

The ltdocinit package

The L^AT_EX Project*

Version 0.95a, released 2021-02-22

1 ltdocinit documentation

This small package 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 and A-3b are accepted as values. 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. 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.

1.2 Container for document properties

The package 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	\AddToDocumentProperties [$\langle label \rangle$]{ $\langle property \rangle$ }{ $\langle value \rangle$ }
---------------------------------	--

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	\AddToDocumentProperties { $\langle label/property \rangle$ }
-------------------------------	--

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} {2021-02-22} {0.95a}
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.

```

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         \cs_if_free:NTF \pdf_uncompress:
20         { \RequirePackage{l3pdf} } { \RequirePackage{expl3} }

```

Now we load the extra backend code and set the boolean to true

```

21         \ExplSyntaxOn\makeatletter
22         \file_input:n {l3backend-testphase-\c_sys_backend_str.def}
23         \ExplSyntaxOff\makeatother
24         \bool_gset_true:N \g__pdfmanagement_active_bool

```

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:NTF \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 and the firstaid code.

```

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

```

(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
46 \prop_new:N \g__pdfmanagement_documentproperties_prop %
```

(End definition for \g__pdfmanagement_documentproperties_prop.)

\AddToDocumentProperties

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

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

\GetDocumentProperties

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

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

\ShowDocumentProperties

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

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

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

```
72 \clist_new:N \g__pdfmanagement_firstaidoff_clist
```

(End definition for \g__pdfmanagement_firstaidoff_clist.)

```

73 \keys_define:nn { document / metadata }
74 {
75   backend .code:n =
76   {
77     \PassOptionsToPackage { driver=#1 } {expl3}
78     \AddToDocumentProperties [document] {backend} {#1}
79   },
80   backend .groups:n = { init } ,
81 }
82
83 \keys_define:nn { document / metadata }
84 {
85   ,pdfversion .code:n =
86   {
87     \pdf_version_gset:n { #1 }
88     \AddToDocumentProperties [document] {pdfversion} {#1}
89   }
90   ,uncompress .code:n =
91   {
92     \pdf_uncompress:
93   }
94   ,uncompress .value_forbidden:n = true
95   ,lang .code:n =
96   {
97     \pdfmanagement_add:nnn {Catalog} {Lang} {(#1)}
98     \AddToDocumentProperties [document] {lang} {#1}
99   }
100   % ,xmpmeta .bool_gset:N = \g_pdfmeta_xmp_bool %see pdfmeta unused and undefined for now!
101   % this uses internal command from pdfmeta, it should probably move there ...
102   ,pdfstandard .code:n =
103   {
104     \exp_args:Nnx
105     \keys_set:nn {document / metadata} { _pdfstandard=\str_uppercase:n{#1}}
106   }
107   ,_pdfstandard .choices:nn =
108   {A-1B,A-2B,A-3B}
109   {
110     \prop_if_exist:cT { g__pdfmeta_standard_pdf/#1_prop }
111     {
112       \prop_gset_eq:Nc \g__pdfmeta_standard_prop { g__pdfmeta_standard_pdf/#1 _prop }
113     }
114     \AddToDocumentProperties [document] {pdfstandard} {#1}
115   }
116   ,_pdfstandard / unknown .code:n =
117   {
118     \msg_warning:nnn {pdf} {unknown-standard} {#1}
119   }
120   ,pdfmanagement .bool_gset:N = \g__pdfmanagement_active_bool
121   ,pdfmanagement .initial:n = {true}
122   ,firstaidoff .clist_gset:N = \g__pdfmanagement_firstaidoff_clist
123 }
124 </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, 2, 4, <u>47</u> , 78, 88, 98, 114
B	
bool commands:	
\bool_gset_true:N	24
\bool_if:NTF	26, 34
C	
clist commands:	
\clist_new:N	72
cs commands:	
\cs_if_eq:NNTF	14
\cs_if_free:NTF	19
D	
\DeclareDocumentMetadata	1, 1, 2, 2, 3, 4, 9, <u>12</u>
\documentclass	1, 2, 10, 14
E	
exp commands:	
\exp_args:NNx	49
\exp_args:Nnx	104
\exp_not:n	2
\ExplSyntaxOff	23
\ExplSyntaxOn	21
F	
file commands:	
\file_input:n	22
G	
\GetDocumentProperties	2, <u>56</u>
H	
hook commands:	
\hook_use_once:n	30
K	
keys commands:	
\keys_define:nn	73, 83
\keys_set:nn	105
\keys_set_filter:nnn	25, 33
\keys_set_groups:nnn	17
M	
\makeatletter	21
N	
\NewDocumentCommand	12, 47, 65
\NewExpandableDocumentCommand	56
P	
\PassOptionsToPackage	28, 37, 77
pdf commands:	
\pdf_uncompress:	19, 92
\pdf_version_gset:n	87
pdfmanagement commands:	
\pdfmanagement_add:nnn	3, 97
pdfmanagement internal commands:	
\g_pdfmanagement_active_bool	24, 26, 34, 120
\g_pdfmanagement_documentproperties_prop	46, 50, 58, 69
\g_pdfmanagement_firstaidoff_clist	72, 122
pdfmeta commands:	
\g_pdfmeta_xmp_bool	100
pdfmeta internal commands:	
\g_pdfmeta_standard_prop	112
prop commands:	
\prop_gput:Nnn	50
\prop_gset_eq:NN	112
\prop_if_exist:NTF	110
\prop_item:Nn	58
\prop_map_function:NN	69
\prop_new:N	46
\ProvidesExplPackage	3
R	
\RenewDocumentCommand	31
\RequirePackage	20, 43
S	
\ShowDocumentProperties	2, <u>60</u>
str commands:	
\str_remove_all:Nn	36, 40
\str_uppercase:n	105
\makeatother	
msg commands:	
\msg_error:nn	15
\msg_new:nnn	7, 60
\msg_show:nnn	67
\msg_show_item:nn	69
\msg_warning:nnn	118

sys commands:		\@twoclasseserror	14
\c_sys_backend_str	22	tl commands:	
		\c_space_tl	9
		\tl_if_blank:nTF	52
T		token commands:	
TeX and L ^A T _E X 2 _ε commands:		\token_to_str:N	9, 10
\@currname	47		