

VelocityReport Plugin - Defect #1302

fromJSON() fails on UTF-8 files with BOM

02/16/2018 07:36 AM - Bernd Korthaus

Status:	Rejected	Start date:	02/16/2018
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	velocity	Estimated time:	0.00 hour
Target version:			
Browser (if web client):			
Description			
<p>We created an interface where we get a JSON file from SAP SuccessFactors EmployeeCentral when employee data got changed there.</p> <p>The first lines of such a JSON look like this:</p> <pre>{ "S136" : { "e_all_in_contract": null, "e_cost_center": "10 3800", ...</pre> <p>We read the JSON into an object variable with</p> <pre>oEC_EIM_persons = plugins.VelocityReport.fromJSON(plugins.file.readTXTFile(jsFile.getAbsolutePath()));</pre> <p>Everything works great when the JSON is ANSI coded. However when I switch the JSON file to UTF-8 coded (e.g. with Notepad++), Servoy prompts this error:</p> <p>JAVASCRIPT ERROR net.stuff.plugin.velocityreport.org.json.JSONException: Expected a ':' after a key at 3 [character 4 line 1]</p> <p>After some investigation I found the reason here: [[https://de.wikipedia.org/wiki/UTF-8#Byte_Order_Mark]]</p> <p>There are UTF-8 files with and without a BOM (Byte Order Mark) at the beginning. That BOM is the sequence EF BB BF. When saving as "UTF-8 without BOM" (can be done easily with Notepad++ in menu 'Coding / Convert to UTF-8 without BOM'), everything works well.</p> <p>So to be able to process also "UTF-8 with BOM", VelocityReport.fromJSON() could check if the BOM is there, and then ignore or strip it before processing.</p>			

History

#1 - 02/16/2018 04:41 PM - Patrick Talbot

- Status changed from New to Rejected

This is a known issue and actually not coming from Velocity but from the way Java File IO is choking on UTF-8 + with BOM files.
Most java program reading text files as UTF-8 have the same issue.

Problem is that the Exception thrown internally in the JVM is trapped and not bubbled up to Velocity's classes so there's no way for Velocity to know in advance and the file is then read incomplete and JSON cannot convert it.

There would be a convoluted way to check out the BOM, by first opening the file as binary, checking the first bytes, etc.
But this is an unnecessary overhead that I have decided against.

#2 - 02/16/2018 04:57 PM - Bernd Korthaus

I agree, that is really too costly.