaAvd>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterLayoutAreaAvdResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
```

```
<return>
  <id>1792</id>
  <nome>TREINAMENTO AREA DE RISCO</nome>
</return>
<return>
  <id>1026</id>
  <nome>Teste Area</nome>
</return>
<return>
  <id>1061</id>
  <nome>Layout areas 1</nome>
</return>
<return>
  <id>862</id>
  <nome>Layout areas 2</nome>
</return>
</ns0:obterLayoutAreaAvdResponse>
</S:Body>
</S:Envelope>
```

6.9. ObterLayoutData

Descripción

Método para consultar id, nombre, fecha de modificación y fecha de creación de Layout.

Declaración del Método

```
List obterLayoutData(String usuario, String senha, String layout)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
layout	String	Sí	Indica el tipo de layout de teclado, pudiendo ser: • LAYOUT_TD40 • LAYOUT_TD50 • LAYOUT_TMCD • LAYOUT_SEQUENCIAMENTO_TD50 • LAYOUT_TMS3

Retorno

Retorna una lista de objetos del tipo 'ObterLayoutData' con las propiedades a continuación:

Nombre	Tipo	Descripción
dataAlteracao	Calendar	Fecha de modificación
dataCriacao	Calendar	Fecha de creación
descricao	String	Nombre del layout
idLayout	Integer	Id del layout
TipoTeclado	Integer	Tipo de teclado utilizado 0 - TD40; 1 - TD50; 2 - TMCD; 3 - Secuenciamiento TD50. 4 - TMS3

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
```

```
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterLayoutData>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <layout/>
    </web:obterLayoutData>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterLayoutDataResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <dataAlteracao>2008-09-01 21:16:26</dataAlteracao>
        <dataCriacao>2008-09-01 21:16:26</dataCriacao>
        <descricao>Fitel</descricao>
        <idLayout>1826</idLayout>
        <tipoTeclado>0</tipoTeclado>
      </return>
    </ns0:obterLayoutDataResponse>
  </S:Body>
</S:Envelope>
```

6.10. ObterLayoutTecladoVeiculos

Descripción

Método para consultar los vínculos entre vehículos y layouts de teclado pertenecientes al INTEGRADOR.

Declaración del Método

```
List obterLayoutTecladoVeiculos(String usuario, String senha)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.

Retorno

Retorna una lista de objetos del tipo 'LayoutTecladoVeiculos' con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Código del vehículo.
idLayout	Integer	Id del layout de teclado.
tipoLayout	String	Indica el tipo de layout de teclado: • LAYOUT_TD40 • LAYOUT_TD50 • LAYOUT_TMCD • LAYOUT_SEQUENCIAMENTO_TD50 • LAYOUT_TMS3

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/" >
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterLayoutTecladoVeiculos>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
    </web:obterLayoutTecladoVeiculos>
  </soapenv:Body>
```

```
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterLayoutTecladoVeiculosResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/" >
      <return>
        <idLayout>1777</idLayout>
        <idVeiculo>89533</idVeiculo>
        <tipoLayout>LAYOUT_SEQUENCIAMENTO_TD50</tipoLayout>
      </return>
    </ns0:obterLayoutTecladoVeiculosResponse>
  </S:Body>
</S:Envelope>
```

6.11. ObterLayoutGrupoPontos

Descripción

Método para consultar los datos de registro de layouts de grupo de puntos pertenecientes al INTEGRADOR.

Declaración del Método

```
List obterLayoutGrupoPontos(String usuario, String senha)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.

Retorno

Retorna una lista de objetos del tipo 'LayoutTecladoVeiculos' con las propiedades a continuación:

Nombre	Tipo	Descripción
idLayoutGrupoPonto	Integer	Id del layout del grupo de puntos
nome	Integer	Nombre del layout de grupo de puntos

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/" >
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterLayoutGrupoPontos>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
    </web:obterLayoutGrupoPontos>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterLayoutGrupoPontosResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/" >
```

```
<return>
  <idLayoutGrupoPonto>982</idLayoutGrupoPonto>
  <nome>ALMIR</nome>
</return>
</ns0:obterLayoutGrupoPontosResponse>
</S:Body>
</S:Envelope>
```

7. Métodos – Telemetria e Câmeras

7.1. ObterEventosTempoDirecao

Descripción

Método para consultar la información de Tiempo de Conducción de los conductores. Todos los eventos de Tiempo de Conducción enviados por el conductor utilizando la tablet (SasMDT) pueden ser accedidos mediante este método. Además del evento actual, el último evento, latitud, longitud, conductor de reserva, odómetro y dirección (calle, ciudad y provincia) también se muestran en el retorno del método.

En la integración, la implementación para entender el **INICIO de JORNADA** debe realizarse en conjunto con el evento anterior **ENCERRAR**. Este par de eventos es el que determinará el fin de una jornada y el inicio de la siguiente.



Durante la jornada, ocurrirá esta alternancia entre el evento de **JORNADA** y **PARADO**, ya que cada vez que finaliza un evento, como CONDUCIENDO, ESPERA o COMIDA, vuelve a JORNADA/PARADO, que son los eventos de ID 1 y 4 respectivamente.

Entonces la implementación puede considerarse algo similar a:

- **Evento de JORNADA con evento anterior de ENCERRAR** = Inicio efectivo de la jornada.
- **Evento de JORNADA con cualquier otro evento anterior** = Forma parte de la jornada activa y puede considerarse como el evento PARADO, ya que tiene la misma función.

Declaración del Método

```
List obterEventosTempoDirecao(String usuario, String senha, Integer quantidade, Integer idMotorista, String dataInicio, String dataFim)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del usuario para acceso al sistema.
senha	String	Sí	Contraseña del usuario para acceso al sistema..
quantidade	Integer	No	Límite de registros como resultado de la consulta. El sistema tiene un límite estándar de 3.000 registros, que se aplicará en los siguientes casos: cuando el cliente no informe este parámetro; cuando el cliente informe un número mayor que el límite estándar. Para obtener el resto de los registros, el INTEGRADOR debe rehacer la solicitud utilizando como fecha de inicio del parámetro de entrada la última fecha devuelta en el campo dataInicio del parámetro de salida.

Nombre	Tipo	Obligatorio	Descripción
idMotorista	Integer	No	ID del Conductor deseado.
dataInicio	String	Sí	Fecha inicial del período deseado.
dataFim	String	Sí	Fecha final del período deseado.

Retorno

Retorna una lista de objetos del tipo 'EventoTempoDirecao' con las propiedades a continuación:

Nombre	Tipo	Descripción
cidade	String	Descripción de la ciudad de la dirección de la posición del evento.
dataInicio	String	Fecha del evento (macro).
descricaoEventoTempoDirecao	String	Descripción del evento actual.
descricaoEventoTempoDirecaoAnterior	String	Descripción del evento anterior.
eventoTempoDirecao	Integer	Identificación del evento actual.
eventoTempoDirecaoAnterior	Integer	Identificación del evento anterior.
idCliente	Integer	Identificación del cliente.
idMotorista	Integer	Identificación del conductor principal.
idMotoristaReserva	Integer	Identificación del conductor de reserva.
idVeiculo	Integer	Identificación del vehículo.
latitude	Double	Latitud de la posición del evento.
longitude	Double	Longitud de la posición del evento.
nomeCliente	String	Nombre del cliente.
nomeMotorista	String	Nombre del conductor principal.
nomeMotoristaReserva	String	Nombre del conductor de reserva.
odometro	Integer	Odómetro de la posición del evento.
placa	String	Patente del vehículo.
rua	String	Descripción de la calle de la dirección de la posición del evento.
uf	String	Provincia de la dirección de la posición del evento.

Relação Evento Descrição

Id del Evento	Descripción
1	Jornada
2	Dirigindo
3	Pausa
4	Parada
5	Refeição
6	Esperar
7	Encerrar
8	Trocar

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterEventosTempoDirecao>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>?</quantidade>
      <!--Optional:-->
      <idMotorista>?</idMotorista>
      <dataInicio>2023-10-10 00:00:00</dataInicio>
      <dataFim>2023-10-10 23:59:00</dataFim>
    </web:obterEventosTempoDirecao>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterEventosTempoDirecaoResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <cidade>Mogi Guacu</cidade>
        <dataInicio>2023-10-10 01:09:53</dataInicio>
        <descricaoEventoTempoDirecao>Jornada</descricaoEventoTempoDirecao>
        <descricaoEventoTempoDirecaoAnterior>Encerrar</descricaoEventoTempoDirecaoAnterior>
        <eventoTempoDirecao>1</eventoTempoDirecao>
        <eventoTempoDirecaoAnterior>7</eventoTempoDirecaoAnterior>
        <idCliente>IDCLIENTE</idCliente>
        <idMotorista>IDMOTORISTA</idMotorista>
        <idMotoristaReserva>0</idMotoristaReserva>
        <idVeiculo>IDVEICULO</idVeiculo>
        <latitude>-22.3563471</latitude>
        <longitude>-46.9690173</longitude>
        <nomeCliente>NOME CLIENTE</nomeCliente>
        <nomeMotorista>NOME MOTORISTA</nomeMotorista>
        <odometro>281420</odometro>
        <placa>PLACA-VEICULO</placa>
        <rua>Sem nome</rua>
        <uf>SP</uf>
      </return>
    </ns0:obterEventosTempoDirecaoResponse>
  </S:Body>
</S:Envelope>
```

7.2. ObterEventosTempoDirecaoDataChegada

Descripción

Método para consultar la información de Tiempo de Conducción de los conductores que contiene la fecha de llegada inicial y la fecha de llegada final del paquete de posición. Todos los eventos de Tiempo de Conducción enviados por el conductor utilizando la tablet (SasMDT) pueden ser accedidos mediante este método. Además del evento actual, el último evento, latitud, longitud, conductor de reserva, odómetro y dirección (calle, ciudad y provincia) también se muestran en el retorno del método.

En la integración, la implementación para entender el **INICIO de JORNADA** debe realizarse en conjunto con el evento anterior **ENCERRAR**. Este par de eventos es el que determinará el fin de una jornada y el inicio de la siguiente.



Durante la jornada, ocurrirá esta alternancia entre el evento de **JORNADA** y **PARADO**, ya que cada vez que finaliza un evento, como CONDUCIENDO, ESPERA o COMIDA, vuelve a JORNADA/PARADO, que son los eventos de ID 1 y 4 respectivamente.

Entonces la implementación puede considerarse algo similar a:

- **Evento de JORNADA con evento anterior de ENCERRAR** = Inicio efectivo de la jornada.
- **Evento de JORNADA con cualquier otro evento anterior** = Forma parte de la jornada activa y puede considerarse como el evento PARADO, ya que tiene la misma función.

Declaración del Método

```
List obterEventosTempoDirecaoDataChegada(String usuario, String senha, Integer quantidade, Integer idMotorista, String dataInicio, String dataFim, String dataChegadaInicial, String dataChegadaFinal)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del usuario para acceso al sistema.
senha	String	Sí	Contraseña del usuario para acceso al sistema..
quantidade	Integer	No	Límite de registros como resultado de la consulta. El sistema tiene un límite estándar de 3.000 registros, que se aplicará en los siguientes casos: cuando el cliente no informe este parámetro; cuando el cliente informe un número mayor que el límite estándar. Para obtener el resto de los registros, el INTEGRADOR debe rehacer la solicitud utilizando como fecha de inicio del parámetro de entrada la última fecha devuelta en el campo dataInicio del parámetro de salida.
idMotorista	Integer	No	ID del Conductor deseado.

Nombre	Tipo	Obligatorio	Descripción
dataInicio	String	No	Fecha inicial del período deseado.
dataFim	String	No	Fecha final del período deseado.
dataChegadaInicial	String	Sí	Fecha de llegada inicial del paquete de posición.
dataChegadaFinal	String	Sí	Fecha de llegada final del paquete de posición.

Retorno

Retorna una lista de objetos del tipo 'EventoTempoDirecaoDataChegada' con las propiedades a continuación:

Nombre	Tipo	Descripción
cidade	String	Descripción de la ciudad de la dirección de la posición del evento.
dataInicio	String	Fecha del evento (macro).
descricaoEventoTempoDirecao	String	Descripción del evento actual.
descricaoEventoTempoDirecaoAnterior	String	Descripción del evento anterior.
eventoTempoDirecao	Integer	Identificación del evento actual.
eventoTempoDirecaoAnterior	Integer	Identificación del evento anterior.
idCliente	Integer	Identificación del cliente.
idMotorista	Integer	Identificación del conductor principal.
idMotoristaReserva	Integer	Identificación del conductor de reserva.
idVeiculo	Integer	Identificación del vehículo.
latitude	Double	Latitud de la posición del evento.
longitude	Double	Longitud de la posición del evento.
nomeCliente	String	Nombre del cliente.
nomeMotorista	String	Nombre del conductor principal.
nomeMotoristaReserva	String	Nombre del conductor de reserva.
odometro	Integer	Odómetro de la posición del evento.
placa	String	Patente del vehículo.
rua	String	Descripción de la calle de la dirección de la posición del evento.
uf	String	Provincia de la dirección de la posición del evento.
dataChegada	String	Fecha de llegada del paquete de posición.

Relação Evento Descrição

Id del Evento	Descripción
1	Jornada
2	Dirigindo
3	Pausa
4	Parada
5	Refeição
6	Esperar
7	Encerrar
8	Trocar

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterEventosTempoDirecaoDataChegada>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>?</quantidade>
      <!--Optional:-->
      <idMotorista>?</idMotorista>
      <!--Optional:-->
      <dataInicio>2023-10-17 00:00:00</dataInicio>
      <!--Optional:-->
      <dataFim>2023-10-17 23:59:00</dataFim>
      <!--Optional:-->
      <dataChegadaInicial>2023-10-17 00:00:00</dataChegadaInicial>
      <!--Optional:-->
      <dataChegadaFinal>2023-10-17 00:00:00</dataChegadaFinal>
    </web:obterEventosTempoDirecaoDataChegada>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterEventosTempoDirecaoDataChegadaResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <cidade>Mogi Guacu</cidade>
        <dataInicio>2023-10-17 01:24:54</dataInicio>
        <descricaoEventoTempoDirecao>Jornada</descricaoEventoTempoDirecao>
        <descricaoEventoTempoDirecaoAnterior>Dirigindo</descricaoEventoTempoDirecaoAnterior>
        <eventoTempoDirecao>1</eventoTempoDirecao>
        <eventoTempoDirecaoAnterior>2</eventoTempoDirecaoAnterior>
        <idCliente>412543</idCliente>
        <idMotorista>2741963</idMotorista>
        <idMotoristaReserva>0</idMotoristaReserva>
        <idVeiculo>1731832</idVeiculo>
        <latitude>-22.2996305</latitude>
        <longitude>-46.9839418</longitude>
        <nomeCliente>ECO FORTE BIOENERGIA LTDA</nomeCliente>
        <nomeMotorista>GABRIEL APARECIDO ROMAO FILHO</nomeMotorista>
        <odometro>285609</odometro>
        <placa>FDR9I01-2</placa>
        <rua>Rodovia Vicinal Vice-Governador Almino Monteiro Alvares Afonso</rua>
        <uf>SP</uf>
        <dataChegada>2023-10-17 01:24:56</dataChegada>
      </return>
    </ns0:obterEventosTempoDirecaoDataChegadaResponse>
  </S:Body>
</S:Envelope>
```

7.3. ObterDeltaTelemetriaIntegracaoInercia



Es necesario contar con Obligación Financiera de Software para el consumo de los Deltas.

Descripción

Servicio responsable de poner a disposición de los clientes los Deltas de Telemetría, respetando las reglas de negocio previamente establecidas al INTEGRADOR.

Declaración del Método

```
List obterDeltaTelemetriaIntegracaoInercia(String usuario, String senha, Integer idVeiculo, String dataInicio, String dataFinal)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idVeiculo	Integer	Sí	Id del vehículo
dataInicio	String	Sí	Fecha inicial para la consulta Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.
dataFinal	String	Sí	Fecha final para la consulta Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.

Retorno

Retorna una lista de objetos del tipo Delta con las propiedades a continuación:

Nombre	Tipo	Descripción
idVeiculo	Integer	Id del vehículo
idMotorista	Integer	Id del conductor
loginMotorista	Integer	Login del conductor
nomeMotorista	String	Nombre Conductor
latitude	Integer	Latitud
longitude	Integer	Longitud
dataPosicao	String	Fecha del GPS (GMT-0) en el momento del envío del paquete

Nome	Tipo	Descrição
tempoDuracaoGiroMotor	Integer	Tiempo en segundos(s) que hubo giro de motor
tempoDuracaoTotal	Integer	Tiempo en segundos(s) transcurrido desde el inicio hasta el cierre del delta
tempoDuracaoMovimento	Integer	Tiempo en segundos(s) que el vehículo estuvo en movimiento
tempoDuracaoParado	Integer	Tiempo en segundos(s) que el vehículo estuvo parado
velocidadeMaximaFaixaAmarela	Integer	Velocidad máxima en la que el vehículo permaneció en la franja amarilla
tipoDelta	Integer	Tipo de Delta: – Ignición ON/OFF – Login/ Logoff
tempoDuracaoFaixaMarchaLenta	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de marcha lenta
tempoDuracaoFaixaMarchaLentaComVelocidade	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de marcha lenta estando en velocidad.
tempoDuracaoFaixaDeTransicaoComInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de transición con inercia.
tempoDuracaoFaixaDeTransicaoSemInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de transición sin inercia.
tempoDuracaoFaixaVerdeEconomicaComInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja verde económica con inercia.
tempoDuracaoFaixaVerdeEconomicaSemInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja verde económica sin inercia.
tempoDuracaoFaixaVerdeComInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja verde con inercia.
tempoDuracaoFaixaVerdeSemInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja verde sin inercia.
tempoDuracaoFaixaAmarerlaSemInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja amarilla con inercia.
tempoDuracaoFaixaAmarelaComInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja amarilla sin inercia.
tempoDuracaoFaixaDePerigoComInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de peligro con inercia.
tempoDuracaoFaixaDePerigoSemInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de peligro sin inercia.
horimetro	Integer	Horímetro final (en el momento del cierre del delta)
odometro	Integer	Hodómetro final (en el momento del cierre del delta)
distanciaPercorrida	Integer	Distancia recorrida (desde la apertura hasta el cierre del delta)
velocidadeMedia	Integer	Velocidad media

Nombre	Tipo	Descripción
rpmMaximo	Integer	RPM máximo alcanzado entre la apertura y el cierre del delta
rpmMedia	Integer	Promedio de RPM (entre la apertura y el cierre del delta)
tempoDuracaoFreioMotor	Integer	Tiempo de accionamiento (en segundos) del freno motor
distanciaPercorridaEmbreagemAccionada	Integer	Distancia (en km) que el vehículo recorrió con embrague accionado
distanciaPercorridaFreioAccionado	Integer	Distancia (en km) que el vehículo recorrió con freno accionado
consumoCombustivel	Integer	Consumo (en mililitros) de combustible desde el inicio hasta el final del delta. En los casos en que el Delta de Telemetría no tiene la información de Consumo de Combustible, este campo no se presentará en el XML de Retorno.
distPercorridaAscendenteFxAmarela	Integer	Distancia (en km) recorrida por el vehículo en ascenso en la Franja Amarilla
distPercorridaAscendenteFxMarchaLenta	Integer	Distancia (en km) recorrida por el vehículo en ascenso en Marcha Lenta
distPercorridaAscendenteFxPerigo	Integer	Distancia (en km) recorrida por el vehículo en ascenso en la Franja de Peligro
distPercorridaAscendenteFxTransic	Integer	Distancia (en km) recorrida por el vehículo en ascenso en la Franja de Transición
distPercorridaAscendenteFxVerde	Integer	Distancia (en km) recorrida por el vehículo en ascenso en la Franja Verde
distPercorridaAscendenteFxVerde_ext	Integer	Distancia (en km) recorrida por el vehículo en ascenso en la Franja Verde Económica
distPercorridaDescendenteFxAmarela	Integer	Distancia (en km) recorrida por el vehículo en descenso en la Franja Amarilla
distPercorridaDescendenteFxMarchaLenta	Integer	Distancia (en km) recorrida por el vehículo en descenso en Marcha Lenta
distPercorridaDescendenteFxPerigo	Integer	Distancia (en km) recorrida por el vehículo en descenso en la Franja de Peligro
distPercorridaDescendenteFxTransic	Integer	Distancia (en km) recorrida por el vehículo en descenso en la Franja de Transición
distPercorridaDescendenteFxVerde	Integer	Distancia (en km) recorrida por el vehículo en descenso en la Franja Verde
distPercorridaDescendenteFxVerde_ext	Integer	Distancia (en km) recorrida por el vehículo en descenso en la Franja Verde Económica
distPercorridaEstavelFxAmarela	Integer	Distancia (en km) recorrida por el vehículo estable en la Franja Amarilla

Nombre	Tipo	Descripción
distPercorridaEstavelFxMarcha Lenta	Integer	Distancia (en km) recorrida por el vehículo estable en Marcha Lenta
distPercorridaEstavelFxPerigo	Integer	Distancia (en km) recorrida por el vehículo estable en la Franja de Peligro
distPercorridaEstavelFxTransi c	Integer	Distancia (en km) recorrida por el vehículo estable en la Franja de Transición
distPercorridaEstavelFxVerde	Integer	Distancia (en km) recorrida por el vehículo estable en la Franja Verde
distPercorridaEstavelFxVerde_ext	Integer	Distancia (en km) recorrida por el vehículo estable en la Franja Verde Económica
distTotalFxAscendente	Integer	Distancia Total (en km) recorrida por el vehículo en ascenso
distTotalFxDescendente	Integer	Distancia Total (en km) recorrida por el vehículo en descenso
distTotalFxEstavel	Integer	Distancia Total (en km) recorrida por el vehículo estable
tempoDuracaoAscendenteFxAmare la	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en la Franja Amarilla
tempoDuracaoAscendenteFxMarch aLenta	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en Marcha Lenta
tempoDuracaoAscendenteFxPerig o	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en la Franja de Peligro
tempoDuracaoAscendenteFxTransi c	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en la Franja de Transición
tempoDuracaoAscendenteFxVerde	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en la Franja Verde
tempoDuracaoAscendenteFxVerde _ext	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en la Franja Verde Económica
tempoDuracaoDescendenteFxAmar ela	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en la Franja Amarilla
tempoDuracaoDescendenteFxMarc haLenta	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en Marcha Lenta
tempoDuracaoDescendenteFxPeri go	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en la Franja de Peligro
tempoDuracaoDescendenteFxTran sic	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en la Franja de Transición
tempoDuracaoDescendenteFxVerd e	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en la Franja Verde
tempoDuracaoDescendenteFxVerd e_ext	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en la Franja Verde Económica

Nome	Tipo	Descrição
tempoDuracaoEstavelFxAmarela	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en la Franja Amarilla
tempoDuracaoEstavelFxMarchaLenta	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en Marcha Lenta
tempoDuracaoEstavelFxPerigo	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en la Franja de Peligro
tempoDuracaoEstavelFxTransic	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en la Franja de Transición
tempoDuracaoEstavelFxVerde	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en la Franja Verde
tempoDuracaoEstavelFxVerde_ext	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en la Franja Verde Económica
tempoTotalFxAscendente	Integer	Tiempo Total en segundos(s) que el vehículo permaneció en ascenso
tempoTotalFxDescendente	Integer	Tiempo Total en segundos(s) que el vehículo permaneció en descenso
tempoTotalFxEstavel	Integer	Tiempo Total en segundos(s) que el vehículo permaneció estable
pressaoMediaDoOleoDoMotor	Integer	Presión media del aceite
pressaoMediaDoSistemaDeFreioAr	Integer	Presión Media del Sistema de Frenos de Aire
temperaturaMediaDoArrefecimento	Integer	Temperatura Media de Enfriamiento
temperaturaMediaDoCombustivel	Integer	Temperatura Media del Combustible
temperaturaMediaDoOleoDoMotor	Integer	Temperatura Media del Aceite
tempoTotalComCinto	Integer	Tiempo Total con Cinturón
tensaoMediaDaBateria	Integer	Tensión Media de la Batería

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterDeltaTelemetriaIntegracaoInercia>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <dataInicio>2023-10-17 00:00:00</dataInicio>
      <dataFinal>2023-10-17 23:59:00</dataFinal>
      <idVeiculo>881456</idVeiculo>
    </web:obterDeltaTelemetriaIntegracaoInercia>
  </soapenv:Body>
</soapenv:Envelope>
```

```

    </web:obterDeltaTelemetriaIntegracaoInercia>
  </soapenv:Body>
</soapenv:Envelope>

```

Response:

```

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterDeltaTelemetriaIntegracaoInerciaResponse
      xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <consumoCombustivel>0</consumoCombustivel>
        <dataPosicao>17/10/2023 07:11:46</dataPosicao>
        <distPercorridaAscendenteFxAmarela>0</distPercorridaAscendenteFxAmarela>
        <distPercorridaAscendenteFxFxMarchaLenta>0</distPercorridaAscendenteFxFxMarchaLenta>
        <distPercorridaAscendenteFxFxPerigo>0</distPercorridaAscendenteFxFxPerigo>
        <distPercorridaAscendenteFxFxTransic>0</distPercorridaAscendenteFxFxTransic>
        <distPercorridaAscendenteFxFxVerde>0</distPercorridaAscendenteFxFxVerde>
        <distPercorridaAscendenteFxFxVerde_ext>0</distPercorridaAscendenteFxFxVerde_ext>
        <distPercorridaDescendenteFxAmarela>0</distPercorridaDescendenteFxAmarela>
        <distPercorridaDescendenteFxFxMarchaLenta>0</distPercorridaDescendenteFxFxMarchaLenta>
        <distPercorridaDescendenteFxFxPerigo>0</distPercorridaDescendenteFxFxPerigo>
        <distPercorridaDescendenteFxFxTransic>0</distPercorridaDescendenteFxFxTransic>
        <distPercorridaDescendenteFxFxVerde>0</distPercorridaDescendenteFxFxVerde>
        <distPercorridaDescendenteFxFxVerde_ext>0</distPercorridaDescendenteFxFxVerde_ext>
        <distPercorridaEstavelFxAmarela>0</distPercorridaEstavelFxAmarela>
        <distPercorridaEstavelFxFxMarchaLenta>0</distPercorridaEstavelFxFxMarchaLenta>
        <distPercorridaEstavelFxFxPerigo>0</distPercorridaEstavelFxFxPerigo>
        <distPercorridaEstavelFxFxTransic>0</distPercorridaEstavelFxFxTransic>
        <distPercorridaEstavelFxFxVerde>0</distPercorridaEstavelFxFxVerde>
        <distPercorridaEstavelFxFxVerde_ext>0</distPercorridaEstavelFxFxVerde_ext>
        <distTotalFxAscendente>0</distTotalFxAscendente>
        <distTotalFxDescendente>0</distTotalFxDescendente>
        <distTotalFxEstavel>0</distTotalFxEstavel>
        <distanciaPercorrida>0</distanciaPercorrida>
        <distanciaPercorridaEmbreagemAccionada>0</distanciaPercorridaEmbreagemAccionada>
        <distanciaPercorridaFreioAccionado>0</distanciaPercorridaFreioAccionado>
        <horimetro>535443</horimetro>
        <idMotorista>0</idMotorista>
        <idVeiculo>881456</idVeiculo>
        <latitude>-22.7543</latitude>
        <loginMotorista>0</loginMotorista>
        <longitude>-43.3877</longitude>
        <nomeMotorista>Sem motorista</nomeMotorista>
        <odometro>4294967</odometro>
        <pressaoMediaDoOleoDoMotor>0</pressaoMediaDoOleoDoMotor>
        <pressaoMediaDoSistemaDeFreioAAR>0</pressaoMediaDoSistemaDeFreioAAR>
        <rpmMaximo>0</rpmMaximo>
        <rpmMedia>0</rpmMedia>
        <temperaturaMediaDoArrefecimento>0</temperaturaMediaDoArrefecimento>
        <temperaturaMediaDoCombustivel>0</temperaturaMediaDoCombustivel>
        <temperaturaMediaDoOleoDoMotor>0</temperaturaMediaDoOleoDoMotor>
        <tempoDuracaoAscendenteFxAmarela>0</tempoDuracaoAscendenteFxAmarela>
        <tempoDuracaoAscendenteFxFxMarchaLenta>0</tempoDuracaoAscendenteFxFxMarchaLenta>
        <tempoDuracaoAscendenteFxFxPerigo>0</tempoDuracaoAscendenteFxFxPerigo>
        <tempoDuracaoAscendenteFxFxTransic>0</tempoDuracaoAscendenteFxFxTransic>
        <tempoDuracaoAscendenteFxFxVerde>0</tempoDuracaoAscendenteFxFxVerde>
        <tempoDuracaoAscendenteFxFxVerde_ext>0</tempoDuracaoAscendenteFxFxVerde_ext>
        <tempoDuracaoDescendenteFxAmarela>0</tempoDuracaoDescendenteFxAmarela>
        <tempoDuracaoDescendenteFxFxMarchaLenta>0</tempoDuracaoDescendenteFxFxMarchaLenta>
        <tempoDuracaoDescendenteFxFxPerigo>0</tempoDuracaoDescendenteFxFxPerigo>
        <tempoDuracaoDescendenteFxFxTransic>0</tempoDuracaoDescendenteFxFxTransic>
      </return>
    </ns0:obterDeltaTelemetriaIntegracaoInerciaResponse>
  </S:Body>
</S:Envelope>

```

```

    <tempoDuracaoDescendenteFxVerde>0</tempoDuracaoDescendenteFxVerde>
    <tempoDuracaoDescendenteFxVerde_ext>0</tempoDuracaoDescendenteFxVerde_ext>
    <tempoDuracaoEstavelFxAmarela>0</tempoDuracaoEstavelFxAmarela>
    <tempoDuracaoEstavelFxMarchaLenta>0</tempoDuracaoEstavelFxMarchaLenta>
    <tempoDuracaoEstavelFxPerigo>0</tempoDuracaoEstavelFxPerigo>
    <tempoDuracaoEstavelFxTransic>0</tempoDuracaoEstavelFxTransic>
    <tempoDuracaoEstavelFxVerde>0</tempoDuracaoEstavelFxVerde>
    <tempoDuracaoEstavelFxVerde_ext>0</tempoDuracaoEstavelFxVerde_ext>
    <tempoDuracaoFaixaAmarelaComInercia>0</tempoDuracaoFaixaAmarelaComInercia>
    <tempoDuracaoFaixaAmarerlaSemInercia>0</tempoDuracaoFaixaAmarerlaSemInercia>
    <tempoDuracaoFaixaDePerigoComInercia>0</tempoDuracaoFaixaDePerigoComInercia>
    <tempoDuracaoFaixaDePerigoSemInercia>0</tempoDuracaoFaixaDePerigoSemInercia>
    <tempoDuracaoFaixaDeTransicaoComInercia>0</tempoDuracaoFaixaDeTransicaoComInercia>
    <tempoDuracaoFaixaDeTransicaoSemInercia>0</tempoDuracaoFaixaDeTransicaoSemInercia>
    <tempoDuracaoFaixaMarchaLenta>0</tempoDuracaoFaixaMarchaLenta>
    <tempoDuracaoFaixaMarchaLentaComVelocidade>
0</tempoDuracaoFaixaMarchaLentaComVelocidade>
    <tempoDuracaoFaixaVerdeComInercia>0</tempoDuracaoFaixaVerdeComInercia>
    <tempoDuracaoFaixaVerdeEconomicaComInercia>
0</tempoDuracaoFaixaVerdeEconomicaComInercia>
    <tempoDuracaoFaixaVerdeEconomicaSemInercia>
0</tempoDuracaoFaixaVerdeEconomicaSemInercia>
    <tempoDuracaoFaixaVerdeSemInercia>0</tempoDuracaoFaixaVerdeSemInercia>
    <tempoDuracaoFreioMotor>0</tempoDuracaoFreioMotor>
    <tempoDuracaoGiroMotor>0</tempoDuracaoGiroMotor>
    <tempoDuracaoMovimento>0</tempoDuracaoMovimento>
    <tempoDuracaoParado>0</tempoDuracaoParado>
    <tempoDuracaoTotal>0</tempoDuracaoTotal>
    <tempoTotalComCinto>0</tempoTotalComCinto>
    <tempoTotalFxAscendente>0</tempoTotalFxAscendente>
    <tempoTotalFxDescendente>0</tempoTotalFxDescendente>
    <tempoTotalFxEstavel>0</tempoTotalFxEstavel>
    <tensaoMediaDaBateria>0</tensaoMediaDaBateria>
    <tipoDelta>254</tipoDelta>
    <velocidadeMaximaFaixaAmarela>0</velocidadeMaximaFaixaAmarela>
    <velocidadeMedia>0</velocidadeMedia>
  </return>
</ns0:obterDeltaTelemetriaIntegracaoInerciaResponse>
</S:Body>
</S:Envelope>

```

7.4. ObterDeltaTelemetriaIntegracaoDataChegada



Es necesario contar con Obligación Financiera de Software para el consumo de los Deltas.

Descripción

Servicio responsable de poner a disposición de los clientes los Deltas de Telemetría, respetando las reglas de negocio previamente establecidas al INTEGRADOR.

Declaración del Método

```
List obterDeltaTelemetriaIntegracaoDataChegada(String usuario, String senha, String dataInicio, String dataFinal, Integer idVeiculo, String dataInicioChegada, String dataFinalChegada)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
dataInicio	String	No	Fecha inicial para la consulta Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.
dataFinal	String	No	Fecha final para la consulta Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.
idVeiculo	Integer	Sí	Id del Vehículo
dataInicioChegada	String	Sí	Fecha de Inicio de registro en el sistema Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.
dataFinalChegada	String	Sí	Fecha final de registro en el sistema Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.

Retorno

Nombre	Tipo	Descripción
idVeiculo	Integer	Id del vehículo
idMotorista	Integer	Id del conductor
loginMotorista	Integer	Login del conductor
latitude	Integer	Latitud

Nome	Tipo	Descrição
longitude	Integer	Longitud
dataPosicao	String	Fecha del GPS (GMT-0) en el momento del envío del paquete
deltmDeltaGiroMotor	Integer	Tiempo en segundos(s) que hubo giro de motor
deltmDeltaIntervalo	Integer	Tiempo en segundos(s) transcurrido desde el inicio hasta el cierre del delta
tempoDuracaoMovimento	Integer	Tiempo en segundos(s) que el vehículo estuvo en movimiento
deltmDeltaParado	Integer	Tiempo en segundos(s) que el vehículo estuvo parado
tempoDuracaoFaixaAmarela	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja amarilla
deltmFxAmareloVelMax	Integer	Velocidad máxima en la que el vehículo permaneció en la franja amarilla
tempoDuracaoFaixaAzul	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja azul
tempoDuracaoFaixaVerde	Integer	Tiempo en segundos(s) que el vehículo permaneció en la verde
tempoDuracaoFaixaVermelha	Integer	Tiempo en segundos(s) que el vehículo permaneció en la roja
tipoDelta	Integer	Tipo de Delta: – Ignición ON/OFF – Login/ Logoff
horimetro	Integer	Horímetro final (en el momento del cierre del delta)
odometro	Integer	Hodómetro final (en el momento del cierre del delta)
deltmHodometroDelta	Integer	Distancia recorrida (desde la apertura hasta el cierre del delta)
velocidadeMedia	Integer	Velocidad media
rpmMaximo	Integer	RPM máximo alcanzado entre la apertura y el cierre del delta
rpmMedia	Integer	Promedio de RPM (entre la apertura y el cierre del delta)
tempoAccionamentoFreioMotor	Integer	Tiempo de accionamiento (en segundos) del freno motor
distanciaPercorridaEmbreagemAccionada	Integer	Distancia (en km) que el vehículo recorrió con embrague accionado
distanciaPercorridaFreioAccionado	Integer	Distancia (en km) que el vehículo recorrió con freno accionado
consumoCombustivel	Integer	Consumo (en mililitros) de combustible desde el inicio hasta el final del delta. En los casos en que el Delta de Telemetría no tiene la información de Consumo de Combustible, este campo no se presentará en el XML de Retorno.
dataChegada	String	Fecha de llegada del paquete.
nomeMotorista	String	Nombre del Conductor

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterDeltaTelemetriaIntegracaoDataChegada>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <dataInicio>2023-10-17 00:00:00</dataInicio>
      <dataFinal>2023-10-17 23:59:00</dataFinal>
      <idVeiculo>881456</idVeiculo>
      <dataInicioChegada>2023-10-17 00:00:00</dataInicioChegada>
      <dataFinalChegada>2023-10-17 00:00:00</dataFinalChegada>
    </web:obterDeltaTelemetriaIntegracaoDataChegada>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterDeltaTelemetriaIntegracaoDataChegadaResponse
      xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <consumoCombustivel>0</consumoCombustivel>
        <dataPosicao>17/10/2023 07:11:46</dataPosicao>
        <distanciaPercorrida>0</distanciaPercorrida>
        <distanciaPercorridaEmbreagemAccionada>0</distanciaPercorridaEmbreagemAccionada>
        <distanciaPercorridaFreioAccionado>0</distanciaPercorridaFreioAccionado>
        <horimetro>535443</horimetro>
        <idMotorista>0</idMotorista>
        <idVeiculo>881456</idVeiculo>
        <latitude>-22.7543</latitude>
        <loginMotorista>0</loginMotorista>
        <longitude>-43.3877</longitude>
        <nomeMotorista>Sem motorista</nomeMotorista>
        <odometro>4294967</odometro>
        <rpmMaximo>0</rpmMaximo>
        <rpmMedia>0</rpmMedia>
        <tempoDuracaoFaixaAmarela>0</tempoDuracaoFaixaAmarela>
        <tempoDuracaoFaixaAzul>0</tempoDuracaoFaixaAzul>
        <tempoDuracaoFaixaVerde>0</tempoDuracaoFaixaVerde>
        <tempoDuracaoFaixaVermelha>0</tempoDuracaoFaixaVermelha>
        <tempoDuracaoFreioMotor>0</tempoDuracaoFreioMotor>
        <tempoDuracaoGiroMotor>0</tempoDuracaoGiroMotor>
        <tempoDuracaoMovimento>0</tempoDuracaoMovimento>
        <tempoDuracaoParado>0</tempoDuracaoParado>
        <tempoDuracaoTotal>0</tempoDuracaoTotal>
        <tipoDelta>254</tipoDelta>
        <velocidadeMaximaFaixaAmarela>0</velocidadeMaximaFaixaAmarela>
        <velocidadeMedia>0</velocidadeMedia>
        <dataChegada>17/10/2023 07:11:46</dataChegada>
      </return>
    </ns0:obterDeltaTelemetriaIntegracaoDataChegadaResponse>
  </S:Body>
</S:Envelope>
```

7.5. ObterDeltaTelemetriaIntegracaoInerciaDataChegada



Es necesario contar con Obligación Financiera de Software para el consumo de los Deltas.

Descripción

Servicio responsable de poner a disposición de los clientes los Deltas de Telemetría, respetando las reglas de negocio previamente establecidas al INTEGRADOR

Declaración del Método

```
List obterDeltaTelemetriaIntegracaoInerciaDataChegada(String usuario, String senha, Integer idVeiculo, String dataInicio, String dataFinal, String dataChegadaInicio, String dataChegadaFinal)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idVeiculo	Integer	Sí	Id del vehículo
dataInicio	String	Sí	Fecha inicial para la consulta Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.
dataFinal	String	Sí	Fecha final para la consulta Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.
dataChegadaInicio	String	Sí	Fecha de Inicio de registro en el sistema Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.
dataChegadaFinal	String	Sí	Fecha final de registro en el sistema Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.

Retorno

Nombre	Tipo	Descripción
deltmIdVeiculo	Integer	Id del vehículo
deltmIdMotorista	Integer	Id del conductor
deltmloginMotorista	Integer	Login del conductor
deltmLatitude	Integer	Latitud

Nome	Tipo	Descrição
deltmLongitude	Integer	Longitud
deltmDataPosicao	String	Fecha del GPS (GMT-0) en el momento del envío del paquete
deltmDeltaGiroMotor	Integer	Tiempo en segundos(s) que hubo giro de motor
deltmDeltaIntervalo	Integer	Tiempo en segundos(s) transcurrido desde el inicio hasta el cierre del delta
deltmDeltaMovimento	Integer	Tiempo en segundos(s) que el vehículo estuvo en movimiento
deltmDeltaParado	Integer	Tiempo en segundos(s) que el vehículo estuvo parado
deltmFxAmareloVelMax	Integer	Velocidad máxima en la que el vehículo permaneció en la franja amarilla
deltmFxVermelhoTipo	Integer	Tipo de Delta: – Ignición ON/OFF – Login/ Logoff
tempoDuracaoFaixaMachaLenta	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de marcha lenta
tempoDuracaoFaixaMachaLentaComVelocidade	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de marcha lenta estando en velocidad.
tempoDuracaoFaixaDeTransicaoComInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de transición con inercia.
tempoDuracaoFaixaDeTransicaoSemInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de transición sin inercia.
tempoDuracaoFaixaVerdeEconomicaComInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja verde económica con inercia.
tempoDuracaoFaixaVerdeEconomicaSemInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja verde económica sin inercia.
tempoDuracaoFaixaVerdeComInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja verde con inercia.
tempoDuracaoFaixaVerdeSemInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja verde sin inercia.
tempoDuracaoFaixaAmarelaSemInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja amarilla con inercia.
tempoDuracaoFaixaAmarelaComInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja amarilla sin inercia.
tempoDuracaoFaixaDePerigoComInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de peligro con inercia.
tempoDuracaoFaixaDePerigoSemInercia	Integer	Tiempo en segundos(s) que el vehículo permaneció en la franja de peligro sin inercia.
deltmHorimetro	Integer	Horímetro final (en el momento del cierre del delta)
deltmHodometro	Integer	Hodómetro final (en el momento del cierre del delta)

Nombre	Tipo	Descripción
deltmHodometroDelta	Integer	Distancia recorrida (desde la apertura hasta el cierre del delta)
deltmHodometroVelocidadeMedia	Integer	Velocidad media
deltmRpmMax	Integer	RPM máximo alcanzado entre la apertura y el cierre del delta
deltmRpmMed	Integer	Promedio de RPM (entre la apertura y el cierre del delta)
deltmTempoFreioMotor	Integer	Tiempo de accionamiento (en segundos) del freno motor
deltmUsoEmbreagemKm	Integer	Distancia (en km) que el vehículo recorrió con embrague accionado
deltmUsoFreioKm	Integer	Distancia (en km) que el vehículo recorrió con freno accionado
delconsumo_combustivel	Integer	Consumo (en mililitros) de combustible desde el inicio hasta el final del delta. En los casos en que el Delta de Telemetría no tiene la información de Consumo de Combustible, este campo no se presentará en el XML de Retorno.
distPercorridaAscendenteFxAmarela	Integer	Distancia (en km) recorrida por el vehículo en ascenso en la Franja Amarilla
distPercorridaAscendenteFxMarchaLenta	Integer	Distancia (en km) recorrida por el vehículo en ascenso en Marcha Lenta
distPercorridaAscendenteFxPerigo	Integer	Distancia (en km) recorrida por el vehículo en ascenso en la Franja de Peligro
distPercorridaAscendenteFxTransic	Integer	Distancia (en km) recorrida por el vehículo en ascenso en la Franja de Transición
distPercorridaAscendenteFxVerde	Integer	Distancia (en km) recorrida por el vehículo en ascenso en la Franja Verde
distPercorridaAscendenteFxVerde_ext	Integer	Distancia (en km) recorrida por el vehículo en ascenso en la Franja Verde Económica
distPercorridaDescendenteFxAmarela	Integer	Distancia (en km) recorrida por el vehículo en descenso en la Franja Amarilla
distPercorridaDescendenteFxMarchaLenta	Integer	Distancia (en km) recorrida por el vehículo en descenso en Marcha Lenta
distPercorridaDescendenteFxPerigo	Integer	Distancia (en km) recorrida por el vehículo en descenso en la Franja de Peligro
distPercorridaDescendenteFxTransic	Integer	Distancia (en km) recorrida por el vehículo en descenso en la Franja de Transición
distPercorridaDescendenteFxVerde	Integer	Distancia (en km) recorrida por el vehículo en descenso en la Franja Verde
distPercorridaDescendenteFxVerde_ext	Integer	Distancia (en km) recorrida por el vehículo en descenso en la Franja Verde Económica
distPercorridaEstavelFxAmarela	Integer	Distancia (en km) recorrida por el vehículo estable en la Franja Amarilla

Nombre	Tipo	Descripción
distPercorridaEstavelFxMarcha Lenta	Integer	Distancia (en km) recorrida por el vehículo estable en Marcha Lenta
distPercorridaEstavelFxPerigo	Integer	Distancia (en km) recorrida por el vehículo estable en la Franja de Peligro
distPercorridaEstavelFxTransi c	Integer	Distancia (en km) recorrida por el vehículo estable en la Franja de Transición
distPercorridaEstavelFxVerde	Integer	Distancia (en km) recorrida por el vehículo estable en la Franja Verde
distPercorridaEstavelFxVerde_ ext	Integer	Distancia (en km) recorrida por el vehículo estable en la Franja Verde Económica
distTotalFxAscendente	Integer	Distancia Total (en km) recorrida por el vehículo en ascenso
distTotalFxDescendente	Integer	Distancia Total (en km) recorrida por el vehículo en descenso
distTotalFxEstavel	Integer	Distancia Total (en km) recorrida por el vehículo estable
tempoDuracaoAscendenteFxAmare la	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en la Franja Amarilla
tempoDuracaoAscendenteFxMarch aLenta	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en Marcha Lenta
tempoDuracaoAscendenteFxPerig o	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en la Franja de Peligro
tempoDuracaoAscendenteFxTrans ic	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en la Franja de Transición
tempoDuracaoAscendenteFxVerde	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en la Franja Verde
tempoDuracaoAscendenteFxVerde_ _ext	Integer	Tiempo en segundos(s) que el vehículo permaneció en ascenso en la Franja Verde Económica
tempoDuracaoDescendenteFxAmar ela	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en la Franja Amarilla
tempoDuracaoDescendenteFxMarc haLenta	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en Marcha Lenta
tempoDuracaoDescendenteFxPeri go	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en la Franja de Peligro
tempoDuracaoDescendenteFxTran sic	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en la Franja de Transición
tempoDuracaoDescendenteFxVerd e	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en la Franja Verde
tempoDuracaoDescendenteFxVerd e_ext	Integer	Tiempo en segundos(s) que el vehículo permaneció en descenso en la Franja Verde Económica

Nome	Tipo	Descrição
tempoDuracaoEstavelFxAmarela	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en la Franja Amarilla
tempoDuracaoEstavelFxMarchaLenta	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en Marcha Lenta
tempoDuracaoEstavelFxPerigo	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en la Franja de Peligro
tempoDuracaoEstavelFxTransic	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en la Franja de Transición
tempoDuracaoEstavelFxVerde	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en la Franja Verde
tempoDuracaoEstavelFxVerde_ext	Integer	Tiempo en segundos(s) que el vehículo permaneció estable en la Franja Verde Económica
tempoTotalFxAscendente	Integer	Tiempo Total en segundos(s) que el vehículo permaneció en ascenso
tempoTotalFxDescendente	Integer	Tiempo Total en segundos(s) que el vehículo permaneció en descenso
tempoTotalFxEstavel	Integer	Tiempo Total en segundos(s) que el vehículo permaneció estable
pressaoMediaDoOleoDoMotor	Integer	Presión media del aceite
pressaoMediaDoSistemaDeFreioAr	Integer	Presión Media del Sistema de Frenos de Aire
temperaturaMediaDoArrefecimento	Integer	Temperatura Media de Enfriamiento
temperaturaMediaDoCombustivel	Integer	Temperatura Media del Combustible
temperaturaMediaDoOleoDoMotor	Integer	Temperatura Media del Aceite
tempoTotalComCinto	Integer	Tiempo Total con Cinturón
tensaoMediaDaBateria	Integer	Tensión Media de la Batería
dataChegada	String	Fecha de Llegada del paquete

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterDeltaTelemetriaIntegracaoInerciaDataChegada>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <dataInicio>2023-10-17 00:00:00</dataInicio>
    </web:obterDeltaTelemetriaIntegracaoInerciaDataChegada>
    </soapenv:Body>
  </soapenv:Envelope>
```

```
<dataFinal>2023-10-17 23:59:00</dataFinal>
<dataChegadaInicio>2023-10-17 00:00:00</dataChegadaInicio>
<dataChegadaFinal>2023-10-17 00:00:00</dataChegadaFinal>
<idVeiculo>881456</idVeiculo>
</web:obterDeltaTelemetriaIntegracaoInerciaDataChegada>
</soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <!-- Resposta truncada ou simplificada -->
  </S:Body>
</S:Envelope>
```

7.6. ObterEventoTelemetriaIntegracaoDataChegada



Es necesario contar con Obligación Financiera de Software para el consumo de los Eventos.

Descripción

Servicio responsable de poner a disposición de los clientes los Eventos de Telemetría, respetando las reglas de negocio previamente establecidas al INTEGRADOR.

Declaración del Método

```
List obterEventoTelemetriaIntegracaoDataChegada(String usuario, String senha, Integer idVeiculo, Integer idEventoList, String dataInicio, String dataFinal, String dataChegadaInicio, String dataChegadaFinal)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
idVeiculo	Integer	Sí	Id del vehículo
idEventoList	Integer	No	Id del/de los evento(s) Ej: 1 2
dataInicio	String	No	Fecha inicial para la consulta Estándar: AAAA-MM-DDTHH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.
dataFinal	String	No	Fecha final para la consulta Estándar: AAAA-MM-DDTHH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.
dataChegadaInicio	String	Sí	Fecha de Inicio de registro en el sistema Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.
dataChegadaFinal	String	Sí	Fecha final de registro en el sistema Estándar: AAAA-MM-DD HH:MM:SS Regla: El período máximo de extracción de datos debe ser 1 (un) día.

Retorno

Nombre	Tipo	Descripción
evetmIdVeiculo	Integer	Id del vehículo
evetmIdMotorista	Integer	Id del conductor

Nombre	Tipo	Descripción
evetmloginMotorista	Integer	Login del conductor
evetmLatitude	Integer	Latitud
evetmLongitude	Integer	Longitud
evetmDataPacote	String	Fecha del paquete del Evento
evetmIdTelemetria	Integer	Id del evento telemetría: • 210 — Sistema de Freno de Aire • 211 — Cinturón • 214 — Fuerza G Lateral Débil • 215 — Fuerza G Lateral Media - Fuerza G Lateral Fuerte - Tiempo parado - Presión de aceite - Tensión del alternador - Temperatura del líquido de enfriamiento - Exceso de rotación - Exceso de velocidad tramo vial con lluvia - Exceso de velocidad tramo vial seco - Exceso de velocidad tramo urbano con lluvia - Exceso de velocidad tramo urbano seco - Tiempo de marcha lenta excesivo - Tiempo largo de ignición sin giro de motor - Pie en el embrague - Arrancada brusca - Frenada brusca - Banguela - Arrancada en segunda marcha - Eventos de recorrido - Entradas y salidas - Horímetro vehículo en movimiento/ hodómetro - Horímetro motor encendido/ vehículo parado - Horímetro franja verde/roja - Horímetro franja azul/amarilla
evetmHodometro	Integer	Hodómetro en el momento del Evento.
evetmHorimetro	Integer	Horímetro en el momento del Evento.
evetmTempoDuracao	Integer	Tiempo de Duración del Evento.
velocidadeMaximaEvento	Integer	Velocidad máxima alcanzada durante el evento.
velocidadeReferencia	Integer	Velocidad de límite registrada en el evento.
dataChegada	String	Fecha de Llegada del Evento

Ejemplo XML

Request:

```
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/"
>
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterEventoTelemetriaIntegracaoDataChegada>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <dataInicio>2023-10-17 00:00:00</dataInicio>
```

```
<dataFinal>2023-10-17 23:59:00</dataFinal>
<dataChegadaInicio>2023-10-17 00:00:00</dataChegadaInicio>
<dataChegadaFinal>2023-10-17 00:00:00</dataChegadaFinal>
<idVeiculo>881456</idVeiculo>
<!--Zero or more repetitions:-->
<idEventoList>?</idEventoList>
</web:obterEventoTelemetriaIntegracaoDataChegada>
</soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterEventoTelemetriaIntegracaoDataChegadaResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <dataPosicao>17/10/2023 05:43:03</dataPosicao>
        <horimetro>217006</horimetro>
        <idEvento>210</idEvento>
        <idMotorista>0</idMotorista>
        <idVeiculo>1207216</idVeiculo>
        <latitude>-26.2539</latitude>
        <loginMotorista>0</loginMotorista>
        <longitude>-48.61469</longitude>
        <nomeMotorista>Sem motorista</nomeMotorista>
        <odometro>484878</odometro>
        <tempoDuracao>0</tempoDuracao>
        <valorConfigurado>600</valorConfigurado>
        <valorVioladoFreioAr>0</valorVioladoFreioAr>
        <velocidadeMaximaEvento>0</velocidadeMaximaEvento>
        <velocidadeReferencia>0</velocidadeReferencia>
        <dataChegada>17/10/2023 05:43:04</dataChegada>
      </return>
    </ns0:obterEventoTelemetriaIntegracaoDataChegadaResponse>
  </S:Body>
</S:Envelope>
```

7.7. getSmartCamerasEvents

Descripción

Método para consultar los eventos de cámaras inteligentes de los vehículos, para visualización de la gestora.



Regla de restricción

Para la obtención de fechas, el método usa el timestamp con UTC 0 (cero).

Declaración del Método

```
List getSmartCamerasEvents(String agrupador)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
agrupador	String	Sí	Usuario que visualizará la información de las empresas que concedieron acceso a los vehículos
senha	String	Sí	Contraseña del integrador para acceso al sistema.
offset	Integer	Sí	Define la paginación de los resultados
limit	Integer	Sí	Usado para recurso de paginación de los resultados
motoristas	String	No	Lista los ids de conductores en el formato "1,2,3,4,5" (sin comillas)
veículos	String	No	Lista de patentes a filtrar en el formato "ABC123,CDE456,FGH789" (sin comillas)
dataInicio	String	Sí	Fecha de inicio del filtro en el formato DD-MM-YYYY
dataFim	String	Sí	Fecha de fin del filtro en el formato DD-MM-YYYY
tipoEvento	Integer	No	Evento presentado por el vehículo
criticidade	Integer	No	Lista de códigos de criticidad a filtrar en el formato "1,2,3" (sin comillas), los códigos deben ser los mismos utilizados en el sistema SMART CAMERAS
turno	String	No	Lista de horarios a filtrar en el formato "hh:mm-hh:mm;hh:mm-hh:mm;hh:mm-hh:mm;esses horários são referentes às opções pré existentes na aplicação Câmeras, a saber: "Madrugada" (compreenderá horários entre 00h e 06h), "Manhã" (compreenderá horários entre 06h:01min e 12h), "Tarde" (compreenderá horários entre 12h:01min e 18h) e "Noite" (compreenderá horário entre 18h:01min e 23h:59min). Poderá ser selecionado mais de um turno;

Nombre	Tipo	Obligatorio	Descripción
diaSemana	Integer	No	lista con días de la semana en el formato "1,2,3,4,5,6,7" (sin comillas), se utiliza el estándar ISO-8601, donde Lunes=1, Martes=2, Miércoles=3, Jueves=4, Viernes=5, Sábado=6, Domingo=7
quantidade	Integer	No	Cantidad que se considerará para mostrar los eventos; por ejemplo, si se informa el valor 2, solo se devolverán los vehículos que tuvieron dos eventos iguales, si se informa el valor 4, solo se devolverán los vehículos que tuvieron cuatro eventos iguales.
status	String	No	Lista con estados a filtrar en el formato " 'validated','not_validated' " (sin comillas dobles, solo comillas simples). Los estados deben ser los siguientes: - Validado: 'validated' - No validado: 'not_validated' - Esperando video: 'waiting_video' - Falla: 'failed'

Retorno

Retorna una lista de objetos do tipo '[SmartCamerasEvento](#)' con as propiedades abaixo:

Nombre	Tipo	Descripción
deviceId	Integer	Reservado del sistema
driver	String	Lista los conductores
cpf	String	Número de documento del conductor
nome	String	Nombre del conductor
registration	String	Matrícula del conductor
eventType	String	Tipo de evento realizado por el vehículo
hwType	String	Reservado do sistema
id	Integer	Reservado do sistema
messageId	String	Reservado do sistema
event_id	Integer	Código del evento
gps_valid	Integer	Reservado do sistema
lat	Double	Valor de la latitud del vehículo que generó el evento
lon	Double	Valor de la longitud del vehículo que generó el evento
timestamp	String	Fecha y hora del evento en UTC 0
vel	Integer	Velocidad del vehículo
pkDeviceDate	Integer	Reservado do sistema

Nombre	Tipo	Descripción
plate	String	Patente del vehículo que generó el evento



OBS.: La información se obtendrá solo si hay al menos una nota registrada. Si no se pasan los parámetros opcionales, se devolverán todos los eventos disponibles.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getSmartCamerasEvents>
      <!--Optional:-->
      <usuario>LOGIN</usuario>
      <!--Optional:-->
      <agrupador>LOGINAGRUPADOR</agrupador>
      <!--Optional:-->
      <senha>SENHA</senha>
      <offset>0</offset>
      <limit>100</limit>
      <!--Optional:-->
      <motoristas/>
      <!--Optional:-->
      <veiculos/>
      <!--Optional:-->
      <dataInicio>17-06-2021</dataInicio>
      <!--Optional:-->
      <dataFim>17-06-2021</dataFim>
      <!--Optional:-->
      <tipoEvento/>
      <!--Optional:-->
      <criticidade/>
      <!--Optional:-->
      <turno/>
      <!--Optional:-->
      <diaSemana/>
      <quantidade>2</quantidade>
      <!--Optional:-->
      <status/>
    </web:getSmartCamerasEvents>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <work:WorkContext xmlns:work="http://oracle.com/weblogic/soap/workarea/">
      XYZ</work:WorkContext>
    </S:Header>
  <S:Body>
    <ns0:getSmartCamerasEventsResponse xmlns:ns0=
"http://webservice.web.integracao.sascar.com.br/">
      <return>
        <deviceId>1265698</deviceId>
        <driver>
          <cpf/>
          <nome>No driver</nome>
          <registration/>
        </driver>
        <eventType>1</eventType>
        <hwType>streamax</hwType>
        <id>STREAMAX_ALARMID_2312615</id>
        <messageId>2312615</messageId>
        <payload>
          <event_id>56006</event_id>
          <gps_valid>1</gps_valid>
          <lat>-22.04364</lat>
          <lon>-47.430843</lon>
          <timestamp>2021-06-17 00:00:18</timestamp>
          <vel>45</vel>
        </payload>
        <pkDeviceDate>STREAMAX_2021-06-17_1265698</pkDeviceDate>
        <plate>GFA3536</plate>
        <timestamp>2021-06-17 00:00:18</timestamp>
      </return>
      <return>
        <deviceId>1265698</deviceId>
        <driver>
          <cpf/>
          <nome>No driver</nome>
          <registration/>
        </driver>
        <eventType>1</eventType>
        <hwType>streamax</hwType>
        <id>STREAMAX_ALARMID_2312617</id>
        <messageId>2312617</messageId>
        <payload>
          <event_id>56006</event_id>
          <gps_valid>1</gps_valid>
          <lat>-22.043132</lat>
          <lon>-47.430967</lon>
          <timestamp>2021-06-17 00:00:23</timestamp>
          <vel>41</vel>
        </payload>
        <pkDeviceDate>STREAMAX_2021-06-17_1265698</pkDeviceDate>
        <plate>GFA3536</plate>
        <timestamp>2021-06-17 00:00:23</timestamp>
      </return>
    </ns0:getSmartCamerasEventsResponse>
  </S:Body>
</S:Envelope>
```

7.8. SolicitarEventosCaixaPreta



Este método está desativado, sem previsão de liberação.

Descripción

Método que permite solicitar la recuperación de eventos caja negra almacenados en el sdcard del equipo para su grabación en la base de datos.

Declaración del Método

```
String solicitarEventosCaixaPreta(String usuario, String senha, String veoid, String placa, String dataPosicaoInicial, String dataPosicaoFinal)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
veoid	String	Sí ¹	Id del vehículo para el cual se desea recuperar los eventos caja negra.
placa	String	Sí ¹	Patente del vehículo para el cual se desea recuperar los eventos caja negra.
dataPosicaoInicial	String	No ²	Indica la fecha inicial para el intervalo del cual se desea obtener los eventos caja negra.
dataPosicaoFinal	String	No ²	Indica la fecha final para el intervalo del cual se desea obtener los eventos caja negra.



Si el campo “veoid” es informado, el campo “placa” pasa a ser opcional. Si el campo “placa” es informado, el campo “veoid” pasa a ser opcional.



No es posible informar solo el campo “dataPosicaoInicial” o el campo “dataPosicaoFinal”, sin embargo, se pueden omitir ambos y el servicio enviará un comando solicitando los últimos 10 minutos almacenados.

Retorno

Retorna el indicativo de ejecución del comando que solicita la recuperación de los datos de caja negra almacenados en el sd-card del rastreador:

Nombre	Tipo	Descripción
Código de retorno	Integer	Indica si el comando fue ejecutado: 7: comando ejecutado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:solicitarEventosCaixaPreta>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <idVeiculo>1575379</idVeiculo>
      <!--Optional:-->
      <placa>?</placa>
      <!--Optional:-->
      <dataPosicaoInicial>?</dataPosicaoInicial>
      <!--Optional:-->
      <dataPosicaoFinal>?</dataPosicaoFinal>
      <!--Optional:-->
      <ticket>?</ticket>
    </web:solicitarEventosCaixaPreta>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:Fault xmlns:ns0="http://schemas.xmlsoap.org/soap/envelope/"
      xmlns:ns1="http://www.w3.org/2003/05/soap-envelope">
      <faultcode>ns0:Server</faultcode>
      <faultstring>Atencao: Método desativado. Sem previsão de liberação.</faultstring>
    </ns0:Fault>
  </S:Body>
</S:Envelope>
```

7.9. RecuperarEventosCaixaPreta

Descripción

Método que permite consultar los eventos caja negra ya obtenidos del rastreador y almacenados en la base de datos.

Declaración del Método

```
CaixaPretaList recuperarEventosCaixaPreta(String usuario, String senha, String placa, String veioId, String dataPosicao)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
placa	String	Sí ¹	Patente del vehículo para el cual se desea recuperar los eventos caja negra.
veioId	String	Sí ¹	Id del vehículo para el cual se desea recuperar los eventos caja negra.
dataPosicao	String	Sí	Indica la fecha inicial para el intervalo del cual se desea obtener los eventos caja negra.



1: Si el campo “veioId” es informado, el campo “placa” pasa a ser opcional. Si el campo “placa” es informado, el campo “veioId” pasa a ser opcional.

Retorno

Retorna una instancia de CaixaPretaList, que encapsula a lista de eventos caixa preta contidos no intervalo de 10 minutos contados a partir do valor do campo “dataPosicao”:

Nombre	Tipo	Descripción
caixaPretaList	List<Caixa Preta>	Elemento que encapsula la lista que contiene los eventos de caja negra obtenidos
dataEvento	String	Representa la fecha en que ocurrió el evento, incluyendo minutos y segundos
latitude	Double	Representa la latitud donde ocurrió el evento
longitude	Double	Representa la longitud donde ocurrió el evento

Nombre	Tipo	Descripción
idOperador	Integer	Representa el id del conductor registrado como conductor en el momento en que ocurrió el evento
velocidade	Integer	Representa la velocidad a la que se encontraba el vehículo en el momento en que ocurrió el evento
rpm	Integer	Representa el valor de las rotaciones en el momento en que ocurrió el evento
ignicao	Integer	Representa el estado de la ignición en el momento en que ocurrió el evento: 0: Apagada 1: Encendida
freio	Integer	Representa el estado del freno en el momento en que ocurrió el evento: 0: Desactivado 1: Activado
limpador	Integer	Representa el estado del limpiaparabrisas en el momento en que ocurrió el evento: 0: Desactivado 1: Activado
buzzer	Integer	Representa el estado del buzzer en el momento en que ocurrió el evento: 0: Desactivado 1: Activado
embreagem	Integer	Representa el estado del embrague en el momento en que ocurrió el evento: 0: Desactivado 1: Activado

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:recuperarEventosCaixaPreta>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <placa?></placa>
      <!--Optional:-->
      <idVeiculo>1903888</idVeiculo>
      <!--Optional:-->
      <dataPosicao?></dataPosicao>
    </web:recuperarEventosCaixaPreta>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:recuperarEventosCaixaPretaResponse
xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        <caixaPreta>
          <dataEvento>26/02/2016 23:08:00</dataEvento>
          <latitude>-25.323203</latitude>
          <longitude>-49.119539</longitude>
          <idOperador>1</idOperador>
          <velocidade>0</velocidade>
          <rpm>807</rpm>
          <ignicao>-1</ignicao>
          <freio>1</freio>
          <limpador>1</limpador>
          <buzzer>1</buzzer>
          <embreagem>1</embreagem>
        </caixaPreta>
        <caixaPreta>
          <dataEvento>26/02/2016 23:08:01</dataEvento>
          <latitude>-25.323203</latitude>
          <longitude>-49.119539</longitude>
          <idOperador>1</idOperador>
          <velocidade>0</velocidade>
          <rpm>807</rpm>
          <ignicao>-1</ignicao>
          <freio>1</freio>
          <limpador>1</limpador>
          <buzzer>1</buzzer>
          <embreagem>1</embreagem>
        </caixaPreta>
        <caixaPreta>
          <dataEvento>26/02/2016 23:08:02</dataEvento>
          <latitude>-25.323203</latitude>
          <longitude>-49.119539</longitude>
          <idOperador>1</idOperador>
          <velocidade>0</velocidade>
          <rpm>807</rpm>
          <ignicao>-1</ignicao>
          <freio>1</freio>
          <limpador>1</limpador>
          <buzzer>1</buzzer>
          <embreagem>1</embreagem>
        </caixaPreta>
        <caixaPreta>
          <dataEvento>26/02/2016 23:08:03</dataEvento>
          <latitude>-25.323203</latitude>
          <longitude>-49.119539</longitude>
          <idOperador>1</idOperador>
          <velocidade>0</velocidade>
          <rpm>807</rpm>
          <ignicao>-1</ignicao>
          <freio>1</freio>
          <limpador>1</limpador>
          <buzzer>1</buzzer>
          <embreagem>1</embreagem>
        </caixaPreta>
        <caixaPreta/>
      </return>
    </ns0:recuperarEventosCaixaPretaResponse>
  </S:Body>
</S:Envelope>
```

26/02/2016 23:08:04 -25.323203 -49.119539 1 0 807 -1 1 1 1 1

8. Métodos – Motoristas

8.1. ObterMotoristas

Descripción

Método para consultar los datos de registro de conductores pertenecientes al INTEGRADOR.



Si el cliente tiene restricción, no será posible consultar los conductores del mismo.

Declaración del Método

```
List obterMotoristas(String usuario, String senha, Integer quantidade, Integer idMotorista)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	Sí	Cantidad máxima de registros de la consulta, siendo el límite de 1000 registros. Para obtener la información pertinente solo a un registro de conductor, complete el campo quantidade con el valor 1 (uno) e indique el id del conductor deseado en el campo idMotorista.
idMotorista	Integer	No	El límite máximo de resultados del método es 1000 registros. En este caso para obtener el resto de los registros, se implementó un sistema de paginación, donde el INTEGRADOR debe seguir las siguientes reglas: Para obtener la primera página con los 1000 registros iniciales enviar 0 (cero) como parámetro; Para obtener la siguiente página, enviar como parámetro el último id del conductor de la última página consultada. Para obtener la información pertinente solo a un registro de conductor, enviar únicamente el id del conductor deseado y en el campo quantidade el valor 1 (uno).

Retorno

Retorna una lista de objetos del tipo 'Motorista' con las propiedades a continuación:

Nombre	Tipo	Descripción
idMotorista	Integer	Id del conductor.
nome	String	Nombre del conductor.

Nombre	Tipo	Descripción
tipoMotorista	Char	Indica el tipo de conductor. A (Agregado) T (Tercero) F (Empleado)
dataContratacao	Calendar	Fecha de contratación del conductor.
tipoDocumento	Char	Tipo de documento utilizado para identificación. C (CNH) R (RG) F (CPF)
numeroDocumento	String	Número del documento utilizado para identificación.
tipoCNH	String	Tipo / Categoría de la licencia de conducir del conductor.
vencimentoCNH	Calendar	Fecha de vencimiento de la licencia de conducir del conductor.
telefone	String	Teléfono principal del conductor.
celular	String	Celular principal del conductor.
login	String	Login del conductor.
senha	String	Contraseña del conductor.
generico	Boolean	Indica si el conductor es genérico a la gestora. true (Conductor es genérico sin vínculo a cliente) false (Conductor tiene vínculo a cliente)

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/" >
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterMotoristas>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>Quantidade</quantidade>
      <!--Optional:-->
      <idMotorista>IDMOTORISTA</idMotorista>
    </web:obterMotoristas>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterMotoristasResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/" >
      <return>
        <celular>(77) 7777-7777</celular>
        <dataContratacao>2004-12-12T00:00:00.000-02:00</dataContratacao>
        <generico>>false</generico>
      </return>
    </ns0:obterMotoristasResponse>
  </S:Body>
</S:Envelope>
```

```
<idMotorista>6530</idMotorista>
<login>2424</login>
<nome>Ruy Leal</nome>
<numeroDocumento>4322222432222222222</numeroDocumento>
<senha>2424</senha>
<telefone>(77) 7777-7777</telefone>
<tipoCNH>AB</tipoCNH>
<tipoDocumento>R</tipoDocumento>
<vencimentoCNH>2009-09-28T00:00:00.000-03:00</vencimentoCNH>
</return>
</ns0:obterMotoristasResponse>
</S:Body>
</S:Envelope>
```

8.2. ObterMotoristasJson

Descripción

Método para consultar los datos de registro de conductores pertenecientes al INTEGRADOR, con retorno en formato JSON. Equivalente al método `obterMotoristas`, con la adición de los campos `ativo` e `dataInativacao`.



Los conductores eliminados o pertenecientes a clientes sin señal dirigida no son devueltos. Los clientes con restricción no permiten consultar sus conductores.

Declaración del Método

```
List obterMotoristasJson(String usuario, String senha, Integer quantidade, Integer idMotorista)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
<code>usuario</code>	String	Sí	Login del integrador para acceso al sistema.
<code>senha</code>	String	Sí	Contraseña del integrador para acceso al sistema.
<code>quantidade</code>	Integer	Sí	Cantidad máxima de registros de la consulta, siendo el límite de 1000 registros. Para obtener la información pertinente solo a un registro de conductor, complete el campo <code>quantidade</code> con el valor <code>1</code> e indique el id en el campo <code>idMotorista</code> .
<code>idMotorista</code>	Integer	No	Paginación: enviar <code>0</code> para obtener los primeros 1000 registros; para las páginas siguientes, enviar el último <code>idMotorista</code> de la página anterior. Para consultar un conductor específico, enviar su id con <code>quantidade = 1</code> .

Retorno

Retorna una lista de objetos del tipo `Motorista` con las propiedades a continuación:

Nombre	Tipo	Descripción
<code>idMotorista</code>	Integer	Id del conductor.
<code>nome</code>	String	Nombre del conductor.
<code>tipoMotorista</code>	Char	Tipo de conductor: <code>A</code> (Agregado), <code>T</code> (Tercero), <code>F</code> (Empleado).
<code>dataContratacao</code>	Calendar	Fecha de contratación del conductor.
<code>tipoDocumento</code>	Char	Tipo de documento: <code>C</code> (CNH), <code>R</code> (RG), <code>F</code> (CPF).
<code>numeroDocumento</code>	String	Número del documento utilizado para identificación.

Nombre	Tipo	Descripción
tipoCNH	String	Tipo / Categoría de la licencia de conducir del conductor.
vencimentoCNH	Calendar	Fecha de vencimiento de la licencia de conducir del conductor.
telefone	String	Teléfono principal del conductor.
celular	String	Celular principal del conductor.
login	String	Login del conductor.
senha	String	Contraseña del conductor.
generico	Boolean	true si el conductor es genérico sin vínculo a cliente; false si tiene vínculo.
ativo	Boolean	Situación de registro del conductor: true — conductor activo false — conductor inactivo
dataInativacao	Calendar	Fecha de inactivación del conductor. Completada cuando ativo = false ; null cuando el conductor está activo.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/">
  <soapenv:Header/>
  <soapenv:Body>
    <web:obterMotoristasJson>
      <usuario>Usuário</usuario>
      <senha>Senha</senha>
      <!--Optional:-->
      <quantidade>1000</quantidade>
      <!--Optional:-->
      <idMotorista>0</idMotorista>
    </web:obterMotoristasJson>
  </soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterMotoristasJsonResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/">
      <return>
        [
          {
            "idMotorista": 6530,
            "nome": "Ruy Leal",
            "tipoMotorista": "F",
```

```
"dataContratacao": "2004-12-12T00:00:00.000-02:00",
"tipoDocumento": "R",
"numeroDocumento": "4322222432222222222",
"tipoCNH": "AB",
"vencimentoCNH": "2009-09-28T00:00:00.000-03:00",
"telefone": "(77) 7777-7777",
"celular": "(77) 7777-7777",
"login": "2424",
"senha": "2424",
"generico": false,
"ativo": true,
"dataInativacao": null
}
]
</return>
</ns0:obterMotoristasJsonResponse>
</S:Body>
</S:Envelope>
```

8.3. ObterMotoristasVeiculos

Descripción

Método para consultar los datos de registro de vínculos entre conductores y vehículos pertenecientes al INTEGRADOR.

Declaración del Método

```
List obterMotoristasVeiculos(String usuario, String senha, Integer quantidade, Long idMotoristaVeiculo)
```

Parámetros

Nombre	Tipo	Obligatorio	Descripción
usuario	String	Sí	Login del integrador para acceso al sistema.
senha	String	Sí	Contraseña del integrador para acceso al sistema.
quantidade	Integer	Sí	Cantidad máxima de registros de la consulta, siendo el límite de 1000 registros.
idMotoristaVeiculo	Long	No	El límite máximo de resultados del método es 1000 registros. En este caso para obtener el resto de los registros, se implementó un sistema de paginación, donde el INTEGRADOR debe seguir las siguientes reglas: Para obtener la primera página con los 1000 registros iniciales enviar 0 (cero) como parámetro; Para obtener la siguiente página, enviar como parámetro el último id del conductor de la última página consultada.

Retorno

Retorna una lista de objetos del tipo 'MotoristaVeiculo' con las propiedades a continuación:

Nombre	Tipo	Descripción
idMotoristaVeiculo	Long	Id del registro de conductor y vehículo
idMotorista	Integer	Id del conductor.
idVeiculo	Integer	Código del vehículo.

Ejemplo XML

Request:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://webservice.web.integracao.sascar.com.br/" >
```

```
<soapenv:Header/>
<soapenv:Body>
  <web:obterMotoristasVeiculos>
    <usuario>Usuário</usuario>
    <senha>Senha</senha>
    <!--Optional:-->
    <quantidade>QuantidadeDeRetorno</quantidade>
    <!--Optional:-->
    <idMotoristaVeiculo>idMotoristaVeiculo</idMotoristaVeiculo>
  </web:obterMotoristasVeiculos>
</soapenv:Body>
</soapenv:Envelope>
```

Response:

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header>
    <WorkContext xmlns="http://oracle.com/weblogic/soap/workarea/">XYZ</WorkContext>
  </S:Header>
  <S:Body>
    <ns0:obterMotoristasVeiculosResponse xmlns:ns0="http://webservice.web.integracao.sascar.com.br/" >
      <return>
        <idMotorista>202</idMotorista>
        <idMotoristaVeiculo>202113516</idMotoristaVeiculo>
        <idVeiculo>113516</idVeiculo>
      </return>
    </ns0:obterMotoristasVeiculosResponse>
  </S:Body>
</S:Envelope>
```

9. Referencia – Estructura de Eventos y Excepciones

9.1. Estructura Actual

El sistema de integración actual tiene una estructura que dificulta la identificación de qué actuador fue el responsable de generar el evento disparado por el equipo. Esta limitación ocurre porque la identificación del actuador no es informada, siendo solo el estado de cada entrada/salida como se muestra en la tabla a continuación:

Nombre	Código Estado ON	Código Estado OFF	Descripción
Entrada 1	17	21	Evento generado por el dispositivo de la entrada 1, siendo 17 para entrada activada y 21 para desactivada.
Entrada 2	18	22	Evento generado por el dispositivo de la entrada 2, siendo 18 para entrada activada y 22 para desactivada.
Entrada 3	19	23	Evento generado por el dispositivo de la entrada 3, siendo 19 para entrada activada y 23 para desactivada.
Entrada 4	20	24	Evento generado por el dispositivo de la entrada 4, siendo 20 para entrada activada y 24 para desactivada.
Entrada 5	40	44	Evento generado por el dispositivo de la entrada 5, siendo 40 para entrada activada y 44 para desactivada.
Entrada 6	41	45	Evento generado por el dispositivo de la entrada 6, siendo 41 para entrada activada y 45 para desactivada.
Entrada 7	42	46	Evento generado por el dispositivo de la entrada 7, siendo 42 para entrada activada y 46 para desactivada.
Entrada 8	43	47	Evento generado por el dispositivo de la entrada 8, siendo 43 para entrada activada y 47 para desactivada.
Saída1	25	29	Evento generado por el dispositivo de la entrada 1, siendo 25 para entrada activada y 29 para desactivada.
Saída2	26	30	Evento generado por el dispositivo de la entrada 2, siendo 26 para entrada activada y 30 para desactivada.
Saída3	27	31	Evento generado por el dispositivo de la entrada 3, siendo 27 para entrada activada y 31 para desactivada.
Saída4	28	32	Evento generado por el dispositivo de la entrada 4, siendo 28 para entrada activada y 32 para desactivada.
Saída5	48	52	Evento generado por el dispositivo de la entrada 5, siendo 48 para entrada activada y 52 para desactivada.
Saída6	49	53	Evento generado por el dispositivo de la entrada 6, siendo 49 para entrada activada y 53 para desactivada.

Nombre	Código Estado ON	Código Estado OFF	Descripción
Saída7	50	54	Evento generado por el dispositivo de la entrada 7, siendo 50 para entrada activada y 54 para desactivada.
Saída8	51	55	Evento generado por el dispositivo de la entrada 8, siendo 51 para entrada activada y 55 para desactivada.
Bloqueio	33	34	Evento de bloqueo, siendo 33 para bloqueo activado y 34 bloqueo desactivado.
GPS	35	36	Evento de bloqueo, siendo 35 para bloqueo activado y 36 bloqueo desactivado.
Ignição	37	38	Evento de bloqueo, siendo 37 para bloqueo activado y 38 bloqueo desactivado.
Pânico	1	-	Evento de código 1 indicando la ocurrencia de pánico. Este tipo de evento no tiene el status OFF.
Alerta Interno	3	-	Evento de código 3 indicando la ocurrencia de alerta interno. Este tipo de evento no tiene el status OFF.
Alerta Bateria	4	-	Evento de código 4 indicando la ocurrencia de alerta de batería. Este tipo de evento no tiene el status OFF.
Velocidade	5	-	Evento de código 5 indicando la ocurrencia de alerta de exceso de velocidad. Este tipo de evento no tiene el status OFF.
Chegada Ponto de Referência	8	-	Evento de código 8 indicando la ocurrencia de llegada a punto de referencia. Este tipo de evento no tiene el status OFF.
Saída Ponto de Referência	9	-	Evento de código 9 indicando la ocurrencia de salida de punto de referencia. Este tipo de evento no tiene el status OFF.
Excesso Tempo	15	-	Evento de código 15 indicando la ocurrencia de exceso de tiempo. Este tipo de evento no tiene el status OFF.
Ponto Referência	16	-	Evento de código 16 indicando la ocurrencia de punto de referencia. Este tipo de evento no tiene el status OFF.

9.2. Nueva Estructura de Eventos

La nueva estructura proporcionará una forma más eficiente de identificación para los eventos generados por los equipos, ya que se proveerá el id del actuador como se muestra en el ejemplo a continuación:

Id del Actuador	Descripción
231	El código 231 es la identificación del sensor de violación de panel. En este caso el número positivo indica que el panel fue violado. Si la violación es corregida, se generará un nuevo evento con el número negativo.
-249	El código 249 es la identificación del sensor del baúl. En este caso el número negativo indica que el baúl fue cerrado. Si el baúl es abierto, se generará un nuevo evento con el número positivo.

El id del evento generado será la identificación exacta del actuador instalado en el equipo y podrá relacionarse con el resultado del método `ObterGrupoAtuadores()`, que tiene la relación y el detalle de todos los actuadores, sensores y accesorios susceptibles de instalación por Sascar.

9.3. Ejemplos de la Nueva Estructura de Eventos

Para ejemplificar el funcionamiento de la nueva estructura de eventos es necesario observar el campo evento **en una de las posiciones obtenidas por el método obterPacotePosicoes**. Este campo es la lista de los objetos Eventos **ocurridos en el espacio de tiempo entre la posición analizada y la anterior**. La lista contiene el **campo código del actuador** alterado y su estado indicado por el signo del código, siendo positivo cuando fue activado y negativo cuando fue desactivado.

Lista de eventos

Código	Descripción
231	Sensor de violación. (Activado)
-249	Sensor de baúl. (Desactivado)
251	Sensor de Puerta del Conductor.(Activado)
-241	Desenganche.(Desactivado)

En este ejemplo, el vehículo tuvo su sensor activado, el sensor de baúl desactivado, el sensor de puerta del conductor activado y desenganche desactivado. Si se altera cualquier estado del sensor, será enviado en la próxima posición.

9.4. Mensajes de Excepción

Mensaje	Motivo
Atención: ¡Vehículo sin permiso!	Contrato sin obligaciones financieras
Atención: ¡Vehículo sin dispositivo instalado!	Contrato sin telemetría instalada
Atención: ¡Vehículo no disponible para la Integradora!	Vehículo no dirigido

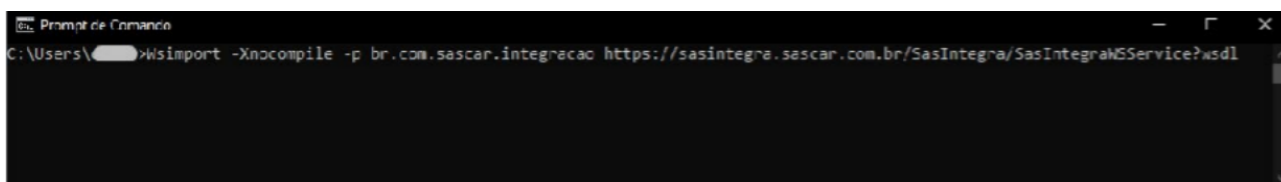
10. Anexo

10.1. Creación de Cliente para Conexión con el Web Service SasIntegra

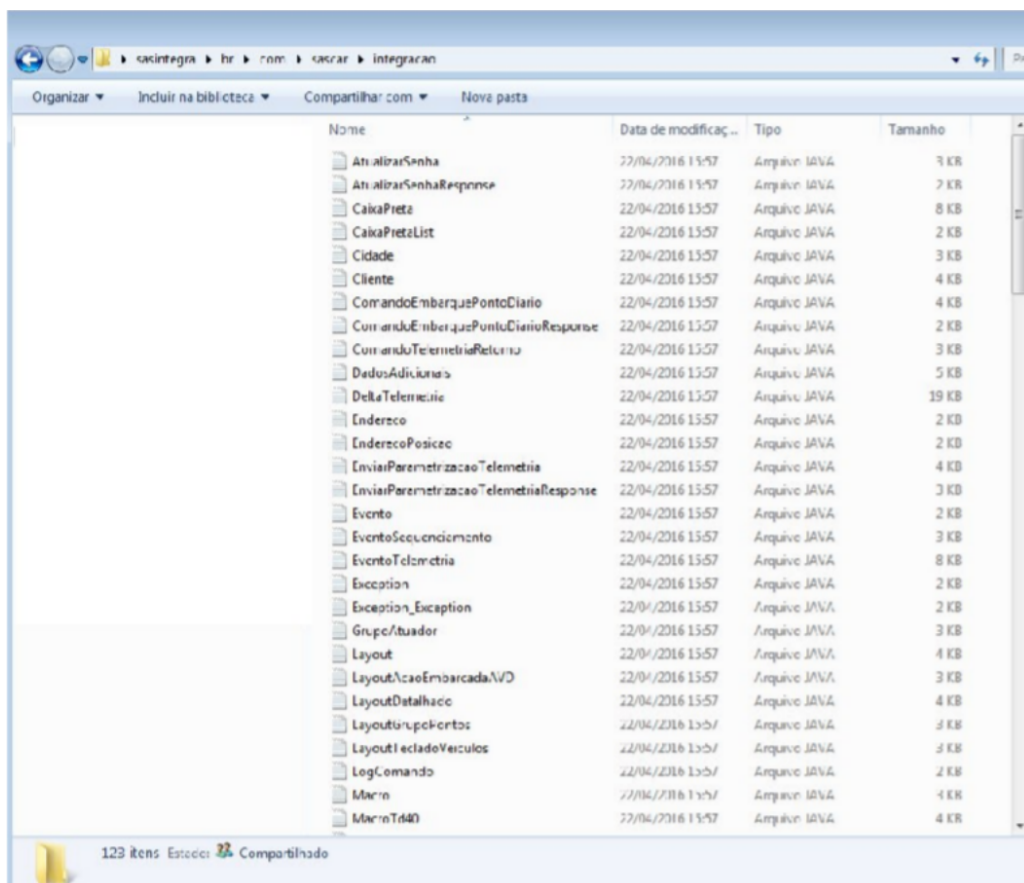
Para crear un cliente para conexión y uso de los métodos disponibles en SasIntegra, presentamos a continuación un ejemplo de un cliente desarrollado en lenguaje Java, para lectura del método de posiciones.

Primero es necesario importar los métodos del web service en su aplicación, mediante el import por WSDL, según el comando de importación utilizado a continuación.

```
wsimport -Xnocompile -p br.com.sascar.integracao  
https://sasintegra.sascar.com.br/SasIntegra/SasIntegraWSService?wsdl
```



Carpeta que contiene el archivo generado por wsimport:



Después de que los métodos estén importados en clases del lenguaje Java, es necesario utilizar los fragmentos de código a continuación para enviar una solicitud de paquetes de posición.

```
1 package br.com.sascar.integracao.main;
2
3 import java.util.List;
4
5 public class Main {
6
7     public static SasIntegraWS port;
8     public static List<PacotePosicao> posicaoList = null;
9
10    public static void main(String[] args) {
11        // para versões inferiores de Java JRE 6u11 o TLSv1.2 não é suportado
12        // para versões iguais ou acima do JRE 6u11 o TLSv1.2 é suportado mas precisa ser explicitado, assim como para o Java 7
13        //System.setProperty("https.protocols", "TLSv1.2");
14
15        SasIntegraWSService service = new SasIntegraWSService();
16        port = service.getSasIntegraWSPort();
17
18        try {
19            posicaoList = Main.port.obterPacotePosicoes("usuario", "senha", 300);
20            for (PacotePosicao pos : posicaoList) {
21                System.out.println("IdPacote: " + pos.getIdPacote());
22                System.out.println("IDVeiculo: " + pos.getIdVeiculo());
23                System.out.println("DataPacote: " + pos.getDataPacote());
24            }
25        } catch (Exception e) {
26
27        }
28    }
29 }
30
31
32
33 }
```

En la línea 19, el método `Main` realiza la conexión con el web service. Después de esto, se ejecuta la llamada para la lectura de 300 paquetes de posiciones, según el método anterior.

En la línea 23 se realiza la llamada del método `ObterPacotePosicoes`, donde debe informarse el usuario y contraseña del cliente y la cantidad de paquetes de posiciones que se desea recibir en el XML de retorno.

El objeto `posicao` puede iterarse dentro del bucle `for`, para su correspondiente procesamiento — por ejemplo, grabar cada posición contenida en el objeto en la base de datos, entre otras posibilidades.