



TransUnion[®] Universal Order Connect

UOC BUY (LSR-UOM) API Guide

March 01, 2024

Version 2.0.0



© 2024 TransUnion LLC

All Rights Reserved

No part of this publication may be reproduced or distributed in any form or by any means, electronic or otherwise, now known or hereafter developed, including, but not limited to, the Internet, without the express written permission of TransUnion. This document is protected by U.S. and International copyright laws. This document and the subject matter contained herein is TransUnion proprietary and confidential information, and may not be shared or used for any purposes other than the purpose for which it was provided by TransUnion, without the express written permission of TransUnion. By using this document, you are agreeing that you will not attempt, directly or indirectly, to reverse engineer, decompile, or disassemble any TransUnion services or service information, any confidential or proprietary criteria developed or used by TransUnion relating to services you receive from TransUnion, or any of the TransUnion confidential and proprietary information contained herein. The entire right, title and interest in and to this document and TransUnion services, and all copyrights, patents, trade secrets, trademarks, trade names, and all other intellectual property rights associated with any and all ideas, concepts, techniques, inventions, processes, or works of authorship including, but not limited to, all materials in written or other tangible form developed or created by TransUnion, shall at all times vest exclusively in TransUnion. You acknowledge that any misuse, misappropriation or threatened misappropriation of TransUnion's intellectual property rights, or any breach or threatened breach of the foregoing restrictions, may cause immediate and irreparable injury to TransUnion, and in such event, TransUnion shall be entitled to seek injunctive relief. Nothing stated herein will be construed to limit any other remedies available to TransUnion including, but not limited to suspension and/or termination of the services provided to you.

Requests for permission to reproduce or distribute any part of, or all of, this publication should be mailed to:

Law Department
TransUnion
555 West Adams
Chicago, Illinois 60661

The "tu" logo, TransUnion, and other trademarks, service marks, and logos (the "Trademarks") used in this publication are registered or unregistered Trademarks of TransUnion LLC or their respective owners. Trademarks may not be used for any purpose whatsoever without the express written permission of the Trademark owner.

[transunion.com](https://www.transunion.com)

Table of contents

- Table of contents 3**

- 1. About This Guide 5**
 - 1.1 Document History 5
 - 1.1.1 Audience 5
 - 1.1.2 Conventions 5
 - 1.1.3 Assumptions 6
 - 1.1.4 Contacting TransUnion 7

- 2. UOC Overview 8**
 - 2.1 The UOC application 8

- 3. Integration & Model Overview 9**
 - 3.1 Integration Overview 9
 - 3.1.1 Accessing the UOC API 9
 - 3.1.2 SOAP Server-Client Interactions 9
 - 3.1.3 Error Handling 10
 - 3.2.4 HTTP Status Codes for SOAP calls 10

- 4. Order 11**
 - 4.1 LSR Order Transactions 11
 - 4.2 Order WSDL 11
 - 4.3 Order APIs 13
 - 4.3.1 Order Submit 13
 - Request 13
 - Response 14
 - 4.3.2 Order Response Push Notifications 14

- 5. Pre-Order 16**
 - 5.1 LSR Pre-Order Transactions 16
 - 5.2 Pre-Order WSDL 16
 - 5.3 Pre-Order APIs 18
 - 5.3.1 Pre-Order Submit 18

Request	18
Response	18
5.3.2 Pre-Order Response Push Notifications	19
7. Appendix	21
7.1. Base URL	21

1. About This Guide

The Universal Order Connect (UOC) API provides SOAP interface to the UOC LSR application. APIs are designed to provide customers with a simple way of interfacing with UOC following standard industry practices.

1.1 Document History

The following table lists the change history of the document to date:

Table 1: Document History

Date	Comments
03/01/2024	UOC LSR Buyer SOAP API Guide baseline

1.1.1 Audience

This document is designed for developers with programming experience and a working knowledge of the terms and procedures used to implement SOAP web-services.

1.1.2 Conventions

The following table lists notational conventions found throughout this document:

Table 2: Conventions

Convention	Description	Example
<p>Consoles, Constant width</p>	<p>Used within command, code, XML request/response and file samples. Indicate SOAP API request/response. Might be emphasized with bold.</p>	<pre data-bbox="943 1297 1459 1493"><soap:Envelope> <soap:Body> <response>OK</response> </soap:Body> </soap:Envelope></pre>
<p><u>Hypertext link</u></p>	<p>Indicates a hypertext link that, if clicked, takes you to either an HTML page or a URL. A default browser must be specified.</p>	<p>Click here to view the link.</p>

Convention	Description	Example
Cross-reference	Used to indicate a cross-reference that, if clicked, takes you to the indicated location in the document.	See Formatting Text on page 13
Note:	A note symbol provides supporting information that is not explicitly addressed in the accompanying text.	Note: This symbol indicates supporting information.
Date/Time	The UOC application is housed and maintained on the east coast. As such, the system records and displays dates and times based on the current eastern time, which, in the summer, is defined more specifically as Eastern Daylight Time (EDT) and as Eastern Standard Time (EST) in the winter.	n/a

1.1.3 Assumptions

The following assumptions were made in the creation of this document:

Users of this document have programming experience with a working knowledge of the terms and procedures used to implement SOAP web-services and XML.

Users of this document are connecting to the TransUnion’s UOC application via the API interface.

1.1.4 Contacting TransUnion

If you need technical assistance, please contact your account manager or technical support using any of the following means:

Registered customers with a username and password can open tickets through Neusupport (<http://www.support.neustar>).

(844) 677-2878, option 1, option 1.

Or write to us at:

OMS Customer Support TransUnion, Inc.

4th Floor

1650 Lyndon Farm Ct.

Louisville, KY 40223

2. UOC Overview

2.1 The UOC application

TransUnion’s Universal Order Connect provides a generic universal interface with focus on customer across various order types that allows customer to order from many Trading Partners. It also provides a flexibility to define and use customer specific limited fields (defined via Product Catalog) per Service thus making it Simplified Operations.

TransUnion’s UOC defines and adheres to the strict industry guidelines for various services and maintains the interconnection relationships with the Service Providers.

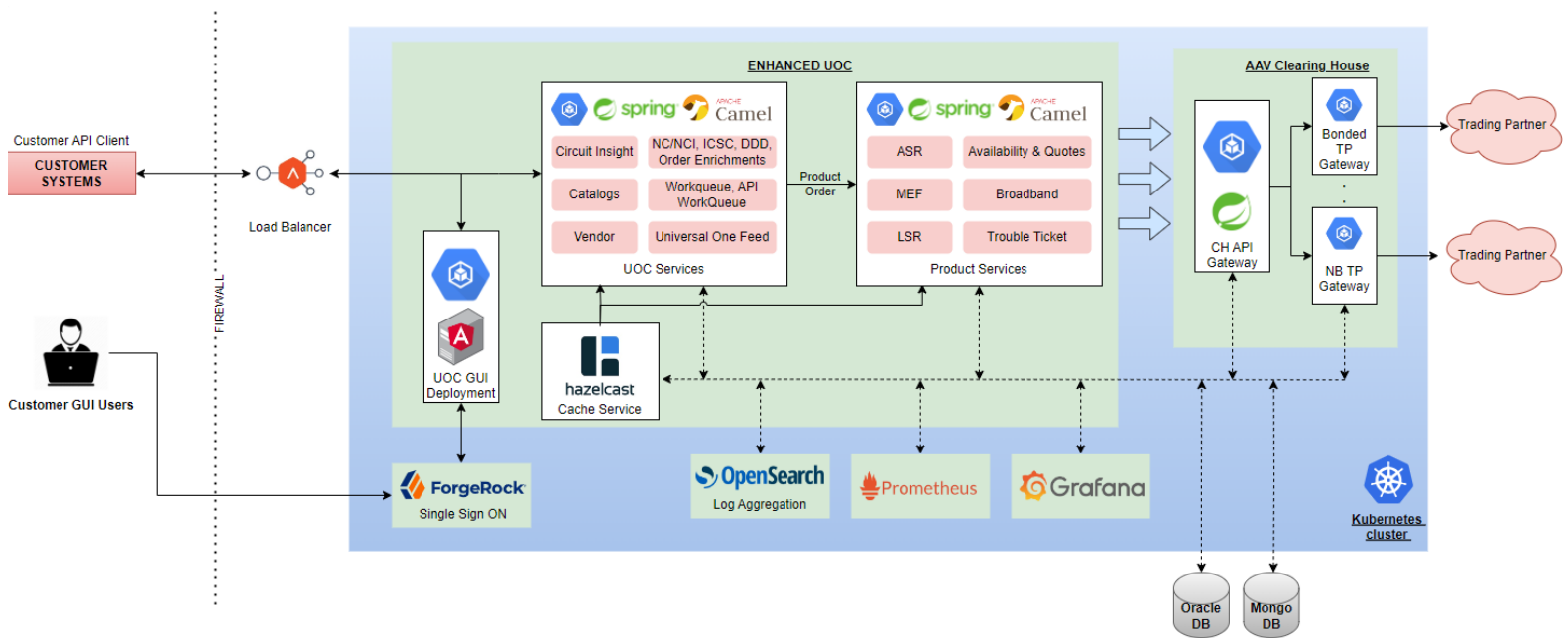


Figure 1: UOC architecture

3. Integration & Model Overview

This chapter provides a description of the SOAP-RPC interface used to communicate with the UOC. SOAP resolves integration problems caused by language and platform dependencies, easily allowing you to integrate with the UOC.

3.1 Integration Overview

3.1.1 Accessing the UOC API

UOC supports SOAP-RPC (Simple Object Access Protocol - Remote Procedure Call) over HTTPS as a means for systems to interact with the UOC (receiving requests and submitting responses). Systems can communicate directly with the UOC via properly formatted XML messages sent via SOAP-RPC. The UOC is accessed either by the SOAP adapter or the UOC GUI as shown below.

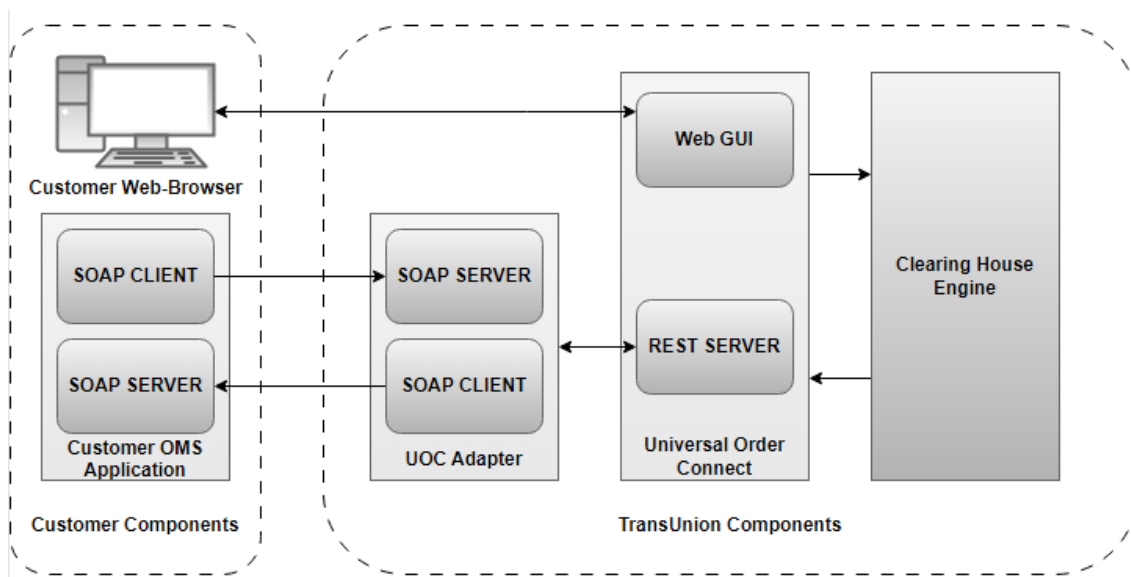


Figure 2: Accessing the UOC

3.1.2 SOAP Server-Client Interactions

The SOAP-RPC communication consists of the following primary components.

- SOAP client: Calls a method on a service.
- SOAP server: Provides the service and the implementation of the method being called.

TransUnion customer interaction with the Clearinghouse API is via SOAP-RPC. SOAP is a lightweight, XML-based protocol for exchanging information in a decentralized and distributed environment. SOAP consists of three primary parts:

- An envelope that defines a framework for describing what is in a message and how to process it.
- A set of encoding rules for expressing instances of application-defined data types.
- A convention for representing remote procedure calls and responses.

For additional information and specifications, see: SOAP Specification: <http://www.w3.org/TR/SOAP>

3.1.3 Error Handling

The success/error condition of the API call should be encoded into the HTTP response code, so that a valid success call will return a 200 if it is returning data; 204 otherwise. On an error, the general status code (404-resource not found, 400 for badly formed input, 409 if the caller is trying to create a duplicate resource or invalid input, 500 if there was a server error which the caller can not work around) is encoded in the http response code, but further details are provided in the body.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soapenv:Fault xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <faultcode>A9999</faultcode>
      <faultstring>Some internal error occurred in server. Please contact
support.</faultstring>
    </soapenv:Fault>
  </soap:Body>
</soap:Envelope>
```

3.2.4 HTTP Status Codes for SOAP calls

Following are the status codes and their meaning in below table. ** is a place holder. For e.g. 200 stands for the request has succeeded.

Table 4: HTTP status codes

Status Code (** - Place holder)	Meaning
1**	Informational Response
2**	Success Response
3**	Redirection Response
4**	Client Error Response
5**	Server Error Response

4. Order

This Chapter cover UOC order API. The primary function of LSR send service is to assist the TransUnion customers in the submission of LSR Order requests and the delivery of responses back.

4.1 LSR Order Transactions

This product encompasses the submission and management of LSR Order transactions by the trading partners. UOC LSR service will support the following suite of LSR Order transactions:

LSR Order:

Sr No	LSR in UOC-M	REQTYP
1	Loop Service	AB
2	Platform Service with Port	MB
3	Loop Service with Number Portability	BB
4	Resale	EB
5	Resale Private Line	KB
6	Centrex Resale	PB
7	ISDN BRI PRI	QB
8	Directory Listings Assistance	JB
9	Port Service	FB
10	Platform Service	DB
11	Resale Frame Relay	LB
12	DID DOD PBX Service	NB

4.2 Order WSDL

Below is the WSDL for LSR order. Please use 'processAsyncRequest' for submission of LSR orders to UOC.

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions targetNamespace="java:lsr.webservice.wisor.com"
xmlns:apachesoap="http://xml.apache.org/xml-soap"
xmlns:tns="java:lsr.webservice.wisor.com"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <message name="request">
    <part xmlns:partns="http://www.w3.org/2001/XMLSchema" type="partns:string"
name="string"/>
  </message>
  <message name="response">
```

```

    <part xmlns:partns="http://www.w3.org/2001/XMLSchema" type="partns:string"
name="response"/>
  </message>
  <portType name="lsrport">
    <operation name="processAsyncRequest">
      <input message="tns:request"/>
      <output message="tns:response"/>
    </operation>
    <operation name="processSyncRequest">
      <input message="tns:request"/>
      <output message="tns:response"/>
    </operation>
  </portType>
  <binding type="tns:lsrport" name="lsrport">
    <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="processAsyncRequest">
      <soap:operation soapAction=""/>
      <input>
        <soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="java:lsr.webservice.wisor.com" use="literal"/>
      </input>
      <output>
        <soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="java:lsr.webservice.wisor.com" use="literal"/>
      </output>
    </operation>
    <operation name="processSyncRequest">
      <soap:operation soapAction=""/>
      <input>
        <soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="java:lsr.webservice.wisor.com" use="literal"/>
      </input>
      <output>
        <soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="java:lsr.webservice.wisor.com" use="literal"/>
      </output>
    </operation>
  </binding>
  <service name="lsrservice">
    <port binding="tns:lsrport" name="lsrport">
      <soap:address
location="http://10.16.4.23:8022/lsruomwebservice/services/lsrport"/>
    </port>
  </service>
</definitions>

```

4.3 Order APIs

Services require you to submit requests directly to the UOC using SOAP payload. This chapter provides specific details on sending Service Requests and receiving Service Response messages.

4.3.1 Order Submit

The order API would be responsible for submission of request to UOC. From above shared WSDL 'processAsyncRequest' operation needs to be invoked for order submission. It accepts string parameter as a LSOG compliant payload.

Request

URL: `https://{baseurl}/order`

HTTP Method: POST

Body:

Sample Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:java="java:lsr.webservice.wisor.com">
  <soapenv:Header>
    <tMLHeader xmlns="http://tml.t1m1.org/tML.Transport.xsd">
      <TransportID>{Unique transaction ID}</TransportID>
      <ApplicationType>LSROR</ApplicationType>
      <ApplicationVersion>1.0</ApplicationVersion>
      <From>{Buyer Name/DOMAIN}</From>
      <To>{Seller name/DOMAIN}</To>
      <SendTimestamp>{timestamp}</SendTimestamp>
      <RetryCount>0</RetryCount>
    </tMLHeader>
  </soapenv:Header>
  <soapenv:Body>
    <java:processAsyncRequest>
      <string>
        . . .
        ---LSOG Request Payload String escaped---
        . . .
      </string>
    </java:processAsyncRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response

The application will respond with status of the API call, with either success (OK) or fail (FAULT) message.

Successful response

Response Code: 200 OK

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <response xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">OK</response>
  </soap:Body>
</soap:Envelope>
```

Failure Response

Response Code: 200

Response Body:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soapenv:Fault xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <faultcode>U1059</faultcode>
      <faultstring>DDD cannot be less than the current date.</faultstring>
    </soapenv:Fault>
  </soap:Body>
</soap:Envelope>
```

4.3.2 Order Response Push Notifications

Below XML format will be used to push order responses to Customers. Customers need to develop a SOAP service which will accept request in below format. UOC supports Basic, OAuth2.0 authentication for Push notifications.

HTTP Method : POST

Body

```
<soapenv:Envelope
```

```

xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<soapenv:Header>
  <ns1:tMLHeader soapenv:actor="" soapenv:mustUnderstand="0"
    xmlns:ns1="http://tml.t1m1.org/tML.Transport.xsd">
    <ns1:ApplicationType>LSROR</ns1:ApplicationType>
    <ns1:From>{SELLER NAME/DOMAIN}</ns1:From>
    <ns1:To>{BUYER NAME/DOMAIN}</ns1:To>
  </ns1:tMLHeader>
</soapenv:Header>
<soapenv:Body>
  <processAsyncResponse>
    <string>
    . . .
    ---LSOG Response Payload string escaped---
    . . .
    </string>
  </processAsyncResponse>
</soapenv:Body>
</soapenv:Envelope>

```

5. Pre-Order

This Chapter cover UOC pre-order API. The primary function of LSR send service is to assist the TransUnion customers in the submission of LSR Pre-Order requests and the delivery of responses back.

5.1 LSR Pre-Order Transactions

This product encompasses the submission and management of LSR Pre-Order transactions by the trading partners. UOC LSR service will support the following suite of LSR Pre-Order transactions:

LSR Preorder:

Transaction Type	Transaction Description
A	Address Validation
H, J, K, Q	Loop Qualification
B	TN Inquiry/Assignment
E	Customer Service Information
N	Collocation Facility Assignment
T	Listings for Telephone Number

5.2 Pre-Order WSDL

Below is the WSDL for LSR pre-order. Please use 'processSyncRequest' for submission of LSR pre-orders to UOC.

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions targetNamespace="java:lsr.webservice.wisor.com"
xmlns:apachesoap="http://xml.apache.org/xml-soap"
xmlns:tns="java:lsr.webservice.wisor.com"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns="http://schemas.xmlsoap.org/wsdl/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <message name="request">
    <part xmlns:partns="http://www.w3.org/2001/XMLSchema" type="partns:string"
name="string"/>
  </message>
  <message name="response">
    <part xmlns:partns="http://www.w3.org/2001/XMLSchema" type="partns:string"
name="response"/>
  </message>
```



```

<portType name="lsrport">
  <operation name="processAsyncRequest">
    <input message="tns:request"/>
    <output message="tns:response"/>
  </operation>
  <operation name="processSyncRequest">
    <input message="tns:request"/>
    <output message="tns:response"/>
  </operation>
</portType>
<binding type="tns:lsrport" name="lsrport">
  <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="processAsyncRequest">
    <soap:operation soapAction=""/>
    <input>
      <soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="java:lsr.webservice.wisor.com" use="literal"/>
    </input>
    <output>
      <soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="java:lsr.webservice.wisor.com" use="literal"/>
    </output>
  </operation>
  <operation name="processSyncRequest">
    <soap:operation soapAction=""/>
    <input>
      <soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="java:lsr.webservice.wisor.com" use="literal"/>
    </input>
    <output>
      <soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="java:lsr.webservice.wisor.com" use="literal"/>
    </output>
  </operation>
</binding>
<service name="lsrservice">
  <port binding="tns:lsrport" name="lsrport">
    <soap:address
location="http://10.16.4.23:8022/lsruomwebservice/services/lsrport"/>
  </port>
</service>
</definitions>

```

5.3 Pre-Order APIs

Services require you to submit requests directly to the UOC using SOAP payload. This chapter provides specific details on sending Service Requests and receiving Service Response messages.

5.3.1 Pre-Order Submit

The order API would be responsible for submission of request to UOC. From above shared WSDL 'processSyncRequest' operation needs to be invoked for order submission. It accepts string parameter as a LSOG compliant payload.

Request

URL: <https://{baseurl}/preOrder>

HTTP Method: POST

Body:

Sample Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:java="java:lsr.webservice.wisor.com">
  <soapenv:Header>
    <tMLHeader xmlns="http://tml.t1m1.org/tML.Transport.xsd">
      <TransportID>{Unique transaction ID}</TransportID>
      <ApplicationType>LSRPO</ApplicationType>
      <ApplicationVersion>1.0</ApplicationVersion>
      <From>{Buyer Name/DOMAIN}</From>
      <To>{Seller name/DOMAIN}</To>
      <SendTimestamp>{timestamp}</SendTimestamp>
      <RetryCount>0</RetryCount>
    </tMLHeader>
  </soapenv:Header>
  <soapenv:Body>
    <java:processSyncRequest>
      <string><![CDATA[
. . .
---LSOG Request Payload---
. . .
]]></string>
    </java:processSyncRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

Response

The application will respond with status of the API call, with either success (OK) or fail (FAULT) message.

Successful response

Response Code: 200 OK

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <response xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">OK</response>
  </soap:Body>
</soap:Envelope>
```

Failure Response

Response Code: 200

Response Body:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>Fault occurred while processing.</faultstring>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

5.3.2 Pre-Order Response Push Notifications

Below XML format will be used to push pre-order responses to Customers. Customers need to develop a SOAP service which will accept request in below format. UOC supports Basic, OAuth2.0 authentication for Push notifications.

HTTP Method : POST

Body

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
    <ns1:tMLHeader soapenv:actor="" soapenv:mustUnderstand="0"
xmlns:ns1="http://tml.t1m1.org/tML.Transport.xsd">
      <ns1:ApplicationType>LSRPO</ns1:ApplicationType>
      <ns1:From>{SELLER NAME/DOMAIN}</ns1:From>
```

```
<ns1:To>{BUYER NAME/DOMAIN}</ns1:To>
</ns1:tMLHeader>
</soapenv:Header>
<soapenv:Body>
  <processAsyncResponse>
    <string>
      . . .
      ---LSOG Response Payload string escaped---
      . . .
    </string>
  </processAsyncResponse>
</soapenv:Body>
</soapenv:Envelope>
```

7. Appendix

7.1. Base URL

{baseurl} mentioned in this document refers to below Hosts:

Environment	HOST
CTE (Customer Test Environment)	<a href="https://<DOMAIN>/services/uom">https://<DOMAIN>/services/uom
PROD	<a href="https://<DOMAIN>/services/uom">https://<DOMAIN>/services/uom