The ltdocinit module

The LaTeX Project*

Version 0.95j, released 2022-01-13

1 ltdocinit documentation

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

When the kernel will provide `\DocumentMetadata` directly this module will slowly disappear.

1.1 `\DocumentMetadata`/`\DeclareDocumentMetadata`

`\DocumentMetadata` `\DocumentMetadata{`⟨*key-value list*⟩`}`
`\DeclareDocumentMetadata` `\DeclareDocumentMetadata{`⟨*key-value list*⟩`}` (deprecated)

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.

`\DocumentMetadata` as a new command should unify such settings in one place. It should be used before `\documentclass`. `\DocumentMetadata` can be used more than once.

With a LaTeX 2022-06-01 or newer (currently provided as `latex-dev`) the command is provided by the kernel and will load the pdfmanagement, with older LaTeX the pdfmanagement must be loaded first manually with `\RequirePackage{pdfmanagement-testphase}`.

The keys defined for `\DocumentMetadata` 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 to load some colorprofiles.

`\DocumentMetadata` 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 `\DocumentMetadata`.

Currently the following keys are implemented

backend Passes the backend name to expl3. This is needed only if the needed backend can't be automatically determined or if the workflow used requires a special backend.

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

pdfversion Sets the PDF version explicitly, e.g., `pdfversion=1.7`

uncompress (no value) Forces an uncompressed pdf — mainly for debugging purposes.

lang Explicitly sets the Lang entry in the Catalog, e.g., `lang=de-DE`. If not given the default value used is `en-US`.

pdfstandard Choice key to set the pdf standard. Currently `A-1b`, `A-2a`, `A-2b`, `A-2u`, `A-3a`, `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 still 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. The `a` variants do *not* enforce (or even test) a tagged pdf yet. 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.

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 or when the code is moved into the kernel.

- **phase-I** This value loads code implementing the first phase of the project [1], i.e., it will load the tagpdf package. It will also activate tagging by issuing `\tagpdfsetup{activate,interwordspace}`. This phase is frozen.
- **phase-II** This is the current development phase. Currently it differs from `phase-I` only in one point: It will additionally activate tagging of paragraphs with `\tagpdfsetup{paratagging}`. In the upcoming months it will also enable automatic tagging of other basic document elements.

The `testphase` key can only be used in the first `\DocumentMetadata`.

debug This key activates some debug options. It takes a list of key-values as value. Currently the following keys are known:

- **para** with the default and only value `show`. It will activate the `paratagging-show` option of tagpdf,
- **log** with the values as described in the documentation tagpdf,
- **uncompress** which does the same as `uncompress` as main key
- **pdfmanagement** a boolean which allows to deactivate the pdfmanagement.
- **firstaidoff** This accepts a comma lists of keywords and disables the patches related to them. More information can be found in the documentation of pdfmanagement-firstaid.

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 `\AddToDocumentProperties[⟨label⟩]{⟨property⟩}{⟨value⟩}`

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

\GetDocumentProperties `\GetDocumentProperties{⟨label/property⟩}`

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

\ShowDocumentProperties `\ShowDocumentProperties`

This show the current content of the container.

References

[1] Frank Mittelbach and Chris Rowley: *LaTeX Tagged PDF — A blueprint for a large project*. https://latex-project.org/publications/indexbyyear/2020/

2 ltdocinit implementation

```
⟨@@=pdfmanagement⟩
⟨*header⟩
\ProvidesExplPackage{ltdocinit}{2022-01-13}{0.95j}
  {Initialize document metadata}
⟨/header⟩
```

`\DocumentMetadata` will be defined by the kernel in short time. So we define it and the keys here only if it is not already defined.

2.1 The keys for `\DocumentMetadata`

We define the keys first so that we can test if `\DocumentMetadata` exist (testing for the format date would be nice but would fail for the current latex-dev).

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

```
⟨*package⟩
\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.

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

(*End definition for* `\g__pdfmanagement_testphase_tl`.)

```
\cs_if_free:NT \DocumentMetadata
  {
    \keys_define:nn { document / metadata }
      {
        backend .choices:nn =
```

```
        { dvipdfmx , dvips , dvisvgm , luatex , pdftex , pdfmode , xdvipdfmx , xetex }
        {
          \sys_load_backend:n {#1}
        },
      backend .groups:n = { init } ,
    }

  \keys_define:nn { document / metadata }
    {
      ,pdfversion .code:n =
        {
          \pdf_version_gset:n { #1 }
          \AddToDocumentProperties[document]{pdfversion}{#1}
        }
      ,uncompress .code:n =
        {
          \pdf_uncompress:
        }
      ,uncompress .value_forbidden:n = true
      ,lang .code:n =
        {
          \pdfmanagement_add:nnn {Catalog} {Lang}{(#1)}
          \AddToDocumentProperties[document]{lang}{#1}
        }
      %,xmpmeta .bool_gset:N = \g_pdfmeta_xmp_bool %see pdfmeta unused and undefined for no
      % this uses internal command from pdfmeta, it should probably move there ...
      ,pdfstandard .code:n =
        {
          \exp_args:Nnx
          \keys_set:nn {document / metadata} {_pdfstandard=\str_uppercase:n{#1}}
        }
      ,_pdfstandard .choices:nn =
        {A-1B,A-2A,A-2B,A-2U,A-3A,A-3B,A-3U}
        {
          \prop_if_exist:cT { g__pdfmeta_standard_pdf/#1_prop }
            {
              \prop_gset_eq:Nc \g__pdfmeta_standard_prop { g__pdfmeta_standard_pdf/#1 _prop
            }
          \AddToDocumentProperties [document]{pdfstandard}{#1}
        }
      ,_pdfstandard / unknown .code:n =
        {
          \msg_warning:nnn{pdf}{unknown-standard}{#1}
        }
      ,testphase .multichoice:
      ,testphase / tagpdf .code:n =
        {
          \tl_gput_right:Nn\g__pdfmanagement_testphase_tl
            {
              \file_if_exist_input:nF {tagpdf-latex-lab-testphase.ltx}
                {
                  \RequirePackage{tagpdf}
                  \AddToDocumentProperties [document]{testphase/tagpdf}{loaded}
                  \tagpdfsetup{activate,paratagging,interwordspace}
```

```
                \AddToDocumentProperties [document]{tagging}{active}
                \AddToDocumentProperties [document]{tagging/para}{active}
                \AddToDocumentProperties [document]{tagging/interwordspace}{active}
              }
          }
      }
    ,testphase / unknown .code:n =
      {
        \tl_gput_right:Nn\g__pdfmanagement_testphase_tl
           {
             \file_if_exist_input:nF {#1-latex-lab-testphase.ltx}
              {
                 \msg_warning:nnn{document}{latex-lab-pkg-missing}{#1}
              }
           }
      }
    ,activate .multichoice:
    ,activate / tagging .code:n =
      {
        \PackageWarning{pdfmanagement-testphase}
         {The~activate~key~is~deprecated.\MessageBreak
          Tagging~is~activated~with~'testphase=tagpdf'~directly}{}
      }
    ,debug .code:n =
      {
        \keys_set:nn { document / metadata / debug } {#1}
      }
    ,debug / para .code:n =
      {
        \AddToHook
          {
            package/tagpdf/after
          }
          {
             \tagpdfsetup{paratagging-show}
          }
      }
    ,debug / log .code:n =
      {
        \AddToHook
          {
           package/tagpdf/after
          }
          {
             \tagpdfsetup{log=#1}
          }
      }
    ,debug / uncompress .code:n =
      {
        \pdf_uncompress:
      }
    ,debug / pdfmanagement .bool_gset:N = \g__pdfmanagement_active_bool
    ,debug / firstaidoff .clist_gset:N = \g__pdfmanagement_firstaidoff_clist
  }
```

```
}
```

2.2 \DocumentMetadata/\DeclareDocumentMetadata

At first we have to define the older alias `\DeclareDocumentMetadata` if `\DocumentMetadata` is already defined in a newer LaTeX version. Here we load the latex-lab support (if it hasn't been loaded before) and redefine then `\DeclareDocumentMetadata` to `\DocumentMetadata`

```
\cs_if_exist:NT \DocumentMetadata
  {
    \cs_new_protected:Npn \DeclareDocumentMetadata
       {
         \cs_if_free:NT \DocumentMetadata
           {
             \file_if_exist_input:nF {documentmetadata-support.ltx}%
              {
                \@latex@error
                  {No~support~files~for~\noexpand\DeclareDocumentMetadata~found}
                  {
                    Is~the~'LaTeX-lab'~bundle~installed?
                    \MessageBreak
                    Without~it,~the~declaration~is~ignored.
                  }
               \let\DeclareDocumentMetadata\@gobble
              }
           }
         \RenewDocumentCommand \DeclareDocumentMetadata {}
          { \DocumentMetadata }
         \DocumentMetadata
       }
   }
```

Now we provide a definition for `\DocumentMetadata` if it doesn't exist yet.

```
\cs_if_free:NT \DocumentMetadata
  {
```

`\DocumentMetadata` 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.

```
    \prop_gput:Nnn \g_msg_module_type_prop { document } { LaTeX }
    \prop_gput:Nnn \g_msg_module_name_prop { document } { DocumentMetadata }
    \msg_new:nnn
      { document } { setup-after-documentclass }
      {
        \token_to_str:N \DocumentMetadata \c_space_tl
        should~be~used~only~before~\token_to_str:N\documentclass
      }
    \msg_new:nnn
      { document } { latex-lab-pkg-missing }
      {
        LaTeX-lab~package~'#1'~not~found.
      }
```

\DocumentMetadata

```
    \NewDocumentCommand\DocumentMetadata { m }
      {
        \cs_if_eq:NNTF \documentclass \@twoclasseserror
          { \msg_error:nn { document }{ setup-after-documentclass } }
          {
```

The wanted backend must be detected first, we read the init key and then force the loading by loading expl3. The backend can contains management commands, so the boolean should be set to true first.

```
            \bool_gset_true:N \g__pdfmanagement_active_bool
            \keys_set_groups:nnn { document / metadata} {init}{ #1 }
        %if no backend has been loaded force it now:
            \str_if_exist:NF \c_sys_backend_str
              {
                \sys_load_backend:n {}
              }
```

Now we load the extra backend code

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

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

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

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

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

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

```
            \RenewDocumentCommand\DocumentMetadata { m }
              {
                \keys_set_filter:nnn  { document / metadata } { init } { ##1 }
                \bool_if:NTF \g__pdfmanagement_active_bool
                  {
                    \str_remove_all:cn
                      {opt@hyperref.sty}{customdriver=hgeneric-testphase}
                    \PassOptionsToPackage
                      {customdriver=hgeneric-testphase}{hyperref}
                  }
                  {
                    \str_remove_all:cn
                      {opt@hyperref.sty}{customdriver=hgeneric-testphase}
                  }
              }
```

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

```
            \g__pdfmanagement_testphase_tl
            \RequirePackage{pdfmanagement-firstaid}
          }
      }
```

At last we need to provide the older alias here too

```
    \NewDocumentCommand\DeclareDocumentMetadata{}{\DocumentMetadata}
  }
```

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

2.3 Container for document Properties

The container for the document properties is a prop

`\g__pdfmanagement_documentproperties_prop`

```
\prop_new:N \g__pdfmanagement_documentproperties_prop %
```

(*End definition for* `\g__pdfmanagement_documentproperties_prop`.)

`\AddToDocumentProperties`

```
\NewDocumentCommand\AddToDocumentProperties{O{\@currname}mm}
  {
    \exp_args:NNx
      \prop_gput:Nnn \g__pdfmanagement_documentproperties_prop
        {
          \tl_if_blank:eTF {#1}{top-level/}{#1/} #2
        }
        { #3}
  }
```

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

`\GetDocumentProperties`

```
\NewExpandableDocumentCommand\GetDocumentProperties{m}
  {
    \prop_item:Nn \g__pdfmanagement_documentproperties_prop {#1}
  }
```

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

`\ShowDocumentProperties`

```
\msg_new:nnn  { pdfmanagement } { show-properties }
  {
    The~following~document~properties~have~been~stored:
    #1
  }
\NewDocumentCommand\ShowDocumentProperties {}
  {
    \msg_show:nnx {pdfmanagement}{show-properties}
      {
        \prop_map_function:NN \g__pdfmanagement_documentproperties_prop \msg_show_item:nn
      }
  }
```

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

```
⟨/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 *3*, 26, 36, 52, 66, 68, 69, 70, 204
\AddToHook 97, 107

B

bool commands:
\bool_gset_true:N 166
\bool_if:NTF 177, 185

C

clist commands:
\clist_new:N 7
cs commands:
\cs_if_eq:NNTF 163
\cs_if_exist:NTF 123
\cs_if_free:NTF 9, 127, 146
\cs_new_protected:Npn 125

D

\DeclareDocumentMetadata *1*, *6*, 125, 132, 138, 141, 201
\documentclass *1*, *6*, 154, 163
\DocumentMetadata *1–3*, *6*, *7*, 9, 123, 127, 142, 143, 146, 153, 161

E

exp commands:
\exp_args:NNx 206
\exp_args:Nnx 42
\exp_not:n *3*
\ExplSyntaxOff 175
\ExplSyntaxOn 173

F

file commands:
\file_if_exist_input:nTF . 63, 78, 129
\file_input:n 174

G

\GetDocumentProperties *3*, 213

H

hook commands:
\hook_use_once:n 181

K

keys commands:
\keys_define:nn 11, 21
\keys_set:nn 43, 93
\keys_set_filter:nnn 176, 184
\keys_set_groups:nnn 167

L

\let 138

M

\makeatletter 173
\makeatother 175
\MessageBreak 88, 135
msg commands:
\msg_error:nn 164
\g_msg_module_name_prop 149
\g_msg_module_type_prop 148
\msg_new:nnn 150, 156, 217
\msg_show:nnn 224
\msg_show_item:nn 226
\msg_warning:nnn 56, 80

N

\NewDocumentCommand . . . 161, 201, 204, 222
\NewExpandableDocumentCommand 213
\noexpand 132

P

\PackageWarning 87
\PassOptionsToPackage 179, 189
pdf commands:
\pdf_uncompress: 30, 117
\pdf_version_gset:n 25
pdfmanagement commands:
\pdfmanagement_add:nnn *7*, 35
pdfmanagement internal commands:
\g__pdfmanagement_active_bool 119, 166, 177, 185
\g__pdfmanagement_documentproperties_prop 203, 207, 215, 226
\g__pdfmanagement_firstaidoff_clist 6, 120
\g__pdfmanagement_testphase_tl 8, 61, 76, 197
pdfmeta commands:
\g_pdfmeta_xmp_bool 38
pdfmeta internal commands:
\g__pdfmeta_standard_prop 50
prop commands:
\prop_gput:Nnn 148, 149, 207
\prop_gset_eq:NN 50

\prop_if_exist:NTF 48
\prop_item:Nn 215
\prop_map_function:NN 226
\prop_new:N 203
\ProvidesExplPackage 3

R

\RenewDocumentCommand 141, 182
\RequirePackage 65, 198

S

\ShowDocumentProperties *3*, 217
str commands:
\str_if_exist:NTF 169
\str_remove_all:Nn 187, 193
\str_uppercase:n 43
sys commands:
\c_sys_backend_str 169, 174
\sys_load_backend:n 16, 171

T

\tagpdfsetup 67, 102, 112
TEX and LATEX 2ε commands:
\@currname 204
\@gobble 138
\@latex@error 131
\@twoclasseserror 163
tl commands:
\c_space_tl 153
\tl_gput_right:Nn 61, 76
\tl_if_blank:nTF 209
\tl_new:N 8
token commands:
\token_to_str:N 153, 154