GL2PS, an OpenGL to Postscript Printing Library

Christophe Geuzaine

Version 0.3, 29 July 2000

Contents

1 Introduction

GL2PS is a library for creating postscript output from any OpenGL application. Though it was primarily designed for three-dimensional geometry, mesh and postprocessing visualization, it may be useful everytime high quality vector output is desired. The main difference between GL2PS and other similar libraries is the use of sorting algorithms capable of handling intersecting and stretched polygons, as well as non manifold objects.

The library, written in C, is released under GNU Library General Public License (see `http://www.gnu.org/` for more details), and is available at `http://www.geuz.org/gl2ps/`. Any corrections, questions or suggestions should be e-mailed to `Christophe.Geuzaine@advalvas.be`.

The interface consists of five functions, all begining with the prefix `gl2ps`. All the data structures and the symbolic constants peculiar to GL2PS begin with `GL2PS`.

2 Usage

2.1 gl2psBeginPage and gl2psEndPage

2.1.1 Specification

```
void gl2psBeginPage( char *title, char *producer, GLint sort,
                     GLint options, GLint colormode,
                     GLint colorsize, GL2PSrgba *colortable,
                     GLint buffersize, FILE *stream )

void gl2psEndPage( void )
```

2.1.2 Parameters

`title` Specifies the plot title. For Postscript output, this string is placed in the `%%Title` field.

`producer` Specifies the plot producer. For Postscript output, this string is placed in the `%%For` field.

`sort` Specifies the sorting algorithm, chosen among: `GL2PS_NO_SORT`, `GL2PS_SIMPLE_SORT`, `GL2PS_BSP_SORT`.

`options` Sets global plot options, chosen among: `GL2PS_NONE`, `GL2PS_DRAW_BACKGROUND`, `GL2PS_SIMPLE_LINE_OFFSET`, `GL2PS_SILENT`, `GL2PS_BEST_ROOT`. Multiple options are combined with the bitwise inclusive OR symbol, `|`.

`colormode` Specifies the color mode: `GL_RGBA` or `GL_COLOR_INDEX`.

`colorsize` Specifies the size of the colormap if `colormode` is `GL_COLOR_INDEX`.

`colortable` Contains the colormap if `colormode` is `GL_COLOR_INDEX`. This colormap must contain `colorsize` elements of type `GL2PSrgba`.

`buffersize` Specifies the size of the feedback buffer.

`stream` Specifies the stream to which data is printed.

2.1.3 Description

`gl2psBeginPage` and `gl2psEndPage` delimit the OpenGL commands that will be caught in the feedback buffer and output to `stream`. The parameters given to `gl2psBeginPage` determine the way primitives are handled:

`GL2PS_NO_SORT` The primitives are not sorted, and are output in `stream` in the order they appear in the feedback buffer.

`GL2PS_SIMPLE_SORT` The primitives are sorted according to their barycenter. This can be sufficient for simple scenes.

`GL2PS_BSP_SORT` The primitives are inserted in a BSP tree. The tree is traversed back to front in a painter-like algorithm.

`GL2PS_DRAW_BACKGROUND` The background frame is drawn.

`GL2PS_SIMPLE_LINE_OFFSET` Adds a small offset in the z-buffer to all lines. This is a simplified version of the `GL2PS_POLYGON_OFFSET_FILL` functionality (cf. section 2.3), putting all lines of the rendered image slightly in front of their actual position. This thus performs a simple anti-aliasing solution, e.g. for finite element like meshes.

`GL2PS_SILENT` Suppresses all messages written by GL2PS on the error stream.

`GL2PS_BEST_ROOT` Try to optimize the BSP tree by choosing as root primitives those leading to the minimum number of splits. This is (really) not efficient yet.

2.2 `gl2psText`

2.2.1 Specification

```
void gl2psText( char *string, char *fontname, GLint fontsize )
```

2.2.2 Parameters

`string` Specifies the text string to print.

`fontname` Specifies the name of a valid postscript font (for example `"Times"` or `"HelveticaBoldItalic"`).

`fontsize` Specifies the size of the font.

2.2.3 Description

`gl2psText` permits to include text in the postscript output in a very simple way. The text is inserted at the current raster position (set by one of the `glRasterPos` OpenGL commands). Beware that text will be sorted according to the position of the leftmost element of the string only.

2.3 `gl2psEnable` and `gl2psDisable`

2.3.1 Specification

```
void gl2psEnable( GLint mode )

void gl2psDisable( GLint mode )
```

2.3.2 Parameters

`mode` Specifies the mode to enable, chosen between `GL2PS_POLYGON_OFFSET_FILL`, `GL2PS_POLYGON_BOUNDARY`, `GL2PS_LINE_STIPPLE`.

2.3.3 Description

`gl2psEnable` and `gl2psDisable` delimit OpenGL commands to which a local mode is applied. These modes are:

`GL2PS_POLYGON_OFFSET_FILL` Tries to mimmic the `GL_POLYGON_OFFSET_FILL` functionnality. The value of the offset is taken as the current value of the corresponding OpenGL offset (set with `glPolygonOffset`). Not fully functionnal yet.

`GL2PS_POLYGON_BOUNDARY` Not implemented yet.

`GL2PS_LINE_STIPPLE` Tries to mimmic the `GL_LINE_STIPPLE` functionnality.

3 Example

Here is a typical calling sequence to produce BSP sorted postscript output in the file `"MyFile"`, with all lines slightly shifted front in the z-buffer. The `draw()` function contains all OpenGL instructions.

```
fp = fopen("MyFile", "w");
buffsize = 0;
state = GL2PS_OVERFLOW;

while( state == GL2PS_OVERFLOW ){
  buffsize += 1024*1024;
  gl2psBeginPage ( "MyTitle", "MySoftware", GL2PS_BSP_SORT,
                   GL2PS_SIMPLE_LINE_OFFSET | GL2PS_SILENT,
                   GL_RGBA, 0, NULL, buffsize, fp );
  draw();
  state = gl2psEndPage();
}

fclose(fp);
```

To output the text `"MyText"` at the current raster position, the `draw()` function should contain something like:

```
gl2psText("MyText", "Courier", 12);
```

4 Contributors

Michael Sweet (`mike@easysw.com`) for the original implementation of the feedback buffer parser; Marc Umé (`marc.ume@digitalgraphics.be`) for the original list code; Jean-François Remacle (`remacle@scorec.rpi.edu`) for plane equation fixes.

Projects similar to GL2PS include: Michael Sweet's GLP library (http://dns.easysw.com/~mike/opengl/index.html); Mark J. Kilgard's rendereps (http://reality.sgi.com/opengl/tips/Feedback.html); the GLpr library from CEI international (http://www.ceintl.com/).

5 Versions

0.1 First distributed version.

0.2 Added GL2PS_POLYGON_BOUNDARY and GL2PS_BEST_ROOT. Changed arguments of gl2psBeginPage and gl2psText. Corrected some memory allocation stuff. First version of this user's guide.

0.21 Initialization fixes.

0.3 Code cleaning. Added GL2PS_LINE_STIPPLE.