

# 텍스트 Formatter TEX의 소개

양 광 호\* 이 근 영\*\*

## 요약문

TEX\*은 미국 Stanford 대학의 D. E. Knuth가 개발한 텍스트 formatter로서 전자출판(Electronic Publishing) 분야에서 좋은 반응을 얻고 있다. TEX은 고품질의 조판을 행하며 이식성이 좋고 구입이 용이하다. 본고는 TEX의 개발경위, 출판절차, 기술적 사항, 전망 등에 대한 개괄적인 설명을 위한 것이다.

## I. 텍스트 Formatter와 TEX

텍스트 formatting은 컴퓨터 사이언스의 텍스트처리 분야에 있어서 주요 부분을 차지하고 있다. 더우기 최근 해외에서는 컴퓨터의 인쇄출판에 대한 응용이 상당한 반응을 얻고 있다. 즉 Electronic Publishing (EP)이라고 하는 것으로 응용 시스템을 구현하기 위해 mark up language와 대화형 "What you see

주\* TEX('tech'로 발음)은 American Mathmactical Society의 등록상표이다.

## 목 차

- I. 텍스트 Formatter와 TEX
- II. TEX의 개발 경위
- III. TEX을 이용한 출판 절차
- IV. TEX의 기술적 설명
- V. 전 망

\* 응용 S/W 연구실 연구원

\*\* 응용 S/W 연구실 연구원

is what you get(WYSIWYG)" 시스템의 두 접근방법을 취하고 있으나 어느 방법이든 텍스트 formatting이 그 기본을 구성하고 있다.

UNIX system에서 troff(typesetter run off, 'teeroff'로 발음)이 텍스트 formatter로서 알려져 있으나 WANG사의 CAT 사진조판기(phototypesetter)외에는 사용할 수 없다. 또한 동작환경이 UNIX 시스템에 제한되어 있는 단점이 있다. UNIX system V에서 documenter's work bench라고 하는 troff을 몇개의 preprocessor와 함께 device independent하게 만든 패키지가 있으나 device dependent한 postprocessor가 약하게 평가되고 있다. [9]

UNIX 환경에서 동작하고 또한 호평받는 formatter로서 T<sub>E</sub>X과 Scribe가 있다. 이들은 내부 알고리즘을 제외하고는 user level에서는 동일하다. [5]

특히 T<sub>E</sub>X은 device independent한 fomatter이고 충분한 메모리와 파스칼 컴파일러가 있는 컴퓨터에는 이식하기가 쉬워서 이미 DEC의 여러 시스템(VAX 등), SUN, IBM PC/XT 및 AT, Macintosh 등에 대한 version이 발매되고 있다. T<sub>E</sub>X의 기능으로 말한다면 텍스트, 테이블, 수학기공식 및 차트 등의 formatting이다. UNIX의 troff, tbl, eqn, pic에 비교될 수 있으나 T<sub>E</sub>X에서는 이 모든 기능을 단일 프로그램으로 처리하며 UNIX에서는 tbl, eqn, pic등이 troff의 preprocessor로 별개의 모듈을 형성한다.

T<sub>E</sub>X은 텍스트 출력시의 word hyphenation과 line justification 알고리즘 및 복잡한 수학기공식의 출력에 있어서 특히 우수하다. 또한 public domain에 release 되어 있고 T<sub>E</sub>X user group을 통하면 실비로 source 및 도큐먼트를 제공받을 수 있다. 그리고 중국어, 일본어 등 동양어를 고려하여 개발하여서 한글 및

한자 처리기능의 추가가 용이할 것으로 생각된다.

이하 T<sub>E</sub>X의 개발경위, T<sub>E</sub>X을 이용한 출판절차, T<sub>E</sub>X의 기술적인 면과 전망 등을 소개하고자 한다.

## II. T<sub>E</sub>X의 개발 경위

T<sub>E</sub>X은 미국 Stanford 대학의 D. E. Knuth 교수에 의하여 개발되었는데, T<sub>E</sub>X이 처음으로 소개된 것은 1978년 1월에 Knuth가 미국 수학회(American Mathematical Society)에서 행한 Gibb's lecture에서이다.

이 강연은 mathematical typography라고 하는 제목으로 수학기공식의 작성시에 컴퓨터가 어떻게 이용될 수 있는가 및 문헌의 컴퓨터에 의한 작성시에 수학이 어떻게 이용될 수 있는가를 설명하는 것이었다. 이때 활자체 기술언어(font description language)인 Metafont도 함께 소개되었다.

Byte지와와의 인터뷰에서 Knuth는 T<sub>E</sub>X의 개발동기를 그의 저작활동과 교정체와의 관계, 수준이 낮은 컴퓨터 조판품질 때문이라고 말하고 있다. 즉 이러한 상황에서 그는 디지털 사진조판기에서 인쇄된 책을 보았을 때, 그 책의 고품질을 보고서 아이디어를 얻어 유사한 디지털 machine에 자신의 컴퓨터에 대한 지식을 투입하여 수준이 낮은 컴퓨터 조판품질을 해결하기 위해서 또 자신이 책을 직접 조판하기 위해 T<sub>E</sub>X개발을 착수했다는 것이다.

당시(1977년)에 그는 "The art of computer programming"을 집필하고 있었는데 그 책의 오식에 대하여 상당히 신경을 기울인 것 같다. 1984년 출판된 "T<sub>E</sub>X: The program"의 서문에서는 "만일 이 책에서 오식이 발견될 경우 그것을 자신에게 통고해

주면 상금 \$ 1.28을 주겠다”고 광고하고 있다.

또한 T<sub>E</sub>X을 개발하던 중 출력 활자체가 마음에 들지 않아서 자신이 직접 활자체를 제작하기 위해 Metafont를 개발하기로 했다고 한다. [6]

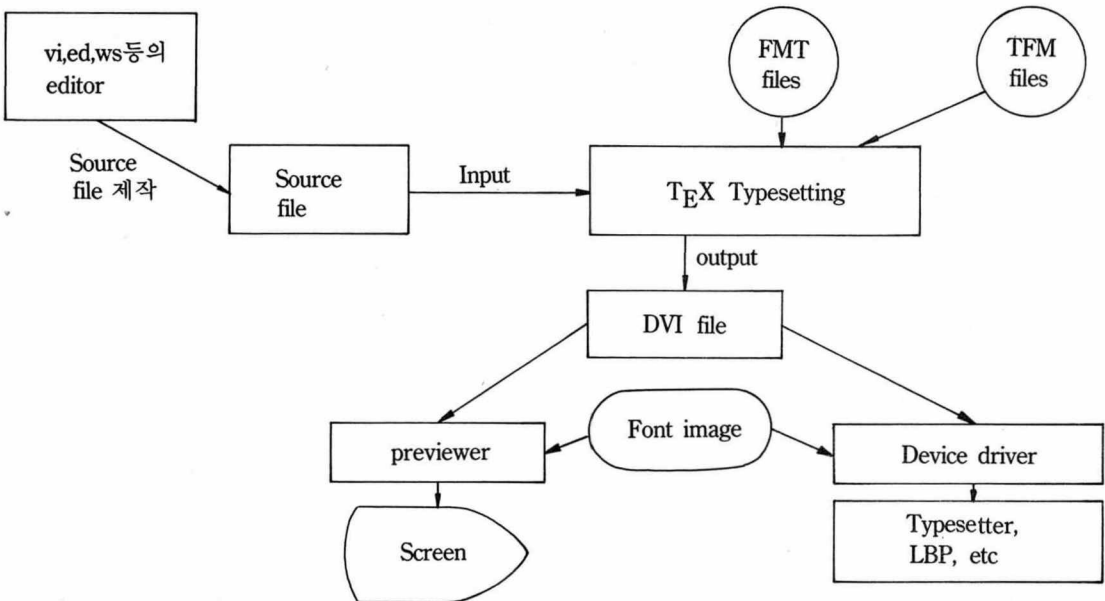
최초의 T<sub>E</sub>X은 1978년초에 SAIL(Stanford Artificial Intelligence Lab.) Language로 구현하였으나 상당부분의 보완을 거쳐 1982년초에 T<sub>E</sub>X 82 Version 0로 upgrade되었다. 현재 배포되고 있는 것은 T<sub>E</sub>X 82 version 2.0이며 “WEB language”로 코딩되어 있다 (WEB은 Knuth가 개발한 structured documentation system인데 이것에 의하여 작성된 프로그램은 Tangle이라는 preprocessor에 의하여 파스칼로 변환된다).

Addison Wesley사에서는 T<sub>E</sub>X 82와 Metafont에 관한 5권의 책을 “Computer and Typesetting”이라는 제목으로 다음과 같이 출간하고 있다.

1. The T<sub>E</sub>X Book.
2. The Source Code for T<sub>E</sub>X
3. The Metafont Book.
4. The Metafont Source Code.
5. Computer Modern Typeface.

### III. T<sub>E</sub>X을 이용한 출판절차

T<sub>E</sub>X을 이용한 출판절차는(그림 1)과 같으며, T<sub>E</sub>X을 이용하여 문서를 출판하려면 먼저 editor를 사용하여 text source 파일을 구성한다. 이때 UNIX의 troff 사용시와 유사하게 출력 format의 구성과 키보드에 없는 외부 문자를 제어하기 위한 제어순서(control sequence)가 삽입된다. 이렇게 구성된 파일의 예를 들면(그림 2)와 같다. 이 파일을 T<sub>E</sub>X에 입력시켜 동작시키면 T<sub>E</sub>X은 자체의 출력알고리즘으



(그림 1) T<sub>E</sub>X 출판 절차

로 이를 처리하여 DVI (Device Independent) 파일을 만든다.

DVI 파일의 format은 원래 David R. Fuchs가 1979년에 디자인하였는데 이 파일에는 byte stream이 들어있고 출력 페이지상의 임의 위치로의 이동, 활자체 및 문자의 선택과 set 등의 일련의 명령을 구성하며 출력될 데이터를 device independent하게 저장하고 있다. 이 DVI 파일은 한 컴퓨터에서 다른 컴퓨터로 변경없이 이동될 수 있다.

Source 파일에 대한 출력결과를 얻으려면 이 DVI 파일을 읽어서 그 제어순서에 따라 페이지를 구성해 내는 DVI driver를 동작시켜야 한다. 이 DVI driver는 각종 device별로 개발되어 있다.

현재 개발되어 있는 driver는 조판기(typesetter), LBP 및 dot matrix 프린터 용의 device driver와

CRT 스크린상으로 출력시키는 previewer의 2종류로 대별할 수 있다. <표 1>은 '85. 3. 당시까지 T<sub>E</sub>X을 이식한 대표적인 컴퓨터 및 지원가능한 device를 나타낸다. <표 2>는 typesetter를 나타낸다.

LBP로도 고품질을 얻을 수 있으나 더 높은 품질을 원한다면 그 DVI 파일을 조판기를 갖춘 컴퓨터로 이송하여 출력할 수 있다. 이때 해상도(resolution) 외에는 이 DVI 파일의 출력결과가 모든 device에 대하여 동일하다. 위 (그림 2)의 source 파일에 대한 DVI 파일을 출력시킨 결과가 (그림 3)과 같다.

T<sub>E</sub>X이 source 파일을 처리할 때는 또한 TFM (T<sub>E</sub>X Font Metric) 파일과 FMT(Formatted) 파일을 사용한다. T<sub>E</sub>X에서 사용되는 모든 활자체는 각각 TFM 파일에 그 활자체의 모든 문자에 대한 높이, 길이, 폭 등의 정보를 갖고 있으며 실제 활자체

```


$$\begin{aligned}
& \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-x^2} dx dy \\
& \int_0^{2\pi} \int_0^{\infty} e^{-r^2} r dr d\theta \\
& \int_0^{2\pi} \left( -\frac{e^{-r^2}}{2} \Big|_{r=0}^{r=\infty} \right) d\theta \\
& = \pi.
\end{aligned}$$


```

(그림 2) T<sub>E</sub>X text source 파일의 예

$$\left( \int_{-\infty}^{\infty} e^{-x^2} dx \right)^2 = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-(x^2+y^2)} dx dy$$

$$= \int_0^{2\pi} \int_0^{\infty} e^{-r^2} r dr d\theta$$

$$= \int_0^{2\pi} \left( -\frac{e^{-r^2}}{2} \Big|_{r=0}^{r=\infty} \right) d\theta$$

$$= \pi.$$

(그림 3) T<sub>E</sub>X의 출력 결과

image는 사용하지 않는다. 실제 활자체 image는 DVI driver가 사용한다. T<sub>E</sub>X에서 기본적으로 사용되는 활자체로는 Knuth가 Metafont로 제작한 것으로 "computer modern fonts"라는 font family가 있다.

T<sub>E</sub>X을 그 기본 제어순서(primitive control sequence)만을 사용하여 일반 사용자가 사용하기에는 상당히 난해하다. 따라서 사용자의 typesetting 스타일에 대한 매크로를 제작하여 그 매크로의 제어순서를

<표 1> T<sub>E</sub>X을 이식한 컴퓨터 및 지원 device

	AMdahl (MTS)	Apollo	CDC Cyber	DEC10	DEC20	DG MV 8000	Ethernet	HP 1000	HP 3000	HP 9000	IBM (MVS)	IBM (VM)	IBM PC	PERQ	Prime	Siemens (BS200)	Sun	VAX (Unix)	VAX (VMS)
C Itoh																			LSU
Canon											GMD			GMD		GMD		Canon	
DEC LNO1																		UWash	LSU
DEC Ltr Ptr 100					OSU <sup>d</sup>														
DEC VT125																			INFN
Diablo								TeXt							OSU <sup>p</sup>				
Epson								JDJW					A-W						
Facit 4542																			INFN
Fla. Data					MR												Textset		
GE3000		COS																	
HP2680							Stnfd		TeXt										
HP2688A										PH; CaTch									
IBM APA 6670												SLAC							
Imagen	UBC	OCLS		Stnfd; Vndbit	SRI; Clmbia		Imagen					SLAC	OCLC				Sun	UCIrv	K&S
NDK 7700												IAM							
QMS Lasgrfx	Textset	ScmLsr; Textset		Textset	Textset	TAMU					Textset	Textset		GMD	TAMU		Textset	Textset	TAMU
Qume Sprint 5								TeXt											
screen Prevue		yale; Textset									GMD			GMD		GMD	Textset		Adld
Symbolics					UWash													UWash	Calme
Talaris				Talrs	Talrs													Talrs	Talrs
Taktronix 4014											UMilan								Adld INFN
Varian					AMS														SciAp
Versates			UKoln	GATch; Vndbit	UWash							Wzmn			Lvmr			UWash	K&S
Xerox Dover					CMU		Stnfd												Stnfd
Xerox 2700II					OSU <sup>d</sup>														
Xerox 9700	UMich; Textset	COS		Udel								Udel					Textset		

\* 표의 교차부분의 항목은 제조회사 및 단체.

〈표 2〉 T<sub>E</sub>X을 이식한 컴퓨터 및 지원 Typesetter

	Amdahl (MTS)	Amdahl (MVS)	Apollo	CDC Cyber	DEC20	HP3000	HP9000	IBM (MVS)	IBM (VM)	Sun	Univac 1100	VAX (Unix)	VAX (VMS)
Agfa P400									IAM				
Alpatype CRS					AMS								
APS-5/Micro-5	Textset	WashSU	COS; Textset		Textset	Textset	HP	Textse	Textset	Textset		Textset	Intergraph; Textset
Compugraphic 9400						USheffield							K&S
Compugraphic 8500		WashSU		RECAU							UWis		
CRTronic													Eire
Harris 7500												SARA	
Linotron 202					Adapt								

\* 표의 교차부분의 항목은 제조회사 및 단체

사용하는데 범용으로 개발되어 release된 패키지를 T<sub>E</sub>X과 함께 구입할 수 있으며 또한 기본적으로 T<sub>E</sub>X에는 Plain T<sub>E</sub>X이라는 매크로패키지가 포함되어 있다.

FMT 파일은 이러한 매크로패키지를 T<sub>E</sub>X이 빠른 속도로 액세스할 수 있도록 변환한 파일이다. 즉 T<sub>E</sub>X의 매크로는 ASCII 파일과 FMT 파일의 두 종류가 있는데 그 크기가 증가할수록 loading하는 데 시간이 많이 요구되므로 loading 시간을 단축시키기 위해서 ASCII 파일을 FMT 파일로 변환하는 것이다.

특히 FMT 파일은 machine dependent한 성질을 가진다. 따라서 범용 패키지도 ASCII 형태로 입수되어 설치시 FMT로 변환하여 설치하며 FMT로의 변환은 T<sub>E</sub>X에 의하여 이루어진다. T<sub>E</sub>X의 매크로에 대하여는 다음의 T<sub>E</sub>X의 특징에서 재차 다룬다.

#### IV. T<sub>E</sub>X의 기술적 설명

##### 1. 제어순서

제어순서(control sequence)는 mack up language로서 “escape char+letter+”의 형태를 갖는 control word와 “escape char+single non-letter”의 형태를 갖는 control symbol의 2 종류가 있다. escape char는 ‘\’이나 사용자가 임의의 char로 재정의 할 수 있다.

예) Control word : \input MS; MS. tex file을 read  
Control symbol:\'; 뒤 따르는 문자에 accent를 얹음

T<sub>E</sub>X은 약 900개의 제어순서를 사용하며 이중 300개 정도는 T<sub>E</sub>X의 기본(primitive) 제어순서이고 약 600개는 기본 매크로패키지인 Plain T<sub>E</sub>X의 제어순서이다. 대부분의 제어순서는 사용자가 재정의 하여 사용할 수 있다.

##### 2. 활자체

현재 256개 문자의 활자체들을 조작할 수 있으며 제어순서로써 그중의 특정 활자체를 선택한다.

예) \rm ; Roman font                    \sl ;slanted Roman  
      \it ; Italic                         \bf ; boldface

Plain T<sub>E</sub>X에서는 각각의 size가 10 point인 16개 활자체들을 사용하는데 이들은 Knuth가 Metafont를 사용하여 제작한 것으로 “computer modern font family”라고 한다. 이 외에 다른 font library의 활자체를 사용할 수 있고 출력 장비에 따라서 특정 활자체의 크기를 축소 혹은 확대하여 사용할 수 있다.

### 3. Grouping

프로그래밍 언어의 scope rule에 유사한 개념으로서 제어순서를 사용하는 것이다.

예) `\centerline{This information should be { \it centered }`

여기서 {, }는 grouping 문자이고 활자체를 이탤릭체로 바꾸는 `\it` 제어 순서는 group내에서만 사용되어 centered만 이탤릭체로 바뀐다. T<sub>E</sub>X의 한 내부 파라미터가 어떤 group내에서 변경되었다면 그 group이 끝날 때 전의 파라미터 값으로 복귀된다.

따라서 파라미터들을 global한 것과 local한 것으로 나누어 사용할 수 있다. 또한 이 grouping은 어떤 제어순서에 대한 argument로도 사용되는데 위에서는 `\centerline`의 argument로 사용되었다.

### 4. Dimensions

T<sub>E</sub>X은 출력 raster의 개개 pixel을 정교하게 다루기 때문에 내부적으로 다양한 단위계를 사용하며 아래와 같은 것들이 있다.

pt(point), pc(pica, 1 pc = 12 pt), in(inch, 1 in = 72.27 pt), bp(big point, 72 bp = 1 in), cm(centimeter, 2.54 cm = 1 in), mm(millimeter, 10 mm = 1 cm),

dd(didot point, 1157 dd = 1238 pt), cc(cicero, 1 cc = 12 dd), sp(scaled point, 65536 sp = 1 pt) 등.

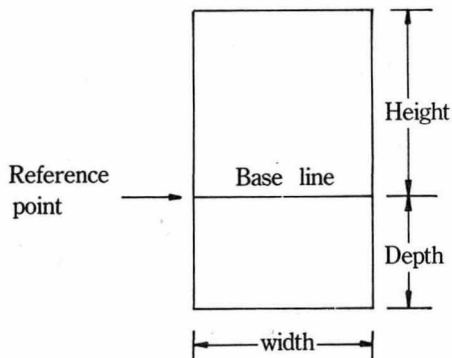
T<sub>E</sub>X 출력은 metric 시스템에 근거하여 위 모든 단위계는 “sp”로 변환되어 조작되기 때문에 rounding error로 인한 pixel 편위등이 거의 없다. 또 이러한 절대단위계 외에 다음과 같은 상대단위(현재 사용중인 활자체의 크기에 근거한)계도 사용한다.

em(현재 활자체에서의 quad 폭), en(현재 활자체에서의 × 높이) 등.

### 5. Box 와 Glue

Box와 glue는 페이지 구성을 위한 것으로 T<sub>E</sub>X이 출력하는 페이지상의 모든 object를 glue에 의하여 연결된 2차원 box들로 modeling한다.

#### 가. Box



(그림 4) Box

가장 간단한 종류의 box는 임의 활자체의 한 문자이며 활자체 제작자는 이 문자의 높이, 폭, 깊이 등을 규정하고 이들은 T<sub>E</sub>X이 box들을 paste

할 때 혹은 페이지상의 문자들에 대한 referenece 위치등을 결정할 때 사용된다.

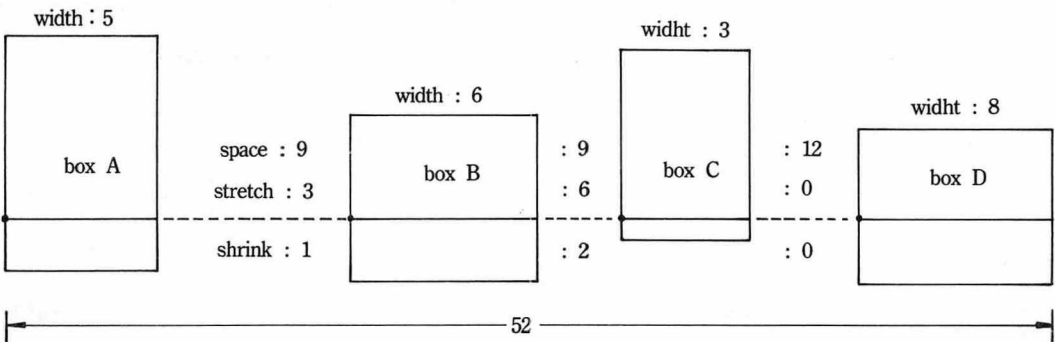
T<sub>E</sub>X은 이 box들을 수평 혹은 수직으로 paste하는데 수평 list(수평 box)를 만들때는 box들을 reference point들이 같은 수평선상에 오도록 배열하고 수직 list(수직 box)를 만들때는 reference point들이 같은 수직선상에 오도록 배열한다. 수평 list의 개개 box들의 reference point 들은 상하 좌우로 이동될 수 있는데 "T<sub>E</sub>X"는 E를 box의 좌 하단으로 이동하여 구성된 수평 box이다. 수직 list상의 개개 box들의 reference point들도 상하 좌우로 이동될 수 있다.

페이지는 수평 box의 수직 list로부터 구성된 가장 큰 box이고 수평 box는 문자 box들의 수평 list로

구성된 box이다. 복잡한 수학기공식, 테이블을 포함하는 상황에서는 임의 level까지도 box내의 box 등으로 중첩시킬 수 있다.

나. Glue

Box들을 paste할 때 사용하는 아교인데 그 기능으로 볼때 스프링이란 말이 더 적절하다. 즉 T<sub>E</sub>X은 box들을 paste하여 수평, 수직 list를 만들때 그 box들 간에 glue를 놓는다. Glue는 space(정상적인 폭), stretch(확장할 수 있는 최대폭), shrink(수축할 수 있는 최대폭)의 3개 attribute를 갖는다.



(그림 5) Glue

위 그림에서 natural width(가장 보기좋은 단어 배열의 폭)은  $5+9+6+9+3+12+8=52$  이다. 만일 이 수평 list를 58의 폭으로 하라고 지시하면 다음과 같이 glue의 stretch를 이용하여 각 glue를 확장시킨다. 먼저 확장폭과의 차 ( $58-52=6$ )와 각 glue의 stretch의 합( $3+6+0=9$ )을 구하여 glue에 동일 stretch 비율을 할당하여 glue의 폭을 확장한다.

$$5+(9+(6/9 \times 3))+6+(9+(6/9 \times 6))+3+(12+0)+8=58$$

또한 만일 이 수평 list를 51로 하라고 지시하면 동일한 방법을 shrink에 적용하여 glue의 폭을 축소한다.

무한 stretch의 glue를 수평 list의 좌편에 놓으면 그 list를 right justify하는 효과를 갖는다. 또한 이



무한 stretch의 glue 2개를 수평 list의 좌우편에 놓으면 그 list를 centering하는 효과를 얻는다.

## 6. Hyphenation과 Justification

Line breaking에 대한 전통적 방법은 line의 우측 margine에 올때마다 최적의 line justification을 하는 것인데 일단 line이 justify되면 재차 고려하지 않는다. 따라서 paragraph의 각 line별로 단어간 space가 상이한 것이 보통이다. 그러나 T<sub>E</sub>X은 paragraph 단위로 line을 justify하므로 한 paragraph내에서는 단어간의 space가 일정하다.

또한 line단위의 justification보다 hyphenate되는 경우가 감소된다.

한 단어의 hyphenation은 line justification 시 T<sub>E</sub>X 자체의 알고리즘에 의하여 적절히 수행되지만 사용자가 직접 단어의 hyphenation point를 지시할 수 있고 line의 break point를 지시할 수 있다.

## 7. 수학 공식

수학공식 등은 math mode라는 독립된 동작모드에서 처리되는데 이 모드에서 각 텍스트는 수학공식 제작관습으로 해석된다. 즉 각 letter는 이탤릭체로, 디지털과 punctuation은 roman 디지털과 Punctuation, hyphen은 -sign으로 해석된다. 실제 이 math mode는 \$로 시작해서 \$로 끝난다.

Math mode에는 in line mode와 display mode의 두가지 종류가 있는데 전자에서는 공식이 수평 list에 삽입 되고 후자에서는 별개의 수평 list로 처리된다. 다음 (그림 6)은 display mode로 수학공식을 출력한 예이다.

```


$$S_p_1(n) = \lim_{m \rightarrow \infty} \sum_{\nu=0}^{\infty} (1 - \cos^{2m}(\nu^n \pi/n))$$


```

\$\$

(그림 6) Display Mathmode에서의 출력예

## 8. Macros

보통 매크로는 기존 language를 단순화하고 확장하는 강력한 tool 인데 T<sub>E</sub>X에서도 기존 제어순서를 이용한 새로운 제어순서의 제작과 재정의에 사용된다. 특히 T<sub>E</sub>X의 mark up language는 사용자의 사용편리성 보다는 출력되는 광범한 object의 표현에 중점을 두어서 대단히 복잡하다.

따라서 매번 T<sub>E</sub>X의 기본 제어순서만을 사용하는 것은 번거로운 일이기 때문에 보통 사용자가 구상하는 typesetting 스타일에 따라서 매크로를 제작하여 사용하는 것이 편리하다. 범용으로 개발된 매크로를 T<sub>E</sub>X의 구입시에 함께 구입할 수도 있다.

또한 T<sub>E</sub>X은 기본적으로 Plain T<sub>E</sub>X이라는 매크로 패키지를 default로 사용한다. MS DOS에서 동작되는 PC T<sub>E</sub>X에는 Vanilla. tex이라는 패키지가 있다. T<sub>E</sub>X과 함께 public domain에서 구입될 수 있는 것으로 AMS-T<sub>E</sub>X, L<sup>A</sup>T<sub>E</sub>X이 있다.

특히 L<sup>A</sup>T<sub>E</sub>X은 Brian Reid의 Scribe 시스템으로부터 개념을 얻어 개발되었고 document preparation system이라고 한다. 이것이 제공하는 스타일로는 article, report, book style의 표준 스타일이 있고, 또한 사용자의 독창적인 스타일 제작환경을 제공한

다. T<sub>E</sub>X을 이용하면 차트 등의 그림을 그릴 수도 있다.

### 9. 테이블

T<sub>E</sub>X으로 테이블을 제작하는 데는 tabbing과 template의 두 방법이 있는데 기본적으로 이들은 line alignment에 관한 문제이다. Tabbing은 보통 tab 기능과 유사하며 한 line을 수개의 컬럼으로 분할한 후 그 컬럼에 항목을 할당하는 것이다. Template는 보다 정교한 방법으로 다음과 같이 \haling 제어순서에 의해 각 컬럼마다 독특한 환경을 규정한다.

예)  
`\haling \indent # \hfil & \quad # \hfil\cr`

```
Horizontal lists & Chapter 14\cr
Vertical lists & Chapter 15\cr
Math lists & Chapter 17\cr
```

첫번째 line은 두개의 template를 규정하고 있는데 “\indent # \hfil”과 “\quad # \hfil”이다. 각각

```
\vbox{\tabskip=0pt \cffiinterlineskip
\def\tablerule{\noalign{\hrule}}
\halign to\dimexpr\textwidth-2\tabskip{\strut#\tablerule#\tableskip=1em plus2em&
\hfil#\tablerule#\tableskip=1em plus2em&
\hfil#\tablerule#\tableskip=0pt\cr\tablerule
&\multispan5\hfil AT&T Common Stock\hfil&\cr\tablerule
&\omit\hidewidth Year\hidewidth&
\omit\hidewidth Price\hidewidth&
\omit\hidewidth Dividend\hidewidth&\cr\tablerule
&1971&41--54&$2.60&\cr\tablerule
& 2&41--54&2.70&\cr\tablerule
& 3&46--55&2.87&\cr\tablerule
& 4&40--53&3.24&\cr\tablerule
& 5&45--52&3.40&\cr\tablerule
& 6&51--59&.95\rlap{*}\cr\tablerule \noalign{\smallskip}
&\multispan7* (first quarter only)\hfil\cr}}
```

(그림 7) 테이블 출력의 예

#가 있는데 컬럼에 들어가는 항목은 #에 대응되고 이 항목은 “\indent”와 “\hfil”, “\quad”와 “\hfil”의 제어를 받아 컬럼에 위치되고 그 출력 결과는 다음과 같다.

Horizontal lists	Chapter 14
Vertical lists	Chapter 15
Math lists	Chapter 17

테이블 주위의 line은 box의 일종인 수직 ruler와 수평 ruler를 사용하여 그린다.

(그림 7)은 보다 구체적인 테이블 출력의 예로 source file과 그 출력 결과를 나타낸다.

### V. 전 망

T<sub>E</sub>X은 그 자체만으로도 고품질의 typesetting을 행할 수 있고 이식성이 양호하며 또한 public domain으로 release되어 있어 구입이 용이하므로 향후 그 이용이 확대될 전망이다.

AT&T Common Stock		
Year	Price	Dividend
1971	41-54	\$2.60
2	41-54	2.70
3	46-55	2.87
4	40-53	3.24
5	45-52	3.40
6	51-59	.95*

더우기 PC에서도 동작될 수 있어서 워드 프로세서의 품질에 만족하지 않는 사무실에서 탁상 출판시스템으로 사용될 수 있다.

이러한 T<sub>E</sub>X의 호응으로 미국에서는 T<sub>E</sub>X User Group(TUG)이 이미 조직되어 T<sub>E</sub>X의 광범한 이용에 기여하고 있다. TUG은 newsletter로 TUG boat라는 비정기 간행물을 출간하고 있는데 이로부터 최근의 T<sub>E</sub>X에 대한 정보를 구할 수 있다. 그리고 실비\*로 source 및 document를 배포\*\*하고 있다. 일본에서도 최근 T<sub>E</sub>X user group의 조직 공고가 Bit지(86.4.)에 소개되고 있다.

국내에서는 그다지 알려져 있지 않은것 같으나 T<sub>E</sub>X의 기능은 국내에서도 이용할 가치가 충분히 있는것으로 생각한다. 이에 따라 ETRI 응용 S/W 연구실에서는 저가격 전자 출판시스템(EP: Electronic publishing system)의 개발을 위하여 T<sub>E</sub>X에 대한 분석외에도 한글 한자 처리기능의 추가, 쉽게 그림을 그릴 수 있는 그래픽 언어에 대한 연구를 추진하고 있다.

주 \* Tape cost : \$82, media cost : \$10, manual cost : \$82

주 \*\* VAX, SUN, PYRAMID용의 UNIX T<sub>E</sub>X 배포 tape에는 다음과 같은 패키지가 들어있다.

- font image 및 TFM file
- T<sub>E</sub>X82
- T<sub>E</sub>XWARE(WEB 등)
- DVI driver(imagen 10/240 5/480 8/300, DEC LNO1, LNO1s, versatec/varian, symbolics LGP 1의 DVI driver, BBN Bitgraph, SUN-II의 previewer)
- macro(LA<sub>T</sub><sub>E</sub>X, AMST<sub>E</sub>X)
- Metafont
- Manual

## 참 고 문 헌

1. Donald E. Knuth, "The T<sub>E</sub>X book", Addison wesley, 1984.
2. Michael Spivak, "PC T<sub>E</sub>X manual", Personal T<sub>E</sub>X, 1985.
3. Leslie Lamport, "LA<sub>T</sub><sub>E</sub>X, A Document Preparation System", Addison Wesley, 1986.
4. Richard Ilson, "Recent Research in Text Processing", IEEE TRAN on PROF. COMM. Dec. 1980.
5. Richard Furuta, "Document Formatting Systems : Survey, Concepts and Issues", ACM Computing Survey, Sep. 1980.
6. 和田英一 : "エディタとテキスト処理" 연재중 T<sub>E</sub>X 편, BIT, Jul. 1983.
7. G. Michael Vose, "Computer Science Considerations", Byte, Feb. 1986.
8. Pierre A Mackay, "Typesetting Problem Scripts", Byte, Feb. 1986.
9. Michael Sweeney, "Electronic Publishing : Technologies, Markets, Applications", Unix/World, Apr. 1986.
10. David fuchs, "Output Devices and Computer", TUGboat, Mar. 1985.