

## Saxon/C - Support #4465

### How can we automatically resolve xi:include?

2020-02-25 12:56 - Rein Baarsma

<b>Status:</b>	In Progress	<b>Start date:</b>	2020-02-25
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	O'Neil Delpratt	<b>% Done:</b>	0%
<b>Category:</b>	PHP API	<b>Estimated time:</b>	0:00 hour
<b>Sprint/Milestone:</b>		<b>Spent time:</b>	0:00 hour
<b>Found in version:</b>			

**Description**

I've read that saxon has a xi:on feature to resolve xi:include tags, but I cannot find any way to enable this using the PHP extension. Can you point me in the right direction?

The only information I can find is here: <https://www.saxonica.com/documentation9.5/sourcedocs/XInclude.html>

I don't even know if the saxon php extension uses Xerces?

There seems to be no information whatsoever on xinclude or xi:on in [https://www.saxonica.com/saxon-c/documentation/index.html#!api/saxon\\_c\\_php\\_api](https://www.saxonica.com/saxon-c/documentation/index.html#!api/saxon_c_php_api)

Is automatically resolving xi:include tags possible in saxon php? And if so, how can I do it? (php example code)

### History

#### #1 - 2020-02-25 13:20 - O'Neil Delpratt

You should be able to apply the configuration feature using `setConfigurationProperty(string $name, string $value)` function on the [SaxonProcessor](#) class. The configuration feature `http://saxon.sf.net/feature/recognize-uri-query-parameters` allows you set the `xinclude` parameter. Documentation on this feature can be found [Configuration Features](here <https://www.saxonica.com/documentation/index.html#!configuration/config-features>).

I have not tested if this actually works, but I will investigate it further.

#### #2 - 2020-02-25 15:42 - Rein Baarsma

Thanks for the quick reply :)

Maybe I'm trying something that's supposed to work differently, but I'm expecting my `xinclude` to get the dummy <https://www.w3schools.com/xml/note.xml> and my xslt to have this result: `{'title': 'Reminder'}`

I do get this result if I manually replace `xi:include` part by the contents of the `note.xml`

No errors.. just get the `[]` result, which indicates that the `part` was not included correctly.

Below is my code:

```
$saxonProcessor = new \Saxon\SaxonProcessor
$saxonProcessor->setConfigurationProperty('RECOGNIZE_URI_QUERY_PARAMETERS', 'true');
// it accepts only strings, so assuming "true"
$transformer = $this->saxonProcessor->newXsltProcessor();

// I'm not sure if the transformFileToString will actually use the query part here, but at least there's no error.. but it also doesn't seem to work.
$output = $transformer->transformFileToString($tmpFilePath, $xsltFilePath.'?xinclude=yes');
```

This is my XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<gcuF version="2.0">
```

```
<xi:include xmlns:xi="https://www.w3.org/2001/XInclude/" xmlns:cts="
http://www.crius-group.com/transformationsservice/1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
href="https://www.w3schools.com/xml/note.xml"/>
</gcuf>
```

## This is my XSLT

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:j="
http://www.w3.org/2005/xpath-functions" xmlns:functx="http://www.functx.com" xmlns:lom="
http://www.imglobal.org/xsd/imsmd_v1p2" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:thip="
http://thip.thiememeulenhoff.nl" version="3.0">
```

```
<xsl:output media-type="application/json" method="text" />
```

```
<xsl:template match="gcuf">
  <xsl:variable name="result">
    <xsl:element name="j:array">
      <xsl:apply-templates select="note" />
    </xsl:element>
  </xsl:variable>
  <xsl:copy-of select="xml-to-json($result, map{'indent':false()})"/>
</xsl:template>
```

```
<xsl:template match="note">
  <xsl:element name="j:map">
    <xsl:element name="j:string">
      <xsl:attribute name="key">title</xsl:attribute>
      <xsl:value-of select="./heading/string()"/>
    </xsl:element>
  </xsl:element>
</xsl:template>
```

```
<xsl:template match="@*|node() ">
  <xsl:copy>
    <xsl:apply-templates select="node()|@*" />
  </xsl:copy>
</xsl:template>
```

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

### #3 - 2020-02-25 16:10 - O'Neil Delpratt

Thanks for sending me your code snippet. What you have will not work. For the configuration property you need to use the full URL:

```
...
$saxonProcessor->setConfigurationProperty('http://saxon.sf.net/feature/recognize-uri-query-parameters', 'true'
);
$output = $transformer->transformFileToString(null, $xsltFilePath);
...
```

To read the XML document use the doc function in the stylesheet and then concat the xinclude parameter as a part of the URI:

```
doc('xml_filename?xinclude=yes')
```

You can also pass in the XML filename as a parameter to the stylesheet.

### #4 - 2020-02-26 11:21 - Rein Baarsma

Okay, it took some time to figure out how to use the doc() function, but I have this now:

```
$saxonProcessor = new \Saxon\SaxonProcessor;
$saxonProcessor->setConfigurationProperty('http://saxon.sf.net/feature/recognize-uri-query-parameters'
, 'true');
$transformer = $saxonProcessor->newXsltProcessor();

$xsltFilePath = \dirname(__DIR__, 4).' /src/Content/Xslt/metadata_gcuf2.xsl';
$xmlPath = \dirname(__DIR__, 3).' /Resources/test-xinclude.xml';

$tmpFilePath = tempnam(sys_get_temp_dir(), 'pub-auto-test');
$xml = <<<XML
<?xml version="1.0" encoding="utf-8" ?>
<xml filename="$xmlPath?xinclude=yes" />
XML;
file_put_contents($tmpFilePath, $xml);

$output = $transformer->transformFileToString($tmpFilePath, $xsltFilePath);
```

In the XSLT I've added this at as the first element:

```
<xsl:template match="xml">
  <xsl:apply-templates select="document(./@filename)" />
</xsl:template>
```

Unfortunately the output is still "[ ]".

If I remove the json matching part and simply try this:

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="3.0">
  <xsl:template match="xml">
    <xsl:apply-templates select="document(./@filename)" />
  </xsl:template>

  <xsl:template match="@*|node()">
    <xsl:copy>
      <xsl:apply-templates select="node()|@*" />
    </xsl:copy>
  </xsl:template>
</xsl:stylesheet>
```

Then the output is:

```
<?xml version="1.0" encoding="UTF-8"?><gcuf version="2.0">\n
  <xi:include xmlns:xi="https://www.w3.org/2001/XInclude/" xmlns:cts="
http://www.crius-group.com/transformationsservice/1.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
href="https://www.w3schools.com/xml/note.xml"/>\n
</gcuf>
```

So it seems everything works except for resolving the xinclude..

On the other hand I may try to simply use the same mechanism to include the xi:include links with doc(). I guess that works too, although not sure if the xi:include would be faster or better somehow.

#### #5 - 2020-02-26 11:42 - Rein Baarsma

I just noticed I used document() instead of doc(). With document() I don't seem to be able to load external resources (over http), so I cannot use this method to replace xinclude functionality.

I've also tried to change my code to use doc() instead, but simply changing document() to doc() and changing the filename to <https://www.w3schools.com/xml/note.xml> leads to an empty result.. the note.xml does not seem to be included. I'm not sure how to use the doc()

function. I cannot find any example, except for this <https://www.saxonica.com/html/documentation/sourcedocs/streaming/streaming-templates.html>, but it says specifically that this is for "EE" license, which we don't have.

Just as an aside, we're using version 1.1.2, because of issue <https://saxonica.plan.io/issues/4371>

#### #6 - 2020-02-26 12:43 - O'Neil Delpratt

Hi,

I noticed an error in the XInclude namespace. It should be:

```
http://www.w3.org/2001/XInclude
```

#### #7 - 2020-02-26 13:01 - Michael Kay

You say:

- I'm not sure how to use the doc() function. I cannot find any example\*

What documentation are you using? You seem to be struggling with the basics. There are plenty of resources available:

- The W3C spec at <https://www.w3.org/TR/xpath-functions-31/#func-doc> (agreed, this goes into a lot of metaphysics about the exact relationship between a URI and a document)
- There's Saxon-specific documentation at <http://www.saxonica.com/documentation/index.html#functions/fn/doc>
- There's 3 pages on the function in my book *XSLT 2.0 and XPath 2.0 Programmer's Reference*, and no doubt good coverage in other XSLT 2.0 books as well

Note that W3Schools is hopeless on XPath 2.0 - the information is very incomplete. The site does a good job on some other specs, e.g. CSS, but not on this one.

O'Neil pointed out a basic error in your code, that is the XInclude namespace. I'd suggest you simplify what you're doing to try to reduce other possible sources of error. (I don't do PHP, so I don't fully understand your code, but one possible problem is that you are using filenames where URIs are expected; that could lead to incorrect processing of relative URIs. (Handling of base URI is one area where doc() and document() differ)

Some suggested steps:

(a) check that you can do a simple transformation like this (test.xsl)

```
<xsl:transform xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="3.0">
<xsl:template name="main">
<x><xsl:copy-of select="https://www.w3schools.com/xml/note.xml"/></x>
</xsl:template>
</xsl:transform>
```

(run this without a source document, specifying "main" as the initial template name)

(b) place a document test.xml in the same directory as the stylesheet, where test.xml is:

```
<?xml version="1.0" encoding="UTF-8"?><z><xi:include xmlns:xi="http://www.w3.org/2001/XInclude" href="https://www.w3schools.com/xml/note.xml"/></z>
```

and execute the stylesheet

```
<xsl:transform xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="3.0">
<xsl:template name="main">
<x><xsl:copy-of select="doc('test.xml?xinclude=yes')"/></x>
```

```
</xsl:template>
</xsl:transform>
```

You might find it useful to check that you can run this from the (Java) command line before trying it from PHP code. This is all intended to reduce the number of "moving parts" and therefore the number of things that you can get wrong.

If this works you can start adding things, like constructing the URI as a variable rather than a string literal.

## #8 - 2020-02-26 15:40 - Rein Baarsma

Hi Michael,

Thanks for your reply.

I've simplified the example. However, in the PHP extension, it is impossible to not give an XML input. (just a stylesheet)

The xample O'Neil gave earlier does not work: `$output = $transformer->transformFileToString(null, $xsltFilePath);`

It gives an error that the first argument should be a string. And if I give an empty string ("), it fails with another error.

That aside, this is my example code:

```
$tmpFilePath = tempnam(sys_get_temp_dir(), 'pub-auto-test');
$xml = '<?xml version="1.0" encoding="utf-8" ?><root />';
file_put_contents($tmpFilePath, $xml);

$saxonProcessor = new \Saxon\SaxonProcessor;
$saxonProcessor->setConfigurationProperty('http://saxon.sf.net/feature/recognize-uri-query-parameters'
, 'true');
$transformer = $saxonProcessor->newXsltProcessor();
$output = $transformer->transformFileToString($tmpFilePath, __DIR__.'/test.xsl');
```

### test.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<z>
  <xi:include xmlns:xi="http://www.w3.org/2001/XInclude" href="https://www.w3schools.com/xml/note.xml"/>
</z>
```

### test.xsl

```
<xsl:transform xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="3.0">
  <xsl:template name="main" match="root">
    <x><xsl:copy-of select="doc('test.xml?xinclude=yes')"/></x>
  </xsl:template>
</xsl:transform>
```

(Note, I had to add `match="root"` to get it working with a dummy input XML)

So what happens?

- If I remove the `"?xinclude=yes"` part after test.xml, it works and the output is the same as test.xml (except it's wrapped in `root`)

- If I run it as described above I get this error:

Execution aborted after 1 second

- If I remove the line in php with the setConfigurationProperty, it is the same as removing the xinclude=yes. The \$output is then the full test.xml surrounded by root
- If I change the xi:include href to test2.xml and put something in that file, it works and shows me test.xml with xi:include replaced with the test2.xml

My conclusion is that the xinclude is working, but does not work with HTTP. This is unfortunate as my use-case is to include resources from Amazon S3 with signed http urls through xi:include.

If you have any idea how to fix this error "Execution aborted after 1 second" for http xi:includes, that would be great :)

#### #9 - 2020-02-26 15:42 - Rein Baarsma

I should have added that I started with trying

```
<x><xsl:copy-of select="doc('https://www.w3schools.com/xml/note.xml')"/></x>
```

But it gives exactly th same error:

Execution aborted after 1 second

#### #10 - 2020-02-26 15:48 - Michael Kay

The PHP interface provides

```
Xslt30Processor.callTemplateReturningString(string $stylesheetFileName, string $templateName)
```

to start a transformation at a named template with no source document.

You say:

*My conclusion is that the xinclude is working, but does not work with HTTP*

Indeed you seem to have narrowed down the problem to something we can attempt to reproduce. Thanks for that.

You say:

*My conclusion is that the xinclude is working, but does not work with HTTP*

Indeed you seem to have narrowed down the problem to something we can attempt to reproduce. Thanks for that.

The fact that your simple call on doc() also fails when given an HTTP URI suggests to me that the problem has nothing to do with XInclude, but is more a question of whether the Java VM and/or the underlying platform are handling HTTP requests at all. We seem to be reducing the number of moving parts...

**#11 - 2020-02-26 16:00 - O'Neil Delpratt**

Rein Baarsma wrote:

The xample O'Neil gave earlier does not work: `$output = $transformer->transformFileToString(null, $xsltFilePath);`  
It gives an error that the first argument should be a string. And if I give an empty string (""), it fails with another error.

If using the XsltProcessor then do something similar:

```
$xsltProc->setSourceFromFile($source_filename); $xsltProc->compileFromFile($stylesheet_filename);  
$result = $xsltProc->transformToString();
```

As Mike suggested you can use the new Xslt30Processor with a call to the function `callTemplateReturningString`.

I am doing some more investigation with you simplified code to see if we can reproduce your issue.

**#12 - 2020-02-26 17:26 - Rein Baarsma**

If using the XsltProcessor then do something similar:

```
$xsltProc->setSourceFromFile($source_filename);
```

Again if I try to `setSourceFromFile(null)`, it gives the same error (that the argument should be a string). If I give an empty string (just tried again with this method), it says:

Error on line 1 column 1 of application: SXXP0003: Error reported by XML parser: Content is not allowed in prolog.

As Mike suggested you can use the new Xslt30Processor with a call to the function `callTemplateReturningString`.

As stated earlier, we are using version 1.1.2, because of issue <https://saxonica.plan.io/issues/4371>. Therefore I cannot (yet) use the new Xslt30Processor. I will try as soon as the other issue is fixed and we can trust the new version in production.

I am doing some more investigation with you simplified code to see if we can reproduce your issue.

Best of luck! Let me know if I can help further.

### #13 - 2020-02-27 01:24 - O'Neil Delpratt

Rein Baarsma wrote:

If using the XsltProcessor then do something similar:

```
$xsltProc->setSourceFromFile($source_filename);
```

Again if I try to `setSourceFromFile(null)`, it gives the same error (that the argument should be a string). If I give an empty string (just tried again with this method), it says:

I was thinking of passing the source document `test.xml` from comment #8

```
$xsltProc->setSourceFromFile("test.xml");
```

### #14 - 2020-02-27 11:27 - O'Neil Delpratt

Rein Baarsma wrote:

I am doing some more investigation with your simplified code to see if we can reproduce your issue.

Best of luck! Let me know if I can help further.

Hi Rein,

I managed to run your example code using the 1.1.2 API and the XInclude works for me.

Please see my code snippet below. Although this is in C++ the same logic would work in PHP. I have read in the XSLT stylesheet as a string but it can be read in as a file (i.e. `trans->compileFromFile(test.xml)`):

```
SaxonProcessor * processor2 = new SaxonProcessor(true);
processor2->setConfigurationProperty("http://saxon.sf.net/feature/recognize-uri-query-parameters", "true");

XsltProcessor * trans = processor2->newXsltProcessor();
trans->compileFromString("
<xsl:transform xmlns:xsl='http://www.w3.org/1999/XSL/Transform' version='3.0'><xsl:template name='main' match=
'root'><x><xsl:copy-of select='\doc('test.xml?xinclude=yes')\"/></x></xsl:template></xsl:transform>");
trans->setSourceFromFile("test.xml");
trans->setProperty("it", "main");
const char *result = trans->transformToString();
printf("XInclude test result = %s", result);
```

## Output:

```
XInclude test result = <?xml version="1.0" encoding="UTF-8"?><x><z>
  <note xml:base="https://www.w3schools.com/xml/note.xml">
    <to>Tove</to>
    <from>Jani</from>
    <heading>Reminder</heading>
    <body>Don't forget me this weekend!</body>
  </note>
</z></x>
```

### #15 - 2020-02-27 13:00 - O'Neil Delpratt

Hi,

The next step is to investigate why your access to the document using the doc function is not working for http.

Is it possible for you to try using the curl command from the command-line to access the file:

```
curl https://www.w3schools.com/xml/note.xml
```

### #16 - 2020-02-28 12:46 - O'Neil Delpratt

Further notes:

As a quick test you can check if you are able to access the document with the following code which is in C++ but you can write the same code in PHP:

```
XPathProcessor * xpath = processor->newXPathProcessor();
XdmItem * result = xpath->evaluateSingle("doc('https://www.w3schools.com/xml/note.xml')");
printf("%s", result->toString());
```

### #17 - 2020-02-28 12:46 - O'Neil Delpratt

- Status changed from New to AwaitingInfo

### #18 - 2020-09-03 13:16 - O'Neil Delpratt

This bug looks related to this issue: [#4664](#)

We will be revisiting this issue shortly.

**#19 - 2020-09-03 13:17 - O'Neil Delpratt**

*- Status changed from AwaitingInfo to In Progress*