

Saxon/C - Bug #4859

Jet Runtime Has detected Unrecoverable Error-libsaxonhec.dll transformToString not found

2020-12-17 15:07 - Brian Higashide

Status:	AwaitingInfo	Start date:	2020-12-17
Priority:	High	Due date:	
Assignee:	O'Neil Delpratt	% Done:	0%
Category:	Python	Estimated time:	0:00 hour
Sprint/Milestone:		Spent time:	0:00 hour
Found in version:	1.2.1	Fixed in version:	
Description			
When trying to run the following code, i am getting this error:			
Error: C:\Program Files\Saxonica\SaxonHEC1.2.1\libsaxonhec.dlltransformToString not found			
12072020			
JET RUNTIME HAS DETECTED UNRECOVERABLE ERROR: system exception at 0x00007ffed7d0a68			
<pre>print(fcfilename) proc=saxonc.PySaxonProcessor(license=False) print('Test Saxon/C on Python') print(proc.version) xslt30_processor = proc.new_xslt30_processor() xslt30_processor.set_cwd(".") transform=xslt30_processor.transform_to_string(source_file=" S:/QA/Projects2/SBE/schematron/NJFC_0814_22000040437.xml", stylesheet_file=' S:/QA/Projects2/SBE/schematron/AccountTransfer-runtime-compiled.xsl', output_file=' S:/QA/Projects2/SBE/schematron/testresults.xml')</pre>			
<pre> msg=xslt30_processor.get_xsl_messages() print(fcfilename) printed=[] if msg is not None: i=0 while i <msg.size: if str(msg.item_at(i)) not in printed: print('\t'+msg.item_at(i)) printed.append(str(msg.item_at(i)))</pre>			
Related issues:			
Is duplicate of Saxon/C - Support #4847: sotransformToString - Python		In Progress	2020-12-02

History

#1 - 2020-12-17 15:59 - Michael Kay

- Project changed from Saxon to Saxon/C
- Assignee set to O'Neil Delpratt

#2 - 2020-12-18 11:12 - O'Neil Delpratt

- Is duplicate of Support #4847: sotransformToString - Python added

#3 - 2020-12-18 11:34 - O'Neil Delpratt

- Category set to Python
- Status changed from New to Duplicate
- Found in version set to 1.2.1

Hi,

Thanks for reporting this issue. This looks like the same bug issue as [#4847](#). In comment [#2](#) we describe the issue and solution. In summary the `get_xsl_messages` function has incorrect JNI code in the underlying C++ code. A fix is available for the next release of Saxon/C

As a workaround there is a patched file `Xslt30Processor.cpp` attached in the same bug issue [#4847](#) please download it, replace the file you have and rebuild the Saxon/C product for python.

#4 - 2020-12-18 15:40 - Brian Higashide

- File `clipboard-202012180939-zcvqr.png` added

```
1239 XdmValue * Xslt30Processor::getXslMessages(){
1240
1241     jmethodID mID = (jmethodID) SaxonProcessor::sxn_environt->env->GetMethodID(cppClass,
1242                                     "getXslMessages",
1243                                     "()[Lnet/sf/saxon/s9api/XdmNode;");
1244     if (!mID) {
1245         stderr << "Error: "<<getDlName() << ".getXslMessages" << " not found in JNI\n"
1246         << stderr;
1247     }
```

The fix from [#4847](#) has been applied and the module rebuilt but I am still seeing the same error message.

#5 - 2020-12-18 17:41 - O'Neil Delpratt

Investigating.

#6 - 2020-12-19 00:29 - O'Neil Delpratt

- Status changed from Duplicate to In Progress

#7 - 2020-12-19 01:02 - O'Neil Delpratt

- Status changed from In Progress to AwaitingInfo

Hi,

you are missing a set property for `xsl:message`:

```
xslt30_processor.set_property('m', '')
```

The bug issue [#4847](#) also mentions:

What actually happens to the `xsl:messages` by default is to output messages to the standard output console, but this does not find its way to python. As mentioned in the bug issue [#4147](#) to capture the messages you will have to call `xslt30_processor.set_property('m', '')` before you do the transformation.

The culprit of the exception thrown is the size property. For some reason we create a `PyXdmValue` object over an empty sequence. We probably should return `None` for when `get_xsl_message` has no `xsl:message` to return.

#8 - 2020-12-19 07:04 - Brian Higashide

- File `jet_err_14200.txt` added

- File `jet_dump_14200.dmp` added

I have updated the code as follows, but still receiving the same error

```
for filename in os.listdir(path):
    print(filename)
    proc=saxonc.PySaxonProcessor(license=False)
    print('Test Saxon/C on Python')
    print(proc.version)
    xslt30_processor = proc.new_xslt30_processor()
    xslt30_processor.set_property('m', '')
    xslt30_processor.set_cwd(".")
    transform=xslt30_processor.transform_to_string(source_file="
S:/QA/Projects2/SBE/schematron/NJFC_0814_22000040437.xml", stylesheet_file='
S:/QA/Projects2/SBE/schematron/AccountTransfer-runtime-compiled.xsl', output_file='
S:/QA/Projects2/SBE/schematron/testresults.xml')

# dink=xslt30_processor.transform_to_string(source_file=path+'/'+filename, stylesheet_file='S:/QA/Projects2/S
BE/schematron/AccountTransfer-runtime-compiled.xsl', output_file='S:/QA/Projects2/SBE/schematron/testresults.xml
l')

    msg=xslt30_processor.get_xsl_messages()
    print(filename)
    printed=[]
```

```
if msg is not None:
    i=0
    while i <msg.size:
        if str(msg.item_at(i)) not in printed:
            print('\t'+msg.item_at(i))
            printed.append(str(msg.item_at(i)))
```

#9 - 2020-12-19 19:17 - O'Neil Delpratt

- Status changed from *AwaitingInfo* to *In Progress*

Please can you send us your stylesheet and XML document either on the bug issue or private email.

#10 - 2021-01-04 00:42 - Brian Higashide

Just wanted to update, this is working fine on my home computer but I'm only having issues on my office pc so it might be difficult to duplicate.

I will try reinstalling it on Monday just in case.

#11 - 2021-01-05 18:42 - O'Neil Delpratt

Thanks for sending me your files privately.

On my linux machine the stylesheet displays the xsl:messages as expected.

On your office pc please double check that you have applied the fixes described in comment #3.

Please can you describe your environment such as platform that you are using. I assume you are using a Windows machine.

#12 - 2021-01-13 17:18 - Brian Higashide

Sure, I am running windows 10 64bit on my office pc.

I have the same setup at home (windows 10, 64 bit) which is running fine after the fix but not on my office machine for some reason.

#13 - 2021-01-14 16:46 - O'Neil Delpratt

It looks like we have a cython caching problem when changes are made to the C++ files:

<https://stackoverflow.com/questions/55836761/cython-c-cache-how-to-clear>

#14 - 2021-01-14 18:19 - O'Neil Delpratt

- Status changed from *In Progress* to *AwaitingInfo*

So the link I posted does not have a solution, but what is clear is that the python extension build, which uses cython is not picking up the modified Xslt30Processor.cpp file. I am not experiencing this on my windows machine, but I know this issue can happen as I once experienced it in the past on my mac platform.

There are a number options here /

1. try to deleting the object file for Xslt30Processor in the directory `../Saxon.C.APIpython-saxon/build/temp.win-amd64-3.7/Xslt30Processor.obj` (directory may be slightly different if not on the same python release) and then rebuild the python extension
2. If 1. fails try reinstalling Saxon/C and apply the patch before you rebuild the python extension

Also being on the latest stable cython release could make the difference. Please can you confirm which cython version you have on your office machine?

In Powershell on my machine if I run `cython --version` I get:

```
Cython version 0.29.14
```

#15 - 2021-01-22 20:19 - Brian Higashide

Hi O'Neil. my cython version is 0.29.21

removing the Xslt30Processor.obj and rebuilding the extension looks to have fixed the problem. Now my code is running without errors.

Thanks for the help!

Files

jet_err_5828.txt	44.9 KB	2020-12-17	Brian Higashide
jet_dump_5828.dmp	169 MB	2020-12-17	Brian Higashide
clipboard-202012180939-zcvqr.png	25.1 KB	2020-12-18	Brian Higashide
jet_err_14200.txt	39.3 KB	2020-12-19	Brian Higashide
jet_dump_14200.dmp	169 MB	2020-12-19	Brian Higashide