About upLaTeX 2$_\varepsilon$

Ken Nakano & Japanese TeX Development Community & TTK

Date: 2019/05/22

upLaTeX is a Unicode version of Japanese pLaTeX 2$_\varepsilon$. This version is based on 'pLaTeX 2$_\varepsilon$ Community Edition.'

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

- The Character set available is limited to JIS X 0208, namely JIS level-1 and level-2
- Difficulty in handlign 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 pTeX, upTeX, has been developed.[1] The Unicode pLaTeX format run on upTeX is called upLaTeX. Current upLaTeX is maintained by Japanese TeX Development Community,[2] in sync with pLaTeX community edition.[3] It runs on ε-upTeX, an engine with both upTeX and ε-pTeX features.

The development version is available from GitHub repository[4]. Any bug reports and requests should be sent to Japanese TeX Development Community, using GitHub Issue system.

[1] `http://www.t-lab.opal.ne.jp/tex/uptex.html`
[2] `https://texjp.org`
[3] `https://github.com/texjporg/platex`
[4] `https://github.com/texjporg/uplatex`

1 Introduction to this document

This document briefly describes upLaTeX 2ε, but is not a manual of upLaTeX 2ε. The basic functions of upLaTeX 2ε are almost the same with those of pLaTeX 2ε and LaTeX 2ε, so please refer to the documentation of those formats.

For upTeX, 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 upLaTeX 2ε. Also describes the standard classes and packages.

Section 3 The compatibility note for users of the old version of upLaTeX 2ε or those of the original pLaTeX 2ε/LaTeX 2ε.

Appendix A Describes DOCSTRIP Options for this document.

Appendix B Description of 'upldoc.tex' (counterpart for 'source2e.tex' in LaTeX 2ε).

Appendix C Description of a shell script to process 'upldoc.tex', etc.

2 About Functions of pLaTeX 2ε

The structure of upLaTeX 2ε is similar to that of pLaTeX 2ε; it consists of 3 types of files: a format (uplatex.ltx), classes and packages.

2.1 About the Format

To make a format for upLaTeX, process "uplatex.ltx" with INI mode of ε-upTeX.[5] A handy command 'fmtutil-sys' (or 'fmtutil') for this purpose is available in TeX Live. The following command generates `uplatex.fmt`.

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

The content of `uplatex.ltx` is shown below. In the current version of upLaTeX, first we simply load `latex.ltx` and modify/extend some definitions by loading `plcore.ltx` (available from pLaTeX) and `uplcore.ltx`.

1 ⟨*plcore⟩

[5] Formerly both upTeX and ε-upTeX can make the format file for upLaTeX, however, it's not true anymore because LaTeX requires ε-TeX since 2017.

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

```
\let\orgdump\dump
\let\dump\relax
```

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

```
\input latex.ltx
```

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

```
\typeout{*************************^^J%
         *^^J%
         * making upLaTeX format^^J%
         *^^J%
         ************************}
\makeatletter
\input plcore.ltx
\input uplcore.ltx
```

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

```
\InputIfFileExists{upldefs.cfg}
           {\typeout{************************************^^J%
                     * Local config file upldefs.cfg used^^J%
                     ************************************}}%
           {\input{upldefs.ltx}}
```

In the previous version, we displayed upLaTeX 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.

```
%\the\everyjob
```

Load `uplatex.cfg` if it exists at runtime of upLaTeX 2_ε. (Counterpart of `platex.cfg` in pLaTeX 2_ε.)

```
\everyjob\expandafter{%
  \the\everyjob
  \IfFileExists{uplatex.cfg}{%
    \typeout{*************************^^J%
             * Loading uplatex.cfg.^^J%
             *************************}%
    \input{uplatex.cfg}}{}%
}
```

Dump to the format file.

```
\let\dump\orgdump
\let\orgdump\@undefined
\makeatother
\dump
%\endinput
```

32 ⟨/plcore⟩

The file `uplcore.ltx`, which provides modifications/extensions to make upLaTeX 2ε, is a concatenation of stripped files below using DOCSTRIP program.

- `uplvers.dtx` defines the format version of upLaTeX 2ε.
- `uplfonts.dtx` extends NFSS2 for Japanese font selection.
- `plcore.dtx` (the same content as pLaTeX 2ε); defines other modifications to LaTeX 2ε.

Moreover, default settings of pre-loaded fonts and typesetting parameters are done by loading `upldefs.ltx` inside `uplatex.ltx`.[6] This file `upldefs.ltx` is also stripped from `uplfonts.dtx`.

Attention:
You can customize upLaTeX 2ε 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 upLaTeX is named after pLaTeX ones, prefixed with "u."

2.1.1 Version

The version (like "2019-04-06u02") and the format name ("pLaTeX2e") of upLaTeX 2ε are defined in `uplvers.dtx`. This is similar to pLaTeX 2ε, which defines those in `plvers.dtx`.

2.1.2 NFSS2 Commands

upLaTeX 2ε shares `plcore.dtx` with pLaTeX 2ε, so the extensions of NFSS2 for selecting Japanese fonts are available.

2.1.3 Output Routine and Floats

upLaTeX 2ε shares `plcore.dtx` with pLaTeX 2ε, so the output routine and footnote macros will behave similar to pLaTeX 2ε.

[6] Older upLaTeX loaded `upldefs.ltx` inside `uplcore.ltx`; however, upLaTeX community edition newer than 2018 loads `upldefs.ltx` inside `uplatex.ltx`.

2.2 Classes and Packages

Classes and packages bundled with upLaTeX 2$_\varepsilon$ are based on those in original pLaTeX 2$_\varepsilon$, and modified some parameters.

upLaTeX 2$_\varepsilon$ classes:

- ujarticle.cls, ujbook.cls, ujreport.cls

  Standard *yoko-kumi* (horizontal writing) classes; stripped from `ujclasses.dtx`. upLaTeX 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`. upLaTeX edition of tarticle.cls, tbook.cls and treport.cls.

We don't provide upLaTeX edition of jltxdoc.cls, but the one from pLaTeX can be used also on upLaTeX without problem.

upLaTeX 2$_\varepsilon$ packages:

- uptrace.sty

  upLaTeX 2$_\varepsilon$ version of `tracefnt.sty`; the package `tracefnt.sty` overwrites upLaTeX 2$_\varepsilon$-style NFSS2 commands, so `uptrace.sty` provides redefinitions to recover upLaTeX 2$_\varepsilon$ extensions. Stripped from `uplfonts.dtx`.

Other pLaTeX packages work also on upLaTeX.

3 Compatibility with Other Formats and Older Versions

Here we provide some information about the compatibility between current upLaTeX 2$_\varepsilon$ and older versions or original pLaTeX 2$_\varepsilon$/LaTeX 2$_\varepsilon$.

3.1 Compatibility with pLaTeX 2$_\varepsilon$/LaTeX 2$_\varepsilon$

upLaTeX 2$_\varepsilon$ is in most part upper compatible with pLaTeX 2$_\varepsilon$, so you can move from pLaTeX 2$_\varepsilon$ to upLaTeX 2$_\varepsilon$ by simply replacing the document class and some macros. However, the default Japanese font metrics in upLaTeX 2$_\varepsilon$ is different from those in pLaTeX 2$_\varepsilon$; therefore, you should not expect identical output from both pLaTeX 2$_\varepsilon$ and upLaTeX 2$_\varepsilon$.

Note that upLaTeX is a new format, so we do *not* provide support for 2.09 compatibility mode. Follow the standard LaTeX 2_ε convention!

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

3.2 Support for Package 'latexrelease'

pLaTeX provides 'platexrelease' 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 upLaTeX, but currently we don't need it; upLaTeX does not have any recent upLaTeX-specific changes. So, you can safely use 'platexrelease' 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:

Option	*Function*
plcore	Generates a fragment of format sources
pldoc	Generates 'upldoc.tex' for typesetting upLaTeX 2_ε 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 upLaTeX 2_ε sources

The contents of 'upldoc.tex' for typesetting upLaTeX 2_ε sources is described here. Compared to individual processings, batch processing using 'upldoc.tex' prints also changes and an index.

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

```
\newif\ifJAPANESE
```

as `uplatex.cfg`, and process `upldoc.tex` (upLaTeX 2$_\varepsilon$ newer than July 2016 is required).

Here we explain only difference between `pldoc.tex` (pLaTeX 2$_\varepsilon$) and `upldoc.tex` (upLaTeX 2$_\varepsilon$).

```
⟨*pldoc⟩
\begin{filecontents}{upldoc.dic}
西暦　　せいれき
和暦　　われき
\end{filecontents}
```

The document of pLaTeX 2$_\varepsilon$ requires `plext` package, since `plext.dtx` contains several examples of partial vertical writing. However, we don't have such examples in upLaTeX 2$_\varepsilon$ files, so no need for it.

```
\documentclass{jltxdoc}
%\usepackage{plext} %% comment out for upLaTeX
\listfiles

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

\ifJAPANESE
\IndexPrologue{\part*{索 引}%
                \markboth{索 引}{索 引}%
                \addcontentsline{toc}{part}{索 引}%
イタリック体の数字は、その項目が説明されているページを示しています。
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。
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。}
\else
\IndexPrologue{\part*{Index}%
                \markboth{Index}{Index}%
                \addcontentsline{toc}{part}{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.}
\fi
%
\ifJAPANESE
\GlossaryPrologue{\part*{変更履歴}%
                  \markboth{変更履歴}{変更履歴}%
                  \addcontentsline{toc}{part}{変更履歴}}
\else
\GlossaryPrologue{\part*{Change History}%
                  \markboth{Change History}{Change History}%
                  \addcontentsline{toc}{part}{Change History}}
\fi

\makeatletter
\def\changes@#1#2#3{%
  \let\protect\@unexpandable@protect
  \edef\@tempa{\noexpand\glossary{#2\space
               \currentfile\space#1\levelchar
               \ifx\saved@macroname\@empty
                  \space\actualchar\generalname
               \else
                  \expandafter\@gobble
                  \saved@macroname\actualchar
                  \string\verb\quotechar*%
                  \verbatimchar\saved@macroname
                  \verbatimchar
               \fi
               :\levelchar #3}}%
  \@tempa\endgroup\@esphack}
\renewcommand*\MacroFont{\fontencoding\encodingdefault
                   \fontfamily\ttdefault
                   \fontseries\mddefault
                   \fontshape\updefault
                   \small
                   \hfuzz 6pt\relax}
\renewcommand*\l@subsection{\@dottedtocline{2}{1.5em}{2.8em}}
\renewcommand*\l@subsubsection{\@dottedtocline{3}{3.8em}{3.4em}}
\makeatother
\RecordChanges
\CodelineIndex
\EnableCrossrefs
\setcounter{IndexColumns}{2}
\settowidth\MacroIndent{\ttfamily\scriptsize 000\ }
```

Set the title, authors and the date for this document.

```
 \title{The \upLaTeXe\ Sources}
 \author{Ken Nakano \& Japanese \TeX\ Development Community \& TTK}

% Get the (temporary) date and up-patch level from uplvers.dtx
\makeatletter
```

```
\let\patchdate=\@empty
\begingroup
   \def\ProvidesFile#1[#2 #3]#4\def\uppatch@level#5{%
      \date{#2}\xdef\patchdate{#5}\endinput}
   \input{uplvers.dtx}
\endgroup

% Add the patch version if available.
\def\Xpatch{}
\ifx\patchdate\Xpatch\else
  \edef\@date{\@date\space version \patchdate}
\fi

% Obtain the last update info, as upLaTeX does not change format date
% -> if successful, reconstruct the date completely
\def\lastupd@te{0000/00/00}
\begingroup
   \def\ProvidesFile#1[#2 #3]{%
      \def\@tempd@te{#2}\endinput
      \@ifl@t@r{\@tempd@te}{\lastupd@te}{%
         \global\let\lastupd@te\@tempd@te
      }{}}
   \let\ProvidesClass\ProvidesFile
   \let\ProvidesPackage\ProvidesFile
   \input{uplvers.dtx}
   \input{uplfonts.dtx}
   \input{ukinsoku.dtx}
   \input{ujclasses.dtx}
\endgroup
\@ifl@t@r{\lastupd@te}{0000/00/00}{%
  \date{Version \patchdate\break (last updated: \lastupd@te)}%
}{}
\makeatother
```

Here starts the document body.

```
\begin{document}
\pagenumbering{roman}
\maketitle
\renewcommand\maketitle{}
\tableofcontents
\clearpage
\pagenumbering{arabic}

\DocInclude{uplvers}   % upLaTeX version

\DocInclude{uplfonts}  % NFSS2 commands

\DocInclude{ukinsoku}  % kinsoku parameter

\DocInclude{ujclasses} % Standard class
```

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

\clearpage
\pagestyle{headings}
% Make TeX shut up.
\hbadness=10000
\newcount\hbadness
\hfuzz=\maxdimen
%
\PrintChanges
\clearpage
%
\begingroup
  \def\endash{--}
  \catcode`\-\active
  \def-{\futurelet\temp\indexdash}
  \def\indexdash{\ifx\temp-\endash\fi}

  \PrintIndex
\endgroup
\let\PrintChanges\relax
\let\PrintIndex\relax
\end{document}
⟨/pldoc⟩
```

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.

```
⟨*shprog⟩
⟨ja⟩rm -f upldoc.toc upldoc.idx upldoc.glo
⟨en⟩rm -f upldoc-en.toc upldoc-en.idx upldoc-en.glo
echo "" > ltxdoc.cfg
⟨ja⟩uplatex upldoc.tex
⟨en⟩uplatex -jobname=upldoc-en 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.[7]

```
⟨ja⟩mendex -U -s gind.ist -d upldoc.dic -o upldoc.ind upldoc.idx
```

[7] The 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.

```
⟨en⟩mendex -U -s gind.ist -d upldoc.dic -o upldoc-en.ind upldoc-en.idx
⟨ja⟩mendex -U -f -s gglo.ist -o upldoc.gls upldoc.glo
⟨en⟩mendex -U -f -s gglo.ist -o upldoc-en.gls upldoc-en.glo
echo "\includeonly{}" > ltxdoc.cfg
⟨ja⟩uplatex upldoc.tex
⟨en⟩uplatex -jobname=upldoc-en upldoc.tex
echo "" > ltxdoc.cfg
⟨ja⟩uplatex upldoc.tex
⟨en⟩uplatex -jobname=upldoc-en upldoc.tex
# EOT
⟨/shprog⟩
```

C.2 Perl Script `dstcheck.pl`

The one from pLaTeX 2_ε can be use without any change, so omitted here in upLaTeX 2_ε.

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 2_ε.

```
⟨*Xins⟩
\input docstrip
\keepsilent
{\catcode`#=12 \gdef\MetaPrefix{## }}
\declarepreamble\thispre
\endpreamble
\usepreamble\thispre
\declarepostamble\thispost
\endpostamble
\usepostamble\thispost
\generate{
   \file{mkpldoc.sh}{\from{uplatex.dtx}{shprog,ja}}
   \file{mkpldoc-en.sh}{\from{uplatex.dtx}{shprog,en}}
}
\endbatchfile
⟨/Xins⟩
```

References

[1] Takuji Tanaka, UpTEX — Unicode version of pTEX with CJK extensions TUGboat issue 34:3, 2013.
(http://tug.org/TUGboat/tb34-3/tb108tanaka.pdf)

Change History