# Table of Contents

- **Change Log** .................................................................................................................................................... 4
- **Introduction to DelVAX** ................................................................................................................................. 5
  - Purpose ..................................................................................................................................................... 5
  - Scope ......................................................................................................................................................... 5
  - Audience ................................................................................................................................................... 5
  - Requirements ............................................................................................................................................ 5
  - References ................................................................................................................................................ 5
    - CDC WSDL Implementation Resources ................................................................................................. 5
    - NIST Immunization Test Suite (NIST IZTS) ............................................................................................. 5
    - Examples ............................................................................................................................................... 5
- **HL7 Web Service Definition** .......................................................................................................................... 6
  - Service Overview ....................................................................................................................................... 6
  - Table of Inputs and Outputs ..................................................................................................................... 6
    - `connectivityTest()` ................................................................................................................................. 6
    - `submitSingleMessage()` ......................................................................................................................... 7
  - Faults and Errors ....................................................................................................................................... 7
    - Soap Faults ............................................................................................................................................ 7
    - SOAP Fault Parameters ......................................................................................................................... 7
    - Custom Error Codes .................................................................................................................................. 8
- **Consume the Service** .................................................................................................................................... 8
  - NIST Immunization Test Suite ................................................................................................................... 8
    - Validate SOAP formatting ..................................................................................................................... 8
    - Test SOAP connectivity ........................................................................................................................ 10
    - HL7 Testing and Validation .................................................................................................................. 13
  - Further Reading .......................................................................................................................................... 14
- **Troubleshooting** .......................................................................................................................................... 14
  - The HL7 carriage return has been replaced by a line feed. ..................................................................... 14
  - I’m having SOAP Header configuration issues. ....................................................................................... 14
  - Obtaining a Certificate ............................................................................................................................ 15
- **Appendix A: WSDL Basics** ............................................................................................................................ 16
  - WSDL Basics ............................................................................................................................................ 16
  - Related Links and References .................................................................................................................. 16
Appendix B: Simple Object Access Protocol ................................................................. 17
Definition ..................................................................................................................... 17
SOAP 1.2 ...................................................................................................................... 17
Envelope ....................................................................................................................... 17
Header ......................................................................................................................... 17
Body ............................................................................................................................ 18
SOAP Faults ................................................................................................................. 18
Related Links and References .................................................................................... 19
Appendix C: HL7 Overview and Code Samples ............................................................. 20
Overview ..................................................................................................................... 20
HL7 Version 2.x ........................................................................................................... 20
HL7 Message Example ............................................................................................... 20
HL7 Message Encoding .............................................................................................. 20
SOAP Request and Response with HL7 Payload ....................................................... 20
SOAP Request with example HL7 Message and optional facility ID parameter ....... 21
SOAP Request with example HL7 Message and Response (HL7 2.4) ...................... 21
Successful Response(ACK) (HL7 2.4) ................................................................... 22
Change Log

The following table captures the progression of this document over time.

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Description</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>02/10/2016</td>
<td>Document Creation</td>
<td>Sarah Voegeli</td>
</tr>
<tr>
<td>1.1</td>
<td>02/19/2016</td>
<td>Added highlights for those areas that need to the modified for each customer</td>
<td>Claire Murchie</td>
</tr>
<tr>
<td>1.2</td>
<td>02/26/2016</td>
<td>Final edits to the document</td>
<td>Claire Murchie</td>
</tr>
<tr>
<td>1.3</td>
<td>07/26/2017</td>
<td>Added NIST-centric instructions, removed references to C# and Java example clients, as they are now redundant</td>
<td>J. Zachry Mathias</td>
</tr>
</tbody>
</table>
Introduction to DelVAX

DelVAX is a Web-based, database-driven immunization registry system currently implemented at multiple state and local government agencies in the US and internationally. It is designed to meet the standard requirements for tracking and administering immunizations in a public health setting. DelVAX also provides customization options and extensibility to serve the needs of sophisticated agencies. Providers may interact with the DelVAX system using a SOAP-based web service or HTTP POST.

Purpose

The purpose of this document is to define the DelVAX HL7 Web Service and to demonstrate proxies that consume the service using the HL7 message format and the SOAP/XML protocol. An example client is provided by NIST for testing purposes, called the NIST Immunization Test Suite.

Scope

This document defines and details the HL7 Service for versions 15.10+ of the DelVAX system. It provides samples and instructions for consuming the with the NIST Immunization Test Suite (NIST IZTS) client and describes additional contextual information around SOAP/XML and HL7.

Audience

This reference is intended for vendors who wish to integrate their systems with the DelVAX software.

Requirements

A modern web browser (Chrome, IE/Edge, Firefox) with current updates installed is required for interacting with the NIST IZTS.

References

CDC WSDL Implementation Resources

Information on the CDC WSDL is available at https://www.cdc.gov/vaccines/programs/iis/technical-guidance/soap/services.html. The WSDL defines the relationship between the sender and receiver for a Simple Object Access Protocol (SOAP) based web service and can be used to generate proxies that consume the service. For more on WSDLs, see Appendix A: WSDL Basics.

NIST Immunization Test Suite (NIST IZTS)

A client application that consumes a given HL7 service is available from NIST here: https://hl7v2-iz-r1.5-testing.nist.gov/iztool/#/home. You can use this client to test the HL7 messages sent to and received from the DelVAX service, and to test the validity and syntactic accuracy of SOAP messages.

Examples

Additional examples of service clients are available through the following resources.

- Creating and Accessing XML Web Service with C#/C++
HL7 Web Service Definition

Service Overview

The purpose of the HL7 Service is to send an HL7 message to the DelVAX system. The service contains two methods: submitSingleMessage and connectivityTest. The submitSingleMessage method passes a username, password, optional facilityID, and HL7 message as a payload. The connectivityTest method is a secondary method that exists to confirm connectivity with the service. It submits a string parameter that is echoed back by the service when connectivity has been achieved.

Throughout this document, the service may be referred to as the HL7 Service, HL7 Web service, or HL7Service.

Client software may interface with the HL7 Service through a proxy that connects to endpoints on the DelVAX system.

![Interfacing with the HL7 Service](image)

**Figure 1 – Interfacing with the HL7 Service**

Table of Inputs and Outputs

<table>
<thead>
<tr>
<th>connectivityTest()</th>
<th>Parameter</th>
<th>Input/Output</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>echoBack</td>
<td>Input</td>
<td>String</td>
<td>String to be echoed back by the test</td>
</tr>
<tr>
<td></td>
<td>Return</td>
<td>Output</td>
<td>String</td>
<td>Data sent back by the test</td>
</tr>
</tbody>
</table>
submitSingleMessage()

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Input/Output</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>username</td>
<td>Input</td>
<td>String</td>
<td>IIS username</td>
</tr>
<tr>
<td>password</td>
<td>Input</td>
<td>String</td>
<td>IIS password</td>
</tr>
<tr>
<td>facilityID</td>
<td>Input (optional)</td>
<td>String</td>
<td>IIS Facility ID</td>
</tr>
<tr>
<td>hl7message</td>
<td>Input</td>
<td>String</td>
<td>HL7 version 2.3.1 or 2.5.1 message intended for IIS¹</td>
</tr>
<tr>
<td>Return</td>
<td>Output</td>
<td>String</td>
<td>HL7 version 2.3.1 or 2.5.1 response from IIS¹</td>
</tr>
</tbody>
</table>

Faults and Errors

SOAP faults are related to SOAP operations. They carry information about the error within a SOAP message.

There are four types of SOAP faults defined in the WSDL. All four faults are listed in the following table for convenience. However, the HL7 Service only implements the UnknownFault_Message.

Soap Faults

<table>
<thead>
<tr>
<th>SOAP fault</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UnsupportedOperationFault_Message</td>
<td>Occurs if the client requests an operation that is not part of the service</td>
</tr>
<tr>
<td>SecurityFault_Message</td>
<td>Occurs if the authentication credentials in the submitSingleMessage method are invalid</td>
</tr>
<tr>
<td>MessageTooLargeFault_Message</td>
<td>Occurs if the HL7 message is too large</td>
</tr>
<tr>
<td>UnknownFault_Message</td>
<td>Occurs when a fault does not fit into the previous three categories</td>
</tr>
</tbody>
</table>

SOAP Fault Parameters

All SOAP faults contain the following parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Input/Output</th>
<th>Data type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>Output</td>
<td>Integer</td>
<td>SOAP fault code number used by client software to identify the fault</td>
</tr>
</tbody>
</table>

¹ [http://www.cdc.gov/vaccines/programs/iis/technical-guidance/soap/wsd.html](http://www.cdc.gov/vaccines/programs/iis/technical-guidance/soap/wsd.html)
Custom Error Codes

Errors other than those listed in the Faults and Errors section above may occur in the HL7 message body (including headers and the HL7 content). See your local implementation guide for more information on message body (including headers and the HL7 content) errors.

Consume the Service

This section provides guides to consume the HL7 web service with the NIST IZTS using the test CDC endpoints publicly available on Envision’s test web server.

NIST Immunization Test Suite

In a web browser, open https://hl7v2-iz-r1.5-testing.nist.gov/iztool/#/home. Once the page loads, you’ll be presented with various tools for validating and testing HL7 messages.

Validate SOAP formatting

The tests in this area will validate any string against the SOAP 1.2 standard, and make sure that particular messages are formatted according to the CDC WSDL. For example, to test generic SOAP formatting, click the SOAPENV_1_MIN_Test test case:
Then, click the **Load Test Case** button in the upper-right corner of the screen:

![Test Case Load](image)

From there, you can either load an example, or paste your own SOAP message in:

![SOAP Envelope](image)

Finally, you can press the **Validate** button to test the formatting of the SOAP message.

You will receive a description of the error if any part of the message is malformed.
Test SOAP connectivity

The tests in this area will help validate incoming and outgoing soap messages, as well as test specific endpoints.

**Basic Connectivity Test**

To test for basic connectivity of a given endpoint, click on the `SOAPCON_1_BasicMessage_ConnectivityResponse` test case:

![Test Cases](image)

Then, click the *Load Test Case* button in the upper right-hand corner of the window,

![Load Test Case](image)

The basic echoBack message is preloaded for you in the outgoing SOAP envelope, and you cannot change it. Click on the *Configure* button to input the desired IIS endpoint. In this case, the Envision AART/Testing endpoint:

http://testing.envisiontechnology.com/HL7Engine_AART_Testing_20160425/CDC/V1/IISService.svc
Click **Save** to save the endpoint destination for this test. Click the **Send** button to send the connectivity test to the endpoint specified. If the endpoint specified was a valid, publicly available endpoint, a response will be sent back to the NIST IZTS and will be displayed in the Incoming SOAP Envelope window:

```
<?xml version="1.0" encoding="UTF-8"?>
<Envelope xmlns="http://www.w3.org/2003/05/soap-envelope">
    <Body>
        <connectivityTestResponse xmlns="urn:cdc:lisb:2011">
            <return>Hello world!</return>
        </connectivityTestResponse>
    </Body>
</Envelope>
```

Any errors in the SOAP response or the CDC specifications for an echoBack will be displayed in the errors section. Note that at the time of this writing, it appears that the NIST IZTS might have a small bug, as it appears to be looking for the ‘retun’ element, rather than the proper ‘return’ element.
Submit Single Message Test

To test a message submittal to a given endpoint, click on the SOAPCON_2_SubmitSingleMessage_Response test case:

The test message is preloaded for you and you cannot change it. Click the Configure button, and input a valid user, password, and facility, and endpoint URL:

Click Save to save the configuration for this test case, then click the Send button to send the test message to the specified endpoint. If the endpoint specified was a valid, publicly available endpoint, a response will be sent back to the NIST IZTS and will be displayed in the Incoming SOAP Envelope window:
In this example, an incorrect username/password combination was provided, and the NIST IZTS displays the response from the endpoint and from the validation result.

**HL7 Testing and Validation**

*Testing an Individual HL7 Message*

The NIST IZTS is also capable of testing discreet HL7 messages, either in free of context or context-based. This section will focus on context-free testing. For this section, you’ll need an HL7 message to test with. The message content window supports cut-and-paste. Additionally, if you have a message saved on your computer, you may click the **Browse** button, or you may also load a generic example message by clicking the **Load Example**

Click on the **HL7 Context-free** tab, select the profile group that your message belongs to, then paste into the **Message Content** window or select it from your filesystem by clicking the **Browse** button.
After the message has been loaded, click the Validate button, and the NIST IZTS will analyze the message and report any errors it finds:

Further Reading

For more information on how to use the NIST IZTS, or for further information on guidelines, etc. visit https://hl7v2-iz-r1.5-testing.nist.gov/iztool/#/doc.

Troubleshooting

The HL7 carriage return has been replaced by a line feed.

The HL7 standard requires that all lines end with a carriage return, whereas the XML standard used by SOAP requires that all end of line terminators be normalized to a line feed. Therefore, carriage returns in the HL7 message payload may be converted to line feeds during transmission. The HL7 service will accept either terminator, but the discrepancy may cause errors within the HL7 message body (including headers and the HL7 content) and response. For more information, please see your local implementation guide.

I’m having SOAP Header configuration issues.

SOAP Header information may differ between systems. Please verify the following parameters:
• Host: URL or Hostname here. You may need to include the port number.
• Content-Type: text/xml; charset=utf-8 || application/soap+xml; charset=utf-8
• Action/SOAPAction: URL of method here, e.g. http://tempuri.org/HL7Service

In SOAP 1.2, it is best practice to always pass in active headers and to define the SOAP action.

Obtaining a Certificate

Depending on system configuration, you may need to obtain a certificate. Check with your system administrator to learn if a certificate is required.
Appendix A: WSDL Basics

WSDL Basics

A Web Services Description Language (WSDL) document uses an XML structure to describe a Web service and its functionality. WSDL documents can be used to auto-generate proxies that consume a Web service.

WSDL documents define services as network endpoints (ports). A port is a single endpoint consisting of a binding and a network address.

Types are containers for data type definitions.

A port type defines an abstract set of operations supported by one or more endpoints.

Bindings consist of a protocol and data format specification for a particular port type.

A message is a typed definition of the data being communicated.

An operation is an abstract definition of an action supported by the service.²

Related Links and References

More information on WSDLs can be found through the following resources.

- WSDL Specification
- W3Schools: XML WSDL
- CDC Transport Specification

² [https://www.w3.org/TR/wsdl](https://www.w3.org/TR/wsdl)
Appendix B: Simple Object Access Protocol

Definition

The Simple Object Access Protocol (SOAP) is a lightweight information exchange protocol that uses XML as a message structure. SOAP can be used for communication between systems regardless of programming language, and it can be used over any transport protocol (e.g. TCP, HTTP, SMTP).³

This reference focuses on SOAP version 1.2, which differs from SOAP 1.1 in certain cases of syntax and semantics. For more on the differences between the two versions, visit https://www.w3.org/TR/soap12-part0/%23l4697.

In SOAP 1.2, the client is referred to as the sender, and the service is referred to as the receiver.

Figure 2 - SOAP sender and receiver

SOAP 1.2

Envelope

In SOAP 1.2, the envelope must be present with a local name of Envelope and the namespace http://www.w3.org/2003/05/soap-envelope. It may optionally include attribute information; however, env:encodingStyle can no longer appear on the env:Envelope element.

Header

The header element may be present with local name Header. If present, the header must include the namespace property and may have character, element, or attribute information as item children.

Although not required, it is best practice to include an active Header element and to define the SOAP action.

³ https://www.w3.org/TR/soap12-part0/%23l4697
The following code shows a Header attribute example with an element identifier of **Transaction** and a value of **5**.

```xml
<env:Header xmlns:env="http://www.w3.org/2003/05/soap-envelope" >
  <t:Transaction xmlns:t="http://example.org/2001/06/tx" env:mustUnderstand="true" >
    5
  </t:Transaction>
</env:Header>
```

**Body**

The **Body** element must be present in SOAP 1.2 with the local name **Body**. It may have character, element, or attribute information as item children. SOAP 1.2 does not permit any element after the body.

**SOAP Envelope**

- SOAP Header
- SOAP Body

*Figure 3 - SOAP envelope structure*

**SOAP Faults**

SOAP faults are returned as HTTP 500 errors and can occur in the following instances:

- The SOAP envelope is not valid, (e.g. parse error, missing elements)
  - This issue is the most common cause of HTTP 500 errors. Verify that the SOAP header and envelope are properly formatted. See above example.
- Out of memory conditions during parsing.
- SQL Server login failures.
- Unsupported SOAP operation, unknown operation or stored procedure specified.
- Invalid number of parameters for the specified operation.
Some HL7 message submission operations may require an HL7 Facility ID as a parameter, while other systems may not.

The following code sample demonstrates a SOAP 1.2 fault.

```xml
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope"
               xmlns:m="http://www.example.org/timeouts"
  <env:Body>
    <env:Fault>
      <env:Code>
        <env:Value>env:Sender</env:Value>
        <env:Subcode>
          <env:Value>m:MessageTimeout</env:Value>
        </env:Subcode>
      </env:Code>
      <env:Reason>
        <env:Text xml:lang="en">Sender Timeout</env:Text>
      </env:Reason>
      <env:Detail>
        <m:MaxTime>P5M</m:MaxTime>
      </env:Detail>
    </env:Fault>
  </env:Body>
</env:Envelope>
```

Related Links and References

More information on SOAP can be found through the following resources.

- [Understanding SOAP](#)
- [SOAP 1.1 Spec Doc](#)
- [SOAP 1.2 Spec Doc](#)
- [Namespaces in XML](#)
- [SOAP Fault Message Structure](#)
Appendix C: HL7 Overview and Code Samples

Overview

The Health Level Seven (HL7) standards provide for the exchange, integration, sharing, and retrieval of electronic health information between healthcare systems. These standards define how information is packaged and communicated from one party to another. They set the language, structure and data types required for integration between different systems on the application layer.\(^4\)

HL7 Version 2.x

HL7 version 2.x (Pipehat) use non-XML encoding with single character delimiters to separate composites (fields), sub-composites (components), and sub-sub-composites (sub-components). All HL7 2.x versions are backwards compatible.

HL7 2.x messages are broken into segments that contain specific types or categories of data. All 2.x messages begin with MSH, the message header segment where the message type is declared. The message type determines the remaining segments.

HL7 Message Example

The following code sample demonstrates an HL7 2.5.1 message containing a search for the record of Bart Simpson, DOB 01/01/01/1999. It is a Query by Parameter (QBP) message.

```
MSH|^~\&|TestApplication|DE9999|DelVAX|DE0000|20060201||QBP^Q11^QBP_Q11|DE999938854000000232|T|2.5.1|||NE|AL|||||Z34^CDCPHINVS
QPD|Z34^Request Immunization History^CDCPHINVS|||SIMPSON^BART^^^^^L||19990101||
RCP|I|5^RD^HL70126|R^real-time^HL70394
CDCPHINVS
```

HL7 Message Encoding

**Depending on system configuration, an HL7 message may need to be encrypted.** The HL7 standard requires that all lines end with a carriage return (ASCII 13, \r, or \#xD). The XML standard used by SOAP requires that lines end with a line feed (ASCII 10, \n, or \#xA), which can cause discrepancies. For more information, see the Troubleshooting guide.

SOAP Request and Response with HL7 Payload

The following code samples demonstrate a SOAP request and response with HL7 messages as the payload.

---

\(^4\) [http://www.hl7.org/](http://www.hl7.org/)
SOAP Request with example HL7 Message and optional facility ID parameter

```xml
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:urn="urn:cdc:iisb:2011">
  <soap:Header/>
  <soap:Body>
    <urn:submitSingleMessage>
      <!--Optional:-->
      <urn:username>MYUSER</urn:username>
      <!--Optional:-->
      <urn:password>MYPW</urn:password>
      <!--Optional:-->
      <urn:facilityID>MYFACILITY</urn:facilityID>
      <!--Optional:-->
      <urn:hl7Message>MSH|^~\&|^PWx|MYFACILITY|MYORG|MYORG|20140101145954||VXU^V04^VXU_V04|Q8519456876857795|P|2.5.1||AL|AL PID &lt; PHI PHI PHI&gt;|PID|1||84055037^^MR|TEST*VERIFIED|20090222|M|1|^^^US|1|20141105||IN2|1|ORC|1|2060759^HMS|
      RXA|0|1|20140101144100|20140101144100|62^HPV, quadrivalent^CVX|0.5|mL^Milliliter^ISO+||00^NEW IMM UNIZATION RECORD^NIP001|Lastname^Firstname||j011272|20160421|MSD^Merck and Co., Inc.^MVX|||CP |20140101145952
      RXR|IM^im^HL70162|RA^right arm^HL70163
      OBX|1|CE|64994-7^eligibility category^LN|1|V02^Medicaid^HL70064^V02^Medicaid^ORG_IMM_RT_251_SRC^^^Medicaid||VXC40^Eligibility captured at the Immunization level^CDCPHINVS
    </urn:hl7Message>
  </soap:Body>
</soap:Envelope>
```

SOAP Request with example HL7 Message and Response (HL7 2.4)

```xml
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:urn="urn:cdc:iisb:2011">
  <soap:Header/>
  <soap:Body>
    <urn:submitSingleMessage>
      <urn:username>MYUSER</urn:username>
      <urn:password>MYPW</urn:password>
      <urn:facilityID>MYFACILITY</urn:facilityID>
      <urn:hl7Message>MSH|^~\&|^11202078^^MR|Lastname^Firstname||19690120|M||ADDR LINE 1^DENVER^CO^ZIP ||PHONE|1|11202078||NH|1|201401011504114||VXU^V04^VXU_V04|Q8519456876857795|P|2.4.1||USA||ENG
      PID|1||11202078^^MR|Lastname^Firstname||19690120|M||ADDR LINE 1^DENVER^CO^ZIP ||PHONE|1|11202078||NH|1|201401011504114||VXU^V04^VXU_V04|Q8519456876857795|P|2.4.1||USA||ENG
    </urn:hl7Message>
  </soap:Body>
</soap:Envelope>
```
Successful Response(ACK) (HL7 2.4)

```xml
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <submitSingleMessageResponse xmlns="urn:cdc:iisb:2011">
      <return>MSH|^~\&amp;|MYORG|5.1.17^^|MYORG|||20130117||ACK^|20100115MP091688|P^|2.4^^|||ER
        MSA|AA|20100115MP091688||0||0^Message Accepted^HL70357^^^</return>
    </submitSingleMessageResponse>
  </S:Body>
</S:Envelope>
```