

제 I 부

개관

제 1 장

TEX이란 무엇인가?

1.1 TEX의 탄생

TEX은 수학자 겸 전산과학자로 유명한 Stanford 대학의 Donald E. Knuth 교수에 의해서 만들어졌다. Knuth 교수의 역작인 *The Art of Computer Programming*의 부산물로 TEX이 나오게 되었다. TEX의 탄생에 관한 이야기를 Knuth 교수의 말로 들어보자.

왜 나는 1977년에 TEX을 만들기 시작했는가? 실제로 전체적인 일은 *The Art of Computer Programming* 책과 연관되어 이미 오래 전에 시작되었다. 그 당시 나는 Volume 2의 2판을 준비했었는데, 1판을 출판한 이후로 인쇄기술이 급격하게 변하여 출판사로부터 형편없는 교정판을 받게 되었다. 교정판은 납으로 식자(typesetting)하는 대신에 사진식자(phototypesetting)로 되어 있고 사람 손 대신에 컴퓨터의 도움을 받아 제작되었다. 그 결과는 수식에서 간격이 엉망이었고 폰트는 원본에 비해서 형편없었다. 나는 이것에 크게 실망하여 어떻게 해야될지를 몰랐었다. Addison-Wesley 사는 다시 납으로 식자해주겠다고 제안했으나 나는 그러한 구식 방법은 머지않아 곧 사라질 것을 알고 있었고 내가 받은 교정판처럼 보이는 책을 쓰고 싶지는 않았다.

그리고... 우리는 남가주(southern california)에서 새로운 기계로 만든 Pat Winston의 Artificial Intelligence의 교정판을 받아보았는데, 모든 것이 고해상도 래스터에 근거를 하고... 디지털 글자는 나의 교정본의 글자보다 훨씬 보기 좋았고... Winston의 교정본을 보고 나서 일주일 후에 나는 모든 것을 취소하고 디지털 인쇄에 전념하기로 결심했다.

1977년에 이렇게 시작된 이후로 TeX 연구 프로젝트는 두 가지 목표를 갖추고 추진되었다. 첫 번째 목표는 질(quality)이었다. 우리는 단순히 좋은 것 이상의 최상의 문서를 만들기를 원했다.... 1977년도 당시만 해도 매우 매력적인 문서를 만들 수 있는 시스템이 몇 개가 있었다. 나의 목표는 거기에서 더 나아가 인쇄된 문서가 얻을 수 있는 최고의 질을 얻는 것이었다.

약 2년간의 작업 끝에, 텍스트 포매팅에 관한 한 이 정도 수준의 질을 얻는 것은 그리 어렵지 않다는 것이 밝혀졌다. 예를 들어, 우리는 TeX으로 Times 잡지를 원래의 Times 잡지보다 보기 좋게 만들 수 있다는 것을 증명했다. 그러나 글자를 만들어내는 것이 우리가 예상한 것보다는 훨씬 어렵다는 것을 알게 되었다. 내가 좋아하기 시작한 글자형을 만드는 데 7년이 걸리게 되었다.

두 번째 목표는 보존성(archival)이었다. 가능한 한 인쇄기술의 변화에 크게 의존하지 않는 독립적인 시스템을 만드는 데 목표가 있었다. 다음 세대의 인쇄 장비가 도래해도 새로이 모든 문제를 해결하는 대신에 이전에 이룩해 놓은 질을 유지하고 싶었다. 나는 100년이 지나도 계속 사용 가능한 무엇인가를 디자인하고자 했다. 다른 말로 하면, 만일 책의 사양이 현재의 상태로 저장된다해도 2086년에 우리의 후손들이 만들어도 지금 상태의 것과 똑같은 책을 만들 수 있게 하는 것이 나의 목표였다....

1.2 TeX의 동작원리

일반적으로 텍스트 처리기(processor)는 크게 두 가지로 나눌 수 있다.

WYSIWYG 시스템 인쇄될 문서의 모양이 화면에 그대로 나타나고 입력을 하면 그 변화가 즉시 화면에 반영되어 나타난다.

마크업(markup) 시스템 텍스트와 포매팅 명령어를 섞어 입력하는데 그 효과를 바로 보지는 못한다. 프로그램을 수행하여 입력된 내용을 처리하면 결과 이미지를 화면이나 프린터로 볼 수 있다. 컴퓨터 용어로 표현하면 마크업 시스템은 입력한 소스 화일을 컴파일해야 한다.

WYSIWYG 시스템¹은 바로 피드백이 되므로 분명히 장점이 있으나, 아주 정확하지는 않다. 예를 들어, WYSIWYG 시스템으로 문서를 만들어 300 dpi(dots per inch) 프린터로 인쇄하여 보면 적당히 좋아 보이나 고해상도로 식자를 하면 더이상 좋게만 보이지는 않는다. 인간의 눈은 매우 민감하다. 마치 오케스트라 연주시 누군가가 작은 실수를 하면 정확히 무엇이 잘못되었는지 지적을 할 수

¹What you see is what you get. 화면과 프린터가 같은 해상도, 서체 그리고 페이지 언어를 사용하면 100% WYSIWYG이다.

는 없으나 귀에 거슬리는 것과 마찬가지로 잘못 인쇄된 문서는 눈에 거슬리게 된다. 실제로 아름답고, 알아보기 쉽고, 그리고 편히 읽을 수 있는 문서를 만들기 위해서는 $\frac{1}{1000}$ 인치내에서 정확히 문서의 모든 요소들이 위치해야 한다. 이러한 유형의 작업에는 즉시 피드백되는 것이 그리 장점이 되지 않는다. 간격, 정렬 등의 세밀한 조정은 저해상도 화면을 통해서 하기에는 너무 힘이 들고 손으로 일일이 작업하는 것이 불가능한 경우도 있다.

이러한 이유로 인해 전문적으로 식자를 하는 곳에서는 마크업 시스템이 환영을 받는다. 마크업 시스템을 사용하면 정밀한 작업을 명령어로 처리하고 자동화시킬 수 있다. 그러나 이러한 접근 방식은 결과를 바로 볼 수가 없기 때문에 초보자에게는 매력적으로 보이지 않게 된다. 따라서 WYSIWYG 입력기가 포함된 마크업 시스템을 사용하면 양쪽의 장점을 모두 얻을 수 있을 것이다.

\TeX 은 마크업 시스템에 속한다. 즉, 사용자는 문서의 내용과 필요한 명령어가 들어간 소스 파일을 \TeX 엔진을 이용하여 컴파일하여 인쇄를 하거나 화면을 통해서 결과를 보게 된다. 컴파일된 결과로 나온 파일을 DVI(DeVice Independent) 파일이라고 하는데 화면이나 프린터의 해상도에 관계없이 드라이버만 있으면 결과를 볼 수 있다.

1.3 \TeX 의 장점과 단점

무엇보다도 \TeX 의 장점은 높은 질의 문서²를 만들어 낸다. \TeX 이 다른 문서 시스템에 비해 텍스트 처리에서 특히 뛰어난 몇 가지 예를 보면 다음과 같다.

- 합자(ligature)³를 자동으로 처리한다.
- 커닝(kerning)⁴을 자동으로 처리한다.
- 정판(Justification)을 하는 복잡한 체계를 갖고 있어 필요하면 하이프네이션⁵을 한다. 하이프네이션 규칙은 사용자가 원한다면 추가나 변경이 용이하여 여러 언어에서 사용 가능하다.
- 수학 공식 내의 여러 요소들 사이의 간격은 최고의 수식 식자공들이 사용하는 전통적인 규칙에 따라 \TeX 에 의해서 결정되기 때문에 수식이 아름답게 처리된다.

잘 씌어진 \TeX 문서는 매크로에 의해서 포맷되어 각 요소들이 어떻게 식자되는지 지정된다. 매크로는 기본적인 포맷팅 명령어들로 만든 짧은 프로그램

²특히 수식의 경우

³영어의 ff, ff 등과 같이 여러 글자가 합쳐져 하나의 글자로 처리된다. 전통적으로 합자는 시각적으로 보기 좋은 결과를 얻기 위해 필요하다.

⁴글자의 모양이 서로 가능하면 가깝게 위치하게 하여 글자 사이의 간격이 균등하게 보이게 한다.

⁵단어의 중간에 하이픈(-)을 삽입하여 줄 사이에서 끊어지게 한다.

램으로서 일일이 세세한 명령어를 사용하는 번거로움을 덜어준다. 예를 들어, `\footnote` 매크로는 텍스트를 페이지의 바닥으로 이동시키거나 줄 위에 작은 번호를 매기는 걱정을 하지 않아도 자동으로 주석 처리가 되게 한다. 매크로들은 `\indent` 혹은 `\par`와 같은 기본 명령어(primitive)로 씌어져 \TeX 의 기본적인 어휘가 된다.

매크로 덕분에 \TeX 은 대단히 다양한 용도의 기능을 갖게 되었다. 예를 들어, 문서의 외관을 고치고자 하면 텍스트를 건드리지도 않고 단순히 특정 매크로의 정의만 고쳐도 충분하다. 온라인 문서의 코딩에 관한 표준을 정의하는 작업이 현재 진행중인데, \TeX 은 고급(high-level) 문서 표현 언어(document description language)로 사용될 수 있기 때문에 쉽게 이 표준에 들어갈 것이다.

\TeX 은 이동이 가능(portable)하다. 텍스트와 매크로를 포함한 \TeX 으로 씌어진 문서는 인쇄 가능한 ASCII 세트⁶로 코드될 수 있다. 따라서 모뎀이나 LAN 같은 통신수단을 이용하여 아무리 복잡한 수식이나 레이아웃(layout)이 포함된 문서라도 세계 어느 곳에서도 서로 주고 받을 수 있다. 예를 들면, 미국 수학회(American Mathematical Society)에서 보유하고 있는 데이터베이스는 \TeX 에 근거를 하여 세상에 출판된 모든 수학 논문의 요약(abstract)들을 포함하고 있다. 따라서 원하는 논문의 요약을 데이터베이스에서 \TeX 으로 포맷팅된 형태로 얻을 수 있다.

\TeX 은 특정 컴퓨터의 특성에 의존하지도 않고 기계와는 독립적인 방법으로 계산을 하기 때문에 여러 컴퓨터들 사이에서 이동 가능하다. 즉, Macintosh, PC, UNIX 워크스테이션, IBM 대형컴퓨터, 심지어는 Cray 등에서 각각 만든 \TeX 문서는 모두 모양이 같게 된다. 따라서 여러 컴퓨터 시스템 사이에서 정보 교환의 수단으로 \TeX 은 매우 적합한 도구가 된다.

\TeX 은 문서의 이미지를 직접 만들어내는 대신에 단지 페이지의 내용을 묘사한다. 문서를 인쇄하기 위해서는 \TeX 의 출력⁶을 받아 이를 드라이버 프로그램으로 원하는 프린터나 화면에 출력해야 한다. 이러한 이유로 인해 \TeX 은 현재의 프린터 기술과도 독립적이 된다. 만약 프린터 기술이 변한다면 문서의 내용을 바꿀 필요 없이 단지 필요한 드라이버 프로그램만 만들면 된다.

\TeX 은 단순한 텍스트 프로세서 이상이다! 프로그램 언어이다! 사용자가 원하는 다양한 기능을 \TeX 의 변수를 변경하거나 새로운 매크로를 정의함으로써 추가할 수 있다. 경험이 많아지면, 새로운 스타일을 정의하고 특정 목적을 위한 복잡한 매크로를 만들 수도 있고 이미 이전에 남들이 만들어 놓은 스타일이나 매크로를 활용할 수도 있다. 물론 이러한 일들이 쉽지만은 않지만 TUG(\TeX User Group)을 통해서 많은 정보를 얻을 수 있으며 네트워크를 통해 public domain에 나와 있는 정보나 데이터를 활용하면 많은 도움이 될 것이다.

\TeX 은 또한 확장 가능하다! 언급했듯이 Knuth는 \TeX 을 만들 당시에 미래를 바라보았었다. 이러한 이유로 그는 \TeX 을 양파와 같은 층구조로 만들었다.

⁶DVI 화일

중앙에는 $\text{T}_{\text{E}}\text{X}$ 의 300개의 기본명령어(primitive)로 되어 있는데 이름 그대로 원시적이어서 실제로는 기본 명령어를 바로 사용하는 경우는 적다. 그 다음 층은 이보다 고급 명령어들 혹은 매크로들로서 $\text{T}_{\text{E}}\text{X}$ 의 포맷 화일에 정의되어 있다. 가장 보통 사용되는 포맷 화일은 plain.tex 으로 약 600개⁷의 명령어들이 정의되어 있다. 기본 명령어와 확장 명령어의 조합을 Plain $\text{T}_{\text{E}}\text{X}$ 이라고 일반적으로 불리운다. 이 위에 확장을 한 층을 패키지라고 하는데 대표적으로는 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, $\text{A}^{\text{M}}\text{S}\text{T}_{\text{E}}\text{X}$ 등이 있다. 이 중에 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 이 가장 알려져 있으며 가장 완벽하다. 따라서 대부분의 $\text{T}_{\text{E}}\text{X}$ 사용자는 실제로는 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ 을 쓰고 있다.

$\text{T}_{\text{E}}\text{X}$ 은 매우 잘 디버그(debug)되어 있다. 물론, 다른 프로그램들과 마찬가지로 결코 버그(bug)가 없지는 않다. 그러나 새로운 버그가 보고될 때마다 Knuth가 상을 준 덕(?)에 전세계적으로 버그 사냥꾼들이 몰려들어 극히 유별하거나 즉시 발생할 사유가 없는 버그들만 남아 있다. 만일 지금이라도 버그를 발견하여 보고하면 상과 함께 TUGboat에 정기적으로 출판되는 공식적인 $\text{T}_{\text{E}}\text{X}$ 버그 목록에 실리게 된다.

$\text{T}_{\text{E}}\text{X}$ 은 public domain에 나와 있다. Knuth는 $\text{T}_{\text{E}}\text{X}$ 을 세상에 공짜로 제공했다. $\text{T}_{\text{E}}\text{X}$ 의 소스 프로그램은 쉽게 복사할 수 있다. 만일 $\text{T}_{\text{E}}\text{X}$ 을 산다면, 이는 특정 기계 및 환경에 설치된 드라이버, 보기 프로그램(viewer), 텍스트 에디터 등에 대해서 지불하게 되는 것이다.

완벽한 것은 없다. $\text{T}_{\text{E}}\text{X}$ 의 단점의 대부분은 확장 혹은 보조 프로그램을 통해서 극복될 수 있다. 단점 중에는 특정 설치환경에 기인하지 $\text{T}_{\text{E}}\text{X}$ 의 본질적인 단점이 아닌 것도 있고, $\text{T}_{\text{E}}\text{X}$ 의 보급 초기에 제기되어 이미 여러 방법으로 해결된 것⁸도 있다.

$\text{T}_{\text{E}}\text{X}$ 프로그래밍은 난해하여 습득하는 데 시간이 걸린다. 물론 주어진 명령어나 매크로를 사용하는 일반 사용자에게는 이것이 문제가 되지는 않는다. 그러나 새로운 매크로나 스타일을 디자인하는 것은 초보자에게 적합한 과제가 아니다.

$\text{T}_{\text{E}}\text{X}$ 은 제한된 메모리를 갖고 있다. 초기에 설치된 $\text{T}_{\text{E}}\text{X}$ 들은 메모리의 한계 때문에 사용할 수 있는 매크로나 서체의 숫자가 제한되었으나 컴퓨터 OS (Operating System)의 발전으로 이를 극복한 환경⁹들이 계속 나오고 있다.

$\text{T}_{\text{E}}\text{X}$ 은 대화형식(interactive)이 아니다. 이 점이 $\text{T}_{\text{E}}\text{X}$ 의 최대 단점으로 일반적으로 지적되고 있다. $\text{T}_{\text{E}}\text{X}$ 은 본질적으로 마크업 시스템이기 때문에 이 문제를 완전히 해결할 수는 없지만 사용하기 쉬운 텍스트 에디터, 보기 프로그램, 프린터 드라이버 등이 잘 합해져 사용자가 마크업 시스템에서 얻기 힘든 피드백을 쉽고 빠르게 받을 수 있다면 극복될 수 있다.

⁷이 숫자에 놀라지는 마라. 왜냐하면 이 중에 알아야 할 것은 100개도 안 되며 그것도 암기하기 쉬운 이름으로 정의되어 있다.

⁸서체 추가, 그래픽 처리, 다양한 레이아웃

⁹X-Window, PC Windows, Mac OS

1.4 누가 TeX을 사용하는가?

Knuth가 TeX을 만든 첫 번째 이유는 자기 자신이 필요했기 때문이다. 즉, Knuth와 같이 학문을 하거나 연구를 하는 사람들이 출판물을 하는 경우 TeX을 사용하는 것이 최선의 선택이 된다. Knuth는 자신이 필요한 것이 무엇인지를 알고 있었고 그것을 TeX이 충실하게 실현하도록 만들었다. 따라서 TeX은 전 세계적으로 학계나 연구기관에서 출판 혹은 정보교환용으로 표준으로 사용되고 있으며, 문서 데이터베이스의 구축이나 온라인 문서의 기록(archive)에 활용되고 있다.

분야에 관계없이, 글을 쓰는 사람은 TeX의 사용을 한번 고려해보아야 한다. 글을 쓰는 사람은 글의 모양¹⁰보다는 내용에 신경을 써야 한다. 컴퓨터의 발달과 이에 따른 보급의 확장으로 대부분의 글을 쓰는 사람들은 워드프로세서를 사용하여 글을 입력한다. 이렇게 입력된 글은 직접 프린터로 인쇄하거나 출판사에 의뢰하여 식자가 되어 출판이 된다. 이때 글의 모양은 글을 쓰는 사람에게 의해서 결정되는 경우가 많고 심지어는 글의 모양을 스스로 책임져야 하는 경우도 있다. 글을 쓰는 사람들은 실제로 전문 식자공이 아니기 때문에 일기 쉽고 아름다운 문서를 만드는 데 어려움이 있으며 이 때문에 많은 노력을 내용과는 상관없는 데 기울이게 된다. 만일 원하는 모양이 복잡한 레이아웃도 아니고 반복하여 계속 사용하는 경우에 TeX을 사용하면 글의 내용만 신경을 쓰면 되니까 훨씬 더 좋은 글을 쓰게 된다.

책의 출판은 다른 문서의 출판과는 다른 점들이 있다.

1. 내용이 길기 때문에 하나의 화일로 입력이 불가능한 경우가 많으며 여러 사람이 입력을 해야 되는 경우도 있어 책의 전반에 걸쳐 모양의 일관성(consistency)을 유지하기 어렵다.
2. 새로운 내용의 추가나 변경으로 전체적인 변화를 요구하는 경우¹¹가 많다.
3. 페이지 내용에 반복되는 디자인 요소들이 많다.

위에 열거된 문제점들은 TeX의 사용을 통해 해결될 수 있기 때문에 책과 같이 긴 문서의 출판에는 TeX의 사용이 적절하다.

TeX의 장점 중의 하나는 수식처리가 쉽고 결과가 아름답게 나온다는 점이다. 이 때문에 수식을 많이 사용하는 이공계와 과학계통의 출판물은 TeX을 많이 사용하고 있다.

외국어¹²가 포함된 문서의 작성시도 TeX의 사용이 불가피하다.

¹⁰레이아웃, 서체의 지정 etc.

¹¹색인, 목차 등의 페이지 번호, 수식 번호, 참고 문헌, 주석 etc.

¹²독일어, 불어, 희랍어, 라틴어, 스페인어 etc.