NeuStar OMS Clearinghouse

SOA Customers: Use the SOA API guide on https://www.neustar.biz/convergentCH/content/docs/SOA_API_Guide.pdf

OMS Clearinghouse

LSR, ASR, LIDB, E911, PPV, ICP

API Guide

Date: April 2009

Table of Contents

Chapter 1 About This Guide 7

1.1 Document History 7

1.2 Audience 8

1.3 Related Documents 8

1.4 Conventions 9

1.5 Assumptions 9

1.6 Contacting NeuStar 10

Chapter 2 Clearinghouse Overview 11

2.1 The Clearinghouse Platform 11

Chapter 3 Integration Overview 12

3.1 Accessing the Clearinghouse API 12

3.2 SOAP-RPC Server-Client Interactions 12

3.3 SOAPRequest Handler 13

3.4 SOAPResponse Handler - Customer Implementation 14

3.5 Error Handling – SOAP Fault Codes 15

3.6 Queuing 16

3.7 Files for Download 17

Chapter 4 Creating API Clearinghouse Messages 18

4.1 Overview of API XML 18

4.2 XML Restrictions 18

4.2.1 Date 18
4.2.2 Time 18
4.2.3 Date and Time 18
4.2.4 Integer 19
4.2.5 Telephone Number 19
4.2.6 Special Characters 19
4.2.7 Extra Attributes 19

4.3 Clearinghouse Transactions 19

4.3.1 LSR Preorder 20
4.3.2 LSR Order 20
4.3.3 LIDB 22
4.3.4 E911 22
4.3.5 Pre-Port Validation 22
4.3.6 ASR Send Order 22
4.3.7 Intercarrier Communication Process (ICP) 22

4.4 Request Header 23

4.4.1 Header DTD 24
4.4.2 Basic Service Header Value Definitions 24
4.4.2.1 Request and Subrequest 24
4.4.2.2 CustomerIdentifier 24
4.4.2.3 InterfaceVersion 24
4.4.2.4 UserIdentifier and UserPassword 24
4.4.2.5 Supplier 24
4.4.2.6 ApplyBusinessRules 24
4.4.2.7 CustomerUse 25
4.5 Response Header 25
4.6 Notification of GUI Submissions 25
Chapter 5 Basic Services 28
5.1 Basic Services 28
5.1.1 Header XML Structure 28
5.1.1.1 Request and Subrequest Values 28
5.1.1.2 Action Values 29
5.2 LSR Send 30
5.2.1 LSR Send Header XML Elements 30
5.2.2 LSR Preorder Request Structure 31
5.2.3 LSR Preorder DTDs 31
5.2.4 LSR Preorder Response Structure 35
5.2.5 LSR Order Request Structure 35
5.2.6 LSR Order DTDs 36
5.2.7 LSR Order Response Structure 36
5.3 ASR Send 37
5.3.1 ASR Send Header XML Elements 37
5.3.2 ASR Send Request Structure 38
5.3.3 ASR DTDs 38
5.3.4 ASR Send Response 38
5.4 E911 39
5.4.1 E911 Header XML Elements 39
5.4.2 Request Message Structure 39
5.4.3 E911 DTD 40
5.4.4 Example E911 Request 40
5.4.5 E911 Responses 40
5.4.6 Example E911 Response 42
5.5 LIDB 43
5.5.1 LIDB Requests 43
5.5.2 LIDB DTD 43
5.5.3 LIDB Responses 43
5.6 Intercarrier Communication Protocol (ICP) 44
5.6.1 ICP API Request header XML elements 44
5.6.2 ICP DTDs 45
5.6.3 MultiPortRequest 45
5.6.3.1 Parameters – MultiPortRequest 45
5.6.3.2 XML Example - MultiPortRequest 48
5.6.3.3 Flow Charts– MultiPortRequest 49

5.6.4 MultiPortResponse 51
5.6.4.1 Parameters – MultiPortResponse 52
5.6.4.2 XML Example – MultiPortResponse: resolutionRequired 53
5.6.4.3 XML Example – MultiPortResponse: delay 54
5.6.4.4 XML Example – MultiPortResponse: confirm 55
5.6.4.5 Flow Charts - MultiPort Response 56
5.6.5 ModifyPortRequest 59
5.6.5.1 Parameters – ModifyPortRequest 60
5.6.5.2 XML Example – ModifyPortRequest: cancel 62
5.6.5.3 XML Example – ModifyPortRequest: dueDateChange 63
5.6.5.4 XML Example – ModifyPortRequest: modify 64
5.6.6 BroadcastNotification 65
5.6.6.1 Parameters - BroadcastNotification 65
5.6.6.2 XML Example - BroadcastNotification 66
5.6.6.3 Flow Chart – BroadcastNotification 67
5.6.7 TestMessageQuery 67
5.6.7.1 Parameters - TestMessageQuery 67
5.6.7.2 XML Example - TestMessageQuery 67
5.6.7.3 Flow Chart – TestMessageQuery 68
5.6.8 TestMessageResponse 68
5.6.8.1 Parameters - TestMessageResponse 68
5.6.8.2 XML Example - TestMessageResponse 69
5.6.8.3 Flow Chart - TestMessageResponse 69
5.6.9 StoreStatus Message 70
5.6.9.1 Parameters - StoreStatus 70
5.6.9.2 XML Example – StoreStatus: ack 71
5.6.9.3 XML Example – StoreStatus - negack 71
5.6.10 ValidationStatus 72
5.6.10.1 Parameters – ValidationStatus 72
5.6.10.2 XML Example – ValidationStatus: valid 73
5.6.10.3 XML Example – ValidationStatus: invalid 73

5.7 Pre-Port Validation 74
5.7.1 Example Request XML: 74
5.7.2 Example Response XML: 74

5.8 Response Re-flow 75
5.8.1 Request Format 75
5.8.2 Response Format 76

5.9 Provider Notifications 76

5.10 Queries 77
5.10.1 Query Requests 77
5.10.1.1 Query DTD 78
5.10.1.2 The Wildcard Construct 79
5.10.1.3 The “or” Construct 79
5.10.1.4 Additional Data elements 80
5.10.2 Query Responses 93

Chapter 6 LSR Integration Overview 98

6.1 Address Validation 98

6.2 PreOrder Message Flow 98

6.3 LSR Order Message Flow *99*

6.4 EDI Message Transition Sets *100*
6.4.1 EDI Message Transition Set 850 100
6.4.2 EDI Transition Set 855 101
6.4.3 EDI Transition Set 860 101
6.4.4 EDI Transition Set 865 101
6.4.5 EDI Transition Set 997 101

6.5 Forms-based Ordering *102*

6.6 LSR OBF Guidelines *103*
6.6.1 Downloading the OBF Guidelines 104
6.6.2 Purchasing the OBF Guidelines 104
6.6.3 Clearinghouse DTD Structures and OBF Guidelines 104

Appendices 105

Appendix A: XML Examples *105*

Appendix B: Tables *122*

Appendix C: Figures *124*

Chapter 1 About This Guide

The unified API provides a common access method for system access to the Clearinghouse Basic services. This document is a guide for accessing the Clearinghouse using the API to order Local Service (Order and Preorder), Access Service Requests (ASR), E911, LIDB, and SOA services.

1.1 Document History

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

Date	Comments
06/01/2004	Document base-lined
07/28/2004	Created individual chapters for Basic and Managed Services. Updated Basic Services Query section. Added E911, LIDB & SOA information.
8/19/2004	Changed LIDB message types in table 13 to remove LIDB and add LIDBRequest and CallingCardRequest. Added comment to Message key in table 14.
8/25/04	Changed Request column in table 4 from SubscriptionRequest to SOARequest. Changed E911 to E911_Request. Changed E911_request in table 3 to E911_Request. Reformatted table 3.
9/27/04	Added order state information. Added bundled service information. Updated document to reflect current CH API functionality. Tables with volatile data externalized to extranet.
10/22/04	Added asynchronous notifications for E911, LIDB, and SOA.
11/15/04	Added additional information about creating and structuring bundle requests. Added additional subrequest values to Table 5.
12/20/04	Added managed header query information. Updated links to external tables. Added reference to additional XML basic examples.
3/14/05	Re-baselined document version to correspond with Clearinghouse release version. Updated with new document template. Added Section to summarize the changes resulting from the 3.1 upgrade. Added provider notifications (section 5.9), enhanced query information (section 5.10), response re-flow section (section 5.8). Updated basic services header section with 'Action' field info (section 5.1.1).
6/13/05	Added new subrequest: service_order_inquiry to Table 5. Added new DTDs for Service Order Inquiry to Table 10. Added Subrequest SOIP to Table 38, Table 40, Table 42, and Table 45. Updated document to reflect version 3.2 enhancements. Added (section 5.7 Pre Port Validation), added PPV elements to Table 5, Table 38, Table 40, Table 42, and Table 45. Added "Action" as a valid request element for E911 and LIDB in Table 40. Added ProviderNotification, SOAPortin, SOAPortOut, and SOAIntraport to Table 38. Removed Local Manager aspects to make this document specific to Basic Services. Updated query-orders XML example and moved to *Appendix A: XML Examples.* Updated Request Data Elements in Table 40. Added section 4.6 GUI Notification Structure.
9/2/05	Added note on new E911 Trading partner, "The Schofield Group." Added CustomerUse field for LSR Orders. Added information for new Request Type K and new REQTYP E product.

Date	Comments
10/17/05	Added ASR Send elements ***Table 5, section 4.3.6, section 5.3***. and new ServiceTypes for Query API: ASROrder, ASRPreOrder: ***Table 38, Table 40, Table 41, Table 42, and Table 45.***
11/16/05	Added ASR Order and Preorder Response Types
1/10/06	• Added Cancel and Abandon Action values. • Added new Query API subrequest value ***query-transactions 5.10.1.4.4***. • Updated LSR Preorder service type value Resale_private_line. • Added Resale_private_line as a response reflow subrequest value in ***section 5.8.1*** • Added ServiceType values FAI, CFAI, CancelReservation, and ProviderNotification for Query API.
1/16/06	• For query-transactions, added MessageType: “info” • For all Query-Messages subrequest xml elements;changed UserId to USERID, updated Datetime.From and Datetime.To definitions.
1/18/06	Added information for query-order-history-997 ***Table 44***
2/27/06	Listed additional DTD for E911 responses –e911_tsg_response.dtd ***5.4.3***
3/17/06	Updated ASR Order interface version value: ASOG
4/5/06	Added NANC 3.3 changes
4/20/06	Updated for OMS Clearinghouse 3.8 enhancements – LIDB Customer Name, CSR responses, ASR Send User Data
5/26/06	Updated for 3.8.1
8/28/06	Updated for Clearinghouse 3.9 impacts for LSR/ASR Send Please refer to “Whats New in This Release” Chapter.
9/18/06	**Changes for ASOG 33:** • New ASR request Type: EVC • ASR Confirmation message renamed SecondaryConfirmation **General update:** Individual ASR request types are now subrequest values in the ASR Header XML.
2/21/07	Added new Synchronous Failure Response Messages
5/2/07	Added changes for 4.0.1: partial submittal for SOA TN ranges.
November 2007	Removed references to SOA. Changed <SubmittingUserId> to <USERID> for LSR preorder Removed all references to CallingCardRequest
April 2008	Added simple_port value, validated Request Header file, added Fairpoint as supplier, updated VZE/VZW interface version values, and updated links.
July 2008	Added ICP content. Corrected .dtd file names, misc fixes.
September 2008	Fixes per TD 8189
January 2009	Added resale_frame_relay, centrex_resale, and ddps as subrequest types for lsr_order request type in the header file.
February 2009	Section 5.5 - LIDB: Removed TNS restriction on responses.

1.2 Audience

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

1.3 Related Documents

This section lists related reference documentation for use in conjunction with this guide.

- *OMS Clearinghouse Database Schema Basic Services*
- *OMS Clearinghouse Standard Reports User Guide*
- *Additional API Related Resources and all of the above documents are found at https://www.neustar.biz/convergentCH/content/ch_docs_api.html*

1.4 Conventions

The following table lists notational conventions found throughout this document:

Convention	Description	Example
Italics	Denotes the introduction of a term or phrase and that its definition is in the vicinity (either right before or right after).	The *engine* is the order management system.
Constant width	Indicates commands, file names, and file and code samples. Might be emphasized with bold.	% xterm -sb -title osagent
Italics, Constant width	Used within command, code, and file samples. Indicates file names or text to replace with words or names that are appropriate to your installation or environment. Might be emphasized with bold.	% perl copy2web.prl <*directory*> AdminPhn=*Administrator's phone number*
<>	Encloses a directory, file name or other information that needs replacement. The actual name should not be enclosed within angle brackets.	D:\<installation root directory>\ If the installation directory is called 'supplier', replace with: D:\supplier\
Hypertext link	Indicates a hypertext link that, if clicked, takes you to either an HTML page or a URL. A default browser must be specified.	Click here to view the link.
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 Clearinghouse 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
XML Notation	It is NeuStar standard to represent data values via Node attributes named "value".	<DSENT value="11-03-2003-0900AM"/>

1.5 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-RPC and XML.
- Users of this document are connecting to the NeuStar Clearinghouse via the API interface.

1.6 Contacting NeuStar

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://neusupport.neustar.com).

Customers that do not have access to Neusupport should call technical support:
(866) 638-9600, option 2.

Or write to us at:

NeuStar, Inc.
46000 Center Oak Plaza
Sterling, VA 20166

Chapter 2 Clearinghouse Overview

2.1 The Clearinghouse Platform

The Clearinghouse provides an advanced set of capabilities that enable a secure, reliable and scalable environment from which to deliver advanced services to our customers. It allows rapid implementation of new customers and trading partners as well as changes to business rules and workflows. The platform provides transaction routing, guaranteed delivery services, data transformation and support for the conversion and delivery of data to multiple trading partners over multiple protocols. It also supports fallout management services, reports and other service management capabilities.

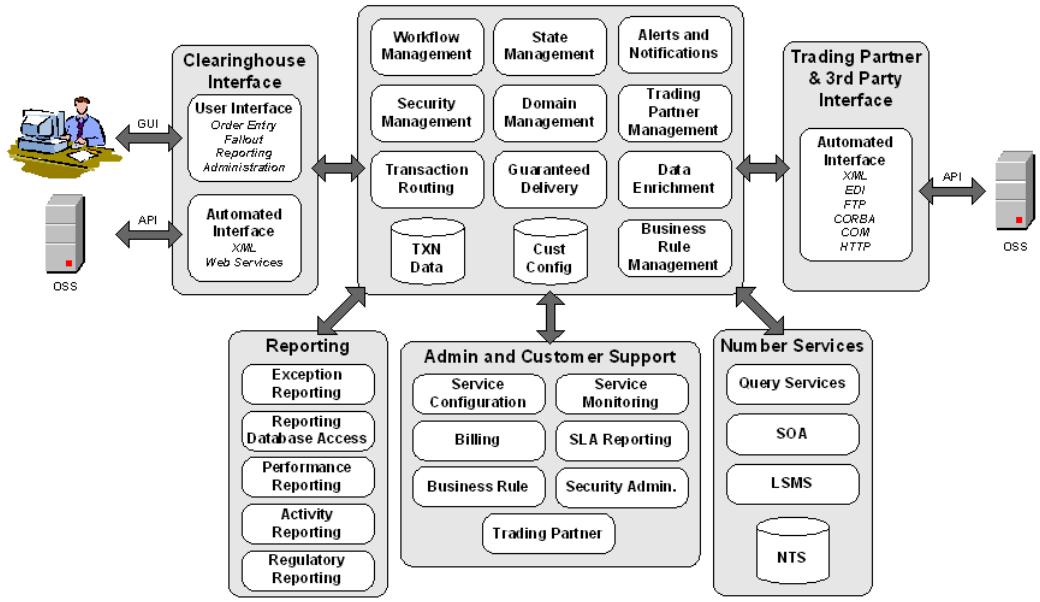


Figure 1: Clearinghouse Platform

Chapter 3 Integration Overview

This chapter provides a description of the SOAP-RPC interface used to communicate with the Clearinghouse. NeuStar currently uses SOAP-RPC version 1.1. SOAP resolves integration problems caused by language and platform dependencies, easily allowing you to integrate with the Clearinghouse.

3.1 Accessing the Clearinghouse API

NeuStar supports SOAP-RPC (Simple Object Access Protocol - Remote Procedure Call) over HTTPS as a means for systems to interact with the Clearinghouse (submitting requests and receiving responses). Systems can communicate directly with the Clearinghouse via properly formatted XML messages sent via SOAP-RPC. A message sent into this interface is referred to as a *request message* and the message returned from this interface is referred to as a *response message*.

The Clearinghouse is accessed either by the SOAP adapter or the Clearinghouse GUI as shown below in ***Figure 2***.

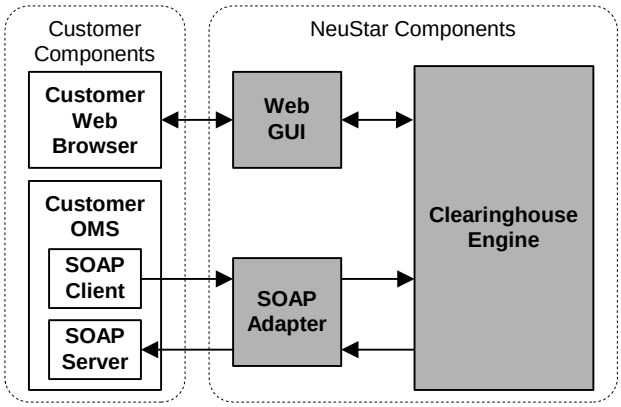


Figure 2: Accessing the Clearinghouse

The NeuStar Clearinghouse allows you to send and receive messages with Trading Partners using a single interface. All messages exchanged between the customer and the Clearinghouse are structured as XML documents.

3.2 SOAP-RPC 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.

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/

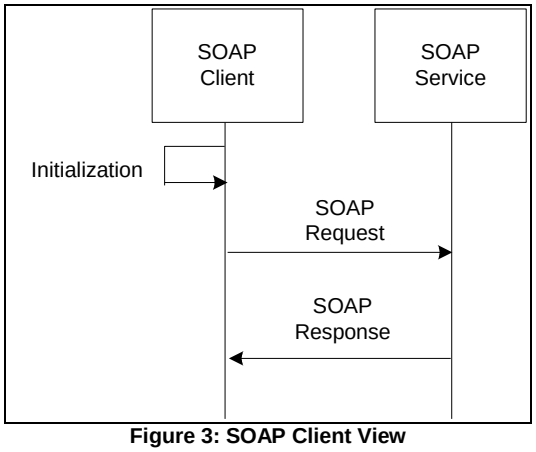


Figure 3: SOAP Client View

3.3 SOAPRequest Handler

The Clearinghouse Web service is named SOAPRequestHandler. External systems can communicate directly with the Clearinghouse with properly formatted XML messages sent via the SOAP-RPC protocol. The detailed structures of these message elements are specific to each request and response type supported.

The SOAPRequestHandler interface is the WSDL interface exposed to SOAP clients. It defines the methods a client invokes to send messages to the Clearinghouse. The WSDL for SOAPRequest handler is:

```
<wsdl:message name="processAsyncRequest">
 <wsdl:part name="in0" type="xsd:string"/>
 <wsdl:part name="in1" type="xsd:string"/>
</wsdl:message>

<wsdl:message name="processSyncResponse">
 <wsdl:part name="processSyncReturn" type="impl:ArrayOf_xsd_string"/>
</wsdl:message>

<wsdl:message name="processSyncRequest">
 <wsdl:part name="in0" type="xsd:string"/>
 <wsdl:part name="in1" type="xsd:string"/>
</wsdl:message>

<wsdl:message name="processAsyncResponse">
</wsdl:message>

<wsdl:portType name="SOAPRequestHandler">
 <wsdl:operation name="processSync" parameterOrder="in0 in1">
  <wsdl:input name="processSyncRequest" message="impl:processSyncRequest"/>
  <wsdl:output name="processSyncResponse" message="impl:processSyncResponse"/>
 </wsdl:operation>
```

```
</wsdl:message>
```

For each method, the first parameter is the request header; an XML structure containing information about the message (see ***Section 4.4***). The second parameter is the request message; an XML structure containing the specific contents of the request. **Table 1** lists the processSync and processAsync invocations used for each of the various Service Types.

Table 1: Basic Service - Request Types

Request Type	Service Type	
processAsync	LSR Order LSR Preorder Address Validation CSR Loop Qualification TN Appt Scheduling	Feature/Service Availability E911 LIDB ASRPreorder ASROrder Wireless Number Port
processSync	Query PPV	

The processSync invocation also returns a String array. The array contains two XML strings: a header and a message.

3.4 SOAPResponse Handler - Customer Implementation

The SOAPResponseHandler interface, shown below, is the WSDL interface which your system must implement to provide the Clearinghouse with a way to return asynchronous responses and notifications to you. It defines the method the Clearinghouse invokes to send messages to your system. The WSDL for SOAPResponseHandler is:

```
<wsdl:message name="processEventRequest">
 <wsdl:part name="in0" type="xsd:string"/>
 <wsdl:part name="in1" type="xsd:string"/>
</wsdl:message>

<wsdl:message name="processEventResponse">
</wsdl:message>

<wsdl:portType name="SOAPResponseHandler">
 <wsdl:operation name="processEvent" parameterOrder="in0 in1">
  <wsdl:input name="processEventRequest" message="impl:processEventRequest"/>
  <wsdl:output name="processEventResponse"
message="impl:processEventResponse"/>
 </wsdl:operation>
</wsdl:portType>
```

For each method, the first parameter indicates the source of the event and the second parameter is the event, in XML format.

Figure 4 describes how asynchronous messages from the Clearinghouse are directed to the SOAPResponseHandler (implemented on your side). This use-case assumes there is a SOAPResponseHandler Web service implemented on your side.

The Clearinghouse provides reliable message delivery from the Clearinghouse to you. This service guarantees message delivery by storing messages in a database until they are successfully delivered.

When the Clearinghouse has an event to deliver to you, the Clearinghouse SOAP client saves and attempts to repeatedly deliver the message to your peer web service until the message is successfully delivered. If the SOAPResponse contains a valid response, the message is removed from the queue. If the SOAPResponse contains a SOAPFault, the queue agent subsequently attempts to resend the message a configurable number of times, effectively repeating this process. This process is shown below in ***Figure 4***.

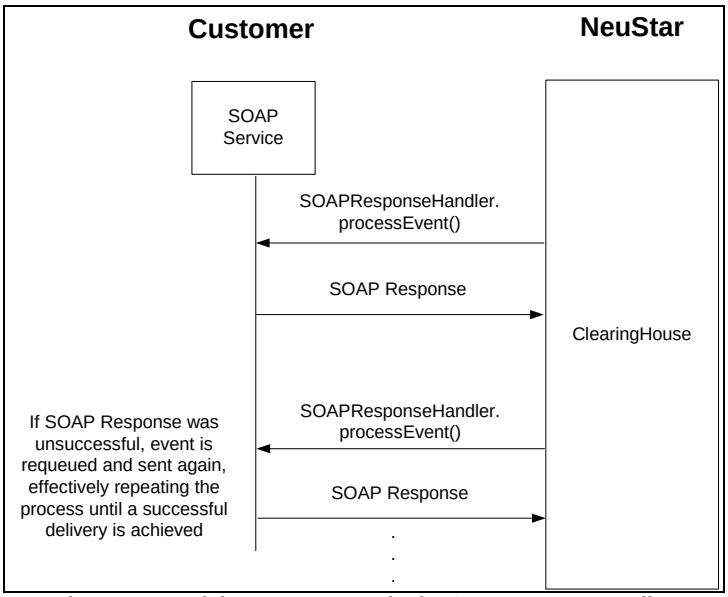


Figure 4: Receiving a response via the SOAPRequestHandler

3.5 Error Handling – SOAP Fault Codes

There are four error codes associated with SOAP fault, they are:

1. SOAP-ENV:Server.MessageException
2. SOAP-ENV:Server.ProcessingException
3. SOAP-ENV:Server.SecurityException
4. SOAP-ENV:Server.userException

If an error occurs, the SOAP fault code displays the cause.

Upon detecting an error in processing, the Clearinghouse's SOAPRequestHandler generates a SOAPException with one of the SOAP fault code values listed in Table 2. The fault code values also appear in the fault string.

Example fault strings:

```
SOAP-ENV:Server.MessageException: Could not parse value for node 'Header'
```

or

```
SOAP-
ENV:Server.ProcessingException:com.nightfire.mgrcore.im.IMProcessingException:
 ERROR: Could not locate context at URL [t3://192.168.8.194:7010]:
```

```
javax.naming.CommunicationException [Root exception is
java.net.ConnectException: t3://192.168.8.194:7010: Destination unreachable;
nested exception is: java.net.ConnectException: Connection refused: connect; No
available router to destination]
```

SOAP clients invoking the SOAPRequestHandler should inspect these fault codes and take the appropriate corrective action.

Table 2: SOAP Fault Codes

SOAP Fault Code	Issue and Corrective Action
SOAP-ENV:Server.MessageException	Problem with the request data: Fix data and resubmit
SOAP-ENV:Server.ProcessingException	Transient system issue: Submit request again
SOAP-ENV:Server.SecurityException	Security violation error: Make sure client is authorized to submit request (check in Security Admin GUI)
SOAP-ENV:Server.userException	Unexpected error: Notify NeuStar admin

When a request is submitted to the Clearinghouse, business rule validation is immediately performed on the input XML. The SOAP Fault Code SOAPENV:Server.MessageException returns business rule errors. All business rule errors are encoded into an XML document included in the SOAP Fault String. The XML document always follows the same format.

An example of a Fault String, which conveys business rule errors:

```
Fault String: SOAP-ENV:Server.MessageException:
<?xml version="1.0"?>
<Errors>
 <ruleerrorcontainer>
  <ruleerror>
   <RULE_ID value="PortResponse_LNUM_2" />
   <MESSAGE value="The valid format for LNUM is NNNNN where N is a
   numeric character." />
      <CONTEXT
      value="/wireless_nport/Body/PortResponse/PortedLineResponsecontainer/Porte
      dLineResponse[1]/LNUM" />
      <CONTEXT_VALUE value="0001" />
  </ruleerror>
  <ruleerror>
   <RULE_ID value="PortResponse_REMARKS_2" />
   <MESSAGE value="REMARKS is required when RCODE = 1P." />
   <CONTEXT value="/wireless_nport/Body/PortResponse/REMARKS" />
   <CONTEXT_VALUE value="" />
  </ruleerror>
 </ruleerrorcontainer>
</Errors>
```

♪ NOTE: SOAP Fault Code errors and business rule errors are returned in synchronous fashion.

3.6 Queuing

The interfaces supported by trading partners (and used by the NeuStar Clearinghouse) occasionally are off-line due to maintenance and operational issues. During these times, the NeuStar Clearinghouse continues to accept SOAP (and GUI) requests from customers. These requests are saved to a database and queued. The orders go through the normal validation and you receive a business rule error, a SOAP Fault Code error or a successful submission

notification. However, no Acknowledgement or response is received, as the order is stored in the database until connectivity is reestablished, at which time the requests are transmitted in their received order.

3.7 Files for Download

A list of business rules, sample XML files, and DTD and schema files can be found on the NeuStar extranet at https://www.neustar.biz/convergentCH/content/cchDownloads.html.

Chapter 4 Creating API Clearinghouse Messages

4.1 Overview of API XML

The API is an Extensible Markup Language (XML) interface that facilitates the integration of the Clearinghouse into a third party system. The API accepts and returns XML messages. Throughout this document, an XML message entering the API from the customer side is called a *request*, and one returning from the supplier side is termed a *response* (regardless of who initiates the exchange).

Messages between customers and trading partners are formatted as XML documents. A reference source for standard XML libraries is provided at http://xml.apache.org.

The structure of XML messages exchanged between a customer and the Clearinghouse is defined in a Document Type Definition (DTD) . The DTD contains the vocabulary and syntax of the document, and specifies a set of rules for the structure of an XML document. The DTD ensures that the Clearinghouse can read the document.

♪ NOTE: The Clearinghouse does not support runtime DTD validation and thus, XML messages should not contain *DOCTYPE* references.

4.2 XML Restrictions

XML requests submitted to the Clearinghouse must adhere to the following restrictions:

4.2.1 Date

All date fields in the U.S. Standard date format: mm-dd-ccyy

- Two Digit Month (01-12)
- Two Digit Day (01-31)
- Two Digit Century (00-99)
- Two Digit Year (00-99)

4.2.2 Time

All time fields in the U.S. Standard format: hhmm[AM|PM] or hhmm[AM|PM]-hhmm[AM|PM]

- Two Digit Hour (01-12)
- Two Digit Minute (00-59)
- AM or PM

4.2.3 Date and Time

All date and time fields in the U.S. Standard format: mm-dd-yyyy-hhmm[AM|PM]. Either lowercase or uppercase variables for AM/PM can be used in the API. Mixed case variables, such as Am or pM, are not permitted, however, and will trigger a business rule validation error and return of the XML.

The following XML illustrates standard date and time field formats:

```
<DTSENT value="02-28-1999-1033AM"/>
```

Clearinghouse date and time fields allow a time range. The following are samples of valid field formats:

- Date only: "11-15-2001"
- Date and time: "11-15-2001-0800AM"
- Date and time range: "11-15-2001-0800AM-1100PM"

```
<DUEDATE>
<From value ="01-03-2006-1200AM"/>
<To value ="01-04-2006-1200AM"/>
</DUEDATE>
```

4.2.4 Integer

All integer fields contain digits.

4.2.5 Telephone Number

Standard format: xxx-xxx-xxxx-xxxx where the first set of digits is the area code, the second set of digits is the exchange, the third set of digits is the number, and the fourth set of digits is the extension. For example, if the telephone number is 510-999-9999-1234, 510 is the area code, 999 is the exchange, 9999 is the number, and 1234 is the extension. The Clearinghouse software is designed and tested to work with numbers from the North American Numbering Plan, therefore, the use of country codes is not allowed.

4.2.6 Special Characters

If any field requires special characters such as &, ‘, “, >, or <, use the corresponding entity shown in ***Table 3*** below. These characters only need replacement when they are embedded values inside the XML, as the XML Syntax uses these characters in its markup.

Table 3: Special characters and corresponding entries

Character	Entity
&	&amp;
‘	&apos;
“	&quot;
>	&gt;
<	&lt;

♪ NOTE: Most XML libraries make these character modifcations for you automatically.

4.2.7 Extra Attributes

Avoid generating “xmlns” attributes as part of XML requests.

4.3 Clearinghouse Transactions

An order request is submitted by the client in order to perform some function on the basic services gateway. This section describes the types of requests supported by the Clearinghouse API.

4.3.1 LSR Preorder

The Clearinghouse can perform Preorder inquiry and response functions that take place prior to the ordering of service. It supports six different LSR Preorder transactions, including:

- **Address Validation (AV)** – provides the ability to validate the address given by the customer. The Trading Partner can indicate if there is a match or if an alternate address exists for you to choose.
- **Loop Qualification (LQ)** – provides the ability to submit an inquiry to determine if facilities support certain loop technology-dependent products prior to placing a firm order.
- **Customer Service Record (CSR)** – queries the Trading Partner on how the existing service is provisioned.
- **Telephone Number (TN) Assignment** – provides the ability to request and reserve a TN(s) and assign that number to an end user.
- **Feature/Service (FS) Availability** –queries the Trading Partner on the availability of specific features, services and PICs for a particular NPA/NXX, TN, ACTL or particular end user address.
- **Appointment (App) Scheduling** – identifies if a dispatch to the end user's location is necessary and establishes the appointment date/time.
- **Collocation Facility Assignment Inquiry (CFAI)** – Provides Collocation assignment information for service at a new location.
- **Fiber Availability Inquiry (FAI)** – Provides information about the facility environment serving the end user. This transaction enables the Clearinghouse user to check the Fiber to the Premise (FTTP) availability after receiving an exact-match Address Validation.
- **Cancel Reservation** – Cancels an entire Pre-Order transaction. Consequently, any due date or telephone number(s) that is reserved under this Pre-Order transaction will be canceled.
- **Service Order Inquiry (SOIP)** – Queries the Trading Partner about a request that has already been submitted.

4.3.2 LSR Order

The Clearinghouse supports the following LSR Order transactions:

- **Loop** - Allows the LSP to lease the equipment that provides local service to the customer premises or a sub-loop location from a Trading Partner. The loop is commonly referred to as the "last mile".
- **Loop with NP** - Allows you to request loop service in conjunction with number portability.

- **Number Port** - Allows you to request number portability in order to provide the customer the ability to change local service providers for basic local exchange service without changing their telephone number.

- **Resale –** A request to purchase services from a Network Service Provider to provide to the customer by the Local Service Provider.

- **Resale Digital Service 1** - A request to purchase a digital services line from a Network Service Provider to provide to the customer by the Local Service Provider.

- **Resale_private_line** – A request to purchase a private line from a Network Service Provider to provide to the customer by the Local Service Provider.

- **Resale Private Line Digital Service 1** - A request to purchase a private digital services line from a Network Service Provider to provide to the customer by the Local Service Provider.

- **Port** - Allows you to request an Unbundled Network Element, which represents the capability derived from the central office switch required to permit customers to transmit and receive information over the NSPs public switched network.

- **Directory Listings** - This changes or removes the directory listing information associated with a customer and provides accurate contact information that directs their LSP to process requests for changes, additions, or deletions to white pages.

- **Directory Listings/Assistance** - This changes or removes the directory listing information associated with a customer and provides accurate contact information that directs their LSP to process requests for changes, additions, or deletions to white pages and directory assistance.

- **Platform (Port & Loop)** - Allows you to request loop service in conjunction with port service.

- **ISDN BRI/PRI** – Integrated Services Digital Network – Basic Rate Interface and Primary Rate Interface. VZE and VZW only. Request of high bandwidth switched network service providing end-to-end digital connectivity over standard phone lines for simultaneous transmission of voice and data. BRI involves a 2-wire electrical interface to a local switch. PRI involves a DS1 electrical interface to a local switch.

- **Simple_Port** – involves an account only for a single line, Does not involve unbundled network elements, complex switch transactions, or a reseller.

- **Centrex Resale** - Requests a system that provides access lines and call management features.

- **Resale Frame Relay** - Resale of "fast packet" network services that permit the transmission of data at high speeds.

- **DID/DOD/PBX Service** - Services that permit incoming and out going call through a Private Branch Exchange extension, thereby bypassing the attendant or switchboard operator.
  - DID = Direct Inward Dialing

- DOD = Direct Outward Dialing
- PBX = Private Branch Exchange Services

4.3.3 LIDB

This service adds, changes or removes the blocking and billing information associated with a telephone number. The Clearinghouse supports subcriber line inserts, changes and deletions.

4.3.4 E911

This service changes or removes the emergency listing information associated with a customer's telephone number.

4.3.5 Pre-Port Validation

This service allows a wireless provider to check whether they can provide service for a TN before it is ported.

4.3.6 ASR Send Order

An Access Service Request allows you to order and provision access service. The following request types are supported:

- **Transport** - Used for ordering television, broadband, high capacity, voice, analog, digital and optical services.
- **Feature Group A** - Used for ordering a line side connection from a customer ACTL to an ILEC Central Office with an associated 7-digit local telephone number. Provides dial tone service either interLATA or interSTATE.
- **Trunking** - Used for establishing an end-to-end connection between two switching systems in a network.
- **WATS Access Line** - Used for specifying the premises at which the WATS Access Line (WAL) terminates, the type of line (two-wire or four-wire), the type of calling (originating, terminating or two way) and the type of supervisory signaling.
- **Ring Service** - A self-healing configuration by which a collection of nodes forms a closed-end loop. Each node is connected by means of a fiber facility. Is generally ordered between two or more locations identified sequentially by segments on the forms.
- **Virtual Connection** - Broadband services. Involves both physical ports and logical circuits.
- **Ethernet Virtual Connection** - An association of two or more User Network Interfaces (UNI) that limits the exchange of Service Frames to those UNIs.

4.3.7 Intercarrier Communication Process (ICP)

ICP requests are the means by which telephone numbers can be ported between two wireless carriers. Like other transactions processed in the Clearinghouse, ICP utilizes the SOAP/XML protocol. ICP transactions in the Clearinghouse conform to the requirements of WICIS 4.0.

The web service that runs on NeuStar's ICP Receive Web Service Box is a PUSH type. Trading partners post messages to this web service. The web service performs the following tasks: receives UOM Message, checks for "CORRELATION_ID exists" validation, checks for "Unauthorized Trading Partner" validation, queues incoming messages, prepares Store Status response, and sends Store Status response back to the TP.

There is one ICP request type:

- **wireless_nport**

The following transactions are supported for each request type:

- **MultiPortRequest** - The New Network Service Provider (NNSP) submits a request to the Old Network Service Provider (ONSP) for a wireless telephone number to be ported. The MultiPortRequest can be for a single telephone number (TN), multiple TNs, or a range of TNs.
- **MultiPortResponse** – The ONSP responds to the MultiPortRequest received from the NNSP.
- **ModifyPortRequest** – The NNSP submits a request to modify an existing MultiPortRequest.
- **BroadcastNotification** - Message sent by either the ONSP or the NNSP to indicate that its system may not be available at a particular point in time or to share operational information related to porting activities.
- **TestMessageQuery** - Sent by a service provider's system to a trading partner, to query for information about the ability of the trading partner's system to accept requests and responses. This is a high level test for connectivity and service provider system status.
- **TestMessageResponse** – Sent by a service provider's system in response to receipt of a TestMessageQuery. The Test MessageResponse provide system status information about the service provider's system.

The following messages are supported:

- **StoreStatus** - The intended recipient system (either the NNSP or the ONSP, depending on the message type), generates a positive or negative acknowledgment that a message has been received.
- **ValidationStatus** - The intended recipient system (either the NNSP or the ONSP, depending on the message type), indicates that the received message meets the structure and business rule requirements of WICIS 4.0.

4.4 Request Header

When a request is submitted to the Clearinghouse via the API, the request header dictates the type of operation performed on the request message. The header XML for the request sent to the Gateway is constructed by the customer.

The following XML example illustrates the format of a request header XML document.

```
<header>
 <Request value="lsr_order"/>
 <Subrequest value="loop"/>
 <CustomerIdentifier value="ACME"/>
 <InterfaceVersion value="15_0"/>
 <Supplier value="BS"/>
 <UserIdentifier value="Fred"/>
 <UserPassword value="example"/>
 <ApplyBusinessRules value="Y"/>
 <CustomerUse value="Test"/>
 <Action value="submit"/>
</header>
```

4.4.1 Header DTD

NeuStar header XML is defined in DTD files. These are found and downloaded at https://www.neustar.biz/convergentCH/content/ch_dl_header.html.The following DTD contains the vocabulary and syntax of NeuStar's XML documents for Header requests for Basic Services.

- **CH-basic-service-header.dtd**

4.4.2 Basic Service Header Value Definitions

4.4.2.1 Request and Subrequest

The valid values for Request and Subrequest determine the Basic Service requested. Refer to https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf for appropriate valid values.

4.4.2.2 CustomerIdentifier

The CustomerIdentifier value corresponds to your Domain. This value is provided by your NeuStar account representative.

4.4.2.3 InterfaceVersion

The InterfaceVersion defines the version of the interface to which a request is submitted. Refer to https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf for appropriate valid values.

4.4.2.4 UserIdentifier and UserPassword

The UserIdentifier and UserPassword values correspond to your Username and Password. These values are provided by your NeuStar account representative. Additionally, you may provision your own username/password via the User Admin functionality.

4.4.2.5 Supplier

The Supplier value defines the trading partner to whom a request is submitted. Refer to https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf for appropriate valid values.

4.4.2.6 ApplyBusinessRules

The ApplyBusinessRules field defines whether the Clearinghouse should apply business rules prior to submitting the message to the trading partner. This is typically set to “Y” (yes). It is only set to “N” (no) when operational necessities dictate the disabling of Clearinghouse business rules to allow specific transactions to flow to a trading partner. If this field is not included, the default is set to apply business rules.

4.4.2.7 CustomerUse

The CustomerUse field allows you to enter free flowing information that is not constrained by LSOG business rules for an LSR Order. You can use this field to search for an order by an non-LSOG attribute.

4.5 Response Header

The response header is only returned from a processSync() invocation. The information in the request header is returned as-is in the response header. The following is a response header for an ASR request.

```
<?xml version="1.0" encoding="UTF-8"?>
<header>
 <Request value="asr_order"/>
 <Subrequest value="Transport"/>
 <CustomerIdentifier value="BASIC_QA"/>
 <InterfaceVersion value="ASOG"/>
 <Supplier value="BS"/>
 <UserIdentifier value="hxu"/>
 <UserPassword value="hxu"/>
 <ApplyBusinessRules value="N"/>
 <Action value="save"/>
<InputSource value="A"/>
<SubDomainId value=""/>
</header>
```

4.6 Notification of GUI Submissions

GUI Notifications are generated and sent via the SOAP API when a transaction is submitted via the CH Basic GUI. These are sent on the NeuStar.Clearinghouse.EventChannel_<Customerid> EventChannel.

Contained within the <Notification> section is a <Header> and <Body>. The <Header> section contains two field values; EventType and Event. Values for these are always the same as noted below:

Field	Value
EventType	status
Event	message-sent

After the <Header> section of the notification, there is a <Body> section. This section contains the details for the transaction submitted in the CH Basic GUI. The <Header> as defined in the **CH-basic-service-header.dtd** file and the <request> as defined in the CH Basic DTD files, is placed inside the <Body> container.

The following table illustrates the definition of the notification.

Table 4: Clearinghouse Basic GUI Notification XML structure.

<Notification>	
<Header>	
<EventType value="status" />	
<Event value="message-sent" />	
</Header>	
<Body>	
<Header>	
…	Defined in CH-basic-service-header.dtd. ♪ NOTE: The UserPassword data element is not included in the notification.
</Header>	
<Message>	
…	Defined in the various CH Basic dtd files available on the NeuStar Extranet.
</Message>	
</Body>	
</Notification>	

Example CH Basic GUI Notification XML:

```
<?xml version="1.0" encoding="UTF-8"?>

<Notification>
 <Header>
  <EventType value="status" />
  <Event value="message-sent" />
 </Header>
 <Body>
  <Header>
   <Request value="lsr_order" />
   <Subrequest value="port_loop" />
   <Supplier value="QWEST" />
   <InterfaceVersion value="IMA16" />
   <ApplyBusinessRules value="Y" />
   <CustomerIdentifier value="MYCH_E2E" />
   <UserIdentifier value="mychuser" />
  </Header>
  <Message>
   <Request>
    <lsr_order>
     <lsr>
      <lsr_adminsection>
       <DTSENT value="05-11-2005-0453PM" />
       <REQTYP value="MB" />
       <AGAUTH value="Y" />
       <PON value="UNEPOTSORDER" />
       <ACT value="W" />
       <DATED value="05-01-2005" />
       <AN value="360-457-0200" />
       <CCNA value="R17" />
       <AUTHNM value="JANE DOE" />
       <DDD value="05-13-2005" />
       <RTR value="C" />
       <TOS value="1" />
      </lsr_adminsection>
      <contactsection>
       <EMAIL value="JANE.DOE@NEUSTAR.BIZ" />
       <INIT value="JANE DOE" />
       <TELNO value="571-434-1000" />
```

```
       <IMPCON>
        <NAME value="JOHN SMITH" />
        <TELNO value="571-434-1000" />
       </IMPCON>
      </contactsection>
     </lsr>
    </lsr_order>
   </Request>
  </Message>
 </Body>
</Notification>
```

Chapter 5 Basic Services

Basic Services require you to submit requests directly to the Gateways located in the Clearinghouse. This chapter provides specific details on sending Basic Service Requests and receiving Basic Service Response messages.

5.1 Basic Services

The Basic Services are single transaction processing requests. They include:

- Local Service Preorder (Address Validation, Customer Service Record, etc)
- Local Service Order (LSR)
- Access Service Request Preorder
- Access Service Request (ASR)
- E911
- Line Information Database (LIDB)
- Pre-Port Validation (PPV)
- Intercarrier Communication Process (ICP)

5.1.1 Header XML Structure

The following is an example section of the Basic Services header XMLRequest type. You may download a sample of it from the NeuStar extranet site:
https://www.neustar.biz/convergentCH/content/ch_dl_header.html

```
<header>
    <Request value="lsr_order"/>
</header>

or

<header>
    <Request value="lsr_preorder"/>
<header>
```

5.1.1.1 Request and Subrequest Values

The **Request** and **Subrequest** values define the type of request performed. ***Table 5*** defines valid values for Request and Subrequest in the request header for Basic Services. All requests are invoked via the processAsync() SOAP call. Business Rule errors are returned synchronously. Responses are returned in an asynchronous fashion.

Table 5: Request and Subrequest Values

Gateway Type	Request	Subrequest	
LSR Preorder	lsr_preorder	address_validation loop_qualification csr service_order_inquiry cancel_reservation	TN_assignment FS_availability appointment_scheduling CFA_inquiry fiber_availability
LSR Order	lsr_order	loop np_order loop_with_np dir_assist port port_loop centrex_resale	resale resale_private_line dir_list dir_assist_list simple_port resale_frame_relay ddps

Gateway Type	Request	Subrequest
ASR Preorder	asr_preorder	asr_preorder
ASR Order	asr_order	Transport – use for E-End User Special Access and S-Special Access Trunking – use for L-CCS Link or Unbundled STP Port and M-Trunking FGA – use for A-Switched Access WATS – use for W-WATS Access Line Ring – use for R-Ring VC – use for V-Broadband services and X-Broadband End User Services EVC – use for S-Ethernet Virtual Connection
E911	E911_Request	ServiceOrder
LIDB	Subscriber_Record_Request	LIDBRequest ChangeTNRecord InsertTNRecord DeleteTNRecord
Pre-Port Validation	PrePortValidation	validate
ICP	wireless_nport	MultiPortRequest MultiPortResponse ModifyPortRequest BroadcastNotification TestMessageQuery TestMessageResponse ValidationStatus
Query	query-messages	query-orders query-order-history query-transaction-details

5.1.1.2 Action Values

An **Action** field was added to the header to allow for additional API functionality. ***Table 6*** shows the valid values for the Action field:

Table 6: Valid Values for the Action Field

Field	Valid Values	Description
Action	submit	The 'submit' action value is the standard action of submitting an order to the Clearinghouse. In previous versions of the Clearinghouse, this was the only valid action performed. ♪ NOTE: If the **Action** field is not present in the header, then the default behavior is 'submit'.
	save	The 'save' action value provides the ability to save an LSR Order, LSR PreOrder, E911, or LIDB request in the Clearinghouse. From here, you may work the order via the Clearinghouse GUI. ♪ NOTE: When attempting to save the order via the API, the system checks to ensure that all of the required fields are populated. If all of the required fields are not populated, you receive an error indicating that the order cannot be saved in its current state. ♪ NOTE: If you attempt to re-save an order via the API, the order is saved with the information contained in your save request. Any changes made to the order via the GUI that are not made to your upstream system are not included in the saved record.
	validate	The 'validate' action value allows you to validate API requests without submitting the request for processing. Specifically, using the business rules contained in the Clearinghouse, an API request is validated to ensure that it complies with these rules, reducing the possibility that the order is rejected due to an error upon submission.
	suspend	The 'suspend' action value prevents a particular order from appearing in a worklist. Specifically, when you set up a worklist in the Clearinghouse, you can specify that orders with a value of 'suspend' are omitted. This allows you to remove an order that is not workable in its current state. Not used with ICP

Field	Valid Values	Description
	resume	The 'resume' action value returns a previously suspended order to the worklist search results. Not used with ICP
	abandon	The 'abandon' action value allows you to declare that an order is no longer valid. If an abandoned order receives a response, the order will be moved into the appropriate state. Not used with ICP
	cancel	The 'cancel' action value allows you to declare that an order is no longer valid. If a cancelled order receives a response, the order will be moved into the appropriate state. Not used with ICP

♪ NOTE: When the save, validate, suspend, or resume actions are used, the API call must use the processSync() method.

♪ NOTE: The submit, save, suspend, resume, cancel and abandon requests are logged in the Clearinghouse database and become part of the order history. The Validate action is the only action that is not logged in the database and thus, does not appear in the order history.

♪ NOTE: XML example for suspend and resume action values:

```
<?xml version="1.0" encoding="UTF-8" ?>

<Body>
   <Info>
     <Datetime value="06-09-2005-0755PM" />
     <ACT value="N" />
     <ServiceType value="Loop" />
     <MessageType value="info" />
     <PON value="PON003" />
     <Locked value="false" />
  </Info>
</Body>
```

5.2 LSR Send

This section defines the request and response structures for LSR Order and Preorder.

5.2.1 LSR Send Header XML Elements

The following table describes LSR order and Preorder header XML elements.

Table 7: LSR Order/PreOrder Header XML elements

Header Node Name	Valid Values	Notes
Request	lsr_order lsr_preorder	Required
Subrequest	See Table 5: Request and Subrequest Values	Required
Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	Required
InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-	

Header Node Name	Valid Values	Notes
	_Supplier_and_Interface_Version_Valid_Values.pdf	
ApplyBusinessRules	Y (default) N	Optional, default value is "Y".
UserIdentifier		
CustomerIdentifier		
UserPassword		Required
CustomerUse		Optional. Used to store textual comment with the order
Action	save validate submit suspend resume abandon cancel	Optional, default value is "submit."

5.2.2 LSR Preorder Request Structure

The ordering interfaces to different suppliers vary. Fields required by one supplier are often not supported by another. In order to define a programming API into the Clearinghouse that remains constant for a given client while supporting ordering interfaces to all suppliers, a request XML document must have two parts. The first part contains a generic set of fields, which are consistent across all suppliers for a given service type. The second part contains fields which are specific to the target supplier of the request. In this way, the supplier interfaces are maintained and released independently, without any interdependencies among the various supplier interface components of the gateway.

5.2.3 LSR Preorder DTDs

Table 8 illustrates how the LSR Preorder DTDs define the structure of requests and responses.

Table 8: LSR PreOrder Request Message XML Structure

<Request>	
<lsr_preorder> … </lsr_preorder>	Defined in the generic_<service>_request.dtd
<SupplierLSRPreorderRequest> … </SupplierLSRPreorderRequest>	Defined in the <supplier>_<service>_request.dtd
</Request>	

Below, ***Table 10*** and ***Table 11*** contain a complete list of the DTD files which define the structure of LSR Preorder XML messages. Copies of these files are on https://www.neustar.biz/convergentCH/content/ch_dl_dtd.html.

Table 9: LSR Preorder XML DTD files (Generic Response)

Request Type	Generic Response
Supplier	
All	PO_Preorder_Generic13_response.dtd

Table 10: LSR Preorder XML DTD files (AV, CSR, LoopQual, Service Order Inquiry)

Request Type	Address Validation	CSR	Loop Qualification	Service Order Inquiry
Supplier	**DTD Files**			
Generic	Generic AV request.dtd Generic AV response.dtd PO_AV_generic13_request.dtd PO_AV_generic13_response.dtd	Generic CSR request.dtd Generic CSR response.dtd PO_CSR_generic13_request.dtd PO_CSR_generic13_response.dtd	Generic LoopQual request.dtd Generic LoopQual response.dtd PO_LQ_generic13_request.dtd PO_LQ_generic13_response.dtd	Generic SOIP request.dtd Generic SOIP response.dtd PO_SOIP_generic13_request.dtd PO_SOIP_generic13_response.dtd
AT&T SE (formerly BS)	PO_AV_ATT_SE_LSOG6_request.dtd PO_AV_ATT_SE_LSOG6_response	PO_CSR_ATT_SE_LSOG6_request.dtd PO_CSR_ATT_SE_LSOG6_response.dtd	PO_LQ_ATT_SE_LSOG6_request.dtd PO_LQ_ATT_SE_LSOG6_response.dtd	
VZE	PO_AV_VZEt_LSOG9_request.dtd PO_AV_VZE_LSOG9_response.dtd	PO_CSR_VZE_LSOG9_request.dtd PO_CSR_VZE_LSOG9_response.dtd	PO_LQ_VZE_LSOG9_request.dtd PO_LQ_VZE_LSOG9_response.dtd	PO_SOIP_VZE_LSOG9_request.dtd PO_SOIP_VZE_LSOG9_response.dtd
VZW	n/a PO_AV_VZW_LSOG9_response.dtd	PO_CSR_VZW_LSOG9_request.dtd PO_CSR_VZW_LSOG9_response.dtd	PO_LQ_VZW_LSOG9_request.dtd PO_LQ_VZW_LSOG9_response.dtd	
QWEST	PO_AV_Qwest_request.dtd PO_AV_Qwest_response.dtd	PO_CSR_Qwest_request.dtd PO_CSR_Qwest_response.dtd	PO_LQ_Qwest_request.dtd PO_LQ_Qwest_response.dtd	
AT&T (formerly SBC)	PO_AV_ATT_LSOG_9_Request.dtd PO_AV_ATT_LSOG_9_Response.dtd	PO_CSR_ATT_LSOG_9_Request.dtd PO_CSR_ATT_LSOG_9_Response.dtd	PO_LQ_ATT_LSOG_9_Request.dtd PO_LQ_ATT_LSOG_9_Response.dtd	PO_SOIP_ATT_LSOG_9_Request.dtd PO_SOIP_ATT_LSOG_9_Response.dtd
Fairpoint	FPC_av_request.dtd FPC_av_response.dtd	FPC_csr_request.dtd FPC_csr_response.dtd	FPC_loopqual_request.dtd FPC_loopqual_response.dtd	FPC_soip_request.dtd FPC_soip_response.dtd

Table 11: LSR Preorder XML DTDs (TN, Appt Scheduling, FS)

Request Type	Telephone Number Selection/Reservation	Appointment Scheduling	Feature/Service Availability
Supplier	**DTD Files**		
Generic	Generic TN request.dtd Generic TN response.dtd PO_TN_Generic13_request.dtd PO_TN_Generic13_response.dtd	Generic APPT request.dtd PO_APPT_Generic13_request.dtd Generic APPT response.dtd PO_APPT_Generic13_response.dtd	Generic FS request.dtd Generic FS response.dtd PO_FA_Generic13 request.dtd PO_FA_Generic13 response.dtd PO_FS_Generic13 request.dtd PO_FS_Generic13 response.dtd PO_SVC_Generic13_request.dtd PO_SVC_Generic13_response.dtd
AT&T SE (formerly BS)	PO_TAF_ATT_SE_LSOG6_request.dtd PO_TAF_ATT_SE_LSOG6_response.dtd	PO_TAF_ATT_SE_LSOG6_request.dtd PO_TAF_ATT_SE_LSOG6_response.dtd	PO_TAF_ATT_SE_LSOG6_request.dtd PO_TAF_ATT_SE_LSOG6_response.dtd
VZE	PO_TAF_VZE_LSOG9_request.dtd PO_TAF_VZE_LSOG9_response.dtd	PO_TAF_VZE_LSOG9_request.dtd PO_TAF_VZE_LSOG9_response.dtd	PO_TAF_VZE_LSOG9_request.dtd PO_TAF_VZE_LSOG9_response.dtd
VZW	PO_TAF_VZW_LSOG9_request.dtd PO_TAF_VZW_LSOG9_reponse.dtd	PO_TAF_VZW_LSOG9_request.dtd PO_TAF_VZW_LSOG9_reponse.dtd	PO_TAF_VZW_LSOG9_request.dtd PO_TAF_VZW_LSOG9_reponse.dtd
QWEST	PO_TAF_Qwest_request.dtd PO_TAF_Qwest_response.dtd	PO_TAF_Qwest_request.dtd Qwest/PO_TAF_Qwest_response.dtd	PO_TAF_Qwest_request.dtd Qwest/PO_TAF_Qwest_response.dtd
AT&T (formerly SBC)	PO_TAF_ATT_ LSOG9_request.dtd PO_TAF_ATT_ LSOG9_response.dtd	PO_TAF_ATT_ LSOG9_request.dtd PO_TAF_ATT_ LSOG9_response.dtd	PO_TAF_ATT_ LSOG9_request.dtd PO_TAF_ATT_ LSOG9_response.dtd
Fairpoint	FPC lsr_preorder_request.dtd FPC lsr_preorder_response.dtd	FPC lsr_preorder_request.dtd FPC lsr_preorder_response.dtd	FPC lsr_preorder_request.dtd FPC lsr_preorder_response.dtd

Table 12: LSR Preorder XML DTDs (Cancel Reservation, CFA)

Request Type	Cancel Reservation	Collocation/Facility Assignment
Supplier		
Generic	PO_CANCEL_Generic13_request.dtd PO_CANCEL_Generic13_response.dtd	PO_CFA_Generic13 request.dtd PO_CFA_Generic13 response.dtd

5.2.4 LSR Preorder Response Structure

The messages delivered upon receipt of an asynchronous response from a trading partner contain one XML string consisting of a generic part and a supplier-specific part. While these messages always contain a generic and a supplier part, the contents of the supplier part are sometimes empty.

Table 13: LSR Preorder Response Message XML Structure

<Response>	
<lsr_preorder_response> … </lsr_preorder_response>	Defined in generic_<service>_response.dtd
<SupplierLSRPreorderResponse> … </SupplierLSRPreorderResponse>	Defined in <supplier>_<service>_response.dtd
</Response>	

The following table lists response type values for LSR Preorder.

Table 14: LSR Preorder Response Type Values

Response type Values	
LSR Preorder	address_validation (LSOG9-11 only) csr loop_qualification appointment_scheduling cancel_reservation CFA_inquiry fiber_availability FS_availability service_order_inquiry TN_assignment

♪ NOTE: The string values returned in the eventChannelName parameter of the processEvent() method are shown at https://www.neustar.biz/convergentCH/content/docs/EventChannelName_Valid_Values_and_Descriptions.pdf. This value identifies the Basic Service which is the source of asynchronous notifications and responses when they are sent to you via the processEvent() method of the SOAPResponseHandler interface. A 'message-sent' event is posted to your SOAPResponse handler each time a request to a gateway is initiated from the Basic Services GUI, regardless of the outcome of the request.

5.2.5 LSR Order Request Structure

Table 15 illustrates how the LSR Ordering DTDs define the structure of requests.

Table 15: LSR Order Request Structure

<Request>	
<lsr_order> … </lsr_order>	Defined in the generic_lsr_request.dtd
<SupplierLSROrderRequest> … </SupplierLSROrderRequest>	Defined in the <supplier>_lsr_request.dtd
</Request>	

You may download samples of the basic lsr_order XML request from the NeuStar extranet site: https://www.neustar.biz/convergentCH/content/ch_dl_xml.html.

5.2.6 LSR Order DTDs

The following table lists the DTD files which define the structure of LSR Order XML messages for Loop, LSNP, NP, DL, DS, Resale, Resale_private_line, Port, Platform, simple port (SPSR), and ISDN BRI/PRI (IS). Copies of these files are on https://www.neustar.biz/convergentCH/content/ch_dl_dtd.html.

Table 16: LSR Order XML DTD files

Supplier DTD	DTD File
Generic	LSR_Generic9-13_request.dtd Generic LSR request (L56).dtd Generic LSR response (L56).dtd LSR_Generic13_response.dtd
AT&T SE (formerly BellSouth)	LSR_ATT_SE_LSOG6_request.dtd LSR_ATT_SE_LSOG6_response.dtd
VZE	LSR_VZE_LSOG9_request.dtd LSR_VZE_LSOG9_response.dtd
VZW	LSR_VZW_LSOG9_request.dtd LSR_VZW_LSOG9_response.dtd
QWEST	LSR_Qwest_IMA_request.dtd LSR_Qwest_IMA_response.dtd
AT&T (formerly SBC)	LSR_ATT_LSOG_9_request.dtd LSR_ATT_LSOG_9_response.dtd
SPRINT	Sprint SLOG8.0 - request.dtd Sprint SLOG8.0 - response.dtd
Fairpoint	FPC_lsr_request.dtd FPC_lsr_response.dtd

5.2.7 LSR Order Response Structure

The messages delivered upon receipt of an asynchronous response from a trading partner contain one XML string, consisting of a generic part and a supplier-specific part. While these messages always contain a generic and a supplier part, the contents of the supplier part are sometimes empty.

Table 17: LSR Order Response Message XML Structure

`<Response>`	

`<lsr_order_response>`
…
`</lsr_order_response>`
	Defined in generic_lsr_response.dtd

`<SupplierLSROrderResponse>`
…
`</SupplierLSROrderResponse>`
	Defined in <supplier>_lsr_response.dtd

`</Response>`	

The following table lists response type values for LSR Order.

Table 18: LSR Order Response Type Values

Response type Values	
LSR Order	ack negack soc billing_completion focaccept focreject status suppaccept suppreject provider_initiated_activity dsrcn dsred jeopardy provider_notification provider_initiated_cancel (LSOG 11 only) fax_ack fax_nack

♪ NOTE: The string values returned in the eventChannelName parameter of the processEvent() method are shown at https://www.neustar.biz/convergentCH/content/docs/EventChannelName_Valid_Values_and_Descriptions.pdf. This value identifies the Basic Service which is the source of asynchronous notifications and responses when they are sent to you via the processEvent() method of the SOAPResponseHandler interface. A 'message-sent' event is posted to your SOAPResponse handler each time a request to a gateway is initiated from the Basic Services GUI, regardless of the outcome of the request.

5.3 ASR Send

The Clearinghouse allows you to submit Access Service Request orders and preorders and it returns asynchronous responses. This section defines the request and response structures for ASR orders and preorders.

5.3.1 ASR Send Header XML Elements

The following table describes ASR order and Preorder header XML elements.

Table 19: ASR Order/PreOrder Header XML elements

Header Node Name	Valid Values	Notes
Request	asr_order asr_preorder	Required
Subrequest	Transport – use for E-End User Special Access and S-Special Access Trunking – use for L-CCS Link or Unbundled STP Port and M-Trunking FGA – use for A-Switched Access WATS – use for W-WATS Access Line Ring – use for R-Ring VC – use for V-Broadband services and X-Broadband End User Services EVC – use for S-Ethernet Virtual Connection asr_preorder	Required
Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	Required
InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	
ApplyBusinessRules	Y (default) N	Optional, default value is "Y".
UserIdentifier		
CustomerIdentifier		

UserPassword		Required
CustomerUse		Optional. Used to store textual comment with the order
Action	save validate submit suspend resume abandon cancel	Optional, default value is “submit.”

5.3.2 ASR Send Request Structure

The following table illustrates how the ASR ordering DTDs define the structure of requests.

Table 20: ASR Order/PreOrder Request structure

<Request>	
<asr_order> … </asr_order>	Defined in the asr_request.dtd
<asr_preorder> … </asr_preorder>	Defined in the asr_preorder_request.dtd
</Request>	

5.3.3 ASR DTDs

ASR Order and preorder requests xml structure is defined in the following DTDs, located at: https://www.neustar.biz/convergentCH/content/ch_dl_dtd.html

- **Preorder_ASR_ Request.dtd**
- **Preorder_ASR_ Response.dtd**
- **Order_ASR_Request.dtd**
- **Order_ASR_ Response.dtd**

5.3.4 ASR Send Response

The following table illustrates how the ASR response DTDs define the structure of responses.

Table 21: ASR Order/PreOrder Response Structure

<Response>	
<asr_order_response> … </asr_order_response>	Defined in the asr_response.dtd
<asr_preorder_response> … </asr_preorder_response>	Defined in the Preorder_ASR_ Response.dtd
</Response>	

The following table lists response type values for ASR Order and Preorder.

Table 22: ASR Order/ Preorder Response Type Values

	Response type Values
ASR Order	ack nack error secondaryconfirmation confirmation clarification dlr
ASR Preorder	CFAResults LocationResults SvcAvailResults

♪ NOTE: The string values returned in the eventChannelName parameter of the processEvent() method are shown at https://www.neustar.biz/convergentCH/content/docs/EventChannelName_Valid_Values_and_Descriptions.pdf. This value identifies the Basic Service which is the source of asynchronous notifications and responses when they are sent to you via the processEvent() method of the SOAPResponseHandler interface. A 'message-sent' event is posted to your SOAPResponse handler each time a request to a gateway is initiated from the Basic Services GUI, regardless of the outcome of the request.

5.4 E911

The Clearinghouse allows you to provide E911 service for access to emergency service providers by dialing 911. You may use the API to submit E911 single order requests to perform the following functions on the gateway:

- Insert an entry into the database
- Change an existing database entry
- Delete an existing database entry

♪ NOTE: There are two trading partners for E911, Intrado and The Schofield Group. You are assigned to one of them by NeuStar during the deployment stage. All E911functions are the same, regardless of which trading partner is assigned.

5.4.1 E911 Header XML Elements

The E911 request header contains control information such as an order number (assigned by you), the identity of the sender, and the submission time of the request.

5.4.2 Request Message Structure

The request body consists of the Service Order information, which includes customer address information.

The following table illustrates how the E911 ordering DTDs define the structure of requests.

Table 23: E911 Request structure

<Request>	
<E911_Request> … </E911_Request >	Defined in E911_TSG_request.dtd and E911_intrado_request.dtdgr

</Request>	

5.4.3 E911 DTD

The NeuStar Header XML is defined in DTD files. You can download these at: https://www.neustar.biz/convergentCH/content/ch_dl_header.html

The following DTDs contain the vocabulary and syntax of NeuStar's XML document for E911 requests. You can find these files on https://www.neustar.biz/convergentCH/content/ch_dl_dtd.html.

- **e911_tsg_request.dtd (Schofield)**
- **e911_intrado_request.dtd (Intrado)**
- **e911_tsg_response.dtd (Schofield only)**
- **e911_intrado_response.dtd (Intrado)**

5.4.4 Example E911 Request

An example E911 request is shown below:

```
<?xml version="1.0" encoding="UTF-8" ?>
<Request>
 <E911_Requestcontainer type="container">
  <E911_Request>
   <RequestHeader>
    <OrderNumber value="NTFR010" />
    <ExtractDate value="11-07-2001" />
    <CompanyID value="NTFR" />
   </RequestHeader>
   <RequestBody>
    <ServiceOrder>
     <ActivityType value="ChangeRequest" />
     <EndUserInformation>
      <CustomerName value="John Doe" />
      <ServiceAddress>
       <SANO value="1234" />
       <SASN value="Main" />
       <SATH value="St" />
       <City value="Anytown" />
       <State value="ND" />
       <CallingTelephoneNumber value="558-890-2087" />
      </ServiceAddress>
     </EndUserInformation>
     <ClassOfService value="1" />
     <TypeOfService value="2" />
     <Exchange value="EXC1" />
     <MainTelephoneNumber value="558-890-2087" />
     <Comments value="Intrado V.13" />
    </ServiceOrder>
   </RequestBody>
  </E911_Request>
 </E911_Requestcontainer>
</Request>
```

5.4.5 E911 Responses

There are several types of responses sent to you from the E911 gateway. Each response is associated with a single order. An E911 response message consists of a Response Header and a Response Body.

The Response Header contains control information such as the order number submitted by you in the associated request and an indication of whether the request was satisfactorily processed and/ or forwarded by Intrado/Schofield. The Response Body contains one E911 response; a confirmation response, an error response or a reject response.

Table 24 illustrates how ResponseType maps to the ResponseBody structure in the response XML document.

Table 24: E911 Mapping from ResponseType to ResponseBody

Response Type	ResponseBody contains	Response Indicates
Confirmed Status (Schofield Only) assumedConfirmed correctedConfirmed correctedForwarded	ConfirmationResponse	• Intrado/Schofield processed the request • the request was forwarded to a non-hosted ILEC for processing Intrado/Schofield assumes the request was processed successfully by a non-hosted ILEC, since they did not receive an error response from the ILEC
error	ErrorResponse	• Intrado/Schofield was unable to process the request • a request was forwarded to a non-hosted ILEC that could not process it
reject (Schofield Only) referred errorDeleted (Intrado Only) deletedForwarded	RejectResponse	That the request contained errors Intrado/Schofield could not correct. You must correct reject errors and then resubmit.

Table 25 shows the self-reported frequency of E911 responses.

Table 25: E911 Response types and creation frequency

Confirmation	Response Type	File Creation Frequency	Description
Confirmation	• confirmed • error • status (Schofield Only)	Created on a per file basis	Usually ready within a few hours of the SOI upload to Intrado/Schofield.
DECC File	• correctedConfirmed • correctedForwarded • errorDeleted (Intrado Only) • deletedForwarded	Created daily	Daily error correction confirmation (DECC).
Outstanding Errors File	outstandingErrors	Created weekly on Sunday or Monday	Contains outstanding errors. ♪ NOTE: Errors must be dealt with in sequential order.
Referred Response File	referred	Created daily	Referred responses are supplied with the deluxe service package (the recommended package) and notifications are automated.
Annotated Errors File	error	Created daily	Contains the same information as the Referred responses but is not unformatted and is not GUI viewable. ♪ NOTE: Annotated errors are supplied with the basic service package and no notifications or other events are published. Retrieval is manual and the users' responsibility.

Confirmation	Response Type	File Creation Frequency	Description
SP Confirmation File	assumedConfirmed	Created daily	
Statistics File	statistics	Created on a per file basis	Usually ready within a few hours of the SOI upload to Intrado/Schofield. ♪ NOTE: These are mapped to XML event is published but these are not viewable in the GUI.

5.4.6 Example E911 Response

Two example E911 responses are shown below. Not all response types are represented here.

Example #1 - Error

```
<?xml version="1.0" encoding="UTF-8" ?>
<E911_Response>
 <ResponseHeader>
  <OrderNumber value="NFTR001" />
  <ExtractDate value="05-15-2001" />
  <ResponseType value="Error" />
  <CompanyID value="NTFR" />
 </ResponseHeader>
 <ResponseBody>
  <ErrorResponse>
   <ConfirmationCodeIdentifier value="709" />
   <ServiceOrder>
    <ActivityType value="InsertRequest" />
    <EndUserInformation>
     <CustomerName value="HOMAN HAGHARI" />
     <ServiceAddress>
      <SANO value="1234" />
      <SASN value="MANE" />
      <SATH value="ST" />
      <City value="ANYTOWN" />
      <State value="WY" />
      <CallingTelephoneNumber value="558-234-3287" />
     </ServiceAddress>
    </EndUserInformation>
    <ClassOfService value="1" />
    <TypeOfService value="2" />
    <Exchange value="EXC1" />
    <MainTelephoneNumber value="558-234-3287" />
    <Comments value="INTRADO V.13" />
   </ServiceOrder>
  </ErrorResponse>
 </ResponseBody>
```

Example #2 - Status

```
<?xml version="1.0" encoding="UTF-8" ?>

<E911_Response>
  <ResponseHeader>
    <OrderNumber value="35669534" />
     <ExtractDate value="05-31-2007" />
    <ResponseType value="Status" />
     <CompanyID value="BHNIS" />
  </ResponseHeader>
  <ResponseBody>
```

```
    <StatusResponse>
      <ServiceOrder>
         <ActivityType value="InsertRequest" />
        <EndUserInformation>
          <CustomerName value="SHACARA SCOTT" />
          <ServiceAddress>
            <SANO value="2105" />
            <SASN value="BLOSSOM CT" />
            <City value="HAINES CITY" />
            <State value="FL" />
            <Zip value="33844" />
            <CallingTelephoneNumber value="863-438-5210" />
          </ServiceAddress>
        </EndUserInformation>
         <ClassOfService value="1" />
        <TypeOfService value="3" />
        <MainTelephoneNumber value="863-438-5210" />
      </ServiceOrder>
      <ErrorCode value="410" />
      <ErrorMessage value="MSAG-NPA/NXX VERIFIED" />
    </StatusResponse>
  </ResponseBody>
</E911_Response>
```

5.5 LIDB

NeuStar's Clearinghouse database provides you with access to the Line Information Database (LIDB). LIDB contains information about a subscriber's telephone service. You can issue requests to make changes to the Originating Line Number Screening (OLNS) and the LIDB databases.

5.5.1 LIDB Requests

NeuStar supports the following LIDB subscriber record request:

- **LIDB** – used to insert, change, or delete a line number record

5.5.2 LIDB DTD

DTD files define the NeuStar header XML. The following DTD files contains the vocabulary and syntax of NeuStar's XML document for LIDB requests and responses. You can find these files on https://www.neustar.biz/convergentCH/content/ch_dl_dtd.html.

- **lidb_request.dtd**
- **lidb_response.dtd**

5.5.3 LIDB Responses

Table 26: LIDB Response Message XML Structure

<Response>	

<subscriber_record_response> … </subscriber_record_response>	Defined in lidb_response.dtd
<subscriber_record_responsecontainer> … </subscriber_record_responsecontainer>	Defined in lidb_response.dtd For batch responses.
</Response>	

The following table lists response type values for LIDB orders.

Table 27: LIDB Response Type Values

Response type Values	
LIDB	reject accept – TNS only ack negack – TNS only

♪ NOTE: The string values returned in the eventChannelName parameter of the processEvent() method are shown at https://www.neustar.biz/convergentCH/content/docs/EventChannelName_Valid_Values_and_Descriptions.pdf. This value identifies the Basic Service which is the source of asynchronous notifications and responses when they are sent to you via the processEvent() method of the SOAPResponseHandler interface. A 'message-sent' event is posted to your SOAPResponse handler each time a request to a gateway is initiated from the Basic Services GUI, regardless of the outcome of the request.

5.6 Intercarrier Communication Protocol (ICP)

5.6.1 ICP API Request header XML elements

The following table describes ICP header XML elements.

Table 28: ICP Header XML elements

Header Node Name	Valid Values	Notes
MessageType	MultiPortRequest MultiPortResponse ModifyPortRequest BroadcastNotification TestMessageQuery TestMessageResponse ValidationStatus StoreStatus	Required
TradingPartner	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	Required

Header Node Name	Valid Values	Notes
Timezone	AKDT – Alaska Daylight Saving Time AKST – Alaska Standard Time AST – Atlantic Standard Time CDT – Central Daylight Saving Time CST – Central Standard Time EDT – Eastern Daylight Saving Time EST – Eastern Standard Time HST – Hawaiian Standard Time MDT – Mountain Daylight Saving Time MST – Mountain Standard Time PDT – Pacific Daylight Saving Time PST – Pacific Standard Time UTC – Coordinated Universal Time	Required
Sender	Max length: 4	Optional
Receiver	Max length: 4	Optional
CORRELATION_ID	Numeric string Maximum length: 64	Optional
WICIS_REL_NO	4.0.0 for requests received electronically 3.1.0 for requests received via fax	Required
MessageSubType	create – for MultiPortRequest cancel – for ModifyPortRequest dueDateChange – for ModifyPortRequest modify – for ModifyPortRequest delay – for MultiPortResponse confirm – for MultiPortResponse resolutionRequired – for MultiPortResponse ack – for StoreStatus message** negack – for StoreStatus message** valid – for ValidationStatus message** invalid – for ValidationStatus message**	Optional
** MessageSubType ack/negack and valid/invalid are generated automatically by the API. These MessageSubTypes are not included in XML messages.		

5.6.2 ICP DTDs

The NeuStar Header XML is defined in a DTD file. You can download this at: https://www.neustar.biz/convergentCH/content/ch_dl_header.html

The following DTD contain the vocabulary and syntax of NeuStar's XML document for ICP requests. You can find this file on https://www.neustar.biz/convergentCH/content/ch_dl_dtd.html.

- **wireless_np_message.dtd**

5.6.3 MultiPortRequest

The MessageSubType value in the header is always "create."

5.6.3.1 Parameters – MultiPortRequest

Table 29: MultiPortRequest Parameters

Parameter	Description	Valid Values
REQNO	Request number	Length: 11-16 Format: NNSPIYYJJJXXXXXX where: ○ NNSP=NNSP value from WPR ○ I=numeric Host ID [ICP HOST ID] (0-9, varies per instance of multiple ICPs per SPID) ○ YY=last two digits of the system year as generated by the NNSP (00-99) ○ JJJ=system julian date as generated by the NNSP (001-366) ○ XXXXXX=1-6 position unique alphanumeric value (0-9, A-Z) Required
VER	Version of the request	Length: 2. Should be 01 on initial requests. Required
ONSP	Old Network Service Provider SPID of the facility-based provider	Length: 4 Required
NLSP	Operating Company Number (OCN) or SPID of the New Service Provider	Length: 4 Required
NNSP	New Network Service Provider SPID of the facility-based provider	Length: 4 Required
NRSELLNM	Name of the New Reseller involved in the port Reques	Max length: 20
NPDI	Direction of the port	WirelessToWireless WirelessToWireline WirelineToWireless Required
DTSENT	Date and time that the request was sent from the NSP	mm-dd-yyyy-hhmm[AM|PM] Required
DDDT	Desired due date and time for the completion of the port and activation of service	mm-dd-yyyy-hhmm[AM|PM] Required
CHC	Coordinated Hot Cut - a request by the NSP to ensure a coordinated effort to port all numbers on the request at the same time	Y = yes
AGAUTH	Agency Authorization Status	Y= authorization on file N= no authorization on file Required
DATED	Date that the port authorization was received	Format: MM-DD-YYYY
AUTHNM	Name of the customer that authorized the request to port the number	Max length: 60
GREQNO	Group Request Number Relates multiple requests back to a single customer	Max length: 20

Parameter	Description	Valid Values
INIT	Initiator Identification NSP representative who originated the request	Max length: 15 Required
IMPCON	Implementation Contact NSP representative who is initiating the port	Max length: 15 Required
IMPTELNO	Implementation Contact telephone number	Valid values are NNN-NNN-NNNN or NNN-NNN-NNNN-NNNN where N is a numeric character. Required
BillingInfo		○ PREFIX – name prefix of the customer (Mr., Ms., etc.) . Max length: 10 ○ FIRSTNM - last name of the customer. Max length: 25 ○ MDINIT - middle initial of the customer. Max length: 1 ○ LASTNM - last name of the customer. Max length: 25 ○ SUFFIX - name suffix of the customer. Max length: 10 ○ BUSNM - last name of the customer. Max length: 25. Required when FIRSTNM and LASTNM are blank. ○ STNUM - Street/house number of the billing address. Max length: 10 ○ STNM - Street name of the billing address. Max length: 60. Required. ○ STDIR - Street directional for the billing address. Max length: 2 ○ CITY - City of the billing address. Max length: 35. Required. ○ STATE - State of the billing address. Max length: 2 ○ ZIPCODE - ZIPCODE of the billing address. Valid format for USA: NNNNN or NNNNN-NNNN where N is a numeric character,. Valid format for Canada: XXXXXX, where X is alphanumeric character. ○ COUNTRY - Country of the billing address. Max length: 3
SSN_TAX_ID	Social Security Number or TaxID	000-00-NNNN for SSN 00-000NNNN for TAX_ID where N is a numeric character.
ACCT	Customer's account number within the OSP's internal systems	Max length: 20 with no embedded blanks.
PSWDPIN	Customer's password or pin number specified on his/her account within the OSP's internal systems	Max length: 15
NPQTY	Quantity of telephone numbers involved in the port request	00001-10000 Required
REMARKS		Max length: 160

Parameter	Description	Valid Values
PortedNumbercontainer	PortedNumber List of numbers included in the port request.	○ LNUM - Line number for each TN or range of TNs involved in a request. Valid values: 00001 – 01000. Required ○ PORTEDNUM - A single telephone number (TN) or range of consecutive TNs to be ported. Valid values are NNN-NNN-NNNN or NNN-NNN-NNNN-NNNN where N is a numeric character. Required ○ NAME - Name of the subscriber for the associated porting number. Max length: 60

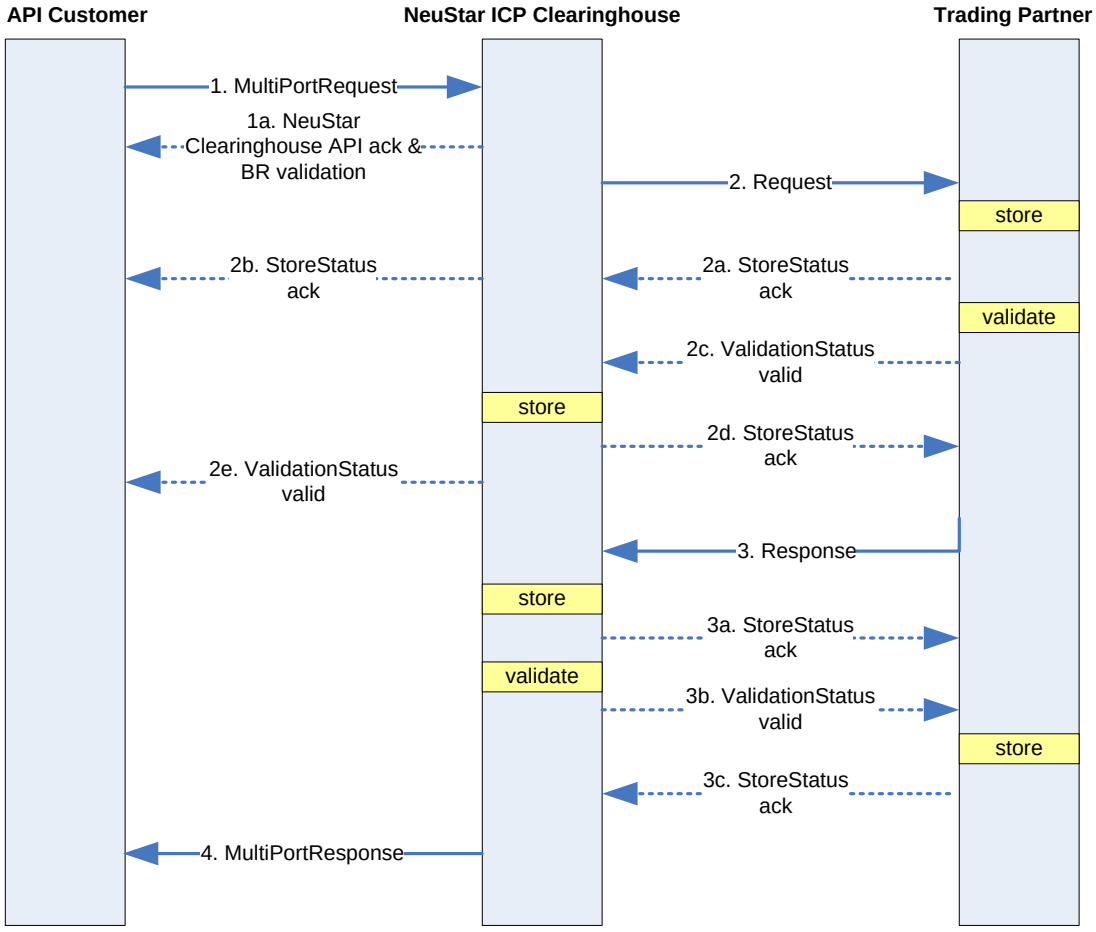
5.6.3.2 XML Example - MultiPortRequest

```
<?xml version="1.0" encoding="UTF-8"?><wireless_nport>
<Header>
<TradingPartner value="S004"/>
<Receiver value="S004"/>
<Sender value="X088"/>
<WICIS_REL_NO value="4.0.0"/>
<MessageSubType value="create"/>
<MessageType value="MultiPortRequest"/>
<Timezone value="UTC"/>
<CORRELATION_ID
value="X088000245458100000000000000000000000000000000000000000000000270"/>
</Header>
<Body>
<PortRequest>
<VER value="01"/>
<DTSENT value="04-24-2008-0301PM"/>
<NLSP value="X088"/>
<REQNO value="X088208115000010"/>
<BillingInfo>
<STNUM value="231"/>
<FIRSTNM value="Mark"/>
<CITY value="San Ramon"/>
<STATE value="CA"/>
<MDINIT value="A"/>
<LASTNM value="Smith"/>
<ZIPCODE value="94583"/>
<STNM value="Market Place PMB 144"/>
<COUNTRY value="USA"/>
</BillingInfo>
<IMPCON value="Mubeen"/>
<SSN_TAX_ID value="000-00-1234"/>
<AUTHNM value="Grandma George"/>
<PortedNumbercontainer type="container">
<PortedNumber>
<LNUM value="00001"/>
<PORTEDNUM value="510-500-1001"/>
<NAME value="Mark A Smith"/>
</PortedNumber>
</PortedNumbercontainer>
<AGAUTH value="Y"/>
```

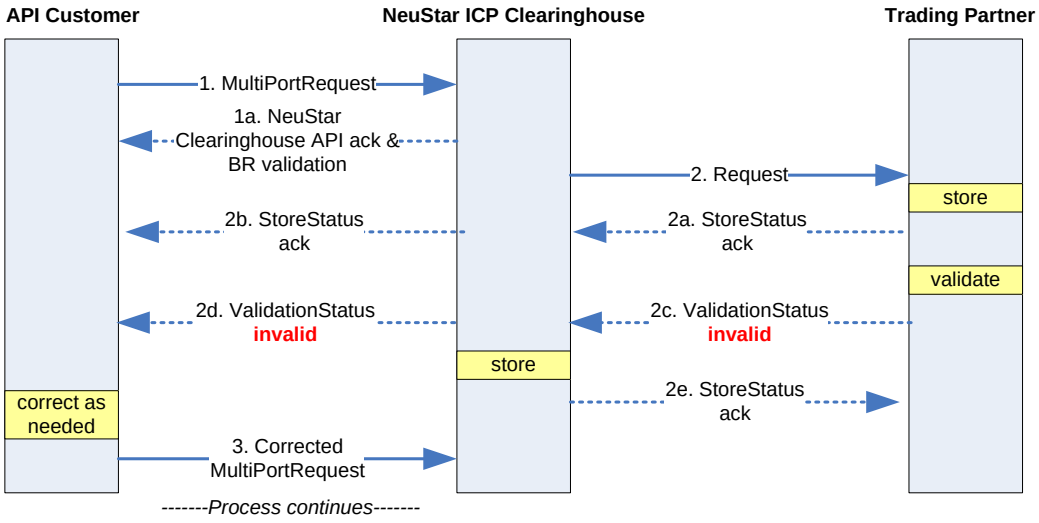
```
<NNSP value="X088"/>
<DDDT value="04-24-2008-1100PM"/>
<NPQTY value="00001"/>
<ONSP value="S004"/>
<INIT value="Mubeen"/>
<DATED value="04-22-2008"/>
<NPDI value="WirelessToWireless"/>
<IMPTELNO value="925-833-1793"/>
</PortRequest>
</Body>
</wireless_nport>
```

5.6.3.3 Flow Charts– MultiPortRequest

MultiPortRequest sent by API Customer
SUNNY Day

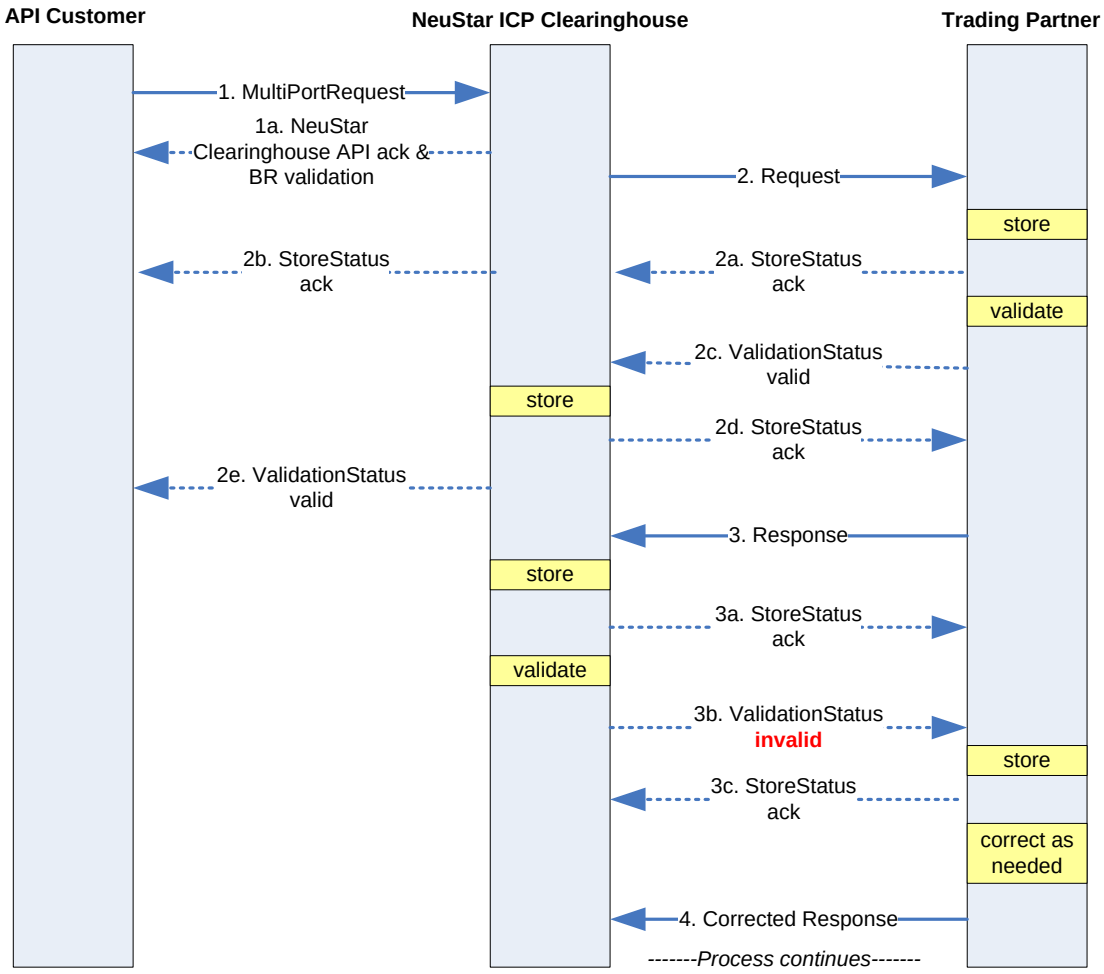


MultiPortRequest sent by API Customer
RAINY Day
Scenario #1



MultiPortRequest sent by API Customer
RAINY Day

Scenario #2



5.6.4 MultiPortResponse

The ONSP responds to the WPR received from the NNSP.

For MultiPortResponse type, the MessageSubType value in the header is one of the following values:

- resolutionRequired – submitted by the ONSP to indicate that customer information validation errors exist in the request.
- delay – submitted by the ONSP to indicate that additional time is required to respond
- confirm – submitted by the ONSP to indicate confirmation of a request.

5.6.4.1 Parameters – MultiPortResponse

Table 30: MultiPortResponse Parameters

Parameter	Description	Valid Values
NNSP	New Network Service Provider SPID of the facility-based provider	Length: 4 Required
OLSP	Operating Company Number (OCN) or SPID of the Old Service Provider	Length: 4 Required
ONSP	Old Network Service Provider SPID of the facility-based provider	Length: 4 Required
ORSELLNM	Name of the Old Reseller involved in the port response	Max length: 20
REQNO	Request Number	Length: 11-16 Format: NNSPIYYJJJXXXXXX where: ○ NNSP=NNSP value from WPR ○ I=numeric Host ID [ICP HOST ID] (0-9, varies per instance of multiple ICPs per SPID) ○ YY=last two digits of the system year as generated by the NNSP (00-99) ○ JJJ=system julian date as generated by the NNSP (001-366) ○ XXXXXX=1-6 position unique alphanumeric value (0-9, A-Z) Required
VERIDREQ	Version identification of the request	Length: 2 Required
VERIDRESP	Version identification of the response	Length: 2
RT	Response type	C = confirmation D = delay R = resolution required
DCODE	Reason for the delay	6G = Port Complexity 6H = System Outage 6J = High Volume 6L = Request received outside of business hours
GRESPNO	Group Response Number Relates multiple responses together	Max length: 20
RESPNO	Response number	The valid format for RESPNO is ONSPIYYJJJXXXXXX where: ○ ONSP=ONSP value from WPR ○ I=numeric Host ID [ICP HOST ID] (0-9, varies per instance of multiple ICPs per SPID) ○ YY=last two digits of the system year as generated by the ONSP (00-99) ○ JJJ=system julian date as generated by the ONSP (001-366) ○ XXXXXX=1-6 position unique alphanumeric value (0-9, A-Z) Required

Parameter	Description	Valid Values
RD_TSENT	Date and time that the OSP sent the response	mm-dd-yyyy-hhmm[AM|PM] Required
REP	OSP Contact Representative for issues concerning a port Response	Max length: 15 Required
REPTELNO	OSP representative's telephone tumber for issues concerning the port response	NNN-NNN-NNNN or NNN-NNN-NNNN-NNNN where N is a numeric character. Required
CHC	Coordinated Hot Cut - a request by the NSP to ensure a coordinated effort to port all numbers on the request at the same time	Y = yes
DDT	Due Date and Time for port completion	mm-dd-yyyy-hhmm[AM|PM] Required
NPQTY	Quantity of telephone numbers involved in the port request	00001-10000
REMARKS		Max length: 160
PortedNumbercontainer	PortedNumber List of numbers included in the port request.	○ LNUM - Line number for each TN or range of TNs involved in a request. Valid values: 00001 – 01000. Required ○ PORTEDNUM - A single telephone number (TN) or range of consecutive TNs to be ported. Valid values are NNN-NNN-NNNN or NNN-NNN-NNNN-NNNN where N is a numeric character. Required ○ NAME - Name of the subscriber for the associated porting number. Max length: 60

5.6.4.2 XML Example – MultiPortResponse: resolutionRequired

```
<?xml version="1.0" encoding="UTF-8"?><wireless_nport>
<Header>
<MessageType value="MultiPortResponse"/>
<TradingPartner value="S004"/>
<Timezone value="UTC"/>
<Sender value="S004"/>
<Receiver value="X088"/>
<CORRELATION_ID value="S004008116171007120915780719554O1"/>
<WICIS_REL_NO value="4.0.0"/>
<MessageSubType value="resolutionRequired"/>
</Header>
<Body>
<PortResponse>
<NNSP value="X088"/>
<OLSP value="S004"/>
<ONSP value="S004"/>
<REQNO value="X088008116365023"/>
<VERIDREQ value="01"/>
<VERIDRESP value="01"/>
```

<RT value="R"/>
<RESPNO value="S004008116311146"/>
<RD_TSENT value="04-25-2008-0910PM"/>
<REP value="Sood "/>
<REPTELNO value="999-999-9999"/>
<DDT value="04-26-2008-1130PM"/>
<NPQTY value="00005"/>
<REMARKS value="RT - R"/>
<PortedLineResponsecontainer type="container">
<PortedLineResponse>
<LNUM value="00001"/>
<PORTEDNUM value="510-500-1117"/>
<RCODE value="6A"/>
<RDET value="MDN not found/doesn't belong to this SP"/>
</PortedLineResponse>
<PortedLineResponse>
<LNUM value="00002"/>
<PORTEDNUM value="510-500-1118"/>
<RCODE value="6A"/>
<RDET value="MDN not found/doesn't belong to this SP"/>
</PortedLineResponse>
<PortedLineResponse>
<LNUM value="00003"/>
<PORTEDNUM value="510-500-1119"/>
<RCODE value="6A"/>
<RDET value="MDN not found/doesn't belong to this SP"/>
</PortedLineResponse>
<PortedLineResponse>
<LNUM value="00004"/>
<PORTEDNUM value="510-500-1120"/>
<RCODE value="7A"/>
<RDET value="LNUM/MDN is valid; errors with other LNUMs/MDNs in request"/>
</PortedLineResponse>
<PortedLineResponse>
<LNUM value="00005"/>
<PORTEDNUM value="510-500-1121"/>
<RCODE value="7A"/>
<RDET value="LNUM/MDN is valid; errors with other LNUMs/MDNs in request"/>
</PortedLineResponse>
</PortedLineResponsecontainer>
</PortResponse>
</Body>
</wireless_nport>

5.6.4.3 XML Example – MultiPortResponse: delay

<?xml version="1.0" encoding="UTF-8"?><wireless_nport>
<Header>
<MessageType value="MultiPortResponse"/>
<TradingPartner value="S004"/>
<Timezone value="UTC"/>
<Sender value="S004"/>

<Receiver value="X088"/>
<CORRELATION_ID value="S004008115161858120906833801721 02"/>
<WICIS_REL_NO value="4.0.0"/>
<MessageSubType value="delay"/>
</Header>
<Body>
<PortResponse>
<NNSP value="X088"/>
<OLSP value="S004"/>
<ONSP value="S004"/>
<REQNO value="X088208115000011"/>
<VERIDREQ value="01"/>
<VERIDRESP value="01"/>
<RT value="D"/>
<DCODE value="6J"/>
<RESPNO value="S004008115311128"/>
<RD_TSENT value="04-24-2008-0818PM"/>
<REP value="Sood"/>
<REPTELNO value="999-999-9901"/>
<DDT value="04-24-2008-1100PM"/>
<REMARKS value="Delay"/>
</PortResponse>
</Body>
</wireless_nport>

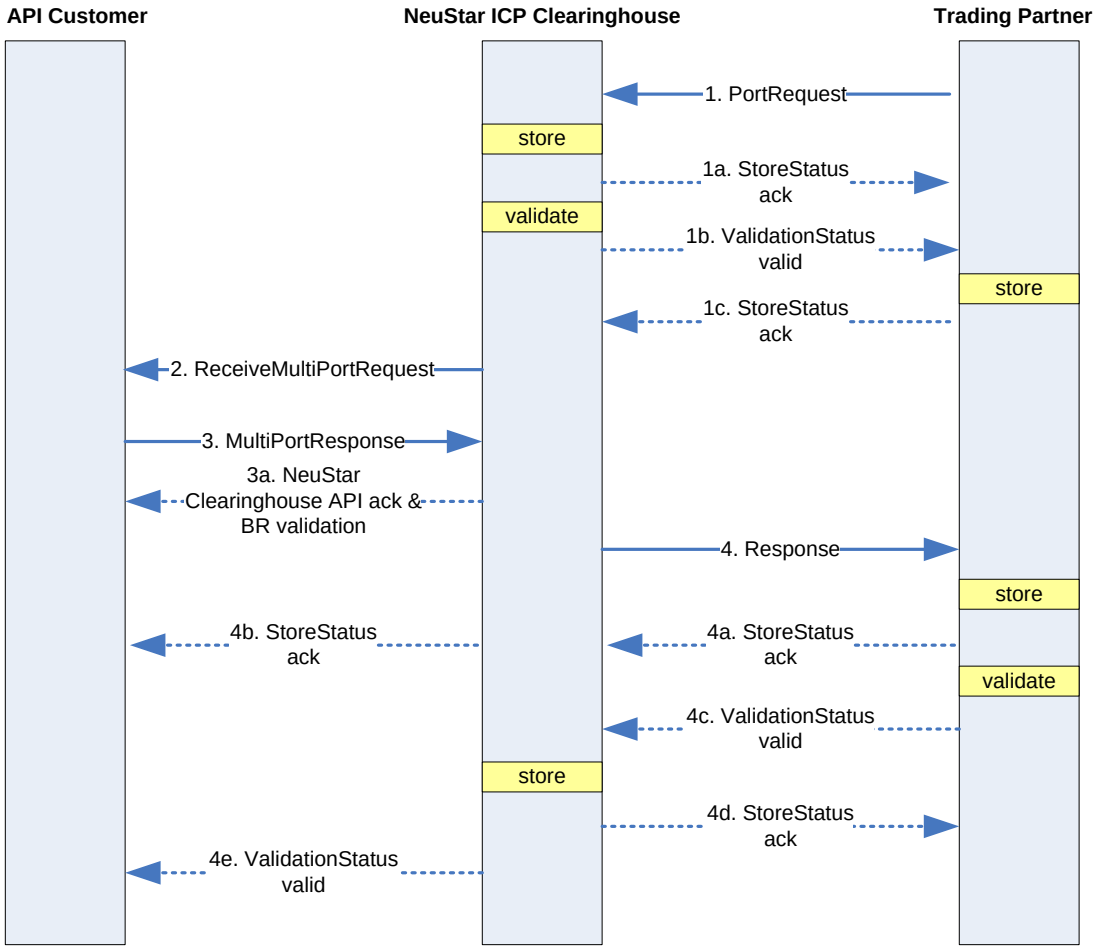
5.6.4.4 XML Example – MultiPortResponse: confirm

<?xml version="1.0" encoding="UTF-8"?><wireless_nport>
<Header>
<MessageType value="MultiPortResponse"/>
<TradingPartner value="S004"/>
<Timezone value="UTC"/>
<Sender value="S004"/>
<Receiver value="X088"/>
<CORRELATION_ID value="S004008115161940120906838070821 23"/>
<WICIS_REL_NO value="4.0.0"/>
<MessageSubType value="confirm"/>
</Header>
<Body>
<PortResponse>
<NNSP value="X088"/>
<OLSP value="S004"/>
<ONSP value="S004"/>
<REQNO value="X088208115000011"/>
<VERIDREQ value="01"/>
<VERIDRESP value="02"/>
<RT value="C"/>
<RESPNO value="S004008115311128"/>
<RD_TSENT value="04-24-2008-0819PM"/>
<REP value="Sood"/>
<REPTELNO value="999-999-9101"/>
<DDT value="04-24-2008-1100PM"/>
<NPQTY value="00001"/>

<REMARKS value="Conf."/>
<PortedLineResponsecontainer type="container">
<PortedLineResponse>
<LNUM value="00001"/>
<PORTEDNUM value="510-500-1002"/>
</PortedLineResponse>
</PortedLineResponsecontainer>
</PortResponse>
</Body>
</wireless_nport>

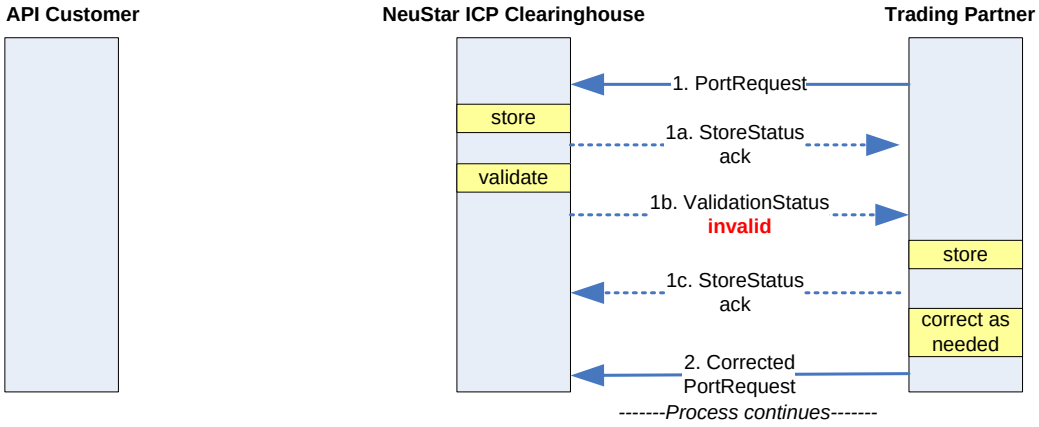
5.6.4.5 Flow Charts - MultiPort Response

Port Out Request received from TP
MultiPortResponse sent by API Customer
SUNNY Day

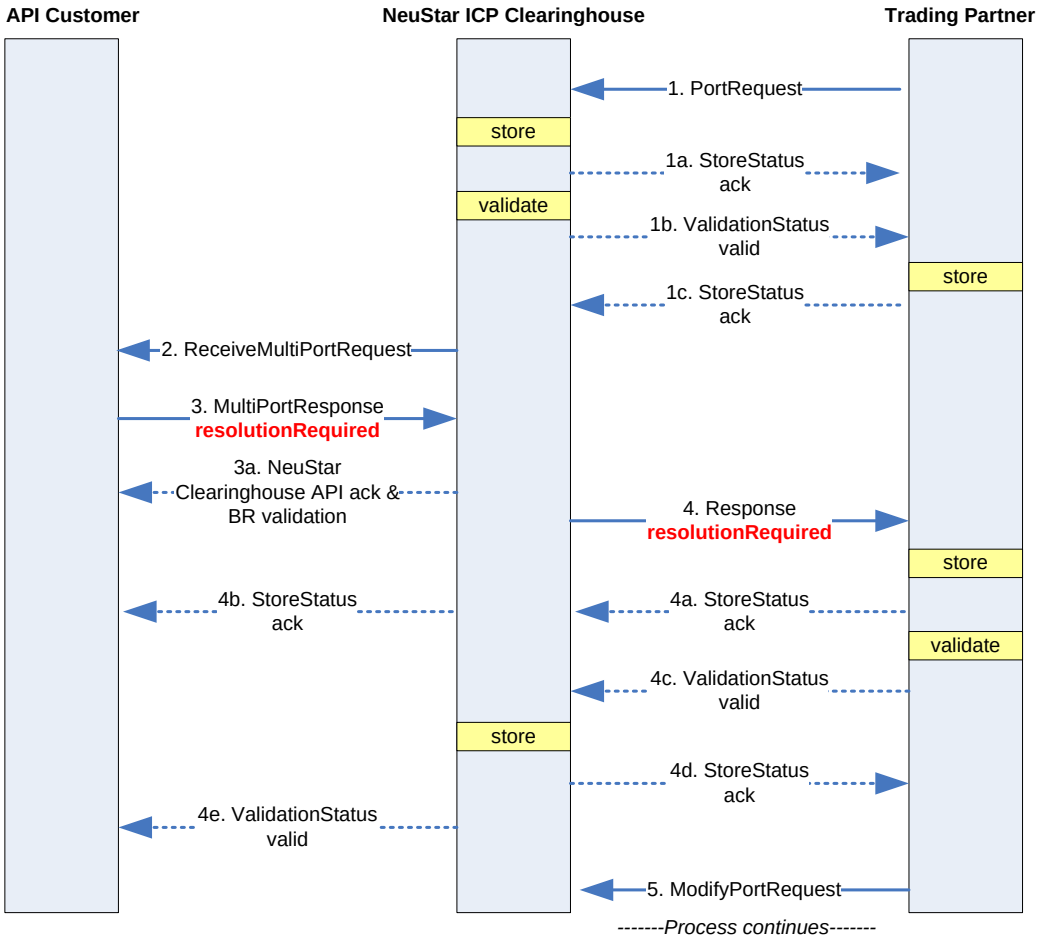


Port Out Request received from TP
MultiPortResponse sent by API Customer
RAINY Day

Scenario #1

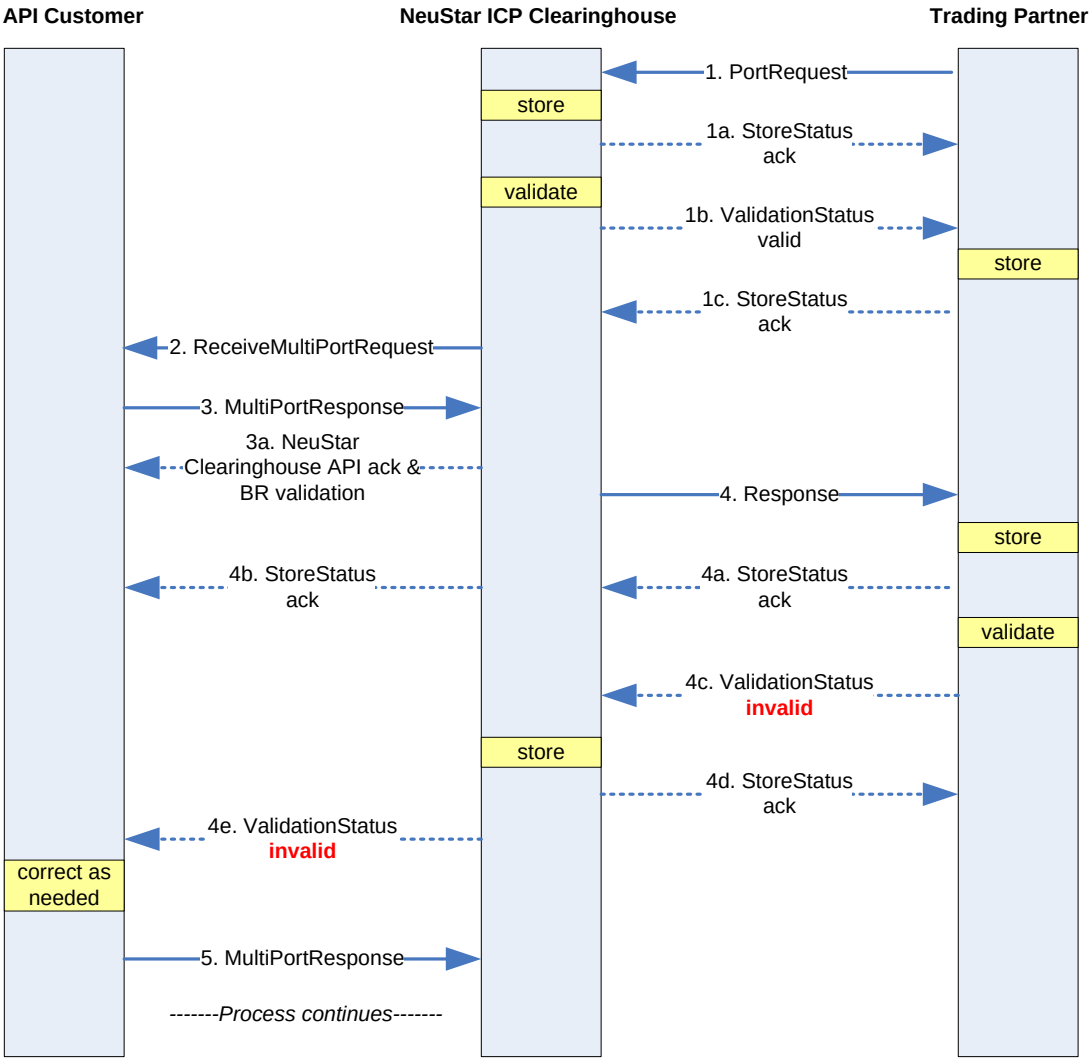


Port Out Request received from TP
MultiPortResponse sent by API Customer
RAINY Day
Scenario #2



Port Out Request received from TP
MultiPortResponse sent by API Customer
RAINY Day

Scenario #3



5.6.5 ModifyPortRequest

The NNSP submits a request to modify an existing WPR.

For ModifyPortRequest type, the MessageSubType value in the header is one of the following values:

- cancel – submitted by the NNSP to cancel a request. The Body of the XML shows a Supplement value 1 (SUP value=”1”).

- dueDateChange - submitted by the NNSP to modify a due date and time on a confirmed request. The Body of the XML shows a Supplement value 2 (SUP value=”2”).

- modify – submitted by the ONSP to make a general change to the request. The Body of the XML shows a Supplement value 3 (SUP value=”3”).

5.6.5.1 Parameters – ModifyPortRequest

Table 31: ModifyPortRequest Parameters

Parameter	Description	Valid Values
SUP	Type of supplement	1 = cancel 2 = dueDateChange 3 = modify
REQNO	Request number	Length: 11-16 Format: NNSPIYYJJJXXXXXX where: ○ NNSP=NNSP value from WPR ○ I=numeric Host ID [ICP HOST ID] (0-9, varies per instance of multiple ICPs per SPID) ○ YY=last two digits of the system year as generated by the NNSP (00-99) ○ JJJ=system julian date as generated by the NNSP (001-366) ○ XXXXXX=1-6 position unique alphanumeric value (0-9, A-Z) Required
VER	Version of the request	Length: 2. Required
ONSP	Old Network Service Provider SPID of the facility-based provider	Length: 4 Required
NLSP	Operating Company Number (OCN) or SPID of the New Service Provider	Length: 4 Required
NNSP	New Network Service Provider SPID of the facility-based provider	Length: 4 Required
NRSELLNM	Name of the New Reseller involved in the port Request	Max length: 20
NPDI	Direction of the port	WirelessToWireless WirelessToWireline WirelineToWireless Required

Parameter	Description	Valid Values
RESPNO	Response Number	Length: 11-16 Format: NNSPIYYJJJXXXXXX where: ○ NNSP=NNSP value from WPR ○ I=numeric Host ID [ICP HOST ID] (0-9, varies per instance of multiple ICPs per SPID) ○ YY=last two digits of the system year as generated by the NNSP (00-99) ○ JJJ=system julian date as generated by the NNSP (001-366) ○ XXXXXX=1-6 position unique alphanumeric value (0-9, A-Z) Required
DTSENT	Date and time that the request was sent from the NSP	mm-dd-yyyy-hhmm[AM|PM] Required
DDDT	Desired due date and time for the completion of the port and activation of service	mm-dd-yyyy-hhmm[AM|PM]
CHC	Coordinated Hot Cut - a request by the NSP to ensure a coordinated effort to port all numbers on the request at the same time	Y = yes
AGAUTH	Agency Authorization Status	Y= authorization on file N= no authorization on file Required
DATED	Date that the port authorization was received	Format: MM-DD-YYYY
AUTHNM	Name of the customer that authorized the request to port the number	Max length: 60
GREQNO	Group Request Number Relates multiple requests back to a single customer	Max length: 20
INIT	Initiator Identification NSP representative who originated the request	Max length: 15 Required
IMPCON	Implementation Contact NSP representative who is initiating the port	Max length: 15 Required
IMPTELNO	Implementation Contact telephone number	Valid values are NNN-NNN-NNNN or NNN-NNN-NNNN-NNNN where N is a numeric character. Required

Parameter	Description	Valid Values
BillingInfo		○ PREFIX – name prefix of the customer (Mr., Ms., etc.) . Max length: 10 ○ FIRSTNM - last name of the customer. Max length: 25 ○ MDINIT - middle initial of the customer. Max length: 1 ○ LASTNM - last name of the customer. Max length: 25 ○ SUFFIX - name suffix of the customer. Max length: 10 ○ BUSNM - last name of the customer. Max length: 25. Required when FIRSTNM and LASTNM are blank. ○ STNUM - Street/house number of the billing address. Max length: 10 ○ STNM - Street name of the billing address. Max length: 60. Required. ○ STDIR - Street directional for the billing address. Max length: 2 ○ CITY - City of the billing address. Max length: 35. Required. ○ STATE - State of the billing address. Max length: 2 ○ ZIPCODE - ZIPCODE of the billing address. Valid format for USA: NNNNN or NNNNN-NNNN where N is a numeric character,. Valid format for Canada: XXXXXX, where X is alphanumeric character. ○ COUNTRY - Country of the billing address. Max length: 3
SSN_TAX_ID	Social Security Number or TaxID	000-00-NNNN for SSN 00-000NNNN for TAX_ID where N is a numeric character.
ACCT	Customer's account number within the OSP's internal systems	Max length: 20 with no embedded blanks.
PSWDPIN	Customer's password or pin number specified on his/her account within the OSP's internal systems	Max length: 15
NPQTY	Quantity of telephone numbers involved in the port request	00001-10000
REMARKS		Max length: 160
PortedNumbercontainer	PortedNumber List of numbers included in the port request.	○ LNUM - Line number for each TN or range of TNs involved in a request. Valid values: 00001 – 01000. Required ○ PORTEDNUM - A single telephone number (TN) or range of consecutive TNs to be ported. Valid values are NNN-NNN-NNNN or NNN-NNN-NNNN-NNNN where N is a numeric character. Required ○ NAME - Name of the subscriber for the associated porting number. Max length: 60

5.6.5.2 XML Example – ModifyPortRequest: cancel

```
<?xml version="1.0" encoding="UTF-8"?><wireless_nport>
<Header>
<TradingPartner value="S004"/>
<Receiver value="S004"/>
<Sender value="X088"/>
<WICIS_REL_NO value="4.0.0"/>
```

```
<MessageType value="ModifyPortRequest"/>
<Timezone value="UTC"/>
<MessageSubType value="cancel"/>
<CORRELATION_ID
value="X088000245458200000000000000000000000000000000000000000000000323"/>
</Header>
<Body>
<ModifyPortRequest>
<SUP value="1"/>
<PortRequest>
<DTSENT value="04-25-2008-1201PM"/>
<REQNO value="X088008116365012"/>
<ONSP value="S004"/>
<NLSP value="X088"/>
<NNSP value="X088"/>
<VER value="02"/>
<SUP value="1"/>
</PortRequest>
</ModifyPortRequest>
</Body>
</wireless_nport>
```

5.6.5.3 XML Example – ModifyPortRequest: dueDateChange

```
<?xml version="1.0" encoding="UTF-8"?><wireless_nport>
<Header>
<TradingPartner value="S004"/>
<Receiver value="S004"/>
<Sender value="X088"/>
<WICIS_REL_NO value="4.0.0"/>
<MessageType value="ModifyPortRequest"/>
<Timezone value="UTC"/>
<MessageSubType value="dueDateChange"/>
<CORRELATION_ID
value="X088000245458200000000000000000000000000000000000000000000000349"/>
</Header>
<Body>
<ModifyPortRequest>
<SUP value="2"/>
<PortRequest>
<DTSENT value="04-25-2008-0300PM"/>
<REQNO value="X088008116365019"/>
<ONSP value="S004"/>
<NLSP value="X088"/>
<NNSP value="X088"/>
<DDDT value="04-25-2008-1130PM"/>
<REMARKS value="Updated DDDT"/>
<VER value="02"/>
<SUP value="2"/>
<RESPNO value="S004008116311142"/>
</PortRequest>
</ModifyPortRequest>
</Body>
</wireless_nport>
```

5.6.5.4 XML Example – ModifyPortRequest: modify

```
<?xml version="1.0" encoding="UTF-8"?><wireless_nport>
<Header>
<TradingPartner value="S004"/>
<Receiver value="S004"/>
<Sender value="X088"/>
<WICIS_REL_NO value="4.0.0"/>
<MessageType value="ModifyPortRequest"/>
<Timezone value="UTC"/>
<MessageSubType value="modify"/>
<CORRELATION_ID
value="X088000245458200000000000000000000000000000000000000000000000000345"/>
</Header>
<Body>
<ModifyPortRequest>
<SUP value="3"/>
<PortRequest>
<DTSENT value="04-25-2008-0248PM"/>
<REQNO value="X088008116365018"/>
<ONSP value="S004"/>
<NLSP value="X088"/>
<NNSP value="X088"/>
<NPDI value="WirelessToWireless"/>
<DDDT value="04-25-2008-1100PM"/>
<AGAUTH value="Y"/>
<DATED value="04-22-2008"/>
<AUTHNM value="Grandma George"/>
<INIT value="Mubeen"/>
<IMPCON value="Mubeen"/>
<IMPTELNO value="925-833-1793"/>
<BillingInfo>
<FIRSTNM value="Mark"/>
<MDINIT value="A"/>
<LASTNM value="Smith"/>
<STNUM value="231"/>
<STNM value="Market Place PMB 144"/>
<CITY value="San Ramon"/>
<STATE value="CA"/>
<ZIPCODE value="94583"/>
<COUNTRY value="USA"/>
</BillingInfo>
<SSN_TAX_ID value="000-00-1234"/>
<PSWDPIN value="ABCDEFG"/>
<NPQTY value="00003"/>
<REMARKS value="Updated Password Information"/>
<PortedNumbercontainer type="container">
<PortedNumber>
<LNUM value="00001"/>
<PORTEDNUM value="510-500-1111"/>
<NAME value="Mark A Smith"/>
</PortedNumber>
<PortedNumber>
<LNUM value="00002"/>
```

<PORTEDNUM value="510-500-1112"/>
<NAME value="Mark A Smith"/>
</PortedNumber>
<PortedNumber>
<LNUM value="00003"/>
<PORTEDNUM value="510-500-1113"/>
<NAME value="Mark A Smith"/>
</PortedNumber>
</PortedNumbercontainer>
<SUP value="3"/>
<RESPNO value="S004008116311140"/>
<VER value="02"/>
</PortRequest>
</ModifyPortRequest>
</Body>
</wireless_nport>

5.6.6 BroadcastNotification

5.6.6.1 Parameters - BroadcastNotification

Table 32: BroadcastNotification Parameters

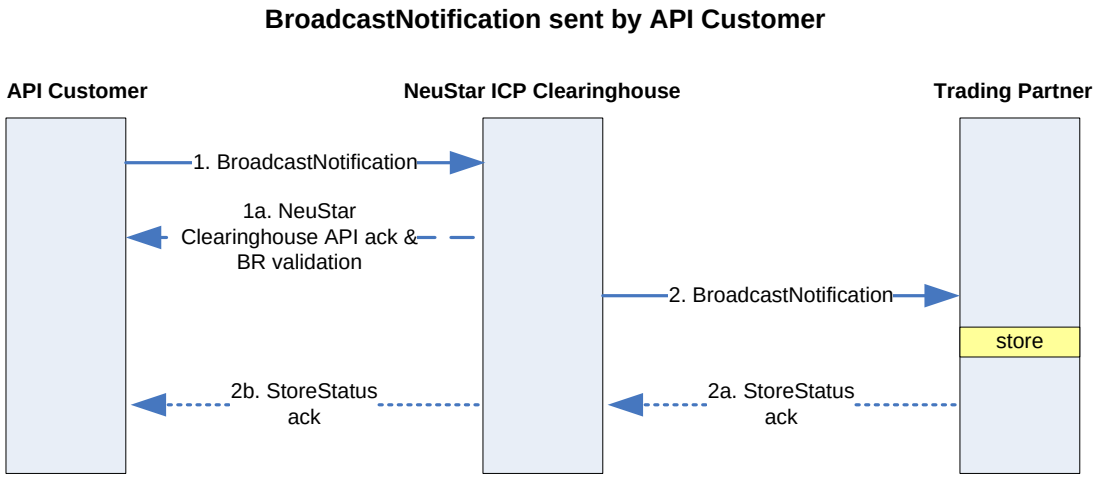
Parameter	Description	Valid Values
BEGIN_MAINTENANCE	Date and time maintenance will begin	mm-dd-yyyy-hhmm[AM|PM]
END_MAINTENANCE	Date and time maintenance will end	mm-dd-yyyy-hhmm[AM|PM]
SERVER_STATUS	Status of the server	Required A - ICP System Outage - Full: All SPIDs belonging to this provider B - ICP System Outage - Partial: One or more SPIDs but not all belonging to this provider C - Billing System Outage – Full: All billing system functions D - Billing System Outage – Partial: Some billing system functions such as for one SPID or Region E - Billing system maintenance scheduled – Full F - Billing system maintenance scheduled - Partial G - Network Outage – Full: Entire network is down H - Network Outage - Partial: Some network pieces are down such as connectivity to call center I - Clearinghouse Outage – Full: Clearinghouse is down J - Clearinghouse Outage - Partial: Clearinghouse is down K - Slow response L - Not Supported: Provider does not utilize broadcast messages M - Maintenance Mode - Scheduled: During stated maintenance windows such as NPAC maintenance N - Normal Operations U - Unavailable: No answer is received from the provider
DTSENT	Date and time the message was sent.	mm-dd-yyyy-hhmm[AM|PM] Required

Parameter	Description	Valid Values
DESTINATION_SPID	SPID of the target of the message	Length: 4 Required
REMARKS	Comments	Max length: 160
SENDER_SPID	SPID of the sender of the message	Length: 4 Required

5.6.6.2 XML Example - BroadcastNotification

```
<?xml version="1.0" encoding="UTF-8"?><wireless_nport>
<Header>
<TradingPartner value="S004"/>
<Receiver value="S004"/>
<Sender value="X088"/>
<WICIS_REL_NO value="4.0.0"/>
<Timezone value="UTC"/>
<MessageType value="BroadcastNotification"/>
<CORRELATION_ID
value="X088000245458600000000000000000000000000000000000000000000000462"/>
</Header>
<Body>
<BroadcastNotification>
<DTSENT value="04-29-2008-1112AM"/>
<SERVER_STATUS value="M"/>
<END_MAINTENANCE value="04-29-2008-1130AM"/>
<BEGIN_MAINTENANCE value="04-29-2008-1110AM"/>
<REMARKS value="R16"/>
<DESTINATION_SPID value="S004"/>
<SENDER_SPID value="X088"/>
</BroadcastNotification>
</Body>
</wireless_nport>
```

5.6.6.3 Flow Chart – BroadcastNotification



5.6.7 TestMessageQuery

5.6.7.1 Parameters - TestMessageQuery

Table 33: TestMessageQuery Parameters

Parameter	Description	Valid Values
DTSENT	Date and time the message was sent.	mm-dd-yyyy-hhmm[AM|PM] Required
SENDER_SPID	SPID of the sender of the message	Max length: 4 Required
DESTINATION_SPID	SPID of the target of the message	Max length: 4 Required

5.6.7.2 XML Example - TestMessageQuery

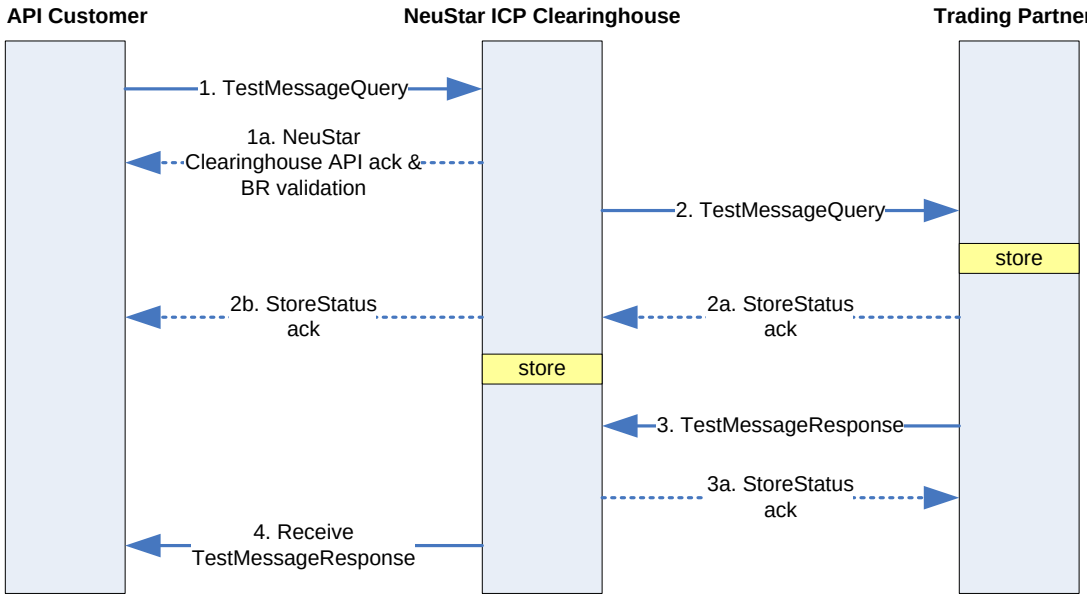
```
<?xml version="1.0" encoding="UTF-8"?>
<wireless_nport>
        <Header>
                <MessageType value="TestMessageQuery"/>
                <TradingPartner value="6604"/>
                <Timezone value="EST"/>
                <Sender value="6017"/>
                <Receiver value="6604"/>

                <WICIS_REL_NO value="4.0.0"/>
        </Header>
        <Body>
```

```
            <TestMessageQuery>
                  <DTSENT value="05-15-2008-0444PM"/>
                  <SENDER_SPID  value="6017"/>
                  <DESTINATION_SPID  value="6604"/>
            </TestMessageQuery>
      </Body>
</wireless_nport>
```

5.6.7.3 Flow Chart – TestMessageQuery

TestMessageQuery sent by API Customer



5.6.8 TestMessageResponse

5.6.8.1 Parameters - TestMessageResponse

Table 34: TestMessageResponse Parameters

Parameter	Description	Valid Values
DTSENT	Date and time the message was sent.	mm-dd-yyyy-hhmm[AM|PM] Required
SENDER_SPID	SPID of the sender of the message	Max length: 4 Required
DESTINATION_SPID	SPID of the target of the message	Max length: 4 Required

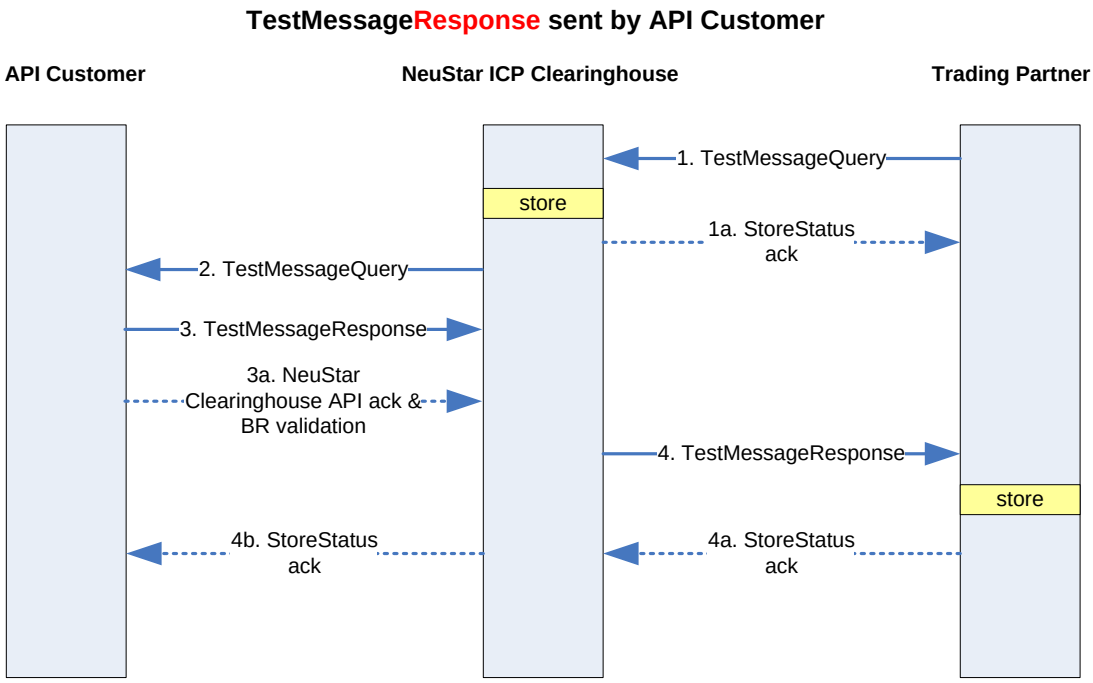
Parameter	Description	Valid Values
SERVER_STATUS	Status of the server	Required A - ICP System Outage - Full: All SPIDs belonging to this provider B - ICP System Outage - Partial: One or more SPIDs but not all belonging to this provider C - Billing System Outage – Full: All billing system functions D - Billing System Outage – Partial: Some billing system functions such as for one SPID or Region E - Billing system maintenance scheduled – Full F - Billing system maintenance scheduled - Partial G - Network Outage – Full: Entire network is down H - Network Outage - Partial: Some network pieces are down such as connectivity to call center I - Clearinghouse Outage – Full: Clearinghouse is down J - Clearinghouse Outage - Partial: Clearinghouse is down K - Slow response L - Not Supported: Provider does not utilize broadcast messages M - Maintenance Mode - Scheduled: During stated maintenance windows such as NPAC maintenance N - Normal Operations U - Unavailable: No answer is received from the provider
END_MAINTENANCE	Date and time maintenance will end	mm-dd-yyyy-hhmm[AM|PM]

5.6.8.2 XML Example - TestMessageResponse

```
<?xml version="1.0" encoding="UTF-8"?>
<wireless_nport>
        <Header>
                <MessageType value="TestMessageResponse"/>
                <TradingPartner value="6604"/>
                <Timezone value="EST"/>
                <Sender value="6017"/>
                <Receiver value="6604"/>

                <WICIS_REL_NO value="4.0.0"/>
        </Header>
        <Body>
                <TestMessageResponse>
                        <DTSENT  value="05-15-2008-1000AM"/>
                        <SENDER_SPID  value="6017"/>
                        <DESTINATION_SPID  value="6604"/>
                        <SERVER_STATUS  value="A"/>
                        <END_MAINTENANCE  value="05-16-2008-1044AM"/>
                </TestMessageResponse>
        </Body>
</wireless_nport>
```

5.6.8.3 Flow Chart - TestMessageResponse



5.6.9 StoreStatus Message

StoreStatus messages are generated by both the ONSP and the NNSP, for messages that are both sent and received by the providers. The MessageSubType value in the header is one of the following values:

- ack – message received successfully
- negack – message not received successfully

5.6.9.1 Parameters - StoreStatus

Table 35: StoreStatus Parameters

Parameter	Description	Valid Values
REQNO	Request number	Max length: 16
VERIDEQ	Version identification of the message	Max length: 2
ONSP	Old Network Service Provider SPID of the facility-based provider	Max length: 4 Required
NNSP	New Network Service Provider SPID of the facility-based provider	Max length: 4 Required
RESPNO	Response Number	Alphanumeric

Parameter	Description	Valid Values
STATUS	Indicates success or failure	0 = failure. MessageSubType value is negack. 1 = success. MessageSubType value is ack.
DTSENT	Date and time that the request was sent from the NSP	mm-dd-yyyy-hhmm[AM|PM]
exceptioncontainer	EXCEPTION_CODE EXCEPTION_MESSAGE	EXCEPTION_CODE - Code identifying the reason for a message failure. Max length: 64. ○ InputDataMissingError ○ InputDataValidationError ○ InputDataFormatError ○ ResourceLimitation EXCEPTION_MESSAGE – information about the exception. Max length: 160

5.6.9.2 XML Example – StoreStatus: ack

```
<?xml version="1.0" encoding="UTF-8"?><wireless_nport>
<Header>
<MessageType value="StoreStatus"/>
<Timezone value="UTC"/>
<CORRELATION_ID
value="X088000245458100000000000000000000000000000000000000000000000270"/>
<MessageSubType value="ack"/>
<TradingPartner value="S004"/>
</Header>
<Body>
<StoreStatus>
<REQNO value="X088208115000010"/>
<VERIDREQ value="01"/>
<ONSP value="S004"/>
<NNSP value="X088"/>
<STATUS value="1"/>
<DTSENT value="04-24-2008-0803PM"/>
</StoreStatus>
</Body>
</wireless_nport>
```

5.6.9.3 XML Example – StoreStatus - negack

```
<?xml version="1.0" encoding="UTF-8"?>
<wireless_nport>
        <Header>
                <MessageType value="StoreStatus"/>
                <TradingPartner value=""/>
                <Timezone value="UTC"/>
                <Sender value=""/>
                <Receiver value=""/>
                <CORRELATION_ID value="44aabbccdd66"/>
                <MessageSubType value="negack"/>
        </Header>
        <Body>
```

```
            <StoreStatus>
                  <REQNO value="NN12632132345AB"/>
                  <VERIDREQ value="01"/>
                  <ONSP value="UIOL"/>
                  <NNSP value="44EE"/>
                  <RESPNO value="1234189345as"/>
                  <STATUS value="0"/>
                  <DTSENT value="12-31-2004-0444PM"/>
                  <exceptioncontainer type="container">
                        <exception>
                              <EXCEPTION_CODE value="InputDataMissingError"/>
                              <EXCEPTION_MESSAGE value="PORTEDNUM"/>
                        </exception>
                  </exceptioncontainer>
            </StoreStatus>
      </Body>
</wireless_nport>
```

5.6.10 ValidationStatus

ValidationStatus messages are generated by both the ONSP and the NNSP, for messages that are both sent and received by the providers. The MessageSubType value in the header is one of the following values:

- valid – the received message's structure and business rules meet the requirements of WICIS 4.0.
- invalid – the received message's structure and business rules fail to meet the requirements of WICIS 4.0.

5.6.10.1 Parameters – ValidationStatus

Table 36: ValidationStatus Parameters

Parameter	Description	Valid Values
REQNO	Request number	Max length: 16
VERIDEQ	Version identification of the message	Max length: 2
ONSP	Old Network Service Provider SPID of the facility-based provider	Max length: 4
NNSP	New Network Service Provider SPID of the facility-based provider	Max length: 4
RESPNO	Response Number	Alphanumeric
STATUS	Indicates success or failure	0 = failure. MessageSubType value is negack. 1 = success. MessageSubType value is ack.
DTSENT	Date and time that the request was sent from the NSP	mm-dd-yyyy-hhmm[AM|PM]

Parameter	Description	Valid Values
exceptioncontainer	EXCEPTION_CODE EXCEPTION_MESSAGE	EXCEPTION_CODE - Code identifying the reason for a message failure. Max length: 64. ○ InputDataMissingError ○ InputDataValidationError ○ InputDataFormatError ○ ResourceLimitation EXCEPTION_MESSAGE – information about the exception. Max length: 160

5.6.10.2 XML Example – ValidationStatus: valid

```
<?xml version="1.0" encoding="UTF-8"?><wireless_nport>
<Header>
<MessageType value="ValidationStatus"/>
<Timezone value="UTC"/>
<CORRELATION_ID
value="X088000245458500000000000000000000000000000000000000000000000000448"/>
<MessageSubType value="valid"/>
<TradingPartner value="S004"/>
</Header>
<Body>
<ValidationStatus>
<REQNO value="X08800811900006A"/>
<VERIDREQ value="01"/>
<ONSP value="S004"/>
<NNSP value="X088"/>
<RESPNO value="S004008119311206"/>
<STATUS value="1"/>
<DTSENT value="04-28-2008-0710PM"/>
</ValidationStatus>
</Body>
</wireless_nport>
```

5.6.10.3 XML Example – ValidationStatus: invalid

```
<?xml version="1.0" encoding="UTF-8"?>
<wireless_nport>
        <Header>
                <MessageType value="ValidationStatus"/>
                <TradingPartner value=""/>
                <Timezone value="UTC"/>
                <Sender value=""/>
                <Receiver value=""/>
                <CORRELATION_ID value="44aabbccdd66"/>
                <MessageSubType value="invalid"/>
        </Header>
        <Body>
                <StoreStatus>
                        <REQNO value="NN12632132345AB"/>
                        <VERIDREQ value="01"/>
                        <ONSP value="UIOL"/>
```

```
                        <NNSP value="44EE"/>
                        <RESPNO value="1234189345as"/>
                        <STATUS value="0"/>
                        <DTSENT value="12-31-2004-0444PM"/>
                        <exceptioncontainer type="container">
                                <exception>
                                        <EXCEPTION_CODE value="InputDataMissingError"/>
                                        <EXCEPTION_MESSAGE value="PORTEDNUM"/>
                                </exception>
                        </exceptioncontainer>
                </StoreStatus>
        </Body>
</wireless_nport>
```

5.7 Pre-Port Validation

The Clearinghouse can check your ability to provide service for a ported telephone number. This is done in two ways:

- Port In- You use the gateway to determine whether you are able to provide service to a telephone number prior to the number being ported to you. The old provider is either a wireless or wireline provider.
- Port Out- You use the gateway to determine whether to allow the porting of a telephone number to another provider. This is done upon receipt of an intent to port message.

♪ NOTE: The header XML conforms to the Clearinghouse Basic Service header.

♪ NOTE: The PPV gateway is synchronous- the processSync SOAP call is used to submit a request to the Gateway.

5.7.1 Example Request XML:

Below is a sample Pre-Port Validation XML request.

```
<?xml version="1.0" encoding="UTF-8" ?>
<PrePortValidationRequest>
 <RequestType value=”PortIn”/>
 <TN value=”510-500-1170”/>
 <Network value=”TDMA”/>
</PrePortValidationRequest>

<?xml version="1.0"?>
<PrePortValidationRequest>
 <RequestType value=”PortOut”/>
 <TN value=”510-500-1170”/>
</PrePortValidationRequest>
```

5.7.2 Example Response XML:

Below is a sample Pre-Port Validation XML response.

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<PrePortValidationResponse>
 <RequestType value=”PortIn”/>
 <TNResponsecontainer type=”container”>
  <TNResponse>
   <TN value=”510-500-1170”/>
   <ResponseType value=”portable”/>
   <CurrentCarrier value=”7890”/>
   <CurrentCarrierName value=”Verizon Wireless”/>
   <CurrentCarrierType value=”wireless”/>
   <RateCenter>
    <RC value=”BERKELEY”/>
          <State value=”CA”/>
   </RateCenter>
   <RateCenterCSA value=”ABERDON”/>
   <Zip value=”94708”/>
   <LRN value=”123-456-7890”/>
  </TNResponse>
 </TNResponsecontainer>
</PrePortValidationResponse>
```

5.8 Response Re-flow

The Clearinghouse provides the ability to re-send responses to the Customer. A Re-flow Gateway was created to allow you to query for responses received from the supplier. The input to this gateway is a query in XML and the output is all the responses received.

♪ NOTE: The Response Re-flow capability exists for LSR Order and PreOrder order types. Based on the SubRequest value in the header, the response is for either LSR Order or PreOrder.

5.8.1 Request Format

The request for the Re-flow is Gateway XML, which has the following header and body elements:

```
<Header>
 <Request value=”query-lsr-responses”/>
 <Subrequest value=”query-lsr-order | query-lsr-preorder”/>
 <Supplier value=”NEUSTAR”/>
 <UserIdentifier value=”example”/>
 <CustomerIdentifier value=”ACME”/>
 <UserPassword value=”example”/>
 <ServerName value=”Nightfire.SPI.NEUSTAR.query-lsr-responses”/>
<InterfaceVersion value=”1_0”/>
</Header>

<body>
   <Info>
      <ServiceType value=”lsr”/>
      <!—Possible values of service type for lsr orders
Loop|LSNP|NP|Resale|Resale_private_line|Port|DL|DS|Platform →
     <!—Possible values of service type for pre orders TN|Appt|FS|AV|CSR|LQ→
      <PON value="<List of PONS>”/>  // for lsr order…
     <!—or →
     <TXNUM value=”<List of TXNUMS”/> // for pre order….
     <DateTimeFrom value=”MM/DD/YYYY”/>
     <DateTimeTo value=”MM/DD/YYYY”/>
     <Supplier value=”<List of Suppliers>”/>
   </Info>
<body>
```

♪ NOTE: PON may also be a list of PONs separated by |. DateTime fields require the date in the MM/DD/YYYY format. A value of DateTime.From gets the responses received after the date. A value of DateTime.To limits the responses to those received between the dates. Absence of PON, DateTime.From, DateTime.To, or Supplier means that field is not considered query criteria. The values of suppliers may be either a fixed supplier or a list of suppliers separated by |.

♪ NOTE: Response Re-flow requests invoke the processAsync SOAP call.

5.8.2 Response Format

The outputs to the request are responses that were split into individual messages for each type of response for the given PON as follows:

```
<Envelope>
 <Response>
  <lsr_order_response>
   <ResponseType value=”focaccept”/>
   <focaccept/>
  </lsr_order_response>
  <SupplierLSROrderResponse>
   <focaccept/>
  </SupplierLSROrderResponse>
 </Response>
</Envelope>
```

5.9 Provider Notifications

Provider notifications are received unsolicited from an ILEC trading partner, via the LSR Order Gateways. These notifications are logged and posted to a different event channel – not the same event channel to which gateway responses are posted.

Rather, the SOAP client must listen on the new Provider Notification event channels so that for API customers, these notifications are delivered to their upstream system via the SOAP service.

Provider Notification messages are identified via the Response.lsr_order_response.ResponseType node. The value of this node is: “provider_notification”.

Example Provider Notification XML message:

```
<?xml version=”1.0” encoding=”UTF-8”?>
<Response>
 <lsr_order_response>
  <ResponseType value=”provider_notification”/>
  <provider_notification>
   <pncontainer type=”container”>
    <pn>
     <INDEX value=”1”/>
     <NT value=”A”/>
     <NTA value=”NTA1”/>
     <WTN value=”111-111-1111”/>
     <CVD value”06-29-2004”/>
     <RDY value=”1”/>
     <ECCKT value=”1”/>
     <TNC value=”111-344-5555”/>
    </pn>
```

```
   </pncontainer>
  </provider_notification>
 </lsr_order_response>
</Response>
```

5.10 Queries

The Clearinghouse supports a set of API-enabled query functionality that allows you to report on various aspects of your orders. This section details the query requests performed for Basic Services. The query message API is invoked using the following header XML values:

Table 37: Query Request and Subrequest valid values

Header Node Name	Value
Request value	query-messages
Subrequest value	• query-orders • query-order-history • query-transaction-details • query-transactions • query-order-history-997
Supplier value	NEUSTAR

5.10.1 Query Requests

NeuStar supports the following subrequest values for query requests:

- **query-orders –** This query is used to see orders that meet a certain specified search criteria. It returns orders whose most recent message corresponds to the value supplied for the MessageType data element. For an example of this query see ***Appendix A***.

- ***QUERY-ORDER-HISTORY*** – This query is used to see all of the transactions that make up the lifecycle of a particular order. It returns non XML columns for messages which match the search criteria. For an example see ***Appendix A.***

- **query-transaction-details** – This query is used to find out more detailed information about a particular transaction. With the MessageKey value (for LIDB messages, use LIDBMessageKey) returned from a query-orders response, you can submit a request to see the corresponding structure for the request or response XML, exposing the actual details of the transaction. For an example of this query see ***Appendix A***.

- **query-transactions –** This query is used to search for transactions of a certain type. This query is available for LSR Order, LSR Preorder, ICP Port In, ICP Port Out, ICP Maintenance, and Provider Notifications only.

- **query-order-history-997-** This query returns all the information for 997 acknowledgement for an order. This query is available for LSR Order, ICP Port In, ICP Port Out, and ICP Maintenance.

To use the query-transaction-details subrequest query with the CustomerUse field to find details of matching LSR orders, complete the following steps:

1. Run a query-orders subrequest using the CustomerUse request data element. This query will return the PON of the matching order.

2. Run a query-transaction-details subrequest. For the request data element, use the PON returned by the query-orders subrequest. This query will return the order details of the matching order.
3. Continue this process for the remaining PONs returned by the query-orders subrequest query.

5.10.1.1 Query DTD

The query message API is invoked using the following request XML values:

```
<Body>
 <Info>
  <ServiceType value="Loop"/>
  <SortBy value="PON"/>
  <SortByOrdering value="DESC"/>
  <EndOffset value="9"/>
  <StartOffset value="0"/>
 </Info>
</Body>
```

Table 38 defines the valid values for the above data elements in the request XML.

Table 38: Query Request Data Elements

Data Element	Description	Valid Values		
ServiceType	Identifies the type of message for which the search is performed	loop np lsnp port platform resale resale_private_line ds dl simple_port resale_frame_relay centrex_resale ddps All *(not valid for query transaction details)*	AV CSR LQ TN Appt FS SOIP FAI CFAI CancelReservation E911 LIDBRequest ICPPortIn ICPPortOut TestMessageQuery TestMessageResponse BroadcastNotification	ProviderNotification PPVPortIn PPVPortOut ProviderNotification ASRPreOrder ASROrder Transport Trunking FGA WATS Ring VC EVC
SortBy	Identifies the field to sort by. This should not correspond to a field listed in the service section below	(name of any data row selected in the query request) ♪ NOTE: The "SortBy" value should correspond to fields that are returned in the response XML, rather than the request XML.		
SortByOrdering	Defines whether to sort in ascending or descending order	ASC DESC		
EndOffset	You can control the number of orders returned via the Start and End offset. If the query returns 10,000 rows and you only want to remove the first 10 then set EndOffset = 9, and StartOffset =0 Offset is 0 based. StartOffset should be less than EndOffset	1-999999		

Data Element	Description	Valid Values
StartOffset	You can control the number of orders returned via the Start and End offset. If the query returns 10,000 rows and you only want to remove the first 10 then set EndOffset = 9, and StartOffset =0 Offset is 0 based. StartOffset should be less than EndOffset	0-999999

The value of ServiceType defines the gateway-logging table being queried. The value of “All” for ServiceType bases the query on a combination of all LSR Order service types, specifically:

- loop
- np
- lsnp
- port
- platform
- resale
- resale_private_line
- ds
- dl
- simple_port
- centrex_resale
- resale_frame_relay
- dps

5.10.1.2 The Wildcard Construct

The wildcard construct is available for data element values. For example, to obtain all LSR Loop orders with PON value beginning with MS, enter PON value as:

<PON value=”MS%”/>

5.10.1.3 The “or” Construct

Use the ‘or’ (|) construct in the data element value to obtain messages matching multiple values. For example, to obtain all LSR Loop orders with PON value of MS% or BS%, enter PON value as:

<PON value=”MS%|BS%”/>

The following data elements can use the ‘or’ (|) construct:

Table 39: OR Construct Data Elements

Supplier	TXACT	OrderNumber	PON
WTN	MessageType	ReferenceKey	RegionId
MessageKey	Tn	TelephoneNumber	CreatedBy
OldSP	TXNUM	NewSP	SvId

TXTYP	Lrn	VER	

5.10.1.4 Additional Data elements

Additional data elements are supplied to refine the query results depending on the type of query (defined in the Subrequest value in the header XML) and the ServiceType value supplied in the request XML.

5.10.1.4.1 Query-Orders Requests

Table 40 details these additional XML elements for the query-orders requests.

Table 40: Request XML Elements for query-orders

Service Type	Request Data Element	Valid Values
Loop NP LSNP Port Platform Resale Resale_private_line DS DL All lsr_order simple_port resale_frame_relay centrex_resale ddps	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	Region	
	PON	
	VER	
	REQTYP	
	ACT	
	LSRNO	
	DIRACT	
	SubmittingUserId	
	LastSubmittedByUserId	
	CreatedByUserId	
	ECCKT	
	CKR	
	DUEDATE.From	
	DUEDATE.To	
	EUNAME	
	ATN.From	
	ATN.To	
	ApplyBusinessRules	
	CustomerUse	XXXX. Wildcard % and Logical OR | are allowed. A maximum of 100 alphanumeric characters. XML special characters (&, ', ", >, <) not allowed; other special characters permitted.
	MessageType	• New • ack • negack • focaccept • focreject • jeopardy • supplement • fax_ack • fax_nack • suppaccept • suppreject • soc • billing_completion • dsrcn • dsred • provider_initiated_activity • provider_initiated_cancel
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]

Service Type	Request Data Element	Valid Values	
	Action	• submit • receive • save	• suspend • resume • cancel • abandon
AV	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	
	InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	
	Region		
	TXNUM		
	TXTYP		
	TXACT		
	USERID		
	WTN		
	ApplyBusinessRules	• true • false	
	MessageType	• request • addressMatch	• addressError • addressAlternatives
	Action	• submit • receive • save	• suspend • resume • cancel • abandon
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]	
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]	
CSR	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	
	InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	
	Region		
	TXNUM		
	TXTYP		
	TXACT		
	USERID		
	WTN		
	ApplyBusinessRules	• true • false	
	MessageType	• request • parsed_response	• unparsed_response • response • **error_response (**New in OMS Clearinghouse 3.9)
	Action	• submit • receive • save	• suspend • resume • cancel • abandon
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]	
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]	

Service Type	Request Data Element	Valid Values
LQ	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	Region	
	TXNUM	
	TXTYP	
	TXACT	
	USERID	
	WTN	
	ApplyBusinessRules	• true • false
	MessageType	• request • response • error_response
	Action	• submit • receive • save • suspend • resume • cancel • abandon
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
TN Appt FS	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	Region	
	TXNUM	
	TXTYP	
	TXACT	
	USERID	
	WTN	
	ApplyBusinessRules	• true • false
	MessageType	• request • response
	Action	• submit • receive • save • error_response • suspend • resume • cancel • abandon
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
SOIP	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	InterfaceVersion	
	Region	
	TXNUM	
	TXTYP	

Service Type	Request Data Element	Valid Values	
	TXACT		
	USERID		
	WTN		
	ApplyBusinessRules	• true • false	
	MessageType	• request • response	
	Action	• submit • receive • save	• suspend • resume • cancel • abandon
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]	
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]	
FAI CFAI CancelReservation	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	
	InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	
	Region		
	TXNUM		
	TXTYP		
	TXACT		
	USERID		
	WTN		
	MessageType	• request • response	
	Action	• submit • receive • save	• suspend • resume • cancel • abandon
	ApplyBusinessRules	• true • false	
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]	
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]	
asr_preorder	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	
	InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	
	MessageType		
	MessageId		
	ICSC		
	CCNA		
	CC		
	STATE		
	IRI		
	IRM		
	ApplyBusinessRules		
	Action		
	SubmittingUserId		

Service Type	Request Data Element	Valid Values
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
ASROrder Transport FGA Trunking WATS Ring VC EVC	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	ICSC	
	SEQ	
	PON	
	VER	
	CCNA	
	ACT	
	ASRNO	
	SUP	
	SubmittingUserId	
	LastSubmittedByUserId	
	CreatedByUserId	
	ATN	
	ECCKT	
	ISSNO	
	CLO	
	ECID	
	DDD.From	
	DDD.To	
	INIT	
	ATN.From	
	ATN.To	
	Action	• submit • receive • save • suspend • resume • cancel • abandon
	ApplyBusinessRules	
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
	CustomerUse	XXXX. Wildcard % and Logical OR | are allowed. A maximum of 100 alphanumeric characters. XML special characters (&, ', ", >, <) not allowed; other special characters permitted.
	MessageType	• Ack • Negative Ack • secondaryconfirmation • Confirmation – B • Confirmation – D • Confirmation – F • Confirmation – S • Clarification • fax_ack • fax_nack • DLR • Error
E911	OrderNumber	
	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	TelephoneNumber	

Service Type	Request Data Element	Valid Values
	InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	MessageType	• InsertRequest • DeleteRequest • ChangeRequest • UnlockRequest • MigrateRequest • ConfirmationResponse • ErrorResponse • RejectResponse
	Action	• submit • receive • save • suspend • resume • cancel • abandon
	ApplyBusinessRules	• true • false
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
LIDBRequest	InterfaceVersion	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	CustomerName	XXX. Wildcard % and Logical OR | are allowed. LIDB requests / Calling Name. A maximum of 15 alphanumeric characters are allowed.
	TelephoneNumber	
	MessageType	When ServiceType=LIDBRequest • InsertLineNumberRecord • ChangeLineNumberRecord • DeleteLineNumberRecord
	Action	• submit • receive • save • suspend • resume
	ApplyBusinessRules	• true • false
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
ProviderNotification		

Service Type	Request Data Element	Valid Values
PPVPortIn PPVPortOut	Telephone Number	
	SubmittingUserID	
	ResponseType	PPVPortIn: • portable • tn-not-portable • exchange-not-found • no-agreement-with-carrier • you-are-the-carrier • no-service-within-rate-center •
		PPVPortOut: • portable • tn-not-portable • exchange-not-found • you-are-not-the-carrier
	Datetime.From	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
	Datetime.To	A date and time. Valid format:s: MM-dd-yyyy-hhmm[AM|PM] Or MM-DD-HH-MM[past|future]
ICP	ServiceType	ICPPortIn ICPPortOut
	Action	receive submit save
	MessageSubType	create modify dueDateChange cancel ack negack valid invalid resolutionRequired delay confirm
	TelephoneNumber	From To
	CreatedByUserId	
	LastSubmittedByUserId	
	MessageType	MultiPortRequest MultiPortResponse ModifyPortRequest ValidationStatus StoreStatus
	Supplier	
	FirstName	
	AccountNumber	
	DueDate	to from
	Datetime	to from
	BusinessName	
	LastName	
	RequestNumber	
ICPMaintenance	MessageSubType	ack negack
	MessageType	TestMessageQuery TestMessageResponse BroadcastNotification StoreStatus-BroadcastNotification StoreStatus-TestMessageResponse StoreStatus-TestMessageQuery
	SERVER_STATUS	

Service Type	Request Data Element	Valid Values
	BEGIN_MAINTENANCE	From To
	END_MAINTENANCE	From To
	UserId	
	ServiceType	ICPAdmin Note: The ServiceType for Query-Orders and Query-Transactions is ICPAdmin, **not** TestMessageQuery, TestMessageResponse, and BroadcastNotification individually.
	SENDER_SPID	
	Supplier	
	Datetime	from to
	Action	save submit receive

5.10.1.4.2 Query-Transaction-Details Requests

Table 41 details the data elements for query-transaction-details requests.

Table 41: Request XML Elements for query-transaction-details

Service Type	Request Data Element	Valid Values	
Loop NP LSNP Port Platform Resale Resale_private_line DS DL All lsr_order simple_port resale_frame_relay ddps centrex_resale	Row Key	Value is obtained from a previous query-orders response.	
	MessageType	• new • ack • negack • focaccept • focreject • jeopardy • supplement • fax_ack • fax_nack	• suppaccept • suppreject • soc • billing_completion • dsrcn • dsred • status • provider_initiated_activity • provider_initiated_cancel
ASROrder ASRPreOrder AV CSR LQ TN Appt FS SOIP FAI CFAI CancelReseervation ProviderNotification PPVPortIn PPVPortOut	MessageKey	Value is obtained from a previous query-orders response.	
LIDBRequest	LIDBMessageKey	Value is obtained from a previous query-orders response.	
PortIn PortOut IntraPort	MessageKey	Value is obtained from a previous query-orders response.	
	ReferenceKey	Value is obtained from a previous query-orders response.	
E911	MessageKey	Value is obtained from a previous query-orders response.	
	MessageType	OutstandingErrorNotification	

Service Type	Request Data Element	Valid Values
ICP	ServiceType	ICPPortIn ICPPortOut
	Direction	In Out
	MessageType	MultiPortRequest MultiPortResponse ModifyPortRequest ValidationStatus StoreStatus
	MessageSubType	create modify dueDateChange cancel ack negack valid invalid resolutionRequired delay confirm
	RequestNumber	
	Supplier	
	MessageKey	
	Datetime	
ICPMaintenance	ServiceType	TestMessageQuery TestMessageResponse BroadcastNotification
	MessageType	TestMessageQuery TestMessageResponse BroadcastNotification
	Supplier	
	MessageKey	
	CORRELATIONID	

5.10.1.4.3 Query-Order-History Requests

Table 42 details the XML elements for the query-order-history request.

Table 42: Request XML Elements for query-order-history

Service Type	Request Data Element	Valid Values
Loop NP LSNP Port Platform Resale Resale_private_line DS DL All lsr_order simple_port resale_frame_relay ddps centrex_resale	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	PON	
AV CSR LQ TN Appt FS SOIP	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	TXNUM	
FAI CFAI	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	TXNUM	

Service Type	Request Data Element	Valid Values
CancelReservation	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	TXNUM	
ASRPreOrder	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	MessageId	
ASROrder FGA Transport Trunking WATS Ring VC EVC	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	PON	
E911	TelephoneNumber	
	Supplier	
LIDBRequest	TelephoneNumber	
PortIn PortOut IntraPort	ReferenceKey	
PPVPortIn PPVPortOut	MessageKey	
ICP	ServiceType	ICPPortIn ICPPortOut
	RequestNumber	
	Supplier	
ICPMaintenance	ServiceType	TestMessageQuery TestMessageResponse BroadcastNotification
	MessageType	TestMessageQuery TestMessageResponse BroadcastNotification
	MessageSubType	BroadcastNotification-Out|BroadcastNotification-In TestMessageQuery-Out|TestMessageQuery-In TestMessageResponse-Out|TestMessageResponse-In
	Supplier	
	CORRELATIONID	

5.10.1.4.4 Query-Transactions Requests

Table 43 details the XML elements available for query-transactions

Table 43: Request XML Elements for query-transactions

Service Type	Request Data Element	Valid Values
Loop NP	REQTYP	
	Supplier	See

Service Type	Request Data Element	Valid Values
LSNP Port Platform Resale Resale_private_line DS DL simple_port resale_frame_relay ddps centrex_resale All lsr_order		https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	PON	
	VER	
	MessageType	• new • ack • negack • focaccept • focreject • jeopardy • supplement • fax_ack • fax_nack • suppaccept • suppreject • soc • billing_completion • dsrcn • dsred • provider_initiated_activity • provider_initiated_cancel • info
	Datetime.From	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]
	Datetime.To	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]
	EUNAME	
	ACT	
	ECCKT	
	CKR	
	USERID	
	ATN	
	Action	• submit • receive • save • suspend • resume • cancel • abandon
AV	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	MessageType	• request • addressMatch • addressAlternatives • addressError
	Datetime.From	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]
	Datetime.To	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]
	Action	• submit • receive • save • suspend • resume • cancel • abandon
	TXNUM	
	TXTYP	
	TXACT	
	USERID	
	WTN	
CSR	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	MessageType	• request • parsed_response • unparsed_response • response

Service Type	Request Data Element	Valid Values	
		• error_response	
	TXNUM		
	Datetime.From	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]	
	Datetime.To	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]	
	Action	• submit • receive • save	• suspend • resume • cancel • abandon
	TXTYP		
	TXACT		
	WTN		
	USERID		
LQ TN Appt FS	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	
	MessageType	• request • response • error_response	
	TXNUM		
	Datetime.From	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]	
	Datetime.To	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]	
	Action	• submit • receive • save	• suspend • resume • cancel • abandon
	TXTYP		
	TXACT		
	WTN		
	USERID		
Provider Notification	Supplier		
	NT		
	NTA		
	ECCKT		
	WTN		
	Datetime.From	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]	
	Datetime.To	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]	
SOIP	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf	
	MessageType	• request • response • error_response	
	Datetime.From	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]	
	Datetime.To	A date and time. Valid formats: MM-dd-yyyy-hhmm[AM|PM] or MM-DD-HH-MM[past|future]	
	Action	• submit • receive	• suspend • resume

Service Type	Request Data Element	Valid Values
		• save • cancel • abandon
	TXNUM	
	TXTYP	
	TXACT	
	USERID	
	WTN	
ICP	ServiceType	ICPPortIn ICPPortOut
	RequestNumber	
	Action	submit receive save
	Direction	In Out
	MessageSubType	create modify dueDateChange cancel ack negack valid invalid resolutionRequired delay confirm
	TelephoneNumber	From To
	MessageType	MultiPortRequest MultiPortResponse ModifyPortRequest ValidationStatus StoreStatus
	Suppler	
	Due Date	To From
	SortByOrdering	
ICPMaintenance	SortBy	
	MessageType	TestMessageQuery TestMessageResponse BroadcastNotification StoreStatus-BroadcastNotification StoreStatus-TestMessageResponse StoreStatus-TestMessageQuery
	UserId	
	ServiceType	ICPAdmin Note: The ServiceType for Query-Orders and Query-Transactions is ICPAdmin, **not** TestMessageQuery, TestMessageResponse, and BroadcastNotification individually.
	SENDER_SPID	
	Supplier	
	Action	submit receive save
	MessageSubType	ack negack
	SERVER_STATUS	
	BEGIN_MAINTENANCE	
	Datetime	
	CORRELATIONID	

5.10.1.4.5 Query-Order-History-997 Requests

Table 44 contains details on XML elements for query-order-history-997 requests.

Table 44: Request XML Elements for query-order-history-997

Service Type	Request Data Element	Valid Values
Loop NP LSNP Port Platform Resale Resale_private_line DS DL simple_port resale_frame_relay ddps centrex_resale All lsr_order	Supplier	See https://www.neustar.biz/convergentCH/content/docs/Basic_Services_-_Supplier_and_Interface_Version_Valid_Values.pdf
	PON	
ICP	ServiceType	ICPPortIn ICPPortOut
	RequestNumber	
	Supplier	
ICPMaintenance	ServiceType	BroadcastNotification TestMessageQuery TestMessageResponse
	MessageType	storestatus
	MessageSubType	BroadcastNotification-Out|BroadcastNotification-In TestMessageQuery-Out|TestMessageQuery-In TestMessageResponse-Out|TestMessageResponse-In
	CORRELATIONID	

5.10.2 Query Responses

Once a query request is submitted, a corresponding response is returned.

Table 45 details query message response fields, contained in the response XML.

Table 45: Query Message Response Fields

XML Structure			Description	Usage and Valid Values
Element	Contains	Field		
Body				
	Info			
		StartOffset	From the request	Numeric 0-99999999
		EndOffset	From the request	Numeric 1-99999999
		TotalRowCount	The total number of rows returned by the query	Numeric
		Count	The number of rows returned in this response. This would correspond to (EndOffset-StartOffset)+ 1	Numeric

XML Structure			Description	Usage and Valid Values
Element	Contains	Field		
Body				
	Info			
	DataContainer Data (0 or more instances of this section – one for each row returned from the query, corresponding to the offset values provided in the request. Fields listed below are only returned in the response if they have a value			
	If Service Type=Loop, LSNP, NP, Resale, Resale_private_line Port, Platform, DL, simple_port, centrex_resale, resale_frame_relay, ddps, or DS, the following fields are returned for each order:			
		ServiceType		
		PON		
		VER		
		REQTYP		
		ACT		
		ATN		
		NAME		
		DUEDATE		
		USERID		
		Supplier		
		Region		
		InterfaceVersion		
		MessageType		
		Datetime		
		MessageKey		
		Action		
		ApplyBusinessRules		
		Locked		
		LockTime		
		LockOwner		
		Message	Only returned when Subrequest in the header XML =’query-transaction-details’	Request/Response structure corresponding to LSR order request/response XML per the DTDs
		CustomerUse		
	If Service Type=AV or LQ the following fields are returned for each order:			
		Service Type		
		Supplier		
		Region		
		InterfaceVersion		
		MessageKey		
		Datetime		
		MessageType		
		TXNUM		
		TXTYP		
		TXACT		
		USERID		
		Action		
		ApplyBusinessRules		
		Locked		
		LockTime		
		LockOwner		
		Message	Only returned when Subrequest in the header XML =’query-transaction-details’	Request/Response structure corresponding to LSR order request/response XML per the DTDs
	If Service Type=CSR, TN, Appt, FS, SOIP, FAI, CFAI, or CancelReservation the following fields are returned for each order:			
		ServiceType		
		Supplier		
		Region		
		InterfaceVersion		
		Datetime		
		MessageType		
		MessageKey		
		TXNUM		
		TXTYP		
		TXACT		
		USERID		
		Action		

XML Structure			Description	Usage and Valid Values
Element	Contains	Field		
Body				
	Info			
		ApplyBusinessRules		
		Locked		
		LockTime		
		LockOwner		
		Message	Only returned when Subrequest in the header XML ='query-transaction-details'	Request/Response structure corresponding to LSR order request/response XML per the DTDs
	If service Type=E911, the following fields are returned for each order:			
		ServiceType		
		Created		
		MessageType		
		MessageKey		
		ResponseType		
		Supplier		
		OrderNumber		
		TelephoneNumber		
		InterfaceVersion		
		Message	Only returned when Subrequest in the header XML ='query-transaction-details'	Request/Response structure corresponding to E911 request/response XML per the DTDs
	If Service Type=LIDBRequest, the following fields are returned for each order:			
		ServiceType		
		Created		
		MessageType		Maps to RequestType in logging table
		SubrequestType		
		TelephoneNumber		
		LIDBMessageKey		
		LIDBBatchNumber		
		InterfaceVersion		
		Message	Only returned when Subrequest in the header XML ='query-transaction-details'	Request structure corresponding to LIDB request XML per the DTDs. Maps to Record in logging table
	If Service Type=PPVPortIn or PPVPortOut the following fields are returned for each order:			
		UserId		
		TelephoneNumber		
		ResponseType		
		Datetime		
		Message	Only returned when Subrequest in the header XML= 'query-transaction-details'	Response structure corresponding to PPV response XML per the DTDs
	If Service Type = ASRPreorder			
		ServiceType		
		Supplier		
		InterfaceVersion		
		MessageKey		
		Datetime		
		MessageType		
		MessageId		
		ICSC		
		CCNA		
		CC		
		STATE		
		IRI		
		IRM		
		Action		
		ApplyBusinessRules		
		Locked		
		LockTime		
		LockOwner		
		Message	Only returned when Subrequest in the header XML = 'query-transaction-details'	Request or Response structure corresponding to ASR Preorder request or response XML per the DTDs

XML Structure			Description	Usage and Valid Values
Element	Contains	Field		
Body				
	Info			
	If Service Type = ASROrder			
		Service Type		
		PON		
		VER		
		ICSC		
		SEQ		
		CCNA		
		ACT		
		ASRNO		
		SUP		
		CNT		
		RT		
		MSG_CODE		
		RTR		
		Status		
		CD		
		INIT		
		DDD		
		DD		
		ATN		
		ISSNO		
		CLO		
		ECID		
		USERID		
		Supplier		
		InterfaceVersion		
		MessageType		
		Datetime		
		MessageKey		
		Action		
		ApplyBusinessRules		
		CustomerUse		
		Locked		
		LockTime		
		LockOwner		
		Message	Only returned when Subrequest in the header XML = 'query-transaction-details'	Request or Response structure corresponding to ASR order request or response XML per the DTDs
	If Service Type=ICPPortIn, ICPPortOut			
Body				
	DataContainer			
		ServiceType		
		InterfaceVersion		
		RequestNumber		
		MessageType		**ICPPortIn:** MultiPortInrequest MultiPortInresponse-delay MultiPortInresponse-resolutionRequired MultiPortInresponse-confirm ModifyPortInrequest-modify ModifyPortInrequest-cancel ModifyPortInrequest-change-duedate StoreStatus ValidationStatus **ICPPortOut:** MultiPortOutrequest MultiPortOutresponse-delay MultiPortOutresponse-resolutionRequired MultiPortOutresponse-confirm ModifyPortOutrequest-modify ModifyPortOutrequest-cancel

XML Structure			Description	Usage and Valid Values
Element	Contains	Field		
Body				
	Info			
				ModifyPortOutrequest-change-duedate StoreStatus ValidationStatus
		MessageSubType		
		Supplier		
		FirstName		
		BusinessName		
		LastName		
		DueDate		
		AccountNumber		
		ResponseNumber		
		Datetime		
		UserId		
		Action		
		VER		
		Direction		
		MessageKey		
		Locked		
		Message	Only returned when Subrequest in the header XML = 'query-transaction-details'	
	If Service Type=ICP Maintenance			
		ServiceType		
		RequestType		
		Supplier		
		InterfaceVersion		
		MessageType		TestMessageQueryrequest StoreStatus-TestMessageQuery TestMessageResponse StoreStatus-TestMessageResponse BroadcastNotification StoreStatus-BroadcastNotification
		MessageSubType		
		Datetime		
		UserId		
		Action		
		Direction		
		SENDER_SPID		
		BEGIN_MAINTENANCE		
		END_MAINTENANCE		
		SERVER_STATUS		
		MessageKey		

Chapter 6 LSR Integration Overview

This chapter contains general information to support a Clearinghouse integration involving LSR order and Preorder requests.

6.1 Address Validation

Figure 5 illustrates the message flow for a Preorder address validation request sent to an ILEC.

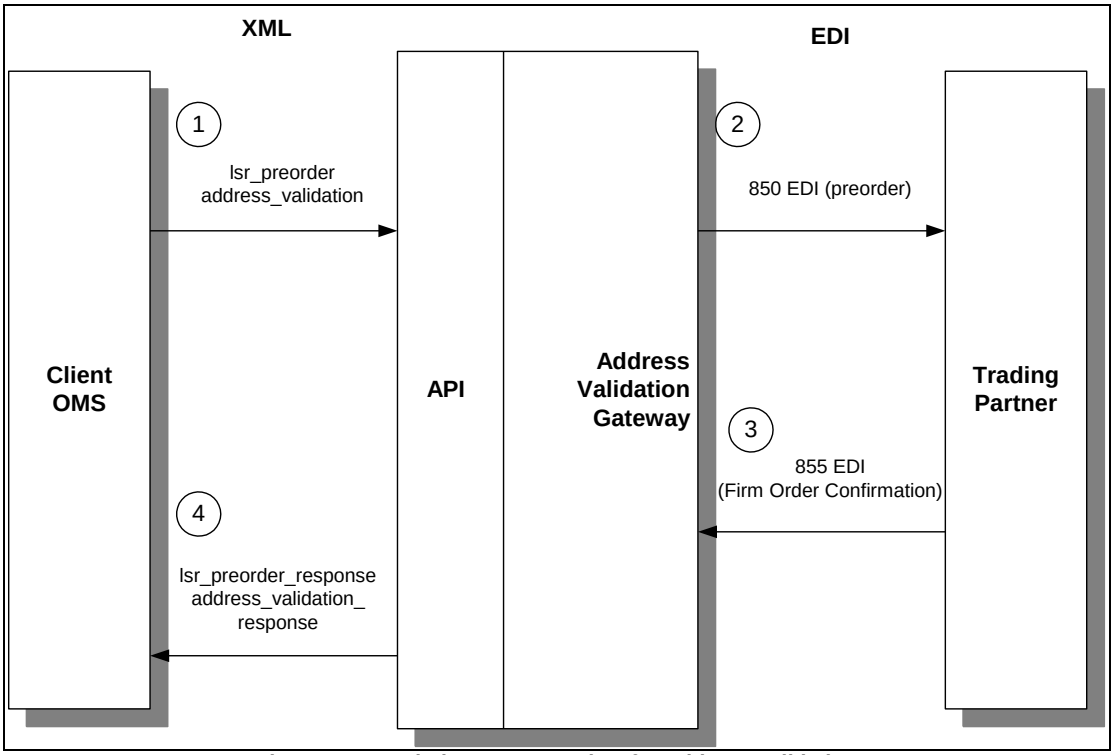


Figure 5: Preordering Message Flow for Address Validation

The address validation request is sent to the Address Validation Gateway (1) and the request is processed by the Gateway, which translates the request into an EDI message for transmission directly to a supplier (2). After looking up the information in its systems, the supplier responds (3) and the SPI processes the response. Depending on the supplier's response, the Gateway then returns either an exact match, address alternatives or invalid address response XML to your OMS (4).

6.2 PreOrder Message Flow

Figure 6 represents the message flow for a Loop Qualification, Customer Service Request (CSR), Telephone Number (TN) Reservation, Appointment Scheduling, Feature/Service Availability , Service Order Inquiry (SOIP), Fiber Availability Inquiry (FAI), Collocation Facility Assignment Inquiry (CFAI), or Cancel Reservation Preorder request sent to an ILEC.

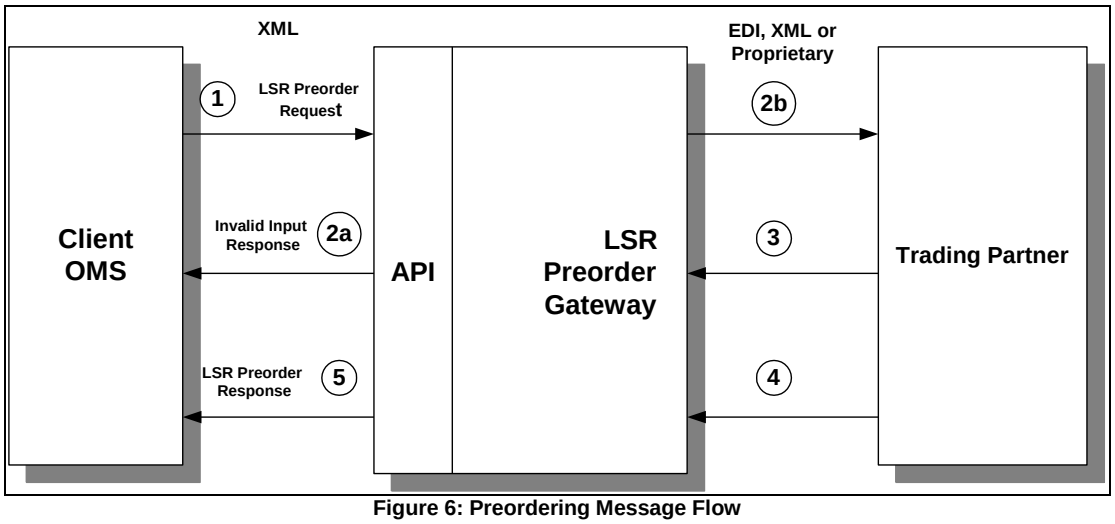


Figure 6: Preordering Message Flow

A Loop Qualification, CSR retrieval, TN Reservation, Appointment Scheduling, or Feature/Service Availability, SOIP, FAI, CFAI, or Cancel Reservation request is sent to the corresponding Gateway (1) and the request is processed by the SPI. If there are any formatting errors or missing required fields in the request, an invalid input response is sent back to your OMS (2a). If the request is properly formatted, the message is forwarded to the supplier (2b). The supplier sends an acknowledgement (3). The supplier then looks up the loop information or customer service record in its systems and returns a response to the Gateway (4). The SPI translates the message and transmits it to your OMS (5).

For Loop Qualification requests, a successful response is returned if the composition of the loop is known. However, the Loop Qualification response may not contain any information about the composition of the Loop, since some ILECs lack data about Loops in some service areas. In this case, a Loop Qualification response of type “error” is returned.

6.3 LSR Order Message Flow

The following diagram illustrates an example interaction among you, the Clearinghouse and a trading partner in the processing of an LSR order. The numbers indicate the sequence of the events.

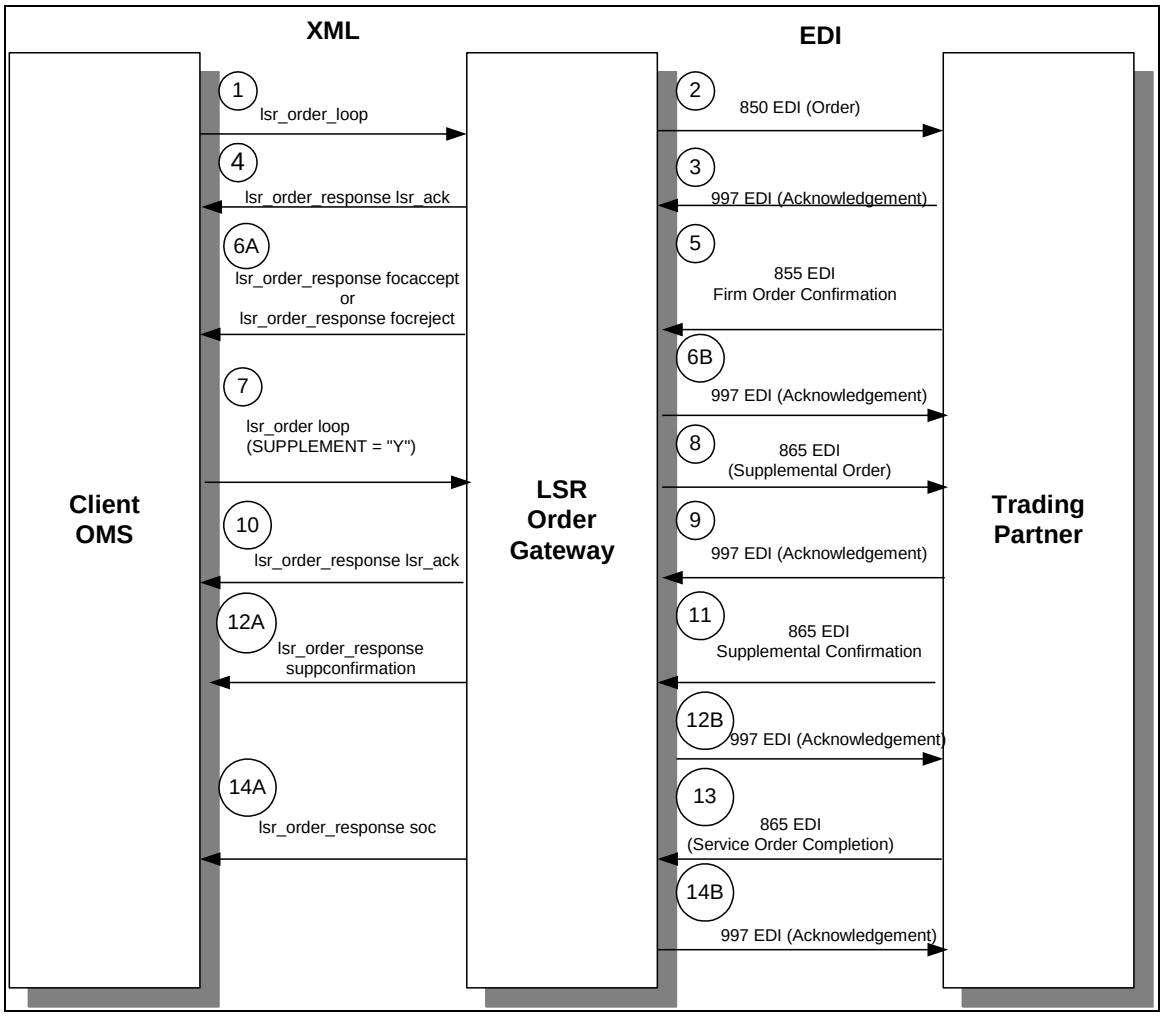


Figure 7: LSR Gateway Order Flow

Although the order flow in ***Figure 7*** is the normal flow adhered to by most trading partners, some do not follow this convention. For example, some trading partners (such as Verizon East) send two SOCs to indicate completion of an order. The Clearinghouse returns to you, all FOCs and SOCs received from a trading partner. So, the general rule to follow when integrating with an LSR Order gateway is to assume that a given order request can receive multiple FOCs and SOCs.

6.4 EDI Message Transition Sets

EDI Transition sets describe bi-directional types of EDI transmission between the LSR Gateway and its trading partners.

6.4.1 EDI Message Transition Set 850

An 850 Electronic Data Interchange (EDI) Transition Set issues an order to the trading partner. In the case of unbundled loop ordering, the order is either a new order, a change order, or a disconnect order.

6.4.2 EDI Transition Set 855

An 855 EDI Transition Set is sent by the trading partner to the LSR Order gateway to either confirm or reject an order submitted via an 850 EDI Transition Set. When the order is being confirmed, the 855 is referred to as a Firm Order Confirmation (FOC).

♪ NOTE: The due date submitted in the order request may be modified in the FOC response.

6.4.3 EDI Transition Set 860

An 860 EDI Transition Set issues a supplement to a previously placed order, for which a Service Order Completion (SOC) has not yet been received. A supplement changes parameters on the original order, corrects errors on the original order, or cancels the original order.

6.4.4 EDI Transition Set 865

An 865 EDI Transition Set is sent by the trading partner to the LSR Order gateway for two purposes:

1. To notify the LSR Order gateway that a previously placed order request was completed. In this case, the 865 Transition Set is referred to as a SOC. The SOC is sent after a FOC, usually on or near the due date indicated in the FOC.
2. To confirm or reject a supplemental order submitted via an 860 EDI Transition Set. When the order is confirmed, the 865 Transition Set is referred to as a FOC.

6.4.5 EDI Transition Set 997

A 997 EDI Transition Set is a hand-shaking mechanism at the transport layer of the EDI interchange. The trading partner sends a 997 to acknowledge receipt of a message from the LSR gateway. Likewise, LSR gateways send 997s to the trading partner to acknowledge receipt of a message from the trading partner.

6.5 Forms-based Ordering

LSR ordering is a forms-based process, meaning several OBF forms must be completed and combined to successfully submit a request to an incumbent carrier. For example, to place an LSNP request with a trading partner, you must complete the forms illustrated in ***Figure 8***.

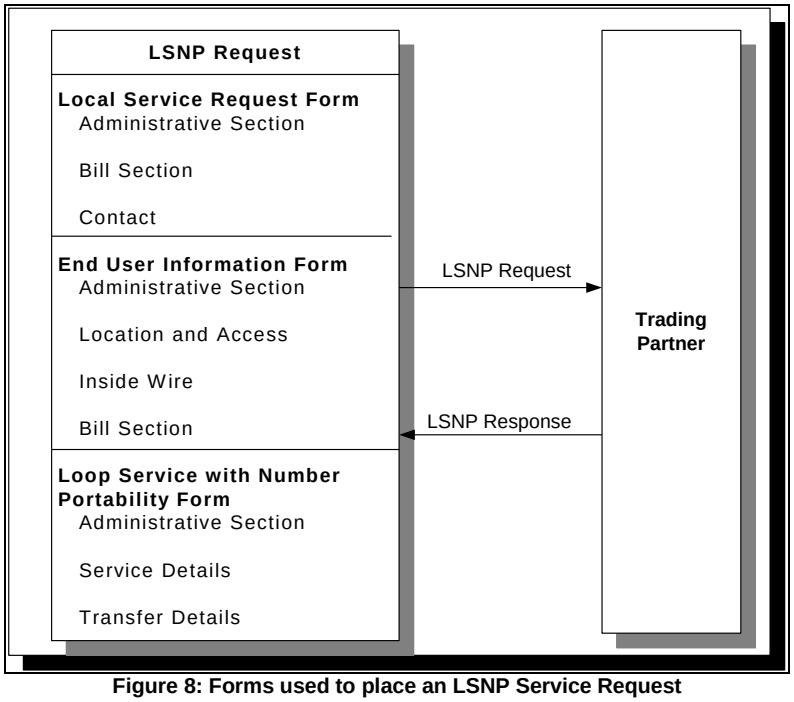


Figure 8: Forms used to place an LSNP Service Request

The required forms vary widely depending on the service type and the trading partner. Platform or directory service requests may require as many as five forms.

Table 46 lists the forms required by each trading partner when ordering LSR-based services (such as voice and loop ordering).

Table 46: LSR - Based Forms

REQTYP	SUPPLIER													
	BS		VZE		VZW		SBC		Qwest		Sprint		Fairpoint	
A	LSR EU	LS DL*	LSR EU	LS DL*	LSR EU	LS* DL*	LSR EU*	LS DL*	LSR EU	LS DL*	LSR EU	LS DL*	LSR EU	LS DL*
B	LSR EU	LSNP DL*	LSR EU	LSNP DL*	LSR EU	LSNP* DL*	LSR EU	LSNP DL*	LSR EU	LSNP DL*	LSR EU	LSNP DL*	LSR EU	LSNP DL*
C	LSR EU	NP DL*	LSR EU	NP DL*	LSR EU	NP* DL*	LSR EU	NP DL*	LSR EU	NP DL*	LSR EU	NP DL*	LSR EU	NP DL*
D	Not Supported		LSR EU PS*	DL* HGI*	Not Supported		Not Supported		Not Supported		Not Supported		LSR EU PS*	DL* HGI*
E	LSR EU RS*	DL* HGI*	LSR EU RS*	DL* HGI*	LSR EU RS*	DL* HGI*	LSR EU*	RS* DL*	LSR EU RS*	DL* HGI*	Not Supported		LSR EU RS*	DL* HGI*
F	LSR EU PS*	DL* HGI*	LSR EU PS*	DL* HGI*	LSR EU	PS* DL*	LSR EU*	PS* DL*	LSR EU PS*	DL* HGI*	Not Supported		LSR EU PS*	DL* HGI*
H	Not Supported		Not Supported		Not Supported		Not Supported		LSR EU	RS* DL	Not Supported		Not Supported	
J	LSR EU DL*		LSR EU DL		LSR EU DL*		DSR DL*		LSR EU DL		Not Supported		LSR EU DL	
K	Not Supported		Not Supported		Not Supported		Not Supported		LSR RPL		Not Supported		Not Supported	
M	LSR EU PS*	DL* HGI*	LS* R EU LS*	PS* DL* HGI*	LSR EU PS	DL* HGI*	LSR EU*	PS* DL*	LSR EU RS*	RPL* DL* HGI*	Not Supported		LSR EU LS*	PS* DL* HGI*
N									Simple_port				Simple_port	

♪ NOTE: Forms marked by an asterisk* are optional or conditional for that type of service request. To determine form requirements, please refer to the ILEC business rules. For example, in directory service requests, the DSCR form is generally optional. The second DL form is also generally optional — it is used when the directory listing is being changed, in which case the first DL form indicates the original listing information and the second DL form indicates the 'change to' information.

♪ NOTE: For Verizon West, the first DL form is also optional in a directory service request.

6.6 LSR OBF Guidelines

The Ordering and Billing Forum (OBF) is a forum of the Local Services Ordering & Provisioning (LSOP) committee. Both are sponsored by the Alliance for Telecommunications Industry Standards (ATIS). The OBF provides a forum for customers and service providers in the telecommunications industry to identify, discuss and resolve national issues, which affect ordering, billing, provisioning and exchange of information about access services and other connectivity and related matters.

6.6.1 Downloading the OBF Guidelines

Employees of OBF funding companies may download documents from the Password Protected Web Site http://www.atis.org/obf/pcc.asp at no charge. Please contact your company's Primary Company Contact (PCC) http://www.atis.org/reginfo.asp for the user name and password.

6.6.2 Purchasing the OBF Guidelines

Employees of other companies may purchase OBF Forum Documents from ATIS. The OBF Document Order Form Web Page is:http://www.atis.org/obf/pcc.asp. Subcommittees of the OBF maintain the manual and mechanized ordering documents and guidelines for ordering local and access services.

6.6.3 Clearinghouse DTD Structures and OBF Guidelines

The Clearinghouse LSR Order DTDs are structured according to the Local Service Ordering Forms defined by the OBF. All fields that appear in Version 1, 2, 3, or 4 of the Guidelines are accommodated in the DTDs. This allows you to use the Clearinghouse LSR Order gateways to communicate with a trading partner regardless of which version of the Guidelines the trading partner is using.

If a field is deleted from the OBF Guidelines from one version to the next, it still appears in the DTDs. If a form name changes from one version of the LSOG to the next, we use the latest name in our DTDs. We also conform to LSOG 4 regarding form usage. For example, the Local Response (LR) form was called the Local Service Request Confirmation (LSC) form in LSOG 3, and this form was used to return all types of responses including FOC, SOC, and Jeopardy. In LSOG 4, the LSC became the Local Response (LR) form and is now only used to return FOC and Jeopardy notifications. A new form, called the Local Service Response/Completion (LSRCM) form, was added in LSOG 4; this form is used to return SOC responses. The DTDs use the LR form to return FOC and Jeopardy notifications, and the LSRCM form to return SOC and billing completion notifications.

Appendices

Appendix A: XML Examples

The OMS Clearinghouse extranet has an extensive set of XML file examples. To view and download these files, visit https://www.neustar.biz/convergentCH/content/ch_dl_xml.html

Query Orders

The following example shows the Request format for a query-orders request. This query is requesting the first ten rows of data that have a Supplier value of “BS”:

```
<?xml version="1.0" encoding="UTF-8"?>
<header>
  <Request value="query-messages"/>
  <Subrequest value="query-orders"/>
  <CustomerIdentifier value="PAT_TPE"/>
  <Supplier value="NEUSTAR" />
  <UserIdentifier value="tester"/>
  <UserPassword value="tester"/>
  <ApplyBusinessRules value="Y"/>
</header>

<?xml version="1.0" ?>
<Body>
  <Info>
    <ServiceType value="NP" />
    <REQTYP value="CB" />
    <Supplier value="BS" />
    <SortBy value="PON"/>
    <SortByOrdering value="DESC"/>
    <StartOffset value="0" />
    <EndOffset value="9" />
  </Info>
</Body>
```

The corresponding result to the above request is shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<header>
  <Request value="query-messages"/>
  <Subrequest value="query-orders"/>
  <CustomerIdentifier value="PAT_TPE"/>
  <Supplier value="NEUSTAR" />
  <UserIdentifier value="tester"/>
  <UserPassword value="tester"/>
  <ApplyBusinessRules value="Y"/>
</header>

<?xml version="1.0"?>

<Body>
 <DataContainer>
  <Data>
   <MessageKey value="48" />
   <ServiceType value="NP" />
   <REQTYP value="CB" />
   <Supplier value="BS" />
   <PON value="TYFCLONE006" />
   <Datetime value="06-02-2005-0318PM" />
   <ATN value="904-384-7450" />
   <ACT value="V" />
```

```
 <DUEDATE value="06-30-2006-1200AM" />
 <USERID value="tfreeberg" />
 <MessageType value="new" />
 <EUNAME value="TEST BS NP" />
 <InterfaceVersion value="LSOG6" />
 <Locked value="false" />
 <Action value="save" />
</Data>
<Data>
 <MessageKey value="31" />
 <ServiceType value="NP" />
 <REQTYP value="CB" />
 <Supplier value="BS" />
 <PON value="TYFCLONE005" />
 <Datetime value="06-02-2005-1029AM" />
 <ATN value="904-384-7430" />
 <ACT value="V" />
 <DUEDATE value="06-30-2006-1200AM" />
 <USERID value="tfreeberg" />
 <MessageType value="info" />
 <EUNAME value="TEST BS NP" />
 <InterfaceVersion value="LSOG6" />
 <Locked value="false" />
 <Action value="resume" />
</Data>
<Data>
 <MessageKey value="128" />
 <ServiceType value="NP" />
 <REQTYP value="CB" />
 <Supplier value="BS" />
 <PON value="TYFCLONE003" />
 <Datetime value="06-08-2005-0342PM" />
 <ACT value="V" />
 <USERID value="tfreeberg" />
 <MessageType value="info" />
 <InterfaceVersion value="LSOG6" />
 <Locked value="false" />
 <Action value="resume" />
</Data>
<Data>
 <MessageKey value="25" />
 <ServiceType value="NP" />
 <REQTYP value="CB" />
 <Supplier value="BS" />
 <PON value="TYFCLONE002" />
 <Datetime value="06-02-2005-0950AM" />
 <ATN value="904-384-7448" />
 <ACT value="V" />
 <DUEDATE value="06-30-2006-1200AM" />
 <USERID value="tfreeberg" />
 <MessageType value="new" />
 <EUNAME value="TEST BS NP" />
 <InterfaceVersion value="LSOG6" />
 <Locked value="false" />
 <Action value="submit" />
</Data>
<Data>
 <MessageKey value="21" />
 <ServiceType value="NP" />
 <REQTYP value="CB" />
 <Supplier value="BS" />
 <PON value="TYFCLONE001" />
 <Datetime value="06-02-2005-0928AM" />
 <ATN value="904-384-7449" />
 <ACT value="V" />
 <DUEDATE value="06-30-2006-1200AM" />
 <USERID value="docteam" />
 <MessageType value="new" />
 <EUNAME value="TEST BS NP" />
 <InterfaceVersion value="LSOG6" />
```

```
 <Locked value="false" />
 <Action value="save" />
</Data>
<Data>
 <MessageKey value="44" />
 <ServiceType value="NP" />
 <REQTYP value="CB" />
 <Supplier value="BS" />
 <PON value="TST1236753" />
 <Datetime value="06-02-2005-0235PM" />
 <ATN value="904-384-7449" />
 <ACT value="V" />
 <DUEDATE value="06-30-2006-1200AM" />
 <USERID value="tfreeberg" />
 <MessageType value="new" />
 <EUNAME value="TEST BS NP" />
 <InterfaceVersion value="LSOG6" />
 <Locked value="false" />
 <Action value="submit" />
</Data>
<Data>
 <MessageKey value="101" />
 <ServiceType value="NP" />
 <REQTYP value="CB" />
 <Supplier value="BS" />
 <PON value="PON1" />
 <Datetime value="06-08-2005-0950AM" />
 <ATN value="303-255-8888" />
 <ACT value="V" />
 <DUEDATE value="06-14-2005-1200AM" />
 <USERID value="tfreeberg" />
 <MessageType value="new" />
 <EUNAME value="Belle" />
 <InterfaceVersion value="LSOG6" />
 <Locked value="false" />
 <Action value="save" />
</Data>
<Data>
 <MessageKey value="53" />
 <ServiceType value="NP" />
 <REQTYP value="CB" />
 <Supplier value="BS" />
 <PON value="Becker" />
 <Datetime value="06-02-2005-0428PM" />
 <ATN value="904-384-7449" />
 <ACT value="V" />
 <DUEDATE value="06-30-2006-1200AM" />
 <USERID value="docteam" />
 <MessageType value="new" />
 <EUNAME value="TEST BS NP" />
 <InterfaceVersion value="LSOG6" />
 <Locked value="false" />
 <Action value="save" />
</Data>
<Data>
 <MessageKey value="67" />
 <ServiceType value="NP" />
 <REQTYP value="CB" />
 <Supplier value="BS" />
 <PON value="BSTESTEM002" />
 <Datetime value="06-06-2005-0923AM" />
 <ATN value="904-384-7449" />
 <ACT value="V" />
 <DUEDATE value="06-30-2006-1200AM" />
 <USERID value="tfreeberg" />
 <MessageType value="new" />
 <EUNAME value="TEST BS NP" />
 <InterfaceVersion value="LSOG6" />
 <Locked value="false" />
 <Action value="submit" />
```

```
  </Data>
  <Data>
   <MessageKey value="52" />
   <ServiceType value="NP" />
   <REQTYP value="CB" />
   <Supplier value="BS" />
   <PON value="BSNPLSOG6TEMPLAT" />
   <Datetime value="06-02-2005-0338PM" />
   <ATN value="904-384-7449" />
   <ACT value="V" />
   <DUEDATE value="06-30-2006-1200AM" />
   <USERID value="jmeyers" />
   <MessageType value="info" />
   <EUNAME value="TEST BS NP" />
   <InterfaceVersion value="LSOG6" />
   <Locked value="false" />
   <Action value="resume" />
  </Data>
 </DataContainer>
 <Info>
  <StartOffset value="0" />
  <EndOffset value="9" />
  <TotalRowCount value="13" />
  <Count value="10" />
 </Info>
</Body>
```

Query Order History

The following example shows the format for a query-order-history request. This query is requesting the first ten rows of data that have a PON value starting with CH2CA002:

```
<?xml version="1.0" encoding="UTF-8" ?>
<header>
 <Request value="query-messages"/>
 <Subrequest value="query-order-history"/>
 <Supplier value="NEUSTAR"/>
 <CustomerIdentifier value="ACM_E2E"/>
 <UserIdentifier value="ACME"/>
 <UserPassword value="password"/>
</header>

<?xml version="1.0"?>
<Body>
 <Info>
  <ServiceType value="Loop"/>
  <PON value="CH2CA002%"/>
  <Supplier value="BS"/>
  <StartOffset value="0"/>
  <EndOffset value="9"/>
  <MessageType value="%"/>
 </Info>
</Body>
```

The corresponding result to the above request is shown below.

```
<?xml version="1.0" encoding="UTF-8" ?>
<Body>
 <DataContainer>
  <Data>
   <MessageKey value="108" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="09-01-2004-0745PM" />
```

```
 <MessageType value="suppaccept" />
 <VER value="02" />
</Data>
<Data>
 <MessageKey value="103" />
 <ServiceType value="Loop" />
 <Supplier value="BS" />
 <PON value="CH2CA002TEST1AAA" />
 <Datetime value="09-01-2004-0717PM" />
 <MessageType value="ack" />
 <VER value="02" />
</Data>
<Data>
 <MessageKey value="102" />
 <ServiceType value="Loop" />
 <Supplier value="BS" />
 <PON value="CH2CA002TEST1AAA" />
 <Datetime value="09-01-2004-0716PM" />
 <MessageType value="supplement" />
 <VER value="02" />
</Data>
<Data>
 <MessageKey value="86" />
 <ServiceType value="Loop" />
 <Supplier value="BS" />
 <PON value="CH2CA002TEST1AAA" />
 <Datetime value="09-01-2004-0412PM" />
 <MessageType value="suppreject" />
 <VER value="01" />
</Data>
<Data>
 <MessageKey value="85" />
 <ServiceType value="Loop" />
 <Supplier value="BS" />
 <PON value="CH2CA002TEST1AAA" />
 <Datetime value="09-01-2004-0411PM" />
 <MessageType value="ack" />
 <VER value="01" />
</Data>
<Data>
 <MessageKey value="84" />
 <ServiceType value="Loop" />
 <Supplier value="BS" />
 <PON value="CH2CA002TEST1AAA" />
 <Datetime value="09-01-2004-0411PM" />
 <MessageType value="supplement" />
 <VER value="01" />
</Data>
<Data>
 <MessageKey value="83" />
 <ServiceType value="Loop" />
 <Supplier value="BS" />
 <PON value="CH2CA002TEST1AAA" />
 <Datetime value="09-01-2004-0314PM" />
 <MessageType value="focaccept" />
</Data>
<Data>
 <MessageKey value="82" />
 <ServiceType value="Loop" />
 <Supplier value="BS" />
 <PON value="CH2CA002TEST1AAA" />
 <Datetime value="09-01-2004-0302PM" />
 <MessageType value="ack" />
</Data>
<Data>
 <MessageKey value="81" />
 <ServiceType value="Loop" />
 <Supplier value="BS" />
 <PON value="CH2CA002TEST1AAA" />
 <Datetime value="09-01-2004-0301PM" />
```

```
   <MessageType value="new" />
  </Data>
  <Data>
   <MessageKey value="65" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="08-31-2004-0445PM" />
   <MessageType value="new" />
  </Data>
 </DataContainer>
 <Info>
  <Count value="10" />
 </Info>
</Body>
```

Query Transaction Details

The following example shows the format for a query-transaction-details request. This query is requesting the transaction details for orders that have a PON value starting with CH2CA002:

```
<?xml version="1.0" encoding="UTF-8" ?>
<header>
 <Request value="query-messages"/>
 <Subrequest value="query-transaction-details"/>
 <Supplier value="NEUSTAR"/>
 <CustomerIdentifier value="ACM_E2E"/>
 <UserIdentifier value="ACME"/>
 <UserPassword value="password"/>
</header>

<?xml version="1.0"?>
<Body>
 <Info>
  <ServiceType value="Loop"/>
  <PON value="CH2CA002%"/>
 </Info>
</Body>
```

The corresponding result to the above request is shown below.

```
<?xml version="1.0"?>
<Body>
 <DataContainer>
  <Data>
   <MessageKey value="83" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="09-01-2004-0314PM" />
   <MessageType value="focaccept" />
   <Message>
    <Response>
     <lsr_order_response>
      <ResponseType value="focaccept" />
      <focaccept>
       <lr>
        <lr_adminsection>
         <CCNA value="ZXL" />
         <PON value="CH2CA002TEST1AAA" />
         <AN value="318M333028" />
         <LSRNO value="999920040901000288" />
         <ORD value="N544NKC7" />
         <INIT value="ADRIENNE GOODEN" />
         <DTSENT value="09-01-2004" />
         <REP value="LCSC" />
         <REP TELNO value="800-667-0807" />
         <FDT value="0700PM" />
```

```
        <DD value="09-10-2004" />
        <BAN1 value="318Q887771771" />
       </lr_adminsection>
       <lr circuitdetailcontainer type="container">
        <lr_circuitdetail>
         <LOCNUM value="000" />
         <LNUM value="00001" />
         <ECCKT value="60.TYNU.503599..SC" />
         <tiedowncontainer type="container">
          <tiedown>
           <cablecontainer type="container">
            <cable>
             <CABLEID value="PAWS2" />
             <CHANPAIR value="0022" />
            </cable>
           </cablecontainer>
          </tiedown>
         </tiedowncontainer>
         <ORDLcontainer type="container">
          <ORDL value="N544NKC7" />
         </ORDLcontainer>
        </lr circuitdetail>
        <lr_circuitdetail>
         <LOCNUM value="000" />
         <LNUM value="00002" />
         <ECCKT value="60.TYNU.503600..SC" />
         <tiedowncontainer type="container">
          <tiedown>
           <cablecontainer type="container">
            <cable>
             <CABLEID value="PAWS2" />
             <CHANPAIR value="0023" />
            </cable>
           </cablecontainer>
          </tiedown>
         </tiedowncontainer>
         <ORDLcontainer type="container">
          <ORDL value="N544NKC7" />
         </ORDLcontainer>
        </lr_circuitdetail>
        <lr_circuitdetail>
         <LOCNUM value="000" />
         <LNUM value="00003" />
         <ECCKT value="60.TYNU.503601..SC" />
         <tiedowncontainer type="container">
          <tiedown>
           <cablecontainer type="container">
            <cable>
             <CABLEID value="PAWS2" />
             <CHANPAIR value="0024" />
            </cable>
           </cablecontainer>
          </tiedown>
         </tiedowncontainer>
         <ORDLcontainer type="container">
          <ORDL value="N544NKC7" />
         </ORDLcontainer>
        </lr_circuitdetail>
       </lr_circuitdetailcontainer>
       <REMARKS value=" LEOTEST SJF2K DISPATCH IS REQUIRED" />
      </lr>
     </focaccept>
    </lsr_order_response>
    <SupplierLSROrderResponse>
     <focaccept>
      <lr>
       <lr_adminsection>
        <CC value="9999" />
        <STATUSCODE value="AO" />
        <STATUSMSG value="ASSIGNABLE ORDER" />
```

```
    <INIT_TELNO value="425-869-5949" />
   </lr_adminsection>
  </lr>
 </focaccept>
</SupplierLSROrderResponse>
<lsr_order_response>
 <ResponseType value="ack" />
 <lsr ack>
  <PON value="CH2CA002TEST1AAA" />
  <TRANSACT value="0101" />
  <GROUP value="101" />
  <STATUS value="ACCEPTED" />
 </lsr_ack>
</lsr order response>
<SupplierLSROrderResponse />
<lsr order response>
 <ResponseType value="ack" />
 <lsr_ack>
  <PON value="CH2CA002TEST1AAA" />
  <TRANSACT value="0102" />
  <GROUP value="102" />
  <STATUS value="ACCEPTED" />
  <VER value="01" />
 </lsr_ack>
</lsr order response>
<SupplierLSROrderResponse />
<lsr order response>
 <ResponseType value="suppreject" />
 <suppreject>
  <REJECTTYPE value="SUPERFATAL" />
  <lr>
   <lr_adminsection>
    <CCNA value="ZXL" />
    <PON value="CH2CA002TEST1AAA" />
    <VER value="01" />
    <AN value="318M333028" />
    <LSRNO value="999920040901000341" />
    <INIT value="ADRIENNE GOODEN" />
    <DTSENT value="09-01-2004" />
    <REP value="LCSC" />
    <REP_TELNO value="800-667-0807" />
    <BAN1 value="318Q887771771" />
    <reasoncontainer type="container">
     <reason>
      <ERRORCODE value="R1131" />
      <ERRORTEXT value="DDD IS LESS THAN CALC DATE ON PRIOR VERSION LSR
                                                    OR SERVICE ORDER DUE DATE" />
     </reason>
    </reasoncontainer>
   </lr adminsection>
   <lr_circuitdetailcontainer type="container">
    <lr circuitdetail>
     <LOCNUM value="000" />
     <LNUM value="00001" />
     <tiedowncontainer type="container">
      <tiedown>
       <cablecontainer type="container">
        <cable>
         <CABLEID value="PAWS2" />
         <CHANPAIR value="0022" />
        </cable>
       </cablecontainer>
      </tiedown>
     </tiedowncontainer>
    </lr_circuitdetail>
    <lr_circuitdetail>
     <LOCNUM value="000" />
     <LNUM value="00002" />
     <tiedowncontainer type="container">
      <tiedown>
```

```
          <cablecontainer type="container">
           <cable>
            <CABLEID value="PAWS2" />
            <CHANPAIR value="0023" />
           </cable>
          </cablecontainer>
         </tiedown>
        </tiedowncontainer>
       </lr_circuitdetail>
       <lr circuitdetail>
        <LOCNUM value="000" />
        <LNUM value="00003" />
        <tiedowncontainer type="container">
         <tiedown>
          <cablecontainer type="container">
           <cable>
            <CABLEID value="PAWS2" />
            <CHANPAIR value="0024" />
           </cable>
          </cablecontainer>
         </tiedown>
        </tiedowncontainer>
       </lr_circuitdetail>
     </lr_circuitdetailcontainer>
     <REMARKS value="SUP TO CHANGE DUE DATE FROM SEPT 10 TO SEPT 3" />
    </lr>
   </suppreject>
</lsr_order_response>
<SupplierLSROrderResponse>
 <suppreject>
  <lr>
   <lr_adminsection>
    <CC value="9999" />
    <INIT_TELNO value="425-869-5949" />
   </lr_adminsection>
  </lr>
 </suppreject>
</SupplierLSROrderResponse>
<lsr order response>
 <ResponseType value="ack" />
 <lsr_ack>
  <PON value="CH2CA002TEST1AAA" />
  <TRANSACT value="0109" />
  <GROUP value="109" />
  <STATUS value="ACCEPTED" />
  <VER value="02" />
 </lsr_ack>
</lsr_order_response>
<SupplierLSROrderResponse />
<lsr order response>
 <ResponseType value="suppaccept" />
 <suppconfirmation>
  <lr>
   <lr_adminsection>
    <CCNA value="ZXL" />
    <PON value="CH2CA002TEST1AAA" />
    <VER value="02" />
    <AN value="318M333028" />
    <LSRNO value="999920040901000346" />
    <ORD value="N544NKC7" />
    <INIT value="ADRIENNE GOODEN" />
    <DTSENT value="09-01-2004" />
    <REP value="LCSC" />
    <REP TELNO value="800-667-0807" />
    <FDT value="0700PM" />
    <DD value="09-17-2004" />
    <BAN1 value="318Q887771771" />
   </lr adminsection>
   <lr_circuitdetailcontainer type="container">
    <lr_circuitdetail>
```

```
      <LOCNUM value="000" />
      <LNUM value="00001" />
      <ECCKT value="60.TYNU.503599..SC" />
      <tiedowncontainer type="container">
       <tiedown>
        <cablecontainer type="container">
         <cable>
          <CABLEID value="PAWS2" />
          <CHANPAIR value="0022" />
         </cable>
        </cablecontainer>
       </tiedown>
      </tiedowncontainer>
      <ORDLcontainer type="container">
       <ORDL value="N544NKC7" />
      </ORDLcontainer>
     </lr_circuitdetail>
     <lr_circuitdetail>
      <LOCNUM value="000" />
      <LNUM value="00002" />
      <ECCKT value="60.TYNU.503600..SC" />
      <tiedowncontainer type="container">
       <tiedown>
        <cablecontainer type="container">
         <cable>
          <CABLEID value="PAWS2" />
          <CHANPAIR value="0023" />
         </cable>
        </cablecontainer>
       </tiedown>
      </tiedowncontainer>
      <ORDLcontainer type="container">
       <ORDL value="N544NKC7" />
      </ORDLcontainer>
     </lr_circuitdetail>
     <lr_circuitdetail>
      <LOCNUM value="000" />
      <LNUM value="00003" />
      <ECCKT value="60.TYNU.503601..SC" />
      <tiedowncontainer type="container">
       <tiedown>
        <cablecontainer type="container">
         <cable>
          <CABLEID value="PAWS2" />
          <CHANPAIR value="0024" />
         </cable>
        </cablecontainer>
       </tiedown>
      </tiedowncontainer>
      <ORDLcontainer type="container">
       <ORDL value="N544NKC7" />
      </ORDLcontainer>
     </lr_circuitdetail>
    </lr_circuitdetailcontainer>
    <REMARKS value="SUP TO CHANGE DUE DATE FROM SEPT 10 TO SEPT 3 LEOTEST
                                        SJF2K" />
   </lr>
  </suppconfirmation>
 </lsr_order_response>
 <SupplierLSROrderResponse>
  <suppconfirmation>
   <lr>
    <lr_adminsection>
     <CC value="9999" />
     <STATUSCODE value="AO" />
     <STATUSMSG value="ASSIGNABLE ORDER" />
     <INIT_TELNO value="425-869-5949" />
    </lr_adminsection>
   </lr>
  </suppconfirmation>
```

```
 </SupplierLSROrderResponse>
</Response>
<Request>
 <lsr order>
  <eu>
   <locationaccesscontainer type="container">
    <locationaccess>
     <SATH value="ST" />
     <CITY value="BUNKER" />
     <SANO value="300" />
     <SASN value="WALNUT" />
     <ZIP value="71322" />
     <NAME value="BUGS BUNNY" />
     <STATE value="LA" />
    </locationaccess>
   </locationaccesscontainer>
  </eu>
  <ls>
   <servicedetailscontainer type="container">
    <servicedetails>
     <cablecontainer type="container">
      <cable>
       <CHANPAIR value="0022" />
       <CABLEID value="PAWS2" />
      </cable>
     </cablecontainer>
     <LNUM value="00001" />
     <LNA value="N" />
    </servicedetails>
    <servicedetails>
     <LNUM value="00002" />
     <cablecontainer type="container">
      <cable>
       <CABLEID value="PAWS2" />
       <CHANPAIR value="0023" />
      </cable>
     </cablecontainer>
     <LNA value="N" />
    </servicedetails>
    <servicedetails>
     <cablecontainer type="container">
      <cable>
       <CABLEID value="PAWS2" />
       <CHANPAIR value="0024" />
      </cable>
     </cablecontainer>
     <LNUM value="00003" />
     <LNA value="N" />
    </servicedetails>
   </servicedetailscontainer>
   <ls_adminsection>
    <LQTY value="00003" />
   </ls_adminsection>
  </ls>
  <lsr>
   <lsr adminsection>
    <DTSENT value="08-31-2004-0437PM" />
    <REQTYP value="AB" />
    <DDD value="09-10-2004" />
    <PON value="CH2CA002TEST1AAA" />
    <PROJECT value="CAVENOBILL" />
    <ACT value="N" />
    <CC value="9999" />
    <AN value="318M333028" />
    <CCNA value="ZXL" />
    <SC value="LCSC" />
    <LSO value="318346" />
    <NC value="TY--" />
    <ACTL value="BUNKLAMARS1" />
    <TOS value="1A--" />
```

```
  </lsr_adminsection>
  <billsection>
   <ACNA value="ZXL" />
   <BAN1 value="318Q887771771" />
  </billsection>
  <contactsection>
   <INIT value="ADRIENNE GOODEN" />
   <FAXNO value="555-555-5555" />
   <TELNO value="425-869-5949" />
   <IMPCON>
    <NAME value="ADRIENNE GOODEN" />
    <TELNO value="425-869-5949" />
   </IMPCON>
  </contactsection>
 </lsr>
</lsr order>
<SupplierLSROrderRequest>
 <ls>
  <servicedetailscontainer type="container">
   <servicedetails>
    <LNUM value="00001" />
   </servicedetails>
   <servicedetails>
    <LNUM value="00002" />
   </servicedetails>
   <servicedetails>
    <LNUM value="00003" />
   </servicedetails>
  </servicedetailscontainer>
 </ls>
</SupplierLSROrderRequest>
<lsr_order>
 <eu>
  <locationaccesscontainer type="container">
   <locationaccess>
    <SATH value="ST" />
    <CITY value="BUNKER" />
    <SANO value="300" />
    <SASN value="WALNUT" />
    <ZIP value="71322" />
    <NAME value="BUGS BUNNY" />
    <STATE value="LA" />
   </locationaccess>
  </locationaccesscontainer>
 </eu>
 <ls>
  <servicedetailscontainer type="container">
   <servicedetails>
    <cablecontainer type="container">
     <cable>
      <CHANPAIR value="0022" />
      <CABLEID value="PAWS2" />
     </cable>
    </cablecontainer>
    <LNUM value="00001" />
    <LNA value="N" />
   </servicedetails>
   <servicedetails>
    <LNUM value="00002" />
    <cablecontainer type="container">
     <cable>
      <CABLEID value="PAWS2" />
      <CHANPAIR value="0023" />
     </cable>
    </cablecontainer>
    <LNA value="N" />
   </servicedetails>
   <servicedetails>
    <cablecontainer type="container">
     <cable>
```

```
        <CABLEID value="PAWS2" />
        <CHANPAIR value="0024" />
       </cable>
      </cablecontainer>
      <LNUM value="00003" />
      <LNA value="N" />
     </servicedetails>
    </servicedetailscontainer>
    <ls_adminsection>
     <LQTY value="00003" />
    </ls_adminsection>
   </ls>
   <lsr>
    <lsr adminsection>
     <DTSENT value="09-01-2004-0301PM" />
     <REQTYP value="AB" />
     <DDD value="09-10-2004" />
     <PON value="CH2CA002TEST1AAA" />
     <PROJECT value="CAVENOBILL" />
     <ACT value="N" />
     <CC value="9999" />
     <AN value="318M333028" />
     <CCNA value="ZXL" />
     <SC value="LCSC" />
     <LSO value="318346" />
     <NC value="TY--" />
     <ACTL value="BUNKLAMARS1" />
     <TOS value="1A--" />
    </lsr_adminsection>
    <billsection>
     <ACNA value="ZXL" />
     <BAN1 value="318Q887771771" />
    </billsection>
    <contactsection>
     <INIT value="ADRIENNE GOODEN" />
     <FAXNO value="555-555-5555" />
     <TELNO value="425-869-5949" />
     <IMPCON>
      <NAME value="ADRIENNE GOODEN" />
      <TELNO value="425-869-5949" />
     </IMPCON>
    </contactsection>
   </lsr>
  </lsr_order>
  <SupplierLSROrderRequest>
   <ls>
    <servicedetailscontainer type="container">
     <servicedetails>
      <LNUM value="00001" />
     </servicedetails>
     <servicedetails>
      <LNUM value="00002" />
     </servicedetails>
     <servicedetails>
      <LNUM value="00003" />
     </servicedetails>
    </servicedetailscontainer>
   </ls>
  </SupplierLSROrderRequest>
  <lsr_order>
   <lsr>
    <REMARKS value="sup to change due date from sept 10 to sept 3" />
    <lsr_adminsection>
     <DTSENT value="09-01-2004-0301PM" />
     <REQTYP value="AB" />
     <DDD value="09-03-2004" />
     <PON value="CH2CA002TEST1AAA" />
     <PROJECT value="CAVENOBILL" />
     <ACT value="N" />
     <CC value="9999" />
```

```
    <AN value="318M333028" />
    <CCNA value="ZXL" />
    <SC value="LCSC" />
    <LSO value="318346" />
    <NC value="TY--" />
    <ACTL value="BUNKLAMARS1" />
    <TOS value="1A--" />
    <SUP value="2" />
    <VER value="01" />
   </lsr adminsection>
   <billsection>
    <ACNA value="ZXL" />
    <BAN1 value="318Q887771771" />
   </billsection>
   <contactsection>
    <INIT value="ADRIENNE GOODEN" />
    <FAXNO value="555-555-5555" />
    <TELNO value="425-869-5949" />
    <IMPCON>
     <NAME value="ADRIENNE GOODEN" />
     <TELNO value="425-869-5949" />
    </IMPCON>
   </contactsection>
  </lsr>
  <eu>
   <locationaccesscontainer type="container">
    <locationaccess>
     <SATH value="ST" />
     <CITY value="BUNKER" />
     <SANO value="300" />
     <SASN value="WALNUT" />
     <ZIP value="71322" />
     <NAME value="BUGS BUNNY" />
     <STATE value="LA" />
    </locationaccess>
   </locationaccesscontainer>
  </eu>
  <ls>
   <servicedetailscontainer type="container">
    <servicedetails>
     <cablecontainer type="container">
      <cable>
       <CHANPAIR value="0022" />
       <CABLEID value="PAWS2" />
      </cable>
     </cablecontainer>
     <LNUM value="00001" />
     <LNA value="N" />
    </servicedetails>
    <servicedetails>
     <LNUM value="00002" />
     <cablecontainer type="container">
      <cable>
       <CABLEID value="PAWS2" />
       <CHANPAIR value="0023" />
      </cable>
     </cablecontainer>
     <LNA value="N" />
    </servicedetails>
    <servicedetails>
     <cablecontainer type="container">
      <cable>
       <CABLEID value="PAWS2" />
       <CHANPAIR value="0024" />
      </cable>
     </cablecontainer>
     <LNUM value="00003" />
     <LNA value="N" />
    </servicedetails>
   </servicedetailscontainer>
```

```
    <ls_adminsection>
     <LQTY value="00003" />
    </ls_adminsection>
   </ls>
  </lsr_order>
  <SupplierLSROrderRequest>
   <ls>
    <servicedetailscontainer type="container">
     <servicedetails>
      <LNUM value="00001" />
     </servicedetails>
     <servicedetails>
      <LNUM value="00002" />
     </servicedetails>
     <servicedetails>
      <LNUM value="00003" />
     </servicedetails>
    </servicedetailscontainer>
   </ls>
  </SupplierLSROrderRequest>
  <lsr_order>
   <lsr>
    <REMARKS value="sup to change due date from sept 10 to sept 3" />
    <lsr_adminsection>
     <DTSENT value="09-01-2004-0301PM" />
     <REQTYP value="AB" />
     <DDD value="09-17-2004" />
     <PON value="CH2CA002TEST1AAA" />
     <PROJECT value="CAVENOBILL" />
     <ACT value="N" />
     <CC value="9999" />
     <AN value="318M333028" />
     <CCNA value="ZXL" />
     <SC value="LCSC" />
     <LSO value="318346" />
     <NC value="TY--" />
     <ACTL value="BUNKLAMARS1" />
     <TOS value="1A--" />
     <SUP value="2" />
     <VER value="02" />
    </lsr_adminsection>
    <billsection>
     <ACNA value="ZXL" />
     <BAN1 value="318Q887771771" />
    </billsection>
    <contactsection>
     <INIT value="ADRIENNE GOODEN" />
     <FAXNO value="555-555-5555" />
     <TELNO value="425-869-5949" />
     <IMPCON>
      <NAME value="ADRIENNE GOODEN" />
      <TELNO value="425-869-5949" />
     </IMPCON>
    </contactsection>
   </lsr>
   <eu>
    <locationaccesscontainer type="container">
     <locationaccess>
      <SATH value="ST" />
      <CITY value="BUNKER" />
      <SANO value="300" />
      <SASN value="WALNUT" />
      <ZIP value="71322" />
      <NAME value="BUGS BUNNY" />
      <STATE value="LA" />
     </locationaccess>
    </locationaccesscontainer>
   </eu>
   <ls>
    <servicedetailscontainer type="container">
```

```
        <servicedetails>
         <cablecontainer type="container">
          <cable>
           <CHANPAIR value="0022" />
           <CABLEID value="PAWS2" />
          </cable>
         </cablecontainer>
         <LNUM value="00001" />
         <LNA value="N" />
        </servicedetails>
        <servicedetails>
         <LNUM value="00002" />
         <cablecontainer type="container">
          <cable>
           <CABLEID value="PAWS2" />
           <CHANPAIR value="0023" />
          </cable>
         </cablecontainer>
         <LNA value="N" />
        </servicedetails>
        <servicedetails>
         <cablecontainer type="container">
          <cable>
           <CABLEID value="PAWS2" />
           <CHANPAIR value="0024" />
          </cable>
         </cablecontainer>
         <LNUM value="00003" />
         <LNA value="N" />
        </servicedetails>
       </servicedetailscontainer>
       <ls_adminsection>
        <LQTY value="00003" />
       </ls_adminsection>
      </ls>
     </lsr_order>
     <SupplierLSROrderRequest>
      <ls>
       <servicedetailscontainer type="container">
        <servicedetails>
         <LNUM value="00001" />
        </servicedetails>
        <servicedetails>
         <LNUM value="00002" />
        </servicedetails>
        <servicedetails>
         <LNUM value="00003" />
        </servicedetails>
       </servicedetailscontainer>
      </ls>
     </SupplierLSROrderRequest>
    </Request>
   </Message>
  </Data>
  <Data>
   <MessageKey value="65" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="08-31-2004-0445PM" />
   <MessageType value="new" />
  </Data>
  <Data>
   <MessageKey value="82" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="09-01-2004-0302PM" />
   <MessageType value="ack" />
  </Data>
```

```
  <Data>
   <MessageKey value="81" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="09-01-2004-0301PM" />
   <MessageType value="new" />
  </Data>
  <Data>
   <MessageKey value="84" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="09-01-2004-0411PM" />
   <MessageType value="supplement" />
   <VER value="01" />
  </Data>
  <Data>
   <MessageKey value="85" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="09-01-2004-0411PM" />
   <MessageType value="ack" />
   <VER value="01" />
  </Data>
  <Data>
   <MessageKey value="86" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="09-01-2004-0412PM" />
   <MessageType value="suppreject" />
   <VER value="01" />
  </Data>
  <Data>
   <MessageKey value="102" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="09-01-2004-0716PM" />
   <MessageType value="supplement" />
   <VER value="02" />
  </Data>
  <Data>
   <MessageKey value="103" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="09-01-2004-0717PM" />
   <MessageType value="ack" />
   <VER value="02" />
  </Data>
  <Data>
   <MessageKey value="108" />
   <ServiceType value="Loop" />
   <Supplier value="BS" />
   <PON value="CH2CA002TEST1AAA" />
   <Datetime value="09-01-2004-0745PM" />
   <MessageType value="suppaccept" />
   <VER value="02" />
  </Data>
 </DataContainer>
 <Info>
  <Count value="10" />
 </Info>
</Body>
```

Appendix B: Tables

TABLE 1: BASIC SERVICE - REQUEST TYPES 14
TABLE 2: SOAP FAULT CODES 16
TABLE 3: SPECIAL CHARACTERS AND CORRESPONDING ENTRIES 19
TABLE 4: CLEARINGHOUSE BASIC GUI NOTIFICATION XML STRUCTURE. 26
TABLE 5: REQUEST AND SUBREQUEST VALUES 28
TABLE 6: VALID VALUES FOR THE ACTION FIELD 29
TABLE 7: LSR ORDER/PREORDER HEADER XML ELEMENTS 30
TABLE 8: LSR PREORDER REQUEST MESSAGE XML STRUCTURE 31
TABLE 9: LSR PREORDER XML DTD FILES (GENERIC RESPONSE) 32
TABLE 10: LSR PREORDER XML DTD FILES (AV, CSR, LOOPQUAL, SERVICE ORDER INQUIRY) 32
TABLE 11: LSR PREORDER XML DTDS (TN, APPT SCHEDULING, FS) 33
TABLE 12: LSR PREORDER XML DTDS (CANCEL RESERVATION, CFA) 34
TABLE 13: LSR PREORDER RESPONSE MESSAGE XML STRUCTURE 35
TABLE 14: LSR PREORDER RESPONSE TYPE VALUES 35
TABLE 15: LSR ORDER REQUEST STRUCTURE 35
TABLE 16: LSR ORDER XML DTD FILES 36
TABLE 17: LSR ORDER RESPONSE MESSAGE XML STRUCTURE 36
TABLE 18: LSR ORDER RESPONSE TYPE VALUES 37
TABLE 19: ASR ORDER/PREORDER HEADER XML ELEMENTS 37
TABLE 20: ASR ORDER/PREORDER REQUEST STRUCTURE 38
TABLE 21: ASR ORDER/PREORDER RESPONSE STRUCTURE 38
TABLE 22: ASR ORDER/ PREORDER RESPONSE TYPE VALUES 39
TABLE 23: E911 REQUEST STRUCTURE 39
TABLE 24: E911 MAPPING FROM RESPONSETYPE TO RESPONSEBODY 41
TABLE 25: E911 RESPONSE TYPES AND CREATION FREQUENCY 41
TABLE 26: LIDB RESPONSE MESSAGE XML STRUCTURE 43
TABLE 27: LIDB RESPONSE TYPE VALUES 44
TABLE 28: ICP HEADER XML ELEMENTS 44
TABLE 29: MULTIPORTREQUEST PARAMETERS 45
TABLE 30: MULTIPORTRESPONSE PARAMETERS 52
TABLE 31: MODIFYPORTREQUEST PARAMETERS 60
TABLE 32: BROADCASTNOTIFICATION PARAMETERS 65
TABLE 33: TESTMESSAGEQUERY PARAMETERS 67
TABLE 34: TESTMESSAGERESPONSE PARAMETERS 68
TABLE 35: STORESTATUS PARAMETERS 70
TABLE 36: VALIDATIONSTATUS PARAMETERS 72
TABLE 37: QUERY REQUEST AND SUBREQUEST VALID VALUES 77
TABLE 38: QUERY REQUEST DATA ELEMENTS 78
TABLE 39: OR CONSTRUCT DATA ELEMENTS 79
TABLE 40: REQUEST XML ELEMENTS FOR QUERY-ORDERS 80
TABLE 41: REQUEST XML ELEMENTS FOR QUERY-TRANSACTION-DETAILS 87
TABLE 42: REQUEST XML ELEMENTS FOR QUERY-ORDER-HISTORY 88
TABLE 43: REQUEST XML ELEMENTS FOR QUERY-TRANSACTIONS 89

TABLE 44: REQUEST XML ELEMENTS FOR QUERY-ORDER-HISTORY-997 93
TABLE 45: QUERY MESSAGE RESPONSE FIELDS 93
TABLE 46: LSR - BASED FORMS 103

Appendix C: Figures

FIGURE 1: CLEARINGHOUSE PLATFORM 11
FIGURE 2: ACCESSING THE CLEARINGHOUSE 12
FIGURE 3: SOAP CLIENT VIEW 13
FIGURE 4: RECEIVING A RESPONSE VIA THE SOAPREQUESTHANDLER 15
FIGURE 5: PREORDERING MESSAGE FLOW FOR ADDRESS VALIDATION 98
FIGURE 6: PREORDERING MESSAGE FLOW 99
FIGURE 7: LSR GATEWAY ORDER FLOW 100
FIGURE 8: FORMS USED TO PLACE AN LSNP SERVICE REQUEST 102