Saxon - Bug #4982

Loading xml schemas which are stored inside a zip archive is very slow compared to Xerces

2021-05-04 10:26 - Tomas Vanhala

Status:	In Progress	**Start date:**	2021-05-04
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0:00 hour
Sprint/Milestone:		**Spent time:**	0:00 hour
Legacy ID:		**Fix Committed on Branch:**	
Applies to branch:		**Fixed in Maintenance Release:**	

Description

We have an in-house application which, prior to processing an xml document, validates it against the xml schema. The xml schemas are stored in zip archives, which we obtain "out of band" from associated parties. We make the schemas available to our application by copying the zip archives to an appropriate location (file path).

We have used Xerces for validation, but now we wish to move to Saxon-EE. We have discovered that Saxon is very slow when loading the xml schema files. We are using version 10.3.

I have attached a small demo application that measures the time it takes for the Xerces and Saxon implementations to load a set of xml schema files. (You will need to adjust the paths to the zip file and the license file.)

Apart from Saxon being very slow, we also observe that Saxon calls LSResourceResolver more often than Xerces.

Can the performance of Saxon be improved?

History

#1 - 2021-05-04 10:59 - Michael Kay

Thanks, we'll take a look at this.

From a very quick first glance, my immediate reactions are:

(a) do you really need to set the MULTIPLE_SCHEMA_IMPORTS option? Because this is going to do what it says: read the same schema document multiple times. You should only need it if you have several schema documents with the same target namespace, and that's not really good practice.

(b) there are a number of instances of maxOccurs="99", or "999", or even "9999". The classic algorithm for building a finite state machine with such rules is very expensive (it's exponential in both time and space). Saxon tries to optimise it when it can by using counters, but it's not always possible and I will check to see how these cases are being handled. (Xerces has the same problem, and I think that it sometimes gives up and treats the constraint as if it were maxOccurs="unbounded"). If we do find a problem here, the best solution might be to replace the maxOccurs with an xs:assert.

#2 - 2021-05-04 11:17 - Tomas Vanhala

Thank you for the initial comments.

1. The schema documents have been authored (by an associated party) as follows: Each xsd file which has the filename prefix "NctsDme_FITransit" (13 files) defines an "xml message", and these files share the same target namespace.

Each one of the mentioned 13 files includes and imports the same set of xsd files.

We need to set MULTIPLE_SCHEMA_IMPORTS because due to the schema design, the same xsd files are imported multiple times.

1. About maxOccurs: We are not able to influence the maxOccurs values. The main reason we wish to move to Saxon is because of this optimisation you mention.

#3 - 2021-05-12 13:33 - Michael Kay

I've been trying to get this to run without success. I don't think I understand the strategy for URI resolution. Should all schema documents be found within the ZIP file, or is the external directory also relevant (perhaps it's just a copy of what's in the ZIP file?)

I'm pretty sure the fact that the files are in a ZIP archive isn't relevant to the problem, and just complicates the repro.

Is there a single root schema document that includes/imports all the others? It seems to start by supplying a long list of independent schema documents.

#4 - 2021-05-15 10:48 - Michael Kay

- Status changed from New to AwaitingInfo

#5 - 2021-06-07 16:44 - Tomas Vanhala

The demo application which we created tries to load the schema documents from within the ZIP file.

I assume that for Saxon-EE the fact that the files reside in a ZIP archive is not relevant. However, the ZIP archive could make loading a file more "expensive". This could in part explain the significantly increased loading time compared to Xerces.

The demo we provided earlier had some dependencies to our internal code. Please find an uploaded ZIP archive containing refined demo code with all dependencies included. To invoke it:

javac -cp ".:./saxon-ee-10.3.jar:./rt.jar:" SaxonBugDemo.java java -cp ".:./saxon-ee-10.3.jar:./rt.jar:./xercesImpl-2.12.1.jar" SaxonBugDemo

You need to modify the path to the license file before compiling.

About the content of the ZIP archive containing the xsd files:

There are indeed a number of root schema documents. Inside the ZIP archive, they are in the directory external/ncts/dme/v1_6. The filenames of the root schema documents start with "NctsDme_FI". An example of such a root schema document is:

NctsDme_FITransitDeclaration.xsd

It contains the following:

```
<xs:include schemaLocation="NctsDme_QualifiedType.xsd"/>
<xs:import namespace="http://tulli.fi/schema/external/common/dme/v1_2/qdt" schemaLocation="../../../common/dme
/v1_2/Dme_QualifiedType.xsd"/>
<xs:import namespace="http://tulli.fi/schema/external/common/dme/v1_2/cdt" schemaLocation="../../../common/dme
/v1_2/Dme_CodeListType.xsd"/>
<xs:import namespace="http://tulli.fi/schema/external/common/dme/v1_2/udt" schemaLocation="../../../common/dme
/v1_2/Dme_UnqualifiedType.xsd"/>
```

In this sample ZIP archive, in total there are 13 root schema documents, each one of them does the same includes/imports.

As I noted earlier, the number of times Saxon calls LSResourceResolver (due to includes/imports) is much greater than Xerces,

#6 - 2021-06-07 16:47 - Tomas Vanhala

- File SaxonBugDemo.zip added

#7 - 2021-06-07 17:13 - Michael Kay

- Status changed from AwaitingInfo to In Progress

Files

SaxonBugDemo.java	8.69 KB	2021-05-04	Tomas Vanhala
SchemaValidatorImplTest_zipped_schemas.zip	31.9 KB	2021-05-04	Tomas Vanhala
SaxonBugDemo.zip	29.8 MB	2021-06-07	Tomas Vanhala