

The `ocgtools` package^{*†}

Robert Mařík
marik@mendelu.cz

April 8, 2010

1 Introduction

The package `ocgtools` is designed to insert OGC (Optional Group Content) into PDF presentations. From the user's point of view, the package allows to insert in an comfortable way any \TeX material into separate layer in PDF document and insert links which toggle this layer on/off. Hence parts of PDF document like formatted text, tables, math formulas or graphics can be turned to visible or invisible state by clicking active links or buttons.

Similar packages are `cooltooltips`, `pdfcomment`, `AcroTeX` and `fancytooltips`. In contrast to `cooltooltips` and `pdfcomment`, we can work with any \TeX material, not only plain text. In contrast to `fancytooltips`, no external file is necessary and more minilayers (see below) can be opened simultaneously on one page. However, `fancytooltips` and `ocgtools` can be combined in the same document, even on the same page. `AcroTeX` has far more possibilities than `ocgtools`, but has three disadvantages: (1) limited support for `pdftex` (no layers with `pdftex` driver) (2) PDF file needs postprocessing (3) for the post-processing the non-free Adobe Acrobat Professional is necessary. However, the user of `ocgtools` must have `AcroTeX` installed, since we use its capability to insert buttons and JavaScripts into document (`eforms` and `insdljs` packages).

Two types of OCG objects can be inserted

- *layers*: OCG's which span across the whole size of paper (scaled if necessary)
- *minilayers*: OCG's which have their natural size and are placed somewhere close to the link which toggles them on/off.

The user should use viewer which allows to hide/reveal layers by JavaScript. This includes especially Adobe Reader. Some limited functionality is also in Foxit Reader (see the option `nobutton` below). In some other viewers (like `xpdf`) users see red warning on the first page (see `\ocgtools@msg`), all layers are visible and cannot be hidden.

Important
comment!

Big warning: Unfortunately, with this package you may make your PDF files **less portable** even if you use Adobe Reader! Till now, we observed the following problems:

- `jpg` pictures may look darker when using `transparent` option and **Linux** version of Adobe Reader – Adobe Reader switches to different rendering method which seems to be system dependent. Hope, Adobe fixes this problem soon.
- The layer with initial message on the first page which should be hidden to the users of Adobe Reader is sometimes still visible, if the PDF file is opened in Internet browser. From this reason, layer `ocgtools0` is switched on and then off when initializing PDF document. Hope, this workaround solves the problem. If not, report the problem with a minimal example, please.

^{*}This document corresponds to `ocgtools` v0.6, dated 2010/04/10.

[†]Supported by grant 131/2010 of Higher Education Development Fund (FRVŠ)

2 Examples

Several examples are distributed with the package. We have one minimal example, one example which shows cooperation with `preview.sty` and mainly – examples which show cooperation with three most popular packages to build presentations. This includes `web.sty`, `pdfscreen.sty` and `beamer.cls`. We have three demo files for each – with no panel, with panel on the right and with panel on the left. All examples can be recompiled by running `ocgtools-test.bat` on Windows and `ocgtools-test.sh` on Linux.

3 Usage

All the packages we use are on CTAN or in T_EXlive2009. The file `ocg.sty` is a part of `asymptote` package¹. AcroTeX is quite old on CTAN but the version on CTAN works well. New version of AcroTeX is at www.acrotex.net, if necessary.

Prerequisites: Only `pdflatex` is supported. The route via `dvips` or `dvipdfm` is not (yet?) supported (this restriction follows from `ocg.sty` and `transparent.sty` packages). To work with the package load `color.sty` and `hyperref.sty` packages. (This is usually done automatically by most packages which are used to build PDF presentations. If not, these packages are loaded by `ocgtools` automatically as soon as `\definecolor` and `\href` remain undefined, respectively.) You may use also `xcolor.sty`, but this package is not compatible with `transparent` option.

You have to compile your `.tex` file *three times* (!). If you change your document and create new layer, you have to compile three times again. After most changes (which include change in position of but not the number of layers) you have to compile twice to put layers on correct position. If you change the content of text layers or minilayers, one compilation is sufficient.

Load package `ocgtools.sty` as usual: `\usepackage[options]{ocgtools}`. Options include:

transparent Layers in fact do not span over the whole page, but the material of layer is scaled to some reasonable size (we keep aspect ratio). If the layer is activated, the page is covered by a uniform color `ocgbg` and the layer is placed on the top. The options `transparent` makes the color `ocgbg` transparent. Looks cool, but it could be slow and could change some colors of bitmap pictures (and this behavior seems to be system dependent – looks differently on Linux and differently on Windows). The transparency is achieved by putting the command `\ocgtools@transparent` on appropriate place. The default setting is `\def\ocgtools@transparent{\transparent{0.4}}`. Use something like like the following three lines to override:

```
\makeatletter
\def\ocgtools@transparent{\transparent{0.6}}
\makeatother
```

nobutton The pages with active layers have a transparent button which can be used to hide this layer. The user simply clicks anywhere and the layers become hidden. In some viewers (like Foxit Reader) the button is not 100% transparent. This option allows not to include the big button to hide layers. The layers can be closed by clicking the red cross below.

insertvisible In some rare cases problems with OCG's occur on Windows installation of T_EX. Temporary (I hope) workaround is to use this option, which inserts OCG's as visible. The visibility is turned off when the PDF is opened on the first page.

¹Do not interchange with the `ocg.sty` provided by Österreichische Computer Gesellschaft. The correct file `ocg.sty` is at <http://www.tug.org/svn/texlive/trunk/Master/texmf/tex/latex/asymptote/ocg.sty>

mouseover Layers can be opened by MouseOver action. The corresponding active area is an invisible square with side 8pt placed on the right bottom corner of the text or picture which opens the layer. More details: MouseOver opens the layer and MouseExit hides this layer again. If mouse button is clicked, the layer remains opened after exiting the button and can be closed by hovering and exiting minibutton, clicking the active area outside the minibutton or clicking the red cross. The field with red cross gets focus after opening the layer and hence pressing Enter key also hides the layer or minilayer.

minimouseover As mouseover, but restricted to minilayers.

noocg The package and its commands are ignored.

noprogressmsg Opening PDF on the first page show the message related to initial processing layers. This option turns the message off.

nopageclose By default, all layers become off and all buttons become hidden if the page is changed. The option **nopageclose** suppresses this behavior.

\ocgtext The macro `\ocgtext[⟨width⟩]{⟨text1⟩}{⟨text2⟩}` is used to create layer which contains any L^AT_EX material. *⟨text1⟩* is a text which is written in an “usual” way (in blue color which indicates that this text can be used to hide/unhide another object) and this text is used to hide/unhide layer with *⟨text2⟩* (which could be text divided into more than one paragraph, figure created by `mfpic` or any L^AT_EX material). *⟨text2⟩* is either placed into `\hbox` (if *⟨width⟩* is 0pt, which is default) or wrapped by `\vbox` with `\hsize` equal to *⟨width⟩* (in the opposite case).

\ocgpicture The macro `\ocgpicture[⟨params⟩]{⟨picture⟩}` is used to insert a picture which is used as a link to layer with bigger version of this picture. The optional parameter *⟨params⟩* is used by `\includegraphics` command to draw picture in text. The layer contains scaled version of the picture (aspect ratio is preserved).

\ocgminitext The macro `\ocgminitext[⟨width⟩]{⟨text1⟩}{⟨text2⟩}` is used to create layer containing *⟨text2⟩* which is placed near the right top corner of *⟨text1⟩*. *⟨text2⟩* is placed into `\hbox` or `\vbox` with given `\hsize` according to the value of *⟨width⟩*, as has been explained at `\ocgtext` macro. Macros `\ocgminitextrb`, `\ocgminitextlt` and `\ocgminitextlb` can be used to place the minilayer to the right bottom, left to and left bottom corner.

\ocgtextstart Macros `\ocgtextstart` and `\ocgtextend` are used to denote the start and the end of hyperlink which is used to hide/reveal layers. Default setting is `\def\ocgtextstart{\color{blue}}` and `\def\ocgtextend{}` and hence, the links are blue and there is no mark at the end of the link.

ocgpaper The colors `ocgbg` and `ocgpaper` are used to set the colors which are used to hide page and color which is used as background for text produced by `\ocgtext` and `\ocgminitext` commands. The default setting is `\definecolor{ocgpaper}{rgb}{1,1,0.2}` and `\definecolor{ocgbg}{rgb}{0,0,0}`.

\layerHshift The dimensions `\layerHshift` and `\layerVshift` are used to place layers exactly on the top of the page. If presentation packages `web`, `beamer` or `pdfscreen` are used, then the length is set automatically.

\ocgtools@shipoutstart@hook `\ocgtools@shipoutstart@hook` and `\ocgtools@shipoutend@hook` are introduced ~~`\ocgtools@shipoutstart@hook`~~ ~~`\ocgtools@shipoutend@hook`~~ to insert some material at the begin and at the end of the box with the page contents. Can be used for example to put background to the presentation. Both commands should produce boxes of zero dimensions, i.e. use something like

```
\makeatletter \def\ocgtools@shipoutstart@hook{\hbox to 0 pt{\kern
-1in \Huge A\hss}} \makeatother
```

to insert letter “A” into the left bottom corner. Note that you will see this letter only if the background of the presentation is transparent. See also the files `ocgtools-example-web*.tex` and `ocgtools-example-web*.pdf` for slightly more complicated background.

\ocgclosechar Macro `\ocgclosechar` is used to print mark which is used to close layers and minilayers and **\ocgtools@msg** `\ocgtools@msg` contains string for users of viewers which do not support layers (see the first few lines of the code for default settings).

4 Possible future development and known problems

The source code is in Mercurial repository at <http://bitbucket.org/robert.marik/ocgtools/>. You can also report problems and issues in the forum at this site.

1. Test and support more packages for slides than current `web.sty`, `pdfscreen.sty` and `beamer.sty`? Which ones?
2. Using `preview` package it is possible to extract equations and figures from the document and redefine `\ref` and `\eqref` commands in such a way that clicking (or mouseover) opens on the current page the layer with this equation (figure) and shift clicking moves the user to the page with this equation (figure). Or would `fancytooltips` produce better results in this case (smaller PDF file)? See also the paper by Ross More at <http://www.tug.org/TUGboat/Articles/tb29-3/tb93moore.pdf> and the demo `examples/fancy-preview` in `fancytooltips` directory. See the `ocgtools-preview.*` files for some initial attempts in this direction.
3. Introduce draft mode, which prints all layers at the end of document with links there and back?
4. Improve placement of `minitextlayers`. Currently we have one the reference point for both `minitextlayers` aligned to top and bottom, which is far from optimal.
5. In some rare cases the layers may be not inserted properly when the \LaTeX file is compiled on **Windows** – see the option `insertvisible` which solves this problems in most cases. Since the author has limited access to \TeX installations on Windows and hence reporting of problems with a minimal example is highly appreciated.
6. The package cannot be used to hide 3D graphics inserted by `movie15` package. However, you can put this graphics into floating window or full-screen using capabilities of Acrobat Reader 9.
7. If you open and close layer and then use "Back" (`Alt+LeftArrow`), the layer opens again, but the button to hide this layer becomes unavailable. Possible solution is to define open action and close action for each layer (is it possible in current PDF specification?).

5 Implementation

Initial settings

```
1 (*package)
2 \def\ocgtools@msg{If message does not disappear after a short time, the
3   author either did not compile the \LaTeX{} file three times, or your
4   PDF viewer does not support OCG. Use Adobe Reader!}
5
6 \def\ocgtextstart{\color{blue}}
7 \def\ocgtextend{ }
8 \ifx\definecolor\undefined\RequirePackage{color}\fi
9 \ifx\href\undefined\RequirePackage[pdfTeX]{hyperref}\fi
10 \definecolor{ocgpaper}{rgb}{1,1,0.2}
11 \definecolor{ocgbg}{rgb}{0,0,0}
12 \RequirePackage{graphicx}
13 \RequirePackage{pifont}
14 \RequirePackage{ocg}
15
16 \def\ocgclosechar{\ding{56}}
```

Make packages and options known.

```
17 \RequirePackage{atbegshi}
18 \RequirePackage[pdftex]{eforms}
19 \newif\if@ocgtools@transparent\@ocgtools@transparentfalse
20 \DeclareOption{transparent}{\@ocgtools@transparenttrue}
21 \newif\if@ocgtools@insertvisible\@ocgtools@insertvisiblefalse
22 \def\ocgtools@initialvisibility{0}
23 \DeclareOption{insertvisible}{\@ocgtools@insertvisibletrue}
24 \def\ocgtools@initialvisibility{1}}
25 \newif\ifocg@hide@button\ocg@hide@buttontrue
26 \DeclareOption{nobutton}{\ocg@hide@buttonfalse}
27 \newif\ifocgtools@noocg\ocgtools@noocgfalse
28 \DeclareOption{noocg}{\ocgtools@noocgtrue}
29 \def\ocgtools@progressmsg{\lower \layerVshift\hbox to 0 pt{ %space
30 \textField[\V{OCGtools: processing OCG's ...}\BG{1}
31 \textColor{1 0 0} \textSize{10}]%
32 {ocgtoolsmsg}{6cm}{20pt}\hss}}%
33 \DeclareOption{noprogessmsg}{\let\ocgtools@progressmsg\relax}
34 \newif\if@ocgtools@minimouseover \@ocgtools@minimouseoverfalse
35 \DeclareOption{minimouseover}{\@ocgtools@minimouseovertrue}
36 \newif\if@ocgtools@mouseover \@ocgtools@mouseoverfalse
37 \DeclareOption{mouseover}{\@ocgtools@mouseovertrue \@ocgtools@minimouseovertrue}
38 \newif\if@ocgtools@pageclose \@ocgtools@pageclosetrue
39 \DeclareOption{nopageclose}{\@ocgtools@pageclosefalse}
40 \ProcessOptions
41
```

We use the hack for Adobe Acrobat suggested by DPS and Jorg at <http://www.acrotex.net/forum/showthread.php?tid=78>

```
42 \def\ocgtools@JS#1{\JS{DirtyBeforeOCGtools=this.dirty; #1
43 this.dirty=DirtyBeforeOCGtools;}}
```

We insert JavaScripts which are evaluated when the file is opened at the first page.

```
44 \edef\ocgtools@pdfpageattr{
45 /AA << /O << /S /JavaScript /JS
46 (
```

We close layers, if the page is opened. If the document is opened and `ocgtoolsOCGs` is not initialized, we skip to catch part and initialize.

```
47 try{
48   var temp = ocgtoolsOCGs.length;
49   \if@ocgtools@pageclose
50   for(var i=0; ocgtoolsOCGs && i<ocgtoolsOCGs.length;i++)
51     {
52       ocgtoolsOCGs[i].state = false;
53     }
54   \fi
55 }
```

We initialize document – we find all layers, put into `ocgtoolsOCGs` field and make them hidden.

```
56 catch (e){
```

No dotted rectangle for buttons which have focus.

```
57   app.focusRect = false;
58   var DirtyBeforeOCGtools=this.dirty;
```

The OCG objects inserted by `ocgtools` are stored in variable `ocgtoolsOCGs` when the PDF document is opened on the first page.

```
59   var iniocgtoolsOCGs = this.getOCGs();
60   var ocgtoolsOCGs = [];
61   for(var i=0; iniocgtoolsOCGs && i<iniocgtoolsOCGs.length;i++)
```

```

62     {
63     if(iniocgtoolsOCGs[i].name.substr(0,8) == "ocgtools")
64         {
65             ocgtoolsOCGs.push(iniocgtoolsOCGs[i]);
66             \if@ocgtools@insertvisible
67             iniocgtoolsOCGs[i].state=false;
68             \fi
69         }
70     }
71     iniocgtoolsOCGs[0].state=true;
72     iniocgtoolsOCGs[0].state=false;
73     \ifx\ocgtools@progressmsg\relax\relax\else
74     this.getField("ocgtoolsmsg").hidden=true;
75     this.dirty=false;
76     \fi
77     }
78     \ifocg@hide@button
79     this.getField("OcgtoolsBtn.HideButton.main").hidden = true;
80     this.dirty=false;
81     \fi
82     )
83     >> >>
84 }
85
86 \expandafter\global\expandafter\pdfpageattr\expandafter{\ocgtools@pdfpageattr}
87
88 \def\ocgtools@transparent{}
89 \if@ocgtools@transparent
90 \RequirePackage{transparent} \def\ocgtools@transparent{\transparent{0.4}}
91 \fi

```

Internal variables

```

92 \newif\ifocg@minitext@left
93 \newif\ifocg@minitext@bottom
94 \newcount\ocgtools@layercount
95 \newskip\ocgtools@left@skip
96 \newskip\ocgtools@bottom@skip
97 \newdimen\ocgtools@maxheight
98 \newdimen\ocgtools@maxwidth

```

We introduce lengths which can be used to finetune position of layers on the screen. These lengths are not necessary for plain pages (like article class with geometry package) and beamer.

```

99 \newdimen\layerHshift \layerHshift=0pt
100 \newdimen\layerVshift \layerVshift=0pt

```

We need small shift for web.sty.

```

101 \ifx\@Rightmargin\undefined\else
102   \layerVshift=3pt
103 \fi

```

We need another shift for pdfscreen.sty.

```

104 \ifx\PDFSCR@Warning\undefined\else
105   \layerVshift=\@Bottommargin
106 \fi
107
108 \newtoks\ocgtools@layer@toks
109 \ocgtools@layer@toks{}
110 \newbox\ocgtools@box@a
111 \newbox\ocgtools@box@b
112 \newif\ifocg@page@contains@layer

```

We modify `shipout` routine and insert content of layers on the top of PDF pages.

```
113 \def\ocgtools@one{1}
114 \let\ocgtools@shipoutstart@hook\relax
115 \let\ocgtools@shipoutend@hook\relax
116 \AtBeginShipout{%
117 \setbox\AtBeginShipoutBox=\hbox{%
```

Hook which can be used to place background.

```
118 \ocgtools@shipoutstart@hook
```

We insert the page first.

```
119 \hbox to 0 pt{\box\AtBeginShipoutBox\hss}\kern -1in\kern \layerHshift
```

We insert the layers.

```
120 \ocg@page@contains@layerfalse
121 \lower \layerVshift \hbox{\the\ocgtools@layer@toks}%
```

If at least one layer has been inserted, we insert button which can be used to hide layers.

```
122 \ifocg@page@contains@layer
123 \vbox to 0 pt{\kern -\paperheight \kern\layerVshift\hbox to 0 pt{\ocgtools@HideBtn\hss}\vss}%
124 \fi
```

We insert progress field on the first page.

```
125 \xdef\ocgtools@currpage{\thepage}%
126 \ifx\ocgtools@currpage\ocgtools@one
127 \ocgtools@progressmsg
128 \fi
```

We finish the box.

```
129 \hss \ocgtools@shipoutend@hook%
130 \kern 1in \kern -\layerHshift%
131 }%
132 }
```

We create buttons which are used to hide all layers (if we create buttons, we keep possibility to make them hidden, in contrast to links).

```
133 \def\ocgtools@HideBtn{\pushButton[\W{0}\BG{ }\S{S}\BC{ }\H{N}\F{\FHHidden}]\A{\ocgtools@JS{
134     for(var i=0; ocgtoolsOCGs && i<ocgtoolsOCGs.length;i++)
135     {
136         ocgtoolsOCGs[i].state = false;
137     }
138     \ifocg@hide@button this.getField("OcgttoolsBtn.HideButton").hidden = true; \fi
139     }]{OcgttoolsBtn.HideButton.main}{\paperwidth}{\paperheight}}
```

We create buttons which are attached to the top right corner of each minilayer and can be used to hide the corresponding minilayer.

```
140 \newcommand\ocgtools@HideMiniLayer[1]
141 {\setbox\ocgtools@box@a=\hbox{{\color{red}\ocgclosechar}}}%
142 \hbox to \wd\ocgtools@box@a%
143 {\hbox{{\color{red}\ocgclosechar}}\hss%
144 \hbox{\pushButton[\W{0}\S{S}\BG{ }\BC{ }\H{N}%
145 \F{\FHHidden}]\A{\ocgtools@JS{
146     for(var i=0; ocgtoolsOCGs && i<ocgtoolsOCGs.length;i++)
147     {
148         if(ocgtoolsOCGs[i].name == "ocgtools#1")
149         ocgtoolsOCGs[i].state = false;
150     }
151     this.getField("OcgttoolsBtn.HideButton.mini.#1").hidden = true;
152     }]}%
153 {OcgttoolsBtn.HideButton.mini.#1}{\wd\ocgtools@box@a}{\ht\ocgtools@box@a}}}
```

We create buttons which are attached to the bottom right corner of the screen and can be used to hide all layers.

```

154 \newcommand\ocgtools@HideLayers[1]{%
155 \setbox\ocgtools@box@a=\hbox{\color{red}\ocgclosechar}}%
156 \hbox to \wd\ocgtools@box@a%
157 {\hbox{\color{red}\ocgclosechar}}\hss%
158 \hbox{\pushButton[\W{0}\S{S}\BG{} \BC{} \H{N}%
159 \F{\FHHidden}\A{\ocgtools@JS{
160     for(var i=0; ocgtoolsOCGs && i<ocgtoolsOCGs.length;i++)
161     {
162         ocgtoolsOCGs[i].state = false;
163     }
164     \ifocg@hide@button
165     this.getField("OcgtoolsBtn.HideButton.main").hidden = true;
166     \fi
167 }}}%
168 {\ocgtoolsBtn.HideButton.corner.#1}{\wd\ocgtools@box@a}{\ht\ocgtools@box@a}}}
```

To hide/unhide layers and minilayers we use modified code from <http://www.texample.net/weblog/2008/nov/02/creating-pdf-layers/>

```

169 \newcommand{\ocgtools@ToggleLayer}[2]{%
170 % #1: layer name,
171 % #2: link text
172 \leavevmode%
173 \pdfstartlink user {
174 /Subtype /Link
175 /Border [0 0 0]%
176 /H /N
177 /A <<
178 /S/JavaScript
179 /JS (
180     DirtyBeforeOCGtools=this.dirty;
181     for(var i=0; ocgtoolsOCGs && i<ocgtoolsOCGs.length;i++)
182     {
183         if(ocgtoolsOCGs[i].name == "ocgtools#1")
184             ocgtoolsOCGs[i].state = !ocgtoolsOCGs[i].state;
185         else
186             ocgtoolsOCGs[i].state = false;
187     }
188     \ifocg@hide@button this.getField("OcgtoolsBtn.HideButton.main").hidden = false;\fi
189     this.getField("OcgtoolsBtn.HideButton.corner").hidden = false;
190     this.getField("OcgtoolsBtn.HideButton.corner.#1").setFocus();
191     this.dirty=DirtyBeforeOCGtools;
192 )
193 >>
194 }#2%
195 \ifocgtools@mouseover
196 \hbox to 0 pt{\hss\pushButton[\W{0}\S{S}\BG{} \BC{} \H{N}%
197 \A{\ocgtools@JS{
198     var OpenedByMouseEnter#1=false;
199     \ifocg@hide@button this.getField("OcgtoolsBtn.HideButton.main").hidden = false;\fi
200     this.getField("OcgtoolsBtn.HideButton.corner").hidden = false;
201     this.getField("OcgtoolsBtn.HideButton.corner.#1").setFocus();
202 }}}%
203 \AA{\AAMouseEnter{\ocgtools@JS{          for(var i=0; ocgtoolsOCGs && i<ocgtoolsOCGs.length;i++)
204     {
205         var OpenedByMouseEnter#1=true;
206         if(ocgtoolsOCGs[i].name == "ocgtools#1")
207             ocgtoolsOCGs[i].state = true;
```

```

208         else
209         ocgtoolsOCGs[i].state = false;
210     }
211 }
212 \AAMouseExit{\ocgtools@JS{
213     if (OpenedByMouseEnter#1)
214     {for(var i=0; ocgtoolsOCGs && i<ocgtoolsOCGs.length;i++)
215     {
216         if(ocgtoolsOCGs[i].name == "ocgtools#1")
217         ocgtoolsOCGs[i].state = false;
218     }}
219 }}}
220 {OcgtoolsBtn.MaxiButton.#1}{8pt}{8pt}}%
221 \fi
222 \pdfendlink%
223 }
224
225 \newcommand{\ocgtools@ToggleMiniLayer}[2]{%
226 % #1: layer name,
227 % #2: link text
228 \leavevmode%
229 \pdfstartlink user {
230 /Subtype /Link
231 /Border [0 0 0]%
232 /H /N
233 /A <<
234 /S/JavaScript
235 /JS (
236     DirtyBeforeOCGtools=this.dirty;
237     for(var i=0; ocgtoolsOCGs && i<ocgtoolsOCGs.length;i++)
238     {
239         if(ocgtoolsOCGs[i].name == "ocgtools#1")
240         ocgtoolsOCGs[i].state = !ocgtoolsOCGs[i].state;
241     }
242     this.getField("OcgtoolsBtn.HideButton.mini.#1").hidden =
243     !this.getField("OcgtoolsBtn.HideButton.mini.#1").hidden;
244     if (!this.getField("OcgtoolsBtn.HideButton.mini.#1").hidden)
245     {this.getField("OcgtoolsBtn.HideButton.mini.#1").setFocus();}
246     this.dirty=DirtyBeforeOCGtools;
247 )
248 >>
249 }#2%
250 \if@ocgtools@minimouseover
251 \hbox to 0 pt{\hss\pushButton[\W{0}\S{S}\BG{ }\BC{ }\H{N}]%
252 \A{\ocgtools@JS{
253     var OpenedByMouseEnter#1=false;
254     this.getField("OcgtoolsBtn.HideButton.mini.#1").setFocus();
255 }}
256 \AA{\AAMouseEnter{\ocgtools@JS{for(var i=0; ocgtoolsOCGs && i<ocgtoolsOCGs.length;i++)
257     {
258         var OpenedByMouseEnter#1=true;
259         if(ocgtoolsOCGs[i].name == "ocgtools#1")
260         ocgtoolsOCGs[i].state = true;
261     }
262     this.getField("OcgtoolsBtn.HideButton.mini.#1").hidden = false;
263 }}
264 \AAMouseExit{\ocgtools@JS{
265     if (OpenedByMouseEnter#1)

```

```

266     {
267         for(var i=0; ocgtoolsOCGs && i<ocgtoolsOCGs.length;i++)
268         {
269             if(ocgtoolsOCGs[i].name == "ocgtools#1")
270                 ocgtoolsOCGs[i].state = false;
271         }
272         this.getField("OcgtoolsBtn.HideButton.mini.#1").hidden = true;
273     }
274     }}}}
275     {OcgtoolsBtn.MiniButton.#1}{8pt}{8pt}}%
276 \fi
277 \pdfendlink%
278 }
279

```

Macros dealing with \savepos from Vit Zyka's paper in CSTUG Bulletin (ISSN 1211-6661, No. 2, 2007)

```

280 \newwrite\posHandle
281 \def\posFile{\jobname.pos}
282 \def\posOpen{\openout\posHandle=\posFile}
283 \def\posClose{\closeout\posHandle}
284 \ifocgtools@noocg
285 \else
286 \AtBeginDocument{\InputIfFileExists{\posFile}{-}{-}}%
287 \ocgtools@maxheight=0.9\paperheight
288 \ocgtools@maxwidth=0.9\paperwidth
289 \expandafter\global\expandafter\def\csname ocgtools@textcontent@0\endcsname
290 {\color{red}\bfseries\ocgtools@msg}
291 \expandafter\ocgtools@layer@toks\expandafter{\the\ocgtools@layer@toks}
292 \ocg@place@text{0}{1}{5cm}
293 \posOpen}
294 \AtEndDocument{\posClose}
295 \fi
296

```

The following commands create links to hide/unhide layers and minilayers and write information on the layer number, page and layer content into the pos file. The content of textlayer is stored in command \ocgtools@textcontent@<n> where <n> is the number of the layer. Similarly, the content of the minitextlayer is stored in \ocgtools@minitextcontent@<n>.

```

297 \newcommand\ocgpicture[2][ ]{%
298 \global\advance\ocgtools@layercount by 1%
299 \ocgtools@ToggleLayer{\the\ocgtools@layercount}%
300 {\leavevmode\includegraphics[#1]{#2}}%
301 \expandafter\write\expandafter\posHandle\expandafter{%
302 \expandafter\string\expandafter\ocgtools@placepicture\expandafter{%
303 \the\ocgtools@layercount}{\thepage}{#2}}%
304 }
305
306 \newcommand\ocgtext[3][0pt]{%
307 \global\advance\ocgtools@layercount by 1%
308 \ocgtools@ToggleLayer{\the\ocgtools@layercount}%
309 {\leavevmode{\ocgtextstart #2\ocgtextend}}%
310 \expandafter\write\expandafter\posHandle\expandafter{%
311 \expandafter\string\expandafter\ocgtools@placetext\expandafter{%
312 \the\ocgtools@layercount}{\thepage}{#1}}%
313 \expandafter\global\expandafter\def
314 \csname ocgtools@textcontent@the\ocgtools@layercount\endcsname{#3}%
315 }
316

```

```

317 \newcommand\ocgminitextlt[3][Opt]{%
318 \ocg@minitext@lefttrue\do@ocg@minitext{#1}{#2}{#3}{lt}}%
319 \newcommand\ocgminitextrt[3][Opt]{%
320 \ocg@minitext@leftfalse\do@ocg@minitext{#1}{#2}{#3}{rt}}%
321 \newcommand\ocgminitextlb[3][Opt]{%
322 \ocg@minitext@lefttrue\do@ocg@minitext{#1}{#2}{#3}{lb}}%
323 \newcommand\ocgminitextrb[3][Opt]{%
324 \ocg@minitext@leftfalse\do@ocg@minitext{#1}{#2}{#3}{rb}}%
325 \let\ocgminitext\ocgminitextrt
326
327 \long\def\do@ocg@minitext#1#2#3#4{%
328 \global\advance\ocgtools@layercount by 1\relax%
329 \leavevmode
330 \ifocg@minitext@left
331 \vbox to 0 pt{\kern 0.5\baselineskip\hbox to 0 pt{\hss\pdfsavepos}\vss}%
332 \fi
333 \ocgtools@ToggleMiniLayer{\the\ocgtools@layercount}%
334 {\ocgtextstart #2\ocgtextend}}%
335 \ifocg@minitext@left\else\pdfsavepos\fi
336 \expandafter\write\expandafter\posHandle\expandafter{%
337 \expandafter\string\expandafter\ocgtools@placeminitext\expandafter
338 {\the\ocgtools@layercount}{\thepage}{\the\pdflastxpos}%
339 {\the\pdflastypos}{#1}{#4}}%
340 \expandafter\global\expandafter\def
341 \csname ocgtools@minitextcontent@the\ocgtools@layercount\endcsname{#3}%
342 }%
343

```

These commands appear in `pos` file and we read these commands at the begin of the document. Pictures are stored in token register, texts for layers and minilayers are stored in commands defined by `\csname`.

```

344 \newcommand\ocgtools@placepicture[3]{%
345 \expandafter\global\expandafter\ocgtools@layer@toks\expandafter
346 {\the\ocgtools@layer@toks \ocg@place@picture{#1}{#2}{#3}}
347
348 \newcommand\ocgtools@placetext[3]{%
349 \expandafter\global\expandafter\ocgtools@layer@toks\expandafter
350 {\the\ocgtools@layer@toks\ocg@place@text{#1}{#2}{#3}}
351
352 \newcommand\ocgtools@placeminitext[7]{%
353 \expandafter\global\expandafter\ocgtools@layer@toks\expandafter
354 {\the\ocgtools@layer@toks \ocg@place@minitext{#1}{#2}{#3}{#4}{#5}{#6}{#7}}
355

```

These commands are called in output routine for each layer on each page. They put the layer on the page, if the layer should be here (i.e., if the second parameter equals `\thepage`).

```

356 \def\ocg@place@picture#1#2#3{\def\tempnuma{#2}\edef\tempnumb{\thepage}%
357 \ifx\tempnumb\tempnuma
358 \global\ocg@page@contains@layertrue
359 \vbox to 0 pt{\vss\hbox to 0pt%
360 {\hbox{\begin{ocg}{ocgtools#1}{ocgtools#1}{ocgtools@initialvisibility}%
361 \ocgtools@drawpicture{#3}{#1}\end{ocg}}\hss}}%
362 \fi}
363
364 \def\ocg@place@text#1#2#3{\def\tempnuma{#2}\edef\tempnumb{\thepage}%
365 \ifx\tempnumb\tempnuma
366 \global\ocg@page@contains@layertrue
367 \vbox to 0 pt{\vss\hbox to 0 pt{\hbox{%
368 \begin{ocg}{ocgtools#1}{ocgtools#1}{ocgtools@initialvisibility}%

```

```

369     \ocgtools@drawtext{#3}{\csname ocgtools@textcontent@#1\endcsname}{#1}%
370     \end{ocg}}\hss}}%
371 \fi}
372
373 \long\def\ocg@place@minitext#1#2#3#4#5#6#7{%
374 \def\tempnuma{#2}\edef\tempnumb{\thepage}%
375 \def\ocg@placement{#7}%
376 \def\ocg@rb{rb}\def\ocg@lb{lb}\def\ocg@rt{rt}\def\ocg@lt{lt}%
377 \ocg@minitext@leftfalse\ocg@minitext@bottomfalse
378 \ifx\ocg@placement\ocg@lb
379   \ocg@minitext@lefttrue\ocg@minitext@bottomtrue
380 \fi
381 \ifx\ocg@placement\ocg@rb
382   \ocg@minitext@leftfalse\ocg@minitext@bottomtrue\fi
383 \ifx\ocg@placement\ocg@lt
384   \ocg@minitext@lefttrue\ocg@minitext@bottomfalse
385 \fi
386 \ifx\tempnumb\tempnuma
387   \ocgtools@left@skip=#3sp minus #3sp%
388   \ocgtools@bottom@skip=#4sp%
389   \setbox\ocgtools@box@a=\hbox{\begin{ocg}{ocgtools#1}{ocgtools#1}{\ocgtools@initialvisibility}%
390 \ocgtools@drawminitext{#6}{\csname ocgtools@minitextcontent@#1\endcsname}{#1}
391 \end{ocg}}}%
392 \ifocg@minitext@left
393   \ifdim \ocgtools@left@skip>\wd\ocgtools@box@a
394     \advance \ocgtools@left@skip by -\wd\ocgtools@box@a plus 0 pt minus -\wd\ocgtools@box@a
395   \else
396     \ocgtools@left@skip=0pt\relax
397   \fi
398 \fi
399 \ifocg@minitext@bottom
400   \advance \ocgtools@bottom@skip by -\ht\ocgtools@box@a
401 \fi
402 \hbox to 0 pt{\hbox to \paperwidth{\hskip \ocgtools@left@skip
403 \vbox to 0 pt{\vss
404 \vbox to \paperheight{\vskip 0 pt plus 1 fill
405 \box\ocgtools@box@a\vskip \ocgtools@bottom@skip}
406 }%
407 \hskip 0 pt plus 1 fill}\hss}%
408 \fi
409 }
410

```

This code actually gives a graphical representation of the layers.

```

411 \def\ocgtools@drawpicture#1#2{\vbox to \paperheight{\vbox to 0 pt{{%
412 \ocgtools@transparent\color{ocgbg}%
413 \hrule width \paperwidth height \paperheight}\vss}\vss
414 \hbox to \paperwidth{\hss
415 \setbox\ocgtools@box@a=\hbox{\includegraphics[height=\ocgtools@maxheight]{#1}}%
416 \ifdim\wd\ocgtools@box@a>\ocgtools@maxwidth
417   \includegraphics[width=\ocgtools@maxwidth]{#1}\else \box\ocgtools@box@a
418 \fi
419 \hss}%
420 \vss
421 \hbox to \paperwidth {\hss\ocgtools@HideLayers{#2}}}}
422
423 \newdimen\ocg@textdimen
424
425 \long\def\ocgtools@drawtext#1#2#3{\vbox to \paperheight{%

```

```

426 \vbox to 0 pt{\ocgtools@transparent\color{ocgbg}%
427 \hrule width \paperwidth height \paperheight}\vss}\vss\hbox to \paperwidth
428 {\hss
429 \ocg@textdimen=#1
430 \ifdim\ocg@textdimen=0pt
431 \setbox\ocgtools@box@a=\hbox{\colorbox{ocgpaper}{#2}}%
432 \else
433 \setbox\ocgtools@box@a=\hbox{\colorbox{ocgpaper}%
434 {\vbox{\hsize=#1\relax\rightskip 0 pt plus 1 fil\relax #2}}}%
435 \fi
436 \setbox\ocgtools@box@b\hbox{\resizebox{!}{\ocgtools@maxheight}{\copy\ocgtools@box@a}}%
437 \ifdim\wd\ocgtools@box@b>\ocgtools@maxwidth
438 \resizebox{\ocgtools@maxwidth}{!}{\box\ocgtools@box@a}
439 \else
440 \box\ocgtools@box@b
441 \fi
442 \hss}%
443 \vss
444 \hbox to \paperwidth {\hss\ocgtools@HideLayers{#3}}}}
445
446 \def\ocg@empty{}
447
448 \newcommand\ocgtools@drawminitext[3]{\ocg@textdimen=#1
449 \ifdim\ocg@textdimen=0pt
450 \setbox\ocgtools@box@a=\hbox{\colorbox{ocgpaper}{#2\ }}%
451 \else \setbox\ocgtools@box@a=\hbox{\colorbox{ocgpaper}%
452 {\vbox{\hsize=#1\relax\rightskip 0 pt plus 1 fil\relax #2}}}%
453 \fi
454 \def\temp{#3}%
455 \ifx\temp\ocg@empty
456 \box\ocgtools@box@a
457 \else
458 \hbox{\copy\ocgtools@box@a\raise\ht\ocgtools@box@a
459 \vbox to 0 pt{\hbox to 0 pt{\hss
460 \ocgtools@HideMiniLayer{#3}}\vss}}}%
461 \fi
462 }
463
464 \ifocgtools@noocg
465 \renewcommand\ocgpicture[2][\leavevmode\includegraphics[#1]{#2}]
466 \renewcommand\ocgtext[3][0pt]{\leavevmode #2}
467 \long\def\do@ocg@minitext#1#2#3#4{\leavevmode #2}%
468 \renewcommand\ocgtoolsplacpicture[3]{}
469 \renewcommand\ocgtoolsplacetext[3]{}
470 \renewcommand\ocgtoolsplaceminitext[7]{}
471 \fi
472
473 </package>

```