Saxon - Bug #3934
Streamability analysis problem SXST0060
2018-10-02 14:09 - Michael Kay

<table>
<thead>
<tr>
<th>Status:</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority:</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee:</td>
<td>Michael Kay</td>
</tr>
<tr>
<td>Category:</td>
<td>Streaming</td>
</tr>
<tr>
<td>Sprint/Milestone:</td>
<td></td>
</tr>
<tr>
<td>Legacy ID:</td>
<td></td>
</tr>
<tr>
<td>Applies to branch:</td>
<td>9.9</td>
</tr>
</tbody>
</table>

| Start date: | 2018-10-02 |
| Due date:   |             |
| % Done:     | 100%        |
| Estimated time: | 0:00 hour   |
| Spent time: | 0:00 hour   |
| Fix Committed on Branch: | 9.9 |
| Fixed in Maintenance Release: | 9.9.0.2 |

Description
Reported by Martin Honnen, test case saved in XSLT3 as si-fork-119.

Saxon 9.9 reports error

Error in xsl:param/@select on line 13 column 79 of si-fork-119.xsl:
   SXST0060: Component cannot be streamed, although it is guaranteed streamable according to W3C rules. No streamable path found in expression ($current-item, ...)

Saxon 9.8 reports:

Static error at xsl:template on line 12 column 71 of si-fork-119.xsl:
   XTSE3430: Template rule is declared streamable but it does not satisfy the streamability rules.
   * Operand {.} of {let $vv:LHS2027961269 := ...} selects streamed nodes in a context that allows arbitrary navigation (line 13)

History
#1 - 2018-10-02 14:13 - Michael Kay
- Description updated

#2 - 2018-10-02 15:49 - Michael Kay
The main issue here seems to be that we don't have any streamability analysis code for xsl:message, and it is falling back to the default. (But I'm not sure why this doesn't give a streamability failure).

A secondary issue is that we aren't being smart about inferring the context item type for the match pattern match="[. instance of ....]". There's justification for this, because the W3C spec doesn't handle it specially. But we could do better.

#3 - 2018-10-02 17:27 - Michael Kay
The streamability rules in 1.b.iii.A.I say

If U is absorption and the intersection of T with U{element(), document-node()} is U[] (that is, if T is a type that does not allow nodes with children), then U is inspection.

But (Streamability line 169) we are actually testing isChildlessNodeKind(), which returns false for items such as maps and atomic values.

This has the effect that even when we compute the context item type for this template rule as map("."), the reference to "." in xsl:message, which has usage absorption, is considered to be a "potentially consuming operand", leading to the select expression of the xsl:message instruction being classified striding and motionless. This in turn leads to the xsl:message being classified as grounded and consuming. Which in this case isn't fatal, but
if there were more than one xsl:message instruction like this then we would get a "multiple consuming operands" streamability failure.

I've changed the test from isChildlessNodeKind() to !t.getUType().overlaps(UType.DOCUMENT.union(UType.ELEMENT)) and unfortunately this doesn't solve the problem. The arguments to xsl:message include both "." and various variables whose type can include element nodes, so the test that the value can't be an element or document node isn't satisfied anyway.

I can't help feeling there's a rule missing here somewhere: if the match pattern on a template rule doesn't match nodes, then the template rule must surely be grounded and motionless, regardless of its contents.

#4 - 2018-10-02 18:16 - Michael Kay
- Fix Committed on Branch 9.9 added

The two changes reported above cause no test regression so I have committed them (for 9.9 only), even though they do not solve this test case:

(a) align the general streamability rules implementation with the spec
(b) infer the context item type from a match pattern such as match=".[ instance of X]"

#5 - 2018-10-08 17:48 - Michael Kay
- Status changed from New to In Progress

I have made a couple of changes to enable this test case to pass:

- When the item type of a match pattern does not allow nodes, assess the template body with a context item posture of GROUNDED. This rule isn't in the spec (perhaps because the rules in the spec for assessing the item type of match patterns aren't actually good enough) -- but it seems to make sense; it ensures that references to "." within the body of the template will be treated as grounded.

- When the context item type does not allow nodes, assess the streamability of the context item expression "." as grounded and motionless.

Adding these rules causes no test regression, though it does arguably mean that we are no longer applying the W3C streamability rules as strictly as we should.

There are other issues raised by the test that should probably be handled with new tests and new issues, notably the streamability and streaming of xsl:message and xsl;assert.

#6 - 2018-10-08 18:05 - Michael Kay
- Status changed from In Progress to Resolved

Fixed on the 9.9 branch.

Not making these changes for 9.8, in the interests of stability.
Bug fix applied to the Saxon 9.9.0.2 maintenance release.