Saxon/C - Bug #4845

PySchemaValidator.validation_report always None, and clears all errors

2020-11-29 18:32 - Craeg Strong

Status:	AwaitingInfo	**Start date:**	2020-11-29
Priority:	Normal	**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

It appears that accessing the validation report clears errors and always returns None

Given that schema_filepath points to a valid XSD, and test_filepath points to an XML with exactly one validation error against that XSD:

```
saxon_proc = saxonc.PySaxonProcessor(license=True)
validator = saxon_proc.new_schema_validator()
assert validator is not None
validator.register_schema(xsd_file=str(schema_filepath))
# print(f'validating {test_filepath}  with {schema_filepath}')
validator.validate(file_name=str(test_filepath))
print(v.exception_count())
print(v.exception_occurred())
print(v.validation_report)
print(v.exception_count())
print(v.exception_occurred())
```

Returns this:

```
1
True
None
0
False
```

History

#1 - 2020-11-29 20:13 - Michael Kay

- Project changed from Saxon to Saxon/C

- Category changed from Schema conformance to Python

- Assignee set to O'Neil Delpratt

#2 - 2020-11-30 02:55 - O'Neil Delpratt

- Description updated

#3 - 2020-11-30 03:17 - O'Neil Delpratt

- Status changed from New to AwaitingInfo

- Found in version set to 1.2.1

Hi,

The validation_report returning None probably means for some reason it has not found any validation errors, or fails before gets to that stage. Given that there is some exception thrown means that maybe the schema file has failed to compile or not been found. Or maybe the issue is with the source document not being found. Due to how JNI works the exceptions will be cleared once you call validation_report.

To better investigate what is happening please can you include the function get_error_message(0) to your code before calling validation_report so we can actually see the exception message,

For example:

```
print(v.get_error_message(0))
```

Another thing to check is that you are in the working directory you expect. You can always set the cwd on the SaxonProcessor object:

```
saxonproc2.set_cwd('/fullpath')
```

Replace the word /fullpath with you directory on your system.