

The l3pdftools package collection of pdf related commands

The L^AT_EX3 Project*

Released XXXX-XX-XX

1 l3pdftools documentation

This module collects a number of commands of the tool type which haven't found another home yet ... Probably this module will disappear again at some time.

<code>\pdf_text_convert:nn</code>	★	<code>\pdf_text_convert:nn {<format>} {<content>}</code>
-----------------------------------	---	--

New: 2020-07-04

This converts *<content>* following the rules defined by *<format>*. Non-ascii input should be utf8 encoded. Currently the following formats exist:

name This will first expand the content with `\text_expand:n` and then escape it in the way needed in a PDF Name with `\str_convert_pdfname:e`.

name-print This does the same as the **name** format, but also adds a slash before, so `\pdf_text_convert:nn{name-print}{abc}` will output `/abc`.

*E-mail: latex-team@latex-project.org

<hr/> <code>\pdf_text_convert:nnN</code> <hr/>	<code>\pdf_text_convert:nnN {<format>} {<content>} {<tlvar>}</code>
<hr/> New: 2020-07-04 <hr/>	This converts <i><content></i> following the rules defined by <i><format></i> and stores the result in <i><tlvar></i> . The assignment is done locally. Non-ascii input should be utf8 encoded. Currently the following formats exist:
	name see above
	name-print see above
	utf8/string this converts with <code>\str_set_convert:Nnnn</code> into utf8/string.
	utf8/string-print this converts into utf8/string and add parentheses around.
	utf8/URIpdf this converts with <code>\str_set_convert:Nnnn</code> into utf8/url and then replace reserved and digits back from the percent encoding. Parentheses are escaped.
	utf8/URIpdf-print this converts into utf8/URIpdf and add parentheses around.
	utf16/string this converts with <code>\str_set_convert:Nnnn</code> into utf16/string.
	utf16/string-print this converts into utf16/string and add parentheses around.
	utf16/hex this converts into utf16/hex
	utf16/hex-print this converts into utf16/hex and add bracket around.

1.1 BDC operator / Properties resource

Entries to the /Properties dictionary in the page resources can be added with dvips only through side-effects: if a BDC-mark is created dvips/ghostscript will automatically create the necessary objects and names. To get a sensible abstraction the code does the same for the other backends if the core management code has been activated. This means that the behaviour of the command is different then. The `\pdf_bdc_obj:...` should only be used if the management is active.

<hr/> <code>\pdf_bdc:nn</code> <hr/>	<code>\pdf_bdc:nn {<tag>} {<dictionary content>}</code>
<hr/> Updated: 2020-07-03 <hr/>	This command adds a BDC marked content operator to the current page stream. <i><tag></i> is the tag of this operator (without the leading slash), <i><dictionary content></i> is the content of the second argument. If the PDF resource management is active an dictionary object with the content is created and referenced with a name in the BDC operator. Without the resource management the content is used directly. It then depends on the backend how it is handled: with dvips a name is used while the pdfmode engines and dvipdfmx write the content into the stream.
<hr/> <code>\pdf_bdc_obj:nn</code> <hr/>	<code>\pdf_bdc_obj:nn {<tag>} {<object name>}</code>
<hr/> New: 2020-07-03 <hr/>	This command adds a BDC marked content operator to the current page stream. <i><tag></i> is the tag of this operator (without the leading slash), <i><object name></i> is a the name of an dictionary object reserved with <code>\pdf_object_new:nn</code> and filled with <code>\pdf_object_write:n</code> with the properties of the BDC. Reusing a predefined object can save space but the command works correctly only if the resources management has been activated and should be used only if this can be ensured.

<code>\pdf_bdc_obj:n</code>	<code>\pdf_bdc_obj:n {<tag>}</code>
-----------------------------	---

Updated: 2020-07-03 This command adds a BDC marked content operator to the current page stream. `<tag>` is the tag of this operator (without the leading slash). As object this commands uses the last anonymous dictionary object created with `\pdf_object_now:nn`. It lies in the responsibility of the user that the last object is the wanted one. Like with `\pdf_bdc_obj:nn` the command works correctly only if the resources management has been activated and should be used only if this can be ensured.

<code>\pdf_bmc:n</code>	<code>\pdf_bmc:n {<tag>}</code>
-------------------------	---------------------------------------

New: 2019-10-17 This command created a BMC marked content operator. The argument is the tag without the leading slash. It can be e.g. used for simple artifact markers.

<code>\pdf_emc:</code>	<code>\pdf_emc:</code>
------------------------	------------------------

New: 2019-06-30 This command closes the BDC marked content operator opened with `\pdf_bdc:nn`. It should be on the same page as the bdc-command.

```

\pdf_object_new:nn {objA}{dict}
\pdf_object_write:nn {objA}{/Type/Artifact}
\pdf_bdc:nn {Span}{objA}
text
\pdf_emc:

```

1.2 references to page objects

<code>\pdf_object_pageref:n *</code>	<code>\pdf_object_pageref:n {<number>}</code>
--------------------------------------	---

New: 2019-08-18 This returns the object reference of a page object. The `{<number>}` is the absolute page number. The count starts with one.

2 l3pdftools implementation

```

1 <*package>
2 \ProvidesExplPackage {l3pdftools} {2021-01-09} {0.2}
3 {various PDF tools}
4 <@@=pdf>

```

2.1 Conversions and export functions

```

\pdf_text_convert:nn
\pdf_text_convert:nV
\pdf_text_convert:nnN
\__pdf_text_convert_name:n
\__pdf_text_convert_name:nN
\__pdf_text_convert_name-print:n
\__pdf_text_convert_name-print:n

```

```

5
6 \cs_new:Npn \pdf_text_convert:nn #1 #2
7 {
8   \cs_if_exist_use:cF { __pdf_text_convert_#1:n }
9   {
10     \__kernel_msg_error:nnn { pdf } { unknown-exp-convert } {#1}
11     \use_none:n
12   }
13 { #2 }

```

```

14 }
15 \cs_generate_variant:Nn \pdf_text_convert:nn {nV}
16
17 \cs_new:Npn \pdf_text_convert:nnN #1 #2 #3
18 {
19   \cs_if_exist_use:cF { __pdf_text_convert_#1:nN }
20   {
21     \__kernel_msg_error:nnn { pdf } { unknown-convert } {#1}
22     \use_none:nn
23   }
24   { #2 } #3
25 }
26
27 \cs_generate_variant:Nn \pdf_text_convert:nnN {nVN}
28 %% tool command escape name
29 %% commands in the argument are expanded if possible
30 %% with \text_expand:n
31 %% name: only name,
32 %% name_print: with / before, see also string naming
33
34 \cs_generate_variant:Nn \str_convert_pdfname:n { e }
35
36 \cs_new:Npn \__pdf_text_convert_name:n #1
37 {
38   \str_convert_pdfname:e { \text_expand:n { #1 } }
39 }
40
41 \cs_new_protected:Npn \__pdf_text_convert_name:nN #1 #2
42 {
43   \tl_set:Nx #2 {\__pdf_text_convert_name:n {#1} }
44 }
45
46 \cs_new:cpn { __pdf_text_convert_name-print:n } #1
47 {
48   / \str_convert_pdfname:e { \text_expand:n { #1 } }
49 }
50
51 \cs_new_protected:cpn { __pdf_text_convert_name-print:nN } #1 #2
52 {
53   \tl_set:Nx #2 {\use:c { __pdf_text_convert_name-print:n } {#1} }
54 }
55

```

(End definition for `\pdf_text_convert:nn` and others. These functions are documented on page 1.)

The convert command must use a different value the source encoding depending on the engines. Until the PR in str-convert is active we add the alias here too

```

56 \bool_lazy_any:nTF
57 {
58   \sys_if_engine luatex_p:
59   \sys_if_engine xetex_p:
60 }
61 {
62   \prop_gput:Nnn \g__str_alias_prop { default } { }
63 }

```

```

64 {
65   \prop_gput:Nnn \g__str_alias_prop { default } { utf8 }
66 }

```

Most converter are simply wrapper around the str-convert commands and so use the same name. The exception is the one for url's: it reverts most of the percent encodings and escapes the parentheses. That's why its name is URIpdf instead of url. The current code is probably quite slow and will need a replacement.

```

__pdf_text_convert_utf8/string:nN
__pdf_text_convert_utf8/string-print:nN
__pdf_text_convert_utf8/URIpdf:nN
__pdf_text_convert_utf8/URIpdf-print:nN
__pdf_text_convert_utf16/string:nN
__pdf_text_convert_utf16/string-print:nN
__pdf_text_convert_utf16/hex:nN
__pdf_text_convert_utf16/hex-print:nN

67 %% TODO Names need a review when it is clear which converters
68 %% are actually needed
69 %% string conversions and printing
70 %% we assume here that the text purify step has been done. The input is
71 %% a list of (utf8) chars.
72 %% str convert, not expandable.
73 % filespec (attachment view) tests:
74 % utf8: gr\303\274\303\237e.txt
75 % %doesn't work, umlaut wrong,
76 % utf8 with BOM \357\273\277gr\303\274\303\237e.txt
77 % %doesn't work, umlaut wrong, bom visible
78 % utf16 with BE: (FEFF)
79 % \376\377\000g\000r\000\374\000\337\000e\000.\000t\000x\000t %works
80 % xetex converts to <feff0067007200fc00df0065002e007400780074>
81 % utf16 with BE / HEX: <FEFF0067007200FC00DF0065002E007400780074> works
82
83 % bookmarks: as pdfoutline uses () currently only utf16 with BE is usable.
84 % check if one can use HEX too when directly writing the object
85 % =====
86 % uri: utf16BE/string seems not to work, hex neither
87 % utf8/string works but not on macos,
88 % so a specific utf8/url variant is needed
89 % =====
90 % "input" is utf8 for pdftex, empty (native) for unicode engine
91 % commands to output literal strings (...)
92
93 \cs_new_protected:cpn { __pdf_text_convert_utf8/string:nN } #1 #2
94 {
95   \str_set_convert:Nnnn #2
96   { #1 }
97   { default }
98   {utf8/string}
99 }
100
101 \cs_new_protected:cpn { __pdf_text_convert_utf8/string-print:nN } #1 #2
102 {
103   \use:c { __pdf_text_convert_utf8/string:nN } { #1 } #2
104   \str_put_left:Nn #2 { ( }
105   \str_put_right:Nn #2 { ) }
106 }
107 % special url command:
108 \cs_new_protected:cpx { __pdf_text_convert_utf8/URIpdf:nN } #1 #2
109 {
110   \exp_not:N \str_set_convert:Nnnn #2
111   { #1 }

```

```

112     { default }
113     {utf8/url}
114     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 3A} {:}
115     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 2F} {/}
116     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 23} {\c_hash_str}
117     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 5B} {[}
118     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 5D} {]}
119     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 40} {\c_atsign_str}
120     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 21} {!}
121     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 24} {\c_dollar_str}
122     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 26} {\c_ampersand_str}
123     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 27} {'}
124     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 2A} {*}
125     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 2B} {+}
126     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 2C} {,}
127     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 3B} {;}
128     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 3D} {=}
129     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 30} {0}
130     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 31} {1}
131     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 32} {2}
132     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 33} {3}
133     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 34} {4}
134     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 35} {5}
135     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 36} {6}
136     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 37} {7}
137     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 38} {8}
138     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 39} {9}
139     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 28} {\c_backslash_str}
140     \exp_not:N \str_replace_all:Nnn #2 {\c_percent_str 29} {\c_backslash_str}
141 }
142
143 \cs_new_protected:cpn { __pdf_text_convert_utf8/URIpdf-print:nN } #1 #2
144 {
145     \use:c { __pdf_text_convert_utf8/URIpdf:nN } {#1} #2
146     \str_put_left:Nn #2 {()}
147     \str_put_right:Nn #2 {}
148 }
149 % with utf16 with BE marker
150 \cs_new_protected:cpn { __pdf_text_convert_utf16/string:nN } #1 #2
151 {
152     \str_set_convert:Nnnn #2
153     { #1 }
154     { default }
155     {utf16/string}
156 }
157
158 \cs_new_protected:cpn { __pdf_text_convert_utf16/string-print:nN } #1 #2
159 {
160     \use:c { __pdf_text_convert_utf16/string:nN } {#1} #2
161     \str_put_left:Nn #2 {()}
162     \str_put_right:Nn #2 {}
163 }
164
165 % commands to output hex strings (...)

```

```

166
167 \cs_new_protected:cpn { __pdf_text_convert_utf16/hex:nN } #1 #2
168 {
169     \str_set_convert:Nnnn #2
170     { #1 }
171     { default }
172     { utf16/hex }
173 }
174
175 \cs_new_protected:cpn { __pdf_text_convert_utf16/hex-print:nN } #1 #2
176 {
177     \use:c { __pdf_text_convert_utf16/hex:nN } {#1} #2
178     \str_put_left:Nn #2 {<}
179     \str_put_right:Nn #2 {>}
180 }
181

```

(End definition for __pdf_text_convert_utf8/string:nN and others.)

2.1.1 BDC operator commands

```

\pdf_bdc:nn
\pdf_bdc_obj:nn
\pdf_bdc_obj:n
\pdf_bmc:n
\pdf_emc:
182 <*package>
183 \cs_new_protected:Npn \pdf_bdc:nn #1 #2 { \__pdf_backend_bdc:nn { #1 } { #2 } }
184 \cs_new_protected:Npn \pdf_bdc_obj:nn #1 #2 { \__pdf_backend_bdc_obj:nn { #1 } { #2 } }
185 \cs_new_protected:Npn \pdf_bdc_obj:n #1 { \__pdf_backend_bdc_obj:n { #1 } }
186 \cs_new_protected:Npn \pdf_bmc:n #1 { \__pdf_backend_bmc:n { #1 } }
187 \cs_new_protected:Npn \pdf_emc: { \__pdf_backend_emc: }
188 </package>

```

(End definition for \pdf_bdc:nn and others. These functions are documented on page 2.)

```

189 </package>

```

2.2 Reference to the page object

```

\pdf_object_pageref:n
190 <*package>
191 \cs_new:Npn \pdf_object_pageref:n #1 { \__pdf_backend_pageobject_ref:n { #1 } }
192 </package>

```

(End definition for \pdf_object_pageref:n. This function is documented on page 3.)

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

Numbers	B
\0 79	bool commands:
\2 74, 76	
\3 74, 76, 79	\bool_lazy_any:nTF 56

C

cs commands:

\cs_generate_variant:Nn ... 15, 27, 34
 \cs_if_exist_use:NTF 8, 19
 \cs_new:Npn 6, 17, 36, 46, 191
 \cs_new_protected:Npn
 41, 51, 93, 101, 143, 150,
 158, 167, 175, 183, 184, 185, 186, 187
 \cs_new_protected:Npx 108

E

exp commands:

\exp_not:N
 110, 114, 115, 116, 117, 118,
 119, 120, 121, 122, 123, 124, 125,
 126, 127, 128, 129, 130, 131, 132,
 133, 134, 135, 136, 137, 138, 139, 140

K

kernel internal commands:

__kernel_msg_error:nnn 10, 21

P

pdf commands:

\pdf_bdc:nn 2, 3, 182, 183
 \pdf_bdc_obj:.. 2
 \pdf_bdc_obj:n 3, 182, 185
 \pdf_bdc_obj:nn 2, 3, 182, 184
 \pdf_bmc:n 3, 182, 186
 \pdf_emc: 3, 182, 187
 \pdf_object_new:nn 2
 \pdf_object_now:nn 3
 \pdf_object_pageref:n ... 3, 190, 191
 \pdf_object_write:n 2
 \pdf_text_convert:nn 1, 5, 6, 15
 \pdf_text_convert:nnN ... 2, 5, 17, 27

pdf internal commands:

__pdf_backend_bdc:nn 183
 __pdf_backend_bdc_obj:n 185
 __pdf_backend_bdc_obj:nn 184
 __pdf_backend_bmc:n 186
 __pdf_backend_emc: 187
 __pdf_backend_pageobject_ref:n 191
 __pdf_text_convert_name-print:n . 5
 __pdf_text_convert_name:n . 5, 36, 43
 __pdf_text_convert_name:nN .. 5, 41
 __pdf_text_convert_utf16/hex-print:nN
 67
 __pdf_text_convert_utf16/hex:nN . 67
 __pdf_text_convert_utf16/string-print:nN
 67
 __pdf_text_convert_utf16/string:nN
 67

__pdf_text_convert_utf8/string-print:nN

..... 67

__pdf_text_convert_utf8/string:nN

..... 67

__pdf_text_convert_utf8/URIpdf-print:nN

..... 67

__pdf_text_convert_utf8/URIpdf:nN

..... 67

prop commands:

\prop_gput:Nnn 62, 65

\ProvidesExplPackage 2

S

str commands:

\c_ampersand_str 122
 \c_atsign_str 119
 \c_backslash_str 139, 140
 \c_dollar_str 121
 \c_hash_str 116
 \c_percent_str
 114, 115, 116, 117, 118,
 119, 120, 121, 122, 123, 124, 125,
 126, 127, 128, 129, 130, 131, 132,
 133, 134, 135, 136, 137, 138, 139, 140
 \str_convert_pdfname:n .. 1, 34, 38, 48
 \str_put_left:Nn ... 104, 146, 161, 178
 \str_put_right:Nn .. 105, 147, 162, 179
 \str_replace_all:Nnn
 114, 115, 116, 117, 118,
 119, 120, 121, 122, 123, 124, 125,
 126, 127, 128, 129, 130, 131, 132,
 133, 134, 135, 136, 137, 138, 139, 140
 \str_set_convert:Nnn
 2, 95, 110, 152, 169

str internal commands:

\g_str_alias_prop 62, 65

sys commands:

\sys_if_engine luatex_p: 58

\sys_if_engine xetex_p: 59

T

text commands:

\text_expand:n 1, 30, 38, 48

tl commands:

\tl_set:Nn 43, 53

U

use commands:

\use:N 53, 103, 145, 160, 177

\use_none:n 11

\use_none:nn 22