The `ocgx` package (version 0.5)

Paul Isambert

Paul Gaborit
paul.gaborit@gmail.com

December 8, 2012

Contents

Abstract

The `ocgx` package extends the `ocg-p` package which allows you to create OCGs (*Optional Content Group*) in PDF documents.

Every OCG includes TeX material into a layer of the PDF file. Each of these layers can be displayed or not. Links can enable or disable the display of OCGs.

The `ocgx` package does not use Javascript embedded in the PDF document to enable (to show) or disable (to hide) OCGs.

> **1**
>
> OCGs are usable with several PDF readers: to date, it has been successfully tested with *Acrobat Reader*, *Foxit Reader*, *PDF-XChange-Viewer*, and *Evince.* The management of OGCs by *Evince* is not yet fully debugged: it still sometimes crash!

1 Usage

Here is a simple example.

— ocgx-example-1.tex —

```latex
\documentclass{article}
\usepackage{ocgx}
\begin{document}
\begin{ocg}{OCG 1}{ocg1}{1}
  first example.
\end{ocg}

\switchocg{ocg1}{Button.}
\end{document}
```

This document creates an OCG called *ocg1* containing the text "*first example.*" which is visible. You can show or hide this OCG by clicking the link "*Button.*".

1.1 Create OCGs

The following code creates an OCG named *OCG name* with *refocg* as internal reference. The content of this OCG is "*content...*". This OCG is visible (the third argument is 1).

```
— ocgx-example-2 —
\begin{ocg}{OCG name}{refocg}{1}
  content...
\end{ocg}
```

ocg   The `ocg` environment (provided by the package `ocg-p`) creates OCGs. It requires three arguments. The first argument is the name of the OCG as it appears in the layer toolbar of the PDF viewer. The second argument is the internal name used to reference this OCG. The third argument sets the initial visibility of the OCG when the document is opened (1 for visible, 0 for invisible). The content of the environment (any TeXmaterial) is added into the OCG.

> **2**   The same reference can be used with several `ocg` environments (not necessarily in the same page). All materials are grouped in the same OCG. Only the first name provided will be used.

> **3**   A reference of an OCG consists of letters (A-Z, a-z), numbers (0-9).

> **4**   The content of the `ocg` environment should not span across multiple pages. Currently, nothing prevents you to try it but the result will certainly not be the one you were expecting!

> **5**   It is possible to nest an OCG in another OCG. To display the internal OCG, both the internal and external OCGs need to be in the visible state.

1.2 Manage the visibility of OCGs

\switchocg   The `\switchocg` macro turns its second argument into a clickable link that toggles the visibility status of all listed OCGs (by their reference) in its first argument: if an OCG was visible, it becomes invisible, and conversely, if an OCG was invisible, it becomes visible.

The following code creates the link *toggle* that switches the visibility status of OCGs whose references are *ocg1* and *ocg2*:

```
\switchocg{ocg1 ocg2}{toggle}
```

\showocg   The `\showocg` macro turns its second argument into a clickable link that make visible all OCGs whose references are listed in its first argument: an invisible OCG becomes visible and an OCG already visible remains visible.

The following code creates the link *show* which makes visible the OCGs whoses references are *ocg1* and *ocg2*:

```
\showocg{ocg1 ocg2}{show}
```

\hideocg   The `\hideocg` macro turns its second argument into a clickable link that make invisible all OCGs whose references are listed in its first argument: a visible OCG becomes invisible and an OCG already invisible remains invisible.

The following code creates the link *hide* which makes invisible the OCGs whoses references are *ocg1* and *ocg2*:

```
\hideocg{ocg1 ocg2}{hide}
```

\actionsocg   The **\actionsocg** macro transforms its fourth argument into a clickable link. Its three first arguments are lists of references of OCGs. The first list contains references of OCGs which visibility status is to be toggled. The second list contains references of OCGs to be set visible. The third list contains references of OCGs to be set invisible.

The following code creates the link *actions* to toggle the visibility status of the OCG named *ocg1*, to make visible the OCG named *ocg3*, and to make invisible OCG named *ocg2*:

```
\actionsocg{ocg1}{ocg3}{ocg2}{actions}
```

1.3   Usage with Ti*k*Z

You can use the **ocgx** package with Ti*k*Z. The package provides a Ti*k*Z library offering some specific styles to add material to OCGs, or to transform a path (**\path** or **\node**) into a clickable link. To use it, simply add the following lines in your preamble:

— *ocgx-tikz-preamble* —
```
\usepackage{tikz}
\usetikzlibrary{ocgx}
```

How to add Ti*k*Z scopes into OCGs

/tikz/ocg=*{<ocg options>}*

The **ocg** family is used to specify the options used to add a scope into an OCG.

/tikz/ocg/ref=*<refname>*   (no default)

This option add the current scope to the OCG referenced by **refname**. If this OCG does not exist, it is created with **name** and visibility **status** specified by the two options below.

/tikz/ocg/name=*<name>*   (no default, initially empty)

It is the *name* of the OCG as it appears in the PDF viewer. If the OCG is already created, this option is useless.

/tikz/ocg/status=*<visibility>*   (no default, initially **visible**)

Specify the initial visibility state of the OCG. Permissible values are **visible**, and **invisible**. If the OCG is already created, this option is useless.

> **6**
>
> If you prefer, you can also use the **ogc** environment in a **tikzpicture**. The **pgfonlayer** environment is special: inside, you must reuse options below, or use a new **ocg** environment.

How to transform nodes or paths into clickable links

/tikz/switch ocg=*{<OCGs list>}*

This style transforms the current path or the current node in a link acting as if it was produced by the macro **\switchocg** (the visibility status of referenced OCGs is reversed).

/tikz/show ocg=*{<OCGs list>}*

This style transforms the current path or the current node in a link acting as if it was produced by the macro **\showocg** (the referenced OCGs are made visible).

/tikz/hide ocg=*{<OCGs list>}*

This style transforms the current path or the current node in a link acting as if it was produced by the macro **\hideocg** (the referenced OCGs are made invisible).

/tikz/actions ocg=*{<OCGs list>}{<OCGs list>}{<OCGs list>}*

This style transforms the current path or the current node in a link acting as if it is produced by the macro **\actionsocg** (the visibility status of OCGs of the first list is

reversed, the OCGs in the second list are made visible and those of the third list are made invisible).

`/tikz/switch ocg with mark on={<ocg reference>}{<OCGs list>}`

`/tikz/switch ocg with mark off={<ocg reference»}{<OCGs list>}`

These styles transform the current path or the current node in a link acting as if it is produced by the macro `\switchocg` (the visibility status of referenced OCGs in the list is reversed).

A mark (currently a simple cross) is drawn over the current path or node in an OCG whose reference is `ocg reference`. The visibility status of this OCG will be reversed as those of the entire list.

If the OCG whose reference is `ocg reference` does not exist, it is created with an empty name and its initial visibility state is true with `on` and false with `off`.

7
> Due to limitation of PDF, whatever the shape of the current path or node is, it is its *bounding box* that is used to make the link: the link is always *rectangular* and *horizontal*!

Example with Ti*k*Z

— *ocgx-example-3* —

```
\begin{tikpicture}
  \begin{scope}[ocg={name=TikZ example,ref=tikzex,status=visible}
    \fill[orange] (0,0) circle (1);
  \end{scope}
  \node[draw,switch ocg=tikzref] at (2,0) {Switch};
\end{tikpicture}
```

2 Examples

The document `demo-ocgx.tex` provides several examples of usage of package `ocgx` with Ti*k*Z (and `beamer`).

3 Limits and bugs

1. Links are always horizontal rectangles!

2. An `ocg` environment spanning across multiple pages are not detected and don't work correctly.

3. The packages `ocg-p` and `ocgx` are not compatible with Plain-TeX.

4. While `ocg-p` is usable with `pdflatex` and `xelatex`, `ocgx` works only with `pdflatex` (and `lualatex`).

4 Development and history

This package is still experimental. It is released on CTAN. You can donwload the latest version from `https://github.com/polgab/ocgx`. Any help to participate in its development is welcome: contact the maintainer (`paul.gaborit@gmail.com`).

version 0.5 (2012-12-07) Now, the `ocgx` package uses the new `ocg-p` package instead of the `ocg` package : the layers can be nested in the layer toolbar of the PDF viewer.

version 0.4 (2012-11-14) Added dependancy to `calc` TikZ library. Fixed bug: bad lists in `\actionsocg`.

version 0.3 (2012-09-30) Complete documentation for part *Usage with TikZ* and correct TDS archive.

version 0.2 (2012-09-27) First release on CTAN.