

ConT_EXt

Presentation Environment 2

category: ConT_EXt Style File

version: 1998.04.21

date: August 20, 1999

author: Hans Hagen

copyright: PRAGMA / Hans Hagen & Ton Otten

This environment can be used to typeset interactive presentations. This module was first used at the 1998 publishers conference of the European Portable Document Association.

```

1 \usemodule[pre-general]

\setupbody.. At PRAGMA ADE we prefer using the Lucida Bright fonts, but one can of course load another typeface.
\switchtob..
\setuplayout \setupbodyfont
2 [lbr,14.4pt]

3 \setuplayout
  [style={\switchtobodyfont[12pt]}]

\setuppage.. The papersize suits the screen dimensions. The layout is rather simple. We use the whole width of
\setuplayout the screen and only have navigational tools at the bottom of the screen.

4 \setuppapersize
  [S6][S6]

5 \setuplayout
  [backspace=1cm,
   topspace=1cm,
   margin=0pt,
   header=0pt,
   footer=0pt,
   bottomdistance=.875cm,
   bottom=1cm,
   width=fit,
   height=fit]

\setupwhit.. We don't have much height, so we use a more cramped spacing. Verbatim text looks better when
\setuptyping indented.

6 \setupwhitespace
  [medium]

7 \setuptyping
  [margin=standard]

\definecolor Of course we enable color. We define some logical colors, of which most default to the same green
\setupcolors shade.

8 \definecolor [Backgrounds] [r= 0, g=.7, b=.4]
  \definecolor [Actions] [Backgrounds]
  \definecolor [Lines] [Actions]
  \definecolor [Page] [r=.8, g=.8, b=.8]

9 \setupcolors
  [state=start]

\setupinte.. We still have to enable interaction mode. We go full screen!
\setupinte..
10 \setupinteraction
    [state=start,
     color=Actions]
```

```

11 \setupinteractionscreen
    [option=max,
    width=fit,
    height=fit]

\setupitem.. And why not bring some color in itemizations too?

12 \setupitemize
    [color=Backgrounds]

\defineove.. The navigational elements and the backgrounds are provided by METAPOST. The next few macros are
\setupback.. meant for the more experienced ConTeXt user. When, as demonstrated here, we make the dimensions
part of the name, we can be sure that new ones are generated when we change the page size in between.

When METAPOST is used, it makes sense to generate the graphics at runtime. This is supported when
one enables system calls in the local texmf.cnf file and add the switch \runMPgraphicstrue to the
local file cont-sys.tex. When direct processing is disabled or not supported, TEXEEXEC will tke care
of graphic generation.

13 \MPinclusions{input mp-tool}

14 \def\MPpagebackground#1#2#3%
    {\startreusableMPgraphic{pb:#1#2#3}
    delta := #3 ;
    fill unitsquare
        xscaled #1 yscaled #2
        withcolor \MPcolor{Page};
    draw unitsquare
        xscaled (#1-2delta) yscaled (#2-2delta)
        shifted (delta,delta)
        withpen pencircle scaled .5delta
        withcolor \MPcolor{Backgrounds} ;
    \stopreusableMPgraphic
    \reuseMPgraphic{pb:#1#2#3}}

15 \defineoverlay
    [PageBackground]
    [\MPpagebackground{\overlaywidth}{\overlayheight}{.25cm}]

16 \setupbackgrounds
    [page]
    [background=PageBackground]

\setuptext.. By clicking on the text area, one goes to the next page. We could also have use the overlay mechanism,
but this takes more settings and requires to generate the backgrounds each page.

    \defineoverlay[BackgroundButton] [\BackgroundButton]
    \setupbackgrounds[text][text][background=BackgroundButton]
    \setupbackgrounds[state=repeat]

We go for the fast one.

17 \def\MPtextbackground#1#2#3%
    {\startreusableMPgraphic{tx:#1#2#3}
    delta := #3 ;
    draw unitsquare

```

```

        xscaled #1 yscaled #2
        withpen pencircle scaled .5delta
        withcolor \MPcolor{Backgrounds} ;
\stopreusableMPgraphic
\reuseMPgraphic{tx:#1#2#3}}

18 \defineoverlay
    [TextBackground]
    [\MPtextbackground{\overlaywidth}{\overlayheight}{.125cm}]

19 \def\BackgroundButton%
    {\button
     [background=TextBackground,
      backgroundoffset=.25cm,
      frame=off,
      height=\vsize,
      width=\hsize]{}[nextpage]}

20 \setuptexttexts
    [\BackgroundButton]
    []

\setupbott.. At the bottom of the screen, we show three buttons. These direct us to the previous or next jump or
              exit the document.

21 \def\MPright#1#2%
    {\startreusableMPgraphic{rt:#1#2}
     z1 = ( 0,    0) ;
     z2 = (#1, .5y3) ;
     z3 = ( 0,   #2) ;
     path p ; p := z1--z2--z3--cycle;
     lw := #2/5 ; pickup pencircle scaled lw;
     filldraw p withcolor \MPcolor{Backgrounds} ;
     set_outer_boundingBox currentpicture;
     \stopreusableMPgraphic
     \reuseMPgraphic{rt:#1#2}}

22 \def\MPleft#1#2%
    {\startreusableMPgraphic{lt:#1#2}
     z1 = (#1,    0) ;
     z2 = ( 0, .5y3) ;
     z3 = (#1,   #2) ;
     path p ; p := z1--z2--z3--cycle;
     lw := #2/5 ; pickup pencircle scaled lw;
     filldraw p withcolor \MPcolor{Backgrounds} ;
     set_outer_boundingBox currentpicture;
     \stopreusableMPgraphic
     \reuseMPgraphic{lt:#1#2}}

23 \def\MPcircle#1#2%
    {\startreusableMPgraphic{ci:#1#2}
     rx := #1; ry := #2;
     path p ; p := fullcircle xscaled rx yscaled ry;
     lw := #2/5 ; pickup pencircle scaled lw;
     filldraw p withcolor \MPcolor{Backgrounds} ;

```

```

        set_outer_boundingbox currentpicture;
        \stopreusableMPgraphic
        \reuseMPgraphic{ci:#1#2}}

24 \def\MPwhat#1#2%
    {\startreusableMPgraphic{wh:#1#2} % could be rotated unitsquare
      z1 = ( 0, 0) ;
      z2 = ( #1, 0) ;
      z3 = (.5x2, #2) ;
      path p ; p := z1--z2--z3--cycle;
      lw := #2/5 ; pickup pencircle scaled lw;
      filldraw p withcolor \MPcolor{Backgrounds} ;
      set_outer_boundingbox currentpicture;
      \stopreusableMPgraphic
      \reuseMPgraphic{wh:#1#2}}

25 \unexpanded\def\BackwardButton%
    {\gotobox{\MPleft{\the\bottomheight}{\the\bottomheight}}[previouspage]}

26 \unexpanded\def\ForwardButton%
    {\gotobox{\MPright{\the\bottomheight}{\the\bottomheight}}[nextpage]}

27 \unexpanded\def\ExitButton%
    {\gotobox{\MPcircle{\the\bottomheight}{\the\bottomheight}}[CloseDocument]}

28 \unexpanded\def\WhateverButton%
    {\doifreferencefoundelse{Whatever}
      {\gotobox{\MPwhat{\the\bottomheight}{\the\bottomheight}}[Whatever]}
      {}}

29 \def\BottomButton%
    {\button
      [frame=off,
       height=\bottomheight,
       width=.3\textwidth]{}[Topics]}

30 \setupbottomtexts
    [\BottomButton
     \hfill
     \WhateverButton
     \kern2\bottomheight
     \BackwardButton
     \kern.5\bottomheight
     \ExitButton
     \kern.5\bottomheight
     \ForwardButton
     \kern.5\bottomheight]
    [] % redundant

```

```

\TitlePage A presentation after loading this module looks like:
\Topics
\Topic      \TitlePage {About Whatever\Topics}
\Subject

\Topics {Todays Talk}

\Topic {Some topic}

```

.....

`\Topic {Next Topic}`

.....

`\StartTitl..` The titlepage is rather simple and can be typeset in two ways:
`\TitlePage`

```
\StartTitlePage
text \\ text \\ text
\StopTitlepage
```

or as one-liner:

```
\TitlePage{text\\text\\text}
```

The first alternative can be used for more complicated title pages.

```
31 \def\StartTitlePage%
    {\startstandardmakeup
     \bfd\setupinterlinespace
     \setupalign[middle]
     \vfil
     \let\\=\vfil}

32 \def\StopTitlePage%
    {\vfil\vfil\vfil
     \stopstandardmakeup}

33 \def\TitlePage#1%
    {\StartTitlePage#1\StopTitlePage}
```

`\definehead` The commands `\Topic` and `\Subject` are defined as copies of `head`. We use `\Nopic` for internal purposes.

```
34 \definehead [Topic]    [chapter]
    \definehead [Subject] [section]

35 \definehead [Nopic]    [title]
```

`\setuphead` We use our own command for typesetting the titles. We hide sectionnumbers from viewing. Each topic is followed by a list of subjects that belong to the topic.

```
36 \setuphead
    [Topic, Nopic]
    [after={\blank[3*medium]}],
    number=no,
    style=\tfb,
    page=yes,
    alternative=middle]

37 \setuphead
    [Subject]
    [after=\blank,
     number=no,
     page=yes,
```

```

    continue=no,
    style=\tfa]

```

`\setuplist` The subject list is automatically placed. We center each subject line by using one of the default alternatives (g). We could have said:

```

\setuplist
  [Subject]
  [alternative=none,
   command=\SubjectListLine,
   interaction=all]

```

```

\def\SubjectListLine#1#2#3%
  {\midaligned{#2}}

```

But why should we complicate things when we can say:

```

38 \setuplist
   [Topic,Subject]
   [alternative=g,
    interaction=all,
    before=,
    after=]

39 \def\Topics#1%
   {\determinelistcharacteristics[Topic]
    \ifcase\utilitylistlength\else
      \Nopic[Topics]{#1}
      \startcolumns
      \placelist[Topic][criterium=all]
      \stopcolumns
    \fi}

40 \def\Subjects%
   {\placelist[Subject]}

41 \endinput

```


<code>\definecolor</code>	<code>1</code>	<code>\setuplist</code>	<code>6</code>
<code>\definehead</code>	<code>5</code>	<code>\setuppapersize</code>	<code>1</code>
<code>\defineoverlay</code>	<code>2</code>	<code>\setuptexttexts</code>	<code>2</code>
		<code>\setuptyping</code>	<code>1</code>
<code>\setupbackgrounds</code>	<code>2</code>	<code>\setupwhitespace</code>	<code>1</code>
<code>\setupbodyfont</code>	<code>1</code>	<code>\StartTitlePage</code>	<code>5</code>
<code>\setupbottomtexts</code>	<code>3</code>	<code>\Subject</code>	<code>4</code>
<code>\setupcolors</code>	<code>1</code>	<code>\switchtobodyfont</code>	<code>1</code>
<code>\setuphead</code>	<code>5</code>		
<code>\setupinteraction</code>	<code>1</code>	<code>\TitlePage</code>	<code>4, 5</code>
<code>\setupinteractionscreen</code>	<code>1</code>	<code>\Topic</code>	<code>4</code>
<code>\setupitemize</code>	<code>2</code>	<code>\Topics</code>	<code>4</code>
<code>\setuplayout</code>	<code>1</code>		

