FED/STATE 1120 STATE CALL – MAY 5, 2006

Participants: PA, IL, MD, ID, OR, CO, MI, NYS, NYC, NE, OK, RI, IN, GA, AL, FL, WI, SC, NC, FTA

1. How do you develop instance documents (test returns with data)? using XMLSpy, etc.?

Response: Most states use XML Spy. MD uses StyleVision, an Altova tool, to create a template for keying in data for a test return, and then uses Spy to convert it to XML.

2. Do you develop one state schema covering all your main forms and supporting forms, or develop form specific schemas?

Response: MD has a schema for each primary form, and a separate schema for credits that is only validated if the primary form is valid. SC has one schema for all forms. WI has separate long form, short form, and Sub-S corporation schemas. PA has a schema for each form. NC has separate schemas for C-corporations and S-corporations, with supporting forms included with the main form.

3. Do you develop your state schema by using the existing data types/elements from the Master schema without restrictions?

Response: GA, MI, NC, MD, SC, WI all restrict. UT restricts byte size. PA is not restricting, because they do not want to reject any electronic returns; they will deal with errors on the back end.

4. Do you incorporate your state specific data restrictions into your schema, or put the data restrictions on a separate, front-end edit/validation program? what are the programming language/tools used to develop this front-end edit/validation program?

Response: SC, GA, UT restrict in the schema. WI does also, but has secondary validations for errors that schema validation can't catch. NC puts as much as possible in the schema, then uses Java for the rest. NYS noted that the more validation in the schema, the cleaner the data. States who want to reject at the front end where schema validation won't work use VB, C#, and Java.

5. Does your main processing system read in the XML data directly? If not, what do you use to parse the XML data? Do you convert the data into a flat file (to be processed by a COBOL program, for example)

Response: WI parses the XML using .Net, loads it to SAL for the back end, and extracts from the SQL tables for processing. SC stores manifest data in SQL tables, stores the XML as a BLOB, and creates a flat file to send to the mainframe. NC parses the XML using Java and stores into a database. UT uses PowerBuilder to pull the data, stores it in Oracle, then sends to the legacy mainframe. MI does similarly.

6. Do you convert the XML data into traditional database elements, or store the data in XML format?

Response: SC uses MS SQL 2000, so there is no native XML data type. They store the return as received from IRS as text data BLOB, and extract search elements. There is a native

XML data type in SQL 2005 and can search with transactional SQL. NC stores statutory data in a database, and the federal return as a BLOB. WI does similarly. PA breaks down the data several ways, sending some to imaging, some to a DB2 data warehouse, and some to IMS DB/DC for processing. NYS noted that the newest Version 9 of DB2, code named Viper, will store native XML. They will extract key elements for the legacy system, and send the rest plus the federal return to the viewing system.

7. What do you use to display the data? stylesheets, pdf, etc.

Response: SC uses PDF. The return XML is transformed in to XML matching the form using a stylesheet, then sent to an Adobe document server. PA uses a stylesheet to transform the XML into HTML; NC does basically the same thing. MI uses a custom-built application built with Delphi. UT uses PowerBuilder, and from a free-form database view superimposes fields into a bitmap of the form. Most try to display in a format that looks like the paper form.

8. Are you currently using SAML with this process?

Response: All responding states agreed that SAML is now required; it is no longer an option. The ICD gives a choice, but IRS is no longer accepting credentials in the HTTP header.

9. Do you have to modify the WSDL files in attempting to get the web service piece up and running?

Response: MD rebuilt their proxy with the latest WSDL and have not yet gotten it to work. SC always has problems with new WSDL when generating proxies; it does not compile, have to fix the method names. MI has the same experience. All three are .Net states. MD noted that IRS is adding a .Net client to test the WSDL before release. NC hasn't gotten very far, just login and logout. NYS finds that the Java client works well with the WSDL.

10. For those working in JAVA: do you create a .jar for the web services (EJBs for use with a process server), or create just a web application?

Response: NC creates a .jar file and treats it as a regular Java application, not as a web application. NYS is creating servelets that kick off at regular times.

11. Have vendors found interest from taxpayers in the program?

Response: WI is not live yet, but is working with CCH. Early in the development they talked to practitioners, taking several calls a week asking why they were dragging their feet – lots of interest. The issue is getting vendors up and running.

12. Is any state using the Oracle XML field type?

Response: NYC is working with Oracle. They are not registering the schema with Oracle, but are loading data into XML type columns. The registration failed because elements were too big; a fix is being worked on.

Additional: Jonathan Lyon has received questions from new states about the cost of the program, particularly out of pocket costs such as X12 membership, FTA-sponsored meetings, and the FTA development fee. If there is interest, he will schedule a call for state business folks to discuss these costs. Please let him know if you would like this call to be scheduled.