

X_YTeX-specimen

Clerk Ma

文档作者: AlphaZTX

2022-10-15

简介

X_YTeX 在 Windows 系统下默认使用 fontconfig 库来获取可用的 OpenType 字体。fontconfig 是 Linux 系统下的字体管理库，移植到 Windows 中会出现速度变慢的现象。X_YTeX-specimen 专为 Windows 下的字体获取做出了优化，在字体获取速度方面较原始的 X_YTeX 有较大的提升。

X_YTeX-specimen 不依赖于 fontconfig，但需要 Python 3。

目录 / CONTENTS

1 安装 X _Y TeX-specimen	1
2 使用	3

1 安装 X_YTeX-specimen

首先，从 [GitHub Releases](#) 中下载编译好的压缩包，例如 build-1015.zip。在下载压缩包中，你会看到类似于下面的目录结构：

```
build-DATE
├─ dll-x86
├─ exe-a64
├─ exe-x64
└─ exe-x86
```

其中, `dll-x86` 路径下的文件是动态链接库 `xetex.dll`, 这里的 `xetex.dll` 不依赖于 `fontconfig` 库。

`exe-a64` 路径中包含了 ARM 64 架构下的 XeTeX 程序本体; `exe-x64` 路径中包含了 x64 (AMD 64) 架构下的 XeTeX 程序本体; `exe-x86` 路径中包含了 x86 架构下的 XeTeX 程序本体。

替换 $\text{T}_{\text{E}}\text{X}$ Live 中的动态库 将 `dll-x86` 下的 `xetex.dll` 复制到 $\text{T}_{\text{E}}\text{X}$ Live 中的 `bin` 目录下 (例如 `C:\texlive\2022\bin`, 具体的路径取决于安装位置), 即完成动态库的替换。

如果你不准备长期使用 XeTeX -specimen, 也可以保留 `bin` 路径下原有的 `xetex.dll` 以便随时替换回来。

替换 XeTeX 本体 对于不同架构的 Windows 系统, 也可以选择不同系统架构下专用的 XeTeX 程序本体。将与当前系统架构对应的 XeTeX 本体 (`xetex.exe`) 从下载的压缩包复制到 `bin` 路径下, 即完成 XeTeX 本体的替换。注意, 这一步是可选的。你也可以选择使用 $\text{T}_{\text{E}}\text{X}$ Live 中原有的 `xetex.exe`。

刷新字体数据库 在 [GitHub: libspecimen](#) 中, 你可以找到一份 Python 脚本 `gen-fontdb.py`。将这段 Python 脚本下载或保存到任意路径下, 首先检查 `gen-fontdb.py` 最后几行中 $\text{T}_{\text{E}}\text{X}$ Live 的路径是否与你选择的 $\text{T}_{\text{E}}\text{X}$ Live 安装路径是否一致, 若不一致, 则需修改。例如, 假定你的 $\text{T}_{\text{E}}\text{X}$ Live 安装在 D 盘中, 则需将 `gen-fontdb.py` 的最后几行由以下代码:

```
229 if __name__ == "__main__":
230     user_path_list = [
231         r"C:\texlive\2022\texmf-dist\fonts\opentype",
232         r"C:\texlive\2022\texmf-dist\fonts\truetype",
233         r"C:\windows\fonts"
234     ]
235     parse(user_path_list)
```

改为:

```
229 if __name__ == "__main__":
230     user_path_list = [
231         r"D:\texlive\2022\texmf-dist\fonts\opentype",
232         r"D:\texlive\2022\texmf-dist\fonts\truetype",
233         r"C:\windows\fonts"
```

```

234     ]
235     parse(user_path_list)

```

如果你还想使用更多路径，可以在上面的 `user_path_list` 中添加需要的路径。

最后，使用 Python 执行 `gen-fontdb.py`。例如：

```
python gen-fontdb.py
```

等候一段时间。生成 JSON 文件后，即完成全部安装过程。

2 使用

$\text{X}_{\text{T}}\text{E}_{\text{X}}$ -specimen 和 $\text{T}_{\text{E}}\text{X}$ Live 中原有的 $\text{X}_{\text{T}}\text{E}_{\text{X}}$ 的使用方法完全相同。编译命令仍然是 `xetex` 和 `xelatex`。 $\text{X}_{\text{T}}\text{E}_{\text{X}}$ 下的 `\font` 原语可正常使用， $\text{X}_{\text{T}}\text{E}_{\text{X}}$ 下的 `fontspec`、`unicode-math`、`xeCJK` 等宏包也可正常使用。举一组例子：

```

\font\0="Inter"
\0 The quick brown fox jumps over the lazy dog.          \par
\font\1="Libertinus Serif/I" at 12pt
\1 The quick brown fox jumps over the lazy dog.          \par
\font\2="[TeXGyrePagella-Regular.otf]:+smcp" scaled 1600
\2 The quick brown fox jumps over the lazy dog.          \par
\font\3="Latin Modern Roman/S=22:color=0000FF,mapping=tex-text"
\3 A blue quoted em-dash: ``---''.

```

The quick brown fox jumps over the lazy dog.
The quick brown fox jumps over the lazy dog.
THE QUICK BROWN FOX JUMPS OVER THE LAZY DOG.
 A blue quoted em-dash: “—”.

```

\font\4="[STIXTwoMath-Regular.otf]"\relax \4
\the\XeTeXcountglyphs\4          \quad
\XeTeXglyph 1699                  \quad
\XeTeXglyphname\4 1699            \quad
\the\XeTeXglyphindex"uni222B.dsp" \relax

```

6760 \int uni222B.dsp 1699

以上代码中的部分命令为 $\text{X}_{\text{E}}\text{T}_{\text{E}}\text{X}$ 的原语，详见 [\$\text{X}_{\text{E}}\text{T}_{\text{E}}\text{X}\$ 的文档](#)。

对于 $\text{X}_{\text{E}}\text{T}_{\text{E}}\text{X}$ ，我们也举一组例子：

```
% \usepackage{fontspec}
\fontspec{Latin Modern Sans}[Color=008800]
The quick brown fox jumps over the lazy dog.
```

The quick brown fox jumps over the lazy dog.

```
% \usepackage{unicode-math}
\[ \int_{-\infty}^{\infty} \delta(x) \, dx = 1. \]
```

$$\int_{-\infty}^{\infty} \delta(x) dx = 1.$$

```
% \usepackage{xecjk}
他说：“我能吞下玻璃而不伤身体。”
```

他说：“我能吞下玻璃而不伤身体。”