

RELEASE HISTORY

8.1.8613 Minor Release

The new features and improvements include:

- Improved support for the creation of accessible documents, including support for PDF/UA as well as generally improved PDF tagging, conforming to the Matterhorn Protocol. (#6082)
- Support for the creation of PDF/X-4p documents. (#6164)
- Text in SVGs is now only converted to vectors when necessary, reducing the average size of created PDF documents. (#6091)
- Rasterization of SVG is now required in fewer cases and produces higher quality output.
- Improved logging and handling of CSS errors, leading to significantly less skipping of correct CSS. (#6153, #6229, #6234)
- Improved logging of internal exceptions, which are now mentioned on log level "warning" with detailed information on log level "finer". (#5452)
- Improved width computation for images with percentage widths inside tables, floats, positioned elements and inline-blocks with auto widths. (#6334, #6394)
- Added option to specify a custom XMP packet for the output PDF. (#5996)
- The log can now be retrieved from the wrapper APIs after the conversion failed. (#6226)
- Improved painting order, now respecting the (otherwise unsupported) property "opacity". (#6214)
- The CSS property "text-align-last" can now be used without the -ro- vendor prefix.
- Improved resource loading, including support for HTTP redirects and more kinds of data URIs. (#6196)
- Added support for "Node.contains()" in JavaScript. (#6110)
- Improved error handling for barcodes.
- HTTP requests for resources (like images or style sheets) now time out if the server does not respond for 60 seconds. Time out can be configured via the new API property "resourceRequestTimeout". (#6345)
- Conversions running on the web service can now be canceled. (#6170)
- The PDFreactor web service is now also available as a self-contained WAR file. (#6031)
- Added Node.js web service wrapper API. (#6302)
- The wrappers are now under the MIT license.
- Updated the Jetty web server used by PDFreactor Web Service to version 9.3.9.
- Implemented hexadecimal notation for rgba colors (four or eight digits).

- The CSS property "box-decoration-break" now supports slicing background images of block elements. (#6348)
- It is now possible to retrieve the PDF directly as a Base64 encoded string using the REST API.

The changes include:

- The Java web service wrapper API is now based on REST. There are the following API changes:
 - All classes are now in the package "com.realobjects.pdfreactor.webservice.client"
 - When manually specifying a web service URL it must now end in "REST" instead of "SOAP".
 - Connections to the web service now have a configurable timeout. The default is 10 minutes.
 - The only explicitly thrown exception type is now "PDFReactorWebserviceException".

Otherwise the API is backwards compatible.

- The .NET web service wrapper API is now based on REST. There are the following API changes:
 - All classes are now in the namespace "RealObjects.PDFReactor.Webservice.Client".
 - When manually specifying a web service URL it must now end in "REST" instead of "SOAP".
 - The default timeout for connections to the web service is now 10 minutes.
 - All method and property names are now capitalized.

Otherwise the API is backwards compatible. It requires the Microsoft .NET Framework 4.0 and is not COM visible.

- Linux control groups are now respected when determining CPU count.
- Font directories specified via the PDFReactor API are now scanned recursively, including their subdirectories.
- The XML schema definition of some SOAP types had been revised to make SOAP messages sleeker. Integrators that use their own SOAP clients should update them.
- Excessively nested "table" elements (starting from the eighth) now have a default style of "table-layout: fixed".
- Removed the bin/pdfreactorwebservice.vmoptions file. If any custom Java parameters (such as "-Xmx") were added to this file by the integrator in a prior PDFReactor version, these parameters should now be set in jetty/start.ini instead.

The bug fixes include:

- Added support for more combining characters, including technical and Arabic ones. (#6207)
- Some bidi control characters were visible in resulting PDFs. (#6208)

- Improved Unicode support in CSS, including a fix for multi-byte characters (e.g. em-quad) in combination with leaders. (#6188)
- Interactive PDF forms and embed elements did not respect transformations and "-ro-scale-content". (#6295, #6355)
- Table footers were laid out multiple times per page for tables with no header. (#6305)
- Dynamically inserted script elements as well as their load events were not executed in the expected order.
- The result of "XMLHttpRequest.getAllResponseHeaders()" contained brackets around its items.
- Backgrounds could be hidden by fully transparent borders for certain combinations of borders applied to the same element. (#6417)
- Relative ICC profile URLs were not resolved. (#6316)
- "Textarea.value" did not work as expected. (#6272)
- The page "size" CSS property treated the singular value "landscape" as "A4" instead of "A4 landscape". (#6330)
- The style "display: inline-block" could not be applied to generated content (was treated as "inline"). (#6177)
- Block images could not be placed on named pages. (#5477)
- The style "text-overflow: ellipsis" was not applied in table cells and caused exceptions in some scenarios. (#6182, #6243)
- When omitted the second parameter of the CSS transform functions "translate" and "skew" did not default to 0. (#6262)
- The width of boxes containing overflowing text was computed incorrectly. (#6223)
- Setting enableDebugMode to false did not disable debug mode. (#6239)
- Font caching could result in unnecessary exceptions being shown. (#6231)
- JavaScript wrapper API did not handle connection timeouts. (#6230)
- Named conversions caused different exception types to be thrown. (#6220)
- Canvas elements could share the same context leading to unexpected results. (#6218)
- HTTP request headers set via the API were not used to load resources referenced via embed or object elements. (#6209)
- Font embedding could not be disabled. (#6057, #6203)
- JavaScript errors in iFrames could cause PDF conversions to abort. (#6124)
- SVGs containing multiple gradient color stops with the same offset could not be converted. (#6133)
- SVG gradient always had their "stop-opacity" values rounded to either fully opaque or fully transparent. (#6159)
- Order of footnotes on a page could be wrong in rare cases.
- Some footnotes were placed too far to the right (#6397)

- Floated or positioned elements inside inline-block elements were laid out incorrectly. (#6260, #6087)
- PDF merging can now handle more uncommon PDF files and has improved error handling, including a specific "PdfMergeException". (#6073, #6088)
- Created PDF/A documents containing certain fonts caused some validators to return "The width for character ... does not match". (#6369)
- pdfreactorcore.jar contained some third party libraries. (#6163)
- Blocks in inline elements were moved after their inline siblings. (#6360)
- Static pages of the PDFreactor Web Service (such as the PDFreactor RIO demo) did not load properly when request logging was enabled. (#6366)
- Hexadecimal CSS color values with an invalid number of digits resulted in black color instead of being ignored. (#6383)

8.0.8234.1 Hotfix Release

The bug fixes include:

- Large table row spans were not applied correctly (#6178)
- Proprietary layout information in JavaScript returned box descriptions in wrong order (#6169)
- The value of the "text-overflow" style was applied even when "overflow" was "visible" (#6161)
- PHP wrapper was not compatible with PHP 7 (#6138)

8.0.8234 Maintenance Release

The bug fixes include:

- The CSS function "local" in @font-face rules did not have any effect (#6084)
- CSS transforms using the "matrix" function missed the translate values (#6129)
- CSS transforms using the "scale" function were inaccurate for negative values (#6118)
- The style "text-transform: uppercase" was not applied to form elements (#6072)
- "unicode-range" styles broke parsing of style sheets (#6147)
- Footnotes were reset per page (#6076)
- HTML5 element "source" (inside "video" or "audio") was not supported (#4997)
- jQuery 2.2.0 was not supported due to missing method "createHTMLDocument" (#6106)
- The node list returned by the JavaScript function "querySelectorAll" was live (#6075)
- In rare cases conversions would not terminate due to layout issues inside tables (#6142)
- Eliminated a memory leak in PDFreactor Web Service (#6152)

- Relative URLs missing only the protocol where not resolved (#6146)
- When text was copied from created PDFs, the font name exported to the clipboard was incorrect for complex font names (#6085)
- PDF/X conformance checks requiring CMYK reported errors on spot colors even when legal colors were specified as alternatives (#6125)
- The PDFReactor RIO demo now works correctly again when not accessed from localhost (#6111)

The new features and improvements include:

- Added support for the style "text-overflow: ellipsis" (#5982)
- Extended proprietary layout information objects in JavaScript with float precision coordinates in any length unit as well as the DOM ranges and base line positions of laid out lines of text (#6097, #6101)
- Conversions can now terminate on conditions like timeouts (#6154)
- Added option to set headers and cookies for connections from the wrappers to the PDFReactor Web Service (#6053)

8.0.8162 First Maintenance Release of PDFReactor 8

The bug fixes include:

- In JavaScript bound functions could not be used as event listeners and "getOwnPropertyNames" could not be called on "window" breaking some libraries incl. React. (#6039, #6060)
- Table and column widths were computed inaccurately for certain combinations of styles (#6015, #6017)
- Width computation was incorrect for absolutely positioned blocks with a width of 0px and for elements with a width of 100% inside elements with a width of auto. (#6047, #6050)
- Lines could wrap prematurely in some situations (#6046)
- Certain combinations of generated content could break BiDi processing. (#5651, #5746, #5981)
- Empty elements in HTML could cause unexpected results after XSLT transformations. (#5994)
- The web service created a superfluous font cache folder inside the home directory on startup. (#6056)

The new features and improvements include:

- Added support for the additional image formats ICO, CUR, PAM, PBM, PGM, PPM and BMP as well as limited support for PSD, DCX, ICNS and RBBE. (#6049)
- Added support for the CSS pseudo classes "checked", "enabled" and "disabled". (#5972)
- SVG now supports the color constant "transparent". (#6040)
- Improved error reporting in web service, e.g. for illegal configurations. (#6027)
- Web service now auto detects configuration format for generic mime types. (#6061)
- When multiple options of a select element are selected the last one is displayed instead of the first one, which is the common browser behavior for this illegal situation. (#5752)
- Python commandline is now executable and contains a shebang (#6029, #6030)

8.0.8096 First Hotfix Release of PDFReactor 8

The bug fixes include:

- Resolved a critical issue with the PDFReactor Web Service that could cause empty documents to be returned when converting multiple documents in parallel.
- Solved exception that sometimes prevented loading documents containing background elements in combination with gradients and borders. (#5987)
- Solved an issue where calling native JavaScript methods with too many arguments caused an exception. (#5663)
- Relative URLs in CSS "xhtml" functions no longer log an exception. (#5985)

The improvements include:

- Added support for the CSS3 general sibling combinator "~" (tilde).
- Improved JavaScript language support for ECMAScript 6 features.

8.0.8064 First Release of PDFReactor 8

The new features include:

- Improvements to the PDFReactor Web Service API:
 - Now supports REST.
 - Added JavaScript, Java and command line wrappers.
 - The command line wrapper for the Web Service API supports batch processing (incl. wildcards).

- Improved merging with existing PDF documents:
 - Pages from existing PDF documents can now be used like images and are embedded without quality loss. (#5192)
 - When appending, the page order can be configured freely using pages from all documents. (#5833)
 - When overlaying documents, the longer document can now be trimmed to the length of the shorter one. (#5009)
- Improved text layout:
 - Implemented support for baseline grids via the properties "-ro-line-grid" and "-ro-line-snap". (#5607)
 - "line-height: auto" is now resolved similarly to common browsers.
 - Improved line height computation and vertical positioning of text.
 - Font fallback no longer impacts the line height or vertical positioning. (#5924)
 - PDFreactor now respects the kerning of fonts. (#5243)
 - Appearance of underline is no longer influenced by the styles of child elements.
- Implemented new proprietary CSS rule for global document properties:
 - Whether the first page is a left or right page, optionally separately for layout and PDF viewer. (#5600)
 - The initial zoom level or setting when opening a PDF. (#5521)
 - Set in which view mode the PDF should be opened, e.g. 2 columns.
 - The initial page to scroll to when opening a PDF.
- PDF/X is now supported, specifically PDF/X-1a:2001, PDF/X-1a:2003, PDF/X-3:2002, PDF/X-3:2003 and PDF/X-4. (#5002)
- Now supporting all PDF/A formats. (#5713)
- The CSS function "calc" is now supported.
- The CSS property "box-sizing" is now supported. (#5414)
- Completed support for CSS3 background images including multiple layers and the sizes "cover" and "contain", as well as proprietary origin values for the entire page, optionally including the bleed.
- Implemented the CSS property "object-fit" including the values "cover" and "contain". (#5682)
- Implemented new CSS page pseudo class ":-ro-nth" to address specific page numbers and patterns.
- Implemented new CSS page pseudo class ":-ro-last" to address the last page of a document. (#4987)
- The "box-decoration-break" CSS property is now supported for block elements.
- Now supporting the CSS property "tab-size". (#5730)

- Added possibility to force PDFReactor to evaluate any desired CSS media type rules, e.g. "@media screen". (#5661)
- Added support for CSS media features (e.g. "@media (max-width: 12cm)") including configuration of returned values. (#5683)
- Added JavaScript property "window.devicePixelRatio", with the same value as the media feature "resolution". (#5967)
- Added media type "-ro-pdfreactor" for styles to be used exclusively by PDFReactor. (#5325)
- Now supporting all CSS length units including "rem", "ch", "vw", "vh", "vmin", "vmax" and "q". (#5822, #5647)
- Added support for the HTML5 element "wbr" (word break opportunity).
- Page order now allows addressing the nth-last page. Use negative page numbers to select the n-th last page. (#5680)
- Added support for embedding JavaScript into PDFs to be executed by Adobe Reader on load or certain events. (#5919)
- Added support for Spot colors (incl. Pantone). (#5457)
- CMYK colors now also support alpha values. (#5877)
- Individual PDF IDs ("named destinations") can now be disabled via the CSS style "-ro-anchor: none". (#5735)
- Added support for SVGZ images. (#5705)
- In addition to images, "safe memory" mode now also reduces memory footprint for SVGs. (#5696)
- Custom messages can now be logged in the PDFReactor log.
- Added constant that returns the PDFReactor version. (#4694)
- The pdfreactor.jar library now supports OSGi. (#5116)
- A custom URLStreamHandler object can now be set for PDFReactor. It will be used by PDFReactor to load resources. (#5732)
- Added support for Chart.js 1.0.2. (#5968)

The improvements include:

- Included Noto Fonts, Droid Sans Fallback and Quivira to support non-Latin characters (including CJK and Arabic) and symbols on systems without appropriate fonts.
- Massively improved performance on IBM JRE 6. (#5677)
- Updated the Jetty web server used by PDFReactor Web Service to version 9.
- Java VM options for the PDFReactor Preview app can now be set in a .vmoptions file.
- Added new PDFReactor packages for Linux that include a JRE. (#5662)
- Missing resources of various types are now logged as warnings. (#5841)

The bug fixes include:

- PDF size can no longer increase because of dashed or dotted borders with small widths. (#5525)
- Form elements no longer need to be enclosed by a "form" element in order to be properly converted to the appropriate interactive PDF form elements. (#3676)
- Fixed issue where soft hyphens were always displayed when the font missed a glyph for them. (#5926)
- Soft hyphens that were not visible in the PDF under certain specific circumstances are now displayed. (#5922)
- Font fallback no longer behaves differently in certain scenarios affected by hyphenation. (#5921)
- Forced line breaks are no longer moved into the next line under specific circumstances. (#5888)
- Fixed excessive creation of temporary font files under certain circumstances. (#5444)
- The SVG "visibility" attribute now behaves correctly for SVG "g" element. (#5672)
- MathML elements now use fonts specified via the API. (#5959)
- Using the "innerHTML" JavaScript function on script elements now returns the expected content. (#5893)
- Table headers were not repeated if the table was split between "tbody" elements. (#5903)
- Fixed JavaScript error when supplying more than one argument to DOMTokenList methods "add" and "remove". (#5894)
- Fixed layout issue when flowing regions onto a named page. (#5876)
- The width of a floated element is now calculated correctly when it contains an element with "max-width". (#5738)
- Fixed issue where documents containing tables with values for "rowspan" much larger than the actual number of rows, could not be converted. (#5870)
- When non-embeddable fonts are encountered, they are now properly skipped and appropriate warnings are logged. (#3389, #5410)
- URLs set via "addFont" are now resolved against the API or document base URL. (#5499)
- The width of an element with "overflow: hidden" is now computed correctly if it is next to a float. (#5739)
- Floats from empty lines no longer overlap with content and are now positioned correctly. (#5313)
- Fixed issue where table column widths were incorrect for tables containing empty table row groups. (#5579)
- Using "enableDebugMode()" no longer leads to an exception when the document contains an element with a namespace attribute. (#5655)

- Fixed issue that caused an exception when replaced elements exceeded the page as ExceedingContent. (#5803)
- Solved issue where the table cell alignment was wrong when the cell contained an empty block inside an inline-block. (#5719)
- Awesomizr.js could generate incorrect page numbers under certain circumstances. (#5641)
- Tab characters are no longer displayed as boxes in certain scenarios. (#5731)
- Fixed issue where "overflow: visible" did not work for SVGs. (#5694)
- applicationValue("com/realobjects/pdfreactor/start-page-number") now returns the correct value instead of always returning "1". (#5117)
- Fixed elements being displayed incorrectly in combinations of "position: relative" and "overflow: hidden".
- The CSS generated content "leader(space)" is displayed correctly again. (#5052)
- Fixed NPE in relation to iFrames. (#5934)
- Elements with "position: fixed" are now sized correctly. (#5367)
- The HTML "figure" element is now mapped to the "Div" PDF tag instead of the "Figure" tag. (#5673)
- The height for "textarea" elements with "height: auto" is now calculated correctly. (#5130)

The changes include:

- The previous PDFReactor API has been moved to the "legacy" sub-package and replaced by a new API. Please see the readme file for further information.
- The officially recommended Java VM now is Oracle JRE 8.
- Apple JRE 6 is no longer supported. Please use the latest Oracle JRE.

7.0.7447.1 First Hotfix Release of PDFReactor 7.0.7447

The bug fixes include:

- Performance issues when viewing the resulting PDFs in Chrome

7.0.7447 Seventh Bugfix Release of PDFReactor 7.0

The new features and improvements include:

- The CSS color function "rgba" is now available in SVGs. This is used by various charting libraries, primarily for 3D charts (#5549)

- The CSS color function "cmyk" is now available in SVGs, making "RO-CMYK" ICC colors obsolete (still supported for compatibility)
- SVG elements are now properly affected by styles set in the containing HTML document (#5543)
- Improved BiDi and Arabic shaping in SVGs (#5588)
- Script tags with an unknown type are now skipped (#5550)

The bug fixes include:

- Prefixed CSS properties in SVG elements caused the following properties to be ignored (#5603)
- SVGs referenced via "embed" elements were not loaded (#5591)
- Converting SVGs including links could in rare cases lead to exceptions (#5633)
- Listeners for the "load" event added via "window.addEventListener" were not triggered (#5621)
- URLs containing legal line breaks could not be processed (#5582)
- In some cases the max-width property of images inside tables was not taken into consideration when determining the width of the table cell and table (#5609)
- "bottom" positioned elements with no fixed height had wrong vertical positioning if they had vertical margins, paddings or borders (#5623)

7.0.7375 Sixth Bugfix Release of PDFreactor 7.0

The new features and improvements include:

- Vastly improved support for JavaScript libraries, including "AngularJS" and "Leaflet", due to numerous JavaScript, SVG and DOM related improvements.
- Lines and shapes are now handled in a different way that avoids issues displaying thin lines in certain PDF viewers.
- Debug mode no longer disables logging to the attached logger or the console. The log attached to the PDF now properly shows HTML and XML content as source.
- The "applyAdaptivePageBreaks" method of the "Awesomizr" JavaScript library is now more robust.

The bug fixes include:

- Tables with the "table-layout" CSS property set to "fixed" and containing rows with varying amount of columns could cause non-terminating layouts.

- Tables with the "table-layout" CSS property set to "fixed" and "border-collapse: collapse" or containing cells with borders or padding could be laid out too wide. (#5595)
- Rasterization of complex SVG graphics at "-ro-rasterization-supersampling" values larger than "2" could fail in certain JREs. (#5556)

7.0.7290 Fifth Bugfix Release of PDFreactor 7.0

The bug fixes include:

- Text of floated elements inside positioned elements was added to the PDF twice at the same location (#5528)

7.0.7279 Fourth Bugfix Release of PDFreactor 7.0

The new features and improvements include:

- Updated Unicode support to Unicode 7.0

The bug fixes include:

- In some cases the widths of thin lines, incl. underlines of small text, were too small
- The library awesomizr.js contained a superfluous log output

7.0.7270 Third Bugfix Release of PDFreactor 7.0

The bug fixes include:

- PDF merging could cause exceptions

7.0.7267 Second Bugfix Release of PDFreactor 7.0

The new features and improvements include:

- The JavaScript library Awesomizr can now be used to create multiple tables of contents
- When creating a table of contents with Awesomizr, you can now specify a function which determines the text of the given entry for the table of contents or skips the entry entirely
- Now setting the document URL as "referer" header for HTTP requests which improves compatibility with external resource providers like Typekit

The bug fixes include:

- PDF merging in conjunction with encryption caused exceptions (#5510)
- Images in table headers caused exceptions when the table was spanning multiple pages (#5481)
- When the subset of a font cannot be created, an exception was thrown instead of embedding the entire font

7.0.7243 First Bugfix Release of PDFreactor 7.0

The new features and improvements include:

- HTMLAnchorElement and Location now implement URLUtils (read-only), which drastically improves compatibility to AngularJS (#5467)
- Tables spanning a large amount of pages are now converted significantly faster
- Improved caching of web fonts, reducing the amount of temporary font files (#5444)
- The Python API now also supports Python 3 (#5473)
- Official support for Oracle JRE 8

The bug fixes include:

- Images embedded in SVGs as data URIs could not be converted (#5432)
- AFM fonts in the system font directory could lead to missing fonts in the resulting PDFs (#5425)
- Some methods were missing in the wrapper APIs or did not work properly (#5422, #5423)
- Clipping for "overflow: hidden" was not determined correctly
- Setting "font-weight: lighter" in an @font-face rule caused exceptions (#5417)
- html(string) method of jQuery 2 caused exceptions (#5415)
- When CMYK was set as default color space and an error page was created an exception was thrown
- Command line parameter "defaultlanguage" was misspelled as "defualtlanguage" (#5455)
- Minor fixes to stacking contexts and related properties

7.0.7174 First Release of PDFreactor 7.0

The new features and improvements include:

- Support for CSS3 Regions (#5122)

- Support for CSS3 Running Elements (#5056)
- Support for PDF/A-3a (#5095)
- Support for PDF annotations, incl. comments, notes and tooltips (#4763, #5067)
- Support for custom JavaScript functionality to retrieve the page number, coordinates and sizes of elements in the layout
- Support for high quality HTML5 Canvas rendering without rasterization
- Support for first-letter, including drop caps
- Support for the CSS3 property "text-align-last" as "-ro-text-align-last"
- Introduced properties "break-before", "break-after" and "break-inside" and improved page breaking, especially in multi-column (#5335, #5302)
- Gradients, transforms and multi-column properties no longer require the prefix "-ro-"
- The properties "width" and "height" of "img" elements now work according to the HTML5 specification, also the properties "naturalWidth" and "naturalHeight" have been added
- The property "-ro-title" now cleans the content, e.g removing line breaks, and can be disabled via the value none (#5263, #5264)
- SVG links are now converted to PDF links, similar to HTML links
- List bullets not available in the current font are now drawn independently from fonts, also the property "list-style-image" now supports CSS gradients
- JavaScript environment now handles attributes case-insensitively (#5260) and provides a "window.plugins" object of the correct type
- Improved compatibility to JavaScript libraries that rely on DOM-object-prototypes
- Improved quality of bidi text layout, especially for mixed content containing parenthesis
- Improved font handling and added option to disable system fonts
- Optimized JavaScript performance in large documents
- The "CleanerThread" of the library "Batik" is now terminated appropriately in Servlet 3.0+ environments (#5332)
- The included JavaScript library Tableizr has been replaced by its successor Awesomizr which includes additional functionality like automatic creation of tables of contents

The changes include:

- Generated content can now be created with a "content" value of empty string
- The pdfreactor.war no longer contains the pdfreactor.jar, so when using the pdfreactor.war in a different application server the pdfreactor.jar has to be added to the class path

The bug fixes include:

- Empty inline elements could offset the bidi runs in mixed content (#5272)

- Canvas did not scale stroke widths (#5256) and produced unexpected results for some uncommon parameter values for paint primitives
- In rare cases PDF Documents could not be printed via PCL
- "not" selectors containing pseudo classes could cause exceptions (#5269)
- "a" elements with uncommon values caused exceptions when tagging was enabled
- Media queries in style sheets caused errors
- First word of line could not be hyphenated and hyphenation did not affect minimal width as expected

6.3.6828.3 Third Hotfix Release of PDFreactor 6.3

The bug fixes include:

- Certain combinations of CSS "not"-selectors could cause exceptions.

6.3.6828.2 Second Hotfix Release of PDFreactor 6.3

The bug fixes include:

- The function "addUserScript" of the PHP wrapper caused conversions to fail
- The Java API documentation did not contain descriptions

6.3.6828.1 First Hotfix Release of PDFreactor 6.3

The new features and improvements include:

- The JavaScript "XMLHttpRequest" now supports the "onload" and "onerror" callbacks required for jQuery 2 ajax.

The bug fixes include:

- Documents containing multi-column layouts could not be printed in some environments.
- Illegal font files, containing an empty font family name, caused IndexOutOfBoundsException, terminating the conversion.
- Accessing the JavaScript "document" object was slow for complex documents.

6.3.6828 First Release of PDFreactor 6.3

The new features and improvements include:

- Support for SVG-based JavaScript libraries, like Highcharts (highcharts.com) and amCharts (amcharts.com) (#5189)
- New JavaScript modes: `JAVASCRIPT_MODE_ENABLED_TIME_LAPSE` (simulates faster progression of time to skip fade in animations) and `JAVASCRIPT_MODE_ENABLED_REAL_TIME` (disables optimizations)
- Support for the CSS properties "letter-spacing" and "z-index" (#4903)
- Support for SVG elements with no width and height attributes, in addition SVGs with certain illegal structures will now be rendered
- SVG images now have to be rasterized in fewer cases, also there are new CSS properties to adjust the rasterization process: "-ro-rasterization" and "-ro-rasterization-supersampling"
- Reduced memory footprint for documents containing a large amount of elements
- Improved accessibility for form elements
- Documents loaded via the CSS function "xhtml()" are now parsed using the HTML5 parser
- Optimized default configuration of PDFreactor web service
- Various improvements to document parsing as well as font and JavaScript handling (#5142)

The bug fixes include:

- Fonts added via the API were not available in documents loaded via the CSS function "xhtml()" (#5213)
- The CSS property border-radius did not handle complex shorthands correctly (#5184)
- "setPrintDialogPrompt" had no effect when the merge mode "overlay" was used
- PDF/A-1a mode did not prohibit attachments
- JavaScript could not be used from pdfreactorcore.jar because of missing library (#5194)

6.2.6560.5 First Maintenance Release of PDFreactor 6.2

The improvements include:

- The Ruby API now officially supports Ruby versions 1.8, 1.9 and 2.0 (#5029)
- Enhanced strictness for PDF/a-1a output

6.2.6560.4 First Release of PDFReactor 6.2

The new features and improvements include:

- Added support for PDF attachments
- Added support for custom document properties
- Added support for overprinting
- Added option to specify how the clickable area of links is determined

The bug fixes include:

- Content inside "xhtml" CSS function did not use the specified default color space
- Some PDF specific exceptions did not contain messages

6.1.6560.3 Third Hotfix Release of PDFReactor 6.1

The bug fixes include:

- Clickable area of links in the "XHTML()" function was not placed correctly. (#5072)
- Column spans could cause wrong widths for certain tables. (#5066)
- Certain combinations of named pages and multi column could cause endless loops. (#5079)
- "inline-block" elements did not support the property "overflow". (#5077)
- Empty pages did not inherit named page value. (#5069)
- The PDFReactor Demo GUI deviated from the actual PDF output on styled "i" and "b" elements. (#5070)

The improvements include:

- More verbose output on failed license key validation.

6.1.6560.2 Second Hotfix Release of PDFReactor 6.1

The bug fixes include:

- jQuery 1.9.0 did not initialize. (#5049)
- "getClientRects" and "getBoundingClientRect" could not be accessed via JavaScript. (*note: This fix improves the results of jQuery "offset" and similar methods of other libraries.*)
- Tables causing named pages could get cut off (#5043)

- Certain combinations of named pages and JavaScript could produce incorrect page breaks. *(note: This can still occur in the demo.)* (#5051)
- Exceptions during license key validation are now appended to the error message.

6.1.6560.1 First Hotfix Release of PDFReactor 6.1

The bug fixes include:

- "offsetWidth", "offsetHeight" and similar JavaScript properties now work properly for inline elements and act more similarly to browsers (#5022)
- Byte order mark was not skipped for style sheets included via "link"-tags (#5034)
- The PHP API is now compatible with PHP 5.4 (#5017)

6.1.6560 First Release of PDFReactor 6.1

The new features and improvements include:

- PDFReactor now includes Raster Image Output functionality to create PNG, JPG, GIF, TIFF or BMP images instead of PDF documents.
Please see chapter 4.2 of the PDFReactor manual or try the PDFReactor Raster Image Output demo.
The Raster Image Output option needs to be licensed separately. Please contact support@realobjects.com for more information.
- The document type of the result of an XSL transformation can now be overridden (#4928)
- The Exceeding Content functionality provides significantly more information about the offending content (#4921)

The bug fixes include:

- The HTML5 parser aborted when encountering attributes with undeclared name space prefixes (#4924)
- Page breaks could occur between the table header and the first regular table row (#4940)
- Web fonts did not evaluate sources after the first local() function (#4992)
- SVGs containing bitmap graphics were not displayed (#4994)
- Exceeding Content functionality reported false positive when line-height was set to low values (#4922)
- Table cells with fixed widths did not adjust their sizes to avoid overflowing content (#4247)
- Certain table constructs could cause multiple pages appearing on one (#4990)

- Inline elements at the beginning of a line near the bottom a first column could cause superfluous columns (#4983)
- Errors (e.g. OutOfMemoryError) during image loading were not passed through to the integration.

6.0.6400c First Release of PDFreactor 6.0

The new features and improvements include:

- Added support for JavaScript
- Added support for HTML5, including the canvas element
- Added support for CSS3 2D transforms
- Added support for CSS3 gradients.
- Fonts that can be used by PDFreactor are no longer restricted to system directories: Arbitrary font directories can now be added using an API methods. Individual font URLs can be specified via the @font-face CSS3 property or an API method
- Added support for additional CSS3 selectors, including "nth-child"
- Added support for transparent colors using the CSS3 function "rgba"
- Outgoing HTTPS connections can be configured to be much more lenient on incorrect certificates

The changes include:

- PDFreactor now requires Java 6 or newer
- The default parser for HTML documents is now the HTML5 parser. The XHTML parser can still be selected.

The bug fixes include:

- "Exceeding content" detection logged false positives for inline elements spanning multiple lines (#4888)
- The method "setMergeURL" in the PHP wrapper did not have any effect (#4747)

5.2.5996 First Release of PDFreactor 5.2

The improvements include:

- PDF tagging adds more information about the placement of elements
- Centered text can no longer overflow on the start (usually left) side

- The log of the web service now contains thread IDs, so lines can be related to conversion processes
- PDFReactor Demo application uses proper menu bar on OS X
- Successfully tested with Java 7

The bug fixes include:

- Certain complex table layouts could slow down conversion processes and cause exceptions (#4143,#4734,#4752)
- Named pages could not be created by floated or absolute positioned elements (#4788)
- The CSS properties "page-break-before" and "page-break-after" along with the values "left" and "right" caused missing pages.
- The method "setMergeURL" could not be called in the PHP wrapper API (#4747)

5.1.5812.2 First Release of PDFReactor 5.1

The new features and improvements include:

- Improved PDF tagging, which provides improved accessibility and better exportability for resulting tagged PDF (incl. PDF/A) documents
- Added functionality to easily add QR codes to documents

The bug fixes include:

- Text content in tagged PDF documents was sometimes added multiple times.
- .Net API did not allow generation of PDF/A documents and always returned an empty log
- Under rare circumstances CPU count could not be determined properly

5.0.5812.1 First Hotfix Release of PDFReactor 5

The bug fixes include:

- TrueType collections (TTC) could not be used in PDFReactor 5. (#4506)
- Version of .NET wrapper DLL was not updated to 5. (#4521)

5.0.5812 First Release of PDFreactor 5

The new features and improvements include:

- Request headers and cookies can be specified for all outgoing HTTP connections (allows to pass session IDs to PDFreactor).
- Support for HTTP basic and digest authentication.
- Automatic font fallback for characters missing in the specified font.
- Enhanced PDF merging, allowing PDF overlay and merging of multiple PDF documents.
- Rearrange the order of pages and N-up functionality (multiple pages per sheet) allow easy business card and booklet printing.
- PDF documents can be digitally signed.
- Improved shrink-to-fit and the option to set the pixels per inch.
- Log and programmatically process content exceeding its page or parent.
- Additional version of PDFreactor that does not have all libraries packaged into one jar.
- Option to throw exceptions on license issues.
- Option not to embed fonts in generated PDF documents.
- Method to retrieve the number of pages of a created document.
- Table row page splitting is done correctly when cells span multiple rows.
- Support for numerous international numbering systems for lists and counters, including all specified by CSS 2.1
- Support for CSS3 color keywords.
- Default styles for HTML5 elements.
- Support for "z-index" property for page margin boxes.
- Illegal base URL are logged as warnings.
- Successfully tested with Java 7 RC.
- Improved performance.

The changes include:

- Included Jetty application server updated to version 7.4.4

The bug fixes include:

- Fonts do no longer have to be named in certain ways. (#4438)
- Non-embeddable fonts are now ignored. (#3389)
- PDFreactor Demo application uses fonts identical to PDF conversion process. (#4388)

- Page breaks set by "page-break-after" also work for elements enclosed in containers. (#4340)
- Perl API wrapper can handle documents including special characters. (#4356)
- Page breaks caused by the CSS values "left" and "right" are laid out correctly. (#4337)
- "text-indent" is no longer applied to the first line of a page if it is not the first line of a paragraph. (#4148)
- Named pages are applied more reliably. (#4379)
- Language or keywords no longer missing in some cases. (#4391, #4437)
- Running table headers are no longer missing in some cases. (#4145)
- The first run inside Weblogic is no longer disproportionately slow. (#4331)

4.1.5102 First Release of PDFreactor 4.1

The new features include:

- The CSS 3 property 'word-wrap' is supported.

The improvements include:

- Layouts are more accurate and space efficient.
- Clickable links in inner XHTML are supported.
- The CSS property "target-text" works with ":before" and ":after" pseudo elements.

The changes include:

- License error notices are added as first page.
- The available languages for automatic hyphenation have changed (see manual for a complete list of languages including their matching language codes).

The bug fixes include:

- White spaces directly succeeding footnote calls are handled correctly.
- Widths or heights of tables with the style "table-layout: fixed" are computed correctly.
- List bullets are no longer placed incorrectly when near a floated box.
- Missing resources or illegal names for form fields are logged correctly.
- Default host and port are set in Perl API.
- The cleanup Tagsoup no longer adds unnecessary attributes to elements.

4.0.4491.1 First Release of PDFReactor 4.0

The new features include:

- Creation of ISO 19005-1 Level a (PDF/A-1a) conformant files
- Multi-column layouts according to CSS 3 "Multi-column layout"
- Footnotes according to CSS 3 "Generated Content for Paged Media"
- Setting the PDF page boxes: trim, bleed, media, crop and art
- Adding printer marks including trim marks, bleed marks, registration marks and color bars
- Support for oversize page formats of the DIN RA and SRA series
- Option to save memory when documents reference many or large images
- Ability to set colors via the CSS function hsl()

The improvements include:

- Significantly improved performance of conversion processes, especially for large input documents
- Support for styling HTML forms including interactive PDF forms
- Form elements adopt their names from the input document
- Option to convert color key words to CMYK colors
- Improved PDF tagging
- Advanced "Create PDF" dialog in PDFReactor Demo application

The changes include:

- CSS cmyk() function changed to accept float values from 0.0 to 1.0 (CSS 3 compliant)
- Removed deprecated logging API

The bug fixes include:

- Improved support for non-Latin characters in non-UTF-8 encoded input documents
- Improved support for mixed left-to-right and right-to-left language content
- Transparent images are rendered correctly and no longer influence the other colors
- Styles for named pages are now prioritized over left and right-hand pages
- The property "background-size" with 2 percentage values no longer breaks images in PDFReactor Demo application

3.1.3956.2 Maintenance Release of PDFReactor 3.1

The new features include:

- Missing font variants (bold, italic) are simulated

The improvements include:

- Improved compatibility with IBM Java VMs and IBM WebSphere

The bug fixes include:

- The cleanup "TagSoup" could not be selected in PDFReactor 3.1
- An empty table row group at a page break could cause an exception under rare circumstances

3.1.3956 First Release of PDFReactor 3.1

The new features include:

- Further improved support for non-Latin content, i.e. right-to-left and bidirectional layouts and texts (e.g. Hebrew and Arabic), including shaping of Arabic characters. (*This feature is experimental*)
- APIs for direct integration into Perl, Python and Ruby scripts.
- The property `-ro-counter-set` which allows resetting counters from within the content, e.g. resetting the page counter at certain headings.
- Support for the `last-child` pseudo class.
- Support for table captions.

The improvements include:

- Repeated background images have been optimized.
- Better naming of PDF bookmarks.
- Reduced memory footprint.

The bug fixes include:

- Owner and user passwords work correctly.
- Removed PHP 5 dependent code from PHP wrapper.
- Multi-byte characters can be used inside the `XHTML ()` CSS function.

- The `white-space` property also works for inline elements.
- Height values expressed in percent are computed correctly and the right fallback mechanisms are used.
- The `target-counter` property also works when no base URL is set.
- Removed dependencies that are not available in IBM Java 1.4.2.

3.0.3311 First Maintenance Release of PDFReactor 3.0

The new features include:

- Added the `SetTimeout` method to the .NET API. This method allows to set the timeout in ms for the SOAP web service to answer. Infinite waiting is also possible by setting the timeout to -1.
- The PDFReactor WAR file can now be used as-is in any servlet container.
- Support for additional list-style-types: box, check, diamond and hyphen.
- Added support for the style "font-variant: small-caps".
- Added support for the "before" and "after" keywords of the "string-set" property.

The improvements include:

- Improved vertical alignment, fragment positioning and line height computation.
- Improved font detection on Unix-like systems.
- Removed dependency to newer versions of the `LSSerializer` interface. This should solve problems with VMs that have their own (older) versions of the `LSSerializer` interface in the classpath (such as the IBM JVM 1.4.2).
- Functionality enhancement: improved error handling for the PHP API.
- Nested counters now function as specified by the CSS 2.1 standard.

The bug fixes include:

- When two documents containing AcroForms are merged, all form fields are now preserved.
- The .NET wrapper is now an assembly with a strong name.
- Fixed abort when using JTidy in the IBM JVM.

Changes compared to previous versions:

- Deprecated PDFReactor 1 `LogEvent` classes
- The "center" element now has the style "text-align: center".

3.0.2995 Release of PDFReactor 3.0

The new features include:

- Support for MathML (this feature requires Java 1.5 or higher)
- It is now possible to use XHTML in headers and footers
- Added SOAP connection interface for the PHP API free of any dependencies to external libraries
- All line breaks are now Unicode compliant (incl. multibyte character sets such as Chinese and Japanese)
- Added ability to embed flash and other object
- Support for named strings
- It is now possible to generate cross references using the CSS properties target-counter, target-counters and target-text
- Leaders can now be added to your document via the CSS function leader()
- It is now possible to dynamically add a table of contents to a document using an XSLT style sheet
- Hyphenation support for several languages
- Automatic scaling of content to the page size or using a specified scale
- Support for setting Viewer Preferences

The bug fixes include:

- Improved handling of floats:
 - Floated images were not moved to the next page when they are too high
 - Text sometimes did not flow correctly around floating boxes
- Incorrect rendering of generated block content upon images
- entities in bookmarks were converted to euro symbols
- "overflow: hidden" was applied to inline elements, causing invisible text
- Image maps caused problems in layout
- @charset entries sometimes caused style sheets to be ignored
- Width and height of SVG images were not always correct
- In HTML lists, uppercase alphanumeric numbering (ol[type="A"]) was not possible
- Invalid characters in attributes aborted PDFReactor

2.0.1823b Second Maintenance Release of PDFReactor 2.0

The new features include:

- Support for images improved:
 - Support for TIFF images incl. CMYK
 - TIFF and SVG supported in IMG tags, background and generated content
 - Improved image quality: direct embedding of images without JPEG recompression
- Additional encoding UTF-16LE now supported

The bug fixes include:

- Improved layout of tables spanning multiple pages
- Improved layout of nested tables
- Fixed handling of the @page:first selector
- Fixed page counter for merged documents
- Fixed support for bottom property of absolutely positioned elements

2.0.1544.3 Maintenance Release of PDFReactor 2.0

The bug fixes include:

- margin-left: auto; margin-right: auto did not work to center tables
- Too much spacing between words in last line of paragraph with text-align: justify;

The new features include:

- Initial support for CMYK images (TIFF, JPG, PNG)

CMYK tiff images can be used by adding this to your style sheet:

```
object[type="image/tiff"] {  
    -ro-replacedelement: "com.realobjects.xml.formatters.PDFImageFormatter";  
    -ro-source: attr(data);  
}
```

This is a sample HTML document, showing how to embedd TIFF images:

```
<html>  
    <head>  
        <style>  
            object[type="image/tiff"] {  
                -ro-replacedelement: "com.realobjects.xml.formatters.PDFImageFormatter";  
            }
```

```

        -ro-source: attr(data);
    }
</style>
</head>
<body>
    <object alt="PDFReactor logo CMYK"
        data="PRlogoCMYK.tif"
        type="image/tiff"
        style="height: 3cm;
        border: 1px dashed red" />
    <br />
    <object alt="PDFReactor logo CMYK"
        data="PRlogoCMYK.tif"
        type="image/tiff"
        style="border: 1px dashed green" />
</body>
</html>

```

2.0.1544 Release of PDFReactor 2.0

The new features include:

- New high-performance .NET and PHP APIs (require local Jetty server with PDFReactor webservice)
- Complete support for HTML form elements
- AcroForms support
- Support for EAN-128, DataMatrix 2D and PDF417 2D barcodes
- Support for XML namespace CSS selectors
- DATA URIs supported for images
- Tagged PDF support
- Images in generated content (page margin boxes and in-flow)
- Resizing of background images supported (based on CSS3 draft, property background-size)
- CYMK color support (e.g. color: CMYK(65%,20%,0%,0%)), CMYK images are not yet supported
- Improved error reporting in CSS parser
- Reduced memory footprint
- Added "Save as" option to PDFReactor Demo application

Important: Java API Changes in PDFReactor version 2.0

You have to adjust your integration of PDFReactor, otherwise run-time errors will occur.

The main change of the PDFreactor 2.0 API when compared to the PDFreactor 1.0 API is that the PDFreactorConfiguration class does not exist anymore.

While you had to create a PDFreactorConfiguration object in version 1.0 to set specific PDFreactor settings, this is no longer necessary. All methods that were part of the PDFreactorConfiguration class can now directly be called within the PDFreactor class.

To illustrate this difference, here is a code sample which shows how to set the PDF Author tag in PDFreactor 1.x and version 2.0.

>Old (Version 1.x):

```
PDFreactor pdfReactor = new PDFreactor();
    PDFreactorConfiguration pdfReactorConfiguration = new PDFreactorConfiguration();

    // In version 1.x, configurations settings are still specified using methods of PDFreactorConfiguration
    pdfReactorConfiguration.setAuthor("Peter Sampler");

    InputSource is = new InputSource("http://www.realobjects.com/");
    FileOutputStream fileOutputStream = new FileOutputStream("realobjects.pdf");
    pdfReactor.renderDocument(is, pdfReactorConfiguration, fileOutputStream);
    fileOutputStream.close();
```

New (Version 2.0):

```
PDFreactor pdfReactor = new PDFreactor();

    // In version 2.0, configuration settings are specified directly using methods of PDFreactor.
    pdfReactor.setAuthor("Peter Sampler");

    InputSource is = new InputSource("http://www.realobjects.com/");
    FileOutputStream fileOutputStream = new FileOutputStream("realobjects.pdf");
    pdfReactor.renderDocument(is, fileOutputStream);
    fileOutputStream.close();
```

As you can see, upgrading your integration to use the new API of PDFreactor version 2.0 is really simple. Just make sure all calls to PDFreactorConfiguration objects are replaced by calls to PDFreactor objects. Also remove the PDFreactorConfiguration parameter from the renderDocument call.

Additional new API methods have been introduced with PDFreactor 2.0, but this should have no effect on your integration. Also, some API methods are now deprecated and should no longer be used. For details about the API changes, please refer to the PDFreactor API documentation.

The bug fixes include:

- setBaseURL did not work for external stylesheets
- "page-break-inside-avoid" was ignored for tables
- Layout error occurred when splitting certain tables

- min-height did not always work
- Document resources with an absolute URL could not be loaded if no base URL is available
- Empty links caused PDFreactor to abort

1.1.936.7 Second Maintenance Release of PDFreactor 1.1

The bug fixes include:

- Exception when running PDFreactor in IBM JVM
- Toolbar looked strange in Sun JRE Version 1.5.0_08

1.1.936.5 Maintenance Release of PDFreactor 1.1

- Significantly improved the performance with documents containing float elements

1.1.936 Release of PDFreactor 1.1

The new features include:

- Added an API method to concatenate the newly rendered PDF content with an existing PDF
- Added ASP and ASP.NET samples

The bug fixes include:

- Pagecounters sample code in manual is wrong
- td height attribute does not work
- Using the stdin option doesn't work in the CLI
- The number of runs parameter has no effect in the CLI
- Certain documents rendered too slowly
- Bookmarks are not generated under certain circumstances
- Pages containing log data are generated with a wrong dpi value
- Improved support for the line-height property

1.0.882 Service release of PDFreactor 1.0

The changes include:

- Added information in the manual about the definition of bitmap images
- CLI version now returns error exit codes
- Changes in the PDFreactorConfiguration class to improve API consistency:
- The methods

```
int getRestrictions(void)
void setRestrictions(int)
```

were removed. To set/get the PDF restrictions, please use the individual API methods.

The methods

```
String getUserStyleSheet(void)
void setUserStyleSheet(String)
ArrayList getUserStyleSheetUrls(void)
void addUserStyleSheetUrl(String)
void addUserStyleSheetUrl(ArrayList)
void addUserStyleSheetUrl(String[])
void setUserStyleSheetUrl(ArrayList)
```

were removed. They were replaced by

```
void addUserStyleSheet(org.w3c.css.sac.InputSource)
void removeAllUserStyleSheets(void)
ArrayList getAllUserStyleSheets(void)
```

which provide the same functionality, but offer more flexibility.

If you were using the removed API methods, please change your code to use the new methods.

E.g. to add a user style sheet from a URL, the old code was:

```
myConfig.addUserStyleSheetURL("http://test/mystyle.css");
```

The new code is:

```
myConfig.addUserStyleSheet( new org.w3c.css.sac.InputSource( "http://test/mystyle.css" ));
```

The bug fixes include:

- Absolute positioning does not work in paged mode and for bottom position
- The height and width properties should not apply to non-replaced inline-level elements

- Width/height of replaced elements does not fall back to intrinsic size if the containing block's height is not set explicitly
- Inline text containing surrounding area is sometimes positioned wrong if line breaks occur
- Curly braces (" { } ") in path of xml-file crash SACParser
- If inline text content extends over element borders no linebreaks should be inserted.
- "%" in path causes exception
- Local dtd files are not loaded using the catalog resolver
- PDF encryption does not work
- XSLT doesn't work in CLI

1.0.800 First public release of PDFreactor
