Saxon-JS - Bug #5001

saxon-js xsl:output method "xml" with indent "yes" eats whitespace

2021-05-25 21:49 - Jamie Peabody

Status:	New	**Start date:**	2021-05-25
Priority:	Low	**Due date:**	
Assignee:	Michael Kay	**% Done:**	0%
Category:		**Estimated time:**	0:00 hour
Sprint/Milestone:		**Spent time:**	0:00 hour
Applies to JS Branch:		**SEF Generated with:**	
Fix Committed on JS Branch:		**Platforms:**	
Fixed in JS Release:			

Description

Using saxon-js 2.2.0, and given XSLT:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="2.0"
                                exclude-result-prefixes="xs"
                                xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
                                xmlns:xs="http://www.w3.org/2001/XMLSchema">
        <xsl:output method="xml" indent="yes"/>
        <xsl:template match="/">
                <login>
                        <xsl:value-of select="/data/text()"/>
                </login>
        </xsl:template>
</xsl:stylesheet>
```

And XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<data>   </data>
```

saxon-js produces:

```
<?xml version="1.0" encoding="UTF-8"?>
<login/>
```

If indent="no", then it outputs:

```
<?xml version="1.0" encoding="UTF-8"?>
<login>   </login>
```

It is unexpected that changing indentation would cause whitespace to be consumed and for elements to collapse.

History

#1 - 2021-05-25 23:14 - Michael Kay

The spec is here:

https://www.w3.org/TR/xslt-xquery-serialization-31/#xml-indent

Quote:

the serializer MAY output whitespace characters in addition to the whitespace characters in the instance of the data model. It MAY also elide from the output whitespace characters that occurred in the instance of the data model or replace such whitespace characters with other whitespace characters.

I've re-read the rules and the behaviour here is entirely consistent with the rules in the spec. Whether it's ideal is another matter. But if the whitespace is significant and you don't want it messed with, try using suppress-indentation="login".

The use of the term "elide" in the spec is a little quirky; it is used without formal definition. Dictionary definitions include to omit, delete, abridge, or ignore: I read it simply as "delete".

#2 - 2021-05-26 00:04 - Jamie Peabody

lol, y, that part of the spec is riddled with *elide*. Anyway, I think the issue is that there is no DTD, so no way of knowing any types. I think this is relevant:

Whitespace characters SHOULD NOT be added, elided or replaced in places where the characters would constitute significant whitespace, for example, in the immediate content of an element that is annotated with a type other than xs:untyped or xs:anyType, and whose content model is known to be mixed.

I think the content *is* significant whitespace (the element is not annotated). Elsewhere, Oracle says https://www.oracle.com/technical-resources/articles/wang-whitespace.html#:~:text=What%20is%20XML%20Whitespace%3F,content%20and%20should%20be%20preserved.

Usually without DTD or XML schema definition, all whitespaces are significant whitespaces and should be preserved.

Also, we use saxon (Java) elsewhere in my company, and I believe the behavior is different in this instance. I have not confirmed, but it was reported by a customer that it is different. Thus this issue.

#3 - 2021-05-26 00:12 - Michael Kay

Clearly when the serialization spec talks of "significant whitespace" it does NOT regard whitespace text nodes in untyped documents as significant, otherwise indentation would not be allowed to do anything at all in the common case of untyped documents.

But although I think this output is conformant, we'll look at whether the algorithm can be tweaked.

#4 - 2021-05-26 00:40 - Michael Kay

- Project changed from Saxon to Saxon-JS

- Category deleted (XSLT 3.0 packages)