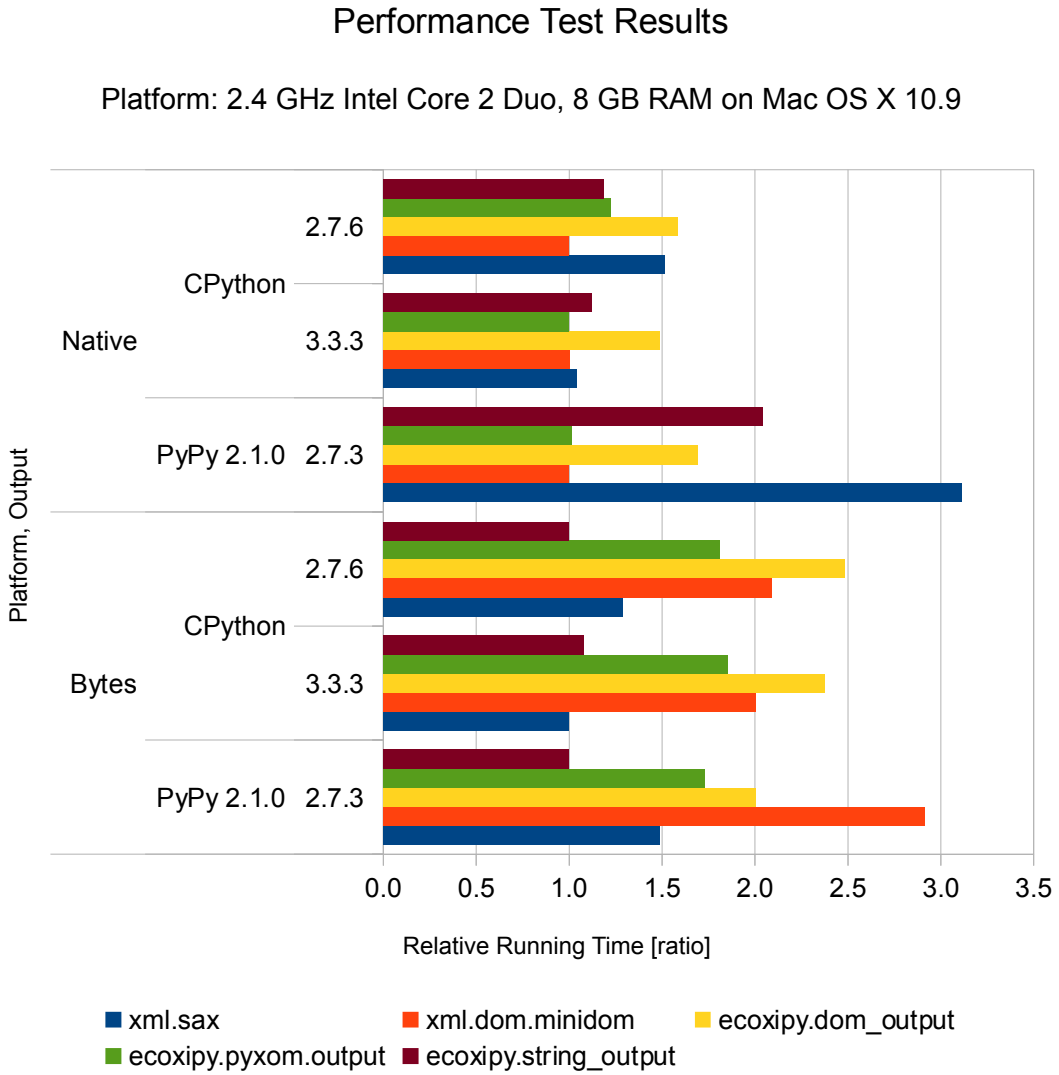
ECoXiPy Peformance Test Results - Relative Results

Setup: The same XHTML5 document is created with different APIs. All output implementations of EcoXiPy are tested as well as “xml.sax” and “xml.dom.minidom”. For each of the APIs one test creates its native representation and one test transforms this into an UTF-8 encoded byte string, as mostly XML is serialized this way. The SAX tests creates byte strings in both test types.
Platform: 2.4 GHz Intel Core 2 Duo, 8 GB RAM on Mac OS X 10.9

The following table shows the relative running times as ratios for the tests for the different Platforms and output types.

Output	***Bytes***	***Bytes***	***Bytes***	***Native***	***Native***	***Native***
Python Platform	PyPy 2.1.0	CPython	CPython	PyPy 2.1.0	CPython	CPython
Python Version	2.7.3	3.3.3	2.7.6	2.7.3	3.3.3	2.7.6
xml.sax	1.5	1.0	1.3	3.1	1.0	1.5
xml.dom.minidom	2.9	2.0	2.1	1.0	1.0	1.0
ecoxipy.dom_output	2.0	2.4	2.5	1.7	1.5	1.6
ecoxipy.pyxom.output	1.7	1.9	1.8	1.0	1.0	1.2
ecoxipy.string_output	1.0	1.1	1.0	2.0	1.1	1.2

ECoXiPy Peformance Test Results - Relative Graph



ECoXiPy Peformance Test Results - Absolute Results

The following table shows the running times in seconds for the tests for the different Platforms and output types.

Output	***Bytes***	***Bytes***	***Bytes***	***Native***	***Native***	***Native***
Python Platform	PyPy 2.1.0	CPython	CPython	PyPy 2.1.0	CPython	CPython
Python Version	2.7.3	3.3.3	2.7.6	2.7.3	3.3.3	2.7.6
xml.sax	11.1	25.1	35.1	11.1	25.0	34.9
xml.dom.minidom	21.9	50.2	57.0	3.6	24.0	23.0
ecoxipy.dom_output	15.0	59.6	67.7	6.1	35.7	36.5
ecoxipy.pyxom.output	13.0	46.5	49.3	3.6	23.9	28.1
ecoxipy.string_output	7.5	27.1	27.3	7.3	26.8	27.3

ECoXiPy Peformance Test Results - Absolute Graph