

*P*<sup>o</sup>*W**E**R*

L*A*T*E*X

# Power What? Presentation Techniques with pdfL<sup>A</sup>T<sub>E</sub>X

Patrick W. Daly

MPS

February 23, 2005

# Outline

- 1 Why L<sup>A</sup>T<sub>E</sub>X?
- 2 Requirements for a presentation
  - Content Requirements
  - Font Requirements
  - Dynamic Requirements
  - Processing Aids
- 3 Solutions before pdfL<sup>A</sup>T<sub>E</sub>X
  - SliT<sub>E</sub>X
  - Seminar
- 4 More recent solutions
  - Several classes
  - PPower4 Post-processor
- 5 Beamer

# Why $\LaTeX$ ?

## Good question

Why does one use  $\LaTeX$  in the first place?

- Better mathematics
- More control over the input
- **Personal preference** ✌✌✌

This is not the place to justify  $\LaTeX$ ,  
we assume you know why you want to use it.  
Or why not.

# Why L<sup>A</sup>T<sub>E</sub>X?

## POW<sub>E</sub>R

### PDF instead of PPT

- One produces a PDF file from the L<sup>A</sup>T<sub>E</sub>X source file
- Best using pdfL<sup>A</sup>T<sub>E</sub>X, but also possible
  - dvi → ps → pdf (dvips plus Ghostview or Distiller)
  - dvi → pdf (with dvipdfm)
- Presentation made with Acrobat Reader, preferably in full screen mode

# How is a presentation different from a paper

The visual support for an oral presentation normally consists of a **slide show** projected on to a screen in front of the audience.

The projected material is a mixture of text and diagrams, which may be dynamic.

- 1 Content
- 2 Fonts
- 3 Dynamics
- 4 Processing aids

## Content Requirements

- ✓ Text is reduced to short sentences, or only keywords
- ✗ the audience is not going to read long novels
- ✓ Figures, diagrams, movies should support the oral statements
- ✗ but should not be so overloaded that no one can figure them out
- ✓ Mathematics should be allowed
- ✗ but a page full of equations in small type is **bad style**

All this is a matter for the author, there is no L<sup>A</sup>T<sub>E</sub>X style that can make him/her less verbose.

## Font Requirements

- Size: Much larger than for a normal article
- Family: ideally sans serif, rather than roman fonts with serifs (Computer Modern), or even roman fonts with serifs (Times Roman)
- Colour: must match the background colour, but should not distract

L<sup>A</sup>T<sub>E</sub>X can manage all this, best in prepared classes or packages.

## Dynamic Requirements

Alternatively: e-documents versus paper.

- Overlays: building a page in steps
- Links: being able to jump about with mouse click
- Active diagrams: figures are animated, or a movie is inserted.
- Transitions: vary the way pages are changed.
- Landscape rather than portrait: really only a requirement of computer monitor shape.

These features can be added with pdfL<sup>A</sup>T<sub>E</sub>X.

Special classes (and some programs) make it “*simple*”.

## Value-added Processing Features

While not part of the actual output, the class/package should contains features to assist processing.

- The slides specialties (fonts, papersize, dynamics) should be preprogrammed.
- Processing of selected slides only should be possible (very useful when building up complicated figures).
- Handouts and/or a complete regular article should be available, to be output as options.

# SliT<sub>E</sub>X, or the slides class

- Even back in the old days of L<sup>A</sup>T<sub>E</sub>X 2.09, there was a “style” for making viewgraphs ⇒ *SliT<sub>E</sub>X*
- Because fonts were hardwired in the T<sub>E</sub>X format in those days, SliT<sub>E</sub>X was actually a parallel format to L<sup>A</sup>T<sub>E</sub>X itself; i.e. it acted like a separate program.
- And it had a most complicated method of producing colour overlays for printing on black-and-white printers.

For these reasons, SliT<sub>E</sub>X has not been taken very seriously.

- In modern L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>, the `slides` class replaces SliT<sub>E</sub>X, much improved, with colour management left to the `color` package.

## The slides class

- makes use of a special set of sans serif fonts that are considerably larger than the regular ones
- the letters are also proportioned differently: compare  
The Family (slides)  
The Family (Computer Modern Sans)  
The Family (Helvetica).
- supports notes and overlays, but of the kind for true viewgraphs
- can output selected slides, a big help during production.

I added some additional features like running heads/footlines, logo.  
As a sample of a talk I gave in London in 2001 ...



Annales Geophysicae (2001) 19: 1–12 © European Geophysical Society 2001



## First results from the RAPID imaging energetic particle spectrometer on board Cluster

B. Wilken<sup>1,\*</sup>, P. W. Daly<sup>1</sup>, U. Mall<sup>1</sup>, K. Aarsnes<sup>2</sup>, D. N. Baker<sup>3</sup>, R. D. Belian<sup>4</sup>, J. B. Blake<sup>5</sup>, H. Borg<sup>6</sup>, J. Büchner<sup>1</sup>, M. Carter<sup>7</sup>, J. F. Fennell<sup>8</sup>, R. Friedel<sup>4</sup>, T. A. Fritz<sup>8</sup>, F. Gliem<sup>9</sup>, M. Grande<sup>7</sup>, K. Keckemety<sup>10</sup>, G. Kettmann<sup>1</sup>, A. Korth<sup>1</sup>, S. Livi<sup>1</sup>, S. McKenna-Lawlor<sup>11</sup>, K. Mursula<sup>12</sup>, B. Nikutowski<sup>1</sup>, C. H. Perry<sup>7</sup>, Z. Y. Pu<sup>13</sup>, J. Roeder<sup>5</sup>, G. D. Reeves<sup>4</sup>, E. T. Sarris<sup>14</sup>, I. Sandahl<sup>15</sup>, F. Søråas<sup>2</sup>, J. Woch<sup>1</sup>, and Q.-G. Zong<sup>1</sup>

<sup>1</sup>Max-Planck-Institut für Aeronomie, Katlenburg-Lindau, D-37191 Katlenburg Lindau, Germany

<sup>2</sup>University of Bergen, Allg. 55, 5007 Bergen-U, Norway

<sup>3</sup>LASP, Boulder-CO 80309, USA

<sup>4</sup>LANL, Los Alamos-NM 87545, USA

<sup>5</sup>Aerospace Corporation, Los Angeles-CA 90009, USA

<sup>6</sup>IRF, 90187 Umea, Sweden

<sup>7</sup>RAL, Chilton, Didcot, Oxfordshire OX110QX, UK

<sup>8</sup>Boston University, Boston-MA 02215, USA

<sup>9</sup>IDA, D-38106 Braunschweig, Germany

<sup>10</sup>KFKI, H-1525 Budapest-114, Hungary

<sup>11</sup>NUI, Maynooth-Co., Kildare, Ireland

<sup>12</sup>University of Oulu, 90571 Oulu, Finland

<sup>13</sup>Peking University, Beijing 100871, China

<sup>14</sup>University of Thrace, Xanthi, Greece

<sup>15</sup>IRF,S-98128 Kiruna-C, Sweden

\*The RAPID team deeply regrets the untimely demise of B. Wilken, PI of the Cluster project. Without him, the RAPID instrument would never have been created



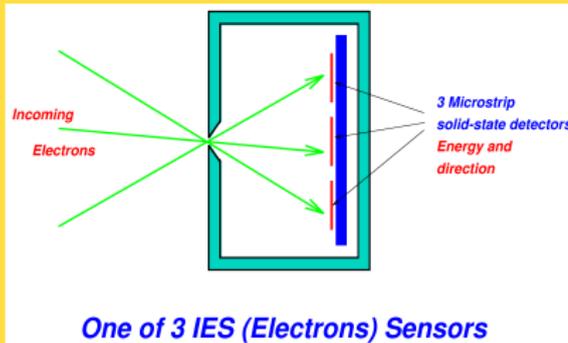
## RAPID (*R*esearch with *A*daptive *P*article *I*maging *D*etectors)



is the energetic particle spectrometer on board Cluster is an advanced particle detector for the analysis of suprathermal plasma distributions in the energy range from 20–400 keV for electrons, 40 keV–1500 keV for hydrogen, and 10 keV/nucleon–1500 keV for heavier ions.



## Electron (IES) Detectors

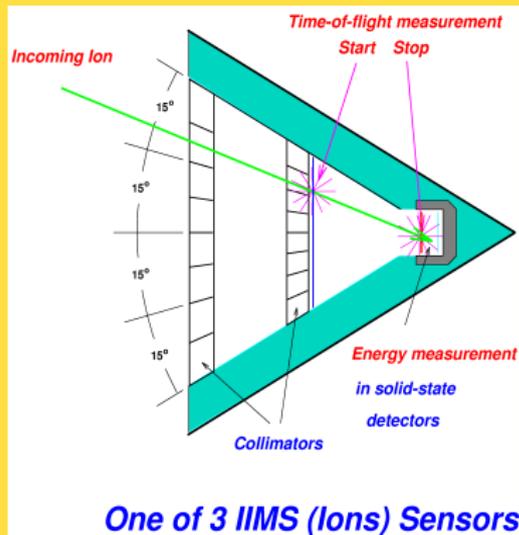


- There are 3 electron detector heads, each covering  $60^\circ$  in plane of spin axis;
- The heads function as a pin-hole camera, each having 3 separate detectors for a fine resolution of  $20^\circ$ ;
- Four read-out ('integration') times available: 2, 5, 15, 50  $\mu\text{s}$ .



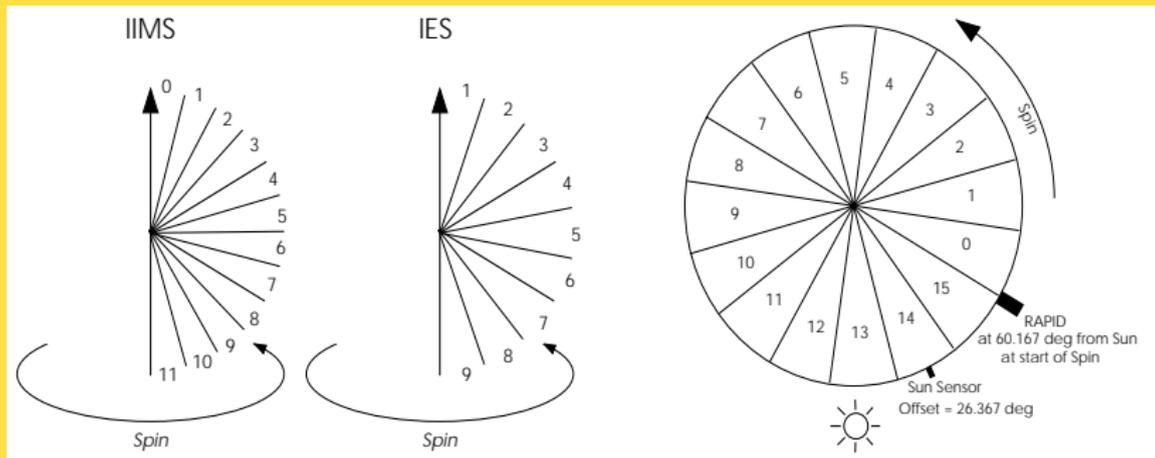
## Ion (IIMS) Detectors

- There are 3 ion detector heads, each covering  $60^\circ$  in plane of spin axis;
- Time-of-flight mass determination: start signal:  $e^-$  from penetrated foil; stop:  $e^-$  from surface of solid-state detector;
- TOF distance is 34 mm; time resolution 80 ns/256;
- The start signal also serves to indicate incoming direction; fine resolution is  $15^\circ$ .





## Angular Coverage in 3-D



**Note:** the spin axis is directed towards the *southern* ecliptic pole!

# The seminar class

The `seminar` class by *Timothy van Zandt* originated in the L<sup>A</sup>T<sub>E</sub>X2.09 era, but has been upgraded for L<sup>A</sup>T<sub>E</sub>X2<sub>ε</sub>. Together with pdfL<sup>A</sup>T<sub>E</sub>X and the `hyperref` package, produces acceptable results.

- has all the features of `slides`
- has no special fonts, but scales all output by a factor 2
- can mix landscape and portrait slides
- includes framed text, if wanted, plus header/footerlines
- background image can be added
- can output handout version for distribution

A sample talk that I gave last year in Kiel, with `seminar` and the PPower4 post-processor for overlay effects ...



The energetic particle spectrometer  
*RAPID* on board Cluster—  
A three-year overview

Patrick W. Daly

Max-Planck-Institut für Aeronomie  
Katlenburg-Lindau

## RAPID Particle Spectrometer



### RAPID

- ▶ stands for *Research with Adaptive Particle Imaging Detectors*
- ▶ is one of 11 experiments on board the 4 Cluster satellites
- ▶ is an energetic ion and electron ( $E > 30$  keV) imaging spectrometer.

## RAPID Particle Spectrometer

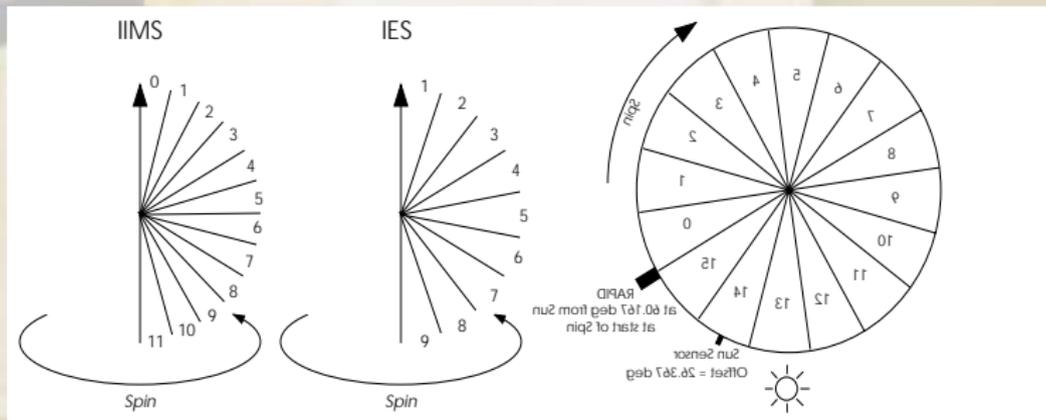
*RAPID* actually consists of two sets of spectrometers:

- ▶ one for ions
- ▶ one for electrons.

Each set contains three units, each covering  $60^\circ$ , for a total of  $180^\circ$  in one plane.



## Angular Coverage in 3-D



**Note:** the spin axis is directed towards the *southern* ecliptic pole!

# Some newer classes

A great deal of work is being done on this issue, and many solutions exist. Some of these are:

- FoilT<sub>E</sub>X by Jim Hafner of IBM (non-free)
- Prosper by Frédéric Goualard, based on [seminar](#) and [PSTricks](#)
- HA-Prosper by Hendri Andriaens, based on [prosper](#)
- pdfscreen by C. V. Radhakrishnan, really for e-docs for screen viewing
- T<sub>E</sub>XPower by Stephan Lehmké and Hans Fr. Nordhaug
- Beamer by Till Tantau, very powerful and ambitious collection.

# Overlays in PDF

The overlay feature is the most important dynamic property added to the PDF file.

- With it, lists are build up successively, but also figures can be embellished with arrows, indicators, or alternatives.
- In a PDF file, this is done by breaking one page up into several.
- The previous classes try to do this by generating the one page several times, with and without the varying text.
- This requires some tricky internal programming, to get the missing text participating in the positioning calculations

# PPower4

An alternative is the PPower4 Java script by Klaus Guntermann and Christian Spannagel.

Method:

- 1 Produce the L<sup>A</sup>T<sub>E</sub>X file with the special `pause` package.
- 2 Add the command `\pause` wherever a break is to occur
- 3 Generate the PDF output with pdfL<sup>A</sup>T<sub>E</sub>X.
- 4 Run the Java script on that file, to produce a new PDF file with page breaks.

This produces the following output:-

In this example, the page number should have been present all the time.

It appears only on the last view because it is set after the last `\pause` statement.

To control this, one can set the level number explicitly.

This results in:

L<sup>A</sup>T<sub>E</sub>X

The `\pauselevel` command can be used to create many special effects:

- `\pauselevel{=n}` to set the level number absolutely
- `\pauselevel{=+n}` to increase the level number
- `\pauselevel{=-n}` to decrease the level number
- `\pauselevel{=n -d}` to set level number and the `\pause` incremental step
- `\pauselevel{:m}` to set maximum level for following text

This example was produced with code:

```
% Left hand minipage with text
\begin{minipage}[c]{0.4\textwidth}
RAPID actually consists of two sets of spectrometers:\pause
\begin{itemize}
  \item one for ions \pause
  \item one for electrons.\pause
\end{itemize}
Each set contains three units, each covering 60\deg, for a
total of 180\deg\ in one plane.
\end{minipage}\quad
```

```
% Right hand minipage with figure
\parbox[c]{0.5\textwidth}{%
\setlength{\unitlength}{0.01\linewidth}
\begin{picture}(100,72)
  \pause\pauselevel{=1}%
  \put(0,0){\includegraphics[width=\linewidth]{rapid}}
  \pause\pauselevel{:+0}
  \put(15,57){\color{Green}\line(-1, 0){58}}
  \put(10,40){\color{Green}\line(-4, 1){53}}
  \put(18,10){\color{Green}\line(-3, 2){61}}
  \pause\pauselevel{:+0}
  \put(40,09){\color{Red}\line(-5, 2){65}}
  \put(40,22){\color{Red}\line(-4, 1){65}}
  \put(40,32){\color{Red}\line(-6, 1){65}}
\end{picture}}
\pause\pauselevel{=1}
```

## Transitions

are the way one page is replaced by another.

- These are a basic feature of the pdfT<sub>E</sub>X program.
- But more (L<sup>A</sup>T<sub>E</sub>X) user-friendly commands are available
  - in the `hyperref` package
  - in the `pagetrans.tex` file (supplied with P<sup>P</sup>ower4)
  - and with `\pause`*trans* commands in the `pause` package

L<sup>A</sup>T<sub>E</sub>X

# Pros and Cons

## Advantages and disadvantages of a Post-Processor

### Advantages

- Works with any L<sup>A</sup>T<sub>E</sub>X class
- Not dependent on L<sup>A</sup>T<sub>E</sub>X reprogramming
- Post-processor is a dumb program, it just does its job
- The author's interaction is entirely in the L<sup>A</sup>T<sub>E</sub>X file

### Disadvantages

- An extra processing step is needed, using an extra program
- which needs to be installed (*I had no problem whatever*)
- Complicated arrangements with `\pauselevel` commands can become confusing (*One can define better commands*)

# Pros and Cons

POWER

## Conclusion

- P<sup>P</sup>ower4 is an excellent tool for adding overlay and transition effects to any L<sup>A</sup>T<sub>E</sub>X document.
- But if a class like [Beamer](#) can do that just as well, then one may have to reconsider.

L<sup>A</sup>T<sub>E</sub>X

# The Beamer Class

## Beamer

- **Beamer** is a L<sup>A</sup>T<sub>E</sub>X class and collection of packages put together by **Till Tantau** in Berlin.
- It started as an extension to **seminar**, worked on privately, with suggestions and criticism from friends.
- In Feb 2003, he used the first version for his PhD defense presentation
- and made it public on CTAN a month later.
- (Which explains why I missed it when preparing the 4th edition of the *Guide to L<sup>A</sup>T<sub>E</sub>X*.)

## Main features

- Very much object oriented programming  
Things like sidebars, header, footlines, navigation bars can be separately turned on/off, with various options, or reprogrammed at a lower level, but also with *objects*.
- Many useful and attractive *themes* are preprogrammed.  
Themes are named after cities; this one is **Ilmenau**, a variation on **Berlin**.
- All requirements are fulfilled: large fonts, overlaying, transitions, backgrounds, selective processing
- and different outputs:
  - beamer** for projection, with all the overlays
  - transparency** for viewgraphs, fewer overlays
  - handout** for paper, two slides per page
  - article** a regular L<sup>A</sup>T<sub>E</sub>X article



## Main features

- ▶ Very much object oriented programming  
Things like sidebars, header, footlines, navigation bars can be separately turned on/off, with various options, or reprogrammed at a lower level, but also with *objects*.
- ▶ Many useful and attractive *themes* are preprogrammed. Themes are named after cities; this one is *Ilmenau*, a variation on *Berlin*.
- ▶ All requirements are fulfilled: large fonts, overlaying, transitions, backgrounds, selective processing
- ▶ and different outputs:
  - beamer* for projection, with all the overlays
  - transparency* for viewgraphs, fewer overlays
  - handout* for paper, two slides per page
  - article* a regular  $\LaTeX$  article

[Outline](#)[Why  \$\LaTeX\$ ?](#)[Presentation needs](#)[Pre-pdf \$\TeX\$](#) [Recent solutions](#)[Beamer](#)

## Main features

- Very much object oriented programming  
Things like sidebars, header, footlines, navigation bars can be separately turned on/off, with various options, or reprogrammed at a lower level, but also with *objects*.
- Many useful and attractive *themes* are preprogrammed. Themes are named after cities; this one is **Ilmenau**, a variation on **Berlin**.
- All requirements are fulfilled: large fonts, overlaying, transitions, backgrounds, selective processing
- and different outputs:
  - beamer** for projection, with all the overlays
  - transparency** for viewgraphs, fewer overlays
  - handout** for paper, two slides per page
  - article** a regular  $\LaTeX$  article

## Main features

- ▶ Very much object oriented programming  
Things like sidebars, header, footlines, navigation bars can be separately turned on/off, with various options, or reprogrammed at a lower level, but also with *objects*.
- ▶ Many useful and attractive *themes* are preprogrammed.  
Themes are named after cities; this one is **Ilmenau**, a variation on **Berlin**.
- ▶ All requirements are fulfilled: large fonts, overlaying, transitions, backgrounds, selective processing
- ▶ and different outputs:
  - beamer** for projection, with all the overlays
  - transparency** for viewgraphs, fewer overlays
  - handout** for paper, two slides per page
  - article** a regular  $\LaTeX$  article

## Main features

- Very much object oriented programming  
Things like sidebars, header, footlines, navigation bars can be separately turned on/off, with various options, or reprogrammed at a lower level, but also with *objects*.
- Many useful and attractive *themes* are preprogrammed. Themes are named after cities; this one is *Ilmenau*, a variation on *Berlin*.
- All requirements are fulfilled: large fonts, overlaying, transitions, backgrounds, selective processing
- and different outputs:
  - beamer for projection, with all the overlays
  - transparency for viewgraphs, fewer overlays
  - handout for paper, two slides per page
  - article a regular L<sup>A</sup>T<sub>E</sub>X article

There are alternate ways to uncover a list:

- 1 This is the first
  - 2 This is the second
  - 3 This is the absolutely most important thing
- POWER*
- 1 This is the first
  - 2 This is the second
  - 3 This is the absolutely most important thing
- LATEX*
- 1 This is the first
  - 2 This is the second
  - 3 This is the absolutely most important thing

Specifying overlays is reasonably straight-forward.

- `\uncover<3>{Text for level 3 only}`
- `\uncover<3->{Text for level 3 and after}`
- `\uncover<+>{Text for next level and after}`
- `\pause`, works as with PPower4, like `\uncover<+>{}`
- `\only<3->{}`, like `\uncover` but text removed when not visible
- `\item<+>` for automatic incrementing in a list
- and the increment can even be made to be a default with
 

```
\begin{itemize}[<+>]
  \item ...
\end{itemize}
```

DOWN-D

And most important ...

You can redesign your own themes as you wish!

If you wish!

Which would make you a Beamer ...

L<sup>A</sup>T<sub>E</sub>X

And most im  
You can rede  
If you  
Which would

