Metacat - Feature #7098

Add the feature to support the noNamespaceSchemaLocation attribute in xml objects

08/30/2016 01:57 PM - Jing Tao

Status:	Resolved	**Start date:**	08/30/2016
Priority:	Normal	**Due date:**	
Assignee:	Jing Tao	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:	2.7.2	**Spent time:**	0.00 hour
Bugzilla-Id:			

Description

Currently, Metacat caches the schemas with the combination keys of namespaces plus format ids. It set the property of "http://apache.org/xml/properties/schema/external-schemaLocation" to the sax parser for the schema validation. This doesn't work for the noNamespaceSchemaLocation since the schema in the attribute can't have a target namespace:
The noNamespaceSchemaLocation attribute references an XML Schema document that does not have a target namespace.
(please see https://msdn.microsoft.com/en-us/library/ms256139(v=vs.110).aspx)

The parser should be set by the property "http://apache.org/xml/properties/schema/external-noNamespaceSchemaLocation"

History

#1 - 08/30/2016 02:17 PM - Jing Tao

Since there is no target namespaces in the schema, in order to identify the schema, we have to use the object format id. However, the Metacat API it doesn't have the object format id.

Here is the fragment the a fgdc document and its object format id is FGDC-STD-001-1998 in dataone
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<metadata xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://www.fgdc.gov/metadata/fgdc-std-001-1998.xsd">
<idinfo>
.....

The Metacat API has to use value of the noNamespaceSchemaLocation - "http://www.fgdc.gov/metadata/fgdc-std-001-1998.xsd" as the format id.

The xml_catalog table will look like

public_id	format_id	system_id
null	FGDC-STD-001-1998	/schema/fgdc/fgdc-std-001-1998.xsd
null	http://www.fgdc.gov/metadata/fgdc-std-001-1998.xsd	/schema/fgdc/fgdc-std-001-1998.xsd

The same schema has two different format id. The first one is for the DataONE api and the second one is for Metacat API.

#2 - 08/30/2016 02:23 PM - Matt Jones

Jing, for documents coming in from the DataONE API, Metacat should have access to the formatId from the system metadata table. So that would be preferred. In cases when the system metadata is not available, such as when documents come in from the Metacat API, then it seems fine to use that as the formatId, but only as a backup. As we will be deprecating the Metacat API once our new editor is in place, it seems this pathway will be short lived anyways.

#3 - 08/30/2016 06:32 PM - Jing Tao

Jing -- I think at one point (and maybe currently) there was a chunk of code that used the name of the root element of an XML document as the DOCTYPE and DOCNAME, back in the DTD days. My memory is shaky on this, but I think the algorithm first checked if the doc had a namespace (and if so used the schema associated with that), then checked if the document has a DOCTYPE declaration (and used the schema/dtd), and then checked if there was a dtd/schema registered with the name of the root elelment of the document (which, in the case of FGDC, was "metadata"). It was common to get FGDC documents with no DOCTYPE and no namespace, and we used the root element to trigger validation. We can certainly revisit that policy decision, but while you're at it you should carefully check all of the conditional logic to see how Metacat falls through the various options for determining the type of a document and locating the schema. It would be nice to see a little pseudocode explanation of 1) how the logic works now/previously, and 2) how you'd like to see it work now. Something like:

If (namespaceDefinedOnRoot)
... do some stuff, like check if schema is registered for that namespace
else if DOCTYPE defined in PROLOG
... do some other stuff
else if (root element name is registered in xml_catalog)
... yet more stuff
else

... fall through of last resort

The logic should define exactly how we will determine the type of the document, how we will check if an appropriate schema or DTD exists, and how to ignore or handle schemaLocation and noNamespaceSchemaLocation and related attributes. Then we can review that approach to be sure it is right for our different use cases before you implement it.

#4 - 09/01/2016 12:31 PM - Jing Tao

Current workflow:

```
formatId = systemMetadata.getObjectFormatId(); // from Metacat API, this is null.
dtdText  = httpRequest.getParameter("doctext"); // from DataONE API, this is null

if(xml.define("<!DOCTYPE " && (xml.define("publicId" || xml.define("systemId") ) {
//need to valid xml by dtd and use the EntityResolver class to locate the dtd
doctype = publicId; //publicId is a parameter of the resolveEntity(publicId, systemId). They come from the xml object during the processing
if(doctype == null && systemId != null) {
doctype= SAXhandler.getDocname();// root element
}
if(doctype != null ) {
InputStream dtd = lookup_xml_catalog(doctype);
if (dtd == null ) {
if(dtdText != null) {
//dtdText provided
registerDTDInMetacat;
dtd = lookup_xml_catalog(doctype);
}
}
} else {
dtd = openFromURL(systemId);
}

} else  {
//the schema part or the no-schema part
if (rootElement.hasPrefix || (!rootElement.hasPrefix && rootElement.hasAnyNamespace) {
//XMLSchemaService.findNamespace(xml) != null
if(findRegisteredFormatIdInxml_catalog) {
setNamespaceLocationByUsingAllNameSpaceWithFormatId
} else {
setNamespaceLocationByUsingAllNamespaceWithoutFormatId
}
} else {
setNoValidation();
}
}
```

Proposed workflow:

```
formatId = systemMetadata.getObjectFormatId(); // from Metacat API, this is null.

if(xml.define("<!DOCTYPE " ) {
//need to valid xml by dtd and use the EntityResolver class to locate the dtd
if(!xml.define("publicId") || !xml.define("systemId") {
//embedded data
throw an exception
}  else  {
if(xml.define("publicId") {
if(foundInXmlCatalog("publicId") {
useDTD("publicId");
} else {
throw an Exception
}
} else {
if(foundInXmlCatalog(rootElementName) {
useDTD(rootElementName);
}  else {
throw an Exception
}
}
}
} else  {
//the schema part or the no-schema part
if (rootElement.hasPrefix || (!rootElement.hasPrefix && rootElement.hasAnyNamespace) {
//XMLSchemaService.findNamespace(xml) != nul
if(findRegisteredFormatIdWithNamespaceInxml_catalog(formatted)) {
```

setNamespaceLocationByUsingAllNameSpaceWithFormatId
} else {
setNamespaceLocationByUsingAllNamespaceWithoutFormatId
}
//addition support for noNamespaceSchemaLocation
if(rootElement.hasNoNamespaceSchemaLocationAttribute()) {
if(formatId != null) {
if(findRegisteredFormatidWithoutNamespaceInxml_catalog(formatId)) {
setNotNamespaceSchemalocationWithLocalCopy
} else {
throw an exception
}
} else {
if(findRegisteredFormatidWithoutNamespaceInxml_catalog(theAttributeValueOfNoNamespaceSchemaLocation)) {
setNotNamespaceSchemalocationWithLocalCopy
} else {
throw an exception
}
}
}
} else {
setNoValidation();
}
}

#5 - 09/01/2016 12:35 PM - Jing Tao

- File schemaLocation.rtf added

All dents are missing in above comment. I have to upload the file.

#6 - 09/01/2016 12:50 PM - Jing Tao

- File deleted (schemaLocation.rtf)

#7 - 09/01/2016 12:51 PM - Jing Tao

- File schemaLocation.rtf added

#8 - 09/01/2016 02:53 PM - Jing Tao

The workflow we will use:

https://docs.google.com/document/d/1tRb1-S_gKfkuCRvYFamfnB2_pBLSus8NuYZ1fADiQ0U/edit?ts=57c89086

#9 - 09/01/2016 04:10 PM - Jing Tao

In case somebody can't read the google doc. Here is the content:

```
IF Document has DOCTYPE set
     IF DOCTYPE registered locally
    validate(lookup DOCTYPE)
     ELSE
    error()
ELSE IF Document has Namespace set (default or with a prefix definition)
     IF NAMESPACE and FORMATID registered locally
         validate(lookup FORMATID)
     ELSE IF NAMESPACE registered locally
    validate(lookup NAMESPACE)
     ELSE
    error()
ELSE IF Document has noNamespaceSchemaLocation set
     IF FormatID registered locally
         validate(lookup FORMATID)
     ELSE IF noNamespaceSchemaLocation registered locally
    // will probably rarely work, as people will point at local URIs, rather than namespace
    // use noNamespaceSchemaLocation as formatId? Or a new column in xml_catalog
    validate(lookup noNamespaceSchemaLocation)
     ELSE
          error()
ELSE
         checkIfWellFormed()
DONE
```

To be clear: under this proposal, the only source of dtd or xsd documents are the locally registered schemas that have been registered with metacat,

and that typically live locally on the Metacat system. Under no circumstances are the hints in xsi:schemaLocation and xsi:noNamespaceSchemaLocation used to actually go and download a schema. This allows us to use a consistent schema for each namespace, but it also forces us to register any given namespace before it can be used. Until it is registered, any documents that reference a namespace will fail validation with an error. That error should be clear that the namespace needs to be added by the Metacat administrator.

#10 - 09/06/2016 05:08 PM - Jing Tao

- File fgdc.xml added

An example of nonamespaceschema xml document.
The uri can be:

https://www.fgdc.gov/schemas/metadata/fgdc-std-001-1998.xsd
or

https://www.fgdc.gov/metadata/fgdc-std-001-1998.xsd

#11 - 09/06/2016 05:09 PM - Jing Tao

The system metadata is:
<?xml version="1.0" encoding="UTF-8"?>
<d1_v2.0:systemMetadata xmlns:d1_v2.0="http://ns.dataone.org/service/types/v2.0" xmlns:d1="http://ns.dataone.org/service/types/v1">
<serialVersion>0</serialVersion>
<identifier>test-jing-15</identifier>
<formatId>FGDC-STD-001-1998</formatId>
<size>2713</size>
<checksum algorithm="MD5">7b412f5f6f910b66c6fefb25b66ecb9c</checksum>
<submitter>uid=tao,o=NCEAS,dc=ecoinformatics,dc=org</submitter>
<rightsHolder>uid=tao,o=NCEAS,dc=ecoinformatics,dc=org</rightsHolder>
<accessPolicy>
<allow>
<subject>public</subject>
<permission>read</permission>
</allow>
</accessPolicy>
<replicationPolicy replicationAllowed="false"/>
<archived>false</archived>
<dateUploaded>2016-05-13T00:05:48.167+00:00</dateUploaded>
<dateSysMetadataModified>2016-05-13T00:05:48.167+00:00</dateSysMetadataModified>
<originMemberNode>urn:node:mnSandboxUCSB2</originMemberNode>
<authoritativeMemberNode>urn:node:mnSandboxUCSB2</authoritativeMemberNode>
</d1_v2.0:systemMetadata>

#12 - 09/13/2016 02:38 PM - Jing Tao

- Status changed from New to Resolved

Implemented above plan and wrote a junit test class for testing.

Files

schemaLocation.rtf	3.53 KB	09/01/2016	Jing Tao
fgdc.xml	2.65 KB	09/07/2016	Jing Tao