Using the STScI Viewgraph Style

Abstract

The use of a LaTeX document style for the preparation of viewgraphs at STScI is described. Some information that may be helpful for people who want to design their own viewgraph format is also included, but no promises are made.

C. D. Biemesderfer

Space Telescope Science Institute • Baltimore, Maryland

Contents

Introduction

There is a document style option available for LaTeX users at the Institute that enables you to prepare viewgraphs in the comfort of your own office. Macros that extend plain TeX are also under development and should be ready Real Soon Now, so stay tuned.

I have endeavored to layer this style on top of vanilla LaTeX as much as possible in order that the usual array of handy environments is available for producing itemized and enumerated lists, displayed equations, and so forth. I have largely been successful at that, I think, but there are plenty of bugs in this, and the reason I am releasing this now is that folks are starting to copy "test" versions of my files that are in intermediate (and not always working) condition. This can be very confusing for me, and it is obviously better to have everyone use the same file so I know where to start when the complaints come rolling in (interspersed, hopefully, with some positive feedback as well).

Please feel free to exercise all the advertised capabilities and report on the things that don't work properly. Do not hesitate to send along suggestions for improvements and extensions, although you have to realize that I may take a while to implement complicated ones. It is preferable to receive communications in a filable form, and VAX Mail is best for this; please note that my username is BIEMES. Have fun.

1 What Is A Viewgraph ?

A 'viewgraph' is a 'page' as far as LaTeX is concerned, so pagebreaking commands can be used to end a slide and get on with the next one. At the moment, there are five page 'styles', and they have to do with viewgraphs not text documents, so don't just assume that the regular LaTeX page styles will be there. Most of the time, you will probably use the default pagestyle, which is called 'sthead'; the others are 'empty', 'plain', 'noaohead', and 'overlay'. They are covered in more detail in §3.2.

2 What You Cannot Do on Viewgraphs

The ST viewgraph style is layered on top of regular LaTeX, and you should be aware of some things that are a function of the document style, and that I have chosen not to implement for viewgraphs.

Sections. Forget sections entirely when you are making slides, and do not try to use commands like `\section` or `\paragraph`. They are not defined.

Marginal notes. The margins of a viewgraph are off the glass of the projector, so don't waste your (and my) time with these. They are not completely disabled, but it could be arranged.

Footnotes. These are not disabled, either, but it seems to me to be rather stupid to me to put teeny-weeny text on a viewgraph that somebody with poor eyesight in the back row is supposed to see on a projection screen (that is out of focus).

Twosided substyle. Do I need to explain this ?

Front and back matter. That's tables of contents, lists of figures, glossaries, appendices, etc.

There are other things are not such a great idea, either; for instance, you shouldn't need things like the 'figure' and 'table' environments to define floats. Not that you shouldn't use 'tabular' to make tables, but you should just have it appear on the slide wherever you typeset it. In general, you should be able to put a `\nofiles` command right after `\documentstyle` because there shouldn't ever need to be any auxiliary files. I can imagine why you might want to use `\label` and `\ref` on viewgraphs, but I don't think this is good idea.

3 What You Can Do on Viewgraphs

Everything else you might expect should work properly. The list making environments ('itemize', 'enumerate', 'description') are probably the most useful, and other environments like 'center', 'quote', and 'verse' can be used also. I think that 'flushleft' and 'flushright' will probably give you trouble, and I know that the 'verbatim' environment misbehaves. The trouble with these is with the margins; vanilla LaTeX makes some assumptions about margin parameters that I use to pull the sides of the page away from the true width of the text (so that words don't mash into the frame), and I have to redefine certain LaTeX macros to solve this. The problem at the present time is finding the right ones.

3.1 Substyles

There are only four substyles (those are the arguments to `\documentstyle` that go in the square brackets) at the moment. Three of them are for the various default type sizes, and the fourth one turns on the frame around the whole slide.

18pt This is the default typesize, so you don't have to specify '18pt' as a substyle, but you can if you like.

20pt This one gives you somewhat larger type, which is probably what you expected.

24pt This is the largest size for which we have a complete set of fonts.

frame Specify this substyle if you want frames around the whole page. By default, only the header box is framed. You can change your mind about framing in the document by using `\framepages` and `\unframepages` commands.

Consider the implications of my statement above about the 24pt fonts. Since that is the largest "set" of fonts (Roman, math, slanted, italic, typewriter, san serif, and small caps faces make up a set), and the "normal" size for viewgraph type is quite large, you will find that the type size changing commands of LaTeX won't always do something when you're trying to get bigger type. The commands that select smaller type sizes should be fine.

3.2 Page styles

The default pagestyle for viewgraphs assembles a header for the transparency consisting of the Institute name and several pieces of information that you can supply for the right-hand box (see §3.4). You can have the entire viewgraph enclosed in a frame (see §3.3). This style is called 'sthead', although it is not necessary to include a `\pagestyle` command to use it.

There is an 'noaohead' pagestyle which is similar to the 'sthead' style, but there is no "right-hand box" in the NOAO header. It just says "National Optical Astronomy Observatories" across the top of the slide. The people at NOAO do not care much for this style.

Two of the "regular" LaTeX pagestyles are implemented for those of you who like to keep things simple (like the people at NOAO). These are 'empty' and 'plain', and they work like you would expect. The 'empty' style doesn't add anything to the page besides what you type, while the 'plain' style adds the page (slide) number in the lower right-hand corner of the viewgraph.

The last style option for viewgraphs is 'overlay', which will not put any borders or extraneous markings on the page. The page size parameters for the overlay style are tuned so that the meeting, date, and presenter information wind up in the appropriate locations on the page so that your LaTeXed output can be photocopied onto the Institute's blank overhead transparency forms. (By blank, I mean that there are not 'Date' and 'Presenter' words in the upper right-hand box.) If the registration of the pages isn't exactly what you like, you will have to twiddle page size parameters yourself.

Please note that, while in principle you can toggle pagestyles, you will probably find that the frames are not the same size when you switch between pagestyles with headers and pagestyles without headers. I am trying to tune

the sizes and the various macros involved so that this works smoothly, but it's very low priority because it seems to me that you should pick a style for a presentation and then stay with it. But I am working on it.

3.3 Framing the Page

You can get the whole viewgraph enclosed in a big frame by either selecting the 'frame' substyle or by putting a `\framepages` command in your text. If you want to disable page framing, there is an `\unframepages` command. These commands take effect on the page on which they are invoked.

3.4 Putting Stuff in the Little Box on the Right Side

Yes, this is the important part, isn't it ? There are three very short lines in the right-hand box in the header; the first one is for the name of the meeting, the middle one is for the date, and the bottom one is for the presenter. You can supply text for these lines with the commands `\setmeeting`, `\setdate`, and `\setpresenter`. These each take one argument, that being the text for the particular line; for instance,

```
\setmeeting{STOPAT}
```

could be issued so that the acronym STOPAT appears at the left side of the top line in the little right-hand box. (Observe that the page number, or slide number, appears at the right side of this line, and at the moment I don't have a way for you to suppress that.) If you do not set the date, today's date is used; if you don't set the meeting or presenter, those items are left blank. No additional text is supplied by these commands, so in principle, you could put whatever you wanted on these very short lines in the little right-hand box. For example,

```
\setpresenter{I $\heartsuit$ SDAS}
```

would produce "I ♥ SDAS" on the bottom line of the little right-hand box.

By now, you have probably noticed that I am somewhat fanatical about the smallish size of the right-hand box. It is small because the Institute name is very long and takes up about 70% of the available text width, so you are left with a little less than 2 inches to play with. If you put a longish bit of text on one of those lines (say, for instance, the presenter's name was something like Christopher D. Biemesderfer), you could very easily wind up with a header that is wider than the page width. I have arranged it so that such a header can be constructed without an overfull hbox, but unfortunately TEX has its own ideas about what do in this situation, and will probably mess up the alignment of the header and the box containing the text of your viewgraph. I am trying to solve this, but it is quite tricky; as long as you are terse, everything will be OK anyway.

3.5 What About the ST Logo ?

You cannot simply pull the ST logo into the viewgraph header because it is generated with the NCAR graphics software, which is very independent of TEX. It is possible to leave some extra space on the left side of the Institute name by using the `\stlogospace` command. This command also takes one argument, and that is the horizontal dimension of the white space you would like added before the word "SPACE".

```
\stlogospace{.65in}
```

would add about of an inch of blank space to the left of the Institute name. Please observe that this makes the left-hand box even bigger, so you have to be that much briefer in the right-hand box.

But, you say, leaving space is not the same thing as putting the logo in there. That is quite correct, and you have to get the logo on the page by "overlaying" the graphics created by NCAR on each viewgraph. Since TEX and NCAR know nothing about one another, it is completely up to you to make sure that the logo that NCAR makes lands in the space that LATEX creates. This is sort of like playing pin the tail on the donkey when you have a broken leg and are drunk. You also have to make the graphics overlay, which is itself a non-trivial exercise.

In short, I would suggest that you not bother with this. The logo might look cute, but overlaying graphics material is the wrong way to get graphics merged into TEX documents, and hardly worth the pain and suffering to get the alignment correct (scissors and glue are far superior). The appropriate mechanisms (a discussion of which is too long and boring to include here) for doing this kind of integration are well known to the TEX cognoscenti, and we are trying to deal with the problem. A proper solution will eventually be available.

For those of you who insist on the presence of the little picture, you can use the 'overlay' pagestyle that disables the page frame as well as the automatically generated ST header. You must then photocopy the output from LATEX onto a blank ST viewgraph form.

3.6 Shut Up and Give an Example

```
%  Select viewgraph document style and frame
substyle so that pages get
%  surrounded by a big box.
%
\documentstyle[frame]{viewgraph}
```

```
%
%   Turn off auxiliary output files.   LaTeX always
says "no file <file>.aux",
%   but that's what you asked for, so it's OK.
%
\nofiles
%
%   That's all you need in the preamble.
%
\begin{document}

\begin{center}
\bf A Sample Viewgraph
\end{center}

Itemized components for the first slide~:
\begin{itemize}
   \item speed
   \item accuracy
   \item portability
   \item nonsense
\end{itemize}

\clearpage        % end of the first slide

\setmeeting{RSDP ICD PDR}         % acronyms don't
have to make sense

Enumerated components for the second slide~:
\begin{enumerate}
   \item speed
   \item accuracy
   \item portability
   \item more nonsense
\end{enumerate}

In order to demonstrate this new style more
completely, let's put some
```

```
sentences here to show that you can also include
running text.  You probably
want to stay away from too much of this though,
because if there's lots of
text on the screen, people will get eyestrain trying
to read it, and they
won't listen to a word you're saying.

\clearpage        % that's it for the second slide

\unframepages    % turn off the page frame
\setpresenter{Charlie the Tasteful Tuna}
\stlogospace .6in % leave white space and try to put
a logo in it because
                      % you don't believe me that
it's hard

{\large Since the default type size is used for this
set of viewgraphs
(and there are bigger ones), the \verb"\large"
command will select a
larger size.}

{\scriptsize This will be pretty small and therefore
hard for your
audience to read.  This is where you put
conclusions you are not sure about.}

\end{document}  % enough of this silliness
```

4 Making Your Own Headers

Some people are never satisfied. If you want to make up a header for viewgraphs that is impossible within the constraints of my macros, you can create your own substyle file to define an alternate pagestyle. If you do not know what a substyle file is, or the concept of an alternate page style is foreign to your mind, please don’t read the rest of this section; it will only make things worse.

As you may know, after LaTeX has finished reading substyle files and executing the correct `\ds@...` commands for defined document substyles, it will try to resolve outstanding substyle requests by reading files (named *substyle*.STY) in the local (default) directory. The easiest way to implement alternate headers is to define a new pagestyle (a `\ps@...` command) and place the custom `\ps@...` description (plus whatever other specialized things are wanted) in a .STY file in your TeX directory. A few remarks about these definitions are in order.

Since the viewgraph document style doesn't support twosided printing, you can save one level of macro replacement by defining `\@thehead` and `\@thefoot` directly (you can skip `\@oddhead` and `\@oddfoot`, in other words). It is unbelievably important that when you define the header and footer macros, you leave absolutely no spaces in the definition. I mean none at all, and that means you have to put a '%' on the end of every line so TeX doesn't interpret the CR-LF as a word break. If you leave spaces in the macros anywhere (other than `\hspace` and `\hfil` and white space you want as part of the header), the header and footer spacing will be completely screwed up.

You will probably need to look at the comments in the VIEWGRAPH.DOC file, because you might want to override the framing boolean or something like that. However, please note that the definitions I have written for the default Institute header have a lot more jazz going on in them than is required for most applications. If you would like to see an example of a special pagestyle (header), you would be better off examining one that I made up for the SDAS workshop. Ask me where it is hiding.