

2. T_EX と L^AT_EX

まず、T_EX と L^AT_EX は異なるものである。しかし、そもそも名前が似てゐるし、両者の区別が曖昧のままでも所望の文書を作成することはとりあえずできてしまうので、別にどっちでもいいという考え方でもできる。が、違いがあるから名前が違うのである。ここでは T_EX と L^AT_EX の何が違うのかを説明し、言葉として T_EX がつく様々なものについて、今後混乱・誤用しないようにざっくりと解説していく。

ところで、T_EX は T_EX であり、`tex` でも `TEX` でもなければ `Tex` でもない。しかし、場合によって `TeX` と表記されることはある。また、本資料では拡張子が `*.tex` となっているものについては `tex` ファイルなどと表記することにする。L^AT_EX も同様で、L^AT_EX かもしれないが `LaTeX` である。ちなみに、T_EX の発音はアルファベットの X /eks/ として読むのではなく、無声軟口蓋摩擦音 /x/ と発音するのが正しい（諸説あるが）*。

2.1. T_EX とはなにか

T_EX とは 1978 年に Donald E. Knuth^[1] が発表した“組版[†]ソフトウェアと組版言語”である^[3]。つまり、“文章そのもの”と“文章の構造を指定する命令”が書かれたテキストファイル（この命令を記す言語も T_EX という）を読み込み、その命令に従って文章を組版するソフトウェアが T_EX である。組版結果は DVI 形式 (DeVice-Independent) に書き出されるが、この `dvi` ファイルは文書の見た目のレイアウト（紙面のどの位置にどの文字を配置する、などの情報）を画像形式・表示デバイス・プリンタにまったく依存しない形で記録している^[4]。結局 T_EX をとてもざっくりまとめると（Knuth の思う）美しい組版がデバイスによらない形式で出力される、ということになる。

2.2. L^AT_EX とはなにか

T_EX での命令は「横方向に何センチメートルの空白を作る」のように非常に原始的なものである。使い勝手が良くない。そこで T_EX を使ってもっと簡単に論文やレポートを作成したいという要望から L^AT_EX は開発された^[5]。T_EX の原始的な命令を組み合わせる新しい命令（マクロ）を作ることができる（`\def` みたいなやつ）が、T_EX で文書作成するには多量のマクロが必要になるため、それらの必要なマクロが一通り揃えられた“マクロ体系”の 1 つが L^AT_EX である。

繰り返しになるが、この世にいくつも存在する T_EX を楽に使うためのマクロ体系のひとつが L^AT_EX である。現代の多くの `tex` ファイルはマクロが準備された L^AT_EX で処理されることが前提であり、それを T_EX で処理しようとして目的のものが得られることはまずない。次節で述べる通り、T_EX にも L^AT_EX にも沢山の種類がある。つまり、同じ L^AT_EX でも `pLATEX` と `LuaLATEX` とでは用意されているマクロや定義に違いがあるため、`tex` ファイルが想定するそれと処理する L^AT_EX が食い違っていればエラーが出るのも当然である。また、L^AT_EX で定義されたマクロのおかげで成

*日本語で表記するなら「てふ」であり、歴史的仮名遣い的にその発音は /tʃo:/ である。

[†]原稿及びレイアウトの指定に従って、文字・図版・写真などを配置する作業の総称^[2]。

り立つ事象を $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 全体で成り立つと理解するのは危険なのである。

2.3. その他のナツカ $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ とか $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ ナツカとかナツカ $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ ナツカとはなにか

$\mathrm{T}_{\mathrm{E}}\mathrm{X}$ と $\mathrm{L}_{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 以外にも名前に $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ がつくものは数多くある。違いがわかるように簡単にまとめておこう^[6]。

(1) $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の仲間

組版を行うソフトウェアとして、もともとの $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の機能が様々に拡張されたものがある。例えば、

- $\varepsilon\text{-}\mathrm{T}_{\mathrm{E}}\mathrm{X}$: オリジナルの $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ と 100% の互換性を保ちながら $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の少々不便なところを補う実用性の高い拡張。
- $\mathrm{p}\mathrm{T}_{\mathrm{E}}\mathrm{X}$: 縦組みに対応し、和文組版できる拡張 $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 。
- $\varepsilon\text{-}\mathrm{p}\mathrm{T}_{\mathrm{E}}\mathrm{X}$: $\varepsilon\text{-}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ を取り入れた $\mathrm{p}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 。
- $\mathrm{up}\mathrm{T}_{\mathrm{E}}\mathrm{X}$: $\mathrm{p}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の Unicode*版。つまり、 $\mathrm{p}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の内部コードを Unicode 化している。簡単に言ってしまうと扱える文字が増える。例えば Fig. 2.1 では同じ日本語で書かれた `tex` ファイルを $\mathrm{p}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ と $\mathrm{up}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ で処理した際の出力の違いである（正確には $\mathrm{p}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ に対応したマクロ体系 $\mathrm{pL}_{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ を利用した場合と $\mathrm{upL}_{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ を利用した場合の違い）。 $\mathrm{p}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の方では一部の文字（例えば踊り字「ゝ」や非常用漢字「冫灬」など）が出力されていないことがわかる。Unicode を用いることでこの世に存在するより幅広い文字を扱えることになる。
- $\mathrm{pdf}\mathrm{T}_{\mathrm{E}}\mathrm{X}$: $\varepsilon\text{-}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の拡張。DVI ではなく、PDF を直接出力する。少しだけ触れたが、`dvi` ファイルとはデバイスに依存しない、 $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ 固有の出力形式である。人間が読めるようになっていないわけではないため、別に PDF などに変換する操作（`dvipdfmx` とかそういうの）が必要である。 $\mathrm{pdf}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ は出力としてこの `dvi` ではなく `pdf` ファイルを直接出力するため、そういった変換も不要なほか、ハイパーテキストリンクや目次といった PDF の機能を直接扱うことができる。
- $\mathrm{X}_{\mathrm{E}}\mathrm{T}_{\mathrm{E}}\mathrm{X}(/zi:tex/)$ ^{[9],[10]}: $\varepsilon\text{-}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の拡張。符号空間を Unicode 全体に拡大したもの（上述 $\mathrm{up}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の説明参照）。また、OpenType フォントに対応している。これはつまり、OS にインストールされているフォントをそのまま利用することが可能ということである^[11]。もともとの $\mathrm{T}_{\mathrm{E}}\mathrm{X}$ では組版を行うとき、組み立てる文字の情報を `tfm` ファイルと呼ばれるものから参照する。したがって好きなフォントを使うためにはそのための `tfm` ファイルを用意する必要があったのだが、 $\mathrm{X}_{\mathrm{E}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ ではその必要はなく、インストールさえしてしまえば好きなフォントが使える。
- $\mathrm{Lua}\mathrm{T}_{\mathrm{E}}\mathrm{X}$: $\mathrm{pdf}\mathrm{T}_{\mathrm{E}}\mathrm{X}$ の後継。OpenType フォント、Unicode に対応しているほか、Lua という軽量スクリプト言語を利用できる。この Lua の利用によりあらゆるものがカスタマイズ可能となり、究極の柔軟性が得られる^[12]。(to-do) Lua を使ったおもちゃをいくつかとフォントの切り替えの楽しさの確認。

*様々な言語の文字を単一の文字コードで取り扱うために開発されている文字コード^{[7],[8]}

1 幸田露伴「雲のいろ——卿雲」

景雲といひ、卿雲といひ、慶雲といへる、しかと指し定められたる雲にはあらざるべし。卿雲爛たり糺縵々たり、といへる、煙にあらざる雲にあらざる紫を曳き光を流す、といへる、大人作矣、五色氤氲、といへる、金柯初めて繞繚、玉葉漸く氤氲、といへる、還つて九霄に入りてを成し、夕嵐生ずる處鶴松に歸る、といへる詩の句などによりて見れば、歸するところは美しき雲といふまでなり。

1 幸田露伴「雲のいろく——卿雲」

景雲といひ、卿雲といひ、慶雲といへる、しかと指し定められたる雲にはあらざるべし。卿雲爛たり糺縵々たり、といへる、煙にあらざる雲にあらざる紫を曳き光を流す、といへる、大人作矣、五色氤氲、といへる、金柯初めて繞繚、玉葉漸く氤氲、といへる、還つて九霄に入りて沆瀣を成し、夕嵐生ずる處鶴松に歸る、といへる詩の句などによりて見れば、歸するところは美しき雲といふまでなり。

Fig. 2.1. $\mathrm{pT}_{\mathrm{E}}\mathrm{X}$ engine (left) vs $\mathrm{upT}_{\mathrm{E}}\mathrm{X}$ engine (right).^[13]

October 27, 2022 ConTeXt template

4 built at four minutes to nine in the morning by Lua_{TeX} (on LuaH_{BT}_{EX})

少し話は変わるが、 $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ のバージョンについても名前がついている：

- $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\varepsilon}$: 現在 (2023-09-16) の主流.
- $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X} 2.09$: 古いバージョン.
- $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X} 3$: 開発中の次期バージョン.

ここまで、組版ソフトウェアの $\text{T}_{\text{E}}\text{X}$ とその拡張、 $\text{T}_{\text{E}}\text{X}$ を楽に使うためのマクロ体系の $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ とその拡張および仲間について説明した。ここからは $\text{T}_{\text{E}}\text{X}$ の名前がついていて、 $\text{T}_{\text{E}}\text{X}$ とともに使うもの*について見ていこう。

(3) 文献管理ソフト

`\cite` 等を使って文献の引用をする場合、そのタイトルや著者、出版年、出版社等々の文献情報を管理しなくてはならない。thebibliography 環境でひとつずつ (`\bibitem`) 自分の手で文献情報を入力することも可能だが、数が増えると大変である上、表記に一貫性を持つための調整も手間である。そんなときに、 $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ と組み合わせて文献データベースから自動的に参考文献リストを作るためのツールが文献管理ソフトである。よく耳にするのは

- Bib $\text{T}_{\text{E}}\text{X}$: 書誌情報のデータベース (bib ファイル) を用いて、自動で参考文献リストを作成する。このデータベース bib ファイルのなかで各書誌情報はまずエントリごとに種別分けされ (@article, book, proceedings, etc.)、それぞれの情報を保存する (title, author, journal, etc.)。
- pBib $\text{T}_{\text{E}}\text{X}$: p $\text{T}_{\text{E}}\text{X}$ に特化。日本語の文献を扱える。
- upBib $\text{T}_{\text{E}}\text{X}$

などがあるが、最近には他にも

- Bib $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ (+Biber): $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ で用いるパッケージ。そもそも Bib $\text{T}_{\text{E}}\text{X}$ は参考文献のフォーマットは bst ファイルというもので設定されるのだが、これまた編集が大変である。そのため、よりカスタマイズが容易なのが、本パッケージである。Bib $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ のバックエンド (文献のソートを担当) として biber か bibtex を用いることになるが、biber のほうが bibtex を用いるよりさらにカスタマイズの柔軟性が上がる^[15]。

が次世代文献管理ソフトとして挙げられる。

(4) 統合開発環境

$\text{T}_{\text{E}}\text{X}$ 周りの作業がしやすいように整備されたエディタ。これさえあれば $\text{T}_{\text{E}}\text{X}$ がつかえる、というものではない。

- $\text{T}_{\text{E}}\text{X}$ Shop: macOS 専用の $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ 統合環境。シンプルな UI と直感的な操作性が特徴。
- $\text{T}_{\text{E}}\text{X}$ works: $\text{T}_{\text{E}}\text{X}$ Shop をモデルにすべてのシステム上で実現すべく開発された $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ 用統合環境。
- $\text{T}_{\text{E}}\text{X}$ studio: 豊富な機能と高度な LaTeX サポートがあり、拡張性が高い。全システム上で動く。

* $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ は $\text{T}_{\text{E}}\text{X}$ “を” 使うもの

(5) ディストリビューション

TeX がまともに使えるようになるためには TeX さえあればいい，わけではない．TeX なのか pTeX なのか LuaTeX なのか，L^ATeX なのか ConTeXt なのか，BibTeX なのか Biber なのか，統合開発環境は使うのか，使うならどれか，さらに言えば，自分の文書作成では LuaL^ATeX のみ使うと決心したとしても，学会によっては upL^ATeX で動くことのみ想定したフォーマットを配っている場合もあり，そのフォーマットを LuaL^ATeX 用書き換えるのはただ面倒で生産性があまりなく，こうなると最初から upL^ATeX を準備しておくのが望ましい．これ以外にも，特定のマクロ体系を更に拡張するための“パッケージ”が必要なことがほとんどで，それらも新たに必要になるたびインストールしなくてはならない．結局，TeX を使えるようになるためには非常に多くのものをインストールしなくてはならないが，一つ一つ個別にインストールするのは大変であり，これらを一括でインストールできるのがディストリビューションである．

日本国内で多く用いられているディストリビューションは

- TeXLive^[16]

であるが，他にも

- MacTeX: TeXLive をベースにした macOS 専用のディストリビューション．
- BasicTeX: L^ATeX の文書作成に必要最低限なものが含まれる macOS 専用ディストリビューション．ε-pTeX などが含まれない．
- MiKTeX: Windows, macOS, Linux で動作するディストリビューション．日本語 pTeX が含まれない．

などが存在する．

2.4. 番外編：文書クラス

さて，TeX と名のつくものについて説明してきたが，ここで TeX を名前に含まないものの，種類がいくつかあってややこしいものとして，文書クラスについて解説する^{documentclass_list}．tex ファイルの一番最初に `\documentclass` とか書くあれである．

- **article**: 論文やレポートなどの一般的な文書．文章の構造は `part`, `section`, `subsection`, `subsubsection`, `paragraph`, `subparagraph` の 7 つのレベルに分けられる．
- **report**: 論文やレポートのより長い文書．構造としては **article** に `chapter` が追加される (`part` と `section` の間)．また，(デフォルトの設定では) `part` が 1 ページ使って表示される (Part I ○○とだけ書かれたページができる)．
- **book**: 書籍．構造は **report** と同じだが，奇数ページと偶数ページのレイアウトが変わる．例えば，`part` の示すページは必ず奇数ページ (見開きの右側) になる．

- `jarticle`: 和文横書き対応の `article`.
- `jreport`
- `jbook`
- `tarticle`: 和文縦書き対応の `article`.
- `jsclasses` (`jsarticle` etc.): 日本語 (u)p \LaTeX 用の新しい文書クラス^{jscls_doc}. JIS フォントメトリックを使用している. 簡単に言えば「文字の組み方」が綺麗に, 読みやすくなる. 例えば小さい「ちょっとチェック」のように小さい文字を並べて表示する際 `jarticle` などの min10 フォントメトリックでは不自然に詰められ, 美しくなかったり読みづらかったりするものが `jsclasses` では解消される^{LaTeX_are}. 他にも日本語中の英語の組み方などに差が出る.
- `ltjsclasses` (`ltjsarticle` etc.): `jsclasses` を Lua \LaTeX 用に改変したもの.
- `bxjsclasses` (`bxjsarticle` etc.): `jsclasses` の派生で, (u)p \LaTeX 依存の部分を切り離したもの. つまり, Lua \LaTeX など他のエンジンを用いても綺麗な日本語文書を作成できる.
- `jlreq`: 「日本語組版処理の要件」*を満たす Lua \LaTeX , (u)p \LaTeX 用の文書クラス.
- `beamer`: スライド. 「Beamer のすゝめ」も読んでね.

References

- [1] “Don Knuth’s home page.” URL: <https://www-cs-faculty.stanford.edu/~knuth/> (visited on Oct. 26, 2022) (cit. on p. 1).
- [2] “組版 - Wikipedia.” URL: <https://ja.wikipedia.org/wiki/%E7%B5%84%E7%89%88> (visited on Oct. 25, 2022) (cit. on p. 1).
- [3] Arthur Reutenauer. “A brief history of T E X, volume II.” In: *EuroBachTeX 2007* (2008), p. 68 (cit. on p. 1).
- [4] “ TEX と $\text{L}\text{A}\text{T}\text{E}\text{X}$ の違い - ラング・ラグー.” URL: <https://blog.wtsnjp.com/2016/12/19/tex-and-latex/> (visited on Oct. 26, 2022) (cit. on p. 1).
- [5] “ $\text{L}\text{A}\text{T}\text{E}\text{X}$ - TEX Wiki.” URL: <https://texwiki.texjp.org/LaTeX#f2915fa5> (visited on Oct. 26, 2022) (cit. on p. 1).
- [6] “ TEX Wiki.” URL: <https://texwiki.texjp.org/> (visited on Oct. 25, 2022) (cit. on p. 2).
- [7] “Unicode - Wikipedia.” URL: <https://ja.wikipedia.org/wiki/Unicode> (visited on Oct. 26, 2022) (cit. on p. 2).
- [8] 師茂樹. “Unicode とのつきあい方—漢字文化圏を中心に—.” In: コンピュータ & エデュケーション 27 (2009), pp. 12–17 (cit. on p. 2).
- [9] “ $\text{X}\text{E}\text{L}\text{A}\text{T}\text{E}\text{X}$ - Wikipedia.” URL: <https://ja.wikipedia.org/wiki/XeTeX> (visited on Oct. 26, 2022) (cit. on p. 2).

*標準化団体 W3C (World Wide Web Consortium) の技術ノート “Requirements for Japanese Text Layout”^{Req4JapTex, JIS_X_4051} のことで, 字詰めやルビ, ページ構成や図表に関してなどあらゆる規定が書かれている.

REFERENCES

- [10] “Xe_ƒTeX - TeX Wiki.” URL: <https://texwiki.texjp.org/?XeTeX> (visited on Oct. 27, 2022) (cit. on p. 2).
- [11] “Xe_ƒLaTeX で日本語する件について.” URL: <http://zrbabbler.sp.land.to/xelatex.html> (visited on Oct. 26, 2022) (cit. on p. 2).
- [12] 八登崇之. “日本人の知らない TeX” 2010 (cit. on p. 2).
- [13] 幸田露伴. “雲のいろいろ.” 反省雑誌, 1987 (cit. on p. 3).
- [14] “ConTeXt - TeX Wiki.” URL: <https://texwiki.texjp.org/?ConTeXt> (visited on Oct. 26, 2022) (cit. on p. 4).
- [15] “Biber (LaTeX) - Wikipedia.” URL: [https://en.wikipedia.org/wiki/Biber_\(LaTeX\)](https://en.wikipedia.org/wiki/Biber_(LaTeX)) (visited on Oct. 30, 2022) (cit. on p. 5).
- [16] “TeX Live をホンキで語る - Acetaminophen’s diary.” URL: <https://acetaminophen.hatenablog.com/entry/texadvent2016-20161205#main-7> (visited on Oct. 27, 2022) (cit. on p. 6).