

L^AT_EX e Beamer

What You See Is What You Mean

Francesco Agnolazza
Luca “Syslac” Mezzalira

Montebelluna Linux User Group

23 ottobre 2010 – Castelfranco Veneto



Licenza d'utilizzo

Copyright © 2010, Francesco Agnolazza

Luca "Syslac" Mezzalana.

Questo documento viene rilasciato secondo i termini della licenza Creative Commons (<http://creativecommons.org>).

L'utente è libero di:

distribuire, comunicare al pubblico, rappresentare o esporre in pubblico la presente opera

alle seguenti condizioni:

Attribuzione Deve riconoscere la paternità dell'opera all'autore originario.

Non commerciale Non può utilizzare quest'opera per scopi commerciali.

Share-Alike Puoi distribuire eventuali modifiche a quest'opera solo con licenza identica o equivalente a questa.

In occasione di ogni atto di riutilizzo o distribuzione, deve chiarire agli altri i termini della licenza di quest'opera.

Se ottiene il permesso dal titolare del diritto d'autore, è possibile rinunciare a ciascuna di queste condizioni. Le utilizzazioni libere e gli altri diritti non sono in nessun modo limitati da quanto sopra. Questo è un riassunto in lingua corrente dei concetti chiave della licenza completa (codice legale), reperibile sul sito Internet

<http://creativecommons.org/licenses/by-nc-sa/3.0/legalcode>



- 1 Introduzione
 - Cosa
 - Perché
 - Come
- 2 Scrivere
 - Basi
 - Struttura
 - Math
 - Editing avanzato
- 3 Beamer
 - Tecnicismi
 - Aspetti grafici
 - Frames
 - Trucchetti & multimedia
- 4 Extra
 - Strumenti
 - Web



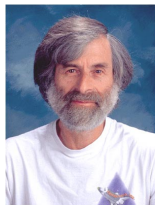
L^AT_EX

1977 Donald Knuth crea T_EX.

1985 Leslie Lamport estende T_EX,
facendo nascere così L^AT_EX.

Ok, ma cos'è? L^AT_EX è un linguaggio di
markup per la creazione di
documenti

E in pratica? è usatissimo per le
pubblicazioni scientifiche e in
editoria in generale.



Word Processors

Approccio completamente diverso, lontano dal *WYSIWYG*, con tutti i vantaggi e gli svantaggi del caso.



HTML

Il L^AT_EX si può pensare come un html per scrivere testi, con foglio di stile automatico.



It's science!

$$e^{\pi i} + 1 = 0$$

$$\nabla \cdot B = 0$$

$$\int e^x dx$$

Perché in realtà non è faticoso

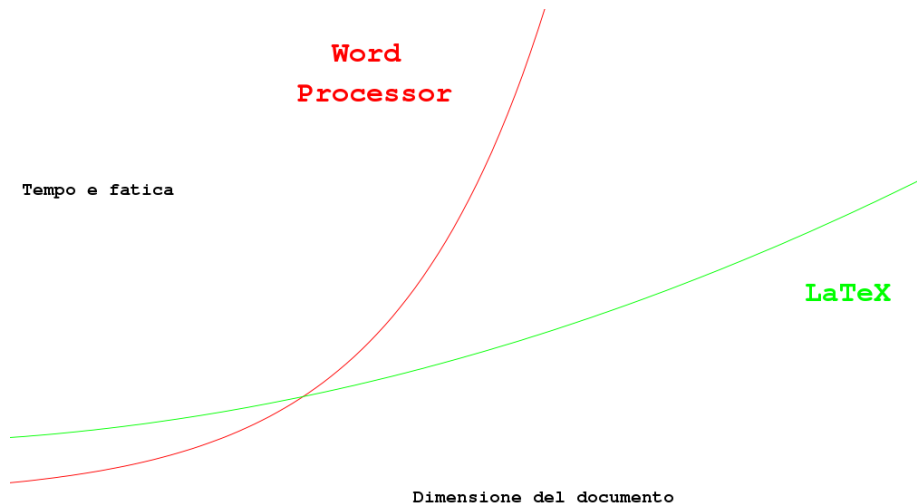
L^AT_EX pensa infatti a tutti i vostri problemi di formattazione, indentazione, litigi con Writer per il posizionamento immagini, etc, permettendovi di concentrarvi su quello che volete scrivere.^a

^aE poi, “It’s free” :D

Perché è bello

La resa grafica, sia in pdf, che una volta stampato, è decisamente superiore alle alternative, dando un look “professionale” ai documenti.

Impossible is nothing



Su Linux



Il nome del pacchetto è, a seconda della distribuzione, `texlive`, `tetex` o `texlive-latex-base`.

Su Debian o Ubuntu si può installare con :

```
apt-get install  
texlive-latex-base
```

Su Windows



La cosa più semplice è scaricare tutto il sistema **MiKTeX** da

<http://miktex.org>, che comprende compilatore, editor, etc.



Editor

Esistono editor appositi, anche visuali, ma in realtà basta creare un file `.tex` con un qualsiasi editor di testo, per quanto semplice.



Compilazione

Dipende dal formato di output scelto: la scelta più comune, per avere direttamente il pdf, è usare questo comando :

```
pdflatex nomefile.tex
```

L'alternativa `latex nomefile.tex` produce un `.dvi` che si può convertire, etc.



Struttura del documento

Un file tex è diviso in due parti fondamentali :

- Preambolo :
 - 1 Tipo di documento
 - 2 Lingua e font
 - 3 Pacchetti da includere
 - 4 Titolo, autore, etc.
- Documento :
 - 1 Corpo del documento, cioè tutto quello che sta tra
`\begin{document}`
e
`\end{document}`

Preambolo d'esempio

```
\documentclass[a4]{report}  
\usepackage[latin1]{inputenc}  
\usepackage[T1]{fontenc}  
\usepackage{graphicx}  
\usepackage{amsmath}
```

Un comando \LaTeX ha questo aspetto ¹:

```
\makecoffee[macchiato]{3}
```

- La parte appena dopo il backslash è il nome del comando
- Tra parentesi quadre, se il comando le accetta, le eventuali opzioni
- Tra parentesi graffe gli argomenti (uno o più) del comando.

Esempi di comandi

```
\textbf{Grassetto}  
\includegraphics  
[width=1cm]{immagine.png}  
\documentclass  
[twocolumns]{book}
```

Comando vs ambiente

Oltre ai comandi esistono anche gli ambienti, con questa sintassi:

```
\begin{nome_ambiente}  
    Parte di documento  
    interessata  
\end{nome_ambiente}
```

¹ Il comando di esempio è previsto solo per qualche versione futura :D

- **Grassetto**

```
\textbf{grassetto}
```

- *Corsivo*

```
\textit{corsivo}
```

- Monospace

```
\texttt{monospace}
```

- β , È, α , ...

```
\ss, \E,  
\alpha, \ldots
```

Whitespace e newline

Normalmente gli spazi e le righe vuote vengono ignorate; si può forzare la fine riga con

```
\\
```

o inserire spazi facendone l'escape.

L'eccezione è la singola (o più di una) riga vuota, che funziona come divisione di paragrafo.



Vediamo una panoramica di elenchi, tabelle e note a pié di pagina ²

Elenchi

Elenchi puntati `\begin{itemize}`
 `\item voce`
 `\end{itemize}`

Elenchi numerati `\begin{enumerate}`
 `\item voce`
 `\end{enumerate}`

Descrizioni `\begin{description}`
 `\item [titolo] voce`
 `\end{description}`

²Che sono assolutamente fenomenali



Titolo

Campo1	campo2	campo3
1	0	0
0	1	0
0	0	1

Tabelle

Usiamo l'ambiente tabular, ecco qualche parte di questa tabella :

```
\begin{tabular}{ccc}  
Camp01 & campo2 & campo3 \\ \hline  
1 & 0 & 0 \\  
0 & 1 & 0 \\  
0 & 0 & 1 \\  
... \\ \end{tabular}
```

Footnotes

```
``ciao \footnote{a tutti} dal MontelLUG''
```

diventa “ciao^a dal MontelLUG”

^aa tutti

Dividere il testo

L^AT_EX mette a disposizione vari modi per dividere il testo, o per “dare dei titoli” alle varie parti.

Se il documento è stato dichiarato come articolo ci sono a disposizione

- `\subsection{titolo}`
- `\section{titolo}`

A questi si aggiungono `chapter`, se il testo è un `report`, e `part`, se il testo è `book`.

Sezioni numerate

Di default, in tutti questi casi, la parte è accompagnata da un numero, che comparirà in output; se si preferisce evitarlo basta modificare il comando in

```
\section*{Titolo}
```



TOC

Il vero vantaggio di dividere in sezioni, etc, il testo, sta nel fatto che \LaTeX è in grado di creare da solo l'indice, basta usare il comando

```
\tableofcontents
```

L'indice risultante aggiorna in automatico numeri di pagina, link (se inseriti), ed è decisamente bello.

Algebra Lineare Applicata

Luigi Salve, Eloisa Detomi

March 9, 2010

Contents

1	Risultati di base	2
1.1	Richiami	2
1.2	Teoremi spettrali	3
2	Decomposizioni di matrici	3
2.1	Decomposizione LU	3
2.2	Decomposizione QR	5
2.3	Decomposizione in Valori Singolari (SVD)	6
2.4	Decomposizione Polar	7
3	Pseudoinversa	8
3.1	Pseudoinversa	8
3.2	Soluzioni ai minimi quadrati	9
4	Norme matriciali	10
4.1	Norme	10
4.2	Raggio Spettrale	11
5	Matrici Hermitiane	12
5.1	Richiami sulle forme quadratiche	12
5.2	Principi Variazionali	13
5.3	Matrici definite positive	14
5.4	Cofattore Diagonale-spettro	16
5.5	Complemento di Schur	17
6	Matrici bistocastiche	17
6.1	Prime definizioni	17
6.2	Combinazioni convesse	18
7	Matrici non-negative	19
7.1	Definizioni	19
7.2	Raggio spettrale	19
7.3	Matrici irriducibili	21
7.4	Matrici primitive	22
8	Modelli	24
8.1	Modello di Leslie	24
8.2	Baricentri dei sottotriangoli	25



Si possono inserire formule in vari modi, vediamo le differenze sullo stesso esempio:

1 Forse sapete già che
`\int e^x dx = e^x`

1 Forse sapete già che $\int e^x dx = e^x$

2 Forse sapete già che
`$$\int e^x dx = e^x$$`

2 Forse sapete già che^a

$$\int e^x dx = e^x$$

3 Forse sapete già che
`\begin{equation}`
`\int e^x dx = e^x`
`\end{equation}`

3 Forse sapete già che

$$\int e^x dx = e^x \quad (1)$$

^aCosa che ha fatto nascere alcune delle peggiori barzellette della storia



Posizionamento

Il carattere che segue un `_` va a pedice delle formula, quello che segue un `^` va ad apice.

Con i comandi giusti “pedice” ed “apice” significano le cose giuste:

```
\sum_1^5 ; \int_0^1 ; \lim_{x \to \infty}
```

Raggruppamento

Per raggruppare qualunque cosa (ad esempio per mettere ad esponente un'intera formula, non solo una cifra), basta racchiudere il testo tra `{}`, come in $e^{\log(x)} = x$

Spaziatura

In ambiente matematico, **tutti** gli spazi sono ignorati, per inserirli si usano `\;` `\:` `\quad` `\qquad` etc

Esempi (giuro che esistono tutti)

$$p(n) = \frac{1}{\pi\sqrt{2}} \sum_{k=1}^{+\infty} A_k(n) \sqrt{k} \frac{d}{dn} \left(\frac{\sinh \left(\frac{\pi}{k} \sqrt{\frac{2}{3} \left(n - \frac{1}{24} \right)} \right)}{\sqrt{n - \frac{1}{24}}} \right)$$

$$i\hbar\delta_t\varphi(t, x) = -\frac{\hbar^2}{2m}\delta_{xx}\varphi(t, x) + V(x)\varphi(t, x)$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} A = A \quad \forall A \in M_3(\mathbb{C})$$

$$\int_1^{+\infty} \frac{1}{x^\alpha} < +\infty \quad \iff \alpha > 1$$

$$a+bi = \rho(\cos(\theta)+i\sin(\theta)) = \rho e^{i\theta}$$



`\newcommand`

Utilizzo

Il titolo qui sopra arriva da questo codice :

```
\newcommand{\titolo}[1]  
{\center\LARGE\underline{\textbf{#1}}}  
...  
\titolo{Utilizzo}
```

La forma generale è :

```
\newcommand{\nome}[#argomenti]{definizione}
```

Attenzione

Ricordarsi comunque che il \LaTeX **non** è un linguaggio di programmazione.

graphicx Quello che permette di inserire le immagini con `\includegraphics`

xcolor Colorazione del testo.

shapepar Per paragrafi con forme strane.

Come questo

testo a forma di

“esa- gono

bu- cato”

(se- condo

il ma- nuale),

sempre che tut-

to funzioni.

Ma soprattutto ...



Beamer

è la classe di \LaTeX che si occupa delle presentazioni.

Ha tutte le potenzialità di un testo scritto in \LaTeX :

- Attenzione rivolta al contenuto;
- Sintassi semplice e intuitiva;
- Particolarmente adatto a presentazione di caratte scientifico;



Per dichiarare che vogliamo usare Beamer come classe per la presentazione, dobbiamo cominciare il nostro file .tex con la seguente istruzione:

Sintassi

```
\documentclass{beamer}
```

Sintassi avanzata

```
\documentclass[draft]{beamer}  
\documentclass[handout]{beamer}
```



Essendo `beamer` una classe di \LaTeX , valgono gli stessi comandi per Titolo, Sottotitolo, Autore, Data e Institute:

```
\title[\LaTeXe]{\LaTeXe e Beamer}
\subtitle{What You See Is What You Mean}
\author[AgNO \& Syslac]{Francesco ''AgNO'' Agnolazza
\ Luca ''Syslac'' Mezzalira}
\date{23 ottobre 2010 -- Castelfranco Veneto}
\institute{Montebelluna Linux User Group}
```

I quali possono essere richiamati con il solito comando:

```
\titlepage
```

magari nella prima slide della vostra presentazione ;)



Indice

Ovviamente questo vale anche per `tableofcontents`.
Infatti si può creare il sommario della presentazione con la stessa facilità vista prima, dichiarando semplicemente il nome delle varie parti con il comando

```
\section{titolosezione}
```

e/o

```
\subsection{titolosottosezione}
```

Logo

Per inserire il logo che verrà visualizzato in ogni diapositiva, esiste il comando apposito:

```
\logo{\includegraphics[options]{name_file}}
```

N.B. Basta inserirlo una sola volta nel file!



Una parte sicuramente importante di una presentazione è la veste grafica. Beamer mette a disposizione dell'utente una serie di temi predefiniti.

Dichiarazione

```
\usetheme{Madrid}  
\useinnertheme{rounded}  
\useoutertheme[right]{sidebar}  
\usecolortheme{albatross}
```



usetheme

È il tema principale e solitamente è il nome di una città. Alcuni esempi sono Madrid, Berlin, Barkeley, ecc...



```
useinnertheme{}
```

Determina il formato degli elenchi e dei blocchi.

Opzioni possibili sono:



`useinnertheme{}`

Determina il formato degli elenchi e dei blocchi.

Opzioni possibili sono:

- `circles;`
Elenchi contrassegnati da cerchi.



`useinnertheme{}`

Determina il formato degli elenchi e dei blocchi.

Opzioni possibili sono:

- `circles;`
Elenchi contrassegnati da cerchi.
- `rounded;`
Elenchi contrassegnati da palline e riquadri con angoli smussati.



`useinnertheme{}`

Determina il formato degli elenchi e dei blocchi.
Opzioni possibili sono:

- `circles;`
Elenchi contrassegnati da cerchi.
- `rounded;`
Elenchi contrassegnati da palline e riquadri con angoli smussati.
- `retangles;`
Elenchi contrassegnati da rettangoli.



`useoutertheme{}`

Permette di inserire una barra laterale di navigazione e di decidere cosa far apparire in alto o sul fondo della slide. Opzioni possibili sono:



`useoutertheme{}`

Permette di inserire una barra laterale di navigazione e di decidere cosa far apparire in alto o sul fondo della slide. Opzioni possibili sono:

- `sidebar;`
Barra laterale la cui posizione è determinata dall'opzione `[right]` o `[left]`.



`useoutertheme{}`

Permette di inserire una barra laterale di navigazione e di decidere cosa far apparire in alto o sul fondo della slide. Opzioni possibili sono:

- `sidebar;`

Barra laterale la cui posizione è determinata dall'opzione `[right]` o `[left]`.

- `infolines;`

In cima alla diapositiva c'è scritto il **nome** della **sezione** in cui si trova la slide corrente e nel fondo invece vengono inseriti il nome dell'**autore**, **istituzione**, **data** e **numero pagina**.



`useoutertheme{}`

Permette di inserire una barra laterale di navigazione e di decidere cosa far apparire in alto o sul fondo della slide. Opzioni possibili sono:

- `sidebar;`
Barra laterale la cui posizione è determinata dall'opzione `[right]` o `[left]`.
- `infolines;`
In cima alla diapositiva c'è scritto il **nome** della **sezione** in cui si trova la slide corrente e nel fondo invece vengono inseriti il nome dell'**autore**, **istituzione**, **data** e **numero pagina**.
- `miniframes;`
Inserisce 4 barre, 2 superiori con **sezione** e **frame corrente** e 2 inferiori con **autore** e **titolo presentazione** + **istituzione**.

`usecolortheme{}`

Permette di definire lo schema di colori da utilizzare, mantenendo invariato quindi il tema usato.



`usecolortheme{}`

Permette di definire lo schema di colori da utilizzare, mantenendo invariato quindi il tema usato.

- `albatross;`
Utilizza come colori principali il **giallo** e il **blu**.



`usecolortheme{}`

Permette di definire lo schema di colori da utilizzare, mantenendo invariato quindi il tema usato.

- `albatross;`
Utilizza come colori principali il **giallo** e il **blu**.
- `beetle;`
Testo in **bianco** (pe revidenziare) e **nero** (per scrivere) mentre lo sfondo è **grigio**.



`usecolortheme{}`

Permette di definire lo schema di colori da utilizzare, mantenendo invariato quindi il tema usato.

- `albatross`;
Utilizza come colori principali il **giallo** e il **blu**.
- `beetle`;
Testo in **bianco** (per revidenziare) e **nero** (per scrivere) mentre lo sfondo è **grigio**.
- `dove`;
Questo scheme (simile a `beetle`) ha una resa particolarmente vantaggiosa se usato per presentazioni che andranno stampate in bianco e nero.

I

Ovviamente si può lavorare anche sui fonts:

Esempi

```
\usefonttheme [options] {serif}  
\usefonttheme [professionalfont]
```

Risultati

$$\int e^x dx = e^x$$



Beamer

Le sue componenti principali sono i `frames`

Cosa sono:

Sono le slide vere e proprie, qui dentro ci va tutto:

- Titoli;
- Blocchi;
- Colonne;
- Elenchi;
- File Multimediali;
- ecc...

Sintassi

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



Per cominciare...

...ma non solo...

Titolo e sottotitolo:

```
\frametitle{titolo}  
\framesubtitle{sottotitolo}
```

Sorgenti

```
\begin{verbatim}  
Qualsiasi codice sorgente,  
anche Latex.  
\end{verbatim }
```

N.B. Viene mantenuta la stessa formattazione del file.tex

Per gli elenchi invece:

```
\begin{itemize}  
\item A;  
\item B;  
\item C;  
\end{itemize}
```

La sintassi è dunque la stessa delle classi per la scrittura di testi.



Riquadri:

Per quanto riguarda i blocchi o riquadri, la sintassi è ancora una volta semplice e intuitiva. Questa volta la cosa carina è che di default ci sono tre tipi di riquadri, i quali differiscono gli uni dagli altri dal colore.

1

Blu

2

Verde

3

Rosso

Sintassi

```
\begin{block}{1}
```

Blu

```
\end{block}
```

```
-----  
\begin{exampleblock}{2}
```

Verde

```
\end{exampleblock}
```

```
-----  
\begin{alertblock}{3}
```

Rosso

```
\end{alertblock}
```

NON VI PIACCIONO??



Tranquilli...ovviamente si possono cambiare :)

```
\setbeamercolor{block body example}  
{bg=color, fg=color}
```

```
\setbeamercolor{block title alert}  
{bg=color, fg=color}
```



...

Ovviamente è possibile solo impostare il testo delle slide anche in colonne. Per far ciò è necessario entrare in un ambiente `columns` e disegnare la varie colonne:

Sintassi

```
\begin{columns}  
\begin{column}{0.5\textwidth}  
Testo incolonnato  
\end{column}  
\end{columns}
```

N.B.

Dobbiamo ricordarci di inserire sempre la larghezza della colonna. `textwidth` rappresenta la larghezza dell'intera pagina.



```
begin{frame}[...]
```

`allowframebreaks` spezza in automatico il frame quando comincia ad essere pieno di contenuti;



`begin{frame}[...]`

`allowframebreaks` spezza in automatico il frame quando comincia ad essere pieno di contenuti;

`shrink` restringe il contenuto in modo da farlo stare tutto in una singola slide;



`begin{frame}[...]`

`allowframebreaks` spezza in automatico il frame quando comincia ad essere pieno di contenuti;

`shrink` restringe il contenuto in modo da farlo stare tutto in una singola slide;

`squeeze` squeezing dello spazio verticale;



Ebbene sì, se il vostro lettore PDF lo permette, si possono far vedere video o sentire file audio come qui:

Movie

```
\movie[options]{Image or Text}{File.avi (o .mpg)}
```



Sound

```
\sound[options]{Image or Text}{File.wav}
```



Transizioni e Overlays

Come qualsiasi presentazione che si rispetti, anche qui ci sono le transizioni e gli overlays.

Per le transizioni:

```
\transboxin  
\transboxout  
\transdissolve
```

Per gli overlays:

- `\pause`
- `\begin{itemize}`
 - `\item<1> unoduetrestella`
 - `\item<2-> ambarabacciccicoccò`
 - `\item<1,3> bubusetete``\end{itemize}`
- `\being{itemize} [<+>]`
 - `\item ...`
 - `...``\end{itemize}`

Cosa interessante: [▶ handout](#)



Oltre agli editor generali, come Vi (e GVIM) o Gedit/Kate, esistono editor appositi per il \LaTeX ; ecco qualche esempio :

Kile L'editor \LaTeX del desktop KDE; si lavora sempre sul sorgente `.tex`, ma con un'utile GUI per inserire i vari elementi, utile per chi inizia.

LyX Non un vero editor \LaTeX , nel senso che si può usare quasi come un word processor, con la differenza che tutta la parte di layout è gestita di nascosto da \LaTeX ; è comunque possibile inserire codice dovunque voluto.



Chiaramente è possibile generare automaticamente codice $\text{T}_{\text{E}}\text{X}$ partendo da altre fonti, essendo un linguaggio con una struttura definita.

L'esempio principe : i traduttori

Basta spulciare gli archivi dei pacchetti di Debian/Ubuntu per imbattersi in (e questi sono solo alcuni) :

- `writer2latex`
- `wiki2beamer`
- `mathtex` (per generare immagini)
- `latexml`
- `latex2rtf`
- `latex2html`



Il problema

Non che ormai il problema sia molto rilevante, ma un'installazione di \LaTeX , magari con font extra e pacchetti aggiuntivi, occupa qualche centinaio di M su disco, con più di 100M da scaricare inizialmente.

La soluzione

Esistono siti che permettono di scrivere codice direttamente in un form e di avere in output il pdf, o ps, compilato. Magari non è esattamente comodo usarlo per una tesi di laurea, ma per qualche documento rapido può essere un'alternativa. Un paio si trovano a :

- <http://nirvana.informatik.uni-halle.de/~thuering/php/latex-online/latex.php>
- <http://sciencesoft.at/latex/?lang=en>

Come lo uso nel mio sito?

Dopo questa presentazione, probabilmente nessuno avrà più voglia di scrivere almeno formule matematiche o scientifiche se non in \LaTeX ☺. Allora sarete felici di sapere che esistono plugin che permettono di inserire codice in pagine web; se il server lo supporta, i comandi inseriti vengono compilati al volo ed inseriti come immagine (ad esempio Wikipedia).

Esistono plugin \LaTeX per :

Wiki

- Mediawiki
- Dokuwiki
- ...

Forum

- phpBB
- vBulletin
- ...



Lecture consigliate per iniziare...

 Tobias Oetiker
The not so short introduction to LaTeX
<http://tobi.oetiker.ch/lshort>

 Charles Batts
A Beamer tutorial in Beamer
<http://www.uncg.edu/cmp/reu/summer2007/>

 Francesco Versaci
LaTalk
<http://www.montellug.it>

 Ki-Joo Kim
Beamer v3.0 Guide
http://faq.ktug.or.kr/wiki/uploads/beamer_guide.pdf

