

# L<sup>A</sup>T<sub>E</sub>X 研習 I

## L<sup>A</sup>T<sub>E</sub>X 的基礎速成

蔡炎龍

政治大學應用數學系

2013 年 8 月於國立清華大學

- 1 簡介
- 2 開始動手!
- 3 標題與章節
- 4 傳說中好打的數學式子...
- 5 AMS- $\LaTeX$  套件
- 6  $\LaTeX$  定理環境的使用
- 7 插入圖片
- 8  $\TeX$  系統的安裝

# 簡介

# 簡單的歷史

- 史丹福大學 **Donald Knuth** (“The Art of Computer Programming” 的作者) 因為無法忍受當時 (1977) 的排版水準, 決定自創一套新的排版系統 - **T<sub>E</sub>X 系統**。

# 簡單的歷史

- 史丹福大學 **Donald Knuth** (“The Art of Computer Programming” 的作者) 因為無法忍受當時 (1977) 的排版水準, 決定自創一套新的排版系統 -  **$\TeX$  系統**。
- **Leslie Lamport** 基於  $\TeX$  系統, 發展了使用上更為便利的  **$\LaTeX$  系統** (1986)。

# 簡單的歷史

- 史丹福大學 **Donald Knuth** (“The Art of Computer Programming” 的作者) 因為無法忍受當時 (1977) 的排版水準, 決定自創一套新的排版系統 -  **$\TeX$  系統**。
- **Leslie Lamport** 基於  $\TeX$  系統, 發展了使用上更為便利的  **$\LaTeX$  系統** (1986)。

# 中文 $\LaTeX$

中文  $\TeX$ 、 $\LaTeX$  有不少解決方案, 介紹幾個目前流行的跨平台方案:

- 1  $\text{cw}\TeX$ : 吳聰敏、吳聰慧、翁鴻翎共同發展, 各平台都有, 但一般需配合特定版本  $\TeX$  系統, 使用上亦和原  $\LaTeX$  有些許不同。
- 2  $\text{CJK-}\LaTeX$ : 使用上最接近原有的英文  $\LaTeX$  系統, 但新增字型及安裝字型較為複雜。
- 3  $\text{Xe}\LaTeX$ : 可直接使用電腦的系統字, 非常方便, 除設定上及需使用 `xelatex` 編譯外, 其於相當接近原有  $\LaTeX$  使用方式。

# L<sup>A</sup>T<sub>E</sub>X 的好處

- 標準 (投稿期刊、任何需要呈現數學的地方)

# L<sup>A</sup>T<sub>E</sub>X 的好處

- 標準 (投稿期刊、任何需要呈現數學的地方)
- 跨平台 (任何平台皆通用)

# L<sup>A</sup>T<sub>E</sub>X 的好處

- 標準 (投稿期刊、任何需要呈現數學的地方)
- 跨平台 (任何平台皆通用)
- 純文字檔 (無限可能的應用)
- 易於專注在文章的内容, 而非排版

# 本次研討的內容

- 寫文章、論文

# 本次研討的內容

- 寫文章、論文
- 論文整理 ( $\text{Bib}\text{\TeX}$ )

# 本次研討的內容

- 寫文章、論文
- 論文整理 ( $\text{Bib}\text{\TeX}$ )
- 撰寫 *CV*、*Resumé*

# 本次研討的內容

- 寫文章、論文
- 論文整理 ( $\text{Bib}\text{\TeX}$ )
- 撰寫 *CV*、*Resumé*
- 出考卷

# 本次研討的內容

- 寫文章、論文
- 論文整理 ( $\text{Bib}\text{\TeX}$ )
- 撰寫 *CV*、*Resumé*
- 出考卷
- 做簡報

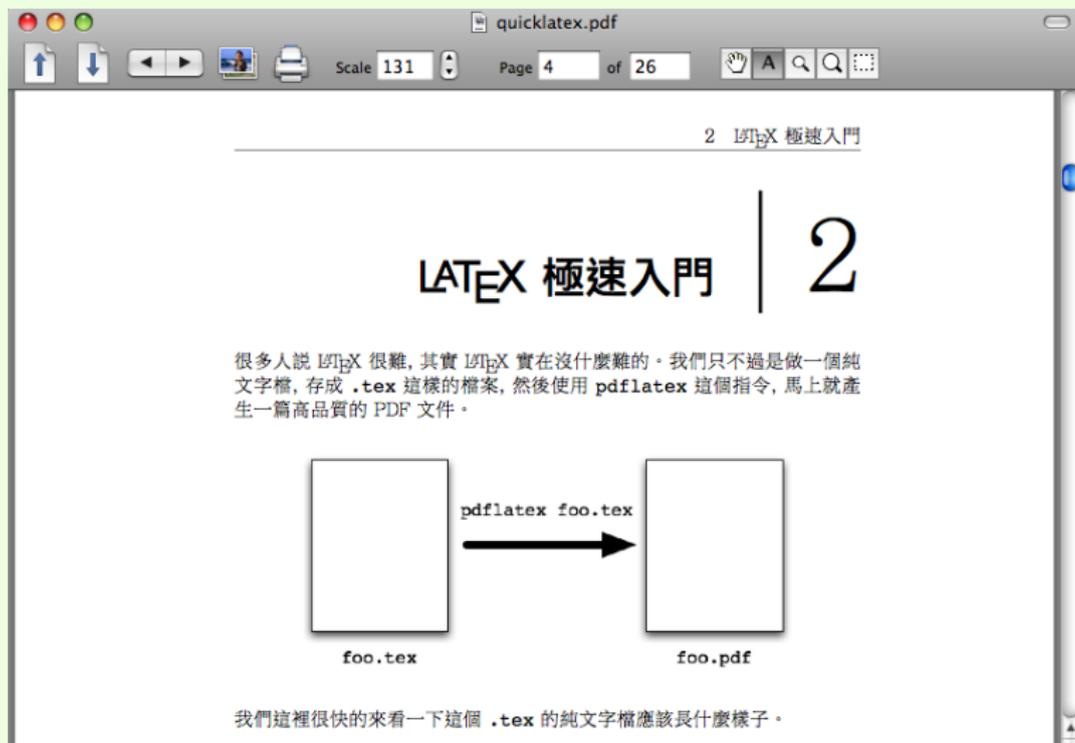
# L<sup>A</sup>T<sub>E</sub>X 原始檔是一個純文字檔



```
quicklatex.tex
Typeset LaTeX Macros Tags Templates
\section{\LaTeX} 極速入門
很多人說 \LaTeX 很難, 其實 \LaTeX 實在沒什麼難的。我們只不過是做一個純文字檔, 存成
\cmd{.tex} 這樣的檔案, 然後使用 \cmd{pdflatex} 這個指令, 馬上就產生一篇高品質的 PDF 文
件。
%
\begin{center}
\includegraphics[width=10cm]{pdflatex.png}
\end{center}

我們這裡很快的來看一下這個 \cmd{.tex} 的純文字檔應該長什麼樣子。
%
\subsection{最簡單的 \LaTeX 文件}
最簡單的 \LaTeX 檔案是長這個樣子。
%
\begin{Verbatim}[frame=single, framerule=3pt, commandchars=+\{\}]
\documentclass{article}
\begin{document}
+fbbox[內文, 文章的內容]
\end{document}
\end{Verbatim}
可以試打一些內容進去看看, 存成 \cmd{.tex} 檔, 再用 \cmd{pdflatex} 編譯。要注意目前還不能
用中文。
%
\subsection{完整的 \LaTeX 格式}
一份完整的 \LaTeX 文件的架構大概如下。
\begin{Verbatim}[frame=single, framerule=3pt, commandchars=+\{\}]
\documentclass{article}
+fbbox[設定區, 我們還不會]
```

# 編譯後才是我們要的結果



quicklatex.pdf

Scale 131 Page 4 of 26

2 L<sup>A</sup>T<sub>E</sub>X 極速入門

## L<sup>A</sup>T<sub>E</sub>X 極速入門 | 2

很多人說 L<sup>A</sup>T<sub>E</sub>X 很難, 其實 L<sup>A</sup>T<sub>E</sub>X 實在沒什麼難的。我們只不過是做一個純文字檔, 存成 `.tex` 這樣的檔案, 然後使用 `pdflatex` 這個指令, 馬上就產生一篇高品質的 PDF 文件。

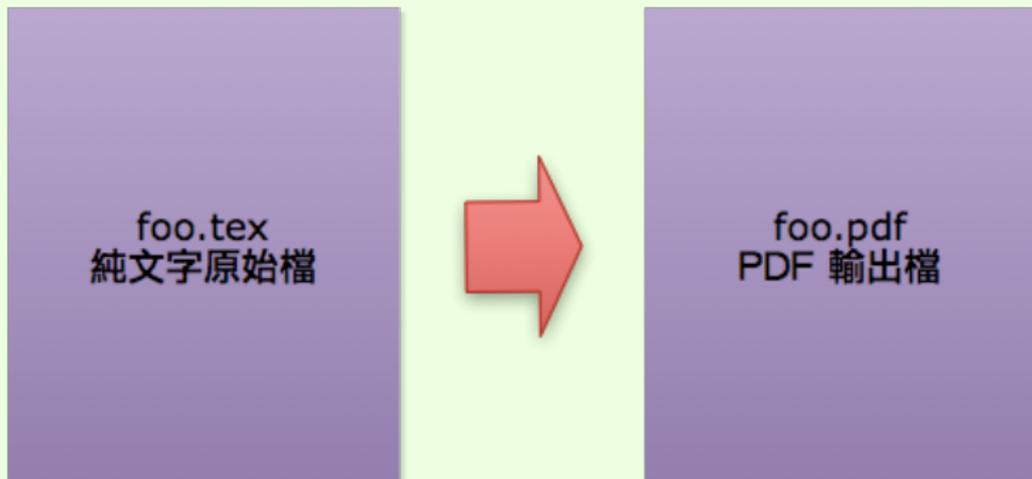


```
graph LR; A[foo.tex] -- "pdflatex foo.tex" --> B[foo.pdf]
```

我們這裡很快的來看一下這個 `.tex` 的純文字檔應該長什麼樣子。

# 迷思: $\LaTeX$ 很難學...

# L<sup>A</sup>T<sub>E</sub>X 編譯



```
pdflatex foo.tex
```

# 我們討論的 L<sup>A</sup>T<sub>E</sub>X 環境

- 英文直接使用 PDFL<sup>A</sup>T<sub>E</sub>X

# 我們討論的 $\text{\LaTeX}$ 環境

- 英文直接使用  $\text{\PDF\LaTeX}$
- 中文採用  $\text{\Xe\LaTeX}$

# 我們討論的 $\text{\LaTeX}$ 環境

- 英文直接使用  $\text{\PDF\LaTeX}$
- 中文採用  $\text{\Xe\LaTeX}$
- 中文字型可直接用電腦裡 TrueType 字型。

# 我們討論的 $\text{\LaTeX}$ 環境

- 英文直接使用  $\text{\PDF\LaTeX}$
- 中文採用  $\text{\Xe\LaTeX}$
- 中文字型可直接用電腦裡 TrueType 字型。
- 使用 UTF8 編碼

# 需要的工具

- $\TeX$  系統。

# 需要的工具

- $\TeX$  系統。
- 好用的文字 /  $\LaTeX$  專用編輯器 (如各平台都有的 TeXmaker, TeXWorks, Vim, emacs; Windows 的 Notepad++; Mac OS X 的 TeXShop 等等)。

# 需要的工具

- $\TeX$  系統。
- 好用的文字 /  $\LaTeX$  專用編輯器 (如各平台都有的 TeXmaker, TeXWorks, Vim, emacs; Windows 的 Notepad++; Mac OS X 的 TeXShop 等等)。
- 配合 Bib $\TeX$  的文件管理工具。

# 開始動手!

# L<sup>A</sup>T<sub>E</sub>X 的基本結構

```
\documentclass{article}
```

全文設定區 (preamble)

```
\begin{document}
```

文章內容

```
\end{document}
```

# L<sup>A</sup>T<sub>E</sub>X 的全文設定區 preamble



每個人使用 L<sup>A</sup>T<sub>E</sub>X 不久後,就自然會在**全文設定區 (preamble)** 做一些設定。可能是引入好用的 L<sup>A</sup>T<sub>E</sub>X 套件,也可能是做一些對自己方便的設定。

# 引入一個套件

在全文設定區, 我們最常做的動作之一就是引入一個套件。  
為了打中文, 我們現在引入我們  $\LaTeX$  的第一個套件 - **xeCJK**。

```
\usepackage{xeCJK}
```

# 準備用 Xe<sup>L</sup>A<sub>T</sub>E<sub>X</sub> 打中文

在引入 xeCJK 套件後, 我們準備用 `\setCJKmainfont` 來設我們的主要中文字型。

```
\setCJKmainfont{cwTeX Q Ming Medium}
```

% 也可改你系統有的字型

# 中文 Xe<sup>L</sup>A<sub>T</sub>E<sub>X</sub>「完整」版

```
\documentclass{article}
```

```
\usepackage{xeCJK}
```

```
\setCJKmainfont{cwTeX Q Ming Medium}
```

```
\begin{document}
```

文章內容

```
\end{document}
```



就是這樣, L<sup>A</sup>T<sub>E</sub>X 是不是超簡單?

# 標題與章節

# 標題、作者、日期

- 全文設定區中, 我們加入:

**`\title`**{文章的標題}

**`\author`**{作者}

**`\date`**{日期} % 不設會自動使用今天的日期

- 到了正文中, 一開始打入:

**`\maketitle`** % 下達這個指令才會把標題印出來

# 分節

要開始一節的時候, 就使用 `\section` 指令, 把標題用大括號括起來:

```
\section{本節的標題}
```

# 分節

**\section**{第一節的標題}

第一節的內容

**\subsection**{1.1 小節的標題}

小節的內容

:

:

**\section**{第二節的標題}

:

:

# 分節

如果你好奇, L<sup>A</sup>T<sub>E</sub>X 的文章節構一共有七層!

指令	層級	附註
<code>\part</code>	-1	
<code>\chapter</code>	0	在 books 或 report 格式中才有
<code>\section</code>	1	
<code>\subsection</code>	2	
<code>\subsubsection</code>	3	
<code>\paragraph</code>	4	
<code>\subparagraph</code>	5	

# 傳說中好打的數學式子...

# L<sup>A</sup>T<sub>E</sub>X 的數學符號

L<sup>A</sup>T<sub>E</sub>X 有兩種數學模式, 分別是:

- 隨文模式 (inline mode)
- 展示模式 (display mode)

# 隨文模式

所謂隨文模式就是數學式子要插在文中, 使用的方式是把數學式子放入兩個  $\$$  的符號中。比方說下面這個例子:

The formula  $f(x) = x^3 - 2x + 6$  is important in this case.

## 輸出結果

The formula  $f(x) = x^3 - 2x + 6$  is important in this case.

# 展示模式

所謂展示模式的數學式子,是把數學式獨立、置中表示。展示模式有很多下指令的方式,我們可以把數學式子用 “ **$...$** ”, “ **$\[ ... \]$** ”, 或 “ **$\begin{equation} ... \end{equation}$** ” 等方式表示。

# 展示模式

比方說

The formula

`\[`

`f(x)=x^3 - 2x +6`

`\]`

is important in this case.

輸出結果

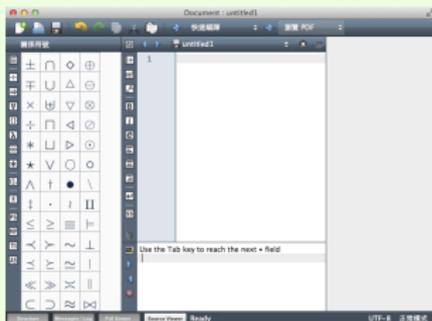
The formula

$$f(x) = x^3 - 2x + 6$$

is important in this case.

# 數學輔助工具

初學 L<sup>A</sup>T<sub>E</sub>X 可能會覺得數學符號很難打。因為 L<sup>A</sup>T<sub>E</sub>X 是一個標準, 有不少付費或免費的輔助工具, 如免費的 **TeXmaker** 編輯器有圖型式選單。



# 手寫式子找 L<sup>A</sup>T<sub>E</sub>X 碼

更酷的一個方案是 WebEquation: 我們手寫數學式子就可以出現 L<sup>A</sup>T<sub>E</sub>X 碼!



<http://webdemo.visionobjects.com>

# AMS-L<sup>A</sup>T<sub>E</sub>X 套件

# AMS- $\LaTeX$ 套件的引入

AMS 美國數學學會的  $\LaTeX$  套件已然成為一種標準。通常會用到的有三個套件：

- **amssymb**: 提供一些原本  $\LaTeX$  沒有的符號, 比方說  $\mathbb{R}$ ,  $\mathbb{C}$ , 等等。
- **amsmath**: 提供一些好用的環境, 比方說 `align` 環境等等。
- **amsthm**: 提供比較好的使用定理、定義等的環境。

# 來個範例

我們之後會常用到  $AMS-L^A T_E X$  套件, 很多書也是當成標準  
套件大量使用。我們先來個簡單範例, 就是打出實數的  $\mathbb{R}$ :

我們考慮一個在實數  $\mathbb{R}$  上的實數值函數  $f$ 。

## 輸出結果

我們考慮一個在實數  $\mathbb{R}$  上的實數值函數  $f$ 。

# 使用 AMS Article Class

使一個使用  $AMS-\LaTeX$  的方式是使用  $AMS$  提供的個文章類型, 叫  $AMS$  Article。要使用就是設定使用 **amsart** 文件格式:

```
\documentclass{amsart}
```

它會自動讀入 `amsmath`, `amsthm` 兩個套件, 和部份 `amssymb` 套件 (比方說有  $\mathbb{R}$ )。如果需要全套的 `amssymb`, 還是要自行讀入:

```
\usepackage{amssymb}
```

# 我們就預設給他全收

我們也可以一次就引入三個套件, 建議當成你的 L<sup>A</sup>T<sub>E</sub>X 預設的樣子。

## 小重點

建議就直接引入

```
\usepackage{amsmath, amsthm, amssymb}
```

# L<sup>A</sup>T<sub>E</sub>X 定理環境的使用

# 基本的定理環境

使用定理環境, 需要在設定區打入如下的指令:

```
\newtheorem{thm}{Theorem}
```

意思是我們要先建一個新的定理環境, 叫做 `thm`, 顯示時標示為 "Theorem"。比方說:

```
\begin{thm}
```

定理的內容

```
\end{thm}
```

# 定理的編號

前面的方式, 定理會自動以 **Theorem 1, Theorem 2, ...**, 等自動編號。但有時我們會希望第一節第一個定理是 "**Theorem 1.1**", 這該如何做呢? 很容易:

```
\newtheorem{thm}{Theorem}[section]
```

再來, 如果我們又定一個 "Lemma" 的環境:

```
\newtheorem{lemma}{Lemma}
```

Lemma 和 Theorem 的記數是**各自獨立**的, 有時我們喜歡這樣。但這樣又可能會發生「**Lemma 5** 後面是 **Theorem 2**」這種情況。

# 定理的編號

如果我們希望 Lemma 和 Theorem 用同一系統記數, 那應該要這樣設:

```
\newtheorem{thm}{Theorem}
```

```
\newtheorem{lem}[thm]{Lemma}
```

# 沒有編號的定理

有時我們會希望定理**不要有編號**, 比方說, 我們的主要定理:

```
\newtheorem* {mainthm} {Main Theorem}
```

# 定理的風格

在 `amsthm` 下, 有三種不同的定理風格: **plain**, **definition**, **remark**。請以下面的例子試驗、比較看看有什麼不一樣:

```
\theoremstyle{plain}  
\newtheorem{thm}{Theorem}
```

```
\theoremstyle{definition}  
\newtheorem{defn}{Definition}
```

```
\theoremstyle{remark}  
\newtheorem{rmk}{Remark}
```

# 定理的引用

$\LaTeX$  有一個重要的好處: 所有編號都是自動產生, 而如果有引用, 也會自動改變。要被引用的定理, 需要先設標籤 (`label`), 比如說:

```
\begin{thm}\label{T:major}
The statements of the theorem.
\end{thm}
```

文章中要引用時就是用 `\ref{T:major}`。比如說  
Applying Theorem `\ref{T:major}`, we obtain...

# 插入圖片

# graphicx 套件的使用

這裡建議使用 **graphicx** 套件, 現在幾乎大家都這麼做:

```
\usepackage{graphicx}
```

現在, 假設我們要插入 `pic.png` 這個圖檔, 使用

```
\includegraphics[width=5cm]{images/pic.png}
```

在我們的環境下, 圖形建議使用 **.png, .pdf, .jpg**。

# 浮動的圖形

```
\begin{figure}[h]  
\begin{center}  
\includegraphics[width= 圖形寬度]{檔案名稱}  
\end{center}  
\caption{圖形的文字說明}  
\end{figure}
```

figure 後面選項 “h” 指圖要放在此處。選項有以下幾種：

- **h**: 放在此處
- **t**: 放在頂端
- **b**: 放在底端
- **p**: 在本頁

# Float Control 相關套件

如果你發現  $\LaTeX$  圖的位置常常和你想的不一樣, 可以考慮幾個和 `float` 控制有關的套件。

---

**placeins** 控制某章節的圖就在某章節

---

**flafter** 還沒引用的圖不要先出現

---

# 文字繞圖

有時我們想要有文字繞圖的效果, 例如:

金杯理論已經存在至少半個世紀了, 但是近年來受到重視的原因之一, 大概就是我們有了很方便的測咖啡 TDS 的工具—VST 推出的 Coffee/Esspresso Refractometer。測 TDS 的工具很多, 還有人說測糖計也可以拿來用, 不過這些都不是那麼可靠, 主要是針對咖啡合適的測量範圍還有精確度的問題。這是很好的嘴炮主題, 我們也準備將來討論, 不過現階段先來看看這種測量計要怎麼使用。

首先要說明的是, 你在 VST 的商店會發現有新/舊兩款的 refractometer, 而每一款中又有:

**Coffee 版** 用來測咖啡的

**Coffee/Esspresso 版** 咖啡和 espresso 通用版

**Bundle 版** 附上 ExtractMoJo Pro 的版本

基本上, 我是覺得, 不一定非要 ExtractMoJo 這個軟體, 因為等你把這個系列看完, 這些你都會算啦。不過自然 ExtractMoJo 就是方便你去這些, 還有有名的金杯圖, 如果不是這麼在意價格其實不



Figure 1: VST 咖啡濃度測訂器

# 文字繞圖

這就引入套件 `\usepackage{wrapfig}`, 接著我們以 `wrapfigure` 取代原有的 `figure`:

```
\begin{wrapfigure}{r}{繞圖區寬度}  
\begin{center}  
\includegraphics[width= 圖的寬度]{圖檔名}  
\caption{圖說}  
\label{引用代碼}  
\end{center}  
\end{wrapfigure}
```

# 圖形的引用

圖形的引用其實和定理引用一樣。比方說：

```
\caption{圖說}\label{引用代碼}
```

引用方式和以前一樣：

參考圖  $\sim$  **\ref**{引用代碼}...

# 並列的系列圖

寫學術文章常會有並列的一系列的圖 (a), (b), (c) 之類的。



(a) 夜鷺成鳥



(b) 夜鷺亞成鳥

Figure 2: 夜鷺成鳥和亞成鳥比較

## 並列的系列圖

這是用 **subfigure** 套件做到的, 以下是範例。

```
\begin{figure}
\begin{center}
\subfigure[夜鷺成鳥]{\label{bird-a}
\includegraphics[scale=0.3]{bird01.jpg}}
\subfigure[夜鷺亞成鳥]{\label{bird-b}
\includegraphics[scale=0.3]{bird02.jpg}}
\end{center}
\caption{夜鷺成鳥和亞成鳥比較}
\label{bird}
\end{figure}
```

# 用 BibT<sub>E</sub>X 做論文管理

# BibT<sub>E</sub>X 是什麼?

BibT<sub>E</sub>X 的檔案是一個純文字檔, 以 `.bib` 為副檔名的檔案, 內容就是紀錄每一篇你有興趣文章的資訊, 可以看成你「可能引用」論文的資料庫。

# 一個例子

一個 Bib $\TeX$  紀錄看起來長這樣。

```
@article{tx07,  
  Author = {Tsai, Yen-lung and Xia, Eugene Z.},  
  Journal = {Proc. Amer. Math. Soc.},  
  Volume = {135},  
  Number = {8},  
  Pages = {2365-2367}  
  Title = {Non-abelian local invariant cycles},  
  Year = {2007}}
```

# BibT<sub>E</sub>X 的輔助工具

一般我們都不太記得「欄位」名稱或整個論文格式, 所以我們會用一些輔助工具來幫助我們整理論文。比如說以下的兩個程式:

跨平台 JabRef (Java 程式)

# BibT<sub>E</sub>X 的輔助工具

一般我們都不太記得「欄位」名稱或整個論文格式, 所以我們會用一些輔助工具來幫助我們整理論文。比如說以下的兩個程式:

跨平台 JabRef (Java 程式)

Mac OS BibDesk

# Cite-key 的建議

討論這個引用代碼的編法。比如說我們要引用

文章

Wilfried Schmid, "Variation of Hodge structure: the singularities of the period mapping," 1973

Lamport 式 `schmid:vhs`

建議 `schmid73`

# 引用格式和指定 Bib $\TeX$ 檔

在我們的  $\LaTeX$  原始檔中, 當然要指出用的是哪一個 Bib $\TeX$  檔,  $\LaTeX$  系統才知道你的論文資料庫在哪裡。使用方式是在 `\end{document}` 之前加入:

```
\bibliographystyle{plain}
```

```
% 使用 plain 格式, 可換其他格式
```

```
\bibliography{reference}
```

```
% 使用 reference.bib
```

# 文章的引用

文章中要引用某篇文章就是用 `\cite`, 比方說要引用前面 Schmid 的文章, 就是用

```
~\cite{schmid'73}
```

# BibT<sub>E</sub>X 的編譯

BibT<sub>E</sub>X 的編譯過程如下:

```
pdflatex foo.tex
```

```
bibtex foo.tex
```

```
pdflatex foo.tex
```

```
pdflatex foo.tex
```

# 文獻引用及排列方式的改變

改變文獻排列方式可以更改 `\bibliographystyle` 的設定。比如說, 我們想要依引用順序為排列順序, 可以選用 `is-unsrt` 風格。

```
\bibliographystyle{is-unsrt}
```

```
\bibliography{reference}
```

更多格式請參考

<http://www.cs.stir.ac.uk/~kjt/software/latex/showbst.html>

許多期刊會提供自己的文獻引用風格, 是以 `.bst` 為結尾的檔案, 放在和自己原始碼或依套件安裝方式安裝好, 就可以使用。

# $\TeX$ 系統的安裝

# Windows 下的安裝

- 安裝完整的 **MikTeX** (這也裝了 TeXWorks)。

# Windows 下的安裝

- 安裝完整的 **MikTeX** (這也裝了 TeXWorks)。
- 安裝 cwTeX-Q True Type 字型。

# Windows 下的安裝

- 安裝完整的 **MikTeX** (這也裝了 TeXWorks)。
- 安裝 cwTeX-Q True Type 字型。
- 下載並安裝 **JabRef**。

# Mac 下的安裝

- 使用 **MacTeX 安裝包**, 基本上這就裝好了完整的 T<sub>E</sub>X 系統, 包括 TeXShop、BibDesk 等程式。

# Mac 下的安裝

- 使用 **MacTeX 安裝包**, 基本上這就裝好了完整的  $\TeX$  系統, 包括 TeXShop、BibDesk 等程式。
- 安裝 cwTeX-Q True Type 字型。

# Linux 下的安裝

使用套件安裝系統, 安裝以下程式:

- $\TeX$  及  $XeTeX$  系統相關套件。

以上細節詳見 《中英文  $\LaTeX$  安裝與應用》。

# Linux 下的安裝

使用套件安裝系統, 安裝以下程式:

- $\TeX$  及  $\XeTeX$  系統相關套件。
- 安裝  $\TeX$ Works。

以上細節詳見 《中英文  $\LaTeX$  安裝與應用》。

# Linux 下的安裝

使用套件安裝系統, 安裝以下程式:

- $\TeX$  及  $XeTeX$  系統相關套件。
- 安裝  $TeXWorks$ 。
- 安裝  $cwTeX-Q$  True Type 字型。

以上細節詳見 《中英文  $\LaTeX$  安裝與應用》。

## 相關連結

---

炎龍老師的教學研究網

<http://yenlung.km.nccu.edu.tw/>

---

炎龍老師的 **blog**

<http://yenlungblog.wordpress.com/>

---

MiKTeX (Windows)

<http://miktex.org/>

---

MacTeX (Mac)

<http://tug.org/mactex/>

---