

데이터에서 출판까지

김강수

(KTUG)

August 27, 2024

책





《직지심체요절》

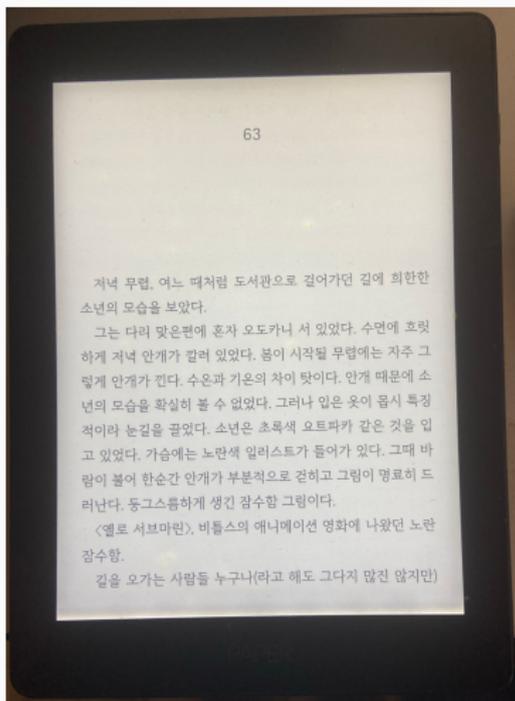
1372년^{공민왕 21}에 백운화상 경한이 임제종 18대 법손 석옥청공^{石屋清珙} 화상^{和尚}으로부터 받아 온 《불조 직지심체요절》을 증보하여 상·하 2권으로 엮은 것이다. 백운화상이 입적하고 3년 뒤인 1377년^{우왕 3}에 흥덕사^{興德寺}에서 금속활자로 찍어 낸 것이 초인본^{初印本}이다. 이는 현존하는 금속활자로 인쇄된 책 중 가장 오래된 것이다. 금속활자본은 현재 하권만이 전해지고, 프랑스 국립도서관에 소장되어 있다.

저녁 무렵, 어느 딱처럼 도서관으로 걸어가던 길에 외한한 소년의 모습을 보았다.

그는 다리 맞은편에 혼자 오도카니 서 있었다. 수면에 흐릿하게 저녁 안개가 깔려 있었다. 봄이 시작될 무렵에는 자주 그렇게 안개가 낀다. 수온과 기온의 차이 탓이다. 안개 때문에 소년의 모습을 확실히 볼 수 없었다. 그러나 입은 옷이 몹시 특징적이라 눈길을 끌었다. 소년은 초록색 오트파카 같은 것을 입고 있었다. 가슴에는 노란색 일러스트가 들어가 있다. 그때 바람이 불어 한순간 안개가 부분적으로 걷히고 그림이 명료히 드러난다. 동그스름하게 생긴 잠수함 그림이다.

〈엘로 서브마린〉, 비틀스의 애니메이션 영화에 나왔던 노란 잠수함.

길을 오가는 사람들들 누구니(라고 해도 그다지 많진 않지만)



전자책

전자책은, 휴대기기(휴대폰, PMP, PDA 등)나 컴퓨터로 볼 수 있는 특수한 포맷의 파일이라 할 수 있다. 일반적으로 전자책이라고 할 때는 텍스트 파일과 같은 범용 파일 포맷이 아니라 저작권 보호를 위해 DRM 기능을 탑재할 수 있는 특수한 포맷을 가진 파일을 말한다.

왜 전자책은 그다지 성공하지 못했는가?

- 검색가능성, 휴대의 편의성, 유연성, 신속성, 멀티미디어와 인터랙티브 등 많은 장점에도 불구하고 전자책은 종이책을 완전히 대체하지 못했다.

왜 전자책은 그다지 성공하지 못했는가?

- 검색가능성, 휴대의 편의성, 유연성, 신속성, 멀티미디어와 인터랙티브 등 많은 장점에도 불구하고 전자책은 종이책을 완전히 대체하지 못했다.
- **주의력의 영역**: 종이책은 늘 손닿는 곳에 있지만 전자책은 전자적으로 가려져 있다.
- **소멸성**: 데이터의 휘발성이 강하고 보존되지 않는다.
- **타이포그래피의 미발달**: 수천년의 역사를 가진 종이책의 타이포그래피를 흉내내는 것 자체가 한계. 그 결과 이독성^{易讀性}이 현저히 떨어진다.
- **레이아웃의 비고정성**: 이것은 장점이기도 하지만 동시에 단점이기도 하다. 특히 교과서와 같은 학습용 문헌에서 문제가 될 수 있다.

왜 전자책은 그다지 성공하지 못했는가?

- 검색가능성, 휴대의 편의성, 유연성, 신속성, 멀티미디어와 인터랙티브 등 많은 장점에도 불구하고 전자책은 종이책을 완전히 대체하지 못했다.
- **주의력의 영역**: 종이책은 늘 손닿는 곳에 있지만 전자책은 전자적으로 가려져 있다.
- **소멸성**: 데이터의 휘발성이 강하고 보존되지 않는다.
- **타이포그래피의 미발달**: 수천년의 역사를 가진 종이책의 타이포그래피를 흉내내는 것 자체가 한계. 그 결과 이독성^{易讀性}이 현저히 떨어진다.
- **레이아웃의 비고정성**: 이것은 장점이기도 하지만 동시에 단점이기도 하다. 특히 교과서와 같은 학습용 문헌에서 문제가 될 수 있다.
- 이후, 전자책은 더 발전할 것이다. 그러나 종이책이 사라지지도 않을 것.

“*A book is more than a verbal structure or series of verbal structures; it is the dialogue it establishes with its reader and the intonation it imposes upon his voice and the changing and durable images it leaves in his memory. A book is not an isolated being: it is a relationship, an axis of innumerable relationships.*”

— Jorge Luis Borges

출판 전략

최종출력물의 형식

- **PDF**: 가장 책에 가까운 전자문서이다. 사실상의 표준이며 레이아웃이 고정된 전자책이나 인쇄에 모두 활용할 수 있다. 고도의 타이포그래피를 적용할 수 있다.
- **EPUB**: 전자책(ebook)의 표준 포맷. EPUB 3.3은 **Fixed Layout**을 지원하고 다양한 멀티미디어를 수용하였다. 그러나 종이책의 인쇄에 활용하는 데 제한이 있고 타이포그래피의 적용에서 신통치못하다.

최종출력물의 형식

- **PDF**: 가장 책에 가까운 전자문서이다. 사실상의 표준이며 레이아웃이 고정된 전자책이나 인쇄에 모두 활용할 수 있다. 고도의 타이포그래피를 적용할 수 있다.
- **EPUB**: 전자책(ebook)의 표준 포맷. EPUB 3.3은 **Fixed Layout**을 지원하고 다양한 멀티미디어를 수용하였다. 그러나 종이책의 인쇄에 활용하는 데 제한이 있고 타이포그래피의 적용에서 신통치못하다.

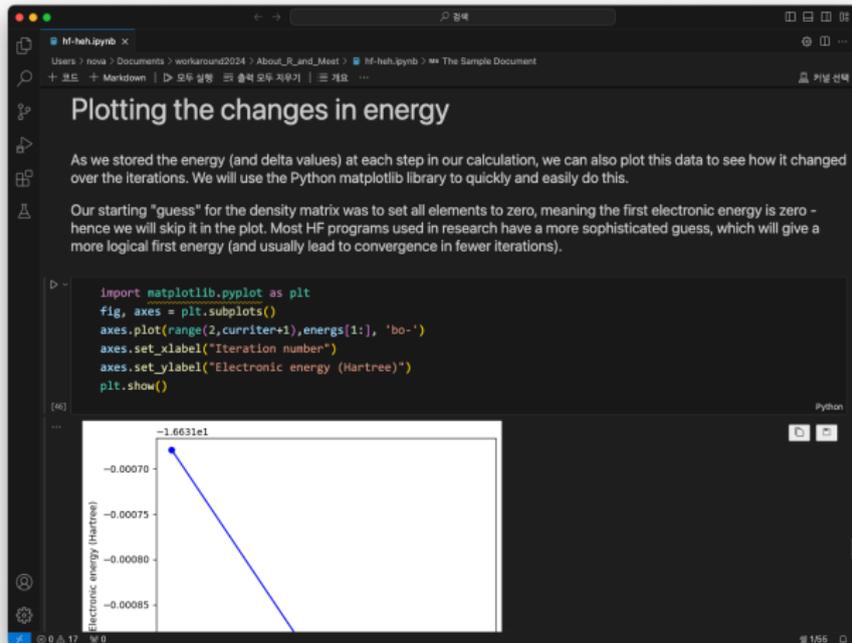
제안

PDF를 최종 출력물로 상정한다.

저자가 내용(contents)을 구성하기 위하여 사용하는 저작 도구가 갖추어야 할 요건.

- 텍스트 프로세싱: 작성중인 텍스트에 대하여 일괄 검색, 변환, 문단의 이동, 문단의 조작, 다른 파일과의 합성, 분리 등이 가능해야 한다.
- 데이터 처리 도구와의 연계: 작성중인 문서의 내부에서 외부 스크립트와 프로그램을 실행하고 그 결과를 포함할 수 있어야 한다. 즉 프로그래밍과 원고 작성이 동시에 이루어져야 한다.
- 미디어: 이미지, 수식 등 비텍스트적 요소를 쉽게 입력·참조할 수 있어야 한다.
- 편의: 도구 사용법을 익히는 데 많은 시간을 할애하지 않아도 되어야 한다.
- 출판가능: 최종적으로 .tex 포맷의 출판용 소스를 변환할 수 있어야 한다.

이를 통해 본다면 Plain text 이외에는 고려 대상이 될 수 없다. 레이아웃 에디터나 워드 프로세서로는 데이터를 처리·가공하여 책을 만드는 데 있어 만족할 만한 결과를 얻기 어렵다.

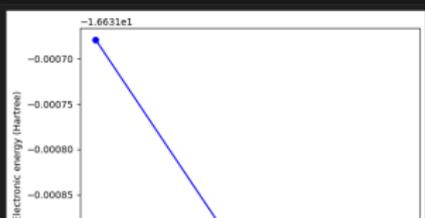


Plotting the changes in energy

As we stored the energy (and delta values) at each step in our calculation, we can also plot this data to see how it changed over the iterations. We will use the Python matplotlib library to quickly and easily do this.

Our starting "guess" for the density matrix was to set all elements to zero, meaning the first electronic energy is zero - hence we will skip it in the plot. Most HF programs used in research have a more sophisticated guess, which will give a more logical first energy (and usually lead to convergence in fewer iterations).

```
import matplotlib.pyplot as plt
fig, axes = plt.subplots()
axes.plot(range(2, curriter+1), energies[1:], 'bo-')
axes.set_xlabel("Iteration number")
axes.set_ylabel("Electronic energy (Hartree)")
plt.show()
```



Iteration number	Electronic energy (Hartree)
2	-1.6631e1

Markdown: Rmarkdown

The screenshot displays the RStudio interface with an R Markdown document open. The source code in the left pane includes:

```
1 ---
2 title: "Untitled"
3 output: pdf_document
4 date: "2024-08-26"
5 ---
6
7 [r setup, include=FALSE]
8 knitr::opts_chunk$set(
9   echo = TRUE)
10
11 ## R Markdown
12
13 This is an R Markdown document. Markdown is a simple formatting syntax for authoring
14 HTML, PDF, and MS Word documents. For more details on using R Markdown see
15 http://markdown.rstudio.com.
16
17 When you click the Knit button a document will be generated that includes both
18 content as well as the output of any embedded R code chunks within the document. You can
19 embed an R code chunk like this:
20
21 [r cars]
22 summary(cars)
23 ---
```

The rendered PDF on the right shows the following content:

Untitled
2024-08-26

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://markdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
## speed      dist
## Min.   7.0-0  Min.   2.00
## 1st Qu: 12.0-0 1st Qu.: 26.00
## Median: 15.0-0 Median: 30.00
## Mean   :15.4   Mean   : 42.00
## 3rd Qu.: 18.0-0 3rd Qu.: 50.00
## Max.   :20.0-0  Max.   :120.00
```

Including Plots

You can also embed plots, for example:

distance	pressure
26	562
30	611
50	688
50	726

- reStructuredText: Sphinx의 바탕이 되는 문서 작성용 Markup 언어.

reStructuredText
Markup Syntax and Parser Component of [Docutils](#)

- AsciiDoc: 주로 technical documents를 작성하는 Markup 언어.

<pre>1 = Hello, AsciiDoc! 2 3 This is an interactive editor. 4 Use it to try https://asciidoc.org/AsciiDoc. 5 6 == Section Title 7 8 • A list item 9 • Another list item 10 11 [,ruby] 12 ---- 13 puts 'Hello, World!' 14 ----</pre>	<p>Hello, AsciiDoc!</p> <p>This is an interactive editor. Use it to try AsciiDoc.</p> <p>Section Title</p> <ul style="list-style-type: none">• A list item• Another list item <pre>puts 'Hello, World!'</pre>
--	--

장점

- 간편하고, 배우기 쉽고, 프로그래밍 작업과 연계하기 용이하다.
- 전용 또는 범용 Editor, Reader 내지 Renderer가 많다.

장점

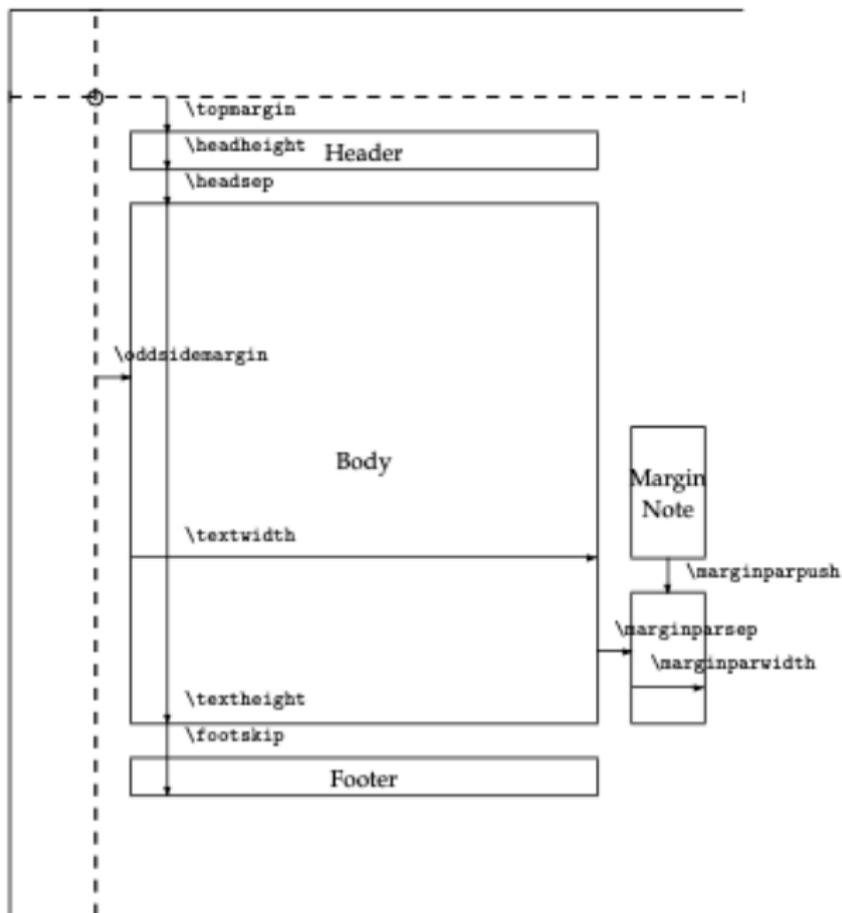
- 간편하고, 배우기 쉽고, 프로그래밍 작업과 연계하기 용이하다.
- 전용 또는 범용 Editor, Reader 내지 Renderer가 많다.

단점

- Markdown만으로 작성된 문서는 풍부한 조판 정보를 포함하지 못한다.
- 대부분의 Markdown Renderer들은 pdf로 직접 출력하는 기능이 있거나 적어도 pandoc을 이용하면 pdf로 변환할 수 있지만 (특히 한글이 사용된 경우에) 출력 품질이 기대에 미치지 못한다.
- 어떤 Markdown-류 언어로 문서를 작성했더라도 최종 출력은 .tex을 거쳐 PDF를 생성하는 것이 좋은 대안이다. 많은 경우에 converter가 있다.

왜 **LaTeX**인가

- 현대 출판의 조판 규칙을 모두 구현할 수 있다. 텍스트의 정렬, 폰트의 사용, 페이지의 구성, 오브젝트(수식, 그림)의 배치, 다단, 다국어 ……
- 프로그래밍가능한 language이므로, 출판을 위한 디자인 요소를 적용할 수 있다.
- 프로그래밍가능한 language이므로 출판을 위한 준비 단계에서 독자적인 소스 프로세싱이 가능하다.
- .tex으로 변환하거나 출력하는 프로그램이 많고, 필요하다면 직접 만들 수 있다.





글자

- 1 폰트: NFSS, 2
- 2 폰트: fontspec, 4
- 3 oblivoir의 폰트 설정 명령, 8
- 4 greenbook의 폰트 설정, 9

출판물의 디자인 (계속)

```
\makechapterstyle{demo}{%
  \renewcommand{\chapterheadstart}{%
    \vspace*{\beforechapskip}
    \begin{tikzpicture}[remember picture,overlay]
      \tznode<0,3mm>(current page.north){%
        \includegraphics[width=\dimexpr\paperwidth+6mm\relax,height
      ]{b}
      \end{tikzpicture}
    }
  \renewcommand{\prechapternum}{}
  \renewcommand{\postchapternum}{}
  \renewcommand{\chapternamenum}{}
  \renewcommand{\printchapternonum}{\par\nobreak\vspace{\dimexpr\midc
  \renewcommand{\printchapternum}{%
    \vspace*{-50pt}
    \begin{adjustwidth}{0pt}{-\dimexpr\foremargin-3em\relax}
    \raggedleft\color{kmcyan!30!white}\fontspec{TeX Gyre Schola}\bf
    \end{adjustwidth}
  }
}
```

동해물과 백두산이

동해물과 백두산이

동해물과 백두산이

동해물과 백두산이

동해물과 백두산이

```
less 'kpsewhich pmhangujamo-frkim.code.tex'
\if_empty:NF \g_jun_tl
{
  \prop_get:NVNTF \l_userjung_prop \g_jun_tl \l_tmptwo_tl
  { \tl_use:N \l_tmptwo_tl }
  {
    \prop_get:NVNT \c_frkJung_prop \g_jun_tl \l_tmptwo_tl
    { \tl_use:N \l_tmptwo_tl }
  }
}
\if_empty:NF \g_jon_tl
{
  \prop_get:NVNTF \l_userjong_prop \g_jon_tl \l_tmpthr_tl
  { \tl_use:N \l_tmpthr_tl }
  {
    \prop_get:NVNT \c_frkJong_prop \g_jon_tl \l_tmpthr_tl
    { \tl_use:N \l_tmpthr_tl }
  }
}

\tl_gclear:N \g_jun_tl
\tl_gclear:N \g_jon_tl

\if_empty:NF \l_delayedtoken_tl
{
```

```
\hg{sksms rm zkazkagks rm wldhrdp rktjeh sksms chtqnfdf zuemfrh  
rmeo thsurk ehfdkdhms, wlswd rmeo thsufmf rlekflrh dlTrpTdmqslek.}
```

나는 그 캄캄한 그 지옥에 가서도 나는 촛불을 켜두고 그대 소녀가 돌아오는, 진정 그대 소녀를 기다리고 있겠습니다.

소스 프로세싱 (3)

소스

```
1 %<*1>
2 It was in the middle of winter, when the broad
3 %</1>
4
5 %<*2>
6 But this queen died; and the king soon married
7 %</2>
8
9 %<*3>
10 \begin{verse}
11 "Tell me, glass, tell me true!\\
12 Of all the ladies in the land,\\
13 Who is fairest? tell me who?"
14 \end{verse}
15 %</3>
```

```
1 %<*1>
2 눈의 넓은 조각이 특정 여왕이 그녀의 창에서 작업 앉으,
3 %</1>
4
5 %<*2>
6 그러나 여왕은 죽었다 왕은 곧 또다른 매우 아름다웠다 아
7 %</2>
8
9 %<*3>
10 \begin{verse}
11 "말해, 유리, 진정한 말해!\\
12 이 땅의 모든 여성의,\\
13 예쁘지는 누구입니까? 누가 말해?"
14 \end{verse}
15 %</3>
```

결과

It was in the middle of winter, when the broad flakes of snow were falling around, that a certain queen sat working at her window, the frame of which was made of fine black ebony; and, as she was looking out upon the snow, she pricked her finger, and three drops of blood fell upon it. Then she gazed thought-

눈의 넓은 조각이 특정 여왕이 그녀의 창에서 작업 앉으, 주위에 떨어지는 때 그것은 겨울의 중간에 있었다 의 프레임은 미세 검은 흑단으로 만들어졌다: 그리고, 그녀가 눈에 찾고 되었을 때, 그녀는 그녀의 손가락을 찔려 피 세 방울을 그 위에 떨어졌다. 그런 다음 그녀는 하얀 눈을 뿌려빨간 방울에 조심스럽게

텍스트으로 준비하기: 텍스트

아마추어 저작자들이 흔히 저지르는 실수

- 예쁜 것과 올바른 것을 구별하지 못한다.
- 변하는 것을 예측하지 못한다.

사례 1:

```
\emph{emphasized text} \textit{italicized text}
```

emphasized text *italicized text*

편집자는 강조를 굵은 글자로 지시하였다.

emphasized text *italicized text*

편집자가 마음이 변해서 밑줄을 그으라고 한다.

emphasized text *italicized text*

비슷한 경우를 폰트의 선택에서도 볼 수 있다.

사례 2:

어떤 저자는 “사례”와 “증거”를 모두 박스 안에 넣었다.

```
\begin{myboxed}
```

...

```
\end{myboxed}
```

그런데 나중에 보니 이 두 종류의 문단은 다른 형식으로 조판해야 한다는 판단이 생겼다.

어떤 일이 일어났을까?

작업용 pdf라는 것은, 현재 작성중인 문서를 pdf 변환한 결과이다.

만약 이 pdf를 인쇄^{출판}하려 한다면, 필연적으로 텍스트의 폭, 페이지 잘림, 행 잘림, 행 사이의 간격, 정렬 방식, 그림의 배치 등이 모두 다 달라질 것이다.

따라서 현재 출력되는 상태를 보고 “약간의 간격을 더 넣어야 보기 좋겠다”든가 “여기는 강제로 행을 나누어야지”한 소스는 출판 단계에서 보면 처치곤란이다.

현재 출력되는 pdf의 모양은 잠정적인 것이다.

Excel로 준비하기: 그림과 표

- 저작권 문제를 살펴야 한다.
- 인쇄를 위해서는 그림을 잘 준비해야 한다.
 - 색도인쇄라면 그림이 혹시 인쇄에 적합하지 않은 색상공간으로 된 것은 아닌가?
 - 화면으로는 멀쩡한 그림도 인쇄하면 해상도가 낮아서 흉한 경우가 많다.
 - 흑백인쇄라면 그림이 grayscale로 잘 변환되었는지 확인한다.
- 이미지, 일러스트레이션에 관한 한 전문가에게 의뢰하는 것이 바람직하다.
- 그림이 여러 장 있을 경우, 소스 레벨에서 그림의 위치(폴더)를 잘 관리해야 한다. 특히 마크다운으로 원고를 작성한다면 더욱 그러하다.

출판에 적합한 표(table)

- 대체로 아마추어 저작자들, 특히 아래아한글 사용자들이 저지르는 잦은 실수가 표 관련된 것이다.
- 워드 프로세서가 만들어주는 표, 마크다운 변종 언어가 만들어주는 표를 그대로 사용하면 안 된다.
- 원고 레벨에서는 표를 하나의 그림처럼 별본으로 취급하든가, 아니면 표를 “의미상으로” 작성하는 도구를 이용하는 것이 바람직하다.
- 표가 일관된 형식으로 만들어져 있다면 간단한 변환을 통하여 쉽게 처리할 수 있지만, 일부 한국어 문헌에서 보는 것과 같은 “내용을 설명하거나 요약하는” 표는 따로따로 처리할 수밖에 없다.

- 소프트웨어 사용법을 설명하는 책은 소프트웨어가 화면에 보여주는 것을 그대로 그림으로 싣는다.
- 이때에 본문의 폰트와의 어긋남, 크기의 비일관성, 낮은 해상도 등으로 책의 품위를 낮추는 일이 빈번하다.
- 수학적 그래프와 같은 것은 쉽게 고품질의 결과를 얻을 수 있으나, 소프트웨어 출력 화면의 경우에는 조판상 고려할 점이 많다. 전문가에게 의뢰하는 것이 좋다.

엑스으로 준비하기: 부수적인 요소들

문헌정보는 **bib** 형식의 데이터베이스를 활용한다. 실제 조판에서는 **biblatex**과 **biber**를 이용한다.

소스 작성 단계에서 가장 괴로운 것 중 하나가 인덱스를 위한 마크업이다.

- 중요한 단어 마크업을 미리 설정해두고, 이 단어는 무조건 `index` 마크업에 포함되게 한다.
- 소스 레벨에서 `TeX`을 위한 령을 미리 입력해둔다.
- 최종 인쇄 전 단계에서 필요하다면 `.ind` 파일을 직접 수정한다.

이 요소들은 마크다운, pandoc, \LaTeX 이 잘 처리한다. 다만 교차참조는 마크다운 언어마다 사용법이 다른데, 한국어 문서라면 조사를 적절하게 처리하는 데 주의를 기울여야 할 것이다.

시나리오

- • •
 - 표준 작업용 출력 형식의 제작
 - ☞ 일반적인 문서에, 일반적인 정도의 환경을 정의하고, 일반적인(수용가능한) 레이아웃을 갖춘 표준 문서 양식을 정의
 - ☞ bitpublish는 좋은 출발점이 될 수 있을 것
 - 마크다운의 확장: 주로 사용자가 정의할 수 있는 `environment`로 변환 가능한 마크업 방법의 제공
 - ☞ 현재로서는 `escape`를 이용하는 방법이 있다. 예를 들면 ````\{=latex}`
 - 도표 변환 도구 기타 부수 유틸리티

\LaTeX 을 얼마나 잘^{많이} 알아야 할까?

1. 최종 출력은 .tex이다. \LaTeX 을 거쳐 출판한다고 생각하면 안 되고 \LaTeX 으로 출력하는 것을 출판이라고 생각하는 것이 옳다.
2. 신뢰할 수 있는 \LaTeX 작업용 스타일이 갖추어져 있어야 한다.
3. 훌륭한 연구 결과는 훌륭한 레이아웃과 타이포그래피를 갖춘 품위있는 출판물이 되어야 한다.