HL7v2 Message Validation Web Service Quick Guide (Java)

1. Create a SOAP-based web service client for connecting to the Web Service using the WSDL document at:

   <http://hl7v2.ws.nist.gov/hl7v2ws/services/soap/MessageValidationV2?wsdl>

   NOTE: Most IDE tools such as Eclipse have built-in functionality to automatically generate code that connects to and invokes the functions of a web service based on its WSDL. For example, Eclipse can generate a JAX-WS web services client using Apache CXF (which must be already installed).

2. Once created, the web service client should contain three methods which provide a programmatic interface to the functionality provided by the Web Service:

   a. ***getServiceStatus()***
   b. ***loadResource()***
   c. ***validate()***

   Detailed documentation (Java language) is available for these methods under the *Documentation* tab at the following URL: <http://hl7v2.ws.nist.gov/hl7v2ws/> (click on the button labeled *Java doc).* An example Java client is also available on this page by clicking on the button labeled *Download NIST Web Service Clients*.

   Note that a connection must first be established to the web service before the above methods are callable.   In Java the code performing the connection will look something like:

   ```
   MessageValidationV2SoapClient client = new MessageValidationV2SoapClient(
   "http://hit-testing2.nist.gov:8090/hl7v2ws/services/soap/MessageValidationV2"
   );
   ```

3. Once the client has established the connection, individual HL7 V2 messages (in XML or ER7 format) may be validated by passing them to the ***validate()*** method, along with references (by OID) to a single profile and one or more tables (value sets) resources. Optionally *a Validation Context* file may be passed in order to perform a *context-based* validation. Note that if a Validation Context file is not provided, the validation is said to be *context-free*.

   Pre-loaded resources are currently available for all NIST Meaningful Use Stage-2 HL7 messaging domains. The profiles and tables, along with their OIDs are listed under the *Documentation* tab at <http://hl7v2.ws.nist.gov/hl7v2ws/.>

An example (Java) of performing a *context-free* validation (i.e. no Validation Context file provided) of an HL7 V2 message using the pre-loaded profile and tables for the Immunization Information System Reporting (IZ) domain is shown below:

```
String iz_profile_oid="2.16.840.1.113883.3.72.2.2.99001";
String iz_table_oids=
"2.16.840.1.113883.3.72.4.2.99001:2.16.840.1.113883.3.72.4.2.99002:2.16.840.1.1
.13883.3.72.4.2.99003";
// Context-free message validation using pre-loaded IZ Profile and Tables
String res=client.validate(iz_message, iz_profile_oid, iz_table_oids, null);
```

If none of the pre-loaded resources listed above are suitable to validate a particular message, then the User may provide their own by passing them to the **loadResource()** method which will temporarily [1]store the resource and return an OID. The ***getServiceStatus()*** method returns a list of all currently loaded resources along with their OIDs.   See the Java Doc for more information.

All Profile and Table XML files must be conformant with the schemas provided in the *Support Files* table under the *Documentation* tab at: http://hl7v2.ws.nist.gov/hl7v2ws/ .

```
// Load Table 1
String userTbl1 = IOUtils.toString(new FileReader(new File("UserTable1.xml")));
String res = client.loadResource(userTbl1, null, "TABLE");
userTable1OID=parseOID(res);
// Load Table 2
String userTbl2 = IOUtils.toString(new FileReader(new File("UserTable2.xml")));
res = client.loadResource(userTbl2, null, "TABLE");
userTable2OID=parseOID(res);
// Load Profile
userProfile = IOUtils.toString(new FileReader(new File("UserProfile.xml")));
res = client.loadResource(userProfile, null, "PROFILE");
userProfileOID=parseOID(res);
// Context-free Validation of Message using Profile, Table1 and Table2
String userTableOIDs=userTable1OID + ":" + userTable2OID;
res = client.validate(message, userProfileOID, userTableOIDs, null);
```

In order to perform a *Context-based* validation, a Validation Context XML file must be provided to the validate() method. Validation Context XML files may be downloaded from the individual NIST Meaningful Use Stage 2 tools online:

- Electronic Laboratory Reporting (ELR) - http://hl7v2-elr-testing.nist.gov
- Syndromic Surveillance - http://hl7v2-ss-testing.nist.gov
- Immunization Messaging - http://hl7v2-iz-testing.nist.gov
- Lab Results Interface (LRI) - http://hl7v2-lab-testing.nist.gov

To download the Validation Context XML file for a specific test, on the web pages above click on the *Documentation* tab, followed by the link to *Test Case Documentation*, then expand the specific test case node and see the column labeled Validation Context for a link to the XML file.

---

[1] The Web Service is restarted daily at 02:55 AM EST which effectively clears all resources loaded by Users. I.e., User loaded resources are not retained for more than 24 hours.

Custom Validation Context XML files may also be created by the User. The schema for the NIST Validation Context file format is found under the *Documentation* tab at: http://hl7v2.ws.nist.gov/hl7v2ws/ (see the table entitled *Support Files*).

4. An example (Java) of performing a Context-based validation (i.e. by providing a Validation Context file) is shown below for the Laboratory Results Interface (LRI) domain is shown below:

```java
String lri_profile_oid=" 2.16.840.1.113883.9.17";
String lri_tbl_oids=
"2.16.840.1.113883.3.72.4.2.99004:2.16.840.1.113883.3.72.4.2.99005:2.16.840.1.1
.13883.3.72.4.2.99006:2.16.840.1.113883.3.72.4.2.99007:2.16.840.1.113883.3.72.4
.2.99008";
String validationContext = IOUtils.toString(new FileReader(
        new File("LRI_1.0-GU_Sed_Rate_Final_ValidationContext.xml")));
// Context-based message validation using pre-loaded LRI Profile and Tables
// for Test Case LRI 1.0-GU Sed Rate Final (LRI_1.0-GU_Final)
res=client.validate(lri_msg, lri_profile_oid, lri_tbl_oids, validationContext);
```