

Saxon - Bug #2160

Attempting to set more local variables (3) than the stackframe can accommodate (1)

2014-10-04 11:16 - Michael Kay

Status:	Closed	Start date:	2014-10-04
Priority:	Normal	Due date:	
Assignee:	Michael Kay	% Done:	100%
Category:	Internals	Estimated time:	0:00 hour
Sprint/Milestone:	9.6.0.2	Spent time:	0:00 hour
Legacy ID:		Fix Committed on	9.6
Applies to branch:	9.6	Branch:	
		Fixed in Maintenance	9.6.0.2
		Release:	
Description			
Bug reported by Vladimir Nestorovsky here: https://saxonica.plan.io/boards/2/topics/6006			

History

#1 - 2014-10-04 20:42 - Vladimir Nesterovsky

This reproduces the problem:

```
<xsl:stylesheet version="2.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:t="this">
```

```
<xsl:template match="/">
  <xsl:variable name="s" as="element()*">
    <h>
      <f/>
    </h>
  </xsl:variable>
```

```
<xsl:apply-templates mode="t:a" select="$s"/>
</xsl:template>
```

```
<xsl:template mode="t:a" match="h">
  <xsl:sequence select="f/t:f2(@a, @b, @c)"/>
</xsl:template>
```

```
<xsl:template mode="t:a" priority="2" match="h">
  <xsl:choose>
    <xsl:when test="t:f1(*) = 'b'">
      <s/>
    </xsl:when>
    <xsl:otherwise>
      <xsl:next-match/>
    </xsl:otherwise>
  </xsl:choose>
</xsl:template>
```

```
<xsl:function name="t:f1" as="item()*">
  <xsl:param name="e" as="element()?"/>
```

```
<xsl:sequence select="'d'"/>
</xsl:function>
```

```
<xsl:function name="t:f2" as="item()*">
  <xsl:param name="a" as="xs:string?"/>
  <xsl:param name="b" as="xs:string?"/>
  <xsl:param name="c" as="xs:string?"/>
```

```
<xsl:sequence select="'d'"/>
```

```
</xsl:function>
</xsl:stylesheet>
```

With some modifications like this:

```
<xsl:stylesheet version="2.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:t="this">

  <xsl:template match="/">
    <xsl:variable name="s" as="element()*">
      <h>
        <f/>
      </h>
    </xsl:variable>

    <xsl:apply-templates mode="t:a" select="$s"/>
  </xsl:template>

  <xsl:template mode="t:a" match="h">
    <xsl:sequence select="f/t:f2(@a, @b, @c)"/>
  </xsl:template>

  <xsl:template mode="t:a" priority="2" match="h">
    <xsl:next-match/>
  </xsl:template>

  <xsl:function name="t:f1" as="item()*">
    <xsl:param name="e" as="element()?"/>

    <xsl:sequence select="'d'"/>
  </xsl:function>

  <xsl:function name="t:f2" as="item()*">
    <xsl:param name="a" as="xs:string?"/>
    <xsl:param name="b" as="xs:string?"/>
    <xsl:param name="c" as="xs:string?"/>

    <xsl:sequence select="'d'"/>
  </xsl:function>

</xsl:stylesheet>
```

I get different error:

```
[java] Caused by: java.lang.IndexOutOfBoundsException: Index: 0, Size: 0
[java]   at java.util.ArrayList.rangeCheck(ArrayList.java:635)
[java]   at java.util.ArrayList.get(ArrayList.java:411)
[java]   at net.sf.saxon.expr.XPathContextMajor.getTargetComponent(XPathContextMajor.java:778)
[java]   at net.sf.saxon.expr.XPathContextMinor.getTargetComponent(XPathContextMinor.java:554)
[java]   at net.sf.saxon.expr.UserFunctionCall.callFunction(UserFunctionCall.java:507)
[java]   at net.sf.saxon.expr.UserFunctionCall.iterate(UserFunctionCall.java:473)
[java]   at net.sf.saxon.expr.SlashExpression.map(SlashExpression.java:852)
[java]   at net.sf.saxon.expr.ContextMappingIterator.next(ContextMappingIterator.java:61)
[java]   at net.sf.saxon.tree.iter.HomogeneityCheckerIterator.next(HomogeneityCheckerIterator.java:65)
[java]   at net.sf.saxon.expr.Expression.process(Expression.java:555)
[java]   at net.sf.saxon.expr.instruct.Template.applyLeavingTail(Template.java:338)
[java]   ... 8 more
```

#2 - 2014-10-04 23:53 - Michael Kay

- *Category set to Internals*

Thanks. Reproduced the problem (with your first example). Occurs with Saxon-HE only. It's a failure in component binding: the call on `f2(@a, @b, @c)` is attempting to call `f1` and is failing because `f1`'s stackframe doesn't have room for the three arguments. I think that the problem is with the `next-match`; the current component in the dynamic context is wrong, so the wrong binding vector is being used.

#3 - 2014-10-31 11:25 - Michael Kay

Sorry for the delay on this, caused by the need to use a diagnostic build with HE-only code.

Now resolved; the problem occurs with a tail-call on `xsl:next-match` because it is not saving the "current component" correctly; the current component is needed for the new dynamic binding mechanism in 9.6 to work. This has been introduced to support calls across separately compiled packages.

The reason the failure doesn't occur on Saxon-EE is that the function calls are all inlined, so no dynamic binding occurs. (Function inlining is possible because the functions are implicitly private to the package, so they cannot be overridden from a different package. This should also mean that the calls can be statically bound even when not inlined. This little optimization has not been implemented in 9.6.

I've committed a patch (9.6 and 9.7 branches) and before marking as resolved I will now look at the "different error" that occurs with the modified stylesheet.

#4 - 2014-10-31 11:37 - Michael Kay

- *Status changed from New to Resolved*

I have confirmed that the "different error" is in fact a different symptom of the same problem; it is solved by the same patch. Therefore closing as resolved. We hope that a maintenance release including the patch will be out shortly!!

#5 - 2014-11-13 19:25 - O'Neil Delpratt

- *Status changed from Resolved to Closed*

- *% Done changed from 0 to 100*

- *Fixed in version set to 9.6.0.2*

Bug fix applied to the maintenance release Saxon 9.6.0.2

#6 - 2015-12-11 13:54 - O'Neil Delpratt

- *Sprint/Milestone set to 9.6.0.2*

- *Applies to branch 9.6 added*

- *Fix Committed on Branch 9.6 added*

- *Fixed in Maintenance Release 9.6.0.2 added*