

1 LuaTeX-CN 开发者指南

本文档旨在为参与 `luatex-cn` 开发的成员提供技术指南。

1.1 本地测试与开发

在开发过程中，为了能够立即看到源码修改的效果，而不必反复重新安装宏包，建议使用以下两种方法。

1.1.1 方法 1: 软链接 (Junction) —— 推荐

该脚本可以自动在你的 `TEXMFHOME` 目录下创建一个指向本项目 `src` 目录的软链接（在 Windows 上为 Junction）。

- 开启链接: `texlua scripts/link_texmf.lua --on`
- 关闭链接: `texlua scripts/link_texmf.lua --off`

开启后，所有对 `src` 目录的修改都会在下一次 `lualatex` 编译时立即生效。

1.1.2 方法 2: 直接运行 (无需配置) —— 推荐 (New!)

得益于最新的模块加载重构，你现在可以直接在 `src` 目录下运行 `lualatex` 而无需设置任何环境变量：

```
cd src
lualatex test.tex
```

或者在项目根目录下直接指定路径编译：

```
lualatex src/test.tex
```

LuaTeX-CN 内部的模块系统会自动处理子目录下的 Lua 文件加载。

1.1.3 方法 3: 环境变量 (无链接测试) —— 高阶

如果你需要在非标准目录下测试，或者需要更程度的路径自定义，仍然可以使用环境变量。

重要提示 (Windows 用户): 即使在 Git Bash 中，TeX Live 通常也期望使用分号 `;`；而非冒号 `:` 作为路径分隔符。

Windows (Git Bash / CMD / PowerShell):

```
# Git Bash
export TEXINPUTS=".;./";
export LUAINPUTS=".;./";
lualatex test.tex
```

Linux / macOS:

```
export TEXINPUTS=".:./:"
export LUAINPUTS=".:./:"
lualatex test.tex
```

[!TIP] `//` 符号在 TeX 路径中表示递归搜索该目录下的所有子文件夹。

1.2 脚本说明

- `scripts/link_texmf.lua`: 跨平台脚本，自动检测 `TEXMFHOME` 路径并管理链接。
- `scripts/tag_version.lua`: 用于版本标记的工具。
- `scripts/build_ctan_windows.py`: 用于构建 CTAN 发布包的脚本。

1.3 目录结构

- `src/`: 核心源代码 (Lua 与 TeX)。
- `scripts/`: 开发辅助脚本。
- `test/`: 单元测试。
- `文档/`: 用户及开发者文档。
- `示例/`: 各类排版示例。