

About up \LaTeX 2 ϵ

Ken Nakano & Japanese \TeX Development Community & TTK

Date: 2018/04/08

Attention:

This document provides a brief description of up \LaTeX 2 ϵ , the Japanese extended version of \LaTeX 2 ϵ . This version is based on ‘p \LaTeX 2 ϵ Community Edition.’ It is now maintained by Japanese \TeX Development Community¹.

ASCII p \TeX is the most popular \TeX engine in Japan and is widely used for a high-quality typesetting, even for commercial printing. However, p \TeX has some limitations:

- The Character set available is limited to JIS X 0208, namely JIS level-1 and level-2
- Difficulty in handling 8-bit Latin, due to legacy double byte Japanese encodings
- Difficulty in typesetting CJK (Chinese, Japanese and Korean) multilingual documents

To overcome these weak points, a Unicode extension of p \TeX , up \TeX , has been developed.² The Unicode p \LaTeX format run on up \TeX is called up \LaTeX . Current up \LaTeX is maintained by Japanese \TeX Development Community,³ in sync with p \LaTeX community edition.⁴ The development version is available from GitHub repository⁵. Any bug reports and requests should be sent to Japanese \TeX Development Community, using GitHub Issue system.

¹<https://texjp.org>

²<http://www.t-lab.opal.ne.jp/tex/uptex.html>

³<https://texjp.org>

⁴<https://github.com/texjporg/platex>

⁵<https://github.com/texjporg/uplatex>

1 Introduction to this document

This document briefly describes $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$, but is not a manual of $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$. The basic functions of $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ are almost the same with those of $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ and $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$, so please refer to the documentation of those formats.

For $\text{upT}_{\text{E}}\text{X}$, please refer to the official website or [1] (in English).

This document consists of following parts:

Section 1 This section; describes this document itself.

Section 2 Brief explanation of extensions in $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$. Also describes the standard classes and packages.

Section 3 The compatibility note for users of the old version of $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ or those of the original $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}/\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$.

Appendix A Describes `DOCSTRIP` Options for this document.

Appendix B Description of ‘`upldoc.tex`’ (counterpart for ‘`source2e.tex`’ in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$).

Appendix C Description of a shell script to process ‘`upldoc.tex`’, etc.

2 About Functions of $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$

The structure of $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ is similar to that of $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$; it consists of 3 types of files: a format (`uplatex.ltx`), classes and packages.

2.1 About the Format

To make a format for $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X}$, process “`uplatex.ltx`” with INI mode of $\epsilon\text{-upT}_{\text{E}}\text{X}$.⁶ A handy command ‘`fmtutil-sys`’ (or ‘`fmtutil`’) for this purpose is available in $\text{T}_{\text{E}}\text{X}$ Live. The following command generates `uplatex.fmt`.

```
fmtutil-sys --byfmt uplatex
```

The content of `uplatex.ltx` is shown below. In the current version of $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X}$, first we simply load `latex.ltx` and modify/extend some definitions by loading `plcore.ltx` (available from $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X}$) and `uplcore.ltx`.

```
1 < *plcore >
```

⁶Formerly both $\text{upT}_{\text{E}}\text{X}$ and $\epsilon\text{-upT}_{\text{E}}\text{X}$ can make the format file for $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X}$, however, it’s not true anymore because $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ requires $\epsilon\text{-pT}_{\text{E}}\text{X}$ since 2017.

Temporarily disable `\dump` at the end of `latex.ltx`.

```
2 \let\orgdump\dump
3 \let\dump\relax
```

Load `latex.ltx` here. Within the standard installation of T_EX Live, `hyphen.cfg` provided by “Babel” package will be used.

```
4 \input latex.ltx
```

Load `plcore.ltx` and `uplcore.ltx`.

```
5 \typeout{*****^J%
6      *^J%
7      * making upLaTeX format^^J%
8      *^J%
9      *****}
10 \makeatletter
11 \input plcore.ltx
12 \input uplcore.ltx
```

Load font-related default settings, `upldefs.ltx`. If a file `upldefs.cfg` is found, then that file will be used instead.

```
13 \InputIfFileExists{upldefs.cfg}
14     {\typeout{*****^J%
15          * Local config file upldefs.cfg used^^J%
16          *****}}}%
17     {\input{upldefs.ltx}}
```

In the previous version, we displayed upL^AT_EX version on the terminal, so that it can be easily recognized during format creation; however `\everyjob` can contain any code other than showing a banner, so now disabled.

```
18 %\the\everyjob
```

Load `uplatex.cfg` if it exists at runtime of upL^AT_EX 2_ε. (Counterpart of `platex.cfg` in pL^AT_EX 2_ε.)

```
19 \everyjob\expandafter{%
20   \the\everyjob
21   \IfFileExists{uplatex.cfg}{%
22     \typeout{*****^J%
23          * Loading uplatex.cfg.^J%
24          *****}%
25     \input{uplatex.cfg}}{%
26   }
```

Dump to the format file.

```
27 \let\dump\orgdump
28 \let\orgdump\@undefined
29 \makeatother
30 \dump
31 %\endinput
```

The file `uplcore.ltx`, which provides modifications/extensions to make $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$, is a concatenation of stripped files below using `DOCSTRIP` program.

- `uplvers.dtx` defines the format version of $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$.
- `uplfonts.dtx` extends `NFSS2` for Japanese font selection.
- `plcore.dtx` (the same content as $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$); defines other modifications to $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$.

Moreover, default settings of pre-loaded fonts and typesetting parameters are done by loading `upldefs.ltx` inside `uplatex.ltx`.⁷ This file `upldefs.ltx` is also stripped from `uplfonts.dtx`.

Attention:

You can customize $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$ by tuning these settings. If you need to do that, copy/rename it as `upldefs.cfg` and edit it, instead of overwriting `upldefs.ltx` itself. If a file named `upldefs.cfg` is found at a format creation time, it will be read as a substitute of `upldefs.ltx`.

As shown above, the files in $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ is named after $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ones, prefixed with “u.”

2.1.1 Version

The version (like “2018-04-01u02”) and the format name (“`pLaTeX2e`”) of $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$ are defined in `uplvers.dtx`. This is similar to $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$, which defines those in `plvers.dtx`.

2.1.2 NFSS2 Commands

$\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$ shares `plcore.dtx` with $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$, so the extensions of `NFSS2` for selecting Japanese fonts are available.

2.1.3 Output Routine and Floats

$\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$ shares `plcore.dtx` with $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$, so the output routine and footnote macros will behave similar to $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\varepsilon}$.

⁷Older $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ loaded `upldefs.ltx` inside `uplcore.ltx`; however, $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ community edition newer than 2018 loads `upldefs.ltx` inside `uplatex.ltx`.

2.2 Classes and Packages

Classes and packages bundled with $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ are based on those in original $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$, and modified some parameters.

$\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ classes:

- `ujarticle.cls`, `ujbook.cls`, `ujreport.cls`
Standard *yoko-kumi* (horizontal writing) classes; stripped from `ujclasses.dtx`.
 $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ edition of `jarticle.cls`, `jbook.cls` and `jreport.cls`.
- `utarticle.cls`, `utbook.cls`, `utreport.cls`
Standard *tate-kumi* (vertical writing) classes; stripped from `ujclasses.dtx`.
 $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ edition of `tarticle.cls`, `tbook.cls` and `treport.cls`.

We don't provide $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ edition of `jltxdoc.cls`, but the one from $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ can be used also on $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ without problem.

$\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ packages:

- `uptrace.sty`
 $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ version of `tracefmt.sty`; the package `tracefmt.sty` overwrites $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ -style NFSS2 commands, so `uptrace.sty` provides redefinitions to recover $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ extensions. Stripped from `uplfonts.dtx`.

Other $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X}$ packages work also on $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X}$.

3 Compatibility with Other Formats and Older Versions

Here we provide some information about the compatibility between current $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ and older versions or original $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}/\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$.

3.1 Compatibility with $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}/\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$

$\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ is in most part upper compatible with $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$, so you can move from $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ to $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ by simply replacing the document class and some macros. However, the default Japanese font metrics in $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ is different from those in $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$; therefore, you should not expect identical output from both $\text{pL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ and $\text{upL}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$.

Note that up \LaTeX is a new format, so we do *not* provide support for 2.09 compatibility mode. Follow the standard $\text{\LaTeX} 2_{\epsilon}$ convention!

We hope that most classes and packages meant for $\text{\LaTeX} 2_{\epsilon}$ /p $\text{\LaTeX} 2_{\epsilon}$ works also for up $\text{\LaTeX} 2_{\epsilon}$ without any modification. However for example, if a class or a package uses Kanji encoding ‘JY1’ or ‘JT1’ (default on p $\text{\LaTeX} 2_{\epsilon}$), an error complaining the mismatch of Kanji encoding might happen on up \LaTeX , in which the default is ‘JY2’ and ‘JT2.’ In this case, we have to say that the class or package does not support up $\text{\LaTeX} 2_{\epsilon}$; you should use p \LaTeX , or report to the author of the package or class.

3.2 Support for Package ‘latexrelease’

p \LaTeX provides ‘latexrelease’ package, which is based on ‘latexrelease’ package (introduced in \LaTeX <2015/01/01>). It could be better if we also provide a similar package on up \LaTeX , but currently we don’t need it; up \LaTeX does not have any recent up \LaTeX -specific changes. So, you can safely use ‘latexrelease’ package for emulating the specified format date.

A DOCSTRIP Options

By processing `uplatex.dtx` with DOCSTRIP program, different files can be generated. Here are the DOCSTRIP options for this document:

<i>Option</i>	<i>Function</i>
plcore	Generates a fragment of format sources
pldoc	Generates ‘upldoc.tex’ for typesetting up $\text{\LaTeX} 2_{\epsilon}$ sources
shprog	Generates a shell script to process ‘upldoc.tex’
Xins	Generates a DOCSTRIP batch file ‘Xins.ins’ for generating the above shell/perl scripts

B Documentation of up $\text{\LaTeX} 2_{\epsilon}$ sources

The contents of ‘upldoc.tex’ for typesetting up $\text{\LaTeX} 2_{\epsilon}$ sources is described here. Compared to individual processings, batch processing using ‘upldoc.tex’ prints also changes and an index.

By default, the description of up $\text{\LaTeX} 2_{\epsilon}$ sources is written in Japanese. If you need English version, first save

```
\newif\ifJAPANESE
```

as `uplatex.cfg`, and process `upldoc.tex` (up \LaTeX 2 ϵ newer than July 2016 is required).

Here we explain only difference between `pldoc.tex` (p \LaTeX 2 ϵ) and `upldoc.tex` (up \LaTeX 2 ϵ).

```
33 <*pldoc>
34 \begin{filecontents}{upldoc.dic}
35 西暦      せいれき
36 和暦      われき
37 \end{filecontents}
```

The document of p \LaTeX 2 ϵ requires `plext` package, since `plext.dtx` contains several examples of partial vertical writing. However, we don't have such examples in up \LaTeX 2 ϵ files, so no need for it.

```
38 \documentclass{jltxdoc}
39 %\usepackage{plext} %% comment out for upLaTeX
40 \listfiles
41
42 \DoNotIndex{\def,\long,\edef,\xdef,\gdef,\let,\global}
43 \DoNotIndex{\if,\ifnum,\ifdim,\ifcat,\ifmmode,\ifvmode,\ifhmode,%
44             \iftrue,\iffalse,\ifvoid,\ifx,\ifeof,\ifcase,\else,\or,\fi}
45 \DoNotIndex{\box,\copy,\setbox,\unvbox,\unhbox,\hbox,%
46             \vbox,\vtop,\vcenter}
47 \DoNotIndex{@empty,\immediate,\write}
48 \DoNotIndex{\egroup,\bgroup,\expandafter,\begingroup,\endgroup}
49 \DoNotIndex{\divide,\advance,\multiply,\count,\dimen}
50 \DoNotIndex{\relax,\space,\string}
51 \DoNotIndex{\csname,\endcsname,\@spaces,\openin,\openout,%
52             \closein,\closeout}
53 \DoNotIndex{\catcode,\endinput}
54 \DoNotIndex{\jobname,\message,\read,\the,\m@ne,\noexpand}
55 \DoNotIndex{\hsize,\vsize,\hskip,\vskip,\kern,\hfil,\hfill,\hss,\vss,\unskip}
56 \DoNotIndex{\m@ne,\z@,\z@skip,\@ne,\tw@,\p@,\@minus,\@plus}
57 \DoNotIndex{\dp,\wd,\ht,\setlength,\addtolength}
58 \DoNotIndex{\newcommand,\renewcommand}
59
60 \ifJAPANESE
61 \IndexPrologue{\part*{索引}}%
62             \markboth{索引}{索引}%
63             \addcontentsline{toc}{part}{索引}%
64 イタリアック体の数字は、その項目が説明されているページを示しています。
65 下線の引かれた数字は、定義されているページを示しています。
66 その他の数字は、その項目が使われているページを示しています。}
67 \else
68 \IndexPrologue{\part*{Index}}%
69             \markboth{Index}{Index}%
70             \addcontentsline{toc}{part}{Index}%
71 The italic numbers denote the pages where the corresponding entry
72 is described, numbers underlined point to the definition,
```

```

73 all others indicate the places where it is used.}
74 \fi
75 %
76 \ifJAPANESE
77 \GlossaryPrologue{\part*{変更履歴}}%
78             \markboth{変更履歴}{変更履歴}%
79             \addcontentsline{toc}{part}{変更履歴}}
80 \else
81 \GlossaryPrologue{\part*{Change History}}%
82             \markboth{Change History}{Change History}%
83             \addcontentsline{toc}{part}{Change History}}
84 \fi
85
86 \makeatletter
87 \def\changes@#1#2#3{%
88   \let\protect\@unexpandable\protect
89   \edef\@tempa{\noexpand\glossary{#2\space\currentfile\space#1\levelchar
90     \ifx\saved@macroname\@empty
91       \space\actualchar\generalname
92     \else
93       \expandafter\@gobble
94       \saved@macroname\actualchar
95       \string\verb\quotechar*%
96       \verbatimchar\saved@macroname
97       \verbatimchar
98     \fi
99     :\levelchar #3}}%
100   \@tempa\endgroup\@esphack}
101 \makeatother
102 \RecordChanges
103 \CodelineIndex
104 \EnableCrossrefs
105 \setcounter{IndexColumns}{2}
106 \settowidth\MacroIndent{\ttfamily\scriptsize 000\ }

```

Here starts the document body.

```

107 \begin{document}
108 \title{The \upLaTeXe\ Sources}
109 \author{Ken Nakano \& Japanese \TeX\ Development Community \& TTK}
110
111 % Get the date and patch level from uplvers.dtx
112 \makeatletter
113 \let\patchdate=\@empty
114 \begingroup
115   \def\ProvidesFile#1[#2 #3]#4\def\uppatch@level#5{%
116     \date{#2}\xdef\patchdate{#5}\endinput}
117   \input{uplvers.dtx}
118 \endgroup
119
120 % Add the patch version if available.

```



```

121 \def\Xpatch{}
122 \ifx\patchdate\Xpatch\else
123   \edef\@date{\@date\space version \patchdate}
124 \fi
125 \makeatother
126
127 \pagenumbering{roman}
128 \maketitle
129 \renewcommand\maketitle{}
130 \tableofcontents
131 \clearpage
132 \pagenumbering{arabic}
133
134 \DocInclude{uplvers} % upLaTeX version
135
136 \DocInclude{uplfonts} % NFSS2 commands
137
138 \DocInclude{ukinsoku} % kinsoku parameter
139
140 \DocInclude{ujclasses} % Standard class
141
142 \StopEventually{\end{document}}
143
144 \clearpage
145 \pagestyle{headings}
146 % Make TeX shut up.
147 \hbadness=10000
148 \newcount\hbadness
149 \hfuzz=\maxdimen
150 %
151 \PrintChanges
152 \clearpage
153 %
154 \begingroup
155   \def\endash{--}
156   \catcode'\- \active
157   \def-\{\futurelet\temp\indexdash}
158   \def\indexdash{\ifx\temp-\endash\fi}
159
160   \PrintIndex
161 \endgroup
162 \let\PrintChanges\relax
163 \let\PrintIndex\relax
164 \end{document}
165 \end{pdoc}

```

C Additional Utility Programs

C.1 Shell Script `mkpldoc.sh`

A shell script to process ‘`pldoc.tex`’ and produce a fully indexed source code description. Run `sh mkpldoc.sh` to use it.

The script is almost identical to that in `pLATEX 2ε`, so here we describe only the difference.

```
166 ⟨*shprog⟩
167 for f in upldoc.toc upldoc.idx upldoc.glo ; do
168 if [ -e $f ]; then rm $f; fi
169 done
170 echo "" > ltxdoc.cfg
171 uplatex upldoc.tex
```

To make the Change log and Glossary (Change History) for `upLATEX` using ‘`mendex`,’ we need to run it in UTF-8 mode. So, option `-U` is important.⁸

```
172 mendex -U -s gind.ist -d upldoc.dic -o upldoc.ind upldoc.idx
173 mendex -U -f -s gglo.ist -o upldoc.gls upldoc.glo
174 echo "\includeonly{" > ltxdoc.cfg
175 uplatex upldoc.tex
176 echo "" > ltxdoc.cfg
177 uplatex upldoc.tex
178 # EOT
179 ⟨/shprog⟩
```

C.2 Perl Script `dstcheck.pl`

The one from `pLATEX 2ε` can be use without any change, so omitted here in `upLATEX 2ε`.

C.3 DOCSTRIP Batch file

Here we introduce a DOCSTRIP batch file ‘`Xins.ins`,’ which generates the script described in Appendix C.1. The code is almost identical to that in `pLATEX 2ε`.

```
180 ⟨*Xins⟩
181 \input docstrip
182 \keepsilent
183 {\catcode'#=12 \gdef\MetaPrefix{## }}
184 \declarepreamble\thispre
185 \endpreamble
186 \usepreamble\thispre
```

⁸The command ‘`uplatex`’ should be also in UTF-8 mode, but it defaults to UTF-8 mode; therefore, we don’t need to add `-kanji=utf8` explicitly.

```
187 \declarepostamble\thispost
188 \endpostamble
189 \usepostamble\thispost

190 \generate{
191   \file{mkpldoc.sh}{\from{uplatex.dtx}{shprog}}
192 }
193 \endbatchfile
194 </Xins>
```

References

- [1] Takuji Tanaka, Up \TeX — Unicode version of p \TeX with CJK extensions
TUGboat issue 34:3, 2013.
(<http://tug.org/TUGboat/tb34-3/tb108tanaka.pdf>)

Change History

2011/05/07 v1.0c-u00	2017/11/29 v1.0q-u01
Created up \LaTeX version based on	New English documentation added
p \LaTeX one (based on	(based on platex.dtx
platex.dtx 1997/01/29 v1.0c) . . 1	2017/11/29 v1.0q) 1
2016/05/08 v1.0h-u00	2017/12/05 v1.0s-u01
Exclude uplpatch.ltx from the	Moved loading default settings
document (based on platex.dtx	from uplcore.ltx to
2016/05/08 v1.0h) 8	uplatex.ltx (based on
2016/06/06 v1.0k-u01	platex.dtx 2017/12/05 v1.0s) . . 3
Update documents for up \LaTeX . . . 1	2017/12/10 v1.0s-u02
2016/06/19 v1.0l-u01	Load plcore.ltx before
Get the patch level from	uplcore.ltx (recent version of
uplvers.dtx (based on	p \LaTeX is assumed) 3
platex.dtx 2016/06/19 v1.0l) . . 8	2018/04/08 v1.0w-u02
2016/08/26 v1.0m-u01	Stop showing banner during
Moved loading uplatex.cfg from	format generation for safety
uplcore.ltx to uplatex.ltx	(based on platex.dtx
(based on platex.dtx	2018/04/08 v1.0w) 3
2016/08/26 v1.0m) 3	