



SDS

WebServices para Depositários

VERSÃO

| DATA | AUTOR | VERSÃO | COMENTÁRIO |
|-------------|--------------|---------------|--|
| 21-11-2017 | AT | 1.0 | Versão inicial |
| 12-02-2019 | AT | 1.1 | Possibilidade de indicar utilizadores orgânicos |
| 04-12-2019 | AT | 1.2 | Mais detalhe nas possíveis respostas em caso de erro |

Índice

| | | |
|----------|--|-----------|
| 1 | Introdução | 4 |
| 1.1 | Objetivo..... | 4 |
| 1.2 | Âmbito..... | 4 |
| 2 | Definições e Siglas | 5 |
| 2.1 | Siglas..... | 5 |
| 3 | WebServices | 6 |
| 3.1 | WebService WarehouseKeepers | 6 |
| 3.1.1 | Método: submitMessages | 6 |
| 3.1.2 | Método: retrieveNewOutputMessagesIds | 7 |
| 3.1.3 | Método: retrieveOutputMessagesByIds | 8 |
| 3.1.4 | Fluxo..... | 8 |
| 4 | Mensagens Suportadas..... | 10 |
| 5 | WSDL | 11 |

1 Introdução

1.1 Objetivo

O objetivo deste documento é descrever os WebServices disponibilizados para a interligação do sistema SDS com os Depositários (quer da Via Marítima quer da Via Aérea), sendo descritas todas as funcionalidades e modo de utilização dos mesmos.

1.2 Âmbito

O âmbito deste documento contempla a especificação técnica do sistema SDS.

2 Definições e Siglas

2.1 Siglas

SDS – Sistema Integrado dos Meios de Transporte e das Mercadorias

XML – Formato das mensagens enviadas e geradas pelo sistema (eXtensible Markup Language)

3 WebServices

De forma a utilizar os WebServices o cliente necessitará de se autenticar no Portal das Finanças, autenticação esta cujo mecanismo deverá estar de acordo com o já definido pelo Portal.

Sob determinadas condições (sobrecarga do sistema, reenvio repetido e mensagens repetidas, entre outras), os WebServices poderão rejeitar os pedidos como política de garantia de estabilidade do sistema, ficando o cliente responsável, nestes casos, de efetuar novamente esses pedidos mais tarde.

Para a interligação entre o sistema SDS e os Depositários é disponibilizado o seguinte WebService.

3.1 WebService WarehouseKeepers

São disponibilizados três métodos no WebService:

- **submitMessages** – Este método permite o envio de uma ou mais mensagens em simultâneo e retorna “OK” no caso do sistema SDS ter recebido as mensagens com sucesso ou “NOK” em caso de existir algum erro na sua receção. O retorno da invocação deste método não está relacionado com o processamento do conteúdo das mensagens que será efetuado num momento posterior.
- **retrieveNewOutputMessagesIds** – Este método retorna a lista de IDs das mensagens novas que o sistema SDS gerou para o depositário. Esses IDs serão passados na invocação do próximo método *retrieveOutputMessagesByIds*.
- **retrieveOutputMessagesByIds** – Este método retorna as mensagens XML indicadas na lista de IDs.

3.1.1 Método: submitMessages

Este método é responsável pela receção e processamento de mensagens enviadas pelos Depositários.

Parâmetros de Input:

- *String sender* – EORI correspondente ao depositário que está a efetuar a invocação deste método. Pode, opcionalmente, ser concatenado no final o caracter “/” + utilizador orgânico (por ex: PT123456789/1). As mensagens de resposta e de notificação resultantes serão geradas contendo o *receiver* igual ao *sender*.

- *List<String> messages* – Lista de mensagens que o depositário pretende enviar para o SDS. Estas mensagens deverão estar codificadas em base64.

Retorno:

- *List<String>* – Lista com o resultado da receção de cada mensagem enviada:
 - “OK” – Quando a mensagem em causa foi recebida com sucesso (não deverá ser reenviada ao SDS);
 - “NOK <message>” – Quando a mensagem em causa não foi recebida com sucesso (deverá ser reenviada ao SDS). Por ex, é retornado “NOK Messages are undefined” caso o parâmetro *messages* seja vazio. Deve assim o sistema cliente validar se as respostas começam com “NOK ”.

3.1.2 Método: *retrieveNewOutputMessagesIds*

Este método é responsável por retornar ao Depositário quais são os IDs das mensagens novas que o sistema SDS gerou e que estão disponíveis para serem obtidas pelo mesmo.

Parâmetros de Input:

- *String receiver* – EORI correspondente ao depositário que pretende obter a lista de IDs das suas mensagens novas. Pode, opcionalmente, ser concatenado no final o carácter “/” + utilizador orgânico (por ex: PT123456789/1).

Retorno:

- Lista com os IDs da mensagens novas que estão disponíveis para serem obtidas com o método *retrieveOutputMessagesByIds*. Caso tenha sido indicado o utilizador orgânico no parâmetro *receiver*, só serão retornados os IDs destinados a esse utilizador orgânico. Caso tenha sido indicado só o EORI coletivo, serão retornados todos os IDs destinados a esse EORI (incluindo todos os utilizadores orgânicos). Caso haja algum erro é enviada a informação “NOK <message>”. Deve assim o sistema cliente validar se as respostas começam com “NOK ”.

Esta invocação não produz qualquer alteração de estado no sistema SDS e deve ser invocada periodicamente (por ex: de 1 em 1 minuto) para que o Depositário tenha conhecimento de que tem novas mensagens para vir buscar ao sistema SDS.

É da responsabilidade do Depositário guardar os IDs recebidos no retorno desta invocação antes de efetuar a invocação ao próximo *WebService retrieveOutputMessagesByIds*, para que eventuais problemas de comunicações não resultem em perda de dados entre os sistemas.

3.1.3 Método: `retrieveOutputMessagesByIds`

Este método é responsável por retornar ao Depositário as mensagens de saída identificadas pelos IDs indicados e marcar as mesmas como já enviadas (para deixarem de constar no retorno do Webservice `retrieveNewOutputMessagesIds`).

Parâmetros de Input:

- *String receiver* – EORI correspondente ao depositário que pretende obter as mensagens XML com os IDs indicados. Pode, opcionalmente, ser concatenado no final o carácter “/” + utilizador orgânico (por ex: PT123456789/1).
- *List<String> outputMessagesIds* – Lista dos IDs das mensagens de saída que o Depositário pretende obter.

Retorno:

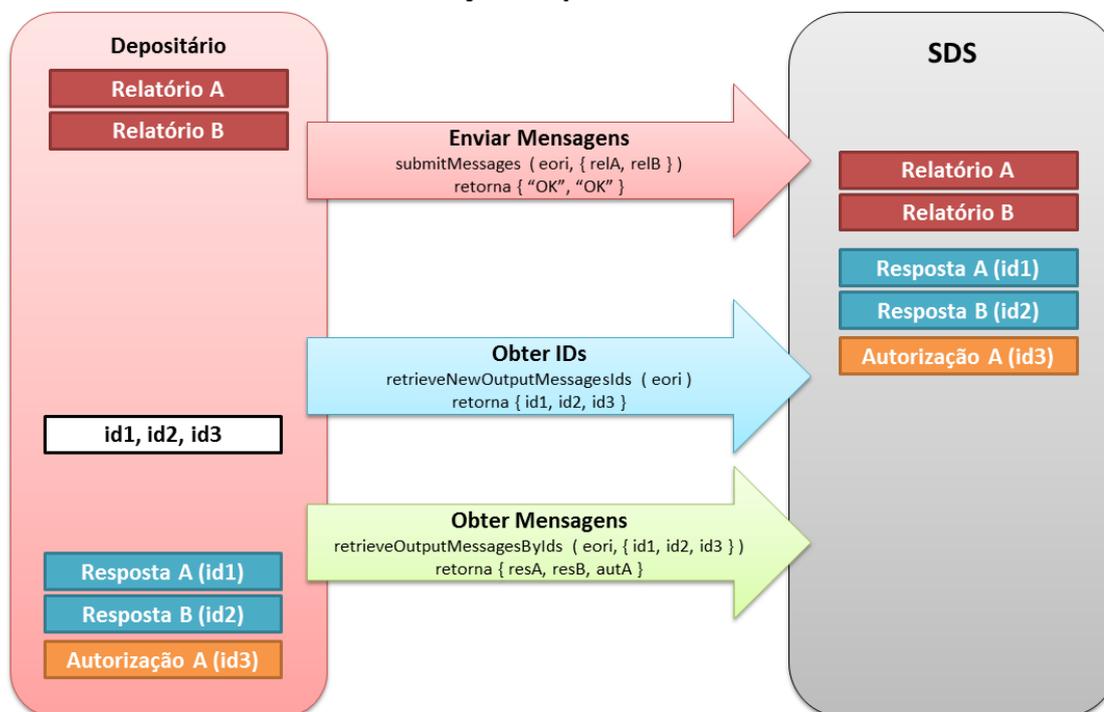
- Lista com as mensagens XML de saída cujos IDs foram indicados no input. Estas mensagens encontram-se codificadas em base64. Caso haja algum erro, é enviada a informação “NOK <message>”. Por ex, é retornado “NOK Some IDs are invalid” caso o parâmetro *outputMessagesIds* tenha IDs inválidos. Deve assim o sistema cliente validar se as respostas começam com “NOK”.

Caso haja problemas de comunicações no retorno das mensagens entre o sistema SDS e os Depositários e as mesmas não cheguem ao destino, os referidos IDs já poderão ter sido marcados como enviados. De qualquer das formas, este método retornará sempre as mensagens com os IDs indicados, pelo que poderá ser invocado mais do que uma vez. É da responsabilidade do Depositário manter os IDs retornados na invocação do método `retrieveNewOutputMessagesIds` e só os descartar quando realmente consegue receber com sucesso as respetivas mensagens XML.

3.1.4 Fluxo

De seguida ilustra-se um exemplo de troca de mensagens entre os Depositários e o SDS.

Comunicação Depositários - SDS



Neste exemplo cronologicamente são realizadas as seguintes ações:

- O Depositário envia dois Relatórios ao SDS invocando o método `submitMessages` passando como argumentos o seu EORI e as duas mensagens XML em base64. O SDS retorna uma lista de duas Strings com "OK", "OK" significando que o SDS recebeu as mensagens com sucesso.
- Logo após, o SDS processa as duas mensagens recebidas, e cria duas mensagens de resposta. Ainda antes da próxima invocação, o SDS entretanto já criou a mensagem de Autorização de Saída para a mercadoria do 1º Relatório.
- O Depositário invoca o método `retrieveNewOutputMessagesIds` que retorna 3 números de identificação de mensagens (as duas respostas e a autorização de saída).
- O Depositário recebe esses 3 identificadores, e guarda na sua Base de Dados para salvaguardar problemas de comunicação que possam ocorrer na invocação seguinte.
- O Depositário invoca o método `retrieveOutputMessagesByIds` indicando os 3 IDs, e o SDS retorna o XML das 3 mensagens em causa (em base64). Neste momento o SDS marca os 3 IDs como já respondidos, deixando de constar a partir desse momento do retorno do método `retrieveNewOutputMessagesIds`.
- O Depositário guarda as 3 mensagens XML recebidas e procede ao seu processamento.

4 Mensagens Suportadas

As mensagens de input suportadas são as seguintes:

- Via Marítima:
 - SeaWarehouseKeeperContainersReport
 - SeaWarehouseKeeperCargoReport
 - SeaWarehouseKeeperContainersMovement
 - SeaWarehouseKeeperCargoMovement
- Via Aérea:
 - AirWarehouseKeeperCargoReport
 - AirWarehouseKeeperCargoMovement

As mensagens retornadas pelo Webservice:

- Via Marítima:
 - SeaCustomsResponse
 - SeaCustomsNotification
- Via Aérea:
 - AirCustomsResponse
 - AirCustomsNotification

5 WSDL

Para a troca de mensagens com os depositários é disponibilizado o seguinte WSDL:

```
<?xml version='1.0' encoding='UTF-8'?>
<definitions xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
xmlns:wsp="http://www.w3.org/ns/ws-policy" xmlns:wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://server.ws.sds.dgita.gov.pt/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://server.ws.sds.dgita.gov.pt/" name="WarehouseKeepers">
<types>
<xsd:schema>
<xsd:import namespace="http://server.ws.sds.dgita.gov.pt/" schemaLocation="WarehouseKeepers.xsd"/>
</xsd:schema>
</types>
<message name="submitMessages">
<part name="parameters" element="tns:submitMessages"/>
</message>
<message name="submitMessagesResponse">
<part name="parameters" element="tns:submitMessagesResponse"/>
</message>
<message name="retrieveOutputMessagesByIds">
<part name="parameters" element="tns:retrieveOutputMessagesByIds"/>
</message>
<message name="retrieveOutputMessagesByIdsResponse">
<part name="parameters" element="tns:retrieveOutputMessagesByIdsResponse"/>
</message>
<message name="retrieveNewOutputMessagesIds">
<part name="parameters" element="tns:retrieveNewOutputMessagesIds"/>
</message>
<message name="retrieveNewOutputMessagesIdsResponse">
<part name="parameters" element="tns:retrieveNewOutputMessagesIdsResponse"/>
</message>
<portType name="WarehouseKeepers">
<operation name="submitMessages">
<input wsam:Action="http://server.ws.sds.dgita.gov.pt/WarehouseKeepers/submitMessagesRequest" message="tns:submitMessages"/>
<output wsam:Action="http://server.ws.sds.dgita.gov.pt/WarehouseKeepers/submitMessagesResponse"
message="tns:submitMessagesResponse"/>
</operation>
<operation name="retrieveOutputMessagesByIds">
<input wsam:Action="http://server.ws.sds.dgita.gov.pt/WarehouseKeepers/retrieveOutputMessagesByIdsRequest"
message="tns:retrieveOutputMessagesByIds"/>
<output wsam:Action="http://server.ws.sds.dgita.gov.pt/WarehouseKeepers/retrieveOutputMessagesByIdsResponse"
message="tns:retrieveOutputMessagesByIdsResponse"/>
</operation>
<operation name="retrieveNewOutputMessagesIds">
<input wsam:Action="http://server.ws.sds.dgita.gov.pt/WarehouseKeepers/retrieveNewOutputMessagesIdsRequest"
message="tns:retrieveNewOutputMessagesIds"/>
<output wsam:Action="http://server.ws.sds.dgita.gov.pt/WarehouseKeepers/retrieveNewOutputMessagesIdsResponse"
message="tns:retrieveNewOutputMessagesIdsResponse"/>
</operation>
</portType>
<binding name="WarehouseKeepersImplPortBinding" type="tns:WarehouseKeepers">
<soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
<operation name="submitMessages">
<soap:operation soapAction=""/>
<input>
<soap:body use="literal"/>
</input>
```

```

<output>
<soap:body use="literal"/>
</output>
</operation>
<operation name="retrieveOutputMessagesByIds">
<soap:operation soapAction=""/>
<input>
<soap:body use="literal"/>
</input>
<output>
<soap:body use="literal"/>
</output>
</operation>
<operation name="retrieveNewOutputMessagesIds">
<soap:operation soapAction=""/>
<input>
<soap:body use="literal"/>
</input>
<output>
<soap:body use="literal"/>
</output>
</operation>
</binding>
<service name="WarehouseKeepers">
<port name="WarehouseKeepersImplPort" binding="tns:WarehouseKeepersImplPortBinding">
<soap:address location="WarehouseKeepers"/>
</port>
</service>
</definitions>

```

Onde WarehouseKeepers.xsd:

```

<xsd:schema xmlns:ns0="http://server.ws.sds.dgita.gov.pt/" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://server.ws.sds.dgita.gov.pt">
<xsd:complexType name="retrieveOutputMessagesByIdsResponse">
<xsd:sequence>
<xsd:element name="return" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="submitMessages">
<xsd:sequence>
<xsd:element name="sender" type="xsd:string" minOccurs="0"/>
<xsd:element name="messages" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="retrieveNewOutputMessagesIdsResponse">
<xsd:sequence>
<xsd:element name="return" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="submitMessagesResponse">
<xsd:sequence>
<xsd:element name="return" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="retrieveNewOutputMessagesIds">
<xsd:sequence>
<xsd:element name="receiver" type="xsd:string" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="retrieveOutputMessagesByIds">
<xsd:sequence>

```

```
<xsd:element name="receiver" type="xsd:string" minOccurs="0"/>
<xsd:element name="outputMessagesIds" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
<xsd:element name="retrieveOutputMessagesByIdsResponse" type="ns0:retrieveOutputMessagesByIdsResponse"/>
<xsd:element name="submitMessages" type="ns0:submitMessages"/>
<xsd:element name="retrieveNewOutputMessagesIdsResponse" type="ns0:retrieveNewOutputMessagesIdsResponse"/>
<xsd:element name="submitMessagesResponse" type="ns0:submitMessagesResponse"/>
<xsd:element name="retrieveNewOutputMessagesIds" type="ns0:retrieveNewOutputMessagesIds"/>
<xsd:element name="retrieveOutputMessagesByIds" type="ns0:retrieveOutputMessagesByIds"/>
</xsd:schema>
```