

About p \LaTeX 2 ϵ

Ken Nakano & Japanese \TeX Development Community

Date: 2020/09/28

p \LaTeX is a Japanese \LaTeX format, which is adjusted/extended to be more suitable for writing Japanese documents. It requires p \TeX ¹, a \TeX engine with extensions for Japanese typesetting, which is designed for high-quality Japanese book “p”ublishing.² Both of them were developed by ASCII Corporation (and its successor ASCII Media Works), so they are often referred to as “ASCII p \TeX ” and “ASCII p \LaTeX ” respectively.

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 (around 2011), the default engine of p \LaTeX changed from original p \TeX to ϵ -p \TeX (p \TeX with ϵ - \TeX extension). Also, 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 Community³) decided to fork ASCII p \LaTeX and distribute the “community edition.” The development version is available from GitHub repository⁴. The forked community edition is different from the original ASCII edition, so any bug reports and requests should be sent to Japanese \TeX Development Community, using GitHub Issue system.

This document (platex-en.pdf) is a brief explanation of the p \LaTeX 2 ϵ community edition. It is somewhat of a historical document now, since p \LaTeX 2 ϵ came into existence in 1995 (although the English translation has been done by Japanese \TeX Development Community since 2017).

¹The p \TeX website: <https://asciidwango.github.io/ptex/> (in Japanese)

²There is another old implementation of Japanese \LaTeX by NTT Electrical Communications Laboratories, named j \LaTeX (unavailable in \TeX Live). Also, MiK \TeX has another program `platex` for Polish, but it has nothing to do with our Japanese p \LaTeX !

³<https://texjp.org>

⁴<https://github.com/texjorg/platex>

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 `plext.dtx` 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 - \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. Some code may be executed after loading.

```

12 \InputIfFileExists{pldefs.cfg}
13     {\typeout{*****^J%
14         * Local config file pldefs.cfg used^J%
15         *****}}
16     {\input{pldefs.ltx}}
17 \ifx\code@after@pldefs\@undefined\else \code@after@pldefs \fi

```

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.

```

18 %\the\everyjob

   Load platex.cfg if it exists at runtime.
19 \everyjob\expandafter{%
20     \the\everyjob
21     \IfFileExists{platex.cfg}{%
22         \typeout{*****^J%
23             * Loading platex.cfg.^J%
24             *****}%
25         \input{platex.cfg}}{}%
26 }

```

Dump to the format file.

```

27 \let\dump\orgdump

```

```

28 \let\orgdump\@undefined
29 \makeatother
30 \dump
31 %\endinput
32 </plcore>

```

The file `plcore.ltx`, which provides modifications/extensions to make pL^AT_εE_X, is a concatenation of stripped files below using DOCSTRIP program.

- `plvers.dtx` defines the format version of pL^AT_εE_X.
- `plfonts.dtx` extends NFSS2 for Japanese font selection.
- `plcore.dtx` defines other modifications to L^AT_εE_X.

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 pL^AT_εE_X 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 “2020-10-01”) and the format name (“pL^AT_εE_X2e”) of pL^AT_εE_X are defined in `plvers.dtx`.

2.1.2 NFSS2 Commands

L^AT_εE_X uses NFSS2 as a font selection scheme, however, it supports only alphabetic fonts. pL^AT_εE_X 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 L^AT_εE_X.

⁶ASCII pL^AT_εE_X loaded `pldefs.ltx` inside `plcore.ltx`; however, pL^AT_εE_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 $\text{p}\text{\LaTeX} 2_{\epsilon}$ are based on those in original $\text{\LaTeX} 2_{\epsilon}$, with some Japanese localization.

$\text{p}\text{\LaTeX} 2_{\epsilon}$ 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`.

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

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

- `ptrace.sty`
`pLATεEX` version of `tracefnt.sty`; the package `tracefnt.sty` overwrites `pLATεEX`-style NFSS2 commands, so `ptrace.sty` provides redefinitions to recover `pLATεEX` extensions. Stripped from `plfonts.dtx`.
- `pfltrace.sty`
`pLATεEX` version of `fltrace.sty` (introduced in `LATεX` 2014/05/01); stripped from `plcore.dtx`.
- `oldfont.sty`
Provides `pLATεEX` 2.09 font commands; stripped from `pl209.dtx`.

The packages “`ascmac.sty`” and “`nidanfloat.sty`”, which had been included in previous versions of `pLATεEX`, 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 `pLATεEX` and older versions or original `LATεX`.

3.1 Compatibility with `LATεX` 2_ε

`pLATεEX` is in most part upward compatible with `LATεX` 2_ε, 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 `LATεX` 2_ε and `pLATεEX`.

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

Some `LATεX` packages are known to be incompatible with `pLATεEX`. For those packages, `pLATεEX`-specific patches might be available. Please refer to the documentation of the `plautopatch` package (by Hironobu Yamashita).

3.2 Compatibility with pL^AT_ΕX 2.09

pL^AT_ΕX 2_ε has ‘pL^AT_ΕX 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_ΕX provides ‘latexrelease’ package, which is based on ‘latexrelease’ package (introduced in L^AT_ΕX <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 ‘pldoc.tex’ for typesetting pL ^A T _Ε X 2 _ε sources
<code>shprog</code>	Generates a shell script to process ‘pldoc.tex’
<code>plprog</code>	Generates a tiny perl program to check DOCSTRIP guards nesting
<code>Xins</code>	Generates a DOCSTRIP batch file ‘Xins.ins’ for generating the above shell/perl scripts

B Documentation of pL^AT_ΕX 2_ε sources

The contents of ‘pldoc.tex’ for typesetting pL^AT_ΕX 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_ΕX 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_ΕX 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 (`\西曆` and `\和曆`).

```
33 \*pldoc
34 \begin{filecontents}{pldoc.dic}
35 西曆    せいれき
36 和曆    われき
37 \end{filecontents}
```

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

```
38 \documentclass{jltxdoc}
39 \usepackage{plext}
40 \listfiles
41
```

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

```
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
```

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

```
60 \ifJAPANESE
61 \IndexPrologue{\part*{索引}}%
62             \markboth{索引}{索引}%
63             \addcontentsline{toc}{part}{索引}%
64 イタリア体の数字は、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}}%
```

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


```

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

```

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

```

86 \makeatletter
87 \def\changes@#1#2#3{%
88   \let\protect\@unexpandable@protect
89   \edef\@tempa{\noexpand\glossary{#2\space
90     \currentfile\space#1\levelchar
91     \ifx\saved@macroname\@empty
92       \space\actualchar\generalname
93     \else
94       \expandafter\@gobble
95       \saved@macroname\actualchar
96       \string\verb\quotechar*%
97       \verbatimchar\saved@macroname
98       \verbatimchar
99     \fi
100     :\levelchar #3}}%
101   \@tempa\endgroup\@esphack}

```

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

```

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

```

Section numbers now reach eg 19.12 which need more space.

```

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

```

Produce a Change Log and (2 column) Index.

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

Set the title, authors and the date for this document.
116 \title{The \pLaTeXe\ 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
138 % Add the last update info, in case format date unchanged
139 % Note: \@ifl@t@r can be used only in preamble.
140 \def\lastupd@te{0000/00/00}
141 \begingroup
142   \def\ProvidesFile#1[#2 #3]{%
143     \def\@tempd@te{#2}\endinput
144     \@ifl@t@r{\@tempd@te}{\lastupd@te}{%
145       \global\let\lastupd@te\@tempd@te
146     }}
147   \let\ProvidesClass\ProvidesFile
148   \let\ProvidesPackage\ProvidesFile
149   \input{plvers.dtx}
150   \input{plexpl3.dtx}
151   \input{plfonts.dtx}
152   \input{plcore.dtx}
153   \input{plexpt.dtx}
154   \input{pl209.dtx}
155   \input{kingsoku.dtx}
156   \input{jclasses.dtx}
157   \input{jltxdoc.cls}
```

```

158 \endgroup
159 \@ifl@t@r{\lastupd@te}{\pfmtversion}{%
160 \edef\@date{\@date\break (last updated: \lastupd@te)}%
161 }{}
162 \makeatother

```

Here starts the document body.

```

163 \begin{document}
164 \pagenumbering{roman}
165 \maketitle
166 \renewcommand\maketitle{}
167 \tableofcontents
168 \clearpage
169 \pagenumbering{arabic}
170
171 \DocInclude{plvers} % pLaTeX version
172
173 \DocInclude{plexpl3} % additions to expl3
174
175 \DocInclude{plfonts} % NFSS2 commands
176
177 \DocInclude{plcore} % kernel commands
178
179 \DocInclude{plext} % external commands
180
181 \DocInclude{pl209} % 2.09 compatibility mode commands
182
183 \DocInclude{kinksoku} % kinksoku parameter
184
185 \DocInclude{jclasses} % Standard class
186
187 \DocInclude{jltxdoc} % dtx documents class
188

```

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

```

189 \StopEventually{\end{document}}
190

```

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

```

191 \clearpage
192 \pagestyle{headings}
193 % Make TeX shut up.
194 \hbadness=10000
195 \newcount\hbadness
196 \hfuzz=\maxdimen
197 %
198 \PrintChanges
199 \clearpage
200 %

```

```

201 \begingroup
202 \def\endash{--}
203 \catcode'\-\active
204 \def-\{\futurelet\temp\indexdash}
205 \def\indexdash{\ifx\temp-\endash\fi}
206
207 \PrintIndex
208 \endgroup

```

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

```

209 \let\PrintChanges\relax
210 \let\PrintIndex\relax
211 \end{document}
212 \pdoc

```

C Additional Utility Programs

C.1 Shell Script `mkpldoc.sh`

A shell script to process ‘`pdoc.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.

```

213 \shprog
214 \ja)rm -f pdoc.toc pdoc.idx pdoc.glo
215 \en)rm -f pdoc-en.toc pdoc-en.idx pdoc-en.glo

```

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

```

216 echo "" > ltxdoc.cfg

```

Now process `pdoc.tex`.

```

217 \ja)platex pdoc.tex
218 \en)platex -jobname=pdoc-en pdoc.tex

```

Make the Change log and Glossary (Change History) using `mendex`. ‘`Mendex`’ is a Japanese index processor, which is mostly upward 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.)

```

219 <ja>mendex -s gind.ist -d pldoc.dic -o pldoc.ind pldoc.idx
220 <en>mendex -s gind.ist -d pldoc.dic -o pldoc-en.ind pldoc-en.idx
221 <ja>mendex -f -s gglo.ist -o pldoc.gls pldoc.glo
222 <en>mendex -f -s gglo.ist -o pldoc-en.gls pldoc-en.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.

```

223 echo "\includeonly{" > ltxdoc.cfg
224 <ja>platex pldoc.tex
225 <en>platex -jobname=pldoc-en pldoc.tex

```

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

```

226 echo "" > ltxdoc.cfg
227 <ja>platex pldoc.tex
228 <en>platex -jobname=pldoc-en pldoc.tex
229 # EOT
230 </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.

```

231 <*plprog>
232 ##
233 ## DOCSTRIP 文書内の環境や条件の入れ子を調べる perl スクリプト
234 ##
235 push(@dst,"DUMMY"); push(@dst,"000");
236 push(@env,"DUMMY"); push(@env,"000");
237 while (<>) {
238   if (/^%<\*(\[>]+\>/) { # check conditions
239     push(@dst,$1);
240     push(@dst,$.);
241   } elsif (/^%<\\[>+\>/) {
242     $linenum = pop(@dst);
243     $conditions = pop(@dst);
244     if ($1 ne $conditions) {
245       if ($conditions eq "DUMMY") {
246         print "$ARGV: '</$1>' (1.$.) is not started.\n";
247         push(@dst,"DUMMY");
248         push(@dst,"000");
249       } else {
250         print "$ARGV: '<*$conditions>' (1.$linenum) is ended ";

```

```

251     print "by '<*$1>' (l.$.)\n";
252   }
253 }
254 }

255 if (/^% *\\begin\{verbatim\}/) { # check environments
256   while(<>) {
257     last if (/^% *\\end\{verbatim\}/);
258   }
259 } elsif (/^% *\\begin\{([^-}]+)\}\{(.*)\}/) {
260   push(@env,$1);
261   push(@env,$.);
262 } elsif (/^% *\\begin\{([^-}]+)\}/) {
263   push(@env,$1);
264   push(@env,$.);
265 } elsif (/^% *\\end\{([^-}]+)\}/) {
266   $linenum = pop(@env);
267   $environment = pop(@env);
268   if ($1 ne $environment) {
269     if ($environment eq "DUMMY") {
270       print "$ARGV: '\\end{$1}' (l.$.) is not started.\n";
271       push(@env,"DUMMY");
272       push(@env,"000");
273     } else {
274       print "$ARGV: \\begin{$environment} (l.$linenum) is ended ";
275       print "by \\end{$1} (l.$.)\n";
276     }
277   }
278 }

279 }

280 $linenum = pop(@dst);
281 $conditions = pop(@dst);
282 while ($conditions ne "DUMMY") {
283   print "$ARGV: '<*$conditions>' (l.$linenum) is not ended.\n";
284   $linenum = pop(@dst);
285   $conditions = pop(@dst);
286 }

287 $linenum = pop(@env);
288 $environment = pop(@env);
289 while ($environment ne "DUMMY") {
290   print "$ARGV: '\\begin{$environment}' (l.$linenum) is not ended.\n";
291   $linenum = pop(@env);
292   $environment = pop(@env);
293 }
294 exit;
295 </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.

```
296 ⟨*Xins⟩
297 \input docstrip
298 \keepsilent
299 {\catcode'#=12 \gdef\MetaPrefix{## }}
300 \declarepreamble\thispre
301 \endpreamble
302 \usepreamble\thispre
303 \declarepostamble\thispost
304 \endpostamble
305 \usepostamble\thispost
306 \generate{
307   \file{dstcheck.pl}{\from{platex.dtx}{plprog}}
308   \file{mkpldoc.sh}{\from{platex.dtx}{shprog,ja}}
309   \file{mkpldoc-en.sh}{\from{platex.dtx}{shprog,en}}
310 }
311 \endbatchfile
312 ⟨/Xins⟩
```

References

- [1] 中野 賢 『日本語 L^AT_εX 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	2016/08/26 v1.0m
first edition 2	Moved loading <code>platex.cfg</code> from
1995/08/25 v1.0a	<code>plcore.ltx</code> to <code>platex.ltx</code> . . . 3
Added 'Compatibility', 'Usage of	2016/09/14 v1.0n
DOCSTRIP' and 'References' . . . 2	Improved banner saving method . . 3
1996/02/01 v1.0b	2017/09/24 v1.0o
Adjusted for the latest DOCSTRIP	Allow negative patch level for
(<code>omake-sh.ins</code> and	pre-release 10
<code>omake-pl.ins</code> 15	2017/11/11 v1.0p
1997/01/23 v1.0c	Moved banner saving code from
Adjusted for the latest DOCSTRIP. 15	<code>platex.ltx</code> to <code>plcore.ltx</code> . . . 3
Don't copy <code>gind.ist</code> and <code>gglo.ist</code>	2017/12/02 v1.0r
from	English references added 2
<code>\$TEXMF/tex/latex2e/base</code>	2017/12/05 v1.0s
directory. 12	Moved loading default settings
1997/01/25 v1.0c	from <code>plcore.ltx</code> to
Add to <code>filecontents</code> environment	<code>platex.ltx</code> 3
for <code>pldoc.dic</code> 7	2018/02/07 v1.0t
1997/01/29 v1.0c	Moved <code>ascmac</code> package to separate
Rename <code>pltpatch.ltx</code> to	bundle 6
<code>plpatch.ltx</code> 10	2018/02/18 v1.0u
2016/01/27 v1.0d	Moved <code>nidanfloat</code> package to
Add <code>-e</code> test before <code>rm</code> command . 12	separate bundle 6
Updated descriptions of pL ^A T _ε E _X	2018/04/06 v1.0v
files 5	Sync with the latest <code>source2e.tex</code> 9
2016/02/16 v1.0e	2018/04/08 v1.0w
Add a description of <code>platexrelease</code> 7	Stop showing banner during
2016/04/12 v1.0f	format generation for safety . . . 3
Update document. 1	2018/09/03 v1.0x
2016/05/07 v1.0g	Mention <code>platexcheat</code> (Japanese
Save L ^A T _ε E _X banner 3	only). 2
2016/05/08 v1.0h	Mention <code>plautopatch</code> 6
Exclude <code>plpatch.ltx</code> from the	Update document. 1
document 10	2018/09/22 v1.0y
2016/05/12 v1.0i	Show last update info on
Undefine temporary command	<code>pldoc.pdf</code> 10
<code>\orgdump</code> in the end. 3	2019/09/29 v1.0z
2016/05/20 v1.0j	Fix typos in document. 1
Add description of 'pfltrace' 5	2020/03/24 v1.1
2016/05/21 v1.0k	Update document. 1
Print also changes. 1	2020/09/26 v1.1a
2016/06/19 v1.0l	Add <code>plexpl3.dtx</code> 11
Get the patch level from	2020/09/28 v1.1b
<code>plvers.dtx</code> 10	Add hook after loading defs 3