

Invitation to L^AT_EX Beamer

김강수 (KTUG)

2009년 12월 16일

요 약

이 글은 L^AT_EX Beamer 클래스를 이용하여 발표자료를 작성하기 위한 실습 수업의 참고자료로 작성되었습니다. 실제 Beamer의 기능과 사용법은 별도의 예제를 통하여 제공됩니다.

차 례

1	T _E X, L ^A T _E X and Beamer	3
	▶ T _E X, L ^A T _E X이란 무엇인가요?	3
	▶ L ^A T _E X 문서 작성의 특징이 있다면 무엇인가요?	3
	▶ Beamer란 무엇인가요?	4
	▶ Beamer로 프레젠테이션 자료를 만드는 것이 좋은 점이 있나요?	4
	▶ 어디에서 도움을 얻을 수 있나요?	5
2	Beamer 시작하기	5
	▶ Beamer로 작업하기 위해 준비해야 할 것은 무엇인가요?	5
	▶ Beamer 작업을 위해 반드시 알아야 할 L ^A T _E X 명령은 어떤 것이 있나요?	5
	▶ 첫번째 beamer 문서의 샘플을 보여주세요	7
	▶ Beamer에 테마를 적용할 수 있나요?	9
	▶ Beamer에 그림, 도형, 도표, 동영상 등을 넣을 수 있나요?	9
	▶ Beamer 문서 작업의 흐름은 어떠한가요?	9
	▶ 왜 beamer로는 파워포인트에서 같은 색시한 문서가 만들기 어렵나요?	9
	▶ Beamer의 주요 기능에 대한 참고문헌은 무엇이 있나요?	10
3	그밖의 문제	10
	▶ 한글과 글꼴	10
	▶ 좋은 프레젠테이션의 조건	11
4	사례를 통해 본 beamer	13
4.1	프레임의 구성요소	13
4.2	칼러	13
4.3	이단 프레임	14
4.4	다이나믹한 효과	14
4.5	장식	15

4.6	handout 만들기	16
4.7	결론	16

1 T_EX, L^AT_EX and Beamer

질문 1 T_EX, L^AT_EX이란 무엇인가요?

T_EX¹은 ‘조판 프로그래밍 언어와 그 운용 프로그램 및 매크로 집합’을 가리키는 말입니다. 맨처음에 T_EX은 1980년대 초에 Donald Knuth에 의하여 만들어졌으며 현재의 발전한 여러 확장-T_EX들이 이 프로그램에 바탕을 두고 있습니다. 조판 프로그램으로서 T_EX은 마치 프로그래밍하듯이 문서를 작성하게 합니다. 문서 작성자가 텍스트와 조판지시 매크로를 일련의 텍스트열로 입력하면 T_EX 프로그램은 이를 해석하여 가상의 페이지에 문자와 그림 등의 박스를 적절하게 배치합니다. 이런 페이지들이 모여서 하나의 문서가 되는 것입니다. 결과로 생성된 가상의 페이지를 프린트 드라이버로 보내면 출력물을 얻을 수 있습니다. 오늘날은 pdf 포맷의 파일로 출력하는 것이 일반적입니다.

L^AT_EX이란, T_EX 엔진에 바탕을 두고 보다 쉽게 **구조적으로** 문서를 작성할 수 있도록 하는 T_EX 매크로 모음으로서 본격적인 문서작성 시스템(Document Preparation System)으로 제작되었습니다. Leslie Lamport에 의하여 1980년대 초에 만들어졌으며 오늘날은 L^AT_EX3 그룹에 의하여 유지되고 있는 사실상 표준의 T_EX 사용 방식이라고 할 수 있습니다. 이 글에서 T_EX이라고 하면 다른 사정이 없는 한 L^AT_EX을 가리키는 것으로 하겠습니다.²

질문 2 L^AT_EX 문서 작성의 특징이 있다면 무엇인가요?

L^AT_EX을 사용한다는 것은 다음 두 가지 면에 그 특징이 있습니다.

1. mark-up 방식으로 문서를 작성한다.
2. 구조적 문서 작성 방식을 따른다.

일반적인 워드 프로세서의 문서 작성 방식을 WYSIWYG 방식이라고 합니다. WYSIWYG이란 What you see is what you get의 첫 글자를 딴 것으로 화면에 보이는 그대로 인쇄되어 나온다는 데서 온 말입니다. 예를 들어 화면에서 글자를 **두껍게** 하면 실제로 인쇄에서도 두꺼운 글자가 표시된다는 뜻입니다.

L^AT_EX에서는 이와는 다른 접근방법을 취하고 있습니다. L^AT_EX에서는 문서를 ‘코딩’하고 ‘컴파일’합니다. 즉, 문서의 모양이나 흐름을 통제하는 제어 명령 부분과 텍스트로 이루어지는 내용 부분을 하나의 소스(원본)로 작성하되 이 가운데 제어 명령 부분을 포매터(=엔진)에게 넘겨주어 문서의 최종적인 모양을 만들어내게 하는 것입니다. 이 과정은 한편으로는 프로그램을 작성하는 것과 비슷하고, 다른 한편으로는 XML 또는 HTML 문서의 mark-up과 대단히 유사합니다. 그 결과, L^AT_EX에서는 ‘논리적인’ 모양과 ‘물리적인’ 모양이 구별됩니다. 문서작성자는 ‘논리적’인 것만 신경쓰면 되고 ‘물리적인’ 최종 형태는 얼마든지 바꿀 수 있다는 것입니다. 이러한 L^AT_EX의 철학이 “저자로 하여금 내용에만 신경쓰게 한다”는 주장을 가능하게 하는 배경입니다.

구조적인 문서란 실은 XML이나 HTML의 mark-up에서 이미 익숙한 것입니다. 즉, 예를 들어 <section>과 </section>이라는 mark-up은 section이라는 형태의 모양을 결정하는 것이 아닙니다. 그냥 섹션타이틀이라는 지시일 뿐이지요. 이런 식으로 문서를 작성하면 adhoc하게 그때그때 폰

¹이 단어는 ‘텍스’라고 읽지 않습니다. 보통 ‘텍’이라고 발음합니다. 한편 L^AT_EX은 ‘라텍스’라고 철대로 읽지 않습니다. 라텍 또는 레이텍 정도가 좋습니다.

²T_EX에 바탕을 둔 매크로 시스템으로 L^AT_EX보다 더 나은 것이 있을 수도 있습니다. 예를 들면 설계나 구조, 기능 면에서 L^AT_EX보다 ConT_EXt가 더 나은 점이 있음을 부인할 수 없습니다. 그러나 전세계적인 사용자의 수와 보편화 정도는 L^AT_EX을 따라올 수 없습니다.

트 모양을 바꾸어서 섹션 타이틀을 설정하는 것과는 아주 다른 문서 구조를 얻게 됩니다.

질문 3 Beamer란 무엇인가요?

LaTeX은 문서의 종류를 그 목적과 용도에 따라 ‘클래스’로 나누고 있습니다. beamer는 PDF 슬라이드 또는 프레젠테이션 자료를 작성하는 목적의 클래스입니다. 실은 슬라이드 또는 발표 자료를 작성하는 LaTeX 관련 툴은 꽤 종류가 많습니다.³

그 중에서 가장 널리 알려져 있고 기능이 강력한 것이 beamer라는 것입니다. 즉, LaTeX으로 PDF 발표자료를 제작하는 클래스(매크로집합)의 명칭이 beamer입니다. 대부분의 LaTeX 클래스들이 주로 ‘책’과 같은 문서를 만드는 데 특화되어 있음에 비하면 beamer는 그 목적이 매우 특별한 것 중의 하나라고 할 수 있습니다.

이 클래스는 Till Tantau라는 분이 만든 것입니다.⁴ 이를 이용하는 사례가 워낙 많고 개발자가 지속적으로 기능을 업데이트하고 있으며 뛰어난 매뉴얼(200페이지가 넘는)을 갖추고 있습니다. 거기에다가 이 모든 툴이 모두 완전한 자유소프트웨어입니다.

LaTeX 문서작성 도구가 그렇듯이 beamer로도 상상가능한 일을 거의 할 수 있습니다. 만약 TeX과 LaTeX에 정통하고 있다면 그 다음 일은 오로지 무엇을 프레젠테할 것이냐는 문제 뿐입니다.

질문 4 Beamer로 프레젠테이션 자료를 만드는 것이 좋은 점이 있나요?

보통 발표자료는 powerpoint로 작성하는 경우가 많습니다. powerpoint도 훌륭한 프로그램입니다. 그러나 어쨌든 powerpoint가 beamer의 잠재적인 경쟁자이기 때문에 이 둘을 비교해보지 않을 수 없습니다.

예를 들면, 2008년에 진행된 KTUG 워크숍에서 Beamer에 대하여 발표하신 이응신 님은 표 1과 같은 비교의 논점을 제시하셨습니다.

표 1: PPT와 Beamer (by 이응신)

	PPT	Beamer
초급	쉽다	어렵다
중급	재미있다	재미있다
고급	짜증난다	쉽다
사용 용도	효과의 화려함	내용의 견고함
빈도	이따금 발표	규칙적으로 자주
기반	MS Office	LaTeX

이보다 더 중요한 점으로, 수식의 문제를 들 수 있습니다. 파워포인트로 프레젠테이션을 작성할 때 만약 발표자료에 수많은 수식이 들어가는 경우라면 거의 감당이 안 될 수밖에 없습니다. 그러나 TeX은 원래부터 수식을 식자하기 위해 개발된 언어였습니다. 수식 작성의 미려함과 편리함과 그 표준성은 다른 무엇과도 바꿀 수 없는 TeX, LaTeX의 장점입니다.

그리고 한 가지 더, 문서가 programmable하다는 바로 그 점이 LaTeX의 최대 장점 중 하나입니다. 그 문서의 모든 요소가 모두 작성자의 통제 하에 놓입니다. 이것은 처음 약간의 진입장벽만을

³Michael Wiedmann의 Screen Presentation Tools 페이지는 꼭 방문해보아야 할 곳입니다. <http://www.miwie.org/presentations/presentations.html>

⁴beamer 홈페이지 <http://latex-beamer.sourceforge.net/>.

넘어서면 그 순간부터 어떤 다른 도구로도 맞볼 수 없는 지적 즐거움을 제공할 것입니다.

질문 5 어디에서 도움을 얻을 수 있나요?

\TeX 사용을 위해서 참고 문서는 필수입니다. \TeX 시스템을 완전하게 설치하거나 인터넷을 통하여 무료로 원하는 참고 문서를 즉시 이용할 수 있습니다.

한글텍사용자그룹(KTUG)⁵은 설치에서 사용까지 \TeX 에 관련된 거의 대부분의 해결책을 제공하고 있습니다. 만약 문서를 작성하다가 막히는 부분이 있더라도 게시판에 질문하면 대부분의 문제에 대하여 친절한 답변을 제공해주는 분들이 많습니다.

2 Beamer 시작하기

질문 6 Beamer로 작업하기 위해 준비해야 할 것은 무엇인가요?

\TeX 시스템을 갖추면 그것으로 충분합니다. 현재 사용자가 선택할 수 있는 \TeX 시스템은 Windows 운영체제의 경우 \TeX Live라는 방대한 프로그램 및 매크로 집합으로 이루어진 프로그램이 있습니다. 한국텍학회(KTS)에서는 이 \TeX Live를 바탕으로 한글 사용을 위한 매크로인 ko \TeX 을 포함하여 ko \TeX Live라는 설치 프로그램을 제공하고 있습니다.

오늘 실습에 사용할 프로그램은 이를 간략화한 TnX \TeX 입니다. 이것은 정식으로 제공되는 프로그램이 아니라 실습을 위하여 \TeX Live를 추려서 만든 일종의 portable 버전입니다. 이것만으로도 거의 대부분의 작업을 하는 데 큰 지장은 없지만 본격적으로 Beamer를 사용할 의향이시라면 ko \TeX Live를 설치해보실 것을 추천합니다. 이 프로그램을 얻고 설치하는 방법은 앞서 소개한 한글텍사용자그룹 홈페이지를 방문하시면 안내를 받을 수 있습니다.

일단 이것만으로 충분합니다만, 자신의 연구분야에 따라 필요한 프로그램에서 그림이나 표, plot diagram 등을 얻어내어 프레젠테이션에 삽입해야 할 필요가 있을 수도 있습니다. 예를 들면 R이나 Maple, Mathematica 등이 있을 수도 있고 좀더 간결한 다이어그램을 위하여 Visio나 Dia를 쓸 수도 있을 것입니다. 경우에 따라 Adobe Illustrator에서 ai 형식의 그림을 다루어야 할 때도 있을 것입니다. 이와 같은 외부 그림들은 해당하는 툴, 그 중에서도 자신에게 익숙한 것을 고르면 됩니다. 이런 프로그램들로부터 얻어진 데이터를 beamer에 삽입하는 데는 여러 방법이 있는데 어떤 경우에도 동작하는 솔루션은 그림을 pdf 형식으로 얻어두는 것입니다. 한편, beamer 자체는 PGF/TikZ라는 그림 언어를 가지고 있기도 합니다.

질문 7 Beamer 작업을 위해 반드시 알아야 할 \LaTeX 명령은 어떤 것이 있나요?

앞서 언급한 대로 beamer는 어차피 \LaTeX 클래스이기 때문에 \LaTeX 의 입력 컨벤션을 알고 있어야 합니다. 이미 오랜 동안 \LaTeX 을 사용해 오셨다면 이 부분은 문제가 되지 않을 것이지만 여기에서는 입문자들이 자주 저지르는 실수와 간단한 참고문헌을 소개해두기로 하겠습니다.

- 입력은 모두 plain text 파일입니다. 파일은 저장한 후에 xelatex 명령이 처리하게 하는데, 이때 UTF-8 인코딩으로 입력하게 하는 것이 중요하지만 (우리가 사용할) texworks라는 프론트엔드 프로그램은 기본적으로 UTF-8 인코딩으로 동작하므로 texworks를 사용하는 한 이 부분은 걱정할 것이 없습니다.

⁵<http://www.ktug.or.kr>

- 입력은 논리적인 것이며 최종 출력의 모양을 결정하지 않습니다. 공백 문자들은 몇 개가 이어져도 모두 하나의 스페이스로만 처리됩니다. 줄바꿈은 빈 줄 하나로 처리됩니다.
- 다음 몇 개의 문자는 소위 “예약된 문자(reserved characters)”로서 그대로 입력하면 의도하지 않은 결과를 만듭니다. # \$ % ^ & _ { } ~ \. 이 문자들을 텍스트로 나타나게 하려면 반드시 다음과 같이 입력해야 합니다.

```
\# \$ \% \^{} \& \_ \{ \} \~{} \textbackslash
```

#, \$, %, & 등은 흔히 저지르는 실수에 속합니다.

- % 기호를 그대로 입력하면 주석문(comment)이 됩니다. 즉 % 기호 이후의 그 행의 어떤 것도 문서에 나타나지 않습니다.
- 모든 \LaTeX 명령은 대소문자를 구별하여 백슬래시(\)로 시작합니다. \LaTeX 명령은 문자로만 이루어지며 숫자나 공백은 명령으로 쓸 수 없습니다. 그래서 숫자나 공백이 나타나면 그 직전까지를 명령어로 간주합니다.
- \LaTeX 명령 뒤에 대괄호(bracket, [])나 중괄호(brace, { })가 오면 그 명령의 인자(argument)로 처리됩니다. 예를 들면 `\textbf {어쩌구}`라고 입력하면 “어쩌구”에 해당하는 부분이 인자로 간주되어 처리됩니다.
- `\begin{...}`과 `\end{...}`는 한 쌍을 이루어 이 범위 안의 텍스트에 대하여 일정한 작용을 가하는 역할을 합니다. 이것을 \LaTeX 에서 ‘환경(environment)’이라 부릅니다. 특정한 환경을 시작하였다면 반드시 끝을 내주어야 합니다. 번호붙은 리스트나 bullet 붙은 리스트, 문단 폭, 박스 등등이 모두 환경으로 처리됩니다. beamer의 기본 작업 단위인 frame도 그 자체로 하나의 환경입니다.
- \LaTeX 문서는 일정한 형식을 가지며 이것은 beamer도 예외가 아닙니다. 다음은 일반적인 \LaTeX 문서의 형식입니다.

```
\documentclass{article}
\usepackage{kotex}
\begin{document}
% here comes your text
\end{document}
```

첫 행의 `\documentclass`는 반드시 있어야 합니다. 우리는 이 부분에서 `article` 대신 `beamer`라고 써넣을 것입니다. 한편 `\begin{document}`에서 `\end{document}`까지가 문서의 본체에 해당합니다.

- 입력 파일을 처리하는 과정을 관행상 ‘컴파일한다’고 합니다. 컴파일은 명령행에서 직접 내릴 수도 있지만 `texworks`의 컴파일 버튼을 누르거나 단축키(Ctrl-T)를 누르면 진행됩니다.

Beamer 작업을 위해서 알아야 할 대부분의 기본적인 \LaTeX 명령은 `lshort-kr`이라는 문서를 보면 확인할 수 있습니다. 우리말로 번역되어 있으므로 참고하기 쉬운 것입니다.⁶ 일단 이 문서를 한번 죽

⁶<http://faq.ktug.or.kr/faq/lshort-kr>

읽으라고 많은 분들이 권합니다. 단지 일부 내용은 오늘날의 $\text{T}_{\text{E}}\text{X}$ 환경에 비해 낡은 내용도 있으므로 그런 부분은 그냥 건너뛰면 됩니다.

텍스트 입력 만큼이나 중요한 것이 수식 입력인데, 우선 수식 입력은 `lshort-kr`의 수식 입력에 관한 절을 참고하십시오. 그래도 어렵다고 느끼실까봐 간단한 유틸리티 하나를 제공하려 합니다. 이것은 실습 시간에 소개하겠습니다.

질문 8 첫번째 beamer 문서의 샘플을 보여주세요

다음은 가장 간단한 beamer 문서의 샘플입니다.

```
\documentclass{beamer}
\usepackage{kotex}

\title{첫번째 예제}
\author{홍 길 등}
\date{오늘}

\begin{document}

\maketitle

\begin{frame}
\frametitle{첫번째 슬라이드}
첫번째 슬라이드에 들어갈 내용
\end{frame}

\begin{frame}
\frametitle{두번째 슬라이드}
두번째 슬라이드에 들어갈 내용
\end{frame}

\end{document}
```

이 파일을 컴파일하면 결과가 어떻게 될까요? 일단 `myfirst.tex`이라는 이름으로 저장한 후 `Ctrl-T`를 눌러보겠습니다. 그림 1은 결과를 보여줍니다.

여기서 각각의 슬라이드가 *frame*이라는 단위로 작성되고 있음을 알 수 있습니다. 프레임에는 `frametitle`이 붙고 `\begin{frame}`과 `\end{frame}`이 한 짝을 이루고 있습니다.

결과를 보면, 아무런 theme이 적용되지 않은 심플하기 짝이 없는 결과가 나왔음을 알 수 있습니다. 그러나 `frametitle`이 적당한 크기로 붙어 있고 네비게이션 바가 있어서 각각의 프레임(슬라이드)이 제 모양을 갖추고 있음을 확인할 수 있습니다. 여기에 여러 장식을 붙이거나 테마를 적용하면 더 멋진 프레젠테이션 문서가 되어갈 것입니다.

한편, $\text{T}_{\text{N}}\text{X}\text{T}_{\text{E}}\text{X}$ 에는 beamer 템플릿이 마련되어 있습니다. $\text{T}_{\text{N}}\text{X}\text{T}_{\text{E}}\text{X}$ 또는 `koTEX Live`의 beamer 템플릿을 그대로 이용하여 작업하는 화면이 그림 2에 그림으로 보였습니다. 이것만으로도 훌륭한 문서를 작성할 바탕이 됩니다.

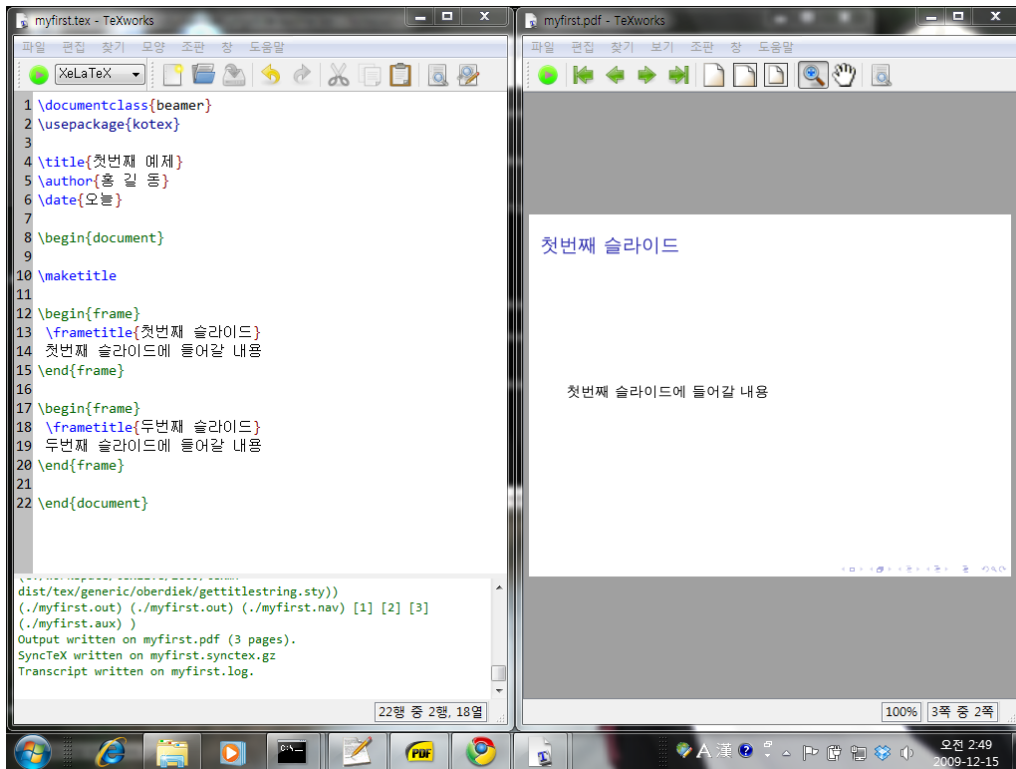


그림 1: 첫번째 beamer 문서

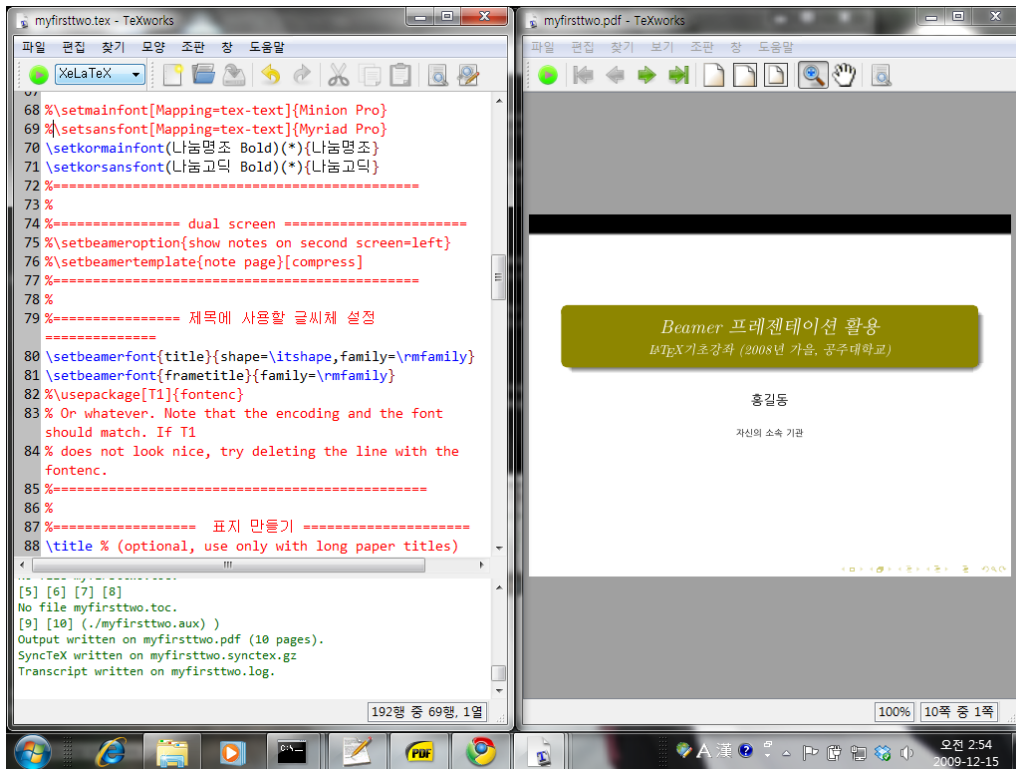


그림 2: TnX_{TeX} beamer template

질문 9 Beamer에 테마를 적용할 수 있나요?

물론입니다. Beamer의 테마는 도시 이름을 딴 것이 많은데, 기본적으로 제공되는 테마 이외에도 개인이 작성한 멋진 테마들도 꽤 있습니다. 우선 <http://www.pletscher.org/writings/latex/beamerthemes.php>에서 몇 가지 테마를 구경해보시지요.

Beamer의 최대의 문제점으로 많은 분들이 개인이 자신의 테마를 작성하기가 어렵다는 점을 들고 있습니다. 어려운 것은 사실인데 그 이유가 테마를 구성하는 요소가 워낙 많고 그 각각을 제어하는 루틴을 작성하는 것이 쉽지 않기 때문입니다. 그러나 이미 만들어져 있는 테마를 조금씩 변경하는 것은 그다지 어렵지 않습니다. 더 쉬운 것은 자신의 입맛에 맞는 테마를 발견하고 그걸 쓰는 것입니다.

질문 10 Beamer에 그림, 도형, 도표, 동영상 등을 넣을 수 있나요?

그림은 두 가지 방법이 있습니다. 하나는 미리 그림을 준비하여 이것을 include하는 것입니다. 이 방법이 가장 쉽습니다. 그런데 더 세련된 프레젠테이션 효과를 원한다면 PGF/TikZ 언어를 이용하는 것이 좋습니다. 말 그대로 상상력이 허용하는 최대한의 효과를 얻을 수 있을 것입니다.

동영상은 avi, mpeg 포맷을 삽입할 수 있습니다. 최근에는 flash 파일도 효과적으로 처리한다고 합니다.

이에 관련된 사항을 실습에서 다루어보겠습니다.

질문 11 Beamer 문서 작업의 흐름은 어떠한가요?

어떤 슬라이드든지 마찬가지로겠지만 다음과 같은 순서를 따릅니다.

1. 발표 내용을 확정한다.
2. 스토리보드를 만들고 주제어를 추려낸다.
3. 효과, 강조할 곳 등에 유의하면서 frame을 설계한다.
4. frame 별로 내용을 코딩한다.
5. 만들어진 pdf로 실제 발표 연습을 하면서 수정한다.

이 과정에서 texworks를 열고 해당하는 슬라이드를 수정하고 컴파일하는 과정을 반복합니다. 실제로 다른 것은 전혀 필요없고 코딩 → 확인 → 코딩을 반복하는 것에 불과합니다. 이 과정은 직접 해 보면 파워포인트로 만든 문서를 수정해가는 것과 그다지 다르지 않다는 것을 알게 될 것입니다. 단지 mark-up일 뿐이지요.

질문 12 왜 beamer로는 파워포인트에서 같은 색시한 문서가 만들기 어렵나요?

색시하다는 것이 어떤 것인지 모르겠지만, 대부분 내용에 자신이 없거나 미심쩍으면 화려한 효과로 무장하지 않습니까? 지나치게 효과에만 치중한 프레젠테이션은 어떤 의미에서 촌스럽게 보일 수 있습니다. beamer는 오히려 simplicity를 통하여 미적 감각을 드러내게 해줄 수 있다고 생각하는 분이 많습니다.

또다른 한편, beamer가 주로 문서를 만드는 도구에서 비롯되다보니 생긴 부작용으로서 텍스트

를 처리하는 능력이 너무 강력하다는 점이 있다고 봅니다. 이것이 장점만은 아닌 것이, 특히 영어나 외국어로 발표하는 경우 자신이 해야 할 말을 전부 프레젠테이션에 다 집어넣으려는 경향을 강하게 띠게 되는데, 이것이 자료를 난잡하게 만들 수 있다고 생각합니다.

beamer는 깔끔하고 세련된 프레젠테이션을 만들어줍니다. 그러나 그만큼 심심한 면도 있죠. 이 껍을 넘어서 주목을 끌고 좋은 프레젠테이션이 되게 하려면 내용이 알차지 않으면 안됩니다. 어떤 의미에서 beamer를 사용하여 만드는 프레젠테이션의 장점이라면 역시 “내용에 주목하도록 요청한다”는 점이 아닐까 생각합니다.

결국 문제는 얼마나 자신의 발표를 잘 설계하느냐는 것이고, beamer는 이 생각, 아이디어를 구현해주는 도구로서 그 기능을 충분히 발휘할 수 있다는 것입니다.

질문 13

Beamer의 주요 기능에 대한 참고문헌은 무엇이 있나요?

이 글의 부록으로 제공하는 “사례를 통해 배우는 beamer”는 짧지만 알찬 내용을 가진 문서입니다. 그리고 beamer에 관한 한, beamer manual은 항상 곁에 두는 것이 좋습니다. 만약 PGF를 사용하기로 결정했다면 PGF manual도 함께 있어야 합니다. 이 두 매뉴얼을 합치면 수백 페이지에 이르는 방대한 양입니다만, 차례로 다 읽을 필요는 없고 개관을 이해한 후에 필요한 부분을 찾아서 보는 것이 좋겠습니다.

- Beamer Manual. <http://ftp.ktug.or.kr/tex-archive/macros/latex/contrib/beamer/doc/beameruserguide.pdf>
- PGF Manual. <http://ftp.ktug.or.kr/tex-archive/graphics/pgf/base/doc/generic/pgf/pgfmanual.pdf>

앞서 소개한 koTeX Live를 full version으로 설치하였다면 인터넷에 접속하지 않아도

```
# texdoc beamer
```

또는

```
# texdoc pgf
```

명령으로 이 매뉴얼을 즉시 읽을 수 있습니다. beamer 매뉴얼은 224쪽이고 pgf 매뉴얼은 560쪽입니다.

이밖에 beamer로 작성된 많은 beamer 소개 문서가 있습니다. 구글 등의 검색 기능을 이용하여 찾아보면 훌륭한 예제를 발견할 수도 있을 것입니다. 그 중 일부는 오늘 강의에 샘플로 제공되는 것도 있습니다. KTUG의 Beamer 페이지와 SampleDocument 페이지⁷에도 좋은 문서가 많이 있습니다. 여기에서 찾을 수 있는 문서만으로도 훌륭한 발표자료를 작성하는 데 충분합니다.

3 그밖의 문제

질문 14

한글과 글꼴

종래 TeX 사용이 그다지 인기를 얻지 못했던 이유 중의 하나가 시스템 글꼴을 자유롭게 사용하

⁷각각, <http://faq.ktug.or.kr/faq/Beamer>, <http://faq.ktug.or.kr/faq/SampleDocument>.

기 힘들다는 점이 있었습니다. 그러나 최근 XeTeX의 등장으로 이 문제는 근본적으로 해결되었습니다. Beamer는 XeTeX과 아주 잘 동작하며, 필요하다면 pstricks와 같은 종래 Beamer에서 사용하기 까다롭던 툴도 잘 사용할 수 있게 되었습니다.

한편, beamer에서 한글을 구현하기 위해서는 ko.TeX을 쓰면 간단합니다. 다음 한 줄을 preamble에 추가하면 됩니다.

```
\usepackage{kotex}
```

실제로 beamer 자체에 대한 많은 소개 문서는 pdflatex이나 latex과 dvips, ps2pdf를 이용하여 컴파일하는 과정을 주로 소개하고 있습니다. 그러나 최근의 경향은 XeTeX을 이용하는 것입니다. 여러 가지 제약들이 모두 사라지고 매우 편리한 작업 환경을 경험할 수 있게 됩니다. TnXTeX은 오직 XeTeX만을 포함하고 있는 것도 이러한 이유입니다. 어쨌든 한글에 관한 한, 컴파일 엔진이 무엇이든 상관없이 위의 한 줄로 모두 해결됩니다.

질문 15

좋은 프레젠테이션의 조건

Beamer는 다만 도구입니다. 도구가 좋아야 자신이 의도한 바를 잘 구현할 수 있는 것이지만 때때로 도구에 매몰되어 도구 자체가 프레젠테이션의 품질을 결정한다는 착각을 하곤 합니다. 스티브 잡스의 키노트가 멋진 것은 매킨토시에서 키노트 프로그램을 사용하기 때문은 아닐 것입니다.

여기서는 많은 분들이 원하는 좋은 프레젠테이션 작성 방법을 단편적으로 몇 가지 제시하려 합니다. 이것은 절대적인 원칙은 아닐 것이며 자신에게 적합한 방법을 찾는 데 참고로 하는 정도로 활용되기를 바랍니다.

- (1) 한 슬라이드에 너무 많은 내용을 넣지 말 것. 기본적으로 슬라이드에 들어가는 것은 주제어, 주목을 요청하는 말이 위주가 되어야 할 것입니다. 자신이 할 말을 전부 슬라이드에 넣는 것은 불가능하기도 하거니와 청중을 피곤하게 하는 것이랍니다. 10/20/30 원칙이라는 것은 “10장의 슬라이드, 20분의 발표시간, 30포인트 글씨크기”라는 말인데 항상 적용될 수 있는 것은 아니지만 일리가 있다고 봅니다.
- (2) 책을 쓰는 것과 혼동하지 말 것. 가끔 슬라이드에 참고문헌 목록, 명제의 번호 등을 상호참조 방식으로 넣거나 각주를 달거나 하는 것을 봅니다. 그러나 듣는 사람 입장에서는 앞서 나온 명제 번호를 기억할 리가 만무합니다. 참고문헌 목록은 간결하게 어떤 문헌이 참조되었다는 정도면 충분하고 상세한 서지를 밝히거나 참조 위치를 번호로 표시하거나 할 필요가 거의 없을 것입니다. 슬라이드 작성은 책을 쓰는 것이 아닙니다.
- (3) 슬라이드의 압축도에 리듬을 부여할 것. 빠르게 진행되는 슬라이드와 한 페이지에서 오래 머무는 슬라이드가 필요에 따라 함께 사용되는 것이 좋습니다. 처음부터 발표의 리듬감을 설계하고 이에 따라 슬라이드의 분량과 압축도를 결정하는 것이 좋습니다.
- (4) 간결하면서도 적절한 자극을 부여할 것. 간결한 것은 슬라이드의 생명입니다. 강의용 슬라이드는 대부분 필기해야 할 것을 보여주기 때문에 장황해져도 어쩔 수 없지만 발표용 슬라이드는 주제를 명확하게 드러내는 몇 단어만으로 슬라이드를 구성하는 것이 훨씬 효과적이라는군요.
- (5) 일관성과 변화. 지나치게 지루한 슬라이드는 정형화된 형식에서 오는 경우가 많습니다. 적절한 변화를 부여하는 것이 필요하지만 변화가 위주가 되어서는 안 될 것입니다. 일관성을 유지하되

변주(variation)를 통해서 흥미를 유지하도록 하는 것이 좋다고 합니다.

4 사례를 통해 본 beamer

이 장은 Andrew Mertz와 William Slough가 *The LaTeX Journal* 2005년 10월호에 발표한 글을 발췌 번역한 것입니다. 원문은 <http://www.ktug.org/pracjourn/2005-04/merz/merz.pdf>에서 볼 수 있습니다.

4.1 프레임의 구성요소

beamer로 문서를 작성하는 것은 새로운 환경인 **frame** 환경 하나를 이해하는 것이다. 하나의 프레임은 다음 몇 가지 요소로 이루어진다. 이 중 어떤 것은 없어도 되는 것이다.

- 상단선과 하단선 (headline and footline)
- 오른쪽, 왼쪽의 사이드바
- 네비게이션 바
- 로고
- 프레임 타이틀
- 배경
- 내용

프레임의 내용에 관한 한, 대부분의 \LaTeX 명령과 환경이 그대로 동작한다. 따라서 예를 들면 `enumerate`나 `itemize`, 수식, `tabular` 등이 보통 \LaTeX 문서에서와 동일하게 들어간다. 유일하게 다른 점은 이것이 모두 beamer의 `frame` 안에 위치해야 한다는 것뿐이다.

`verbatim` 텍스트를 포함하는 프레임은 특별한 주의가 필요한데 이 프레임에는 `fragile` 마크가 필요하다. 즉

```
\begin{frame}[fragile]
...
\end{frame}
```

과 같이 작성되어야 한다.

그림을 포함하려면 `graphicx` 패키지의 기능을 이용할 수 있다. beamer는 이 패키지를 자동으로 로드하므로 별도로 선언해줄 필요가 없다. 그림 크기와 관련하여 beamer의 기본값으로 한 장의 슬라이드 크기가 128mm×96mm임을 기억하는 것이 도움이 될 것이다. 그림의 포맷은 JPEG, PNG, PDF, MPS와 (\XeLaTeX 을 사용한다면) EPS도 가능하다.

4.2 칼러

폰트 변경으로도 되지만, 색상을 바꾸어서 강조의 효과를 얻을 수 있다. `xcolor` 패키지를 beamer가 자동으로 로드하고 색상에 관련된 모든 설정은 `xcolor`의 방식을 따른다. 특히 “이름으로 참조하는” 색상과 ! 부호로 설정하는 색상 농도는 매우 유용하다. `{\color{BlueViolet!30} A B C}`는 “A B C”라는 텍스트를 BlueViolet 색상 30% 농도로 찍을 것이다.

특히 효과적인 색상 사용 방법 중에 `tabular` 환경에 대하여 `rowcolors`를 적용하는 것이 있다.

이 명령으로 두 개의 색상을 설정할 수 있는데 이 두 색상이 하나씩 교대로 각 row에 배경으로 쓰인다.

```
\rowcolors{1}{RoyalBlue!20}{RoyalBlue!5}
```

이렇게 하면 RoyalBlue라는 색상이 첫째 줄에는 20% 농도로, 둘째 줄에는 5% 농도로 입혀지고 그 이후 반복된다. 이 명령을 쓰려면 beamer 클래스를 선언할 때 다음과 같은 옵션을 주어야 한다.

```
\documentclass[xcolor={svgnames,table}]{beamer}
```

이 중 tabular에 \rowcolors를 쓰기 위한 옵션이 table이다.

4.3 이단 프레임

프레임은 가로가 긴(landscape) 방향으로 되어 있기 때문에 프레임을 두 개의 단으로 나누고 싶을 때가 있다. columns 환경은 이 때 쓰도록 만들어진 것이다.

columns 환경은 정렬 옵션을 줄 수 있다. 각각 꼭대기(t), 바닥(b), 가운데(c) 세 가지 옵션을 선택할 수 있다. columns 환경 안에서 각 단은 \column 명령을 두 번 써서 구분한다.

```
\begin{frame}
\frametitle{이단 프레임}

\begin{columns}[c]
\column{1.5in}
Practical \TeX\ 2005\
\column{1.5in}
Practical \TeX\ 2005
\end{columns}

\end{frame}
```

4.4 다이나믹한 효과

몇 개의 그림을 겹쳐가면서 진행하여 최종적으로 특정한 이미지를 보여주고 싶을 때가 있다. beamer에서 이것을 *overlay*라고 부른다. 지금까지 작성해 본 샘플은 전부 하나의 프레임이 하나의 오버레이로 되어 있었다. 이럴 경우 프레임은 그 자체로 모든 내용을 한번에 전부 보여준다.

한편, 여러 개의 오버레이로 하나의 프레임을 구성할 수 있다. 이 때는 순차적으로 각 오버레이가 디스플레이되면서 진행된다. 애니메이션된 효과 같은 것을 만들어내는 데 순차 오버레이가 필요하다. 여기서는 beamer가 제공하는 세 가지 방법을 알아보겠다. 각각은 \pause를 이용하는 방법, 오버레이 위치의 지정 방법, 그리고 복수의 그래픽 파일을 포함하는 방법이다.

\pause를 이용하는 방법은 가장 간단하다. 이 명령이 있는 곳까지를 하나의 오버레이로 만드는 것이다. 이런 식으로 하나의 프레임을 여러 개의 오버레이로 조각낼 수 있다. 실제 PDF가 상연될 때 각 오버레이는 차례대로 보여진다. 한편, \setbeamercovered{dynamic}이라는 명령을 활성화하면 각 오버레이들은 완전히 가려지는 것이 아니라 희미하게 보이게 된다. 이것은 발표자가 현재

프레임의 모든 정보를 숨기지 않으면서도 발언하고 있는 주제에 주목하게 만들어준다.

두 번째 방법은 오버레이 번호를 붙이는 방법이다. beamer에서 모든 오버레이는 내부적으로 1에서 시작하는 번호가 붙는다. <번호>라는 오버레이 지정자를 이용하여 어떤 오버레이에서 해당 요소가 활성화될 것인지를 정해줄 수 있다. 사용방법은

```
\onslide<m->{text}
```

과 같다. 이렇게 하면 text가 m번째 이후 슬라이드에서 모두 visible하게 된다. 현재 표시된 내용 이외의 오버레이를 숨기기 위해서는 \setbeamercovered{invisible}이라고 선언한다. beamer에서는 일일이 \onslide를 쓰지 않아도 예컨대 \item과 같은 항목 명령에 바로 오버레이 지정자를 달 수 있도록 해두고 있다.

세 번째 방법은 xmpmulti 패키지와 그림그리기 툴을 이용했다. Tic-Tac-Toe 게임을 순차적으로 애니메이션하기 위해서 먼저 각 레이어에 해당하는 그림들을 그려서 pdf 그림으로 저장해두었다. 이들을 각각 game-0.pdf, game-1.pdf 등이라고 하기로 하자. xmpmulti 패키지가 제공하는 \multiinclude 명령을 이용하여 다음과 같이 각 오버레이별로 하나씩의 그림을 디스플레이하도록 하였다.

```
\begin{frame}
\frametitle{Tic-Tac-Toe}
\setbeamercovered{invisible}
\begin{center}
\multiinclude[format=pdf,width=3in]{game}
\end{center}
\end{frame}
```

[역주] 이 글에서는 xmpmulti 패키지를 언급하였으나 현재는 animate 패키지가 표준이다. 위와 똑같은 기능을 가진 명령은 \animategraphics이다.

4.5 장식

beamer에 여러 방식으로 장식하여 “웃입히기”할 수 있다. 바닥선에 자신의 이름, 소속, 기관, 로고 등을 넣을 수도 있다.

```
\setbeamertemplate{footline}[text line]{A Sample Talk}
```

이렇게 하면 발표 제목이 바닥선에 표시된다.

beamer는 기본 글꼴이 \sfamily이다. 따라서 이 글꼴 군을 다른 것으로 바꾸면 원하는 폰트를 사용할 수 있다. X_YLA_TE_X(-ko)에서는 다음과 같이 한다.

```
\setsansfont[Mapping=tex-text]{ComicSansMS}
\setsanshangulfont[Mapping=tex-text]{맑은 고딕}
```

다양한 테마를 선택하거나 서로 결합하여 사용할 수 있다. 예를 들어 다음을 preamble에 적어 넣으면 Berkeley 테마에 dolphin 컬러 테마를 적용할 수 있다.

```
\usepackage{beamerthemesplit}
\usetheme{Berkeley}
\usecolortheme{dolphin}
```

4.6 handout 만들기

프레젠테이션 내용을 인쇄된 문서로 나누어주거나 해야 할 때가 있다. 여기에는 두 가지가 필요하다. 하나는 오버레이로 이루어진 각 슬라이드를 오버레이 없이 제작해야 하고 그 뒤에 몇 장의 슬라이드를 한 장의 종이에 찍어야 한다.

오버레이를 제거하는 것은 간단하다. `beamer` 클래스 옵션으로 `handout`을 적어주기만 하면 된다.

```
\documentclass[handout,xcolor={svgnames,table}]{beamer}
```

이렇게 하여 오버레이들이 제거되고 모든 오버레이 요소가 한 장의 프레임으로 압축된다. 그 다음에 여러 장의 프레임을 하나의 종이에 찍는 것인데, 이 때는 `pdfpages` 패키지를 사용한다.

4.7 결론

\LaTeX 공동체의 많은 사람들이 고품질 조판 표준을 이용한 슬라이드 제작에 기여하였다. 특히 Till Tantau와 다른 패키지 제작자들에게 우리는 크게 빚지고 있다.