

About p \LaTeX 2 ϵ

Ken Nakano & Japanese \TeX Development Community

Date: 2018/04/08

Attention:

This document provides a brief description of p \LaTeX 2 ϵ , the Japanese extended version of \LaTeX 2 ϵ . The original version of p \LaTeX 2 ϵ ('ASCII p \LaTeX 2 ϵ ') was developed by ASCII MEDIA WORKS¹ (formerly ASCII CORPORATION) during 1995 and 2006. The current version, which is now distributed in CTAN and \TeX Live, is a forked version called 'p \LaTeX 2 ϵ Community Edition.' It is now maintained by Japanese \TeX Development Community².

p \LaTeX is a Japanese \LaTeX format, which is adjusted/extended to be more suitable for writing Japanese documents. It requires p \TeX (\TeX engine with extensions for Japanese typesetting; it is designed for high-quality Japanese book publishing, and the 'p' stands for 'publishing'³), and the development of both p \TeX and p \LaTeX was done by ASCII Corporation.

In 2010, ASCII p \TeX was incorporated into the world-wide \TeX distribution ' \TeX Live.' Since then, p \TeX has been maintained/improved/changed along with \TeX Live sources. In recent versions of \TeX Live and W32 \TeX , the default engine of p \LaTeX changed from original p \TeX to ϵ -p \TeX (p \TeX with ϵ -p \TeX extension), and the original \LaTeX itself is also frequently updated. On the other hand, p \LaTeX remained unchanged since 2006, which resulted in some incompatibility and limitations.

To follow these upstream changes, we (Japanese \TeX Development Community) decided to fork ASCII p \LaTeX and distribute the 'community edition.' The development version is available from GitHub repository⁴. The forked community

¹<http://ascii.asciimw.jp/pb/ptex/>

²<https://texjp.org>

³There is another old implementation of Japanese \LaTeX , called j \LaTeX 2 ϵ (but not included in \TeX Live). Also, MiK \TeX has another program called 'platex,' but it has nothing to do with our Japanese p \LaTeX !

⁴<https://github.com/texjporg/platex>

edition is different from the original ASCII edition, so any bug reports and requests should be sent to Japanese T_EX Development Community, using GitHub Issue system.

This document (platex-en.pdf) is a brief explanation of the pL^AT_EX 2_ε community edition. It is somewhat of a historical document now, since pL^AT_EX 2_ε came into existence in 1995 (although the English translation has been done by Japanese T_EX Development Community since 2017). The detail of source codes are described separately in pldoc-en.pdf.

1 Introduction to this document

This document briefly describes p \LaTeX 2 ϵ , but is not a manual of p \LaTeX 2 ϵ . For the basic functions of p \LaTeX 2 ϵ , see [1] (in Japanese). For extensions of some commands for vertical writing (which were first described in [2] in Japanese), see `plex.ttx` section in `pldoc-en.pdf`.

For Japanese typesetting, please refer to the documentation of p \TeX (or “Japanese \TeX ”; the preliminary version of p \TeX), [3] (in Japanese), [4] (in English) and [5] (in English).

This document consists of following parts:

Section 1 This section; describes this document itself.

Section 2 Brief explanation of extensions in p \LaTeX 2 ϵ . Also describes the standard classes and packages.

Section 3 The compatibility note for users of the old version of p \LaTeX 2 ϵ or those of the original \LaTeX 2 ϵ .

Appendix A Describes DOCSTRIP Options for this document.

Appendix B Description of ‘`pldoc.tex`’ (counterpart for ‘`source2e.tex`’ in \LaTeX 2 ϵ).

Appendix C Description of a shell script to process ‘`pldoc.tex`’, and a tiny perl program to check DOCSTRIP guards, etc.

2 About Functions of p \LaTeX 2 ϵ

The structure of p \LaTeX 2 ϵ is similar to that of \LaTeX 2 ϵ ; it consists of 3 types of files: a format (`platex.ltx`), classes and packages.

2.1 About the Format

To make a format for p \LaTeX , process “`platex.ltx`” with INI mode of ϵ -p \TeX .⁵ A handy command ‘`fmtutil-sys`’ (or ‘`fmtutil`’) for this purpose is available in \TeX Live. The following command generates `platex.fmt`.

```
fmtutil-sys --byfmt platex
```

⁵Formerly both p \TeX and ϵ -p \TeX can make the format file for p \LaTeX , however, it’s not true anymore because \LaTeX requires ϵ -p \TeX since 2017.

The content of `platex.ltx` is shown below. In the current version of pL^AT_EX, first we simply load `latex.ltx` and modify/extend some definitions by loading `plcore.ltx`.

```

1 <{*plcore}

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 TEX Live, hyphen.cfg
provided by “Babel” package will be used.
4 \input latex.ltx

Load plcore.ltx.
5 \typeout{*****^J%
6      *^J%
7      * making pLATEX format^J%
8      *^J%
9      *****}
10 \makeatletter
11 \input plcore.ltx

```

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

```

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

```

In the previous version, we displayed pL^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.

```

17 %\the\everyjob

Load platex.cfg if it exists at runtime.
18 \everyjob\expandafter{%
19     \the\everyjob
20     \IfFileExists{platex.cfg}{%
21         \typeout{*****^J%
22             * Loading platex.cfg.^J%
23             *****}%
24         \input{platex.cfg}}{%
25     }

```

Dump to the format file.

```

26 \let\dump\orgdump
27 \let\orgdump@undefined

```

```

28 \makeatother
29 \dump
30 %\endinput
31 \</plcore>

```

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

- `plvers.dtx` defines the format version of $\text{p}\text{L}\text{A}\text{T}\text{E}\text{X } 2_{\varepsilon}$.
- `plfonts.dtx` extends `NFSS2` for Japanese font selection.
- `plcore.dtx` 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 `pldefs.ltx` inside `platex.ltx`.⁶ This file `pldefs.ltx` is also stripped from `plfonts.dtx`.

Attention:

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

2.1.1 Version

The version (like “2018-04-01”) and the format name (“`pLaTeX2e`”) of $\text{p}\text{L}\text{A}\text{T}\text{E}\text{X } 2_{\varepsilon}$ are defined in `plvers.dtx`.

2.1.2 NFSS2 Commands

$\text{L}\text{A}\text{T}\text{E}\text{X } 2_{\varepsilon}$ uses `NFSS2` as a font selection scheme, however, it supports only alphabetic fonts. $\text{p}\text{L}\text{A}\text{T}\text{E}\text{X } 2_{\varepsilon}$ extends `NFSS2` to enable selection of Japanese fonts in a consistent manner with the original `NFSS2`.

Most of the interface commands are defined to be clever enough, so that it can automatically judge whether it is going to change alphabetic fonts or Japanese fonts. It works almost fine with most of the widely used classes and packages, without any modification.

For the detail of (the original) `NFSS2`, please refer to `fntguide.tex` in $\text{L}\text{A}\text{T}\text{E}\text{X } 2_{\varepsilon}$.

⁶ASCII $\text{p}\text{L}\text{A}\text{T}\text{E}\text{X}$ loaded `pldefs.ltx` inside `plcore.ltx`; however, $\text{p}\text{L}\text{A}\text{T}\text{E}\text{X}$ community edition newer than 2018 loads `pldefs.ltx` inside `platex.ltx`.

2.1.3 Output Routine and Floats

`plcore.dtx` modifies and extends some $\text{\LaTeX 2}_{\epsilon}$ commands for Japanese processing.

- Preamble commands
- Page breaking
- Line breaking
- The order of float objects
- Crop marks (“tombow”)
- Footnote macros
- Cross-referencing
- Verbatim

2.2 Classes and Packages

Classes and packages bundled with $\text{\LaTeX 2}_{\epsilon}$ are based on those in original $\text{\LaTeX 2}_{\epsilon}$, with some Japanese localization.

$\text{\LaTeX 2}_{\epsilon}$ classes:

- `jarticle.cls`, `jbook.cls`, `jreport.cls`
Standard *yoko-kumi* (horizontal writing) classes; stripped from `jclasses.dtx`.
- `tarticle.cls`, `tbook.cls`, `treport.cls`
Standard *tate-kumi* (vertical writing) classes; stripped from `jclasses.dtx`.
- `jltxdoc.cls`
Class for typesetting Japanese `.dtx` file; stripped from `jltxdoc.dtx`.

$\text{\LaTeX 2}_{\epsilon}$ packages:

- `plext.sty`
Useful macros and extensions for vertical writing; stripped from `plext.dtx`.

- `ptrace.sty`
 $\text{p}\text{\LaTeX} 2_{\epsilon}$ version of `tracefmt.sty`; the package `tracefmt.sty` overwrites $\text{p}\text{\LaTeX} 2_{\epsilon}$ -style NFSS2 commands, so `ptrace.sty` provides redefinitions to recover $\text{p}\text{\LaTeX} 2_{\epsilon}$ extensions. Stripped from `plfonts.dtx`.
- `pfltrace.sty`
 $\text{p}\text{\LaTeX} 2_{\epsilon}$ version of `fltrace.sty` (introduced in $\text{\LaTeX} 2_{\epsilon}$ 2014/05/01); stripped from `plcore.dtx`.
- `oldpfont.sty`
Provides $\text{p}\text{\LaTeX} 2.09$ font commands; stripped from `p1209.dtx`.

The packages “`ascmac.sty`” and “`nidanfloat.sty`”, which had been included in previous versions of $\text{p}\text{\LaTeX}$, is now distributed as a separate bundle.

3 Compatibility with Other Formats and Older Versions

Here we provide some information about the compatibility between current $\text{p}\text{\LaTeX} 2_{\epsilon}$ and older versions or original $\text{\LaTeX} 2_{\epsilon}$.

3.1 Compatibility with $\text{\LaTeX} 2_{\epsilon}$

$\text{p}\text{\LaTeX} 2_{\epsilon}$ is in most part upper compatible with $\text{\LaTeX} 2_{\epsilon}$, but some parameters are adjusted to be suitable for Japanese. Therefore, you should not expect identical output, even though the same source can be processed on both $\text{\LaTeX} 2_{\epsilon}$ and $\text{p}\text{\LaTeX} 2_{\epsilon}$.

We hope that most classes and packages meant for $\text{\LaTeX} 2_{\epsilon}$ works also for $\text{p}\text{\LaTeX} 2_{\epsilon}$ without any modification. However for example, if a class or a package redefines a command which is already modified by $\text{p}\text{\LaTeX} 2_{\epsilon}$, it might cause an error at the worst case. We cannot tell whether a class or a package works fine with $\text{p}\text{\LaTeX} 2_{\epsilon}$ beforehand; the easiest way is to try to use it. If it fails, please refer to the log file or a package manual.

3.2 Compatibility with $\text{p}\text{\LaTeX} 2.09$

$\text{p}\text{\LaTeX} 2_{\epsilon}$ has ‘ $\text{p}\text{\LaTeX} 2.09$ compatibility mode’; use `\documentstyle` to enter it, but the support might be limited. Note that the 2.09 compatibility mode is

provided solely to allow you to process very old documents, which were written for a very old system.

3.3 Support for Package ‘latexrelease’

pL^AT_EX provides ‘platexrelease’ package, which is based on ‘latexrelease’ package (introduced in L^AT_EX <2015/01/01>). It may be used to ensure stability where needed, by emulating the specified format date without regenerating the format file. For more detail, please refer to its documentation.

A DOCSTRIP Options

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

<i>Option</i>	<i>Function</i>
<code>plcore</code>	Generates a fragment of format sources
<code>pldoc</code>	Generates ‘ <code>pldoc.tex</code> ’ for typesetting pL ^A T _E X 2 _ε sources
<code>shprog</code>	Generates a shell script to process ‘ <code>pldoc.tex</code> ’
<code>plprog</code>	Generates a tiny perl program to check DOCSTRIP guards nesting
<code>Xins</code>	Generates a DOCSTRIP batch file ‘ <code>Xins.ins</code> ’ for generating the above shell/perl scripts

B Documentation of pL^AT_EX 2_ε sources

The contents of ‘`pldoc.tex`’ for typesetting pL^AT_EX 2_ε sources is described here. Compared to individual processings, batch processing using ‘`pldoc.tex`’ prints also changes and an index. The whole document will have about 200 pages.

By default, the description of pL^AT_EX 2_ε sources is written in Japanese. If you need English version, first save

```
\newif\ifJAPANESE
```

as `platex.cfg`, and process `pldoc.tex` (pL^AT_EX 2_ε Community Edition newer than July 2016 is required).

First, create `pldoc.dic`; it serves as a dictionary for ‘`mendex`’ (Japanese index processor⁷), which is necessary for indexing control sequences containing Japanese characters (`\西暦` and `\和暦`).

⁷Developed by ASCII Corporation; the program ‘`makeindex`’ cannot handle Japanese characters properly, especially Kanji characters which should be sorted by its readings.


```

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

```

We use `jltxdoc` class; we also require `plext` package, since `plext.dtx` contains several examples of partial vertical writing.

```

37 \documentclass{jltxdoc}
38 \usepackage{plext}
39 \listfiles
40

```

Do not index some \TeX primitives, and some common plain \TeX commands.

```

41 \DoNotIndex{\def,\long,\edef,\xdef,\gdef,\let,\global}
42 \DoNotIndex{\if,\ifnum,\ifdim,\ifcat,\ifmmode,\ifvmode,\ifhmode,%
43             \iftrue,\iffalse,\ifvoid,\ifx,\ifeof,\ifcase,\else,\or,\fi}
44 \DoNotIndex{\box,\copy,\setbox,\unvbox,\unhbox,\hbox,%
45             \vbox,\vtop,\vcenter}
46 \DoNotIndex{\@empty,\immediate,\write}
47 \DoNotIndex{\egroup,\bgroup,\expandafter,\begingroup,\endgroup}
48 \DoNotIndex{\divide,\advance,\multiply,\count,\dimen}
49 \DoNotIndex{\relax,\space,\string}
50 \DoNotIndex{\csname,\endcsname,\@spaces,\openin,\openout,%
51             \closein,\closeout}
52 \DoNotIndex{\catcode,\endinput}
53 \DoNotIndex{\jobname,\message,\read,\the,\m@ne,\noexpand}
54 \DoNotIndex{\hsize,\vsize,\hskip,\vskip,\kern,\hfil,\hfill,\hss,\vss,\unskip}
55 \DoNotIndex{\m@ne,\z@,\z@skip,\@ne,\tw@,\p@,\@minus,\@plus}
56 \DoNotIndex{\dp,\wd,\ht,\setlength,\addtolength}
57 \DoNotIndex{\newcommand,\renewcommand}
58

```

Set up the Index and Change History to use `\part`.

```

59 \ifJAPANESE
60 \IndexPrologue{\part*{索引}}%
61             \markboth{索引}{索引}%
62             \addcontentsline{toc}{part}{索引}%
63 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。
64 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。
65 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。}
66 \else
67 \IndexPrologue{\part*{Index}}%
68             \markboth{Index}{Index}%
69             \addcontentsline{toc}{part}{Index}%
70 The italic numbers denote the pages where the corresponding entry
71 is described, numbers underlined point to the definition,
72 all others indicate the places where it is used.}
73 \fi
74 %

```

```

75 \ifJAPANESE
76 \GlossaryPrologue{\part*{変更履歴}%
77     \markboth{変更履歴}{変更履歴}%
78     \addcontentsline{toc}{part}{変更履歴}}
79 \else
80 \GlossaryPrologue{\part*{Change History}%
81     \markboth{Change History}{Change History}%
82     \addcontentsline{toc}{part}{Change History}}
83 \fi
84

```

Modify the standard `\changes` command slightly, to better cope with this multiple file document.

```

85 \makeatletter
86 \def\changes@#1#2#3{%
87     \let\protect\@unexpandable@protect
88     \edef\@tempa{\noexpand\glossary{#2\space
89         \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}

```

Codelines are allowed to run over a bit without showing up as overfull.

```

101 \renewcommand*{\MacroFont}{\fontencoding\encodingdefault
102     \fontfamily\ttdefault
103     \fontseries\mddefault
104     \fontshape\updefault
105     \small
106     \hfuzz 6pt\relax}

```

Section numbers now reach eg 19.12 which need more space.

```

107 \renewcommand*{\l@section}{\@dottedtocline{2}{1.5em}{2.8em}}
108 \renewcommand*{\l@subsubsection}{\@dottedtocline{3}{3.8em}{3.4em}}
109 \makeatother

```

Produce a Change Log and (2 column) Index.

```

110 \RecordChanges
111 \CodelineIndex
112 \EnableCrossrefs
113 \setcounter{IndexColumns}{2}
114 \settowidth\MacroIndent{\ttfamily\scriptsize 000\ }

```

Here starts the document body.

```
115 \begin{document}
116 \title{The \pLaTeX\ Sources}
117 \author{Ken Nakano \& Japanese \TeX\ Development Community}
118
119 % Get the date and patch level from plvers.dtx
120 \makeatletter
121 \let\patchdate=\@empty
122 \begingroup
123   \def\ProvidesFile#1\pfmtversion#2#3\ppatch@level#4{%
124     \date{#2}\xdef\patchdate{#4}\endinput}
125   \input{plvers.dtx}
126 \endgroup
127
128 % Add the patch version if available.
129 \def\Xpatch{0}
130 \ifx\patchdate\Xpatch\else
131 % number is assumed
132 \ifnum\patchdate>0
133   \edef\@date{\@date\space Patch level\space\patchdate}
134 \else
135   \edef\@date{\@date\space Pre-Release\patchdate}
136 \fi\fi
137 \makeatother
138
139 \pagenumbering{roman}
140 \maketitle
141 \renewcommand\maketitle{}
142 \tableofcontents
143 \clearpage
144 \pagenumbering{arabic}
145
146 \DocInclude{plvers} % pLaTeX version
147
148 \DocInclude{plfonts} % NFSS2 commands
149
150 \DocInclude{plcore} % kernel commands
151
152 \DocInclude{plext} % external commands
153
154 \DocInclude{pl209} % 2.09 compatibility mode commands
155
156 \DocInclude{kinsoku} % kinsoku parameter
157
158 \DocInclude{jclasses} % Standard class
159
160 \DocInclude{jltxdoc} % dtx documents class
161
```

Stop here if `ltxdoc.cfg` says `\AtEndOfClass{\OnlyDescription}`.

```
162 \StopEventually{\end{document}}
```

```
163
```

Print Change History and Index. Please refer to Appendix C.1 for processing of Change History and Index.

```
164 \clearpage
```

```
165 \pagestyle{headings}
```

```
166 % Make TeX shut up.
```

```
167 \hbadness=10000
```

```
168 \newcount\hbadness
```

```
169 \hfuzz=\maxdimen
```

```
170 %
```

```
171 \PrintChanges
```

```
172 \clearpage
```

```
173 %
```

```
174 \begingroup
```

```
175   \def\endash{--}
```

```
176   \catcode'\-\active
```

```
177   \def-\{\futurelet\temp\indexdash}
```

```
178   \def\indexdash{\ifx\temp-\endash\fi}
```

```
179
```

```
180   \PrintIndex
```

```
181 \endgroup
```

Make sure that the index is not printed twice (`ltxdoc.cfg` might have a second command).

```
182 \let\PrintChanges\relax
```

```
183 \let\PrintIndex\relax
```

```
184 \end{document}
```

```
185 \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.

C.1.1 Content of `mkpldoc.sh`

First, delete auxiliary files which might be created in the previous runs.

```
186 <shprog>
```

```
187 for f in pldoc.toc pldoc.idx pldoc.glo ; do
```

```
188 if [ -e $f ]; then rm $f; fi
```

```
189 done
```

First run: empty the config file `ltxdoc.cfg`.

```
190 echo "" > ltxdoc.cfg
```

Now process `pldoc.tex`.

```
191 platex pldoc.tex
```

Make the Change log and Glossary (Change History) using `mendex`. ‘Mendex’ is a Japanese index processor, which is mostly upper compatible with ‘makeindex’ and automatically handles readings of Kanji words.

Option `-s` employs a style file for formatting. Here we use `gind.ist` and `gglo.ist` from L^AT_EX 2_ε.

Option `-o` specifies output index file name.

Option `-f` forces to output Kanji characters even non-existent in dictionaries. (Makeindex does not have this option.)

```
192 mendex -s gind.ist -d pldoc.dic -o pldoc.ind pldoc.idx
193 mendex -f -s gglo.ist -o pldoc.gls pldoc.glo
```

Second run: append `\includeonly{}` to `ltxdoc.cfg` to speed up things. This run is needed only to get changes and index listed in `.toc` file.

```
194 echo "\includeonly{" > ltxdoc.cfg
195 platex pldoc.tex
```

Third and final run: restore the `cfg` file to put everything together.

```
196 echo "" > ltxdoc.cfg
197 platex pldoc.tex
198 # EOT
199 </shprog>
```

C.2 Perl Script `dstcheck.pl`

Here we provide a perl script which helps checking the nested DOCSTRIP guards. Usage:

```
perl dstcheck.pl <file-name>
```

The description of this script itself is available only in Japanese.

```
200 <*plprog>
201 ##
202 ## DOCSTRIP 文書内の環境や条件の入れ子を調べる perl スクリプト
203 ##
204 push(@dst,"DUMMY"); push(@dst,"000");
205 push(@env,"DUMMY"); push(@env,"000");
206 while (<>) {
```

```

207 if (/^%<\*([>]+)>/) { # check conditions
208     push(@dst,$1);
209     push(@dst,$.);
210 } elsif (/^%<\[/([>]+)>/) {
211     $linenum = pop(@dst);
212     $conditions = pop(@dst);
213     if ($1 ne $conditions) {
214         if ($conditions eq "DUMMY") {
215             print "$ARGV: '</$1>' (l.$.) is not started.\n";
216             push(@dst,"DUMMY");
217             push(@dst,"000");
218         } else {
219             print "$ARGV: '<*$conditions>' (l.$linenum) is ended ";
220             print "by '<*$1>' (l.$.)\n";
221         }
222     }
223 }
224 if (/^% *\\begin\{verbatim\}/) { # check environments
225     while(<>) {
226         last if (/^% *\\end\{verbatim\}/);
227     }
228 } elsif (/^% *\\begin\{([~}]*)\}/) {
229     push(@env,$1);
230     push(@env,$.);
231 } elsif (/^% *\\begin\{([~}]*)\}/) {
232     push(@env,$1);
233     push(@env,$.);
234 } elsif (/^% *\\end\{([~}]*)\}/) {
235     $linenum = pop(@env);
236     $environment = pop(@env);
237     if ($1 ne $environment) {
238         if ($environment eq "DUMMY") {
239             print "$ARGV: '\\end{$1}' (l.$.) is not started.\n";
240             push(@env,"DUMMY");
241             push(@env,"000");
242         } else {
243             print "$ARGV: \\begin{$environment} (l.$linenum) is ended ";
244             print "by \\end{$1} (l.$.)\n";
245         }
246     }
247 }
248 }
249 $linenum = pop(@dst);
250 $conditions = pop(@dst);
251 while ($conditions ne "DUMMY") {
252     print "$ARGV: '<*$conditions>' (l.$linenum) is not ended.\n";
253     $linenum = pop(@dst);
254     $conditions = pop(@dst);

```

```

255 }
256 $linenum = pop(@env);
257 $environment = pop(@env);
258 while ($environment ne "DUMMY") {
259     print "$ARGV: '\\begin{$environment}' (l.$linenum) is not ended.\n";
260     $linenum = pop(@env);
261     $environment = pop(@env);
262 }
263 exit;
264 </plprog>

```

C.3 DOCSTRIP Batch file

Here we introduce a DOCSTRIP batch file ‘Xins.ins,’ which generates the scripts described in Appendix C.1 and C.2.

```

265 <*Xins>
266 \input docstrip
267 \keepsilent
268 {\catcode'#=12 \gdef\MetaPrefix{## }}
269 \declarepreamble\thispre
270 \endpreamble
271 \usepreamble\thispre
272 \declarepostamble\thispost
273 \endpostamble
274 \usepostamble\thispost
275 \generate{
276     \file{dstcheck.pl}{\from{platex.dtx}{plprog}}
277     \file{mkpldoc.sh}{\from{platex.dtx}{shprog}}
278 }
279 \endbatchfile
280 </Xins>

```

References

- [1] 中野 賢『日本語 L^AT_EX 2_ε ブック』アスキー, 1996.
- [2] インプレス・ラボ監修, アスキー書籍編集部編『縦組対応 パーソナル日本語 T_EX』アスキー出版局, 1994
- [3] アスキー出版技術部責任編集『日本語 T_EX テクニカルブック I』アスキー, 1990.
- [4] Haruhiko Okumura, pT_EX and Japanese Typesetting The Asian Journal of T_EX, Volume 2, No. 1, 2008.
(<http://ajt.ktug.org/2008/0201okumura.pdf>)
- [5] Hisato Hamano, Vertical Typesetting with T_EX. TUGboat issue 11:3, 1990.
(<https://tug.org/TUGboat/tb11-3/tb29hamano.pdf>)
- [6] Donald E. Knuth. “*The T_EXbook*”. Addison-Wesley, 1984. (邦訳：斎藤信男監修, 鷺谷好輝訳, T_EX ブック 改訂新版, アスキー出版局, 1989)
- [7] Laslie Lamport. “*L^AT_EX: A Document Preparation System*”. Addison-Wesley, second edition, 1994.
- [8] Laslie Lamport. “*L^AT_EX: A Document Preparation System*”. Addison-Wesley, 1986. (邦訳：倉沢良一監修, 大野俊治・小暮博通・藤浦はる美訳, 文書処理システム L^AT_EX, アスキー, 1990)
- [9] Michel Goossens, Frank Mittelbach, Alexander Samarin. “*The L^AT_EX Companion*”. Addison-Wesley, 1994.
- [10] 河野 真治『入門 Perl』アスキー出版局, 1994

Change History

1995/05/08 v1.0		2016/05/20 v1.0j	
first edition	3	Add description of ‘pfltrace’	6
1995/08/25 v1.0a		2016/05/21 v1.0k	
Added ‘Compatibility’, ‘Usage of		Print also changes.	1
DOCSTRIP’ and ‘References’ . . .	3	2016/06/19 v1.0l	
1996/02/01 v1.0b		Get the patch level from	
Adjusted for the latest DOCSTRIP		plvers.dtx	11
(omake-sh.ins and		2016/08/26 v1.0m	
omake-pl.ins	15	Moved loading platex.cfg from	
1997/01/23 v1.0c		plcore.ltx to platex.ltx . . .	4
Adjusted for the latest DOCSTRIP.	15	2016/09/14 v1.0n	
Don’t copy gind.ist and gglo.ist		Improved banner saving method . .	4
from		2017/09/24 v1.0o	
\$TEXMF/tex/latex2e/base		Allow negative patch level for	
directory.	12	pre-release	11
1997/01/25 v1.0c		2017/11/11 v1.0p	
Add to filecontents environment		Moved banner saving code from	
for pldoc.dic.	8	platex.ltx to plcore.ltx . . .	4
1997/01/29 v1.0c		2017/12/02 v1.0r	
Rename pltpatch.ltx to		English references added	3
plpatch.ltx	11	2017/12/05 v1.0s	
2016/01/27 v1.0d		Moved loading default settings	
Add -e test before rm command .	12	from plcore.ltx to	
Updated descriptions of pL ^A T _Ε X 2 _ε		platex.ltx	4
files	6	2018/02/07 v1.0t	
2016/02/16 v1.0e		Moved ascmac package to separate	
Add a description of platexrelease	8	bundle	7
2016/04/12 v1.0f		2018/02/18 v1.0u	
Update document.	1	Moved nidanfloat package to	
2016/05/07 v1.0g		separate bundle	7
Save L ^A T _Ε X banner	4	2018/04/06 v1.0v	
2016/05/08 v1.0h		Sync with the latest source2e.tex	10
Exclude plpatch.ltx from the		2018/04/08 v1.0w	
document	11	Stop showing banner during	
2016/05/12 v1.0i		format generation for safety . . .	4
Undefine temporary command			
\orgdump in the end.	4		