한글 LaTeX 가이드

(버전 0.95)

殷 光 熙(koaunghi.un@zdv.uni-tuebingen.de)

1996年 04月 04日

目 次

要 約

LaTeX은 文章을 整列하는데 卓越한 機能을 가진 文書編集機입니다. 한글LaTeX은 이런 優秀한 編集機로 한글도 쓸 수 있도록 하자는 趣旨에서 만들어진 프로그램입니다. 더구나 LaTeX은 使用에 아무런 制限이 없이 普給되고 있고, 이에 따라 한글LaTeX도 한글을 쓰고자 하는 모든 이에게 有用한 道具가 될 수 있기를 바라고 있읍니다.

이 說明書에는 다음과 같은 事項이 手錄되어 있읍니다.

- TeX에서 한글LaTeX까지
- 漢字의 登場
- 새로운 한글字型
- 한글LaTeX의 設置
- hangul.sty파일의 分析 및 使用法

그리고 附錄에는 한글LaTeX의 設置와 한글 및 漢字字型에 대해서, 그리고 코드입력기인 Mule과 漢字變換서버인 Wnn에 대해 간단히 說明하였읍니다.

이 글은 LaTeX을 使用하여 한글 文書를 쓰고자 하는 모든 이에게 조그마하나마 도움이 될 수 있고자 작성되었읍니다.

한글LaTeX을 발표하면서 한글글꼴작성에 도움을 주신 高麗大學校 言語學科 李基用教授님[1], 文化體育部 포스트스크립트글꼴을 普及하신 檀國大學校 國文科 洪允杓教授님, 한글LaTeX을 위해 도움을 주신 원세연님[2]과 이천우님[3] 그리고 한글LaTeX의 發展을 促進하신 다른 여러분께 感謝의 마음을 전합니다.

— 殷 光熙.

[1]KLEE@ling.korea.ac.kr
[2]sywon@sorak.kaist.ac.kr
[3]clee@achem2.ajou.ac.kr

第 1 節　한글LaTeX이 나오기까지

TeX은[4] 미국 Stanford大學의 數學科 敎授인 *Donald E. Knuth*에 의해서 1977년 5월에 처음으로 씌여진 것을 그해 여름에 Michael F. Plass씨와 Frank M. Liang씨가 지금의 TeX의 전신을 만들었고 같은 해 말에서 다음해 초 사이에 다시 Knuth교수가 완성을 시켰읍니다. 이는 SAIL언어로 씌여졌는데 1979년 초에는 이를 Knuth씨와 Luis Trapp Pardo씨에 의해 개발된 Web언어로 전환시키는 작업이 시작되어 1979년과 1980년에 Ignacio A. Zabala씨에 의해 완성되었읍니다. TeX82는 1979년 후반기와 1980년 전반기에 걸쳐 Knuth씨에 의해 더욱 정밀한 프로그램으로 정착되어서 그해 9월에 버전 0으로 나타나기 시작했읍니다. 다른 사람들의 조언등을 감안하여 1989년 9월에는 최종의 심사를 거쳐 발표된 TeX는 현재, 專門的인 自然科學文書를 作成하는데에는 가장 좋은 文書編集機로 認定되고 있읍니다([Knu86a] – [Knu86d]를 參考). Knuth敎授는 원래가 數學科 敎授인만큼 그는 이 文書作成機 TeX을 통해서 數學公式도 自由自在로 쓸 수 있도록 神經을 많이 썼읍니다. METAFONT는 良質의 字形을 만들어 내므로 TeX은 出版書籍을 만드는데 遜色이 없는 文書編集機로 자리를 잡게 되었읍니다.

좋은 藥은 입에 쓰듯이 TeX의 文書編集機能이 卓越하면 卓越한 그만큼 사용자의 側面에서는 많은 勞力과 經驗이 있어야 이런 機能을 理解할 수 있었고 따라서 自身이 願하는 樣式의 文書를 作成하기에 必要한 時間도 오래 걸리게 되었읍니다. 그래서 TeX의 사용은 專門적인 programmer層에서나 이루어질 수 있었읍니다. 즉, 使用者는 글꼴을 定義하고 文章과 文章사이의 間隔을 넓혀주는 등의 일을 일일이 손으로 處理해야 했읍니다.

그러나 美國의 컴퓨터과학자인 *Leslie Lamport*는 使用者와 TeX의 사이를 좁히는 役割을 할 수 있도록 LaTeX을 開發하여 쉽게 TeX을 통하여 文書를 作成하게끔 寄與하였읍니다([Lam85]를 參考). 그는 一般적인 文書 形式을 TeX의 매크로로 定義함으로서, 使用者는 專門적인 TeX의 知識이 없이도 짧은 時間내에 좋은 文書를 作成할 수 있도록 하였읍니다. 그리하여 使用者는 쓰고자 하는 글의 全體的인 輪廓에 맞는 매크로를 指定한 後 글을 쓰면, 願하는 形式의 文書가 作成될 수 있게 되었읍니다.

長期間이 흐르면서 LaTeX은 全世界的으로 많은 使用者를 얻게 되었고 그에 따라 LaTeX이 해내야 하는 課題의 폭도 넓어졌읍니다. 그 結果로 이제는 이 LaTeX의 根本적인 코드를 바꾸고 國際化되어야 한다는 輿論이 높아졌고

[4] 그리이스어 $\tau\epsilon\chi$에서 유래됨.

> English words like 'technology' stem from a Greek root beginning with the letters $\tau\epsilon\chi\ldots$; and this same Greek word means *art* as well as technology. Hence the name TeX, which is an uppercase form of $\tau\epsilon\chi$.
>
> Insiders pronounce the χ of TeX as Greek chi, not as an 'x', so that TeX rhymes with the word blecchhh. It's the 'ch' sound in Scottish words like *loch* or German words like *ach*; it's a Spanish 'j' and a Russian 'kh'. When you say it correctly to your computer, the terminal may become slightly moist.

The TeXbook에서 拔萃.

1992末부터는 LaTeX3 project team[5]이 構成되어 새로운 LaTeX의 作業에 着手하였읍니다. 1994年 6月 1日을 기해 LaTeX 2$_\varepsilon$를 公式的으로 내 놓은 이 팀은 현재(94年 11月) patchlevel 4를 發表하면서 계속 LaTeX3를 爲해 努力하고 있읍니다. Frank Mittelbach와 Rainer Schöpf는 [Sch94]에서 LaTeX3–Project의 目標를 다음과 같이 說明하고 있읍니다.

- LaTeX의 使用범위를 이전의 自然科學분야에서 벗어나, 각분야의 文書形式에도 滿足할 수 있는 style파일을 提供하고 이런 여러 形式을 간단한 方法으로 使用者가 選擇할 수 있도록 한다.
- 위와같은 스타일파일을 자세히 說明함으로서 使用者가 이런 基本적인 文書形式을 바탕으로 自身의 獨自的 要求를 쉽게 이룰 수 있게 한다.

이렇게 LaTeX이 國際化되어가는 동안 韓國 KAIST 電算學部에서는 LaTeX으로 한글을 쓸 수 있도록 매크로를 만들고 한글글꼴을 준비하여 hlatex(최우형, 백윤주)을 普給하였읍니다.([최우형92]를 參考) 여기에는 기존의 LaTeX에 한글字型을 追加하는 `hfont.tex`, 한글 文書作成 環境을 定義하는 `harticle.sty`과 `hreport.sty`, `hbook.sty` 그리고 한글 KS完成形코드를 TeX이 理解를 할 수 있는 TeX매크로로 變換해 주는 前處理機 htex로 構成되어 있읍니다. 普給되는 글꼴의 使用을 除限받으면서 出發한 hlatex은 그런대로 電算網을 통해 퍼지게 되었고 이에 바탕을 두고 漢字의 使用을 追加한 jhtex(殷 光熙)가 생길 수 있게 되었읍니다. 여기에는 손쉽게 구할 수 있는 日本의 Kanji字型을 HLaTeX에 덧붙쳐 使用할 수 있게 하고 文書樣式을 좀더 한글화할 수 있도록 하였으며 글꼴이 늘어남으로서 不可避해진 NFSS의 導入이 이루어졌읍니다. 이 두 프로그램은 좀더 낳은, 하나의 프로그램으로 一般에게 普給될 수 있고자 한글LaTeX으로 統合되었는데 이로서 한글LaTeX는 다음과 같이 變化하였읍니다(한글LaTeX-0.92e).

① LaTeX 2$_\varepsilon$에 의한 한글文書作成

② KSC式 漢字 使用

③ 前處理機의 廢棄

물론 多樣한 글꼴의 普及으로 어느 정도 變化있는 한글文書를 作成할 수 있었지만, 글꼴들은 쏘스가 없이 이미 만들어진 300dpi와 600dpi의 pk파일로만 提供되어 한글LaTeX의 使用도 당연히 여기에 制限을 받아 왔읍니다. 한글LaTeX-0.93부터는 文化體育部에서 公開한 글꼴과 Postscript type I글꼴, 그외에도 bitmap에서 抽出된 외각선글꼴이 METAFONT쏘스로 普及됨으로서 한글LaTeX의 사용도 一般化되게 되었읍니다. 글자의 配置도 이전의 KAIST式인 粗合完成混合式을 抛棄하고 一般的인 KS完成型에 根據하게 되었읍니다[6].

第 2 節　한글LaTeX의 使用 說明

[5] Leslie Lamport, Johannes Braams, David Carlisle, Alan Jeffrey, Frank Mittelbach, Chris Rowley, Rainer Schöpf: 이들은 [GMS94]을 出刊하였는데 이 책의 受入은 그 절반이 LaTeX3 project team에 割當되어 이 team을 支援하게끔 되어 있읍니다.

[6] 使用者의 要求에 의해 粗合完成混合式도 以前처럼 使用할 수 있도록 하였읍니다. 이 混合式은 使用자층을 制限하는 關係로 많이 支援되지 않습니다.

이 單元에서는 LaTeX의 基本的인 使用法에 대한 說明을 피하고, 주로 한글LaTeX에 關係하는 部分에 力點을 두어 記述합니다. LaTeX의 使用法에 관해서는 이에 該當하는 參考書籍을 보시기 바랍니다. [Lam85]를 參考로하면 LaTeX에는 수많은 命令들이 있음을 알 수 있습니다. 一般 使用者로서 이런 모든 命令을 全部 알아야 할 必要는 없읍니다. 個人의 慾求에 따라 약간의 基礎智識에, 必要할 때마다 [Lam85]를 參考하여 使用하시면 어렵지 않게 LaTeX文書作成을 始作할 수 있읍니다. 여기 獨逸에서는 [Kop91]이 잘 알려져 있읍니다. 美國에서 英語로 飜譯된 版이 韓國에 나와 있는 것으로 알고 있읍니다. 參考文獻으로 가장 많이 推薦되는 책입니다.

2.1　한글LaTeX文書의 틀

文書는

`\documentclass[option]{class}`

로 始作합니다. `option`에는 LaTeX의 class option외에도 한글LaTeX의 option `hanja`가 指定될 수 있읍니다. 이 option `hanja`로 文書의 單元名이나 날짜를 漢字로 表記할 것인가 아닌가를 決定합니다. option `pshan`은 한글포스트스크립트글꼴을 쓰고자 할 때, `oldhan`은 以前의 粗合式 한글글꼴을 쓰고자 할 때 그리고 `hardbold`는 METAFONT에 의해 만들어지는 두꺼운 글씨체를 쓰고자 할 때 指定되어져야 합니다.

`\usepackage{hangul}`

로 한글 매크로를 부릅니다. `hangul` 이외에 다른 package를 함께 쓰고자 할 境遇에는 콤마(,)를 사이에 두고 그 package이름을 씁니다. `hangul`의 option은 class의 option에서 정합니다. option의 處理方式은 다음과 같읍니다. 즉, `documentclass`의 option은 global합니다. 어느 class나 package에서든지 이 option을 處理할 수 있읍니다. `usepackage`의 option은 local입니다. 그 package에서 이 option을 받아들일 수 있어야 합니다. 그러므로

`\usepackage[pshan]{hangul,a4}`

에서는 지정된 옵션`pshan`을 두 package `hangul`과 `a4`가 모두 處理할 수 있어야 합니다. package `a4`는 `pshan`이라는 option을 處理할 수 없으므로 당연히 에러가 發生하게 됩니다. 다음과 같이 각기 따로 따로 指定하거나

`\usepackage[pshan]{hangul}`

`\usepackage{multicol}`

아니면 問題가 되는 option `pshan`을 global하게 `documentclass`의 option으로 指定하면 에러의 發生을 막을 수 있읍니다.

`\begin{document}`

로 文書를 시작하여 쓰고자 하는 글을 作成한 뒤

`\end{document}`

로 文書作成을 마칩니다.

이전에 作成한 文書는 LaTeX 2_ε의 compatibility모드를 통해 處理됩니다. 이 compatibility모드는

`\documentclass`

대신

`\documentstyle`

로 文書를 始作할 때 自動的으로 불러지는데 이때 文書의 포맷은 既存의 LaTeX209때와 같은 식으로 됩니다. 字型을 選擇하는 方法도 舊式이므로 文書를 짜는데 드는 時間이 더 길게 걸립니다.

이제 `latex`으로 파일을 돌리면 됩니다.

2.2　한글LaTeX 매크로

한글LaTeX은 LaTeX을 한글화하는 文書樣式(`hangul.sty`)과 한글글꼴을 紹介하는 매크로파일(`hfont.sty`), 글꼴을 定義하는 파일(`Uhangul.fd`와 `Uhanja.fd` 내지는 `Uhsymbol.fd`)로 構成되어 있읍니다.

hangul.sty은 LaTeX에 定義된 매크로들 중에서 우리의 한글 文書環境에 必要한 部分만을 修正하여 한글화한 것입니다. 그외에 로고 \HLaTeX과 enumerate環境에 쓰일 한글 숫자들 \jaso, \gana, \ojaso, \ogana, \pjaso, \pgana, \onum, \pnum, \oeng, \peng등이 定義되어 있읍니다. 그래서 우리가 다음과 같은 定義를 할 때

```
\renewcommand{\labelenumi}{\gana{enumi})}
\renewcommand{\labelenumii}{\jaso{enumii}.}
\renewcommand{\labelitemi}{☞}
```

다음의 text는

```
\begin{enumerate}
\item 한글
  \begin{enumerate}
  \item 현대어
  \item 고어
  \end{enumerate}
\item 漢字
  \begin{enumerate}
  \item 國語
  \item 中國語
  \item 日本語
  \end{enumerate}
\item 심벌
  \begin{itemize}
  \item ㉿
  \item ₩
  \item ㉾
  \end{itemize}
\end{enumerate}
```

이렇게 나타납니다.

가) 한글
   ㄱ. 현대어
   ㄴ. 고어
나) 漢字
   ㄱ. 國語
   ㄴ. 中國語
   ㄷ. 日本語
다) 심벌
   ☞ ㉿
   ☞ ₩
   ☞ ㉾

글꼴에 관계되는 매크로들은 모두 hfont.tex에 定義되어 있읍니다. 字型轉換 매크로로는 \mj=\명조, \gt=\고딕이 있읍니다. 심벌은 字型轉換 매크로를 必要로 하지 않습니다. 위와 같은 基本的인 한글LaTeX글꼴 이외에도 \gs=**궁서**, \gr=\그래픽, \mg=**목각**, \yt=**옛글**, \ol=\외곽 등의 글꼴을 設置하여 使用하실 수 있읍니다. 漢字는 여기에 該當하는 글꼴이 없으므로 基本 글꼴인 明朝體나 고딕체로 代身하도록 하였읍니다. 즉 그래픽, 木刻 그리고 옛글체는 고딕체가 代身하고 궁서와 外廓체는 明朝體가 代身합니다. 글자의 屬性에 따라 木刻, 옛글 그리고 궁서체는 漢字를 두껍게 찍히도록 합니다.

그리고 전 文書의 한글글꼴을 定義해 주는 매크로가 있는데

```
\mjdefault 명조
\gtdefault 고딕
\grdefault 그래픽
\gsdefault 궁서
\mgdefault 목각
\ytdefault 옛글
```

이 命令들은 單元名이나 페이지머리의 글꼴에도 影響을 줍니다. 이들이 文書作

成에 도움이 되기를 바랍니다. 예를들어 文書에서 `\gsdefault`를 指定해 주었을 경우 특별히 字型轉還을 시키지 않는 한 모든 한글글꼴이 궁서체로 써지게 됩니다. 역으로 本文에서 글꼴轉換을 시키면 單元이나 페이지머리등은 궁서체가 되고 本文은 다른 글씨체가 됩니다. 글꼴轉換시 留意해야 할 점은 **궁서**, **목각**, **옛글**, 외곽의 경우 `bfseries`와 `mediumseries`의 區別이 없는 것입니다. 더 以上의 說明이 필요없을 것으로 생각듭니다만 **궁서**, **목각**, **옛글**은 두꺼운 글씨에 속하고 외곽은 보통의 글씨에 속합니다.

우리말의 분철은 英語권 言語와는 다른 特別한 處理方法이 要求되어 왔는데, `hfont.tex`內部에서는 우리말의 글자 사이가 `\hyphenchar` 없이 분리될 수 있게 하고 있습니다. 한글LaTeX-0.94에서 도입된 `\noHyphen`命令은 필요없게 되었읍니다.

한글LaTeX 버전 0.92부터는 KSC 5601 코드중 256개의 심벌字型을 使用하여 한글字素(현대어·古語) 및 일반의 LaTeX매크로로서 얻을 수 없는 몇몇 심벌들을 쓸 수 있도록 하였읍니다. 附線 C.4에는 한글LaTeX에서 쓸 수 있는 심벌코드들이 나열되어 있읍니다. 참고하시기 바랍니다.

`hanterm`을 使用하면 이런 심벌중 몇몇 자소를 쓸 수 있고 `hanterm-hanja`와 `mule`을 使用하면 모든 심벌을 다 쓸 수 있읍니다[7].

表 1에는 "LaTeX의 한글 및 漢字化"가 要約되어 있고 [최우형92]에서는 hlatex에 대한 情報를 얻을 수 있읍니다. 表 1의 한글화는 `\begin{document}`앞에서 `\renewcommand`로 재정의하면 원하는데로 변경될 수 있읍니다.

`hangul.sty`이 위와 같은 많은 部分을 한글化하는 반면에, 境遇에 따라서는 그저 文書에 한글 및 漢字를 包含하기만 하여야 할 때가 있읍니다. `hfont.sty`[8]은 이런 技能만을 提供합니다.

2.3 한글 포스트스크립트글꼴

포스트스크립트글꼴은 원래 TeX과는 상관이 없는 글꼴입니다. TeX은 METAFONT로 글꼴을 만들어 쓰도록 되어 있읍니다. Donald Knuth敎授는 TeX에 使用될 글꼴을 TeX이 출력하게 될 프린터에 適當한 形態로 만들어 使用하도록 具象하였읍니다. 그 半面에, 廣範圍하게 普及되어져 있는 포스트스크립트는 글꼴을 모든 포스트스크립트프린터에 다 使用될 수 있도록 構成되어 있읍니다. 어느 方法이 더 낳다고는 할 수 없지만 서로 自己 주장을 내새우며 論爭을 벌이는 境遇가 많이 눈에 띄입니다. 거의 모든 프린터는 曲線을 그릴 수 없읍니다. 適當한 곳에 점을 찍어 曲線처럼 보이도록 할 뿐입니다. METAFONT쏘스코드는 이런 曲線을 描寫해 놓고 있읍니다. 이 쏘스로부터 만들어지는 pk파일은 프린터가 실제로 프린트할 때 찍을 点들을 프린터의 屬性에 適合한 형태로 構成되어 있읍니다. Knuth敎授는 이러한 pk파일들을 貯藏할 수 있는 hard disk의 容量에 觀點을 두지 않고 프린트되어져 나오는 文書의 質에 力點을 두었읍니다. 포스트스크립트코드는 METAFONT쏘스코드와 같은 種類의 形態입니다. 그러나 이 코드

[7] `mule`의 경우 `hsymbol.el`파일이 必要합니다. 이에 대해서는 11페이지를 參照하십시요.
[8] 이 파일은 버전 0.92e 以前까지는 `hfont.tex`였었읍니다.
[9] a:article, b:book, r:report, l:letter

LaTeX명령	한글LaTeX(한글)	한글LaTeX(漢字)	스타일[9]
ksabstractname	요 약	要 約	a r
ksccname	사본	寫本	l
ksenclname	첨부물	添附物	l
ksfigurename	그림	그림	a b r
ksheadtoname	받는이	受信人	l
ksindexname	색 인	索 引	a b r
kslistoffigurename	그림 목차	그림 目次	a b r
kslistoftablename	표 목차	表 目次	a b r
kspagename	페이지	페이지	l
ksseename	참조	參照	a b r
kstablename	표	表	a b r
kscontentsname	목 차	目 次	a b r
ksbibname	참고 문헌	參考 文獻	b r
ksrefname	참고 서적	參考 書籍	a
kstoday	1994년 3월 6일	1994年 3月 6日	a b r
ksappendixname	부록	附錄	a b r
ksTHE	편, 장, 절에 사용		
kspartname	제 I 편	第 I 篇	a b r
kschaptername	제 1 장	第 1 章	b r
kssectionname	제 1 절	第 1 節	a b r

表 1: LaTeX의 한글 및 漢字化

를 달리 變換시키지 않고 그대로 프린터에 보내게 되어 있읍니다. 이 코드에는 많은 曲線이 있읍니다. 프린터는 曲線을 그릴 수 없읍니다. 그러므로 프린터내에서 이 曲線에 대한 情報를 점들로 變換시키는 作業이 이루어집니다. 작은 解像度의 프린터에서는 힘든 作業입니다. 技術의 發達로 프린터의 技能이 많이 向上되었읍니다. 포스트스크립트글꼴의 使用도 많아지게 되었읍니다. 文書作成에 포스트스크립트글꼴을 쓰는 것도 많이 要求되어졌읍니다. 많은 사람들의 努力으로 TEX에서 포스트스크립트글꼴을 쓸 수 있도록 되었읍니다. dvi파일이 만들어질 때까지는 TEX은 포스트스크립트에 대해 아는 바가 없읍니다. 프린트하는 driver가 포스트스크립트를 認識하여 종이에 포스트스크립트글꼴을 使用하여 출력될 수 있게 되는 것입니다.

이런 상황에 맞춰 한글LATEX는 한글 포스트스크립트글꼴을 普及하면서 한글LATEX에 포스트스크립트글꼴을 使用할 수 있도록 하였읍니다. 위에서 說明한 바와 같이 포스트스크립트는 TEX와 그 根本을 달리 하고 있으므로 포스트스크립트글꼴을 使用하는 한글LATEX에 다른 特別한 使用法은 없읍니다. 모든 文書를 普通의 境遇와 같이 作成하고 글꼴만 포스트스크립트글꼴을 쓰도록 하면 됩니다. hangul.sty의 옵션으로 pshan을 쓰시면 됩니다. 다음을 參考하십시요.

```
\usepackage[pshan]{hangul}
```

혹은

```
\documentclass[pshan,...]{book}
\usepackage{hangul,..}
```

힘든 部分은 dvi파일의 driver를 setup하는 것입니다. 7페이지의 附線 A.2에 약간의 도움이 될 수 있는 情報가 있읍니다.

附線 A 한글LATEX의 設置

A.1 한글LATEX

한글LATEX의 設置는 簡單합니다. 한글LATEX에서 提供하는 스타일파일과 글꼴을 定義하는 파일, 글꼴을 띄우는 파일을 TEX이 찾을 수 있는 곳에 두면 됩니다. 다음의 표를 參考하십시요. pshan은 PostScript 한글 글꼴을 使用하고자 할 때, oldhan은 以前의 KAIST 粗合形 한글 글꼴을 使用하고자 할 때, "基本"은 이런 옵션에 該當하지 않는 境遇를 나타냅니다.

예를 들어

```
mkdir $TEXINPUTS/hangul
cp hangul.sty hfont.tex Uhangul.fd Uhanja.fd Uhsymbol.fd \
   halpha.bst $TEXINPUTS/hangul
cp hbibtex /usr/TeX/bin
```

以上으로 한글LATEX의 核心部分이 設置됩니다.

A.2 글꼴

글꼴은 CTAN(Comprehensive TeX Archive Network)의 fonts 디렉토리에 있읍니

	基本	pshan	oldhan
hangul.sty	○	○	○
hfont.sty	○	○	○
hfont.tex	○	○	
hfontold.tex			○
Uhangul.fd	○		
Uhanja.fd	○		○
Uhsymbol.fd	○	○	○
Uphangul.fd		○	
Uphanja.fd		○	
Uoldhan.fd			○
halpha.bst	bibtex		
hbibtex	bibtex		

表 2: 한글LATEX파일

다. 다음의 파일들을 `~ftp/tex-archive/fonts/korean/HLaTeX`에서 찾아 보시기 바랍니다.

`ftp.linguistik.uni-erlangen.de`의 `/pub/hangul/HLaTeX`에는 CTAN에서 구할 수 없는 *em*TEX용 `.fli`포맷의 글꼴들도 준비되어있음을 알려드립니다.

MetaFont 글꼴	PostScript 글꼴	hlatex PK 글꼴
Hangul-Myoungjo-mf	PS-HLaTeX-pfb	Johab-Hangul-pk300
Hangul-Gothic-mf	PS-HLaTeX-afm	Johab-Hangul-pk600
Hangul-Symbol-mf	PS-HLaTeX-tfm	Johab-Hangul-tfm
Hangul-Graphic-mf	PS-HLaTeX-vf	
Hangul-Gungseo-mf		
Hangul-Mokgak-mf		
Hangul-Yetgul-mf		
Hanja-Myoungjo-mf		
Hanja-Gothic-mf		
HLaTeX-tfms		
bzrbase.mf		

表 3: 보급되는 글꼴

完成形글꼴을 씀으로서 이전의 KAIST식 粗合完成方式에서 눈에 띠던 몇몇의 弱點을 補强할 수 있었던 반면, 많은 貯藏容量이 要求되어서 글꼴設置시 이에 대한 考慮를 하여야 합니다.[10] 하드웨어의 값이 暴落하여 그리 큰 問題点이라고는 生

[10] 이러한 問題点을 深刻하게 여기는 분들은 옛날식의 한글LATEX 글꼴을 使用하십시요. 이 境遇 300dpi와 600dpi의 프린터만이 支援됩니다. 한글LATEX의 使用이 이런 두가지 種類의 프린터에만

覺되지 않는데 사용할 글꼴의 選擇은 使用者가 決定하시기를 바랍니다.

다음의 電子郵便을 읽어보시면 하드웨어의 값이 暴落하므로서 프로그래머들이 얼마나 많은 수고를 덜을 수 있는가를 알 수 있읍니다.

```
Date: Wed, 18 Oct 1995 14:04:51 -0700 (PDT)
From: Paul Vojta <vojta@math.berkeley.edu>
To: koaunghi.un@zdv.uni-tuebingen.de
Subject: Re: Xdvi + Postscript Fonts
```

No, I have no plans for an xdvi that can take type 1 fonts. That would be a huge amount of work.

Your best bet would be either to accept the loss of 100MB disk space (disks are getting cheaper all the time), or use dvips in combination with ghostview.

Sincerely,

```
Paul Vojta vojta@math.berkeley.edu
```

글꼴이 차지하는 容量을 節約하는 方案으로 Postscript Type I 글꼴를 提供하기로 하였읍니다. previewer인 `xdvi`가 아직 Postscript 폰트를 읽을 수가 없어, 하나의 프린터만 使用하는 시스템의 境遇 별 커다란 節約을 할 수 있는 것은 아니지만 境遇에 따라서는 많은 容量이 節約될 수 있읍니다. 이런 境遇에 속하려면

- `dvips+ghostview`로 preview를 합니다. 다음과 같은 命令을 使用하면 됩니다.

  ```
  dvips -f <file> | ghostview -
  ```

  이 境遇 포스트스크립트글꼴이 `ghostscript`에 알려져 있어야 하며, ghostview에서는 擴大의 기능이 되지 않습니다[11]. 글꼴 登錄을 위해 한글LATEX의 Fonts/config디렉토리에 Fontmap.hangul을 參考로 하도록 하였읍니다. preview 作業의 速度는 상당히 느립니다. pfb글꼴이 지저분하게 코딩되어서 作業속도에 더욱 支障을 줍니다.[12] 勿論 print하는데도 지장을 줍니다.

- `ps2pk` 및 `gsftopk`를 使用하여 `xdvi`가 Postscript字形을 보여 줄 수 있도록 하는 方法이 있지만 결국은 必要한 PK파일을 만들어야 하기는 마찬가지입니다. 高度의 解像度를 가진 프린터를 使用할 境遇에는 300dpi로 preview하면 hard space를 節約할 수도 있읍니다. 즉 preview의 모드를 300dpi로 設定하여 ps2pk나 gsftopk를 利用하여 preview용 pk파일들을 만들도록 하고 프린트는 pfb파일을 씁니다. 300dpi以外의 다른 프린터가 함께 共存하는 시스템에서는 많은 容量을 節約할 수 있읍니다.

制限되기를 원치 않음으로 이 글꼴에 대한 支援은 거의 없게 될 것으로 보입니다.

[11]ghostview가 standard input을 作業하면 擴大등의 技能이 痲痺됩니다.

[12]얘기 듣기로는 Fontlab for Windows와 같은 프로그램으로 글꼴을 簡單히 操作할 수 있다고 합니다. 이런 프로그램을 가지고 계신 분이 있으시면 글꼴 修整에 도움을 주실 수 있기를 바랍니다. 제게 連絡 주십시오.

한글 포스트스크립트글꼴들은 그런대로 使用에 適合합니다. 漢字 포스트스크립트 중 明朝體는 失望을 불러 일으킬 것입니다. 水平線이 너무 가늘어서 찍히지 않은 效果를 보여줄 것이기 때문입니다. 4888개의 글자를 손으로 修整하려니 時間이 너무 많이 要求되어서 엄두를 내지 못하고 있습니다.

글꼴設置는 명조와 고딕을 基本글꼴로 하고 그외에 하드디스크의 남은 容量에 따라 아래의 表를 보고 다른 글꼴을 追加하고 削除하는 것이 決定에 도움이 될 수 있겠다고 생각듭니다 .

글꼴	MF쏘스	PK(300dpi)	포스트스크립트
한글명조	8 102 kB	15 094 kB	899 kB
한글고딕	272 kB	10 504 kB	404 kB
漢字明照	26 777 kB	38 492 kB	2 795 kB
漢字고딕	16 305 kB	31 257 kB	1 839 kB
한글㉿あ	1 277 kB	3 641 kB	
한글그래픽	4 141 kB	12 221 kB	537 kB
한글궁서	10 758 kB	7 649 kB	1 114 kB
한글목각	25 121 kB	6 365 kB	2 331 kB
한글옛글	9 974 kB	6 233 kB	1 078 kB
한글외곽	21 kB	11 107 kB	
總	102 749 kB	142 563 kB	10 998 kB

글꼴의 設置는 두가지 方法을 提案합니다.

1. METAFONT쏘스를 $MFINPUTS에 두고 이 글꼴에 해당하는 TFM파일들은 시스템의 $TEXTFMS에 둡니다. 이렇게 하면 必要한 PK파일들은 그때 그때 必要에 따라 한글LaTeX파일을 짜고 난 후 만들어진 DVI파일을 preview하거나 print할 때 만들어지게 됩니다.
2. METAFONT쏘스로부터 必要한 PK파일들을 만들어 이를 $TEXFONTS에 두고 TFM파일들을 $TEXTFMS에 둡니다. 쏘스는 다시 없앱니다.

METAFONT쏘스들이 상당한 크기이므로 첫번째 提案은 그렇게 賢明하지가 않다고 생각합니다. 특히 한글은 하나의 \magstep에 2350개의 글자가 있게 되고 漢字의 境遇는 모두 4888개이므로 作業 中間에 PK파일을 만들게 하는 것은 作業의 흐름을 막는 境遇가 되기 쉽습니다. 어차피 必要하게 될 글꼴이라고 생각이 들면 컴퓨터가 일하지 않는 밤시간을 利用해 미리 PK파일을 만들어 놓는 것이 바람직하다는 意見입니다.

모든 決定이 끝나면 Fonts 디렉토리로 場所를 옮겨서 Makefile을 修整합니다. 이렇게 修整한 結果에 따라 make를 實行시킵니다. Makefile에 仔細한 說明이 있읍니다. 參考하시기 바랍니다. 作業이 끝나면 Makefile에서 정한 $MODE디렉토리에 字型들이 만들어져 있읍니다. 여기서 모든 .tfm파일들은 $TEXTFMS디렉토리로 .pk파일들은 $TEXFONTS디렉토리로 옮깁니다.

A.3 $\LaTeX\,2_\varepsilon$

한글LaTeX은 LaTeX 2ε를 基盤으로 움직입니다. 그러므로 system에 LaTeX 2ε가 設置되어 있어야 한글을 쓰실 수 있읍니다. 가까운 CTAN에서는 모두가 anonymous ftp로 /tex-archive에 있는 모든 TeX software를 自由롭게 가져다 쓸 수 있읍니다. 다음과 같이 LaTeX 2ε를 가져옵니다.

```
hpserv:> ftp ftp.uni-stuttgart.de
login: ftp
passwd: ftp
ftp> cd /tex-archive/macros
ftp> bin
ftp> get latex.tar.gz
ftp> bye
```

가져온 파일을 풀고 LaTeX 2ε를 設置를 합니다. 以前의 버전 LaTeX209가 設置되어 있으면 먼저 이를 backup합니다.

```
mv $TEXINPUTS/latex $TEXINPUTS/latex209
mkdir $TEXINPUTS/latex
tar xzvf latex.tar.gz
cd latex/base
initex unpack.ins
initex latex.ltx
cp latex.fmt $TEXFORMATS
cp *.sty *.tex *.fd *.clo *.cls *.def $TEXINPUTS/latex
```

이 외에도 `packages` 및 `contrib` directiory의 매크로들을 設置합니다. 大部分의 境遇 새로 設置된 `latex`으로 `.ins`파일을 돌리면 됩니다. 예를 들어

```
cd ../packages/tools
latex tools.ins
```

附錄 B Mule

Emacs는 지금까지 强力한 editor로서 君臨해왔읍니다. 이런 强力한 道具로 한글도 쓸 수 있읍니다. 日本의 Kenichi Handa氏(handa@etlken.etl.go.jp)는 GNU Emacs를 修整하여 多國語를 쓸 수 있도록 Mule(MULtilingual Enhancement of GNU Emacs)을 만들었읍니다. `auctex`은 Emacs내에서 손쉽게 LaTeX文書를 作成하고 作業하는데에 필요한 모든 環境을 提供하며 Mule 및 Nemacs를 支援합니다. 그러므로 한글LaTeX文書를 이러한 Editor와 보조프로그램으로 作成하면 많은 便利를 얻을 수 있읍니다.

여기서는 Mule에서의 한글 部分만을 說明합니다.

B.1 Mule 버전 2.0

1. **Mule을 設置하기 전에** `contrib/mule-2.0` **디렉토리에 있는** `hanja.el`**과** `hanja3.el` **및** `hsymbol.el`**을 Mule의** `lisp/quail` **디렉토리에 둡니다.**

2. `lisp/site-init.el`[13]**에서** `korean.el`**을 loading하도록 합니다.** 다음을 參考로 하십시오.

```
(define-program-coding-system nil ".*mail.*" *iso-2022-kr*)
(load "korean")
(set-default-file-coding-system *euc-kr*)
```

3. `lisp/korean.el`**에 쓰고자 하는 한글 package를 loading하도록 합니다.** 2벌식 자판용으로는 `quail/hangul.el`과 `its/hangul.el`이 있고 3벌식용으로는 `quail/hangul3.el`이 있으며 漢字의 境遇 `quail/hanja.el`이 사용됩니다. Quail–package에 依存하여 한글과 漢字를 입력할 境遇에는 먼저 漢字를 loading하고 다음에 한글을 loading하여야 후에 `mule`을 實行한 후 Quail모드에 들어가면(`C-]`) 먼저 한글을 입력하는 모드가 자동으로 選擇되고 이 狀態에서 quail–package를 바꿀 때(`M-s`) 漢字입력 모드를 選擇할 수 있읍니다. 그러므로 loading은 심벌, 漢字, 한글의 順序로 하는 것이 좋습니다.

4. **Mule을 設置합니다.**

Mule이 이미 設置되어 있을 境遇는 `$HOME/.emacs`나 `lisp/site-start.el`에서 變更 및 追加되어야 할 事項을 정해줍니다. 다음을 參考하십시오.

```
(define-program-coding-system nil ".*mail.*" *iso-2022-kr*)
(load "korean")
(set-default-file-coding-system *euc-kr*)
(load "quail/hsymbol")
(load "quail/hanja")
(load "quail/hangul")
(if (boundp 'EGG)
    (progn
      (load "its/hangul")
      (setq its:*standard-modes*
            (cons (its:get-mode-map "hangul")
                  its:*standard-modes*))
      (setq-default its:*current-map*
                    (its:get-mode-map "hangul"))))
```

[13] `site-init.el`은 `mule-init.el`의 修整할 部分들을 뽑아서 만들어야하는 파일로서 Mule의 쏘스에는 없읍니다.

B.2 Mule 버전 2.1부터

Mule 버전 2.1에는 hanja-???.el이 있읍니다. 이중에서 hanja-ksc.el은 漢字를 KS식으로 코딩하고 hanja-jis.el은 漢字를 이 전의 jhtex에서 使用하던 JIS식으로 코딩합니다. 심벌용 lisp코드는 버전 2.2서부터 提供되므로 contrib디렉토리의 hsymbol.el을 Mule의 Quail디렉토리에 複寫하고 이를

```
byte-compile-file
```

합니다. 그리고 모든 Quail모드들은 Mule을 띄운 후 必要에 따라 loading되므로 이전과 같이 mule을 compile할 때 korean.el이나 site-init.el에서 loading시킬 必要는 없어졌읍니다[14]. 어떤 모드가 要求되어졌을 때 어디에 있는 lisp코드를 읽으라는 情報만으로 Mule은 compile되어집니다. 이런 情報는 이미 mule-init.el에 定義되어져 있으므로 우리는 hsymbol.el만 追加로 認識시키면 됩니다. 다음과 같은 코드들이 site-init.el의 對象이 됩니다.

```
(set-primary-environment 'korean)
(define-program-coding-system nil ".*mail.*" *iso-2022-kr*)
(add-hook 'quail-package-alist '("hsymbol" "quail/hsymbol"))
(set-default-file-coding-system '*euc-kr*)
```

Quail모드의 한글코드 處理方式이 改善됨으로서 QUAIL을 使用하여도 한글입력이 편리합니다. 그러나 漢字 辭典을 利用한다던가 하는 理由로 Wnn/EGG를 利用한 ITS식 한글處理方式을 使用하시고자 할 때에는 다음의 코드가 追加됩니다[15].

```
(if (boundp 'EGG)
    (progn
      (setq wnn-server-type 'kserver)
      (load "its/hangul")
      (setq its:*standard-modes*
            (cons (its:get-mode-map "hangul")
                  its:*standard-modes*))
      (setq-default its:*current-map*
                    (its:get-mode-map "hangul"))))
```

Mule을 compile할 때 --mcpath옵션을 주었을 경우 다음을 定義해 줍니다.

```
(setq pathname-coding-system '*euc-kr*)
```

이미 Mule이 設置되어 있을 境遇는 위의 모든 코드를 lisp/site-start.el이나 $HOME/.emacs에서 정해줍니다.

[14]그러나 미리 loading하면 한글·漢字 입력모드로 들어가는 時間이 短縮됩니다.

[15]Wnn4.2에는 kserver가 따로 있읍니다. Wnn6.0부터는 商業用으로 변했읍니다. Wnn4.2가 마지막 public버전이 되었읍니다. Wnn6.0의 長點은 아는 바가 없지만 Wnn4.2를 使用하는 현재별 問題없이 많은 도움을 받고 있읍니다.

B.3 Mule을 update할 때

JIS漢字用 lisp코드에는 아직도 뜻모르는 漢字가 많습니다[16]. 그래서 새로운 漢字를 찾을 때 마다 우리는 Mule을 update해야 합니다. 이 외에도 `mule` binary에 dumping되어 있는 어떤 lisp코드가 必要에따라 變更될 수도 있습니다. 그 때마다 Mule을 새로이 compile하는 것은 바람직하지 않습니다. 이를 위해서 다음의 trick을 알고 있는 것이 좋습니다. Mule 버전 2.1에서는 모든 Quail코드들이 binary에 dumping되어있지 않고 必要에 따라 loading되므로 그저 修整된 lisp코드를 `lisp/quail`에 複寫한 후 이를 `byte-compile-file`하시면 됩니다.

Mule 버전 1.1.pl04까지 Mule을 compile하면 `temacs`가 생깁니다. 이는 아무런 Emacs lisp파일들이 loading되어 있지 않은 pure emacs입니다. Mule의 設置가 끝난 후 이 `temacs`를 Mule의 `etc` directory에 옮깁니다. 그 후에 dumping된 lisp파일들이 變更되면 이 파일들을 Mule의 `lisp/quail` directory에 複寫한 후 다음과 같이 새로운 實行파일 `xemacs`를 만듭니다.

```
cd $(MULE)/etc
temacs -batch -l loadup dump
```

Mule 버전 2.0부터 Emacs-19부터는 compile時에 `EMACS_DUMP_LOADPATH`가 정해져 있게 됩니다. 이는 emacs를 dumping할 때에만 쓰이는 `load-path`로서 여러분이 `$src`에서 compile했을 경우 이 path는 `$src/lisp`로 정해지게 됩니다. compile이 끝난 후 모든 쏘스를 지우지 않고 그대로 살려놓을 경우는 아무런 問題가 없겠지만 보통은 이를 지워버립니다. 그러므로 쏘스를 지운 후 다시 `$src` directory를 만들고 `$src/lisp`를 Mule이 install된 `lisp` directory에 link합니다. Install prefix가 `$install`이었을 경우:

```
mkdir $src
cd $src
ln -s $install/lib/mule/19.25/lisp .
```

그런 후 'Mule 버전 1.1.pl04까지'에서처럼 `temacs`를 dumping하여 `emacs`와 `emacs-19.25.1`[17]를 만듭니다. `temacs`는 `$install/lib/mule/19.25/etc`에 둡니다. 이렇게 해서 만들어진 實行파일을 `mule[-19.25]`로 이름을 바꾼 후 이를 이전의 `mule`이 있는 곳으로 옮깁니다.

B.4 Wnn4.2

Mule을 使用하여 漢字를 입력할 때 우리는 簡單히 QUAIL을 利用한 hanja-ksc.el을 씁니다. 그러나 Wnn을 設置하면 漢字의 입력이 매우 쉽게 이루어집니다. 예를들

[16]이 漢字들은 Mule의 漢字입력모드에서 `unknown`으로 불러낼 수 있읍니다. 이 파일은 이제 사용할 필요성이 없읍니다.

[17]이 둘은 그저 서로간의 hard link입니다

어 QUAIL방법은 한글의 글자 하나하나를 漢字로 變換시키는 半面에 Wnn을 使用하면 單語單位나 文章單位의 漢字變換이 可能합니다. 아직도 Wnn에 대해서 詳細하게 아는 바는 없지만 關心있는 분들에게 약간의 도움이 될 수 있지 않을까 하는 바람으로 아는 範圍내에서 이에 대해 간단히 紹介를 할까 합니다.

B.4.1　設置

두가지 方法이 있읍니다.

1. MIT X11R5가 設置되어 있을 경우 X11쏘스를 미리 compile하고 이를 基盤으로 Wnn을 compile합니다.
2. MIT X11R6가 設置되어 있을 경우 이미 設置되어 있는 X11의 library와 include header file을 사용하여 Wnn을 compile합니다.

자세한 事項은 Wnn4.2의 쏘스에 잘 說明이 되어 있읍니다. 쏘스는

```
ftp.kuis.kyoto-u.ac.jp:~ftp/Wnn
```

를 살펴보시기 바랍니다.

B.4.2　서버의 動機

Wnn이 시스템에 設置되면 辭典서버를 動機시킵니다. jserver는 日語辭典용이고 cserver 및 tserver는 中國語辭典용이며 kserver는 한글辭典용입니다. 한글辭典용 kserver를 띄우면 /tmp/kd_sockV4라는 소켓을 통해 가상의 유저인 wnn이 한글漢字變換辭典을 擔當하여 일을 하게 됩니다. wnn이 하는 일은 一般使用者의 要請에 따라 辭典을 管理하고 委託받은 한글에 적당한 漢字를 辭典에서 찾아 一般使用者에게 알려주는 것입니다. Linux에서는 처음에 kserver를 機動시킨 후, 소켓을 통한 커뮤니케이션이 잘 않되었었는데 chmod 777 /tmp/kd_sockV4로 問題가 解決되었던 經驗이 있읍니다. 이 경우 다음과 같은 alias로 辭典서버를 機動시키는 것도 도움이 될 수 있읍니다.

```
alias kserver 'su -l wnn -c /usr/local/bin/kWnn4/kserver'
```

wnn의 umask는 0으로 합니다. SUN-OS에서는 이런 문제가 없었읍니다. kserver를 動機시키면 다음과 같은 메세지가 나타납니다.

```
~> kserver

Korean Multi Client Server (4.20)
Reading /usr/local/lib/wnn/ko_KR/dic/sys/hword.dic       Fid = 1
Reading /usr/local/lib/wnn/ko_KR/dic/sys/single.dic      Fid = 2
Reading /usr/local/lib/wnn/ko_KR/dic/sys/full.fzk        Fid = 3
Finished Reading Files
```

B.4.3 Mule에서의 使用

$HOME/.emacs에 다음과 같은 定義가 必要합니다.

```
(if (and (boundp 'MULE) (boundp 'WNN4))
    (progn
      (setq-default wnn-server-type 'kserver)
      (set-kwnn-host-name "unix")))
```

위에서 호스트이름 unix는 실제의 호스트이름을 記入하거나 그대로 unix를 使用하여도 되는 것으로 알고 있읍니다. 初期化파일인 eggrc-wnn은 Mule의 lisp디렉토리에 있읍니다. 이 파일의 마지막부분에는 kserver에 該當하는 定義들이 있읍니다. 여기에는 辭典을 追加하는 난들이 많이 comment out되어 있읍니다. 이 중에서 sys/hword.dic을 追加하는 줄의 comment를 없앱니다[18].

mule을 띄운 후 C-\을 누르면 漢字變換모드에 들어 갑니다. 한글을 입력하면 SPC를 누르기 전까지 한글이 fense속에 기입이 됩니다. SPC를 누르면 다음과 같은 메세지가 나타날 것입니다.

```
"호스트 local 의 kWnn 를 機動했읍니다."
```

이 후에 wnn은 eggrc-wnn을 읽고 Mule로부터 委託받은 fense속의 한글에 맞는 漢字를 辭典에서 찾아 보여줄 것입니다. 원하는 漢字가 아니면, C-p, C-n으로 다른 漢字들이 있는가 봅니다. Return으로 이 fense-mode를 나올 수 있읍니다.

이미 한글로 쓰여져 있는 境遇에는 region을 정해 그 部分에 한해서 漢字變換을 할 수 있고, 또는 한 文章 내지는 한 文節 혹은 한 單語을 單位로 漢字變換을 할 수 있읍니다. 이 境遇 Mule은 이 region을 조각조각 나누어서 하나씩 wnn에 넘겨 주어서 漢字變換을 시키도록 하는데 한 部分의 變換이 원하지 않게 되었을 境遇에는 C-f, C-b로 그 部分으로 가서 위에서 처럼 C-p, C-n으로 다른 漢字 내지는 한글을 選擇할 수 있읍니다. 한 單語로 조각난 部分이 너무 길거나 짧음으로 해서 제대로 漢字變換이 이루어 지지 않았다고 判斷이 될 境遇에는 그 곳에 커서를 놓고 M-i, M-o로 그 길이를 길게 또는 짧게 함으로서 맞는 漢字를 選擇할 수 있읍니다. fense-mode에서 C-h로 도움말을 얻을 수 있읍니다.

diced를 통해 個人辭典에 單語를 追加하거나 削除할 수 있읍니다.

B.4.4 例文

[C-\ 한글 SPC]의 例

|한자| — |漢字|

[M-x henkan-sentence]의 例

[18] 事實 kserver가 機動되면서 시스템의 辭典은 모두다 읽혀지게 되어 있읍니다. — kserver가 機動될 때의 메세지를 읽으면 알 수 있읍니다. — 그런데 mule내에서는 이 事實이 認識이 되지 않는지 생각한데로 作業이 되지 않았읍니다.

> 자연과학은 이미 18세기에 그 기틀을 열었지만, 19세기에 들어가 산업 혁명이 진행되고 근대자본주의가 발달함에 따라 더욱 연구분야가 세분되고 특수화되었다.
>
> —
>
> 紫煙과학은 이미 18세기에 그 기틀을 열었지만, 19세기에 들어가 産業 혁명이 進行되고 近代자본주의가 發達함에 따라 더욱 硏究분야가 細分되고 特殊화되었다.

위에서 "紫煙과학은"은 잘못 變換된 것입니다. 이는 辭典에 登錄되어 있는 單語 "紫煙"의 使用頻度가 지금까지 가장 높은 것으로 나타나서 wnn은 이 漢字를 먼저 推薦한 結果입니다. "과학", "세기", "혁명", "자본주의", "분야" 등은 變換이 되지 않았는데 커서를 이 單語에 두고 M-i로 文節을 짧게 하면 變換이 됩니다. 최종으로 다음의 결과를 얻게 됩니다.

> 自然科學은 이미 18世紀에 그 기틀을 열었지만, 19世紀에 들어가 産業 革命이 進行되고 近代資本主義가 發達함에 따라 더욱 硏究分野가 細分되고 特殊化되었다.

附錄 C 한글·漢字 글꼴 만들기

한글LaTeX을 쓰는데는 한글·漢字字型이 必要합니다. 그러나 우리는 마음대로 쓸 수 있는 글꼴의 不足으로 한글LaTeX의 使用에 隘路를 겪고 있었읍니다. 이제는 除限없이 쓸 수 있는 글꼴이 많이 만들어졌읍니다. 이 單元이 글꼴을 만드는데 關心이 있으신 분들에게 도움이 되기를 바랍니다.

C.1 字型에 관련되는 Program들

JemTEX2 *Jem*TEX2는 DOS/OS2용의 *em*TEX에서 日本語를 쓸 수 있도록 만들어진 日本語 *em*TEX입니다([Jal]을 參考). 여기에는 日本에서 사용되는 모든 字型이 함께 提供되는데, TEX이 移餰性이 좋으므로 이 字型을 우리는 어느 System에서나 쓸 수 있읍니다. 그러나 이 字型들은 각각 128개의 글자로 構成되어 있읍니다. *Jem*TEX2의 著者 住所는 다음과 같읍니다.

François Jalbert	jalbert@CS.UBC.CA
220 Forest	jalbert@IRO.UMontreal.CA
Châteauguay, QC	
CANADA	
J6J 1R1	

ChTEX ChTEX은 中國語를 다룹니다. 여기에는 상당히 많은 漢字가 있읍니다[19]. 그러나 ChTEX은 postscript字形을 使用합니다. 더구나 現代式 中國漢字는 많이 변해 있읍니다. 그래서 韓國人으로서 알아보기 어려운 略字가 상당한 部

[19] 대략 2만자 이상

分을 차지하고 있읍니다. 著者의 住所는 다음과 같읍니다.

J.B. Wang jbw@unix.cis.pitt.edu
1921 Letsche St. jwang@pittvms.bitnet
Pittsburgh, Pa 15214
(412)-323-2060

Watanabe Kanji 字型 J*em*TEX2와 거의 같은 形式의 字型입니다[20]. 그러나 한 字型이 256개의 글자로 構成되는 差異가 있읍니다. 그래서 TEX이 認識하여야 할 字型情報를 J*em*TEX2와 比較하여 $\frac{1}{2}$밖에 必要로 하지 않습니다. bold face字型도 만들 수 있게 되어 있읍니다. 이 프로그램은 日本의 JaWaTEX의 著者인 Masatoshi Watanabe씨가 그의 JaWaTEX에 使用할 用度로 만들었읍니다.

Poorman's Chinese Poorman's Chinese는 中國語를 쓰기 위해 出發하여 이제는 日本語 및 한글 등을 같이 使用할 수 있도록 發展하고 있읍니다. 역시 여기에도 漢字 및 한글 字型이 있는데 이들은 그 構成을 매우 獨自的으로 하고 있읍니다. 즉 한 字型의 처음 127개 字形들은 그 크기를 달리하여 나머지 127개의 字形들로서 다시 같은 字型에 나타납니다. 그리하여 이 Poorman's Chinese는 두가지 크기의 字形만을 支援합니다. 著者의 住所는 다음과 같읍니다.

Thomas Ridgeway, Director
Humanities and Arts Computing Center, DR-10
University of Washington,
Seattle WA 98195 USA
telephone (206)-543-4218 ridgeway@blackbox.hacc.washington.edu

GNU Fontutils 이 유틸리티는 scanner를 통해 生成된 이미지를 出發占으로 하여 ghostscript字型과 METAFONT쏘스를 만드는 過程에 必要한 모든 環境을 提供합니다. `imageto`는 `.pbm`파일(portable bitmap)이나 `.img`파일(GEM)을 Encapsulated Postscript파일 등으로 變換시킵니다. `imgrotate`는 IMG파일을 90°나 180° 돌립니다. `fontconvert`는 pk나 gf파일을 epsf파일이나 gf 혹은 text파일로 變換합니다. `charspace`는 pk나 gf파일의 각글자 넓이를 調節합니다. `limn`은 pk나 gf파일로부터 bezier outline font形式의(BZR) 파일을 만듭니다. `bzrto`는 BZR파일로부터 ghostscript type I 내지 type III字型이나 METAFONT파일 그외 BPL(Bezier Property List)파일을 만듭니다. `bzredit.el`은 emacs내에서 BZR파일을 修整作業할 수 있는 環境을 提供합니다. `bpltobzr`는 修整된 BPL파일을 다시 BZR파일로 變換시킵니다. 이외에도 Xwindow에서 pk 내지는 gf파일을 修整할 수 있는 `xbfe`와 ghostscript字型을 bitmap포맷으로 바꾸어주는 `gsrenderfont`등이 있읍니다.

Karl Berry karl@cs.umb.edu
Kathryn Hargreaves letters@cs.umb.edu

PBMplus 이 toolkit에는 Jef Poskanzer氏에 의해 開發된 portable bitmap format 파일을 操作하는데 必要한 많은 유틸리티들이 있읍니다.

Jef Poskanzer jef@netcom.com

[20]사실 J*em*TEX2는 Watanabe씨의 도움으로 좀 더 낳은 프로그램이 될 수 있었읍니다.

NETpbm 電算網에 흩어져 있는 PBM에 關聯된 다른 프로그램들을 모아 여러사람들이 共同으로 위의 PBMplus에 包含시켜 하나의 package로 發展시켰읍니다.
netpbm@fysik4.kth.se

hp2xx HPGL言語를 METAFONT 및 다른 포맷으로 轉換합니다.
Heinz W. Werntges, Biokybernetik, werntges@convex.rz.uni-duesseldorf.de
Heinrich-Heine-Universitaet,
D-4000 Duesseldorf 1, Germany

fig2MF FIG포맷 파일을 METAFONT로 轉換합니다.
Anthony Starks ajs@merck.com

ps2mf PostScript type I 字型으로부터 METAFONT파일을 만듭니다. 여기에서는 PostScript binary format 字型을 ascii format으로 바꾼 후 이를 disasemble하여 여기에서 METAFONT파일을 만들어냅니다.
E.J. Vens E.J.Vens@icce.rug.nl
Rijks Universiteit Groningen
ICCE
P.O.Box 335
9700 AH Groningen
the Netherlands

ps4mf 위의 `ps2mf`를 發展시킨 프로그램입니다.
Markus Neteler neteler@laum.uni-hannover.de

ps2pk PostScript type I 字型으로부터 pk파일을 만듭니다.
Piet Tutelaers rcpt@urc.tue.nl

C.2 한글LaTeX에서 사용되던 글꼴들

HLaTeX 字型 hlatex-0.91에서 提供되는 여러가지 자형들[21]은 MacIntosh에서 만들어진 것으로 보입니다.

외곽(hol) 이 字型은 文化部字型을 根據로 합니다. 文化部에서는 96×96 bitmap 字型을 公開하였는데 이 bitmap은 `bm2mf`에서 使用하고 있는 字型形式과 同一합니다. 여기에는 完成形의 `96mt.bitmap`과 粗合形의 `96mht.bitmap`이 있읍니다. 그러나 한글LaTeX에서 使用하는 TeX字型은 完成形과 粗合形을 複合한 形式이므로 이 文化部字型은 一般의 그림format으로 轉換되 한글LaTeX에 맞도록 修整作業됩니다. 예를 들어 `96mht.bitmap`에는 粗合形의 字型이 들어 있읍니다. 이는 96×96 bitmap의 글자들이 header가 없이 모아진 파일들입니다. bitmap의 形式은 `pbm`입니다. `pbm` format은 8bit를 1byte로 coding하고 있으므로 $96 \times 96 \div 8 (= 1152 byte)$가 한 單位로 되어 있읍니다. 그러므로 `96mht.bitmap`을 1152byte씩 쪼갭니다.

[21]명조체, 고딕체, 궁서체, 필기체, 공작체, 디나루체, 그래픽체

```
split -b1152 96mht.bitmap
```

이렇게 쪼개진 파일들에 pbm파일의 header를 揷入합니다.

```
P4
96 96
```

이 파일들을 普通의 xpaint와 같은 그림프로그램으로 修整作業을 합니다.

이렇게 만들어진 pbm파일들을 모두 trace하여 hp plotter파일을 만듭니다. 이 plotter파일들을 각자 hp2xx를 통해 METAFONT쏘스로 轉換합니다. 이 쏘스들을 모아 한 파일로 만든 후(正確히 말하면 두 파일:holA.mf와 holB.mf) 適當히 修整하면 作業이 끝납니다.

고딕(hgt) 고딕체는 hlatex-0.9에서 提供되는 kgt[AB].746pk로부터 만듭니다. 이를 GNU의 fontutils중 limn을 使用하여 BZR format을 만듭니다. 이 bzr파일을 bzrto로 處理해 METAFONT쏘스를 얻습니다. 고딕체는 원래가 單純한 字型이므로 이렇게 얻어진 METAFONT쏘스의 複雜한 outline 情報를 直線과 曲線으로 單純化합니다.

漢字(hmj) hanja40.bit를 pbm파일로 만들어 각 그림의 不必要한 餘白을 없앤 후 hanja38.bit를 만듭니다. 이 bitmap을 바탕으로 bm2mf을 使用하여 일차 518pk를 얻어냅니다. emtex에서 提供하는 pkedit으로 이 pk파일들을 修整합니다. 이렇게 修整된 pk파일들로부터 pktopbm으로 낱개의 글꼴들을 얻어냅니다. header를 없앤 후 한데 모아 hanja65를 만듭니다.

```
pktopbm -H65 -W65 -c[#] [file] | tail -c585 -q >> hanja65
```

이 hanja65를 바탕으로 bm2mf에 의해 必要한 字型을 만듭니다.

심벌(hsym) 한글 KS코드중 必要하다고 생각되는 글꼴 256개를 選定하여 字型 만드는 프로그램으로 PostScript Type I의 글꼴을 만듭니다. 이렇게 만든 .pfb파일을 ps2mf로 處理해 METAFONT쏘스를 만듭니다. 이 METAFONT쏘스를 약간 修整하여 mdseries와 bfseries를 만듭니다.

C.3 새로운 完成形 글꼴

한글궁서, 한글그래픽, 한글옛글, 한자명조, 한자고딕 오래전 X11R4때에 發表된 GNU의 fontutils-0.6은 아직까지 아무런 修整이 없이 埋葬되어 가는 것으로 보입니다. 그래서 요즈음의 Xwindow시스템에서 이 유틸리티를 設置하는 것은 順調롭지가 않습니다. 일단 이 유틸리티가 設置되면 글꼴을 만드는데 必要한 것은 bitmap글꼴입니다. BZR파일이 만들어지면 어느 정도 한숨을 돌릴 수 있는 段階에 도달하는데, 이번의 境遇는 그 후에 더 많은 作業을 하여야 했읍니다. 다른 形態의 글꼴들이 서로 呼吸을 맞추지 못하

거나 크기 및 位置가 正確하지 않았기 때문입니다. 이 境遇 `bzrto -text`로 BPL파일을 만들어 숫자操作을 함으로서 많은 問題가 解決될 수 있었읍니다. `bpltobzr`로 다시 BZR파일을 만들면 다시 한숨을 쉴 수가 있었던 것입니다. `bzredit.el`을 약간 補强하여 `emacs`와 `ghostscript`를 잘 利用할 수 있었던 것이 決定적인 도움이 되었읍니다. GNU fontutils는 아주 有用한 프로그램임을 實感하였읍니다.

한글고딕 以前에 쓰던 KAIST粗合방식의 고딕글꼴을 한데 모아 METAFONT수준에서 完成형으로 粗合하였읍니다.

한글명조, 한글목각 文化體育部에서 公開한 PFB글꼴을 `t1disasm`한 후 이들을 全部 합쳐 다시 256개씩으로 나눕니다. PFB폰트는 font path의 숫자들이 서로 相對적이므로 이를 絶代치로 바꿉니다. 簡單히 `awk`를 利用하여 處理할 수도 있지만 이 境遇 C言語를 利用하였읍니다. 出力은 BPL포맷으로 하여 후에 `bzredit.el`로 修整할 수 있게 합니다. 文化體育部 바탕체(?)는 잘 만들어진 글꼴이지만 낮은 解像度에서의 質이 떨어져 많은 修整을 하였고, 木刻체는 font path가 한방향으로만 돌도록 되어 있어서 힘든 問題로 登場했었읍니다. `ps4mf`를 使用하였다면 이런 問題는 認識도 하지 못했을 것을 뒤늦게 알게 되었읍니다.

포스트스크립트글꼴 한글 고딕체는 文化體育部의 pfb파일을 거의 그대로 使用하였읍니다. 파일의 數를 줄이기 위해 `t1disasm`로 풀어 놓은 후 이 파일들을 再整理하였읍니다. 한 파일에 256개의 글자가 채워지고 Encoding이 HLaTeX에 適合하도록 修整하였읍니다. 그외의 글꼴들은 위에서 METAFONT코드를 만드는 過程에서 생긴 bzr파일들을 利用하여 一旦 ghostscript Type III字形을 만들고 이를 바탕으로 font path를 相對적으로 計算하여, 예를 들어 moveto를 rmoveto로 curveto를 rrcurveto 혹은 hvcurveto/vhcurveto로 修整하였읍니다. awk를 利用하면 簡單히 處理할 수 있읍니다. 이렇게 修整된 파일들을 `t1asm`으로 다시 PostScript Type I字形을 만들었읍니다.

C.4 KSC 5601 심벌

다음은 한글LaTeX文書에서 직접 쓰실 수 있는 심벌들입니다.

〔〕〈〉《》「」『』【】¥♂♀※

‰♨☏☎☜☞㉿㈜

₩

ㄱㄲㄳㄴㄵㄶㄷㄸㄹㄺㄻㄼㄽㄾㄿㅀㅁㅂㅃㅄㅅㅆㅇㅈㅉㅊㅋㅌㅍㅎ

ㅏㅐㅑㅒㅓㅔㅕㅖㅗㅘㅙㅚㅛㅜㅝㅞㅟㅠㅡㅢㅣ

ㅥㅦㅧㅨㅩㅪㅫㅬㅭㅮㅯㅰㅱㅲㅳㅴㅵㅶㅷㅸㅹㅺㅻㅼㅽㅾㅿㆀㆁㆂㆃㆄㆅㆆ

ㆇㆈㆉㆊㆋㆌㆍㆎ

㉠㉡㉢㉣㉤㉥㉦㉧㉨㉩㉪㉫㉬㉭

㉮㉯㉰㉱㉲㉳㉴㉵㉶㉷㉸㉹㉺㉻

ⓐⓑⓒⓓⓔⓕⓖⓗⓘⓙⓚⓛⓜⓞⓝⓟⓠⓡⓢⓣⓤⓥⓦⓧⓨⓩ

①②③④⑤⑥⑦⑧⑨⑩⑪⑫⑬⑭⑮
㈀㈁㈂㈃㈄㈅㈆㈇㈈㈉㈊㈋㈌㈍
㈎㈏㈐㈑㈒㈓㈔㈕㈖㈗㈘㈙㈚㈛
⒜⒝⒞⒟⒠⒡⒢⒣⒤⒥⒦⒧⒨⒪⒩⒫⒬⒭⒮⒯⒰⒱⒲⒳⒴⒵
⑴⑵⑶⑷⑸⑹⑺⑻⑼⑽⑾⑿⒀⒁⒂

附錄 D 글꼴 例文

TeX에서 使用할 수 있는 글꼴의 數가 制限되어 있음으로 모든 한글글꼴을 同時에 使用하기가 不可能합니다. 다음번 버전 web2c-7.0에서는 256 font register의 制限이 緩和되어지는 patch가 適用될 것이라는 소식이 있읍니다. 이 境遇에는 이 글꼴 例文이 더욱 多樣해질 수 있읍니다.

오늘은 어머님 生辰인데, 엎드려 절하고 술·한잔을 드리지 못하니 平生의 恨이다.
오늘은 어머님 生辰인데, 엎드려 절하고 술·한잔을 드리지 못하니 平生의 恨이다.
오늘은 어머님 生辰인데, 엎드려 절하고 술·한잔을 드리지 못하니 平生의 恨이다.
오늘은 어머님 生辰인데, 엎드려 절하고 술·한잔을 드리지 못하니 平生의 恨이다.
오늘은 어머님 生辰인데, 엎드려 절하고 술·한잔을 드리지 못하니 平生의 恨이다.
오늘은 어머님 生辰인데, 엎드려 절하고 술·한잔을 드리지 못하니 平生의 恨이다.

參考 書籍

[최우형92] 최우형. hlatex 가이드. Technical report, KAIST, 1992.

[GMS94] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion.* Addison-Wesley, Reading, MA, USA, 1994.

[Jal] François Jalbert. *Japanese LaTeX for DOS – User's Guide.*

[Knu86a] Donald E. Knuth. *The TeXbook*, volume A of *Computers and Typesetting.* Addison-Wesley, Reading, MA, USA, 1986.

[Knu86b] Donald E. Knuth. *TeX: The Program*, volume B of *Computers and Typesetting.* Addison-Wesley, Reading, MA, USA, 1986.

[Knu86c] Donald E. Knuth. *The METAFONTbook*, volume C of *Computers and Typesetting.* Addison-Wesley, Reading, MA, USA, 1986.

[Knu86d] Donald E. Knuth. *METAFONT: The Program*, volume D of *Computers and Typesetting.* Addison-Wesley, Reading, MA, USA, 1986.

[Kop91] Helmut Kopka. *LaTeX—Eine Einführung.* Addison-Wesley Verlag, Bonn, Germany, third edition, 1991.

[Lam85] Leslie Lamport. *LaTeX—A Document Preparation System—User's Guide and Reference Manual.* Addison-Wesley, Reading, MA, USA, 1985.

[Sch94] Peter Scherber. *Offizin—Schriftenreihe zu TeX, LaTeX, und Metafont (Band 1, Hrsg. von DANTE e.V).* Addison-Wesley, Reading, MA, USA, 1994.

索 引