TeX as presentation tool—an introduction to the ConTeXt presentation environments*

Hans Hagen
Pragma ADE
pragma@wxs.nl

Abstract

In this article I will introduce a few styles I wrote on behalf of presentations. These styles are part of the ConTeXt distribution and can serve as an example of defining layouts in this macro package. More details can be found in the documented styles.

Introduction

Typesetting presentations is not the first thing that comes to mind when one reads books about TeX. Nevertheless I hope to convince the reader that TeX can produce wonderful presentations. In this article I will introduce six of the presentation environments that come with ConTeXt. Although each was written for a specific situation, they are general enough to be used in other situations as well. They also serve as an example of how to define styles in ConTeXt. These styles can best be used along with pdfTeX.

The original style

This style was used first at TUG '1997 in sunny San Francisco. Because a lot of information was presented, this style had quite a few navigational gimmicks. The look and feel was derived from the (interactive) module documentation styles.

This style is rendered in soft (sunny) yellow, blue, red and black. One might code such a presentation as follows:

```
\usemodule[pre-original]

\starttext
 \TitlePage {A simple presentation}
 \Topics {Today's talk}
 \Topic  {First item} ... some text ...
 \Topic {Second item} ... some text ...
 \Subject {A subitem}  ... some text ...
 \Subject {Another subitem} ... text ...
  ...
\stoptext
```

Because chapters and sections don't make sense in presentations, we use more meaningfull sectioning commands. The `\Topic` commands generates a list of subjects (see second screen) and the topics themselves are visible in the margin. Long entries are automatically truncated. At the left bottom there is a status bar, at the right there are navigational buttons.

The green style

In 1998 I did several presentations on interactive documents and screen-document design. For that purpose I needed a simple style, which avoided the typical TeX look. Because I had METAPOST support at hand, I decided to use it for drawing the graphics (button shapes) that are part of the style.

In this style, lines, button shapes and bullets are dark green on a gray background. This style provides a lot of room for text. There is an optional extra button with the logical reference `Whatever`. One can use `\definereference` to assign a meaning to this reference if needed. I once used it to activate a movie and to launch programs.

The funny style

Because I was doing a few talks at TUG '1998 and was rather short on time, I decided to write an additional quick-and-dirty presentation style, which afterwards happened to be a rather nice one. Watch the darker segment within the dark red border: it moves with the incrementing page numbers. Of course this shape is generated by METAPOST.

In this style all non textual elements come in dark shades of red. The page counter is hardly visible in gray scales, but it is definitely there.

The colorful style

I wrote this style while preparing a tutorial for the UKTUG users group. Like the previous ones, it's screaming METAPOST again. I use primary colors

* This article was first published in the *Minutes and Appendices* of the NTG (Nederlandstalige TeX Gebruikersgroep), Volume 23.

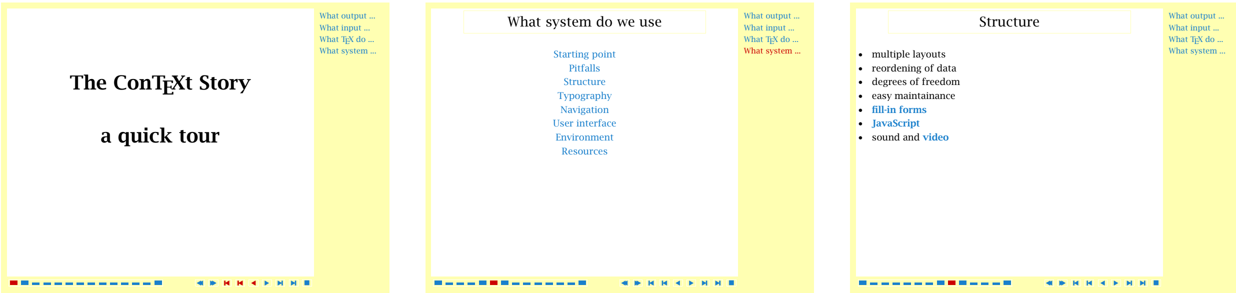

Figure 1: The original style.

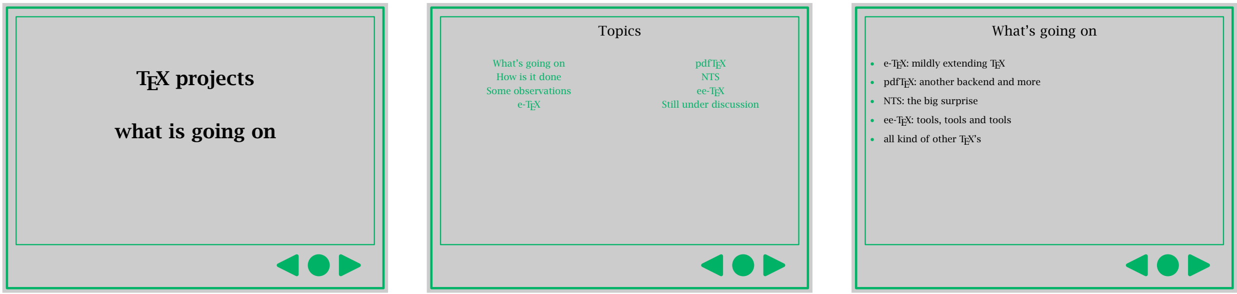

Figure 2: The green style.

Figure 3: The funny style.

only and the title page is a wink to the Dutch government which at the time I wrote this style was spending a huge amount of money on buying a Mondriaan painting (which was criticized in the press).

Like the green style, this one provides a lot of room for text. The list of topics is automatically typeset in columns when it becomes too large to comfortably fit in one column. The status bar at the bottom tells the audience how many screens there are left.

The fuzzy style

The fuzzy style gets its name from the fact that each rectangle used in it is slightly different. I needed this style at a presentation for publishers but decided to use a similar style for the new interactive ConTeXt manual. Although TeX can do the job, I used METAPOST, which I happen to like more and more.

This style is in light blue on light gray. Each presentation will look slightly different, due to the random lines. The list of topics is shown on the right. Make sure to use short topic titles and similar items. Hyperlinks and alike are shown in the same blue color.

The Polish style

One cannot go to Bachotek, the GUST paradise, and do a presentation without coming up with a PostScript trick. This style was written in 1999 and uses the Antikwa Torunska, a font characterized by its prominent backward slant. In the background of the screen we see several O's, slightly distorted by METAPOST in a random way. This style should be

Figure 4: The colorful style.

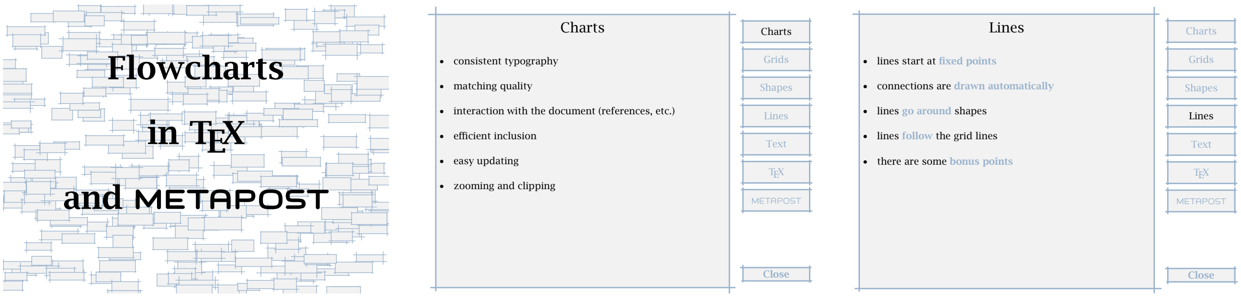

Figure 5: The fuzzy style.

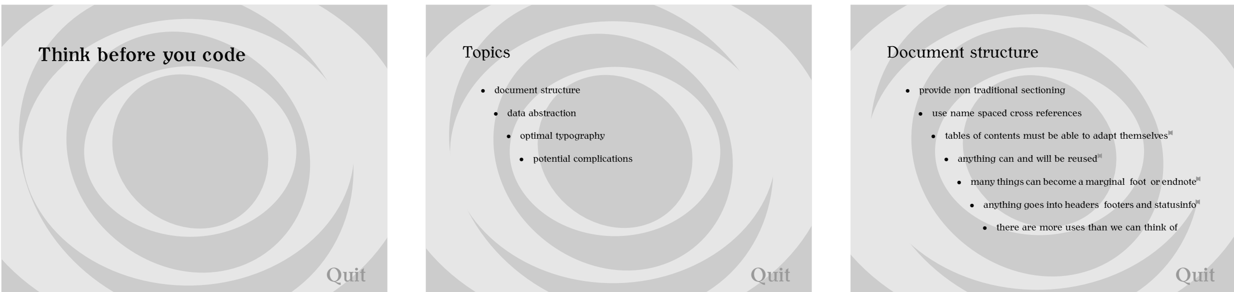

Figure 6: The Polish style.

used with itemized lists: they follow the background slant.

This style has no buttons, apart from **Quit**. Clicking on the page moves you one page forward. This style is in grayscale.

More styles

Because I don't want to repeat myself too often, some more styles are under construction. These styles will be made public when I've used them a few times myself and when they are properly documented.

Commands

A presentation is set up with a minimum of commands. The title page is generated with:

```
\TitlePage {title}
```

One can use `\\` to force a new line or start a subtitle. In case one wants a more complicated title page, there is `\StartTitlePage...\StopTitlePage`.

In a presentation we often want to start with a summary. Such a summary is generated with:

```
\Topics{title}
```

Sometimes it makes sense to use the summary as title page. The summary lists the topics as defined by:

```
\Topic{title} ... text ...
```

In most cases, the text is an itemized list or a few text lines. The less text is used, the better it often suits the presentation. In principle one can use all ConTEXt commands here.

module name	file name
pre-original	s-pre-01
pre-green	s-pre-02
pre-funny	s-pre-03
pre-colorful	s-pre-04
pre-fuzzy	s-pre-05
pre-polish	s-pre-06

Table 1: The module names.

In a large presentation it makes sense to divide the topics into subjects using:

```
\Subject{title}
```

Depending on the style, the `\Topic` command generates a list of subjects.

Loading a style

Being part of the ConTeXt distribution these styles follow the module naming scheme, with filenames such as `s-pre-01` and alike. For clarity we also provide more verbose names as shown table **??**.

For instance, the 5th presentation style is loaded with:

```
\usemodule[pre-fuzzy]
```

This happens to mean the same as `\usemodule [pre-05]`; when loading a module we omit the leading `s-`.

Each module is documented and the typeset source is available as reference. The styles are not that complicated and the documentation takes more lines than the actual code. Although changing those styles is not that hard, users are strongly advised not to. It is far better to change layout characteristics in the document source.

Fonts

The styles use a 14.4pt body font. Some styles use the Lucida Bright fonts because they look very good on a computer screen. When these fonts are not available on the system, it makes sense to map them onto another font family. One way of doing this is adding the next line to the local `cont-sys.tex` file:

```
\definefilesynonym [font-lbr] [font-pos]
```

That way ConTeXt will use the three PostScript fonts Times, Helvetica, and Courier, instead of Lucida Bright fonts. The three PostScript fonts mentioned are nearly always available on the system.

Running METAPOST

Most of these styles rely on METAPOST. There are three ways to make sure that the graphics are generated:

number	transition effects
1 2	`{split,in,vertical}` `{split,in,horizontal}`
3 4	`{split,out,vertical}` `{split,out,horizontal}`
5 6	`{blinds,horizontal}` `{blinds,vertical}`
7 8	`{box,in}` `{box,out}`
9 10 11 12	`{wipe,east}` `{wipe,west}` `{wipe,north}` `{wipe,south}`
13	`{dissolve}`
14 15	`{glitter,east}` `{glitter,south}`

Table 2: The transition effects.

- automatically run METAPOST at run-time
- let `texexec` take care of processing the graphics
- process the graphics manually between runs

The first alternative is the one I prefer. To enable this feature, one has to add `\runMPgraphicstrue` to the local `cont-sys.tex` file as well as to enable `\write18` in the (users) `texmf.cnf` file.

When `texexec` is properly set up, it will take care of generating the graphics, unless run time generation is enabled.

When both options are out of the question, the final alternative is to generate the graphics by hand. Just say: `mpost mpgraph` between runs, and the graphics will be there.

Graphics are stored in files with the name `mpgraph`. You can safely remove these files afterwards.

Page transitions

Fancy page transitions only make sense in presentations, and unfortunately the ones provided by the Acrobat viewers are just ugly. Anyhow, one automatically gets them by saying:

```
\setuppagetransitions[random]
```

That way one gets random transitions. Resetting transitions is done by:

```
\setuppagetransitions[reset]
```

If needed one can specify transitions, but beware: these commands are very viewer dependant. Table **??** shows some alternatives. One can use numbers as well as sets. Using numbers is safer, i.e. less viewer dependant.

The following settings are all valid:

```
\setuppagetransitions
\setuppagetransitions[1]
\setuppagetransitions[3,5,8,random]
```

Don't use transitions indiscriminately. Using transitions with for instance the Polish style spoils the design. I must admit that so far I never use transitions myself.

Slowly building up

The macro `\presentationstep` provides a basic slideshow functionality. It records portions of the page that will successively become visible. It can be used as follows:

```
\startitemize
\item eerste
\item tweede
\stopitemize

\presentationstep

\startformula
ax^2+bx+c
\stopformula

\presentationstep
```

When the document is opened, the two text fragments are hidden by a covering. Each page has its own stack of coverings. The reference `NextStep` can be used to hide the shield, for instance in:

```
\setupfootertexts
  [{\button{Show Up}[NextStep]}]
```

Special effects can be accomplished by setting up the related framed text:

```
\setupframedtexts
  [presentationshield]
  [background=color,backgroundcolor=red]
```

Expect this and similar features to turn up in future presentations. They are available when one loads the auxiliary module:

```
\usemodule[pre-general]
```

One can save some work by using the automatic covering mechanism. Just say:

```
\autopresentationsteptrue
```

For the moment this switch is not yet embedded in a more user friendly setup command.

More information

At `www.pragma-ade.nl` one can find the style documentation as well as the previous images in full color. The styles themselves are part of the standard ConTEXt distribution.

◇ Hans Hagen
pragma@wxs.nl