Java™API for XML Processing Maintenance Release 1.5

Description:
Maintenance review of the JAXP 1.4 Specification

Maintenance Lead:
Joe Wang, Oracle Corporation

Feedback:
Please send comments to eg@jaxp.java.net

Rationale for Changes:
JAXP supports XML, XML Schema and XSLT standards that contains constructs that reference external resources such as external DTD, external entity references, and etc. Although there are features in the JDK implementation, the JAXP specification lacks of means for controlling the resolutions of external references.

Proposed changes

1. Add the following properties:

   1) http://javax.xml.XMLConstants/property/accessExternalDTD: restrict access to external DTDs, external Entity References to the protocols specified. The parser should check the protocol of a connection URL against the value of this property before it attempts to make connection to resolve any external DTDs. If the protocol that the connection is attempted to is listed in the value of the property, the connection is allowed. Otherwise, it should be rejected with a runtime exception.

   2) http://javax.xml.XMLConstants/property/accessExternalSchema: restrict access to the protocols specified for external reference set by the schemaLocation attribute, Import and Include element. The schema parser should check the protocol of a connection URL against the value of this property before it attempts to make connection to resolve any external schemas. If the protocol that the connection is attempted to is listed in the value of the property, the connection is allowed. Otherwise, it should be rejected with a runtime exception.

   3) http://javax.xml.XMLConstants/property/accessExternalStylesheet: restrict access to the protocols specified for external reference set by the stylesheet processing instruction, document function, Import and Include element. The parser of the XSL transformer should check the protocol of a connection URL against the value of this property before it attempts to make connection to resolve any external stylesheets. If the protocol that the connection is attempted to is listed in the value of the property, the connection is allowed. Otherwise, it should be

rejected with a runtime exception.

2. Add system properties corresponding to the JAXP properties above to provide users the ability to change the settings without code change.

1) javax.xml.accessExternalDTD: same as accessExternalDTD.

2) javax.xml.accessExternalSchema: same as accessExternalSchema.

3) javax.xml.accessExternalStylesheet: same as accessExternalStylesheet.

3. <Java Home>/lib/jaxp.properties

The above properties can be specified in jaxp.properties to define the behavior for the entire JDK installation. The format is "property-name=[value][,value]", for example:

javax.xml.accessExternalDTD=file,http

The property-names are the same as those of the System Properties that are: javax.xml.accessExternalDTD, javax.xml.accessExternalSchema, and javax.xml.accessExternalStylesheet.

4. Values of the proposed properties

All of the proposed properties above have values in the same format.

Value: a list of protocols separated by comma. A protocol is the scheme portion of an URI, or in the case of the JAR protocol, "jar" plus the scheme portion separated by colon. A scheme is defined as:

scheme = alpha *(alpha | digit | "+" | "-" | ".")
where alpha = a-z and A-Z.

And the JAR protocol:

jar[:scheme]

Protocols are case-insensitive. Any whitespaces as defined by Character.isSpaceChar in the value will be ignored. Examples of protocols are file, http, jar:file.

Default value: the default value is implementation specific and therefore not specified. The following options are provided for consideration:

-- an empty string to deny all access to external references;

-- a specific protocol, such as file, to give permission to only the protocol;
-- the keyword “all” to grant permission to all protocols.

When FEATURE_SECURE_PROCESSING is enabled, it is recommended that implementations restrict external connections by default, though this may cause problems for applications that process XML/XSD/XSL with external references.

Granting all access: the keyword "all" grants permission to all protocols. For example, setting javax.xml.accessExternalDTD=all in jaxp.properties would allow a system to work as before with no restrictions on accessing external DTDs and Entity References.

5. Setting JAXP properties and features

JAXP properties can be set through JAXP factories as follows:

```
DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
dbf.setAttribute(name, value);

SAXParserFactory spf = SAXParserFactory.newInstance();
SAXParser parser = spf.newSAXParser();
parser.setProperty(name, value);

XMLInputFactory xif = XMLInputFactory.newInstance();
xif.setProperty(name, value);

SchemaFactory schemaFactory = SchemaFactory.newInstance(schemaLanguage);
schemaFactory.setProperty(name, value);

Schema schema = schemaFactory.newSchema();
Validator validator = schema.newValidator() ;
validator.setProperty(name, value);

TransformerFactory factory = TransformerFactory.newInstance();
factory.setAttribute(name, value);
```

6. Scope and order

javax.xml.XMLConstants#FEATURE_SECURE_PROCESSING is a required feature for XML processors including DOM, SAX, Schema Validation, XSLT and XPath. It is recommended that implementations associate security related features and properties with the feature. When the secure feature is set to true, it requires that implementations limit XML processing to conform to implementation limits. When it is false, it instructs the implementation to process XML without such restrictions. For the new properties introduced in JAXP 1.5, it is recommended that when the secure feature is set to true, implementations

restrict external connections, and when it is false, allow full access.

Properties specified in the jaxp.properties have effect all invocations of the JDK or JRE, and will override their default values, or those that may have been set by FEATURE_SECURE_PROCESSING.

System properties, when set, will affect one invocation only, and will override the default settings or those set in jaxp.properties, or those that may have been set by FEATURE_SECURE_PROCESSING.

JAXP properties specified through JAXP factories or SAXParser will take preference over system properties, the jaxp.properties file , as well as javax.xml.XMLConstants#FEATURE_SECURE_PROCESSING.

The new JAXP properties will have no effect on the relevant constructs they attempt to restrict in the following situations:

a) When entity resolvers are set on SAX and DOM parsers, XML resolvers on StAX parsers, or URIResolver on a transformer.

b) When a schema is created explicitly by calling SchemaFactory's newSchema method

c) When external resources are not required. For example, the following features/properties are supported by the reference implementation and may be used to instruct the processor to not load the external DTD or resolve external entities.

http://apache.org/xml/features/disallow-doctype-decl	true
http://apache.org/xml/features/nonvalidating/load-external-dtd	false
http://xml.org/sax/features/external-general-entities	false
http://xml.org/sax/features/external-parameter-entities	false

7 Relationship with the Security Manager of the Java platform

a) The JAXP properties will be checked first before a connection is attempted whether or not a Security Manager is present. This means that a connection may be blocked even if it is granted permission by the Security Manager. For example, if the JAXP properties are set to disallow http protocol, they will effectively block any connection attempt even when an application has SocketPermission.

b) For the purpose of restricting connections, Security Manager is in a lower layer. Permissions will be checked down the process after the JAXP properties are evaluated. If an application does not have SocketPermission for example, it will receive a SecurityException even if the JAXP properties are set to allow http connection.

c) When Security Manager is present, the JAXP FEATURE_SECURE_PROCESSING is set to

true. It is recommended that implementations set the values of the new JAXP properties as described in Item 4. Values and Item 6. Scope and order.

8. Error handling

If access to external resources is denied due to restrictions specified by the above access properties, an exception will be thrown in accordance with that specified by the relevant processor as listed below.

a) Exceptions
org.xml.sax.SAXException in the process of parsing an XML file with javax.xml.parsers.SAXParser, javax.xml.parsers.DocumentBuilder, and javax.xml.stream.XMLStreamException with javax.xml.stream.XMLInputFactory.

org.xml.sax.SAXException while creating a javax.xml.validation.Schema through javax.xml.validation.SchemaFactory if the Schema file contains external DTD, or reference to external schema.

org.xml.sax.SAXException while validating an XML file using javax.xml.validation.Validator if the XML file references a Schema through schemaLocation attribute

javax.xml.transform.TransformerConfigurationException while creating new javax.xml.transform.Transformer using javax.xml.transform.TransformerFactory and javax.xml.transform.TransformerException during transformation.

b) Error message format
Implementations may consider error messages in the following format to provide users identifiable hint on why an error has been reported.

When access to external DTD is denied:
External DTD: Failed to read external DTD '<filename>', because '<protocol>' access is not allowed.

When access to external entity is denied:
External Entity: Failed to read external document '<filename>', because '<protocol>' access is not allowed.

When access to external Schema is denied:
schema_reference: Failed to read schema document '<filename>', because '<protocol>' access is not allowed.

When access to external Stylesheet is denied:
Could not read stylesheet target '<filename>', because '<protocol>' access is not allowed.

In all of the above error messages:

<filename> is the name of the external resource without file path;
<protocol> is the protocol denied, such as ‘file’.