The documentmetadata-support code*

Frank Mittelbach, Ulrike Fischer, LaTeX Project

October 30, 2026

1 Introduction

The kernel command \DocumentMetadata, which can be used as the very first declaration in a document (i.e., before \documentclass), defines metadata and other configuration data that applies to the document as a whole (using a key/value syntax). It loads and activates the PDF management code from the pdfmanagement package and loads and activates code currently stored in latex-lab modules needed for various features developed as part of the multi-year "Tagged PDF" project. [1]

While the underlying functionality is still under development (e.g., further keys will be added over time and keys marked temporary may vanish again) the code for \DocumentMetadata is placed in a separate bundle, so that it is easier to update it without the need to build a full LaTeX release. Over time the functionality will move fully into the kernel.

As the loading of the PDF management forces the loading of the l3backend files, a backend that can't be detected automatically like dvipdfmx, must be set in the first \DocumentMetadata.

From a process perspective \DocumentMetadata loads the pdfmanagement code and the latex-lab modules the first time it is called and then redefines itself to error if called more than once. In particular, this means that a document without a \DocumentMetadata declaration has no code available for extended management of PDF output and for tagging support.

2 The \DocumentMetadata command

\DocumentMetadata `\DocumentMetadata{⟨key-value list⟩}`

The command should be used as the first command in a document, before \documentclass. It takes a key-value argument.

Starting with the release 2025-11-01 \DocumentMetadata will always load the new modules which changes LaTeX internals and add tagging support code.

For documents that want to load the PDF management but do not want the new tagging support code we provide a dedicated package. Such documents should replace

```
\DocumentMetadata{pdfversion=1.7,
  pdfstandard=a-3b}
```

*This file has version 1.1f dated 2026-10-20, © LaTeX Project.

by

```
\RequirePackage{pdfmanagement}
\SetKeys[document/metadata]{pdfversion=1.7,
 pdfstandard=a-3b}
```

Currently the modules loaded by `\DocumentMetadata` are the following. Details and documentation can be found in the various `latex-lab-`⟨*module*⟩`.pdf`:

names This module declares tag names for the structures.

new-or-2 This module changes output routine sockets and adds support for the paragraph tagging. It also loads the new footnote code.

block This module reimplements lists and blocks environments and adds tagging support.

context This module provides context for template instances and code that needs to know where and when it is executed.

sec,sec-template These modules reimplement heading commands and make them tagging aware.

template This module contains code extending `lttemplate.dtx`. Once fully tested it will be moved to the kernel file.

toc This module adapts commands related to the table of contents and similar lists to make them tagging aware.

minipage This module adds tagging support to `minipage` and `\parbox`.

graphic This module enables tagging support for the `\includegraphics` command and the `picture` environment.

float This module adds tagging support to floats.

bib This module adds tagging support to citations and bibliographies.

text This module adds tagging support to small text commands like the LaTeX logo, the `\emph` command and super and sub scripts.

marginpar This module adds tagging support to the `\marginpar` command.

title This module adds tagging support to the `\maketitle` command if a standard class is used. It also enhances the `\title` and `\author` commands to fill the XMP-metadata and set the window title. It is not compatible with packages and classes which redefine these commands too.

table This module provides tagging for `tabular`, `longtable` and similar table environments. Its use and restrictions are documented in `latex-lab-table.pdf`.

math This module adapts math for tagging.

firstaid This module contains small adjustments to external packages.

tikz This module adds support for the tikz package.

3 Keys and values

Currently the following keys are implemented for `\DocumentMetadata`:

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

`pdfversion` Sets the PDF version explicitly, e.g., `pdfversion=1.7`. The key definition is in the pdfmanagement in l3pdfmeta.

`uncompress` (no value) Forces an uncompressed pdf — mainly for debugging purposes. Do not unnecessarily set it in normal production because it increases the size of the final PDF considerably.

`language` The language key should be used to set the main language of the document. The value is given in BCP 47 format, so, e.g., `language=de-DE` or `language=fr`. The value is stored in the document properties and can be retrieved (expandably) with `\GetDocumentProperty{document/language}`.

The key does *not* change the language setup of a document: it doesn't change the hyphenation patterns or the fix names or forces the loading of a language package. The expectation is that packages that need to know the document language, like babel or polyglossia, read the value and react accordingly. The BCP 47 value can, for example, be parsed with `\text_bcp_parse:n` which produces 7 brace groups, see interface3. To extract, for example, the main language in the first group this code could be used:

```
\cs_generate_variant:Nn\text_bcp_parse:n{e}
\tl_if_empty:eTF { \GetDocumentProperty{document/language} }
 {
   % other code to set the main language
 }
 {
   \tl_set:Ne\mainlanguage
    {\exp_last_unbraced:Ne\use_i:nnnnnnn
    {\text_bcp_parse:e{\GetDocumentProperty{document/language}}}}
 }
```

Note that `\text_bcp_parse:e` will error if the argument is empty so it is necessary to test this case.

The key updates the default `\BCPdata` command to use with the given language value. Note that this can lead to empty fields for region or script, if, e.g., only `language=de` instead of `language=de-Latn-AT` is used. This is only relevant if neither babel nor polyglossia are used as both packages overwrite the `\BCPdata` command with their own version.

The value of the `language` key is also used to set the `/Lang` key in the PDF. If a different (shorter) value is wanted there, an optional argument can be used: `language=[de]{de-DE}`, or the `lang` key can be used to overwrite the value: `language=de-DE,lang=de`.

`other-languages` The `other-languages` key can be used to declare more languages that are used in the document. The argument is a comma list of languages

in BCP 47 format: `other-languages={fr,en-US,ar}`. The comma list is stored in the document properties and can be retrieved (expandably) with `\GetDocumentProperty{document/other-languages}`. As with the `language` key, the expectation is that language packages read the value and react accordingly.

`lang` Explicitly sets the Lang entry in the Catalog, e.g., `lang=de-DE`. If not given, the value of `\BCPdata{language}` or `\BCPdata{main.language}` at the beginning of the document is used (these commands are set by babel, the default value is `en-US`). The key is implicitly set if the `language` key is used.

`pdfstandard` Choice key to set the pdf standard. Currently `A-1b`, `A-2a`, `A-2b`, `A-2u`, `A-3a`, `A-3b`, `A-3u`, `A-4`, `A-4E` and `A-4F` are accepted as A-standard. values. The casing is irrelevant, `a-1b` works too. Note that using this key doesn't mean that the document actually follows the standard. LaTeX can neither ensure nor check all requirements of a standard, and not everything it can do theoretically has already been implemented. When setting an A-standard a color profile is included and the `/OutputIntent` is set and any javascript action in hyperref are suppressed. The `u` variants do not enforce unicode, but they will pass the information to hyperref. The `a` variants do *not* enforce (or even test) a tagged pdf yet.

Beside the A-standards it is also possible to use the values `X-4`, `X-4p`, `X-5g`, `X-5n`, `X-5pg`, `X-6`, `X-6n`, `X-6p` for a PDF/X and `UA-1` and `UA-2` for PDF/UA standard. `UA-2` should only be used together with PDF 2.0. Currently these keys set *only* the relevant XMP-metadata. They do not validate or enforce special requirements (e.g., the `UA` standards do not automatically activate tagging).

`pdfstandard` can be used more than once to set overlapping standards, e.g: `pdfstandard=A-2b,pdfstandard=X-4,pdfstandard=UA-1`. It is also possible to pass a list like `pdfstandard={UA-2,A-4F}`.

If XMP-metadata are added (see the following key `xmp`) the necessary conformance markers for the standards are set.

Standards can require some minimal or maximal PDF version and the keys will try to adjust the PDF version as necessary. At the beginning of the document the versions are checked again, and a warning is issued if the PDF version and standard conflict.

The key is defined in the pdfmanagement in l3pdfmeta and more information can be found in the documentation.

`xmp` A boolean, if set to false no XMP metadata are added to the PDF. The initial value is true. Details are described in the documentation of l3pdfmeta.

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

`tagging` This key allows to activate, deactivate or partially deactivate the tagging support. It accepts the three values `on`, `off` and `draft`. The tagpdf package and all standard modules of the tagging support are loaded regardless of the use and the value of the key!

- `tagging=on` activates tagging.

- `tagging=off` deactivates in the `class/before` hook the tagging commands, including the `\tagpdfsetup` command. It also deactivates the use of real space chars. This can be reactivated by using `tagging-setup={activate/spaces}`.
- `tagging=draft` leaves the tagging commands active but it deactivates the writing of the structure tree at the end of the compilation. This can save time when drafting a longer document but preserves, e.g., MC-content marker in the PDF stream and warnings and errors from tagpdf if the structure is faulty.

`tagging-setup` This key allows to configure the tagging. It accepts all keys that can also be used in `\tagpdfsetup`; see the tagpdf documentation. Additionally, it accepts two keys to extend the list of modules loaded:

`modules` This key previously allowed to change the list of modules. Starting with the release 2025-11-01 all standard modules are loaded always, so its only use is to load a non-standard module, e.g. `modules=verbatim-af` will load an experimental module changing the verbatim tagging.

`extra-modules` This key allows to load non-standard modules and starting with the release 2025-11-01 it is an alias of `modules`.

`check-tagging-status` This key is provided to help to identify packages that are potentially problematic when used with the tagging code. When used (with no value or with the value `listfiles`), it reads the file `latex-tagging-status.ltx` from the latex-tagging-status package and then writes at the end of the compilation a report about the compatibility of the class and the packages with the tagging project. It follows the classification done at https://latex3.github.io/tagging-project/tagging-status.

This is only a rough overview and a debugging aid, not a final report! Using packages that are classified as incompatible or partially incompatible does not mean that the tagging is necessarily broken. For example, hyperref is partially incompatible as the form fields are not properly tagged (this requires the use of the l3pdffield package), but in documents without form fields it is unproblematic. (In case of partially-compatible or incompatible packages check the table at https://latex3.github.io/tagging-project/tagging-status as it often contains an explanation what is not yet working.)

The package latex-tagging-status will be regularly updated to reflect changes in package compatibility. Erroneous messages should be reported at https://github.com/latex3/tagging-project/issues. It is also possible to create a pull request which updates the data.

`testphase` This key loaded in older LaTeX versions specific sets of modules from the testphase code. Starting with the release 2025-11-01 all standard modules are loaded automatically by `\DocumentMetadata` and with this change the key lost most of its original purpose. The values `phase-I`, `phase-II`, `phase-III` of the `testphase` key will no longer load different code variants but only activate tagging. The key can still be used to load additional experimental modules, it then works similar to the `modules` and `extra-modules` key and does not automatically activate tagging.

`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 **debug/show=para** option of tagpdf;

log with the values as described in the documentation of tagpdf;

uncompress which does the same as **uncompress** as main key;

pdfmanagement a boolean which allows to deactivate the pdfmanagement (deprecated);

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;

xmp-export this will export the XMP-metadata to a file **\jobname.xmpi**. With **debug={xmp-export=filename}** the file name can be changed; More information can be found in the documentation of l3pdfmeta of the pdfmanagement-testphase bundle;

tagpdf This loads the package tagpdf-debug which enhances various commands from tagpdf with additional debugging options. This can slow down the compilation!

BBox This helps to debug BBox values of graphics, see the documentation of latex-lab-graphic.

References

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

4 The Implementation

```
⟨@@=pdfmanagement⟩
⟨*code⟩

\RequirePackage{pdfmanagement-init}

\ExplSyntaxOn\makeatletter
```

4.1 Variables

These variable definitions are currently also done in ltdocinit. They can be removed from there once latex-lab has been updated to provide them too.

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

```
\clist_if_exist:NF \g__pdfmanagement_firstaidoff_clist
  { \clist_new:N \g__pdfmanagement_firstaidoff_clist }
```

(*End of 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_if_exist:NF \g__pdfmanagement_testphase_tl
 { \tl_new:N \g__pdfmanagement_testphase_tl }
```

(*End of definition for* \g__pdfmanagement_testphase_tl.)

4.2 Kernel changes with \DocumentMetadata

\@kernel@before@DocumentMetadata

```
\cs_new_protected:Npn \@kernel@before@DocumentMetadata
  {
    \ifx\Umathcode\@undefined
    \fontencoding{T1}
    \renewcommand\encodingdefault{T1}
    \fi
  }
```

(*End of definition for* `\@kernel@before@DocumentMetadata`. *This function is documented on page* **??**.)

4.3 \DocumentMetadata

\DocumentMetadata

\DocumentMetadata should 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.

```
\cs_set_protected:Npn \DocumentMetadata #1
  {
    \cs_if_eq:NNTF \documentclass \@twoclasseserror
      { \msg_error:nn { meta } { after-class } }
      {
```

Load general format changes

```
        \@kernel@before@DocumentMetadata
```

The wanted backend must be detected first, we read the init key and then force the loading of the backend. The backend can contain 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

          \pdf_version_gset:n { 2.0 }
```

tagpdf currently requires that the pdf version is set first.

```
          \keys_set_groups:nnn { document / metadata} {pdf}{ #1 }
```

Load the latex-lab modules:

```
          \RequirePackage{latex-lab-testphase-latest}
          \AddToDocumentProperties [document]{tagging}{inactive}
          \AddToDocumentProperties [document]{tagging/para}{inactive}
```

Process the non-init keys.

```
        \keys_set_exclude_groups:nnn  { document / metadata } { init,pdf } { #1 }
```

Finally we set up the language default. If the `language` key hasn't been set yet, we set it to the value of the `lang` key and so to empty if this key hasn't been set either. This way a language package like babel can correctly react to the situation that neither language nor lang has been explicitly set.

```
\tl_if_empty:eTF{\GetDocumentProperty{document/language}}
 {
   \tl_set:Ne\l__pdfmanagement_tmpa_tl{\GetDocumentProperty{document/lang}}
   \tl_if_empty:NF\l__pdfmanagement_tmpa_tl
    {
     \keys_set:ne{document/metadata}{language=\l__pdfmanagement_tmpa_tl}
     \__pdfmanagement_update_BCPdata:
    }
 }
 {
   \__pdfmanagement_update_BCPdata:
 }
```

At last we set up the lang default. This is done after the begindocument hook so that it can pick up settings from babel. If the Catalog dictionary already contains a lang value we do nothing, otherwise we use the value stored in `\BCPdata`, either the main language (if it exists) or the fall back language. Note: if babel is loaded without a language this gives the language `und`.

```
\g@addto@macro\@kernel@after@begindocument
 {
   \pdfdict_get:nnN {g__pdf_Core/Catalog}{Lang}\l__pdfmanagement_tmpa_tl
   \quark_if_no_value:NT\l__pdfmanagement_tmpa_tl
    {
      \tl_if_empty:eTF { \BCPdata{main.language} }
       { \tl_set:Ne \l__pdfmanagement_tmpb_tl { \BCPdata{language} } }
       { \tl_set:Ne \l__pdfmanagement_tmpb_tl { \BCPdata{main.language} } }
      \exp_last_unbraced:Ne
       \AddToDocumentProperties{[document]{lang}{\l__pdfmanagement_tmpb_tl}}
      \pdfmanagement_add:nne {Catalog} {Lang}{(\l__pdfmanagement_tmpb_tl)}
    }
 }
```

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

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

Now we redefine `\DocumentMetadata` so that it errors on future calls

```
\cs_set_protected:Npn \DocumentMetadata ##1
  {
    \@latex@error
      {Two~\noexpand\DocumentMetadata~declarations}%
      {The~document~may~use~only~one~\noexpand\DocumentMetadata~declaration.}
  }
```

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

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

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

4.4 Tagging status report

Hide our own files

```
\clist_map_inline:nn
 {
   pdfmanagement-init.sty,
   color-ltx.sty,
   xcolor-patches-tmp-ltx.sty,
   tagpdf-base.sty,
   latex-lab-testphase-latest.sty,
   tagpdf.sty,
   tagpdf-mc-code-generic.sty,
   tagpdf-mc-code-lua.sty,
   luamml-patches-kernel.sty,
   luamml-patches-amsmath.sty,
   latex-lab-enumitem.sty,
   latex-lab-titlesec.sty,
   latex-lab-kernel-changes.sty,
   latex-lab-testphase-names.sty,
   latex-lab-testphase-new-or-2.sty,
   latex-lab-testphase-block.sty,
   latex-lab-testphase-context.sty,
   latex-lab-testphase-sec.sty,
   latex-lab-testphase-sec-template.sty,
   latex-lab-testphase-template.sty,
   latex-lab-testphase-toc.sty,
   latex-lab-testphase-minipage.sty,
   latex-lab-testphase-graphic.sty,
   latex-lab-testphase-float.sty,
   latex-lab-testphase-bib.sty,
   latex-lab-testphase-text.sty,
   latex-lab-testphase-marginpar.sty,
   latex-lab-testphase-title.sty,
   latex-lab-testphase-table.sty,
   latex-lab-testphase-math.sty,
   latex-lab-testphase-firstaid.sty,
   latex-lab-testphase-tikz.sty,
   pdfmanagement-firstaid.sty,
    }
 {\expandafter\chardef\csname t@status@#1\endcsname8\relax }
\tl_new:c {l__tag_status_0_tl}
\tl_new:c {l__tag_status_1_tl}
\tl_new:c {l__tag_status_2_tl}
\tl_new:c {l__tag_status_3_tl}
\tl_new:c {l__tag_status_4_tl}
\tl_new:c {l__tag_status_-1_tl}
\cs_new_protected:Npn\__tag_status_sort:
 {
   \clist_map_inline:Nn\g__tag_status_filelist_tl
     {
       \filename@parse{##1}
       \cs_if_exist:cTF { t@status@##1 }
         {
          \str_if_eq:onF{\filename@ext}{cls}
```

```
            {
             \tl_put_right:cn
               {l__tag_status_\int_use:c { t@status@##1 }_tl}
               {~~##1\iow_newline:}
            }
          }
          {
           \str_if_eq:onT{\filename@ext}{sty}
            {
              \tl_put_right:cn
                {l__tag_status_-1_tl}
                {~~##1\iow_newline:}
            }
          }
     }
 }
\cs_new_protected:Npn \__tag_status_type:nn #1 #2
% #1 number, -1 for not listed, #2 title
 {
   \iow_term:n{~~#2}
   \iow_term:n{~~~~~~~~~~~~~~~~~~~~~~~~~~}
   \tl_if_empty:cTF{l__tag_status_#1_tl}
    {
      \iow_term:e{~~NONE\iow_newline:}
    }
    {
      \iow_term:e{\tl_use:c{l__tag_status_#1_tl}}
    }
 }
\cs_new_protected:Npn \__tag_status_log:
 {
   \file_if_exist_input:n {latex-tagging-status.ltx}
   \AddToHookNext{class/before}
     {
       \tl_const:Ne \c__tag_current_class_tl{\@currname.\@currext}
       \tl_const:Ne \c__tag_status_class_tl
        {
          \cs_if_exist:cTF { t@status@\@currname.\@currext }
           {
             \int_case:nn { \int_use:c { t@status@\@currname.\@currext } }
              {
                {0}{unknown}
                {1}{unsupported}
                {2}{currently~incompatible}
                {3}{partially~compatible}
                {4}{compatible}
              }
           }
           { not~in~the~status~list}
        }
     }
   \AddToHook{begindocument}{\tl_gset_eq:NN \g__tag_status_filelist_tl\@filelist}
   \AddToHook{enddocument/info}
     {
```

```
      \iow_term:n{}
      \iow_term:n
       {====================================}
      \iow_term:n{~~Status~report~of~the~tagging~support}
      \iow_term:n{====================================}
      \iow_term:n{~~Details~at~https://latex3.github.io/tagging-project/tagging-
status}
      \iow_term:n{~~Report~erroneous~entries~at~https://github.com/latex3/tagging-
project}
      \iow_term:n{}
      \iow_term:n{~~A.~CLASS}
      \iow_term:n{~~********}
      \iow_term:n{}
      \iow_term:e{~~\c__tag_current_class_tl\c_space_tl is~\c__tag_status_class_tl}
      \iow_term:n{}
      \iow_term:n{~~B.~PACKAGES,~LIBRARIES,~etc}
      \iow_term:n{~~***********}
      \iow_term:n{}
      \__tag_status_sort:
      \__tag_status_type:nn{1}{1.~Unsupported}
      \__tag_status_type:nn{2}{2.~Currently~incompatible}
      \__tag_status_type:nn{3}{3.~Partially~compatible}
      \__tag_status_type:nn{4}{4.~Compatible}
      \__tag_status_type:nn{0}{5.~Unknown}
      \__tag_status_type:nn{-1}{6.~Unclassified~files~with~extension~.sty}
       \iow_term:n{====================================}
      \iow_term:n{~~End~of~status~report}
      \iow_term:n{====================================}
    }
 }
```

4.5 \DocumentMetadata keys

__pdfmanagement_update_BCPdata:n A helper command for the language key. This command updates the \BCPdata command.

```
\cs_generate_variant:Nn \text_bcp_parse:n {e}
\cs_new_protected:Npn\__pdfmanagement_update_BCPdata:
 {
   \tl_set:Ne\l__pdfmanagement_tmpa_tl { \text_bcp_parse:e{\GetDocumentProperty{document/lang
   \use:e
    {
     \exp_not:N\renewcommand*\exp_not:N\BCPdata[1]
      {
       \exp_not:N \str_case:nn {##1}
        {
         { language } %was en
           { \exp_last_unbraced:No\use_i:nnnnnnn { \l__pdfmanagement_tmpa_tl} }
         { region }   %was US
           { \exp_last_unbraced:No\use_iv:nnnnnnn { \l__pdfmanagement_tmpa_tl} }
         { script }   %was Latn
           { \exp_last_unbraced:No\use_iii:nnnnnnn { \l__pdfmanagement_tmpa_tl} }
         { tag }      %was en-US
           { \exp_last_unbraced:No\use_i:nnnnnnn { \l__pdfmanagement_tmpa_tl}-
             \exp_last_unbraced:No\use_iv:nnnnnnn { \l__pdfmanagement_tmpa_tl}
           }
```

```
        }
      }
  }
}
```

(*End of definition for* `\__pdfmanagement_update_BCPdata:n`.)

`\__pdfmanagement_language_set:n` Command to handle the optional argument of the `language` key.

`\__pdfmanagement_language_set_aux:wn`

```
\cs_new_protected:Npn \__pdfmanagement_language_set:n #1
 {
   \tl_if_head_eq_charcode:nNTF {#1}[ %]
    {
      \__pdfmanagement_language_set_aux:wn  #1
    }
    {
      \AddToDocumentProperties[document]{language}{#1}
      \keys_set:ne{document/metadata}{lang={#1}}
    }
 }

\cs_new_protected:Npn \__pdfmanagement_language_set_aux:wn [#1] #2
  {
     \AddToDocumentProperties[document]{language}{#2}
     \keys_set:ne{document/metadata}{lang={#1}}
  }
```

(*End of definition for* `\__pdfmanagement_language_set:n` *and* `\__pdfmanagement_language_set_aux:wn`.)

The key definition:

```
\keys_define:nn { document / metadata }
 {
   language      .code:n =
    {
      \__pdfmanagement_language_set:n {#1}
    },
   other-languages.code:n = {\AddToDocumentProperties [document]{other-languages}{#1}},
 }

\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 }
 {
    ,check-tagging-status .choice:
    ,check-tagging-status / off .code:n = {}
    ,check-tagging-status / listfiles .code:n =
      {
        \__tag_status_log:
      }
```

```
    ,check-tagging-status .default:n = {listfiles}
 }
\keys_define:nn { document / metadata }
  {
    ,pdfversion .groups:n = { pdf }
    ,pdfstandard .groups:n = { pdf }
    ,lang .code:n =
      {
        \pdfmanagement_add:nnn {Catalog} {Lang}{(#1)}
        \AddToDocumentProperties[document]{lang}{#1}
      }
```

As the latest set of modules is loaded by default, many testphase keys only activate tagging.

```
    ,testphase .multichoices:nn =
     {latest,tagpdf,phase-I,phase-II,phase-III}
     {
       \keys_set:nn
        {document / metadata}
        {tagging=on}
     }
```

The following modules are all always loaded, therefore they can be ignored when used in a testphase key:

```
    ,testphase .multichoices:nn =
     {math,new-or,sec,sec-template,template,toc,graphic,block,minipage,float,
      bib,context,text,firstaid,table,title,marginpar,tikz}
     {}
    ,testphase / unknown .code:n =
      {
        \tl_gput_right:Nn\g__pdfmanagement_testphase_tl
           {
             \tl_if_empty:eT{\GetDocumentProperty{document/testphase/#1}}
              {
                \AddToDocumentProperties [document]{testphase/#1}{loaded}
                \file_if_exist_input:nF {#1-latex-lab-testphase.ltx}
                 {
                  \msg_warning:nnn{meta}{latex-lab-pkg-missing}{#1}
                  \AddToDocumentProperties [document]{testphase/#1}{missing}
                 }
              }
           }
      }
 }
```

Activating and deactivating tagging after tagpdf has been loaded requires a redefinition of various commands. We store the needed settings in auxiliaries that we can redefine if tagging should be used.

__tag_document_set_tagging_off:
__tag_document_set_tagging_on:
__tag_document_set_tagging:

```
\cs_new_protected:Npn \__tag_document_set_tagging_off:
 {
```

Avoid warning from non-existing hook label. This can be removed after the next tagpdf update (2025-10-13).

```
    \sys_if_engine_luatex:F
     { \AddToHook{shipout/firstpage}[tagpdf/space]{} }
    \tagpdfsetup{activate/mc=false,activate/tree=false,
       activate/struct=false,activate/socket=false,
       activate/spaces=false,
       para/tagging=false,math/setup=}
    \AddToDocumentProperties [document]{tagging}{inactive}
    \AddToDocumentProperties [document]{tagging/para}{inactive}
    \AddToDocumentProperties [document]{tagging/interwordspace}{inactive}
```

We must avoid that a command like `\LaTeX` which contains a `\ResumeTagging` command restarts tagging, so we start a global suspend. Perhaps the commands should even be set to noop.

```
    \tag_suspend:n {global}
```

We disable the setup command, as various keys assume that the tagging structure is there, see tagging-project issue #994.

```
            \cs_set_protected:Npn\tagpdfsetup##1{}
```

perhaps, need to test:

```
%\cs_set_eq:NN\tag_suspend:n\use_none:n
%\cs_set_eq:NN\tag_resume:n \use_none:n
 }
\cs_new_protected:Npn \__tag_document_set_tagging_on:
 {
   \tagpdfsetup{activate,para/tagging,activate/spaces}
   \AddToDocumentProperties [document]{tagging}{active}
   \AddToDocumentProperties [document]{tagging/para}{active}
   \AddToDocumentProperties [document]{tagging/interwordspace}{active}
 }
```

By default tagging is off.

```
\cs_new_eq:NN \__tag_document_set_tagging:\__tag_document_set_tagging_off:
\AddToHookNext{class/before}{\__tag_document_set_tagging:}
```

(*End of definition for* `\__tag_document_set_tagging_off:`, `\__tag_document_set_tagging_on:`, *and* `\__tag_document_set_tagging:`.)

```
\bool_new:N\l__tag_document_set_tagging_in_setup_bool
\bool_set_true:N\l__tag_document_set_tagging_in_setup_bool

\keys_define:nn { document / metadata }
  {
    ,tagging .choice:,
    ,tagging / on .code:n =
      {
        \cs_set_eq:NN \__tag_document_set_tagging:\__tag_document_set_tagging_on:
      }
    ,tagging / off .code:n =
      {
        \cs_set_eq:NN \__tag_document_set_tagging:\__tag_document_set_tagging_off:
        \bool_set_false:N\l__tag_document_set_tagging_in_setup_bool
```

```
    }
  ,tagging / draft .code:n =
    {
      \keys_set:nn { document / metadata }{ tagging=on }
      \AddToHook{shipout/lastpage}[tagging-setup]{\tagpdfsetup{activate/tree=false}}
      \DeclareHookRule{shipout/lastpage}{tagging-setup}{before}{tagpdf}
    }
  ,tagging-setup .code:n =
    {
      \bool_if:NT \l__tag_document_set_tagging_in_setup_bool
       { \keys_set:nn { document / metadata }{ tagging=on } }
      \tl_gclear:N\g__pdfmanagement_testphase_tl
      \keys_set_groups:nnn{__tag/setup}{load}{#1}
      \AddToHook{class/before}
       {\keys_set_exclude_groups:nnn {__tag/setup}{load}{#1}}
    }
  ,debug .code:n =
    {
      \keys_set:nn { document / metadata / debug } {#1}
    }
  ,debug / para .code:n =
    {
      \AddToHook
        {
          package/tagpdf/after
        }
        {
           \tagpdfsetup{debug/show=para}
        }
    }
  ,debug / log .code:n =
    {
      \AddToHook
        {
         package/tagpdf/after
        }
        {
           \tagpdfsetup{debug/log=#1}
        }
    }
  ,debug / tagpdf .code:n =
    {
      \AddToHook
        {
         package/tagpdf/after
        }
        {
           \RequirePackage{tagpdf-debug}
        }
    }
  ,debug / uncompress .code:n =
    {
      \pdf_uncompress:
    }
```

```
    ,debug / pdfmanagement .bool_gset:N = \g__pdfmanagement_active_bool
    ,debug / firstaidoff .clist_gset:N = \g__pdfmanagement_firstaidoff_clist
    ,debug / BBox .code:n =
      {
        \AddToHook{package/latex-lab-testphase-graphic/after}
         { \bool_set_true:N \l__tag_graphic_debug_bool}
      }
  }
\keys_define:nn{__tag/setup}
 {
   ,modules .code:n =
     {
      \keys_set:nn { document / metadata }{testphase={#1}}
     }
   ,modules .groups:n = { load }
   ,extra-modules .code:n =
     {
       \keys_set:nn { document / metadata }{testphase={#1}}
     }
   ,extra-modules .groups:n = { load }
 }
```

4.6 Messages

```
%UFi is meta the right module name here?
\prop_gput:Nnn \g_msg_module_type_prop { meta } { LaTeX }
\prop_gput:Nnn \g_msg_module_name_prop { meta } { DocumentMetadata }

\msg_new:nnn { meta } { after-class }
             {
                \token_to_str:N \DocumentMetadata \c_space_tl
                should~be~used~only~before~\token_to_str:N\documentclass
             }
\msg_new:nnn { meta } { latex-lab-pkg-missing }
             {
               LaTeX-lab~package~'#1'~not~found.
              }

\ExplSyntaxOff\makeatother
```

⟨/code⟩

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 35, 36, 59, 238, 245, 254, 282, 301, 305, 319, 320, 321, 330, 331, 332
\AddToHook 176, 177, 314, 354, 363, 372, 382, 392, 408
\AddToHookNext 157, 335
\author *2*

B

backend (key) *3*
\BCPdata *3*, *8*, *11*, 55, 56, 57, 213
bool commands:
\bool_gset_true:N 22
\bool_if:NTF 359
\bool_new:N 336
\bool_set_false:N 349
\bool_set_true:N 337, 409

C

\chardef 110
check-tagging-status (key) *3*
clist commands:
\clist_if_exist:NTF 5
\clist_map_inline:Nn 119
\clist_map_inline:nn 74
\clist_new:N 6
colorprofiles (key) *3*
cs commands:
\cs_generate_variant:Nn 207
\cs_if_eq:NNTF 18
\cs_if_exist:NTF 122, 162
\cs_new_eq:NN 334
\cs_new_protected:Npn 9, 117, 141, 154, 208, 231, 243, 311, 327
\cs_set_eq:NN 324, 325, 344, 348
\cs_set_protected:Npn 16, 64, 323
\csname 110

D

debug (key) *3*
\DeclareHookRule 355
\documentclass *1*, *7*, 18, 432
\DocumentMetadata *1–3*, *5*, *7*, *8*, *11*, 16, 431

E

\emph *2*
\encodingdefault 13
\endcsname 110
exp commands:
\exp_last_unbraced:Ne 58
\exp_last_unbraced:No 218, 220, 222, 224, 225
\exp_not:N 213, 215
\expandafter 110
\ExplSyntaxOff 31, 438
\ExplSyntaxOn 4, 29

F

\fi 14
file commands:
\file_if_exist_input:n 156
\file_if_exist_input:nTF 302
\file_input:n 30
\fontencoding 12

G

\GetDocumentProperty . . . 38, 40, 210, 299

H

hook commands:
\hook_use_once:n 63

I

\ifx 11
\includegraphics *2*
int commands:
\int_case:nn 164
\int_use:N 127, 164
iow commands:
\iow_newline: 128, 136, 148
\iow_term:n 144, 145, 148, 151, 179, 180, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 202, 203, 204

K

keys commands:
\keys_define:nn 248, 256, 265, 275, 339, 412
\keys_set:nn 43, 239, 246, 287, 353, 360, 368, 416, 421
\keys_set_exclude_groups:nnn 37, 364
\keys_set_groups:nnn 23, 33, 362

L

lang (key) *3*
language (key) *3*
\LaTeX *14*

M

\makeatletter 4, 29
\makeatother 31, 438
\maketitle *2*
\marginpar *2*
metadata keys:
backend *3*
check-tagging-status *3*
colorprofiles *3*
debug *3*
lang *3*
language *3*
other-languages *3*
pdfstandard *3*
pdfversion *3*
tagging *3*
tagging-setup *3*
testphase *3*
uncompress *3*
xmp *3*
msg commands:
\msg_error:nn 19
\g_msg_module_name_prop 427
\g_msg_module_type_prop 426
\msg_new:nnn 429, 434
\msg_warning:nnn 304

N

\noexpand 67, 68

O

other-languages (key) *3*

P

\parbox *2*
pdf commands:
\pdf_uncompress: 402
\pdf_version_gset:n 32
pdfdict commands:
\pdfdict_get:nnN 52
pdfmanagement commands:
\pdfmanagement_add:nnn . . . *8*, 60, 281
pdfmanagement internal commands:
\g__pdfmanagement_active_bool 22, 404
\g__pdfmanagement_firstaidoff_clist 5, 405
__pdfmanagement_language_set:n 231, 231, 252
__pdfmanagement_language_set_aux:wn 231, 235, 243
\g__pdfmanagement_testphase_tl 7, 70, 297, 361
\l__pdfmanagement_tmpa_tl 40, 41, 43, 52, 53, 210, 218, 220, 222, 224, 225
\l__pdfmanagement_tmpb_tl 56, 57, 59, 60
__pdfmanagement_update_BCPdata: 44, 48, 208
__pdfmanagement_update_BCPdata:n 207
pdfstandard (key) *3*
pdfversion (key) *3*
prop commands:
\prop_gput:Nnn 426, 427

Q

quark commands:
\quark_if_no_value:NTF 53

R

\relax 110
\renewcommand 13, 213
\RequirePackage 3, 34, 71, 397
\ResumeTagging *14*

S

str commands:
\str_case:nn 215
\str_if_eq:nnTF 124, 132
\str_if_exist:NTF 25
sys commands:
\c_sys_backend_str 25, 30
\sys_if_engine_luatex:TF 313
\sys_load_backend:n 27, 261

T

tag commands:
\tag_resume:n 325
\tag_suspend:n 322, 324
tag internal commands:
\c__tag_current_class_tl . . 159, 190
__tag_document_set_tagging: 311, 334, 335, 344, 348
\l__tag_document_set_tagging_in_setup_bool 336, 337, 349, 359
__tag_document_set_tagging_off: 311, 311, 334, 348
__tag_document_set_tagging_on: 311, 327, 344
\l__tag_graphic_debug_bool 409
\c__tag_status_class_tl . . . 160, 190
\g__tag_status_filelist_tl 119, 176
__tag_status_log: 154, 271
__tag_status_sort: 117, 195
__tag_status_type:nn 141, 196, 197, 198, 199, 200, 201
tagging (key) *3*
tagging-setup (key) *3*
\tagpdfsetup *5*, 315, 323, 329, 354, 377, 387

testphase (key) 3
TEX and LATEX 2ε commands:
\@currext 159, 162, 164
\@currname 159, 162, 164
\@filelist 176
\@kernel@after@begindocument 50
\@kernel@before@DocumentMetadata 9, 21
\@latex@error 66
\@twoclasseserror 18
\@undefined 11
\filename@ext 124, 132
\filename@parse 121
\g@addto@macro 50
text commands:
\text_bcp_parse:n 3, 207, 210
\title 2
tl commands:
\c_space_tl 190, 431
\tl_const:Nn 159, 160
\tl_gclear:N 361
\tl_gput_right:Nn 297
\tl_gset_eq:NN 176
\tl_if_empty:NTF 41, 146
\tl_if_empty:nTF 38, 55, 299
\tl_if_exist:NTF 7
\tl_if_head_eq_charcode:nNTF . . . 233
\tl_new:N 8, 111, 112, 113, 114, 115, 116
\tl_put_right:Nn 126, 134
\tl_set:Nn 40, 56, 57, 210
\tl_use:N 151
token commands:
\token_to_str:N 431, 432

U

\Umathcode 11
uncompress (key) 3
use commands:
\use:n 211
\use_i:nnnnnnn 218, 224
\use_iii:nnnnnnn 222
\use_iv:nnnnnnn 220, 225
\use_none:n 324, 325

X

xmp (key) 3