NISTIR 7547

NDRProfile Schema Version 1.0 User Guide

Betty Harvey
Electronic Commerce Connection, Inc.
Email: harvey@eccnet.com

Joshua Lubell
Puja Goyal
KC Morris
Manufacturing Systems Integration Division
National Institute of Standards and Technology
Email: {lubell,pgoyal,kcm}@nist.gov

December 2008
Abstract
The NDRProfile schema provides a common syntax for exchanging, managing, and reusing XML Schema naming and design rules (NDRs). NDRProfile, used by NIST’s Quality of Design (QOD) application as a format for import and export of rule sets, is also useful for general exchange of NDRs and, with accompanying stylesheets and other materials, provides an environment for authoring and formatting NDR documentation. This paper describes the content of the NDRProfile schema and its various uses.

Table of Contents
1. Introduction .......................................................................................................................... 2
2. NDRProfile Schema Elements .......................................................................................... 3
   2.1. Classification ................................................................................................................ 4
   2.2. Comment ....................................................................................................................... 4
   2.3. ContactURI ................................................................................................................... 4
   2.4. Coverage ...................................................................................................................... 4
   2.5. DerivedFrom ................................................................................................................ 5
   2.6. Description .................................................................................................................... 5
   2.7. EnforcementLevel ........................................................................................................ 6
   2.8. Guidance ....................................................................................................................... 6
   2.9. GuidanceName ............................................................................................................. 8
   2.10. GuidanceText ............................................................................................................. 8
   2.11. Justification ................................................................................................................ 8
   2.12. NDRProfile ................................................................................................................ 8
   2.13. NDRTitle .................................................................................................................... 9
   2.14. OrganizationInformation ............................................................................................ 10
   2.15. OrganizationName ...................................................................................................... 10
   2.16. OrganizationPrefix ..................................................................................................... 10
   2.17. ProfileDerivedFrom ................................................................................................... 10
   2.18. Rationale ..................................................................................................................... 11
   2.19. RuleVersion ................................................................................................................. 11
   2.20. SchemaExample ......................................................................................................... 11
   2.21. Scope .......................................................................................................................... 11
   2.22. Script .......................................................................................................................... 11
   2.23. SourceDate ................................................................................................................ 12
   2.24. SourceNDR ................................................................................................................ 12
   2.25. SourceName ................................................................................................................. 12

NDRProfile Schema
Version 1.0 User Guide

Betty Harvey <harvey@eccnet.com>
Joshua Lubell <lubell@nist.gov>
Puja Goyal <pgoyal@nist.gov>
KC Morris <kcm@nist.gov>
1. Introduction

XML schema [1] Naming and Design Rules (NDRs) [2] are intended to help organizations enforce best practices, a particular modeling methodology, or conformance to common nomenclature policies. But there are barriers that can prevent NDRs from improving schema interoperability and quality:

- NDR proliferation - no single rule set meets everyone's requirements.
- Absence of NDR document structure - NDR documentation typically is created using a word processor and exists only in a proprietary binary format.
- Lack of automated rule enforcement - the onus is on XML schema developers to ensure that their schemas obey (sometimes ambiguously worded) NDR rules.
- Limited versioning and traceability - it is cumbersome to compare NDRs with one another or to explicitly specify rule reuse.

NIST created the Quality of Design (QOD) web application (http://www.nist.gov/qod) to address these barriers. QOD provides a mechanism for checking XML schema design quality against NDRs in a collaborative environment. Plans for the next version include better support for sharing and publication of rules with the intention of maintaining an extensive set of reusable test cases.

Recognizing that some groups might want to develop their own XML schema design quality testing tools rather than use QOD, NIST subsequently created NDRProfile - an XML schema providing a common format in which NDRs can be exchanged, managed and reused. NDRProfile provides the following capabilities:
• Import and export of rule collections into and from the QOD.

• The ability to publish NDR documentation from XML.

• The ability to exchange NDR rule collections, including tests for the rules.

The NDRProfile schema's target audience includes NDR authors, groups wishing to exchange, share, or reuse rule sets, and users of schema quality assurance tools such as QOD that import/export XML conforming to the NDRProfile. Although the NDRProfile schema's development was inspired by QOD, the schema can be used independently of QOD, either as a means of authoring an NDR document or as a vehicle for sharing rules between NDR Profile schema-compliant software applications.

This document is intended both for NDR authors as well as for developers of NDR authoring or testing software. The document contains the following sections:

• Section 2, “NDRProfile Schema Elements” provides information about the elements and attributes that are available in the NDRProfile schema.

• Section 3, “Developing an NDRProfile Instance” describes the various methods for developing an NDRProfile document.

• Section 4, “NDRProfile and QOD” describes QOD and discusses using NDRProfile for import and export. Non-QOD users may wish to skip reading this section.

• Section 5, “Lessons Learned and the Future of NDRProfile” concludes the user guide with some thoughts on improving NDRProfile.

The NDRProfile schema and the accompanying stylesheets and resources mentioned in Section 3, “Developing an NDRProfile Instance” are available in a single package available for download from http://qod.sourceforge.net.

2. NDRProfile Schema Elements

This section provides an alphabetical listing of all elements and attributes included in the NDRProfile XML Schema. Each element is defined in its own subsection. The information provided for each element includes:

1. Element name

2. Brief definition of each element

3. Data type if the element has simple content (character data with no sub-elements)

4. A graphical representation of the content if the element contains sub-elements

5. A table of attributes providing information about each attribute, if the element has attributes

6. A list of parent elements.

The table below shows how to interpret the symbols used in content diagrams.
The best way for a user to get an overview of the schema is to first read the definition for NDRProfile, the root element of an NDRProfile document instance. Another good element to look at for gaining an overall sense of the schema is Guidance.

2.1. Classification

**Definition:** Classification specifies a category for the guidance rule. The taxonomy of categories is determined by the NDR developers.

**Data type:** xsd:string

**Parent Elements**

- Guidance

2.2. Comment

**Definition:** Comment is used to specify any information about a guidance rule not specifiable using other NDRProfile elements.

**Data type:** xsd:string

**Parent Elements**

- Guidance

2.3. ContactURI

**Definition:** ContactURI is used to provide a web or email address for an organizational point of contact.

**Data type:** xsd:anyURI

**Parent Elements**

- OrganizationInformation

2.4. Coverage

**Definition:** Coverage defines what portion of the rule the test case covers.
Data type: xsd:string

Parent Elements

- TestCase

2.5. DerivedFrom

Definition: DerivedFrom contains information about the original source of the rule when this is a new rule that is a modification of a rule from a different NDR. This element is used for traceability.

Content:

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Definition</th>
<th>Allowed Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>QODUniqueIdentifier (optional)</td>
<td>QODUniqueIdentifier is an optional attribute containing a unique identifier assigned by the QOD web application. This attribute is populated by QOD when the rule is exported from QOD into a valid NDRProfile instance.</td>
<td>xsd:string</td>
<td>None</td>
</tr>
<tr>
<td>URI</td>
<td>URI is an XML unique ID type. The first character of an XML ID must be a letter followed by any XML name characters (letter, number, _, - ,</td>
<td>xsd:anyURI</td>
<td>None</td>
</tr>
</tbody>
</table>

Parent Elements

- Guidance

2.6. Description

Definition: Description specifies text documenting the NDR.

Data type: xsd:string

Parent Elements

- NDRProfile
2.7. EnforcementLevel

**Definition:** EnforcementLevel defines whether the guidance rule is mandatory, optional, or recommended.

**Data type:** Mandatory, Optional, or Highly_Recommended

**Parent Elements**

- Guidance

2.8. Guidance

**Definition:** Guidance is the top level element for an NDR rule. It represents a test requirement.

**Content:**
### NDRProfile Schema
Version 1.0 User Guide

<table>
<thead>
<tr>
<th>Attribute Name (optional)</th>
<th>Definition</th>
<th>Allowed Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>documentationID</td>
<td>documentationID is a link to a unique identifier within the NDRDocumentation. The use of the &quot;NDRDocumentBaseURI&quot; and the &quot;documentationID&quot; attributes provides the mechanism to link internally to the NDR documentation from the NDRProfile schema.</td>
<td>xsd:string</td>
<td>None</td>
</tr>
<tr>
<td>guidanceID</td>
<td>guidanceID is a link to a unique identifier within the NDRProfile document and should be unique for each rule or guidance.</td>
<td>xsd:ID</td>
<td>None</td>
</tr>
</tbody>
</table>

* A guidanceID must conform to additional constraints if the NDRProfile is to be imported into NIST’s QOD web application.

**Parent Elements**
- NDRProfile

### 2.9. GuidanceName

**Definition:** GuidanceName defines the succinct title for the rule.

**Data type:** xsd:string

**Parent Elements**
- Guidance

### 2.10. GuidanceText

**Definition:** GuidanceText is an English language specification of the rule.

**Data type:** xsd:string

**Parent Elements**
- Guidance

### 2.11. Justification

**Definition:** Justification provides additional supporting information, beyond the value for Rationale, regarding the reason for the guidance.

**Data type:** xsd:string

**Parent Elements**
- Guidance

### 2.12. NDRProfile

**Definition:** NDRProfile is the root element for a document conforming to the NDRProfile schema.
### Attribute Name | Definition | Allowed Values | Default Value
--- | --- | --- | ---
NDRProfileID | NDRProfileID is a unique identifier for the NDR. | xsd:ID | None
NDRProfileBaseURI (optional) | NDRProfileBaseURI contains the location of the NDRProfile. | xsd:anyURI | None
NDRDocumentBaseURI (optional) | NDRDocumentBaseURI points to the location of the documentation for the NDR. It is envisioned that the documentation will be encoded using DocBook. | xsd:anyURI | None

### Parent Elements
- Root Element

### 2.13. NDRTitle

**Definition:** NDRTitle specifies a brief description of the rule set. If the NDRProfile represents a formal NDR document, NDRTitle should contain the document's official title.

**Data type:** xsd:string

**Parent Elements**
- NDRProfile
2.14. OrganizationInformation

**Definition:** OrganizationInformation specifies data about the entity responsible for the rule set or guidance rule.

**Content:**

```plaintext
<OrganizationInformation>
  <OrganizationName>0..1</OrganizationName>
  <OrganizationPrefix>0..1</OrganizationPrefix>
  <ContactURI>0..1</ContactURI>
</OrganizationInformation>
```

**Parent Elements**

- NDRProfile
- Guidance

2.15. OrganizationName

**Definition:** OrganizationName defines the name for an organization.

**Data type:** xsd:string

**Parent Elements**

- OrganizationInformation

2.16. OrganizationPrefix

**Definition:** OrganizationPrefix defines a character sequence for an organization. This character sequence is used internally by QOD to create identifiers for guidance created within QOD by the organization.

**Data type:** xsd:string

**Parent Elements**

- OrganizationInformation

2.17. ProfileDerivedFrom

**Definition:** ProfileDerivedFrom specifies the original source of the NDR.

**Content:**

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Definition</th>
<th>Allowed Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>URI</td>
<td>URI is a valid URI.</td>
<td>xsd:anyURI</td>
<td>None</td>
</tr>
</tbody>
</table>
Parent Elements

• NDRProfile

2.18. Rationale

**Definition:** Rationale provides a reason for creating the rule. Although Rationale's data type permits any string value, our intention for the next version of NDRProfile is to restrict the value to an enumerated set similar to that proposed in [3].

**Data type:** xsd:string

Parent Elements

• Guidance

2.19. RuleVersion

**Definition:** RuleVersion defines the version number of the guidance rule.

**Data type:** xsd:integer

Parent Elements

• Guidance

2.20. SchemaExample

**Definition:** SchemaExample associates a sample schema with a guidance rule in order to give an example of rule compliance or violation.

**Data type:** xsd:string with white space preserved

Parent Elements

• Guidance

2.21. Scope

**Definition:** Scope defines the level of granularity of the guidance, i.e., whether it applies to a single value or field, an aggregate component (such as a complex type definition), or to the XML schema as a whole.

**Data type:** xsd:string

Parent Elements

• Guidance

2.22. Script

**Definition:** Script specifies code implementing a test case. The code should be written in the language specified by the TestCase element's ruleType attribute.

**Data type:** xsd:string with white space preserved
Parent Elements

- TestCase

### 2.23. SourceDate

**Definition:** SourceDate is the date the derived rule was published.

**Data type:** xsd:date

### 2.24. SourceNDR

**Definition:** SourceNDR specifies the original source of the rule. It is used to indicate if the rule is a copy of a rule already existing in another NDR.

**Content:**

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Definition</th>
<th>Allowed Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>QODUniqueIdentifier</td>
<td>QODUniqueIdentifier is an optional attribute which contains the unique identifier assigned by QOD. This attribute will be populated by QOD when the rule is exported from QOD into a valid NDRProfile instance.</td>
<td>xsd:string</td>
<td>None</td>
</tr>
<tr>
<td>URI</td>
<td>URI is an XML unique ID type. The first character of an XML ID must be a letter followed by any XML name characters (letter, number, -, _,</td>
<td>xsd:anyURI</td>
<td>None</td>
</tr>
</tbody>
</table>

### 2.25. SourceName

**Definition:** SourceName is the name of the NDR the rule was derived from.
Data type: xsd:string

Parent Elements

- DerivedFrom
- SourceNDR

### 2.26. SourceVersion

**Definition:** SourceVersion is the version of the NDR the rule was derived from.

Data type: xsd:string

Parent Elements

- DerivedFrom
- SourceNDR

### 2.27. Status

**Definition:** Status specifies the standing of a guidance rule or NDRProfile, e.g., "Draft."

Data type: xsd:string

Parent Elements

- NDRProfile
- Guidance

### 2.28. SupportingGuidance

**Definition:** SupportingGuidance allows the association of additional information with a guidance rule.

Data type: xsd:string

Parent Elements

- Guidance

### 2.29. SupportingGuidanceURI

**Definition:** SupportingGuidanceURI allows a URI to be provided for additional documentation for the guidance.

Data type: xsd:anyURI

Parent Elements

- Guidance
2.30. Testability

**Definition:** Testability defines whether a rule can be programmatically evaluated. Actual assessment or confirmation of a rule's quality is outside the scope of NDRProfile.

**Data type:** Fully-Testable, Partially-Testable, or Untestable

**Parent Elements**
- Guidance

2.31. TestCase

**Definition:** TestCase is a computer executable script, i.e., a binding of the guidance to a specific implementation. A test case determines whether an XML schema complies with or violates the guidance rule.

**Content:**

```
<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Definition</th>
<th>Allowed Values</th>
<th>Default Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ruleType</td>
<td>ruleType identifies the type of validation rule that has been created. If &quot;Other&quot; is selected, the attribute 'typeOther' should be used to identify the type of validation rule that was created.</td>
<td>ISO-Schematron, Schematron1.5, Jess, Other</td>
<td>None</td>
</tr>
<tr>
<td>typeOther</td>
<td>typeOther identifies a validation rule that has been created in another language besides Schematron [4] [5] and Jess [6].</td>
<td>xsd:string</td>
<td>None</td>
</tr>
<tr>
<td>testCaseID</td>
<td>The testCaseID is a unique identifier for the test case. It must conform to the XML ID rule construct.</td>
<td>xsd:ID</td>
<td>None</td>
</tr>
</tbody>
</table>
```
Parent Elements

- TestCases

2.32. TestCases

Definition: TestCases defines one or more test cases.

Content:

Parent Elements

- Guidance

2.33. TestName

Definition: TestName defines a name that a user can assign to a test.

Data type: xsd:string

Parent Elements

- TestCase

2.34. URI

Definition: URI defines a link to a single test case.

Data type: xsd:anyURI

Parent Elements

- TestCase

3. Developing an NDRProfile Instance

This section describes various ways users can develop a document conforming to the NDRProfile schema document. The conforming document can then be formatted for viewing using the NDRProfile-to-HTML.xsl stylesheet, imported into QOD (see Section 4, “NDRProfile and QOD”), or imported into some other NDRProfile-aware application.

NDRProfile-to-HTML.xsl and other files mentioned in this section are listed in Appendix A, NDRProfile Resources, and are included in the NDRProfile download package on SourceForge.

3.1. DocBook Documentation

Organizations can develop their NDR using the standard DocBook schema. DocBook [7] is a standard for developing technical documentation. We have developed an XML Style Language Tranformation (XSLT) [8] stylesheet that extracts the rules from a DocBook instance and produces XML conforming to the NDRProfile schema.
Several standards bodies, such as the Universal Business Language Technical Committee [http://www.oasis-open.org/committees/ubl], have started developing and maintaining their NDRs using the DocBook standard. The use of DocBook XML can provide a mechanism for automatically extracting an NDRProfile document from the full NDR documentation.

We have developed an XSLT stylesheet (NDRDocumentDocbook-to-NDRProfile.xsl) to convert a DocBook tagged NDR to the NDRProfile XML document. You should be able to use any XSLT processor to create an NDRProfile xml file from a DocBook XML document, provided the DocBook is tagged as described below.

The figure below shows how rules are extracted from the DocBook XML document. The resulting output from the XSLT extraction is a fully conformant NDRProfile XML document.

**Figure 1. DocBook Documentation to NDRProfile**

For this method to be used, rules and rule prefixes must be identified within the document instance using DocBook's `role` attribute. The XSLT stylesheet uses the `role` values to transform the DocBook XML into NDRProfile guidance rules.

The XSLT extraction uses the `blockquote` element with a `role` attribute value of `Rule` to extract the rule. The `blockquote` element must also include a unique identifier for uniquely identifying rules.

**Note**

Both the `role` and `id` attributes are optional attributes within the DocBook schema. Therefore, you must ensure that all rules include the `role` and `id` attributes. We have developed a
Schematron [4] schema, CheckDocbookNDRRules.sch, that can be used to ensure that the blockquote element includes the required information to support conversion to an NDRProfile.

**Example 1. Using blockquote with role and id attributes to define a rule**

```
<blockquote role="Rule" id="rule-ELD2">
  ...
</blockquote>
```

DocBook uses the para element to define a paragraph. The text documenting the rule should be included in a para element.

If your NDR standard uses rule prefixes to disambiguate rules within your NDR, the emphasis element should be used with the role attribute set to rule-prefix within the para element. In the following example, the NDR uses "ELD2" as a unique identifier with implied semantics. "ELD" is an abbreviation for "Element Declaration." "2" is appended to the identifier to differentiate the rule from other rules in the NDR dealing with element declaration.

**Example 2. Using para to provide text of a rule**

```
<blockquote role="Rule" id="rule-ELD2">
  <para>
    <emphasis role="rule-prefix">ELD2</emphasis>...
  </para>
</blockquote>
```

Schematron test cases can be included in the documentation using the DocBook element programlisting. The programlisting element should be included after the para element. The example below shows the complete DocBook markup of an NDR rule that includes a Schematron test case. A CDATA section is used to prevent the Schematron from being parsed during transformation from DocBook to NDRProfile. CDATA sections specify to an XML processor that characters such as angle brackets and ampersands should not be treated as markup. [9]
Example 3. Rule with Schematron test case

```xml
<blockquote role="Rule" id="rule-ELD2">
  <para>[<emphasis role="rule-prefix">ELD2</emphasis>] All element declarations MUST be global</para>
  <programlisting><![CDATA[
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.ascc.net/xml/schematron">
  <title>Navy Naming and Design Rules</title>
  <ns prefix="xsi" uri="http://www.w3.org/2000/10/XMLSchema-instance"/>
  <ns prefix="xsd" uri="http://www.w3.org/2001/XMLSchema"/>
  <ns prefix="html" uri="http://www.w3.org/1999/xhtml"/>
  <pattern id="ELD" name="Element Rule">
    <rule id="ELD1" context="//xsd:all |
                              //xsd:choice |
                              //xsd:complexType |
                              //xsd:sequence">
      <assert test="xsd:element/@ref">ELD2: All element declarations MUST be global</assert>
    </rule>
  </pattern>
</schema>
]]></programlisting>
</blockquote>

Your favorite XSLT processor can be used to create the NDRProfile XML document from the NDR DocBook document. Below is a command line example (with a newline added for readability) using Saxon [10] (an XSLT processor written in Java [11]), of converting an NDR document created with DocBook into an NDRProfile document. For guidance using other XSLT processors, refer to their documentation.

```bash
java -jar saxon.jar -o NDRProfile.xml NDRDocument.xml
NDRDocumentDocbook-to-NDRProfile.xsl
```

Saxon interprets the command as follows:

<table>
<thead>
<tr>
<th>Command Part</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>java -jar saxon.jar</td>
<td>Uses the Java Runtime Environment to invoke the Saxon XSLT processor</td>
</tr>
<tr>
<td>-o NDRProfile.xml</td>
<td>Sends the result tree to an output file called NDRProfile.xml. The name of the file is arbitrary and user controlled</td>
</tr>
<tr>
<td>NDRDocument.xml</td>
<td>The input file - a complete NDR document tagged using DocBook</td>
</tr>
<tr>
<td>NDRDocumentDocbook-to-NDRProfile.xsl</td>
<td>The XSLT stylesheet performing the conversion</td>
</tr>
</tbody>
</table>

The **blockquote** element will be transformed into the following NDRProfile output:

```xml
<Guidance guidanceID="rule-ELD2" documentationID="rule-ELD2"
          prefixNumber="2" prefixToken="ELD">
  <Testability>Fully-Testable</Testability>
  <GuidanceText>All element declarations MUST be global</GuidanceText>
  <TestCases ruleType="Schematron1.5">
```

<TestCases>
  <TestCase>
    <Script><![CDATA[
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.ascc.net/xml/schematron">
<title>Navy Naming and Design Rules</title>
<ns prefix="xsi" uri="http://www.w3.org/2000/10/XMLSchema-instance"/>
<ns prefix="xsd" uri="http://www.w3.org/2001/XMLSchema"/>
<ns prefix="html" uri="http://www.w3.org/1999/xhtml"/>
<pattern id="ELD" name="Element Rule">
  <rule id="ELD1" context="//xsd:all |
               //xsd:choice |
               //xsd:complexType |
               //xsd:sequence">
    <assert test="xsd:element/@ref">
      ELD2: All element declarations MUST be global
    </assert>
  </rule>
</pattern>
]]></Script>
</TestCase>
</TestCases>

3.2. Using an XML Editor

Any XML editor can be used to develop an NDRProfile document. We have developed a cascading stylesheet (CSS) [12] and XSLT stylesheet to format an NDRProfile document for viewing in a browser or editor. These stylesheets should work with any XML editor supporting XSLT or CSS.

3.3. Spreadsheet Transformation

Several standards organizations use spreadsheets to maintain NDR rules. We have created a spreadsheet template in Microsoft's Excel format [13] based on review of the several spreadsheets currently being used. The spreadsheet will allow users to create an NDR using the Excel spreadsheet application. We have also developed an XSLT stylesheet that will transform the NDR spreadsheet to NDRProfile XML. This XSLT can be used to transform a generic NDR to the NDRProfile document instance as long as the fields in the spreadsheet are the same as our fields. If an organization decides to modify the spreadsheet, they can modify the XSLT to support their new structure.

The following table shows the correspondence between spreadsheet columns and NDRProfile schema objects.

<table>
<thead>
<tr>
<th>Column</th>
<th>Title</th>
<th>NDRProfile Mapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Guidance ID</td>
<td>Guidance/@guidanceID</td>
</tr>
<tr>
<td>B</td>
<td>Guidance Name</td>
<td>Guidance/GuidanceName</td>
</tr>
<tr>
<td>C</td>
<td>Derived From</td>
<td>Guidance/DerivedFrom</td>
</tr>
<tr>
<td>D</td>
<td>Enforcement</td>
<td>Guidance/EnforcementLevel</td>
</tr>
<tr>
<td>E</td>
<td>Guidance Text</td>
<td>Guidance/GuidanceText</td>
</tr>
<tr>
<td>F</td>
<td>Testability</td>
<td>Guidance/Testability</td>
</tr>
<tr>
<td>G</td>
<td>Comment</td>
<td>Comment</td>
</tr>
</tbody>
</table>
The spreadsheet must be saved as an XML file prior to conversion. You can save the spreadsheet to an XML file in Excel by:

1. Clicking on File -> Save As.
2. Type the name of the file in the "File name" block.
3. Change the "Save as type" to XML Data.
4. Click Save.

Once the spreadsheet has been saved to an XML file, it is converted to the Excel XML Format. The XSLT stylesheet, Excel-to-NDRProfile.xsl, converts the XML from the Excel format to the NDRProfile XML format. The conversion can be accomplished using an XSLT processor. The following command uses Saxon (newline added for readability):

```
java -jar saxon.jar -o NDRProfile.xml NDRProfile-Template.xml
```

The command is interpreted as follows:

<table>
<thead>
<tr>
<th>Command Part</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>java -jar saxon.jar</td>
<td>Invoke Saxon</td>
</tr>
<tr>
<td>-o NDRProfile.xml</td>
<td>Send the result tree to the output file NDRProfile.xml.</td>
</tr>
<tr>
<td>NDRProfile-Template.xml</td>
<td>The input file - in this case, a spreadsheet exported to XML</td>
</tr>
<tr>
<td>Excel-to-NDRProfile.xsl</td>
<td>The XSLT stylesheet which performs the conversion</td>
</tr>
</tbody>
</table>

### 3.4. XForms

An XForms [14] form has been developed for creating an NDRProfile using a web browser. The form requires a browser such as Mozilla Firefox with client-side support for the XForms standards.

**Note**

To use this form, Mozilla's XForms plugin [15] must be installed. Information about obtaining the plugin can be found at http://www.mozilla.org/projects/xforms. Follow the instructions for installing the plug-in. The form document, NDRProfile-Form.xhtml, uses a template file NDRProfile.xml for the initial population of the fields.

The XForm is designed in 2 pages\(^1\). The first page, shown below, obtains metadata concerning the entire NDR. The second page obtains individual Guidance elements, i.e., NDR rules.

\(^1\)This form is intended as a proof of concept. Therefore, we have not paid much attention to user interface design principles or readability. Also, as of November 2008, the Mozilla XForms extension is still pre-version 1.0. We present screen shots of the form and show the generated XML in order to illustrate the potential of XForms as a low-cost easy-to-use authoring solution.
The second page of the form allows creation of the components for the Guidance portions of the NDR. Page 2 is a repeating group that will allow the user to add as many Guidance elements as needed.
Figure 3. Page 2 (First Screen Shot) Mozilla Form

**Guidance**

- **Guidance Identifier:** rule-FLD2
- **Enforcement Level:** Mandatory
- **Documentation Identifier:**
- **Rule Classification:** General
- **NDR Prefix Token:** ELD
- **NDR Prefix Number:** 1
- **Rationale for Rule:**
- **Comments:**
Figure 4. Page 2 (Second Screen Shot) Mozilla Form

When the user clicks the "Submit" button, an NDRProfile XML document is created in a file called NDRProfile-new.xml. If you want to modify the resulting NDRProfile XML document, copy this file to NDRProfile.xml in the directory where NDRProfile-Form.xhtml file is located.
Below is the XML file generated from the screenshots shown above:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<NDRProfile NDRDocumentBaseURI="http://www.oasis-open.org/committees/download.php/10323/cd-UBL-NDR-1.0Rev1c.pdf"
    NDRProfileBaseURI="http://www.eccnet.com/NDRProfile"
    NDRProfileID="NDR100" URI="" NDRPublishDate="2006-09-08"
    contactURI="mavis.Cournane@cognitrans.com"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="NDRProfile_v-1.xsd">
    <NDRTitle>Universal Business Language (UBL) Naming and Design Rules</NDRTitle>
    <Description>This specification documents the naming and design rules and guidelines for the construction of XML components for the UBL vocabulary.</Description>
    <Status>Draft</Status>
    <ProfileDerivedFrom URI=""/>
    <Guidance derivedFromURI="" documentationID="rule-ELD2" guidanceID="rule-ELD2" prefixNumber="1"
        prefixToken="ELD" ruleLocation="">
        <Classification>General</Classification>
        <DerivedFrom>
            <SourceName/>
            <SourceVersion/>
            <SourceDate>2000-01-01</SourceDate>
        </DerivedFrom>
        <EnforcementLevel>Mandatory</EnforcementLevel>
        <Status>Draft</Status>
        <Testability>Fully-Testable</Testability>
        <Scope/>
        <GuidanceText>All element declarations MUST be global </GuidanceText>
        <Comment/>
        <Justification/>
        <TestCases>
            <TestCase ruleType="Schematron1.5">
                <Script><![CDATA[
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.ascc.net/xml/schematron"
    title="Navy Naming and Design Rules"/>
<ns prefix="xsi" url="http://www.w3.org/2000/10/XMLSchema-instance"/>
<ns prefix="xsd" url="http://www.w3.org/2001/XMLSchema"/>
<ns prefix="html" url="http://www.w3.org/1999/xhtml"/>
<pattern id="ELD" name="Element Rule">
    <rule id="ELD1" context="/xsd:all | 
        //xsd:choice | 
        //xsd:complexType | 
        //xsd:sequence">
        <assert test="/xsd:element/@ref">
            ELD2: All element declarations MUST be global</assert>
        </rule>
    </pattern>]]>
</script>
</TestCase>
</TestCases>
</Guidance>
</ProfileDerivedFrom>
</NDRProfile>
```
4. NDRProfile and QOD

QOD assists standards developers in consistently using XML Schema for the specification of information. Consistent design of XML schemas within an organization or single integration project can reduce the number and the severity of interoperability problems. In addition, this consistency makes the XML schema easier to extend, understand, implement, and maintain; and, it paves the way for automated testing and mapping. Applying best XML schema design practices and following agreed-upon naming conventions are ways to achieve this design consistency.

Users can develop an NDRProfile using QOD. QOD enables a user to create Test Cases and Guidance Rules (called "test requirements" in the QOD web application), associate them with each other, and store them in a central repository. Test cases can then be selected and grouped into an NDRProfile (or "test profile" in QOD terminology). This QOD test profile can then be exported to an XML file conforming to NDRProfile. QOD can import NDRProfile instances as well.

4.1. QOD Handling of Imported Guidance and Test Cases

QOD's import functionality allows a user to upload an NDRProfile written in XML and conforming to the NDRProfile schema. This enables making changes to an NDRProfile offline and importing these changes back into QOD for testing and validation. If a user uploads a new NDRProfile (one which does not currently exist as a test profile in the QOD repository), a new test profile is created. Based on whether the rules or test cases already exist in the QOD database, different options are displayed to the user to determine how these elements are handled. These options are described below.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>The NDRProfile element does not already exist in QOD. If the user clicks &quot;Add,&quot; the element is added to the database, and the user is assigned ownership of it.</td>
</tr>
<tr>
<td>Insert</td>
<td>The NDRProfile element already exists in QOD, but is not associated with the NDRProfile being uploaded. Selecting the insert option establishes an association in QOD between the existing item and the uploaded NDRProfile.</td>
</tr>
<tr>
<td>Update</td>
<td>The NDRProfile element already exists in QOD, is owned by the user, and is already associated with the NDRProfile being uploaded. Selecting this option updates QOD to match the NDRProfile element. If the user is not the owner of the NDRProfile element, the update option is not available.</td>
</tr>
<tr>
<td>Remove</td>
<td>This option appears when both of these conditions occur:</td>
</tr>
<tr>
<td></td>
<td>• The user uploads an NDRProfile whose identifier matches that of an existing QOD test profile (s)he already owns.</td>
</tr>
<tr>
<td></td>
<td>• The QOD test profile is associated with guidance rules or test cases not included in the uploaded NDRProfile.</td>
</tr>
<tr>
<td></td>
<td>The option appears for each item associated with the QOD test profile but not in the uploaded NDRProfile. Clicking &quot;Remove&quot; removes the connection between the guidance rule or test case from the test profile.</td>
</tr>
</tbody>
</table>
**Note**

This will *not* remove the rule or test case completely from the QOD database.

### 4.2. NDRProfile Identifiers

The existence of items in the QOD database is determined by the ID assigned to the item, either as the NDRProfileID, the guidanceID, or the testCaseID. To ensure QOD consistency when creating the NDRProfile outside of the QOD environment, the user is free to assign any ID to the NDRProfileID, while the guidanceID and testCaseID must be a concatenation of the Guidance's organizationPrefix and SourceVersion elements, and the prefixToken and prefixNumber attributes from the NDRProfile. However, when creating test requirements and test cases within the QOD, these IDs are automatically structured and generated based on user input. The composition of the guidanceID and testCaseID are as follows:

**guidanceID**: organizationPrefix-SourceVersion-[prefixToken]-prefixNumber

Where:

- organizationPrefix is the prefix related to the name of the organization that created the NDR
- SourceVersion is the NDR version
- prefixToken (optional) and prefixNumber create the unique identifier for a rule in the NDR.

Example of a guidanceID: OAGI-9.0-R-16

**testCaseID**: guidanceID_implementationNumber

Where:

- ImplementationNumber is a number that should be unique among test cases implementing the same test requirement. For example, if test case A and test case B both implement test requirement C, suggested implementation numbers for test case A and test case B would be 1 and 2 respectively. Note ImplementationNumber is not an element in the NDRProfile.

Example of a testCaseID: OAGI-9.0-R-16-1

### 5. Lessons Learned and the Future of NDRProfile

QOD informed NDRProfile's design, and our implementation experience provided important and beneficial insight toward specifying the schema. On the other hand, the NDRProfile schema may suffer from too much QOD influence in its identifier constructs. For example, the OrganizationPrefix element and QODUniqueIdentifier attribute both assume QOD-specific implementation details that might not be relevant for other NDRProfile use cases. We hope that others will develop XML schema design quality tools supporting NDRProfile. Only through multiple independent implementations can we determine whether the current NDRProfile identifier representations are helpful for other tools besides QOD.

NDRProfile's DerivedFrom element provides a rudimentary way to say that one guidance rule is a modification of another rule. It would be useful if NDRProfile could represent additional information about the modification. For example, two rules may be identical in all respects except that they refer to different
namespaces. Or one rule may be the same as another rule, but with an additional restriction added. We are currently studying NDRs from multiple sources with the ultimate goal of incorporating a taxonomy of reuse scenarios into the NDRProfile schema to better support management of rule sets.

The focus of this document has been on the content of the NDRProfile schema, use of the schema for creating new NDRProfile instances, and NDRProfile's relationship with the QOD application. In Section 1, “Introduction” we enumerated four barriers to NDR effectiveness: proliferation of rule sets, lack of NDR document structure, lack of automation, and poor versioning and traceability. The current NDRProfile schema addresses the first three barriers by specifying an XML language for representing sets of guidance rules and associating guidance rules with computer-interpretable code to determine compliance. Our vision is for NDRProfile to continue to evolve through results from further analysis of NDRs and of inter-rule relationships, and from additional implementation experience.

Acknowledgment

The authors are grateful to Mark Palmer for his thorough and helpful review of an earlier draft of this document.

Bibliography


A. NDRProfile Resources

This appendix lists the stylesheets and templates available for working with the NDRProfile schema. The resources were discussed in Section 3, “Developing an NDRProfile Instance” and are included with the NDRProfile Schema package, available from the QOD Sourceforge website (http://qod.sourceforge.net/).

- **NDRProfile-to-HTML.xsl**: XSLT stylesheet that can be used to either display an NDRProfile instance within a browser supporting XSLT or to convert the NDRProfile instance to HTML.

- **NDRProfile_v-1.css**: CSS stylesheet for formatting an NDRProfile document for viewing in an XML editor or browser.

- **NDRDocumentDocbook-to-NDRProfile.xsl**: XSLT stylesheet to convert an NDR DocBook instance to an instance of NDRProfile. The DocBook instance must use `role` and `id` attributes as explained in Section 3.1.

- **CheckDocbookNDRRules.sch**: Schematron schema for ensuring that NDR documentation marked up using Docbook contains attributes needed for conversion to NDRProfile.

- **NDRProfile-Template.xls**: Excel template for authoring an NDRProfile document.

- **Excel-to-NDRProfile.xsl**: XSLT stylesheet for converting an NDRProfile spreadsheet exported as XML into a document conforming to the NDRProfile schema.

- **NDRProfile-Form.xhtml**: XForms form for creating an NDRProfile instance.

- **NDRProfile.xml**: Template for initial population of the fields of NDRProfile-Form.xhtml.