

Docmosis Torando v2.0 Release Notes

Oct 2015

Docmosis core updated from version 3.3.0 to 4.0.1.

Tornado Changes

#	Change
1	Added the ability to generate XML sample data based on the template (new button on the GUI)
2	Test renders are now done in DEV mode from the web console. This means errors will be injected into the document to help with locating problems. The render REST API supports the choice of dev/prod mode.
3	There is a new REST API endpoint <code>"/convert"</code> which can be used to convert (not populate) files between formats including spreadsheets and drawings.
4	GUI Updates for the web console including: <ul style="list-style-type: none">• Text wrapping• Display 200 templates in browser tree• New scroll bars• XML dummy data button

Web Service API Changes

#	Change
1	render endpoint allows <code>storeto</code> to be empty as well as not specified
2	web services - updated render end point to set header <code>X-Docmosis-PagesRendered</code> when successful.
3	web services - updated <code>getTemplateStructure</code> to correctly detect <code>TemplateNotFoundException</code> and return as a 400 failure with the correct message
4	web services - updated render to have new parameter <code>"ignoreUnknownParams"</code> to allow systems to pass unreasonable params. This is because one customer has a system they call to issue the request to Docmosis. It adds it's own params and the customer can't avoid that.
5	web services - added PING endpoint (<code>"/ping"</code>)
6	web services - allowed <code>"+"</code> character in output file names
7	New endpoint: <code>convert</code> for converting documents without requiring data or templates. This endpoint is considered experimental. Contact Docmosis support if you would like access to this service.
8	Image data can reference images by fetching from URLs.
9	Updated render service where response message was being constructed in JSON rather than XML or choosing between XML and JSON correctly.

New Core Engine Features

#	Change
1	<h3>If / Else / Else-If Support in Conditional Sections</h3> <p>Simple “else” looks like this:</p> <pre><<cs_true>> true <<else>> false <<es_>></pre> <p>“else-if” is written like this:</p> <pre><<cs_isPerson>> I have a person <<else_isPlant>> I have a plant <<else>> I have something I didn't expect <<es_isPerson>></pre> <p>Conditional Sections Support the new Expression Syntax (described below)</p> <p>Eg:</p> <pre><<cs_{val < 10.0}>> Low value = <<val>> <<else_{val > 100.0}>> High value = <<val>> <<else>> Nominal value = <<val>> <<es_>></pre>
2	<h3>New Expression Engine</h3> <p>A new expression engine has been added that improves computational capabilities of templates. Operators and functions can be applied to string and numeric data. The following lines summarize the way this affects the templates.</p> <p>Expressions in Docmosis templates are still delimited by the braces (“{” and “}”) characters. Expressions can now include:</p> <ul style="list-style-type: none">• Precedence using the brackets “(” and “)” characters• Mathematical expressions• Mathematical and String functions• Boolean logic
3	<h3>Direct Expression Evaluation</h3> <p>Fields which are expressions can be used to insert data into documents. For example:</p> <pre><<{1 + 2 + 3}>> <<{round(val/100,2)}>> <<{titleCase(firstName + ' ' + lastName)}>></pre>

#	Change
4	Assignment of Expressions to Variables Variables can now be assigned the results of expressions: << \$m={expr} >> Eg. <<\${m={round(1+ceil(2*3.5))}}>> round(1+ceil(2*3.5)) = <<\${m}>>
5	Boolean Logic <<cs_{val1 (val2 && val3)}>> val1 is true or both val2 and val3 are true <<es_>>
6	Maths functions Typical math functions are supported. Eg. <<{max(4.522, 4.5)}>> The round() function has been extended to support an optional precision: <<{round(1.236)}>> <<{round(1.236, 2)}>> The precision also pads: <<{round(1.2, 5)}>>
7	String Functions charAt, compareTo, compareToIgnoreCase, concat, endsWith, equals, equalsIgnoreCase, indexOf, lastIndexOf, length, replace (character replacement), startsWith, substring, toLowerCase, toUpperCase, Trim, map, titleCase, split, eg: <<{equals('a', 'b')}>> <<{indexOf('abc', 'b')}>> <<{startsWith('this is', 'this')}>> <<{titleCase('joe blogs')}>>
8	Formatting Functions numFormat and dateFormat functions have been created to perform numeric and date formatting functions. These functions are based on the similarly named FieldRenderers that already existed in Docomsis. numFormat(<value>, <format>[, <locale>]) <<{numFormat(value1, '###,###.00')}>> dateFormat(<value>[, <output format>[, <input format>]]) <<{dateFormat(value1, 'dd/MM/yy', 'dd-MMM-yyyy')}>>
9	Operators The well known operators are supported: () + - * / % + - = == != < <= > >= && !
10	XML population has been updated to allow "<r>" to result in paragraph insertion (in addition to "<r\n>" and "<n>"). This helps XML data processing where the xml contains this type of new lines.

#	Change
11	Logging is by default quieter now with more logging information moved to DEBUG/FINE level.
12	Hyperlink processing has been updated to be able to use variables and variables set from expressions. The hyperlinks are now expected to be of the format <<link:xxx>> but the <<link_ xxx>> format is still supported.
13	Sub-templates can now set template-variables that are visible to the “master” template and subsequently processed templates.

Bug Fixes / Technical Changes

#	Change
1	Fixed issue where no data being given by client results in an error
2	New license key format
3	Fixed bug with JSON image dummy data provision which didn't work for ODT templates
4	JSON dummy data structure processor improved to handle case where image and text field have same key.
5	Updated Tomcat to 7.0.65 (from 7.0.27) which fixes runtime JSP compilation errors under Java 8. This fixes the problem when an administration password was set in the console causing console errors.
6	Fixed issue where spaces around data input via web console would stop data being recognized as JSON format.
7	Improved hyperlink insertion to deal with being given blank data. This was corrupting DOC output.
8	Updated field parsing to handle more cases where set-variable fields occurring in redundant paragraphs (blank lines) was causing parsing errors.
9	Improved injection of errors into the document (in “dev” mode): <ul style="list-style-type: none"> - Errors not being visible in the document - stopping template errors highlighting in red
10	Improved JSON data provision for anonymous arrays (arrays that have a list of values without keys). Internally, the new key "" is used for items that are anonymous which fits current data provision perfectly - and is semantically a good fit.
11	Improved population of <<rs_ \$this>> and <<rs_ \$current>> to work with JSON anonymous arrays of arrays.
12	Updated analysis to detect when overlapping sections are caused by an over-zealous image bookmark and raise a better error message. This occurs when the bookmark captures more than just the image so other content could be accidentally removed from the result.
13	Updated field processing in numbered and bullet lists. Removes the default assumption that a bullet list with a field in it is intended for repetition and makes it much more "normal".
14	Increased the default window size for xml processing for templates with large/complex paragraphs particularly in docx format.
15	Fixed issue where hidden ("_GoBack") bookmarks created by new versions of word (Mac) were causing blank paragraphs to be left behind.
16	Updated error handling for when a start or end tag is in a list (but not the matching tag) - causing

#	Change
	the related other tag to be lost/remote
17	Fixed issue where looping over including sub-templates more than 2 levels deep would incorrectly determine a cycle in template referencing and raise an error.
18	Updated number renderer to be able to take a second parameter which identifies the locale to use when parsing/formatting the date.