

# Lua $\text{\LaTeX}$ & Bib $\text{\LaTeX}$ with Debian

環境構築と作業手順の解説

© [ru\\_museum](#)(GitHub)

2022 年 2 月 3 日

# Debian で LuaL<sup>A</sup>T<sub>E</sub>X を使用する

## 環境構築

- インストールパッケージ  
texlive-full は容量大なので必要なもののみを選択。  
**texlive**  
**texlive-lang-japanese**  
**texlive-luatex**  
**texlive-bibtex-extra** (BibLatex)  
**texlive-extra-utils** (PythonTex)
- luatex は直接 PDF を作成するので pdftex 等は不要。
- Debian では texlive-bibtex-extra に **biblatex** が同梱されています。

## 作業環境

### gedit の LaTeX プラグイン

- LaTeX Plugin では機能不具合により「外部ツール」を使用します。
  - PDF Viewer の Atril 又は evince 等を入れて置きます。
1. パッケージをインストール  
**gedit-latex-plugin**
  2. gedit :  
「設定」→「プラグイン」→「外部ツール」にチェックを入れます。

3. メニューに表示された「Manege External Tools(外部ツールの設定)」→「外部ツールの管理」に登録します。

ツール名：**Build LuaLatex**

sh スクリプト：

```
#!/bin/sh
```

```
lualatex $GEDIT_CURRENT_DOCUMENT_NAME
```

ショートカットキー：**Alt + F5**



図 1: 外部ツールの設定

- 「ショートカットキー」は設定が重複していなければ自由に登録出来ます。
  - 設定は ” /.config/gedit/tools/build-lualatex ” に登録されます。
4. 登録した「Alt + F5」でビルドをします。  
処理状況は下部 **Tool Output** に表示され、**Done** と表示されれば正常終了です。

5. 同時に PDF ファイル を開くコマンドも登録します。

ツール名: **Open PDF**

sh スクリプト:

```
#!/bin/sh
FILENAME="$GEDIT_CURRENT_DOCUMENT_NAME"
PDF_FILE="$(basename "$FILENAME" .tex).pdf"
# atril or evince で開く
atril $PDF_FILE
```

ショートカットキー: **Shift + Alt + P**

- 初回もし PDF ファイル が生成されていれば  
「Shift + Alt + P」で開かれます。
- 2 度目のビルド以降は変更が自動で反映されます。

6. 次に、ビルド時 **biber** 用の .bcf ファイルが既に生成されていれば同時にコンパイルする様に **Build LuaLatex** を修正します。

- この設定は「4 Bib $\text{\LaTeX}$  + biber」を導入済みの場合です。
- 詳しくは「4 Bib $\text{\LaTeX}$  + biber」をご覧ください。
- **biber** は Bib $\text{\LaTeX}$  用のコンパイラーです。

ツール名: **Build LuaLatex**

sh スクリプト:

```
#!/bin/sh
# .tex ファイル名
FILENAME="$GEDIT_CURRENT_DOCUMENT_NAME"

# .bcf ファイル名 (BibLatex)
FILE_BCF="$(basename "$FILENAME" .tex).bcf"

# .tex をビルド
lualatex $FILENAME

# biber での .bcf コンパイル (BibLatex)
if [ -e $FILE_BCF ]; then
```

```

echo "BCF File をコンパイルしています -----"
# "File exists."
biber $FILE_BCF

echo "BCF File のコンパイルを終了しました -----"

lualatex $FILENAME

fi

```

## VSCodet の LaTeX プラグイン

1. LaTeX Workshop をインストール
2. settings.json に追記する。

```

"latex-workshop.latex.recipes": [
{
  "name": "lualatex",
  "tools": ["lualatex"]
}
],
"latex-workshop.latex.tools": [
{
  "name": "lualatex",
  "command": "lualatex",
  "args": [
    "--cmdx",
    "-file-line-error",
    "-synctex=1",
    "-interaction=nonstopmode", % エラー中断の回避 (重要)
    "-halt-on-error",
    "\\%DOC\\%"
  ],
  "env": {}
}
]

```

## SVG 画像の表示

※ "Inkscape" のインストールが必要です。

1. `\usepackage{svg}`  
`\usepackage{float} % [H]`  
`\begin{figure}[H]`  
`\centering`  
`\includesvg[width=6cm]{./filename.svg}`  
`\end{figure}`

2. gedit 【外部コマンド】に追加します。

`lualatex -shell-escape $FILENAME`

※ "Inkscape export" で変換してくれます。

以下のエラーが表示された場合：

---

Package svg Warning: You didn't enable 'shell escape' (or 'write18')  
(svg) so it wasn't possible to launch the Inkscape export

3. **Example SVG:**



図 2: L<sup>A</sup>T<sub>E</sub>X Logo: ウィキメディア・コモンズ

## PythonT<sub>E</sub>X の利用

- pythontex により Python スクリプトを動作させます。  
※ Ruby や Javascript その他も可能。
- PythonT<sub>E</sub>X は texlive-extra-utils に含まれています。
- ビルドは LuaLatex ⇒ pythontex ⇒ LuaLatex で行います。
- 初回のビルド lua<sub>l</sub>atex sample.tex を行うと **sample.pytxcode** が生成されます。
- ビルド時 **pythontex** 用の .pytxcode ファイルが生成されていれば LuaLatex のコンパイルが動作する様に「外部コマンド」に追加します。

ツール名: **Build LuaLatex**

sh スクリプト追加部分:

```
FILE_PYTHON=`basename "$FILENAME" .tex`.pytxcode"

# PYTHONTEX
if [ -e $FILE_PYTHON ]; then

    echo "PYTHONTEX File をコンパイルしています -----"
    # "File exists."
    pythontex $FILE_PYTHON

    echo "PYTHONTEX File のコンパイルを終了しました -----"

    lualatex $FILENAME

fi
```

- 「Alt + F5」のビルドで Python スクリプトが PDF へ反映されます。

## BibL<sup>A</sup>T<sub>E</sub>X + biber の導入

- Debian には BibLatex パッケージは無く  
texlive-bibtex-extra に含まれます。  
/usr/share/texlive/texmf-dist/tex/latex/biblatex
- biber(参考文献処理ライブラリー) をインストール。

### 【使用例】

```
\usepackage[  
    backend=biber,  
    bibstyle=ieee,  
{biblatex}  
  
\addbibresource{data.bib}  
\printbibliography[title=参考文献]
```

biber でのコンパイルの流れ

- backend=biber の設定で .bcf ファイルが生成されます。

1. lualatex test.tex   % ビルド
2. biber test.bcf       % biber でコンパイル
3. lualatex test.tex   % 再ビルド

- 外部ツールに登録し自動化する（「Build LuaTeX」の修正）  
.bcf ファイルが存在すれば biber を動作させる。

**参照：「gedit の LuaL<sup>A</sup>T<sub>E</sub>X プラグイン：6」**

- データファイル (.bib) の作成

```
% 書籍  
@book{  
    文献参照名,
```



```

author = "著者名",
title = "タイトル",
isbn={},
series={},
year = "出版年"
publisher = "出版社",
keywords={}
}

```

```

% 小冊子
@booklet{
文献参照名,
author = "著者名",
title = "タイトル",
publisher = "出版社",
note = "",
month = "",
year = "出版年"
type= ""
}

```

## 編集処理作業の流れ

1. .tex ファイルを編集保存し、「Alt + F5」でビルドします。
2. .pdf, .aux, .out, .log (biblatex: .bcf, .bbl, .blg) が生成されます。
3. 「Shift + Alt + P」で PDF ファイルを開きます（初回のみ）。
4. .tex ファイルを再編集保存し、再ビルドします。
5. PDF ファイルに更新が反映されます。

※「Alt + F5」と「Shift + Alt + P」は gedit での設定。