

The `ltdocinit` module

The L^AT_EX Project*

Version 0.95g, released 2021-07-21

1 `ltdocinit` documentation

This small module defines `\DeclareDocumentMetadata` and the related keys. It also defines commands to store document properties in a global container.

1.1 `\DeclareDocumentMetadata`

<code>\DeclareDocumentMetadata</code>	<code>\DeclareDocumentMetadata{<key-value list>}</code>
---------------------------------------	---

Currently there is no dedicated location to declare settings concerning a document as a whole. Settings are placed somewhere in the preamble or with the class options or even with some package options. For some settings this can be too late, for example the pdf version can no longer be changed if a package has used code which already opened the PDF.

`\DeclareDocumentMetadata` as a new command should unify such settings in one place. It should be used before `\documentclass`, directly after loading `pdfmanagement-testphase`. `\DeclareDocumentMetadata` can be used more than once.

The keys defined for `\DeclareDocumentMetadata` currently allows to set the PDF version, to set the PDF `/Lang`, to uncompress a pdf, to set the language and to declare a few PDF standards and some colorprofiles.

`\DeclareDocumentMetadata` is also used to activate the new PDF management code and it loads a number of required files for the PDF management code. As this forces the loading of the backend files, a backend which can't be detected automatically like `dvipdfmx`, must be set in the first `\DeclareDocumentMetadata`.

Currently the following keys are implemented

backend passes the backend name to `expl3`. This will probably be extended to pass the value also to packages.

pdfversion e.g. `pdfversion=1.7`

uncompress no value. Forces an uncompressed pdf.

lang to set the `Lang` entry in the Catalog. E.g. `lang=de-DE`. The initial value is `en-US`

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

pdfstandard Choice key to set the pdf standard. Currently A-1b, A-2b, A-2u, A-3b and A-3u are accepted as values. The casing is irrelevant, a-1b works too. The underlying code to ensure the requirements (as far as they can be ensured) is incomplete, but a color profile is included and the /OutputIntent is set. The u variants for example do not force unicode, but they will pass the information to hyperref and hyperxmp. More information can be found in the documentation of l3pdfmeta.

colorprofiles This allows to load icc-colorprofiles. Details are described in the documentation of l3pdfmeta.

pdfmanagement Boolean. This activates/deactivates the core management code. By default the value is true.

firstaidoff This accepts a comma lists of keyword and disable the patches related to them. More information can be found in the documentation of pdfmanagement-firstaid.

testphase This key is used to load testphase code. The values it accepts and their effect will change over time, when testphase packages are added or removed when the code is moved into the kernel. Currently the accepted values are **tagpdf**, this load the tagpdf package, **headings**, this loads code which reimplements heading commands, and **ptagging** this loads code to allow paragraph tagging to work with engine other than luatex.

activate This key is used to enable some document wide functions. It is currently in an experimental state. The values and their behaviour are subject to change. Currently the only value is **tagging**, which will do `\tagpdfsetup{activate,paratagging,interwords}`. It requires that **tagpdf** has been loaded first with the **testphase** key.

debug This key activates some debug options. Currently only the keys **para** (with the default and only value **show**), and **log** (with the values of **tagpdf**) and **uncompress** (which does the same as **uncompress** as main key) are known.

1.2 Container for document properties

The module provides a container where classes, packages and users can store properties of the document which are perhaps of interest or use for other packages or the author.

The properties are stored with a key **label/property**. The values can be retrieved expandably.

\AddToDocumentProperties	<code>\AddToDocumentProperties[\langle label \rangle]{\langle property \rangle}{\langle value \rangle}</code>
---------------------------------	---

This stores $\langle value \rangle$ under the key $\langle label \rangle/\langle property \rangle$. By default $\langle label \rangle$ is the current package name `\@currname`. If another label is choosen, it should be one which avoids clashes with other packages using the container. The label **document** is reserved.

\GetDocumentProperties	<code>\GetDocumentProperties{\langle label/property \rangle}</code>
-------------------------------	---

Expands to the $\langle value \rangle$ corresponding to $\langle label/property \rangle$ in the container. If $\langle label/property \rangle$ is missing, this has an empty expansion. The result is returned within `\exp_not:n`, which means that the $\langle value \rangle$ does not expand further when appearing in an x-type argument expansion.

`\ShowDocumentProperties` `\ShowDocumentProperties`

This show the current content of the container.

2 ltdocinit implementation

2.1 `\DeclareDocumentMetadata`

```
1 <@@=pdfmanagement>
2 <*header>
3 \ProvidesExplPackage{ltdocinit}{2021-07-21}{0.95g}
4   {Initialize document metadata}
5 </header>
```

`\DeclareDocumentMetadata` should for now not be used after `\documentclass` so we error in this case. It can be used more than once but follow-up calls should not do the initialization code.

```
6 <*package>
7 \msg_new:nnn { document } { setup-after-documentclass }
8   {
9     \token_to_str:N \DeclareDocumentMetadata \c_space_tl
10    should-be-used-only-before~\token_to_str:N\documentclass
11  }
```

`\DeclareDocumentMetadata`

```
12 \NewDocumentCommand\DeclareDocumentMetadata { m }
13   {
14     \cs_if_eq:NNTF \documentclass \@twoclasseserror
15     { \msg_error:nn { document } { setup-after-documentclass } }
16     {
```

The wanted backend must be detected first, we read the init key and then force the loading by either loading l3pdf (+ expl3) or only expl3. The backend contains now management command, so the boolean should true first.

```
17     \keys_set_groups:nnn { document / metadata } {init}{ #1 }
18     %should be loaded after the backend is set, and only if not in the kernel
19     \bool_gset_true:N \g__pdfmanagement_active_bool
20     \cs_if_free:NNTF \pdf_uncompress:
21     { \RequirePackage{l3pdf} } { \RequirePackage{expl3} }
```

Now we load the extra backend code

```
22     \ExplSyntaxOn\makeatletter
23     \file_input:n {l3backend-testphase-\c_sys_backend_str.def}
24     \ExplSyntaxOff\makeatother
```

set the default language, process the rest of the keys, and setup the generic driver

```
25     \keys_set_filter:nnn { document / metadata } { init } { lang=en-US, #1 }
26     \bool_if:NT \g__pdfmanagement_active_bool
27     {
28       \PassOptionsToPackage{customdriver=hgeneric-testphase}{hyperref}
29     }
```

`\pdfmanagement_add:nnn` has collected values in this hook.

```
30     \hook_use_once:n {pdfmanagement/add}
```

Redefine `\DeclareDocumentMetadata` so that it only process the keys. We need to update the `hyperref` option if the active status changes.

```

31 \RenewDocumentCommand\DeclareDocumentMetadata { m }
32 {
33   \keys_set_filter:nnn { document / metadata } { init } { ##1 }
34   \bool_if:NTF \g__pdfmanagement_active_bool
35   {
36     \str_remove_all:cn {opt@hyperref.sty}{customdriver=hgeneric-testphase}
37     \PassOptionsToPackage{customdriver=hgeneric-testphase}{hyperref}
38   }
39   {
40     \str_remove_all:cn {opt@hyperref.sty}{customdriver=hgeneric-testphase}
41   }
42 }

```

Load more modules, the testphase code and the firstaid code.

```

43 \g__pdfmanagement_testphase_tl
44 \RequirePackage{pdfmanagement-firstaid}
45 }
46 }

```

(End definition for `\DeclareDocumentMetadata`. This function is documented on page 1.)

2.2 Container for document Properties

The container for the document properties is a prop

`\g__pdfmanagement_documentproperties_prop`

```

47 \prop_new:N \g__pdfmanagement_documentproperties_prop %

```

(End definition for `\g__pdfmanagement_documentproperties_prop`.)

`\AddToDocumentProperties`

```

48 \NewDocumentCommand\AddToDocumentProperties{0{\@currname}mm}
49 {
50   \exp_args:NNx
51   \prop_gput:Nnn \g__pdfmanagement_documentproperties_prop
52   {
53     \tl_if_blank:eTF {#1}{top-level/}{#1/} #2
54   }
55   { #3}
56 }

```

(End definition for `\AddToDocumentProperties`. This function is documented on page 2.)

`\GetDocumentProperties`

```

57 \NewExpandableDocumentCommand\GetDocumentProperties{m}
58 {
59   \prop_item:Nn \g__pdfmanagement_documentproperties_prop {#1}
60 }

```

(End definition for `\GetDocumentProperties`. This function is documented on page 2.)

`\ShowDocumentProperties`

```
61 \msg_new:nnn { pdfmanagement } { show-properties }
62 {
63   The~following~document~properties~have~been~stored:
64   #1
65 }
66 \NewDocumentCommand\ShowDocumentProperties {}
67 {
68   \msg_show:nnx {pdfmanagement}{show-properties}
69   {
70     \prop_map_function:NN \g__pdfmanagement_documentproperties_prop \msg_show_item:nn
71   }
72 }
```

(End definition for `\ShowDocumentProperties`. This function is documented on page 3.)

2.3 The keys for `\DeclareDocumentMetadata`

As they use `\AddToDocumentProperties` they have to come after it has been defined.

`\g__pdfmanagement_firstaidoff_clist` A list to store the firstaid code which should be disabled

```
73 \clist_new:N \g__pdfmanagement_firstaidoff_clist
```

(End definition for `\g__pdfmanagement_firstaidoff_clist`.)

`\g__pdfmanagement_testphase_tl` a tl to store the testphase loading code so that we can load them at the end of the command.

```
74 \tl_new:N \g__pdfmanagement_testphase_tl
```

(End definition for `\g__pdfmanagement_testphase_tl`.)

```
75 \keys_define:nn { document / metadata }
76 {
77   backend .code:n =
78   {
79     \PassOptionsToPackage { driver=#1 } {expl3}
80     \AddToDocumentProperties[document]{backend}{#1}
81   },
82   backend .groups:n = { init } ,
83 }
84
85 \keys_define:nn { document / metadata }
86 {
87   ,pdfversion .code:n =
88   {
89     \pdf_version_gset:n { #1 }
90     \AddToDocumentProperties[document]{pdfversion}{#1}
91   }
92   ,uncompress .code:n =
93   {
94     \pdf_uncompress:
95   }
96   ,uncompress .value_forbidden:n = true
97   ,lang .code:n =
98   {
```

```

99         \pdfmanagement_add:nnn {Catalog} {Lang}{(#1)}
100         \AddToDocumentProperties [document] {lang}{#1}
101     }
102     %,xmpmeta .bool_gset:N = \g_pdfmeta_xmp_bool %see pdfmeta unused and undefined for now!
103     % this uses internal command from pdfmeta, it should probably move there ...
104     ,pdfstandard .code:n =
105     {
106         \exp_args:Nnx
107         \keys_set:nn {document / metadata} {_pdfstandard=\str_uppercase:n{#1}}
108     }
109     ,_pdfstandard .choices:nn =
110     {A-1B,A-2B,A-2U,A-3B,A-3U}
111     {
112         \prop_if_exist:cT { g__pdfmeta_standard_pdf/#1_prop }
113         {
114             \prop_gset_eq:Nc \g__pdfmeta_standard_prop { g__pdfmeta_standard_pdf/#1 _prop }
115         }
116         \AddToDocumentProperties [document] {pdfstandard}{#1}
117     }
118     ,_pdfstandard / unknown .code:n =
119     {
120         \msg_warning:nnn {pdf} {unknown-standard}{#1}
121     }
122     ,pdfmanagement .bool_gset:N = \g__pdfmanagement_active_bool
123     ,firstaidoff .clist_gset:N = \g__pdfmanagement_firstaidoff_clist
124     ,testphase .multichoice:
125     ,testphase / tagpdf .code:n =
126     {
127         \tl_gput_right:Nn \g__pdfmanagement_testphase_tl
128         {
129             \RequirePackage {tagpdf}
130             \AddToDocumentProperties [document] {testphase/tagpdf} {loaded}
131         }
132     }
133     ,testphase / headings .code:n =
134     {
135         \tl_gput_right:Nn \g__pdfmanagement_testphase_tl
136         {
137             \RequirePackage {headings-testphase}
138             \AddToDocumentProperties [document] {testphase/headings} {loaded}
139         }
140     }
141     ,testphase / ptagging .code:n =
142     {
143         \tl_gput_right:Nn \g__pdfmanagement_testphase_tl
144         {
145             \AddToHook {class/after}
146             {
147                 \RequirePackage {ptagging-testphase}
148                 \AddToDocumentProperties [document] {testphase/ptagging} {loaded}
149             }
150         }
151     }
152     ,testphase / unknown .code:n =

```

```

153 {
154   \tl_gput_right:Nn\g__pdfmanagement_testphase_tl
155   {
156     \AddToHook{class/after}
157     {
158       \RequirePackage{#1-testphase}
159       \AddToDocumentProperties [document]{testphase/#1}{loaded}
160     }
161   }
162 }
163 ,activate .multichoice:
164 ,activate / tagging .code:n =
165 {
166   \AddToHook{package/after/tagpdf}
167   {
168     \tagpdfsetup{activate,paratagging,interwordspace}
169     \AddToDocumentProperties [document]{tagging}{active}
170     \AddToDocumentProperties [document]{tagging/para}{active}
171     \AddToDocumentProperties [document]{tagging/interwordspace}{active}
172   }
173 }
174 ,debug .code:n =
175 {
176   \keys_set:nn { document / metadata / debug } {#1}
177 }
178 ,debug / para .code:n =
179 {
180   \AddToHook{package/after/tagpdf}
181   {
182     \tagpdfsetup{paratagging-show}
183   }
184 }
185 ,debug / log .code:n =
186 {
187   \AddToHook{package/after/tagpdf}
188   {
189     \tagpdfsetup{log=#1}
190   }
191 }
192 ,debug / uncompress .code:n =
193 {
194   \pdf_uncompress:
195 }
196 }
197 \</package>

```

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.

A

`\AddToDocumentProperties`
 2, 5, 48, 80, 90, 100, 7

116, 130, 138, 148, 159, 169, 170, 171	\NewExpandableDocumentCommand 57
\AddToHook 145, 156, 166, 180, 187	
B	P
bool commands:	\PassOptionsToPackage 28, 37, 79
\bool_gset_true:N 19	pdf commands:
\bool_if:NTF 26, 34	\pdf_uncompress: 20, 94, 194
C	\pdf_version_gset:n 89
clist commands:	pdfmanagement commands:
\clist_new:N 73	\pdfmanagement_add:nnn 3, 99
cs commands:	pdfmanagement internal commands:
\cs_if_eq:NNTF 14	\g__pdfmanagement_active_bool 19, 26, 34, 122
\cs_if_free:NTF 20	\g__pdfmanagement_documentproperties_prop 47, 51, 59, 70
D	\g__pdfmanagement_firstaidoff_clist 73, 123
\DeclareDocumentMetadata . . . 1, 3–5, 9, 12	\g__pdfmanagement_testphase_tl 43, 74, 127, 135, 143, 154
\documentclass 1, 3, 10, 14	pdfmeta commands:
E	\g_pdfmeta_xmp_bool 102
exp commands:	pdfmeta internal commands:
\exp_args:NNx 50	\g__pdfmeta_standard_prop 114
\exp_args:Nnx 106	prop commands:
\exp_not:n 2	\prop_gput:Nnn 51
\ExplSyntaxOff 24	\prop_gset_eq:NN 114
\ExplSyntaxOn 22	\prop_if_exist:NTF 112
F	\prop_item:Nn 59
file commands:	\prop_map_function:NN 70
\file_input:n 23	\prop_new:N 47
G	\ProvidesExplPackage 3
\GetDocumentProperties 2, 57	R
H	\RenewDocumentCommand 31
hook commands:	\RequirePackage 21, 44, 129, 137, 147, 158
\hook_use_once:n 30	S
K	\ShowDocumentProperties 3, 61
keys commands:	str commands:
\keys_define:nn 75, 85	\str_remove_all:Nn 36, 40
\keys_set:nn 107, 176	\str_uppercase:n 107
\keys_set_filter:nnn 25, 33	sys commands:
\keys_set_groups:nnn 17	\c_sys_backend_str 23
M	T
\makeatletter 22	\tagpdfsetup 168, 182, 189
\makeatother 24	TeX and L ^A T _E X 2 _ε commands:
msg commands:	\currname 48
\msg_error:nn 15	\@twoclasseserror 14
\msg_new:nnn 7, 61	tl commands:
\msg_show:nnn 68	\c_space_tl 9
\msg_show_item:nn 70	\tl_gput_right:Nn . . . 127, 135, 143, 154
\msg_warning:nnn 120	\tl_if_blank:NTF 53
N	\tl_new:N 74
\NewDocumentCommand 12, 48, 66	token commands:
	\token_to_str:N 9, 10