

UpTeX 電子書製作指南

子 康 譔

ver.1.0

庚子仲春二月廿三

成 都

目次

第一章 發凡	3
1.1 UpTeX 之優缺點	3
1.1.1 優點	3
1.1.2 缺點	3
1.2 製作電子書的幾大原則	3
1.2.1 確定對象讀者群	3
1.2.2 內容和樣式分離	4
1.2.3 前後代碼要統一	4
1.2.4 樣式的命名要能望文生義	4
1.2.5 未定稿之前請專注于文本	4
1.2.6 盡量減少髒代碼	4
第二章 L ^A T _E X 簡單入門	5
2.1 如何提問	5
2.2 善用搜索	5
2.3 查找資料	5
2.4 認識 L ^A T _E X 編輯器	6
2.5 認識 L ^A T _E X 文件棧	6
2.6 如何編譯 PDF	7
2.7 常見錯誤	7
2.8 如何把 L ^A T _E X 代碼書寫得漂亮	7
2.8.1 縮進、對齊、換行	7
2.8.2 添加適當的文本注釋	8
2.8.3 空格的消歧義	8
2.9 入手魔改 class	8
第三章 虛擬字体	9
第四章 排版流程	10
4.1 創建模板	10
4.1.1 製版	10
4.1.2 製圖	11
4.1.3 製表	11
4.2 文本的前期準備	11

4.3	文檔結構	11
4.4	排版源文件	11
第五章	GitHub 協作	12

第一章 發凡

本手冊使用繁體中文書寫，並使用日文字體排版。

本章旨在說明利用 UpTeX 製作 PDF 電子書籍的一些重要原則和注意事項。

UpTeX 是 TeX Live 組件之一。安裝 TeX Live 2019 或是 MacTeX 2019，從安裝界面勾選 Japanese 選項，就可以把 UpTeX 組件安裝到你的電腦。Mac 用戶還可使用小川 弘和老師的 UpTeX 2019。注意：不可使用其他工具包安裝 UpTeX。

1.1 UpTeX 之優缺點

1.1.1 優點

完善的虛擬字體配置。可以利用虛擬字體實現字符擠壓、單體字符升降、單行注釋的跨行對齊、行間標點、自動字體回滾 (font fallback)。因為虛擬字體不用加載字體緩存，UpTeX 編譯速度在眾多 LaTeX 中，也是最快的。完善的直書排版配置。原生直書（不是旋轉頁面得到的）。直書支持的類文檔非常豐富。完整的一致性，跨平臺的統一。

1.1.2 缺點

代碼不敷靈活。不支持宏配置字體。不支持漸變色文本；不支持漸變色繪圖。不支持透明圖層。髒代碼太多。使用 ruby 排版的行間注不能換行，使用 warichu 排版的割注也不能換行。當 PDF 中需要排版割注和行間注時，不可避免的會出現一些髒代碼。導致移植性變差。

1.2 製作電子書的幾大原則

本章簡單介紹製作 PDF 電子書的幾個不成文原則。

1.2.1 確定對象讀者群

製作一部電子書籍，最終將呈現在讀者面前。預期讀者群決定你的文本能達到的深度。比方說，給小孩子看的內容，字要大，圖要多，文要簡潔，注釋要清晰明了，使小讀者一看就懂。再如給科技工作者看的文本，大量的數學公式和英語單詞，使用橫排、緊密排版，字小圖多，信息密度大。如果你作為創作者或整理者，同時需要對內容和版面負責；如果你只是做排版，就只是關注版面好了。排版者需要針對不同的讀者群體，制訂不同的體例。

比方我曾經製作一些電子書僅供自用，後來分享給身邊朋友們，反響也都不錯，因此滋生製作一部分電子書放網上大家看的想法。我的主觀審美趨向，會深深地烙印在我製作的 PDF 中，那都是我的風格啊。因此，製作電子書，應該盡量避免攜帶私貨（比方說異體字的使用趨勢）。

1.2.2 內容和樣式分離

譚到這一節，有一個不可避免的問題，即文本文檔（`maintext.tex`）和主文檔（`main.tex`）的分離。見第 2.5 節。

內容：指文本的初始形態，作為最元始的備份，用於不同 LaTeX 格式之間的移植。理論上，先得到精校文本，再得到精校注釋，然後把二者合起來，得到清樣.txt。

樣式：是指文本的最終呈現形式，亦稱源代碼.tex。見第 4 章。

1.2.3 前後代碼要統一

這個原則是說，在寫作 LaTeX 腳本時，應該做到前後一致的風格。比方說，段落之間空出相同的兩個換行符，而非前面空三個，後面空二個。再如，前面使用 `\gezhu{}` 表示割注，後面使用 `\warichu{}` 表示割注。這些都是不好的例子。

再如，epub 使用的 html 排版流，格式語句 `` 的閉合等等。做到風格的統一，提高自己檢查代碼的效率，同時也方便他人的移植（儘管我一直強調 UpTeX 的移植性比較差）。

1.2.4 樣式的命名要能望文生義

接前一節示例：前面使用 `\gezhu{}` 表示割注，後面使用 `\warichu{}` 表示割注。同樣是割注，一個使用拼音命名，一個使用日文假名命名，格式不統一，造成閱讀麻煩；不能望文生義更是加重了這種困擾。樣式的命名應該簡單明了，使人一眼就能猜解。

再如，下劃綫的樣式命名為 `\a{}`，若是命名為 `\underline{}` 是否更清晰明了呢？

1.2.5 未定稿之前請專注于文本

不要一邊搞校訂，一邊搞排版。這樣你的格式容易寫死，移植性就非常差。原始版本具備相當的重要性。未定稿之前，請專注于文本的校訂和攷釋。一旦清樣.txt 定稿，以之為原始備份，來製作不同的排版項目。若是中途發現錯別字，則需要同時修改清樣.txt 和源代碼.tex，兩處內容應當保持相當的一致。

1.2.6 盡量減少髒代碼

移植性：是指一個項目從 TeX，到 Html、Epub 等的再現。這種再現是折衷的，是不完全的，是有偏離的。

一致性：在空間上，指一個項目在不同設備之間的重現。在時間尺度上，指今年編譯的結果，要和去年的保持一致。

髒代碼，顧名思義，就是骯髒的代碼。它不利於閱讀，不利於移植性，更不利於跨機器編譯時保持穩定的一致性。那些通過宏實現的行間標點、通過宏對齊的單行注釋文本、魔改的縱向破折號、魔改的挪格樣式，都是髒代碼。

寫髒代碼是人的本能。我也寫過不少髒代碼。我也一直在寫髒代碼。大唐西域記的 PDF 就是利用髒代碼編譯的。髒代碼不可避免，但是我們可以盡量減少使用髒代碼。

第二章 L^AT_EX 簡單入門

本章旨在提示 LaTeX 入門用戶，建立一些基本的概念。

2.1 如何提問

如何提問，即如何快速、準確地描述你的問題。紅口白牙，口說無憑，要拿出憑據，來說明你的問題。答主並不想從頭到尾再敲出來你的代碼。

針對社區的提問，一定要附帶 MWE (Minimal working environment)，也就是能復現問題的最小運行案例。將原始項目刪改後打包，即是最小運行案例。

任何人都有權拒絕回答不帶 MWE 的提問。

2.2 善用搜索



好消息！好消息！

本群已經跟著名搜索引擎 Google 達成战略合作伙伴关系。今后大家有什么不懂的，可以直接上 Google 搜索，不用在群里问了。



好消息！好消息！

本群已經跟著名莆田引擎 百度達成战略合作伙伴关系。今后大家有什么不懂的，可以直接百度搜索，不用在群里问了。

把錯誤碼、關鍵詞等的信息，粘帖在搜索框中，點擊搜索。不斷地嘗試，就是學習、進步的過程。

2.3 查找資料

LaTeX 中文社區擁有相當豐富的入門文檔。比如安裝文檔教程就有：

嘯行的 TeX Live 2019 安裝教程；

入門教程就有 TeX Live 2019 自帶的 `lshort-zh-cn.pdf`；

雷叔的入門教程：`lnotes2 包太累 2019 版.pdf`。

2.4 認識 L^AT_EX 編輯器

推薦使用 MS Visual Studio Code，安裝 LaTeX workshop 插件，可以提高 LaTeX 輸入效率。關於配置編輯器正向、逆向搜索功能，可見本手冊附件 settings.json。

對於初學者，建議手寫全部代碼，不要自動補全；不要直接複製粘貼別人的代碼。有些代碼是針對 XeLaTeX 的，複製到 UpTeX 下會直接報錯。搞不清楚代碼的原理，這種複製粘貼就會適得其反，成為學習進步的阻礙。

2.5 認識 L^AT_EX 文件棧

（表格）

項目	描述
main.tex	主文檔。通用入口文檔。
maintext.tex	源代碼.tex。用於存儲具有 LaTeX 樣式的文本。
my.cls	類文檔，又叫 class 文檔。LaTeX 配置終極奧義。
mysettings.sty	外挂的 LaTeX 配置信息，一些不宜寫入 class 的內容。
myfonts.sty	外挂的字体配置。不宜寫入 class 之中。
*.aux	臨時文件。最常見編譯錯誤皆由未刪除臨時文件引起。
*.bak	備份文件。
*.bbl	參考文獻臨時文件。
*.blg	臨時文件。
*.ent	臨時文件。
*.fdb_latexmk	latexmk 臨時文件。
*.lof	圖目錄。
*.log	日志文件。
*.lol	臨時文件。
*.lot	表目錄。
*.out	臨時文件。
*.synctex.gz	同步臨時文件。
*.thm	臨時文件。
*.toc	目錄。
*.dvi	輸出 PDF 的過渡文件。不宜刪除。
編譯 main.bat	UpTeX 的編譯棧。

什麼是通用入口文檔？我稱之為入口文檔，當我創建一個陌生的項目，我會默認使用這個入口，以便隨時可以更換不同的內容。

my.cls 是通用 class，因此針對 class 的修改，可以放在外挂配置信息 mysettings.sty 之中，以便和通用類文檔做隔離。或者，你也可以修改 class 內部名稱，並將外部名稱亦修改，拷貝放到項目目錄中，在 main.tex 中從新調用新的 class 名稱。

編譯 main.bat 是 UpTeX 的編譯棧。UpTeX 不宜使用編輯器自動編譯，因為 JIS B5 的幅面在編譯鏈中不是太好調用。不如直接使用 Windows 批處理命令。

2.6 如何編譯 PDF

在編輯器上配置“一鍵編譯”非常複雜，且不容易成功。建議使用 Windows 自帶的批處理。編譯 main.bat 示例如下：

```
chcp 65001
latexmk -c
uplatex main
uplatex main
dvi2pdf -V 7 main.dvi
PAUSE
latexmk -c
exit
```

chcp 65001 是將頁面代碼轉換到 Unicode，以便終端中能顯示 TeX 錯誤信息中的漢字字符。

latexmk -c 清理上一次編譯產生的臨時文件。

uplatex 兩次編譯以產生目錄。三次、四次編譯等，以產生索引、參考文獻等內容。

dvi2pdf -V 7 main.dvi 輸出 PDF。-V 7 表示以 PDF 1.7 格式輸出。

PAUSE 產生一個中斷，顯示錯誤信息。

exit 退出命令提示符。

2.7 常見錯誤

最常見的錯誤包括：花括弧不閉合，著色命令中有兩個及以上換行符，有美元符 (\$)、井號符 (#)、百分號 (%) 未作消歧義處理（因此直接屏蔽後面的內容，使花括弧閉合不上）；命令正斜綫 (\) 和轉義反斜綫 (/) 衝突等等。

二分法查錯，通常用於錯誤不明的情況（即 TeX 將甲錯誤顯示成乙錯誤，輸出到終端，造成了誤解）。這種情況十分常見，但又別於前面那些常見錯誤。

正向查錯就是將 \endinput 沿正向不斷向後變換位置，以在長文檔中鎖定錯誤位置。

反向查錯就是將 \endinput 沿反向不斷向前變換位置，以在長文檔中鎖定錯誤位置。

2.8 如何把 L^AT_EX 代碼書寫得漂亮

2.8.1 縮進、對齊、換行

Python 之中，依靠縮進嚴格定義執行邏輯。

LaTeX 中，雖沒有規定必須縮進多少格，但是我們在書寫的時候，應當盡量遵循一定的縮進，使得代碼清晰亮行。

縮進的代碼一定要閉合。中間縮的多，兩頭不用縮，前後一致，上下對齊，使人一看就知道那是一個完整的代碼塊。

減少無謂的換行，以減少潛在的錯誤（在 LaTeX 中，兩個換行符意味著分段）。增加必要的換行，用來區分代碼塊。

2.8.2 添加適當的文本注釋

有的代碼一時之間並不清楚他的作用。可以試探性的修改某些參數，來察看修改後的微妙變化。通常情況下，需要備份原文，複製新的語句用於修改。

通常漂亮的注釋，是用來提示自己，某年月日，我修改了哪裡哪裡，期待得到什麼樣的效果，結果如何，將來的改進應該從哪個方向著手等等。方便自己察看，也方便別人閱讀。

2.8.3 空格的消歧義

LaTeX 是空格敏感型、是大小寫敏感型，它的特性甚至繼承自 Unix。

漢字與漢字之間的空格會元封不動的輸出到 PDF 中。因此需要消除漢字之間的空格。但要避免改變命令和漢字之間的空格，這是一個十分棘手的事。在編寫源代碼的時候就要刻意去避免空格的產生。

2.9 入手魔改 class

在掌握了一定程度的 LaTeX 基礎之後，可以試圖修改 class，以滿足局部的需求。改之前一定要備份 class。（修改任何一個 class，都需要單獨新建項目，並且為之重命名，以避免系統中存在相同的兩個 class 造成調用歧義。）改之前一定要備份被修改的語句。

比如將橫排的書眉改成直書的書眉，利用 TikZ 來製作水印，通過 pagestyle 調用水印命令，去輸出直書書眉。創建多個不同的 pagestyle，當然也就需要創建配套的水印命令，以滿足書眉的多樣化配置。

為水印添加可見的背景圖片，嵌入文本的下方，作為文本背景使用。當然啦，一頁文本一個圖片的話，就需要每一頁單獨訂製圖片，可以利用輪詢代碼去調用水印命令。扯遠了。

修改目錄樣式、修改章標題樣式、修改詩歌樣式等。

比如為詩歌配置的文本，採用某種字體搭配某個配色，可以製作成樣式模塊，一次調用，不用每次都在源代碼去設置它。

第三章 虛擬字體

虛擬字體是 UpTeX 的靈魂。用好虛擬字體，可以實現很多奇妙的運用。可以利用虛擬字體實現字符擠壓、單體字符升降、單行注釋的跨行對齊、行間標點、自動字體回滾 (font fallback)。

虛擬字體創建和使用，詳見拙著 [UpTeX 虛擬字體與直書排版要素 ver.1.1 \(03.15\) .pdf](#)。(點擊跳轉到 GitHub)

第四章 排版流程

本章內容主要介紹製作電子書的流程。

4.1 創建模板

創建模板，就像搭建積木，應該由淺入深的做。

像是 `jlreq.cls` 這種 5888 行的巨無霸，是不適合入門選手進行修改的，它太龐大，它太精緻，隨便改個地方都容易出錯。

入門選手應該選取結構明朗、邏輯簡單的模板，諸如 `utbook.cls` `ujbook.cls` 之流。

第一步，縷清 class 的結構，通過一遍一遍瀏覽代碼，做到爛熟於心。

第二步，標記疑點。自己不清楚功能的代碼塊千萬改不得。

第三步，製版。

第四步，改書眉。

第五步，測試輸出。

4.1.1 製版

明確概念：單開、對開；左開、右開；切口、訂口；內側（書脊）、外側（書口）。

單開：即單面印刷。每張紙只印刷一頁。

對開：每張紙正面印完印背面。展開的書頁，左右對稱，稱之為對開。

左開：書脊在右側，開口在左側。章標題總是出現在左側。

右開：書脊在左側，開口在右側，章標題總是出現在右側。

內側（書脊）：指打孔裝訂一側。拱起如同山脊。在包背裝和綫裝本中，指打孔裝訂一側。在蝴蝶裝之中，指書脊一側。

外側（書口）：書冊開口一側。直書最常用見左開式章標題。設計模板時，書眉必須位於外側，而非內側。書眉位於頁面的兩側時，應當左右對稱。

常見的幾種紙張有：ISO A3、A4、A5、A6；ISO B4、B5、B6；JIS B4、B5、B6。非標的紙張就任意配置了。比如配置長卷，直接改 class，並為之配置相應的文本長度。

修改版面時，可以不考慮書眉。版心尺寸是固定的，版心尺寸與書眉無關。這是最基本的原則。

版心寬度（行長）應當設置為正文字號的整數倍，版心高度應該設置為正文行距的整數倍，並且應比整數倍還長約 1mm 左右，使得一頁內容伸長時也能容納。

研究表明，直書排版時一行文本不宜超過 35 字；雙欄時，一行文本不宜超過 30 字。B5 直書排版時，一欄不宜超過 20 行（文本寬度有限，B5 容納以 145mm 為極限的版心寬度）。

對於正文，可以調整 `\evensidemargin` 參數和 `\oddsidemargin` 參數，使正文在兩個頁面上恰好對稱。不應調整 `\hoffset` 參數強行水平偏移，它會改變 TikZ 坐標參數，得不償失。

4.1.2 製圖

UpTeX 有個弊端，似乎對 PDF 之外的圖片格式不太友好。當出現無法插入圖片，甚至插入的圖片出現空白時，考慮將圖片轉制為 PDF 再插入。圖片的 PDF 版本，應當比最後編譯的 PDF 版本低（或是相同，但不要比它版本號更高，會引起錯誤）。

這是插圖出問題時，最好的解決辦法。

4.1.3 製表

利用 Excel 製作表格。再使用網絡工具將 excel 轉為 LaTeX 代碼。參見：[Excel 2 LaTeX](#)。

4.2 文本的前期準備

獲取文本的方法八仙過海，各顯神通，可以從開源網站下載，可以通過掃描圖書 OCR，可以手動錄入。等等。一個良好的文本，應當具備幾個特點：1、乾淨整潔，沒有髒文本（諸如亂碼的漢字、各種不常見的符號等）。2、沒有雜七雜八和文本無關的異體字。3、沒有類推簡化的簡化字。4、具備豐富的校釋、攷訂。5、完善的注釋。

格式上，應當是結構規整的，尤其是縮進、對齊、換行符等。

4.3 文檔結構

目前常見的做法，文檔結構應當劃分為 contents、figure 等不同的文件夾，單獨封裝。比如把圖片全部收納到 figure 中，再在源碼中調用相對路徑。比如把源代碼收納到 contents 中，按 chapter00 -100 進行文檔拆分，再在通用入口文檔 main.tex 中批量調用相對路徑。

另一個做法則是，不對源代碼進行拆分，直接在入口文檔中調用。

4.4 排版源文件

覓缺俟補。

第五章 GitHub 協作

暫缺俟補。