CONSTRUCTING THE NIF XML SCHEMA

Steven Riley Boone and Donna McKenzie
E.H. Pechan & Associates, Inc.
3622 Lyckan Parkway, Suite 2002
Durham, NC 27707
steve.boone@pechan.com; donna.mckenzie@pechan.com

ABSTRACT

The United States Environmental Protection Agency (EPA) prepares a national criteria and hazardous air pollutant (HAP) emission inventory with input from numerous states, local and tribal air agencies. These data are used for air dispersion modeling, regional control strategy development, air toxics risk assessment and tracking trends in emissions over time. The national emission inventory preparation, analysis and review is supported by two databases - the criteria National Emission Inventory (NEI) database and the HAP National Toxics Inventory (NTI) database.

State and local agencies provide their point and non-point source category emissions inventory data to the NEI through the EPA's Central Data Exchange (CDX) facility. At present, there is one acceptable and existing data transfer format for submitting data to the NEI - the NEI Input Format (NIF). However, the NIF data transmittal format can be provided in three separate file types: ASCII flat files, Microsoft Access or eXtensible Markup Language (XML). The NIF has recently been updated to implement relevant final data standards required by the Agency and which are administered by the Office of Environmental Information (OEI). The NIF Version 3.0 was published in spring of 2003, updated in November 2003 and complies with the current final standards.

In developing the NEI XML standards, a roadmap from the XML to the existing NIF format (ASCII) was developed and finally the XML schema. The NEI XML schema was presented to the TRG (Technical Resource Group) for conformance review and to an initial group of pilot states for testing using the full compliment of the Exchange Network nodes and EPA' Central Data eXchange (CDX). The project also includes the development of a translator utility to translate the data into a format that could be read by the existing tools.

INTRODUCTION

The United States Environmental Protection Agency (EPA) prepares a national criteria and hazardous air pollutant (HAP) emission inventory with input from numerous states, local and tribal air agencies. These data are used for air dispersion modeling, regional control strategy development, air toxics risk assessment and tracking trends in emissions over time. The national emission inventory preparation, analysis and review is supported by two databases - the criteria National Emission Trends (NET) database and the HAP National Toxics Inventory (NTI) database.

State and local agencies provide their point and non-point source category data to the NEI through the EPA's Central Data Exchange (CDX) facility. At present, there are two acceptable and existing data transfer formats for submitting data to the NEI - the NEI Input Format (NIF) and extensible markup language (XML). The NIF has recently been updated to implement relevant final data standards required by the Agency and which are administered by the Office of Environmental Information (OEI). The NIF Version 3.0 was published in spring of 2003, revised in November 2003 and complies with the current final standards. The NEI XML schema is closely aligned with the NIF in terms of data content and relationships and has been adjusted for the new NIF Version 3.0. The NIF data standards (ASCII and Access formats) can be found at http://www.epa.gov/ttn/chief/nif/index.html#qa.

In developing the NEI XML standards, a roadmap from the existing NIF format (ASCII) to the XML elements was developed and finally the XML schema. The NEI XML Version 3.0 schema was originally based on the NEI XML Version 2.0 schema; however, in addition to the changes in required by the data element changes from NIF 2.0 to NIF 3.0, the NEI XML Version 3.0 has been structurally redesigned. The NEI XML schema was presented to the TRG (Technical Resource Group) for conformance review and to an initial group of pilot states for testing using the full compliment of the Exchange Network nodes and EPA's Central Data eXchange (CDX). The project also includes the development of a translator utility to translate the data into a format that could be read by the existing tools.

Throughout this paper, links are listed for specific items of data. Over time, these links may be changed or the data available may vary. In general, the two high-level links which will be most useful are http://www.epa.gov/ttn/chief/nif/index.html (the Technology Transfer Network Clearinghouse for Emissions Inventories and Emission Factors National Emission Input Format site) and http://www.exchangenetwork.net/common/default.asp (the National Environmental Exchange Network site).

PROCESS

The distinct stages of the development of the NEI XML Version 3.0 schema can be summarized as follows:

- Schema Development (including Data Element Mapping and TRG Conformance Review)
- Release of Schema to Pilot Submitting Agencies
- Creation and Validation of XML Instance Documents by Pilot Submitting Agencies
- Submission of XML Documents by Pilot Submitting Agencies through CDX
- Official Release of NEI Schema
- Development of XML Translator Utility

While these stages are listed in order, in actual process, each stage was iterative and overlapped with the other stages.

Schema Development

The first step in the development of the NEI XML Version 3.0 Schema was to map each element in the schema to both the NIF Version 3.0 standards (available at http://www.epa.gov/ttn/chief/nif/index.html) and the applicable data standards of the Environmental Data Registry (EDR) (available at http://oaspub.epa.gov/edr/epastd$.startup). Both naming convention standards and data type standards were compared to ensure compatibility between the NEI XML Version 3.0 Schema and the NIF Version 3.0 data standards as well as conforming to the EDR data standards for XML data elements where applicable. The results of this mapping are in the NEI XML - NIF Element Cross Reference file.

The second step in the development of the NEI XML Version 3.0 schema was to review the existing NEI XML Version 2.0 schema and the results of the TRG conformance review performed on this version in September 2003. The TRG reviews XML schemas that are submitted for use at the Exchange Network. This review checks for conformance to established standards which include compliance with W3C (World Wide Web Consortium) schema requirements, the Exchange Network's XML Design Rules and Convention (DRC), and modularity as referenced in the Core Reference Model (CRM).[1] Further information on the TRG's XML schema review process is available at http://www.exchangenetwork.net/common/default.asp under the Resources link.

The primary change required by the TRG-based review of the NIF Version 2.0 Schema was the addition of a targeted namespace, fully qualified elements, and versioning information. 'An XML namespace is a collection of names, identified by a URI reference [RFC2396], which are used in XML documents as element types and attribute names.'[2] A targeted namespace shows that the schema is grouping the components of a schema. Qualifying elements indicates which namespace is reference in using the element.

In addition to these requirements, it was strongly suggested in EPA XML design documents that data-centric XML schemas should include key and key references to enforce uniqueness and referential integrity. Keys are used to enforce uniqueness of data within a record - for example, only one instance of a particular state, county, tribe, and SCC can be submitted within an Emission Process record for nonpoint emission inventory data. Referential integrity ensures that a detail record has an appropriate 'parent' record - for example, an entry in the Emission table for nonpoint record has an appropriate related record in the Emission Period and Emission Process tables. XML implements uniqueness and referential integrity through the key and keyref approaches, where key indicates the definition of uniqueness and keyref defines the relationship between element groups referring to a defined key. It was also suggested that common elements should be held in a common schema and referenced (included / imported) in other schemas. The organization of the schemas is as follows:

- EN_NEI_AreaNonroad_v3_0.xsd
- EN_NEI_Biogenic_v3_0.xsd
- EN_NEI_Onroad_v3_0.xsd
- EN_NEI_Point_v3_0.xsd
- EN_NEI_Common_v3_0.xsd

where the source-specific information is carried in the schemas identified as such, and elements common to all schemas are carried in the 'Common' schema. This has served the purpose of streamlining maintenance and QA of the schema.

In addition to these EPA standard design issues, the September 2003 schema was designed so that each 'table' had multiple levels of organization. The multi-level approach is useful for display purposes, but it added complexity to the conversion process. Essentially, the schema was 'flattened.' The example below indicates the difference in structure before 'flattening' of the schemas and after for a portion of the point source Emission Release (ER) Point group.

Before (the example below only expands some of the grouped data)

```
    <xsd:complexType name="EmissionReleasePointSubmissionGroupType">
        <xsd:sequence>
            <xsd:element ref="RecordTypeCode"/>
            <xsd:element ref="EmissionReleasePointKeyFieldsGroup"/>
            <xsd:element ref="TransactionSubmittalCode" />
            <xsd:element ref="EmissionReleasePointDetails"/>
        </xsd:sequence>
    </xsd:complexType>
..... (defined separately)
    <xsd:complexType name="EmissionReleasePointDetailsType">
        <xsd:sequence>
            <xsd:element ref="ReleasePointTypeCode"/>
            <xsd:element ref="ReleasePointDescription" minOccurs="0"/>
            <xsd:element ref="StackDetails" minOccurs="0"/>
            <xsd:element ref="ExitGasDetails" minOccurs="0"/>
            <xsd:element ref="GeographicCoordinateDetails"/>
            <xsd:element ref="FugitiveDetails" minOccurs="0"/>
        </xsd:sequence>
    </xsd:complexType>
..... (defined separately)
    <xsd:complexType name="FugitiveDetailsType">
        <xsd:sequence>
```

```
                <xsd:element ref="FugitiveUnitOfMeasureCode" minOccurs="0"/>
                <xsd:element ref="FugitiveHorizontalAreaValue" minOccurs="0"/>
                <xsd:element ref="FugitiveReleaseHeightValue" minOccurs="0"/>
            </xsd:sequence>
        </xsd:complexType>
..... (defined separately)
        <xsd:complexType name="GeographicCoordinateDetailsType">
            <xsd:sequence>
                <xsd:element ref="LongitudeMeasure"/>
                <xsd:element ref="LatitudeMeasure"/>
                <xsd:element ref="HorizontalAccuracyMeasure" />
                <xsd:element ref="HorizontalCollectionMethodCode" />
                <xsd:element ref="HorizontalReferenceDatumCode" />
                <xsd:element ref="ReferencePointCode" />
                <xsd:element ref="SourceMapScaleNumber" minOccurs="0"/>
                <xsd:element ref="CoordinateDataSourceCode" minOccurs="0"/>
                <xsd:element ref="XYCoordinateTypeCode"/>
                <xsd:element ref="UTMZoneCode"/>
            </xsd:sequence>
        </xsd:complexType>
.....
```

After

```
        <xsd:complexType name="EmissionReleasePointSubmissionGroupType">
            <xsd:sequence>
                <xsd:element ref="nei:EmissionReleasePointRecordTypeCode"/>
                <xsd:element ref="nei:CountyStateFIPSCode"/>
                <xsd:element ref="nei:StateFacilityIdentifier"/>
                <xsd:element ref="nei:ReleasePointIdentifier"/>
                <xsd:element ref="nei:ReleasePointTypeCode"/>
                <xsd:element ref="nei:StackHeightValue" minOccurs="0"/>
                <xsd:element ref="nei:StackDiameterValue" minOccurs="0"/>
                <xsd:element ref="nei:StackFencelineDistanceValue" minOccurs="0"/>
                <xsd:element ref="nei:ExitGasTemperatureValue" minOccurs="0"/>
                <xsd:element ref="nei:ExitGasStreamVelocityRate" minOccurs="0"/>
                <xsd:element ref="nei:ExitGasFlowRate" minOccurs="0"/>
                <xsd:element ref="nei:LongitudeMeasure" minOccurs="0"/>
                <xsd:element ref="nei:LatitudeMeasure" minOccurs="0"/>
                <xsd:element ref="nei:UTMZoneCode" minOccurs="0"/>
                <xsd:element ref="nei:XYCoordinateTypeCode" minOccurs="0"/>
                <xsd:element ref="nei:FugitiveHorizontalAreaValue" minOccurs="0"/>
                <xsd:element ref="nei:FugitiveReleaseHeightValue" minOccurs="0"/>
                <xsd:element ref="nei:FugitiveUnitOfMeasureCode" minOccurs="0"/>
                <xsd:element ref="nei:ReleasePointDescription" minOccurs="0"/>
                <xsd:element ref="nei:TransactionSubmittalCode" minOccurs="0"/>
                <xsd:element ref="nei:HorizontalCollectionMethodCode" minOccurs="0"/>
                <xsd:element ref="nei:HorizontalAccuracyMeasure" minOccurs="0"/>
                <xsd:element ref="nei:HorizontalReferenceDatumCode" minOccurs="0"/>
```

```
            <xsd:element ref="nei:ReferencePointCode" minOccurs="0"/>
            <xsd:element ref="nei:SourceMapScaleNumber" minOccurs="0"/>
            <xsd:element ref="nei:CoordinateDataSourceCode" minOccurs="0"/>
            <xsd:element ref="nei:TribalCode"/>
        </xsd:sequence>
        <xsd:attribute name="schemaVersion" type="xsd:decimal" use="required"/>
    </xsd:complexType>
```

These changes (in addition to changes required by changes to the NIF elements) were made to the schema and the schema was resubmitted to the TRG for conformance review.[3] Additionally, some basic QA pattern matching was added for certain fields such as SIC (Standard Industrial Classification) and SCC (Source Code Classification).

The results of this review and iterations with pilot submitting agencies required some changes to the schema. For example, 'xs' to 'xsd' was changed as a namespace prefix for all W3C Schema constructs. Target namespace was changed to the appropriate namespace for the exchange network. Schema header documentation was added. Alterations were made in the mandatory status of some elements, as well as resolving data type issues, changing some data types to 'double' to manage scientific notation, and reordering the elements with the submission groups to more closely resemble the order of the elements in the NIF format. In general, the elements that are mandatory in the XML schemas are the key values, values related directly to the emission amount, and certain identifying information in the transmittal submittal group and the site submittal group (for the point source data).

During this iterative process of refining the schemas, two specific issues came to light which highlighted the complexities of schema development and use in multiple environments. The first issue was the addition of referential integrity to the schema. This schema is intended for use only for initial submittals of data, therefore it was appropriate (and required by EPA standards) to include referential integrity checks. These referential integrity checks were implemented using the key/keyref approach.

The development environment for the schemas was XMLSpy (Altova). It was within this development environment that the schemas were validated and tested with XML instance documents. However, it was found during the conformance review that the validation software - Xerces (Apache Software Foundation), was more restrictive in its interpretation of W3C standards concerning the scope of key reference definitions. The schemas failed validation under that standard, and the key reference definitions were relocated within the schemas in order to conform to the Xerces requirement.

The second issue which came to light was the potential need for a submitting agency to submit empty tags. In XML, unless a tag is required as mandatory it does not need to be included (however, order of tags does matter). Therefore, if a state, for example, never collects certain non-mandatory information, the tag should not be included in the XML instance document that it submitted. However, in many cases, data is available inconsistently for a field which would leave the state with two options - conditionally include the tag only if it is non-null or include the tag as empty. This second option was initially incompatible with the design of the schema and data type were redefined in order manage potential "empty" tags. Character fields were redefined to permit a length of zero (unless mandatory) and numeric fields (decimal or integer) were redefined as 'double' and then pattern matched to integer or decimal or double as appropriate. This method permitted empty tags in both character and numeric fields when validating against XMLSpy, however, it did not permit them when validating an instance document against the schema when using the CDX submittal process (since that method uses a different XML parsing software). It has been proposed that if a state is unable to remove the empty tags programmatically in the course of the generation of the XML instance document, empty tags can be removed through find/replace editing process (since the empty tags will have the same format wherever they are found) in the document. However, as with the previous issue it points out that XML validation software packages can vary in their implementation of W3C standards.

Instance Document Creation

Agencies submitting data use the XML Schema definition in two ways - first, an initial mapping of the data elements in their system, and then to validate the resulting XML instance document in order to ensure that it conforms to the NEI XML Schema. Many different software packages can be used to generate an XML instance document from existing database, or in some cases the XML instance document is created wholly programmatically. The method used may vary by state, but the end result should be an XML instance document that conforms to W3C standards, and conforms to the NEI XML Version 3.0 schema. An instance document may be valid structurally, however may fail when validated against the NEI XML Version 3.0 schema for uniqueness, referentially integrity, tag element naming or order errors, data type and size errors, and pattern matching errors (for example, including a letter in a date field).

Submission to EPA

Once the NEI XML Version 3.0 has officially been released, its location will be posted at http://oaspub.epa.gov/emg/xmlsearch$.startup under the National Emission Inventory link along with appropriate target namespace information and location information in order to permit states to structure valid references in their instance documents. Upon official release, the process for submitting data to the EPA has the following steps:

- The XML instance document is submitted for validation at http://naas.epacdxnode.net and http://naas.epacdxnode.net/xml/validator.wsdl. This is applicable for CDX submitters and Node submitters.
- The XML instance document is then submitted through either the CDX process or the Node process as appropriate for the submitting agency. If submitting through the Node, certain additional header information is required in the XML instance document.

Additional information about CDX can be found at http://www.epa.gov/cdx/ and additional information about the Node process can be found at http://www.exchangenetwork.net/common/default.asp.

XML Translator Utility

Upon receipt of the XML submissions, the data (at this point) will need to be converted to NIF 3.0 ASCII in order for current tools to import and review the data. There was, therefore, the need to include the development of a translator utility as part of this project. The development environment is Microsoft Visual Basic using Microsoft's XML Parser (MSXML). To summarize the technical approach, the translator will accept a file path/file name from the user and use XML Stylesheet Transformations (XSLT) to transform the file from an XML document to a NIF 3.0 compliant ASCII file.

In order to manage the size of the files expected for XML submissions (large files may be expected to reach 100 MB or more), it is appropriate to select an appropriate document management approach. There are three types of methods of parsing XML documents:

- DOM (Document Object Model) 'is a tree-based parsing technique that builds up an entire parse tree in memory. It allows complete, dynamic access to a whole XML document.'[4] The down side is that for large document the memory requirement may be overwhelming.
- SAX (Simple API for XML) 'is an event-driven push model for processing XML. Rather than building a tree representation of an entire document as DOM does, a SAX parser fires off a series of events as it reads through the document.'[5] This method is very memory efficient, however, because the application needs to manage the details of the documentation hierarchy, application logic may become complex. It is a useful method for one-time passes through the document, but not recommended for modifying documents.
- StAX (Streaming API for XML) 'is a new parsing technique that, like SAX, uses an event-driven model. However, instead of using SAX's push model, StAX uses a pull model for event processing. Instead of using a callback mechanism, a StAX parser returns events as requested by the application.

StAX also provides user-friendly APIs for read-in and write-out.' [6] Essentially, it provides the memory efficiency of SAX, while also providing some of the ease of document management access that DOM provides.

Based on the needs of the XML Translator utility, it was determined that a combination of both DOM and SAX would work best (since the XML instance document does not require modification). A SAX parser is used to select out a specific submission group (for example Emission Unit) and a DOM tree is built for that specific submission group. In this manner, the XML instance document is never read into the utility in its entirety, and memory efficiency issues are reduced.

After the DOM subdocument has been created, this document is transformed into an ASCII document using a XSLT document.

SUMMARY

In summary, the NEI XML Version 3.0 schema has been developed and refined for the purposes of submitting the NEI data to the EPA has been completed. Throughout this process, the importance of maintaining consistent definitions and mapping between NIF submission formats and the XML schema document. This process also highlight the need to be aware of the differences in XML validation and parsing software implementations, and that multiple testing environments may be required for a complete review of schema behavior.

This process has also served as an implementation guide for the steps required in the development and implementation of schemas for future EPA data flows. For example, the development of the 'Common' schema may provide the groundwork for a 'Common' schema which could be implemented across many EPA data flows, thus reducing the chance of incompatibilities in data types, and allowing multiple data flows to be integrated in common systems.

REFERENCES

1. National Air Emission Inventory (NEI) XML Schema Revisions 3.0 Schema Review Conformance Report (Draft) Prepared for The Office of Environmental Information (OEI) United States Environmental Protection Agency (USEPA) Prepared by Computer Sciences Corporation 15000 Conference Center Drive Chantilly, VA 20151 February, 2004.
2. Namespaces in XML World Wide Web Consortium 14-January-1999 REC-xml-names-19990114 http://www.w3.org/TR/REC-xml-names/
3. National Air Emission Inventory (NEI) XML Schema Revisions 3.0 Schema Review Conformance Report (Draft) Prepared for The Office of Environmental Information (OEI) United States Environmental Protection Agency (USEPA) Prepared by Computer Sciences Corporation 15000 Conference Center Drive Chantilly, VA 20151 February, 2004.
4. Parsing XML Efficiently by Ping Guo, Julie Basu, Mark Scardina, and K. Karun http://otn.oracle.com/oramag/oracle/03-sep/o53devxml.html 2004
5. Parsing XML Efficiently by Ping Guo, Julie Basu, Mark Scardina, and K. Karun http://otn.oracle.com/oramag/oracle/03-sep/o53devxml.html 2004
6. Parsing XML Efficiently by Ping Guo, Julie Basu, Mark Scardina, and K. Karun http://otn.oracle.com/oramag/oracle/03-sep/o53devxml.html 2004

KEYWORDS
CDX
Central Data Exchange
DOM
Exchange Network
National Emission Inventory
National Emission Inventory Format
NEI
NIF
Node
Parsing
SAX
Schema
StAX
Validation
XML
XSLT