Epspline: An Editor for *POV-Ray* Prism and Lathe Objects

Ed Hynan

December 15, 2013

Contents

List of Figures

List of Tables

Q: Why is it that the more accuracy you demand from an interpolation function, the more expensive it becomes to compute?
A: That's the Law of Spline Demand.

A UNIX FORTUNE

Introduction

0.1 Purpose

Epspline is a simple, single-purpose program that might be helpful to users of the *POV-Ray* program. For those not already familiar with *POV-Ray*, a brief introduction to *POV-Ray* follows in the next section, and the section following that will introduce epspline.

0.1.1 Brief Introduction to *POV-Ray*

POV-Ray is a *ray-tracing* program with a long history. Ray-tracing is a method for rendering graphical images with a model of optics. Put simply, the user describes objects in space, a viewing position, and sources of light. The program 'traces' the path of a ray of light from the view, among the objects in space, and back to the sources of light. The method is repeated for each pixel of the image that is being generated, and the pixels are colored according to the interaction of the light and the objects in space that it encounters, as seen from the view position defined by the user. The result of this method can be very impressive, and with its long history of active development *POV-Ray* does it very well.

The user of a ray-tracing program must describe objects in space and their attributes. For that purpose *POV-Ray* provides a *scene description language*, or 'SDL.' The SDL allows users to describe objects in numerous ways, all of which are explained in detail, with examples, in *POV-Ray*'s excellent documentation at http://www.povray.org/documentation/. Many of the objects that can be defined in the SDL are easily edited (i.e., written or composed) by hand, but some types of objects are more suited to generation by an interactive program that provides visual editing and feedback. *POV-Ray* is not interactive[1], it works with prepared input files. Fortunately, there are several third-party programs available for

[1]The version of *POV-Ray* for the Microsoft Windows platform includes an interactive text editor and some extra features, but the actual ray tracer still works with prepared input non-interactively.

interactively editing *POV-Ray* SDL objects. Such programs typically produce include files with object definitions that the user will then refer to and arrange in a main scene file.

POV-Ray source code and ready-to–run binaries for some platforms are available at the *POV-Ray*'s web site: http://www.povray.org/. [2]

Figure 1: An image rendered with *POV-Ray*.

0.1.2 Introducing Epspline

Two of the object types that may be defined in *POV-Ray*'s scene description language (SDL) are the 'prism' and the 'lathe.' The purpose of epspline is to provide a

[2] *POV-Ray* is distributed under the *GNU AFFERO GENERAL PUBLIC LICENSE* as of version 3.7, which was announced at http://www.povray.org/ on November 08, 2013.

In previous versions of *POV-Ray*, although the source code is available, it is *not* redistributable in the manner allowed by *free software* or *open source* licenses. It is important that software licenses be respected. The *POV-Ray* legacy license for personal use is at http://www.povray.org/povlegal-3.6.html, and for distribution at http://www.povray.org/distribution-license-3.6.html.

graphical, interactive editor of those objects. The prism and lathe objects are based on interpolated curves, often called *spline* curves. Such curves are defined by the method of interpolation and *control points*. Several sets of control points can be grouped together, and if done well, can compose complex, and *scalable*, shapes. The shapes of the characters on this page are, almost certainly, defined by spline curves in a computer typeface, or 'font.'

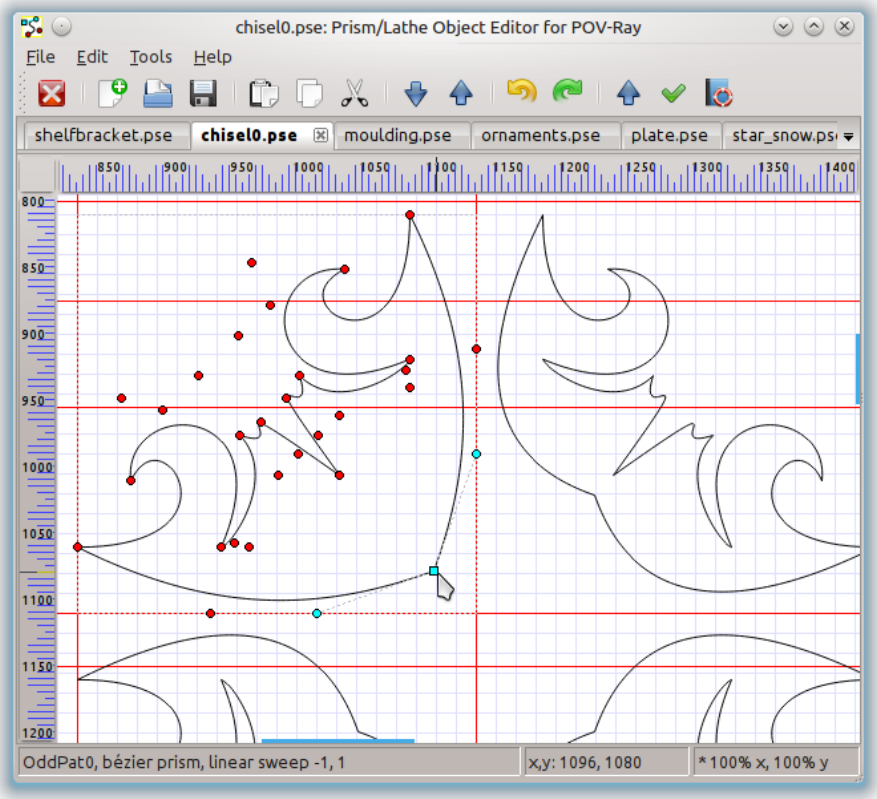

Figure 2: Bézier spline with linked control points selected.

POV-Ray renders spline curves as three dimensional objects. Figure 1 is a sample scene composed mostly of *POV-Ray*'s prism and lathe objects (there is also a 'plane' to provide a background, and of course light sources and a 'camera', which is the point-of–view). The objects arranged in space are chisels. The handles and ferrules of the chisels are *POV-Ray* lathes, and the blades and decorative imprints on the handles are prisms.

The difference between the prism and lathe is that with the former the curve is extruded in the y-direction, and with the latter it is rotated around the y-axis. Making either of these objects in the *POV-Ray* SDL is similar: the control points necessary to define the basic curves are placed, as text, in a description of the object along with other attributes such as texture. That can be done by hand in a text editor for a small number of simple objects, but numerous complex objects would be difficult with hand editing, and without graphical feedback.

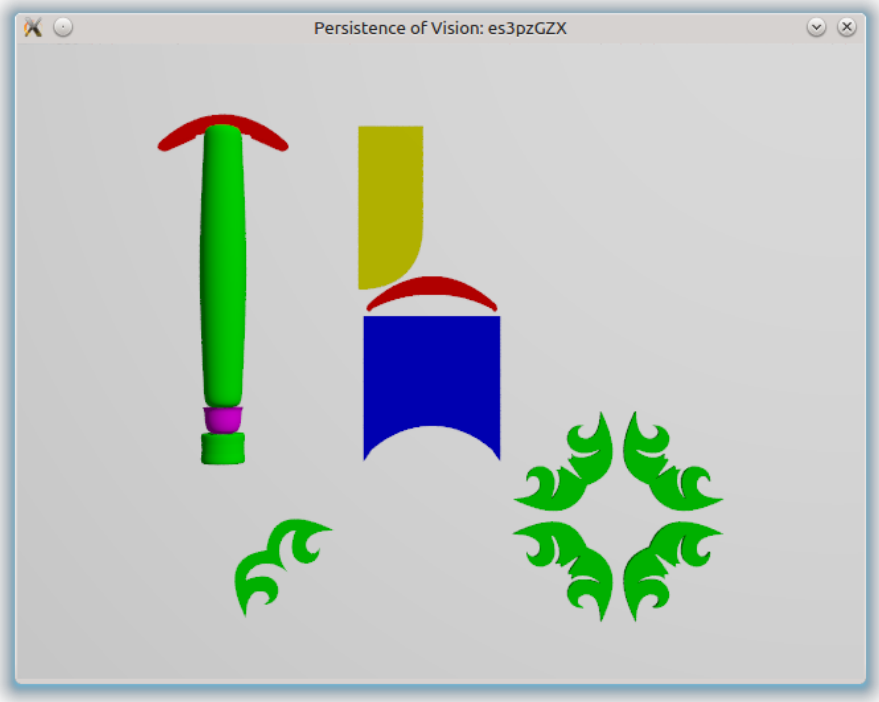

Figure 3: *POV-Ray* preview of the chisel parts.

Epspline lets the user place control points with the *mouse* in sequence to create a shape. Existing shapes can be edited in several ways, and also duplicated, deleted, or transformed. Figure 2 shows a *bézier* spline being edited with the mouse. The cyan colored square is a *selected* control point. The cyan colored circles are control points associated with the selected point and may be moved along with it. The red colored circles show control point positions when they are not selected. The shapes visible in figure 2 were used for the imprint on the chisel handles in figure 1. The truth about the selected control point, the cyan colored square, is that it is two

control points with the same coordinates, and they belong to neighboring curves (which might be called segments if it is easier to think of the whole shape as one curve). This mode of editing should be easier than simply placing numbers in a text editor.

Epspline saves the prism and lathe data in its own type of file rather than *POV-Ray* SDL. To use the objects in a *POV-Ray* scene they must be *exported* to an SDL file that will be included by another SDL file. While working with epspline the current file's objects can be previewed. *POV-Ray* is invoked for this with a simple SDL file that is deleted when *POV-Ray* is closed. Figure 3 shows the sample chisel parts in a *POV-Ray* preview. Some of the shapes seen in the preview are not visible in the scene, but are used for *solid contructive geometry* for features such as the sharp end of the blade and the tang inserted in the handle.

0.2 Requirements and Status

Epspline uses a software library called *wxWidgets* (formerly called *wxWindows*), which provides the window interface, and event model. *wxWindows* features portability across several computer platforms. Currently, epspline can be built for, and has been tested on, the X Window System with the GTK2 toolkit on several Unix-like systems,[3] and Microsoft Windows.[4] Other platforms that are supported by *wxWidgets* have not been tested and more work would certainly be needed to build epspline on those platforms (excepting other Unix-like systems, which might need only small changes if *wxWidgets* is supported). Epspline currently works as expected with *wxWidgets* versions 2.8 and 3.0.

[3] *OpenBSD*, *NetBSD*, *FreeBSD*, *OpenSolaris*, *OpenIndiana*, and *Debian* and *Ubuntu GNU/Linux*.

[4] The *Vista* and *7* releases have been tested. The binaries are built with the *MinGW* tools, and the the cost-free *DMC* tools have been used too, and as of this writing, will still build the code against *wxWidgets* 2.8.

There, before my ravished eye, a Cube, moving in some altogether new direction, but strictly according to Analogy, so as to make every particle of his interior pass through a new kind of Space, with a wake of its own—shall create a still more perfect perfection than himself, with sixteen terminal Extra-solid angles, and Eight solid Cubes for his Perimeter. And once there, shall we stay our upward course?

Flatland
EDWIN A. ABBOTT

Chapter 1

Using Epspline

1.1 The Main Window Interface

The elements of the interface presented by epspline have an arrangement that has been common in windowing systems for many years, and will probably seem familiar to most users. Most individual elements are common, but a few are specific to epspline. A few are similar to common elements in most windowed applications, but with some difference in behavior.

The main groups of interface elements are discussed briefly, from the top down in order of appearance, in the following subsections. An image of the main window is shown in figure 2.

1.1.1 The Title Bar

The *title bar* is not actually part of the interface provided by an application. It is created and managed, if it exists at all, by a component of the windowing system. When present, it is usually similar for most applications. Some windowing systems (or system configurations) do not attach a title bar to the application window, and may use another graphical device such as a single 'bar' at the top of the screen that will change to represent any window that has the *focus*.

Regardless of the way in which a title bar might be presented, an application such as epspline has little control over it. The presence and position of buttons on the title bar, for example, are a feature of the system. Epspline only sets the text of the title.

1.1.2 The Menu Bar

The menu "bar" consists of the labels, arranged horizontally, nearest the top of the window. Each label can be selected with the mouse and will produce a small rectangular window, the *menu*, with selectable labels, or *items*.

Selecting a menu item invokes an action such as saving or closing data. When an item does not make sense for the current state of the data, it cannot be selected, and this is indicated by showing the label "greyed," or dimmed, in appearance.

Many of the menu item labels have additional text, usually near the right-side edge, such as "Ctrl+S." These indicate combinations of keys that will invoke the action associated with the menu item without selecting the menu with the mouse. These key combinations are often called "accelerators."

In the following subsections on the menus, the menu items are presented in tables with short descriptions of the associated actions. Items that differ from common use or are unique to epspline are described in more detail following the table. Also, note that the accelerator keys that are shown may differ from those seen in a running instance of epspline. The current windowing toolkit or environment might provide accelerators for some common menu items, and epspline might use them.

The File Menu

This menu provides items that generally apply to the current data, or "file."

ITEM:	KEYS:	ACTION:
Save	Ctrl+S	save data with current name
Save As	—	save data with a new name
New	Ctrl+N	create a tab for new data
Open Here	—	open (load) file data in current tab
Open	Ctrl+O	open (load) file data in a new tab
Close	Ctrl+W	close the data (file), leave tab
Close Tab	—	close the data (file), close tab
Export	Ctrl+E	save the data as *POV-Ray* SDL
Export As	—	save as SDL with a new name
Quit	Ctrl+Q	quit the epspline program

Table 1.1: The Items of the File Menu

The unusual items that need more explanation are:

New The action is to create a new tab in the tabbed interface, without contents, suitable for creating a new work. Alternatively, a file can be opened in the

new empty tab with "Open Here."

Open Here Open a file and load the data into the current tab. If the tab contains data with unsaved changes the user will be asked whether to proceed (which will close the current file, discarding changes) or cancel.

Close Tab This will close the file data *and* the tab, while the item "Close" will only close the file data, and leave the tab open and empty. If there are unsaved changes the user will be prompted.

Export Epspline saves files in its own simple format; to use the data with *POV-Ray* it must be "exported" to SDL. This item will export the data to a file with an existing name that had be given with "Export As."

Export As Like "Export" but shows a file selector dialog window so that the user may provide a file name. This is necessary the first time the file is exported; thereafter "Export" can be used to overwrite the current file with changes. (Note that the file name suffix '.inc' is commonly used for such SDL fragments, but it is not required by *POV-Ray*. It is possible to set a preferred suffix in the Preferences dialog, *POV-Ray* Settings section.)

The Edit Menu

This menu provides items that affect or change the current data, or in the case of *preferences*, display or application settings.

The unusual items that need more explanation are:

Copy, Cut, Paste These item names are common in windowing system programs, but in epspline data is *not* placed on a system "clipboard," but is only available within epspline. Moreover, data copied or cut by these items can only be pasted into the same tab.

Copy, Cut, Paste — Global These differ from the items without "Global" only in that data copied or cut from one tab is available to be pasted into the data of another tab.

Down, Up The drawing area may have several objects, and these are kept in a stacking order, or "z-order." This may not be apparent in the two-dimensional interface, but will be noticed when the user attempts to select an object that is below another. In such a case, it might be necessary to move the obstructing object down in the order to make the wanted object selectable. Note that there is one z-order for all objects, even those that don't overlap, so it

ITEM:	KEYS:	ACTION:
Undo	Ctrl+Z	revert the last change made
Redo	Ctrl+R	do again a change that was reverted
Copy	Ctrl+C	copy the selected object onto internal "clipboard"
Cut	Ctrl+X	remove, after copying the selected object onto internal "clipboard"
Delete	—	remove selected object or point
Paste	Ctrl+V	place an object from internal "clipboard" in the data
Copy Global	—	like "Copy", onto a "clipboard" available to all tabs
Cut Global	—	like "Cut", onto a "clipboard" available to all tabs
Paste Global	—	like "Paste", from a "clipboard" available to all tabs
Down	—	move the selected object down in z-order
Up	—	move the selected object up in z-order
Preferences	—	invoke the dialog window for setting preferences

Table 1.2: The Items of the Edit Menu

might be necessary to select "Down" several times (and on more than one object) before the wanted object is no longer obstructed. If the wanted object can be selected, but only around the edges of an obstructing object, then "Up" can be used to raise the selection and make it easier to select with the next attempt. Note also that the stacking or z-order has no effect on the way *POV-Ray* will render the objects, and will only affect the order in which the objects appear in exported SDL.

Preferences This item has its own section, Preferences.

A final note on the "clipboard": it's an omission that the system clipboard is not used. It would be useful if a selection could be copied (or cut) to the system clipboard as a SDL fragment which could be pasted into a text editor. Likewise, an attempt could be made to paste from the system clipboard, if in text form, by trying to parse a prism or lathe definition in SDL. This feature might be added in the future if epspline warrants more work.

The Tools Menu

This menu has miscellaneous items that are specific to epspline.

ITEM:	KEYS:	ACTION:
Set Scale	—	show a dialog to set arbitrary scale of display
Cycle Scale	—	cycle scale of display through normal, large, and small
Toggle Quick Drawing	—	switch between anti-aliased, and not, outlines
Save Visible to File	—	save the visible part of drawing area to an image file

Table 1.3: The Items of the Tools Menu

The Tools menu items all need explanation:

Set Scale The view of the drawing area can be displayed with a scaling factor. This feature uses a capability provided by the *wxWidgets* library; that is, epspline is not drawing lines at different scales, but the view in the window is scaled. Therefore, with a large scale lines (and all drawn items) will appear thicker, and details such as anti-aliasing will become apparent. With a small scale, some lines of one-pixel–width are lost. Nevertheless, this feature can

be useful at times. Generally, most work will be easiest at 100% (normal) scale.

Cycle Scale This is a quick way to rotate through 100%, 200%, and 50% scales, and is probably more generally useful than "Set Scale."

Toggle Quick Drawing The object outlines, the curves, are drawn by default with anti-aliasing. The alternative drawing method, the "quick" method, simply uses several simple lines drawn with *wxWidgets*' graphics *API* calls to approximate the curve forms. The latter is not likely to be useful unless the drawing area is on a display to which raster image data transfers are slow. Most modern computers are much faster than necessary for the anti-aliased drawing if the the display is on the same machine that is running epspline. If the display is not on the same host that is running epspline, the anti-aliased drawing might be too slow to be usable (particularly over a slow link such as 802.11). That is because the anti-aliasing must be done on a raster image, and the whole image, which is a large chunk of data, transferred to the display. With "quick drawing" the API line drawing is accomplished with small data transfers, and will be quicker, and probably usable (but less accurate in appearance).

Save Visible to File The part of the drawing area that is visible within the *scrolled* window can be saved to a raster image file. This is (the remains of) a development feature, and as it is difficult to imagine how it might be useful to users, it is likely to be removed at some point. It remains, for now, mostly because it is harmless and adds very little size.

Note on "Toggle Quick Drawing": above, it was said that anti-aliased drawing is the default. There is an exception under the *X Window System*, where the "DISPLAY" environment variable is checked for a host component, i.e., "somehost:0.0" rather than simply ":0.0". If a host part is found, even if it is *localhost*, the default will be switched to "quick" (not anti-aliased) drawing, on the obvious assumption that the display is remote (or remote-like). Of course, the user may still select "Toggle Quick Drawing" and decide whether performance with ant-aliasing is acceptable.

The Help Menu

This menu appears in most windowing system applications, and provides items that might be helpful in some way.

Additional explanation:

ITEM:	KEYS:	ACTION:
Help	Ctrl+H	show the epspline online help mouse and key reference
Help Contents	—	show the epspline online help Table of Contents
Preview	—	invoke *POV-Ray* with temporary SDL to view the work
About	—	show the "About" window

Table 1.4: The Items of the Help Menu

Help This will show (or bring to the forefront, if already shown) a *help viewer* window showing a reference to mouse and key inputs and their actions. (The help viewer is a non-trivial piece of software provided entirely by *wxWidgets*, and much to its credit — developers take note.)

Help Contents This will show (or bring to the forefront, if already shown) a help viewer window with the "Contents" page visible. (Since the reference, above, might be more useful to a familiar user, that appears first, and is also invoked from The Tool Bar.)

Preview The *POV-Ray* preview does not attempt to make a very nice rendering. It does attempt to make an image with a perspective that includes all objects. Only a few simple colors are used. Although crude, the preview will often be useful. An important thing to note is that epspline does not delete the temporary SDL files and output until the running *POV-Ray* has been quit. If epspline is quit while a *POV-Ray* child is still running, temporary files are left in place.

About The "About" window includes copyright and license information.

A further note on the preview: *POV-Ray* for Unix-like systems is based on a command-line interface, but can also draw in real-time to a simple *X* window, and this suits the preview well. *POV-Ray* will quit with a single click on its window, or the *Q* key. *POV-Ray* for Microsoft Windows is quite different. That version will show an interactive control and editing window before the real-time display window, and might prompt the user with several dialog windows too (and might even play a little music). It is a nice tool for developing *POV-Ray* images, but it is a little awkward when used for a simple preview by another program. Nevertheless, the preview remains useful. The Microsoft Windows version of *POV-Ray* is quit

with a menu or button; not a click. The same caveat about quiting epspline with a child *POV-Ray* instance running applies.

1.1.3 The Tool Bar

The *tool* bar just below the menu bar. It is a row of buttons with icons that can be clicked to invoke actions. Each action available on the tool bar is equivalent to a menu item action. Naturally, the tool bar will be more convenient while working than the equivalent menu items. From left to right, the tool bar items are:

Quit Same as "Quit" in the "File" menu.

Open New Tab Same as "New" in the "File" menu.

Open Same as "Open" in the "File" menu.

Save Same as "Save" in the "File" menu.

Paste Same as "Paste" in the "Edit" menu.

Copy Same as "Copy" in the "Edit" menu.

Cut Same as "Cut" in the "Edit" menu.

Move Down Same as "Down" in the "Edit" menu.

Move Up Same as "Up" in the "Edit" menu.

Undo Same as "Undo" in the "Edit" menu.

Redo Same as "Redo" in the "Edit" menu.

Cycle Scale Same as "Cycle Scale" in the "Tools" menu.

Preview Same as "Preview" in the "Help" menu.

Show Application Help Same as "Help" in the "Help" menu.

Figure 1.1: Epspline's tool bar.

Figure 1.1 shows the tool bar with its buttons. The icons shown are from a *desktop theme*, and may be different for each theme, in the GTK2 version of epspline.

In other versions a set of icons included with *wxWidgets* is used. Depending on the current platform, it might be possible to grab the tool bar and drag it off the main window, and, of course, back onto the window.

1.1.4 The Tab Selection Area

The *tab* area is between the tool bar and the drawing area. There may be one or several tabs present. Epspline uses tabs to allow holding several sets of data, or files, open at once. Each tab functions like a button: when the tab is clicked the associated data is brought to the front (i.e., shown). Each tab has a label with the name of its file, or symbolic text meant to suggest that the data has not been saved to a file yet (or the tab is empty).

As of *wxWidgets* version 2.8 two forms of tabs are available: a simple type available in previous versions, and a new type with more features. The simple type does not provide a close button on the tabs, and the order cannot be rearranged. The new type does provide for close buttons, and a tab can be dragged with the mouse to a new position among the others. The new type also provides for a button at the right side of the tab area, which produces a menu of the available tabs for direct selection; this is convenient when there are many tabs and some cannot be seen due to lack of space.

Epspline uses the new type of tabs by default (except when built with *wxWidgets* 2.6). There is little reason to use the simple type, and to do so requires setting a condition when the program is built from source (which is left as an exercise for those so inclined).

1.1.5 The Drawing Area and Canvas

This is where the work is done. At the top and to the left are graduated scales, and as the mouse is moved in the *drawing* area a thin line is drawn on the scales to help locate the mouse active point with precision. To the right and at the bottom are scroll bars to move the visible part of the drawing area.

The ‘real’ drawing area may also be called the *canvas*. The canvas has a white background with a grid of thin blue lines. This is meant to suggest physical *graph paper*. (In the future there may be an option to disable the grid; there is none now).

1.1.6 The Status Bar

The *status* bar is at the bottom of the main window. It accepts no input; it exists only to provide information.

The status bar has three sections, or "panes." The left, and longest, pane shows details about a selected object, such as its name and type, and if it is a prism, the type and extent of its *sweep*. The left pane also shows temporary help messages associated with tool bar or menu items. Such help messages are shown when the mouse is over the item, and any original text is restored when the mouse leaves the item. The pane in the middle shows the coordinates of the mouse pointer (the pointer's active spot). The right pane shows the scale at which the drawing area is displayed. Also, if there are unsaved changes to the data, the right pane will show an asterisk.

1.2 Creating and Editing

Working with epspline is working with spline curves. Spline curves are common to many programs of several types, such as 'drawing' programs and typeface editors. Of course, any such program must design a means of using an input device, usually a *mouse*, to place splines on the drawing area. Programs meant for drawing might devise powerful and intuitive mouse interactions that focus on a graphical result. Epspline does not. In epspline, placing spline objects on the drawing area is meant to follow the *POV-Ray* documentation on lathe and prism objects.

As mentioned in Introducing Epspline, there is an important difference in the ways *POV-Ray* will render prisms and lathes. The prism is extended (like an "extrusion") along the *y*-axis. Therefore, the spline object for the prism is drawn parallel to the *x*-*z* plane in three-dimensional space. The lathe is rotated *around* the *y*-axis. So, its spline object is drawn parallel to the *y*-axis. Put another way, when editing with epspline, if the object is a prism then the user's *view* is parallel to the *y*-axis, and perpendicular to the *x*-*z* plane. If the object is a lathe, the view is perpendicular to the *y*-axis (and *x*-axis, even though the lathe is rotated).

Editing with epspline, the user may mix prism and lathe objects in the same file, but should remember that the *view* of each type is different. When generating temporary SDL for a preview with *POV-Ray*, epspline adds a rotate transform to the prism objects so that the preview will be more useful. Epspline does not rotate objects in SDL exported for inclusion in the user's scene, so the user must add rotations as necessary.

The coordinate system of the drawing area in epspline is, frankly, simply the most convenient for the programming of epspline. The graduated scales beside the drawing area will make it clear that the vertical axis increases in the down direction. This is easily corrected using transforms available in the SDL, and will be discussed in the section Useful Transforms.

Hopefully, the following subsections will be clear enough to get started and

begin realizing ideas. Where this document is insufficient, the *POV-Ray* documentation should help; it is necessary to understand how *POV-Ray* handles the prism and the lathe objects. Although the following subsections are not written as a step-by–step tutorial, it should be possible (and it is suggested) to follow along with a running instance of epspline.

1.2.1 Creating New Objects

Started without file arguments, epspline's main window will open with one empty tab. The drawing area (the area with the grid of light blue lines) is ready, but note that it doesn't respond to a (primary) mouse click alone. To start a new spline, press and hold the *shift* key and click a suitable point with the primary mouse button. For practice, a "suitable" point might be anywhere. A small red-filled circle should appear at the point of the mouse click, and that is the first control point. Release the shift key to continue placing control points. Another shift+click before completing at least one spline *segment* will cancel, and the control points placed so far will be removed. (Recall from the "Introduction" chapter that the spline objects are really several splines taken together as a "path"; from here on "segment" will refer to one spline part of the whole spline object.) Once at least one segment is complete, another shift+click will stop editing the object, and the points placed so far will remain. It will be clear that at least one segment is complete when a curve is drawn between some of the control points.

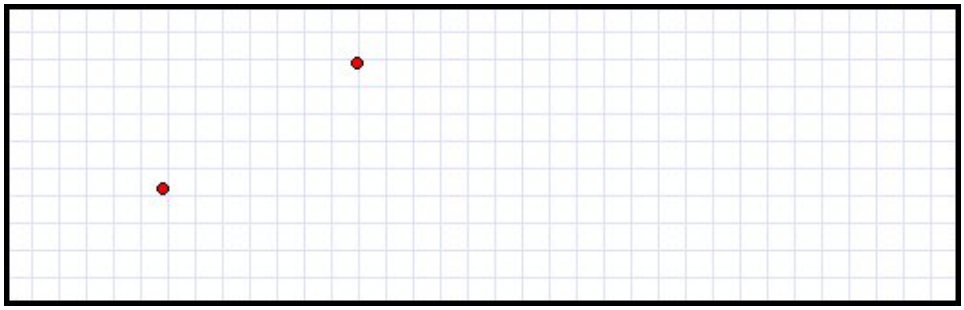

Figure 1.2: First two control points starting a cubic spline.

Figure 1.2 shows the start of a spline object before a segment is complete, and figure 1.3 shows the first curve appear after two more control points have completed the segment. The object in the figures is a *cubic* spline, which requires four control points for a segment. (The number of control points per segment differs among the four available spline types.)

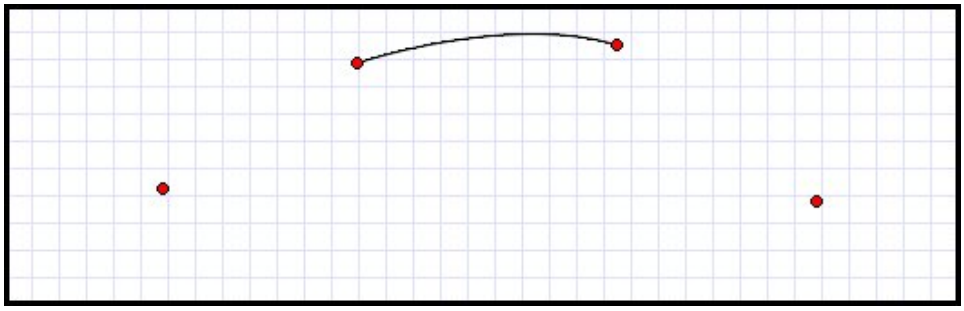

Figure 1.3: Curve appears between control points of a cubic spline.

If shift+click is given at the stage of figure 1.2, the two points will be deleted, but at the stage of figure 1.3 the points will be retained, and editing mode stopped. Editing is continued if the object is selected with a click, and shift+click given again.

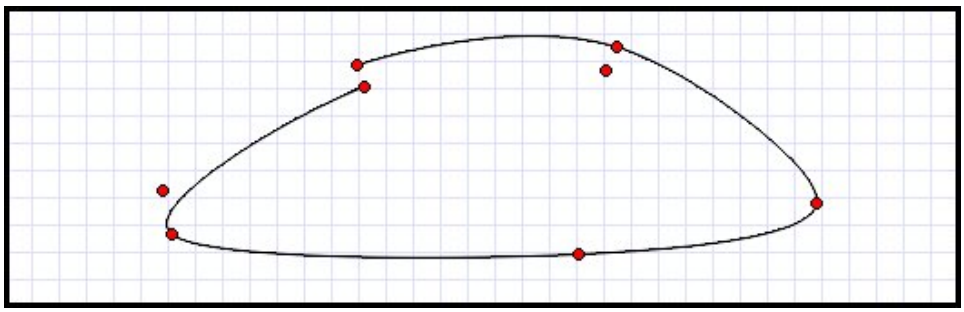

Figure 1.4: Cubic spline, not closed.

Figure 1.4 shows the spline object nearly complete. More points have been added, and another shift+click has terminated editing. *POV-Ray* would reject the object as it is because the curve is not closed, and epspline will not export it to SDL.

In figure 1.5 the object has been selected with a click. The selected state state is indicated by the dotted rectangle (on some platforms the rectangle might be dashed). With the whole object selected, a control point at an open end was selected, indicated by the cyan colored square, and moved to coincide with the matching control point. Now that it is properly closed, *POV-Ray* will render the object. Also note that the two points that closed the curve are not the first or the last, but rather the second and the next-to–last. With the cubic spline curve, those are the points that must match, and the first and last are not drawn through. This differs in

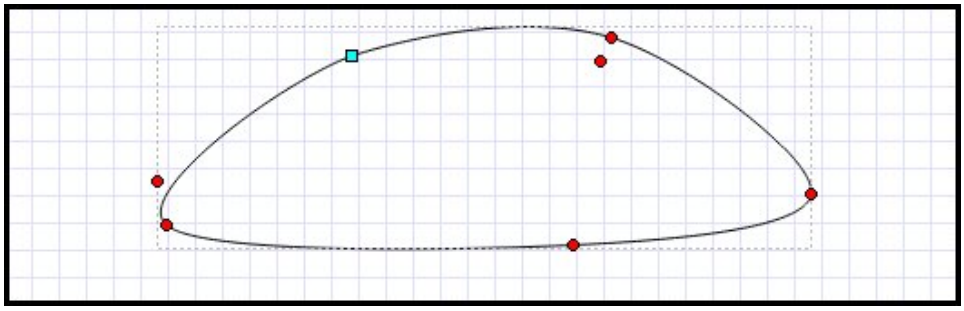

Figure 1.5: Cubic spline, closed and complete.

the other spline types. *POV-Ray* requires a minimum of six points in a cubic spline (epspline will draw a cubic spline with five points if points two and four coincide, but *POV-Ray* will reject the object).

The *POV-Ray* documentation on the lathe and prism objects will explain in more detail how points should be placed for each spline type, and other characteristics of the types.

The cubic type of spline has been discussed so far because it is the default type for new objects if another type has not been selected. To choose among the four spline types, make sure that no existing object is selected by clicking the drawing area, outside of the selection rectangle of any selected object, with the primary mouse button. Next, click the drawing area with secondary button, which will invoke a "pop-up" menu. The drawing area menu is seen in figure 1.6.

The first (topmost) menu item "Curves" names the menu, and also serves to dismiss the menu without making a selection. The menu items beginning with "Set ..." select the type of object to be created. The first four of these offer the spline type, and the next two offer the *POV-Ray* object type: prism or lathe. The last four items on the menu are equivalent to the "File" menu items with the same labels (table 1.1), and are placed on this menu for convenience. Note that an existing object can be changed between a lathe and prism, so it's not necessary to make that choice initially, but an existing object cannot be changed to a different spline type, so a type should be chosen before starting each object.

Figures 1.7 and 1.8 are similar to figures 1.4 and 1.5, but show a quadratic spline. In this case the points that must match to close the curve are the second and the *last* (rather than the second-to–last point in the cubic type). *POV-Ray* requires at least five points in the closed quadratic spline.

The equivalent for the linear spline type is seen in figures 1.9 and 1.10. There must be at least three points in the linear spline.

The bézier spline object must be closed too, of course, as seen in figures 1.11 and 1.12.

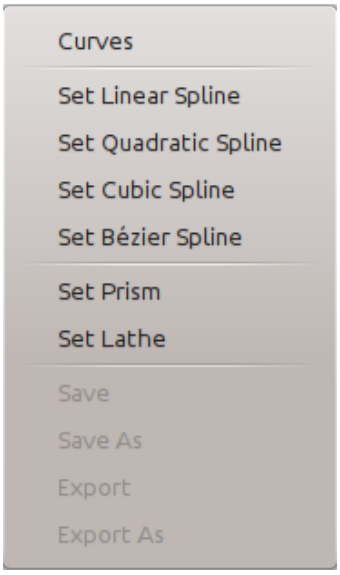

Figure 1.6: The drawing area "pop-up" menu.

POV-Ray will render a bézier spline object with only four points, which is one complete segment, if the first and last points coincide. The figures show a bézier object with two segments.

The Points of the Bézier Type

Quoting the start of section 1.2, "In epspline, placing spline objects on the drawing area is meant to follow the *POV-Ray* documentation on lathe and prism objects." That means that a point must be placed on the canvas in the same manner that a *vector* would be added to a prism or lathe when editing SDL, with the same requirements of order and position. Epspline does not add missing points or automatically close curves; but, a little help is provided when the graphical interface makes it necessary: when two points are clearly meant to coincide, epspline may make them do so, because such precise placement with a mouse is very difficult. In general, a curve or *sub-curve* must be closed with the user's knowledge of which points must coincide.

The bézier spline type is discussed here in more detail because:

- It differs from the other types more than they differ from each other.
- Regardless of the difference, mastering the bézier might make mastering the other types easier.

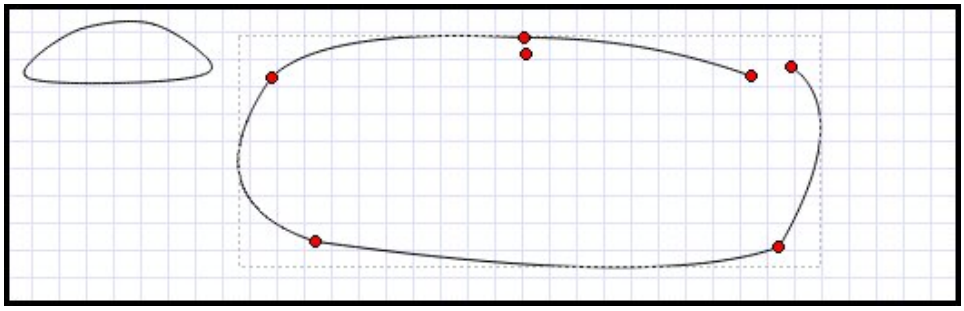

Figure 1.7: Quadratic spline, not closed.

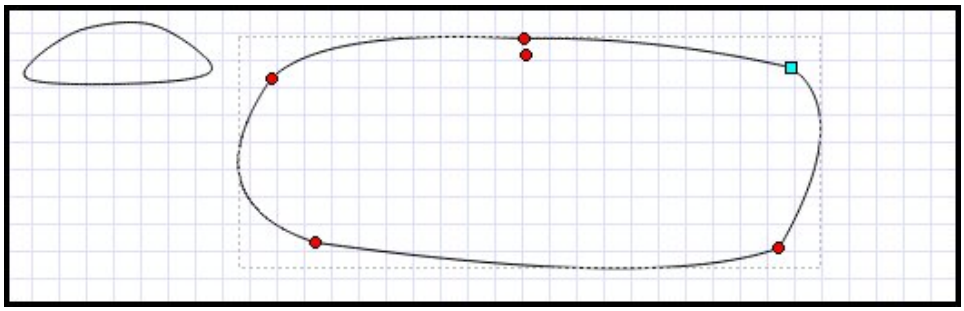

Figure 1.8: Quadratic spline, closed and complete.

- The bézier might be the most useful type [in the author's opinion that is so].

To begin a new object, and to stop editing it, is accomplished the same way for all the spline types. (If that was not made clear above, look forward to the New Object Summary section.) Once a bézier spline object has been started it *must* be remembered that the whole curve is composed of segments, that each segment is composed of four control points, that the segments must be joined so that the last (fourth) point of the previous segment coincides with the first point of the next segment, and that the whole curve must be closed by placing the last point of the last segment coincident with the first point of the first segment of the curve. (A bézier spline object may consist of a single segment, in which case the first and last points of this one segment must coincide.) Therefore, after first starting a bézier curve, place the second, third and fourth points where they make sense, but do not move the mouse after placing the fourth point. Click again at the position of the fourth point to begin the next segment, and its first point will coincide with the last of the previous segment, as it should. Finally, close the curve by placing a segment's last point just at the point that started the curve.

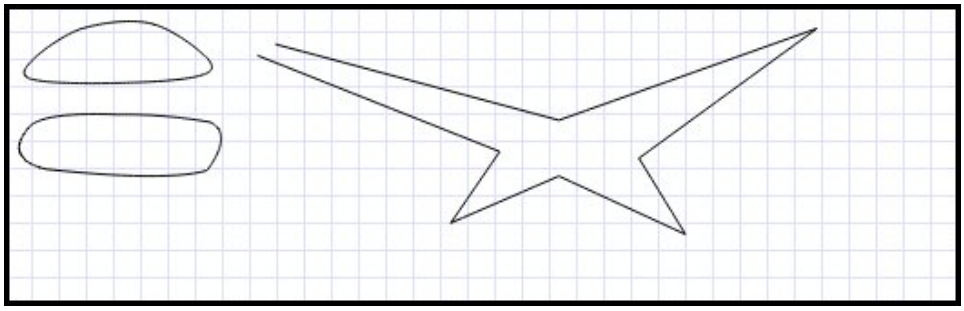

Figure 1.9: Linear spline, not closed.

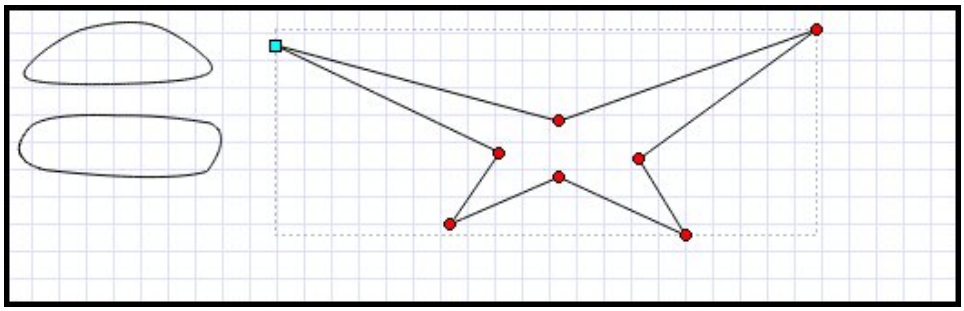

Figure 1.10: Linear spline, closed and complete.

Although epspline does little to enforce these requirements, *POV-Ray* will enforce them. Epspline does provide a little help: if the fourth and first points of two segments in sequence were placed offset by a pixel or two, epspline will join them, so a slight movement of the mouse before placing the first point of a new segment is not an irrecoverable error. If two points that should coincide are separated by more than a couple of pixels, epspline will not join them, but they can be brought together with subsequent editing, i.e, after the initial creation of the object has been finished with another shift-key+primary-click.

Some trial and error practice should make the points of this discussion more clear. (The "Undo" facility will always be very helpful.) After a few successful bézier creations, try the other spline types and discover how they differ.

New Object Summary

As a summary of the creation of a new prism or lathe, consider this step-by–step list:

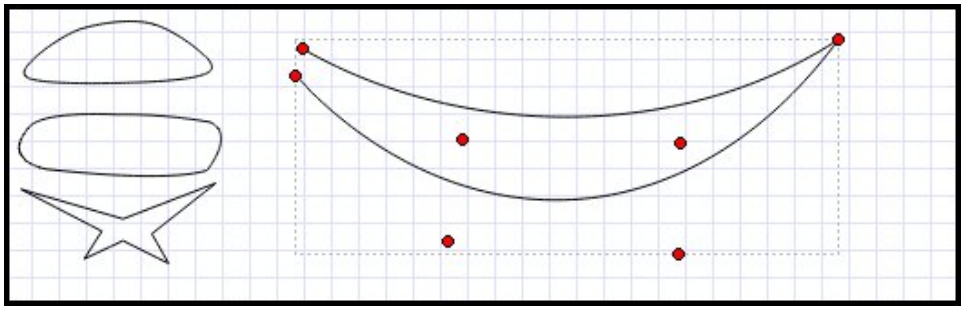

Figure 1.11: Bézier spline, not closed.

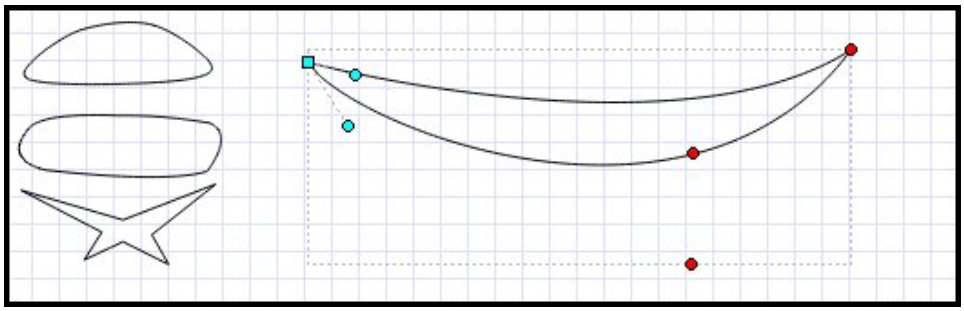

Figure 1.12: Bézier spline, closed and complete.

1. If any current object is selected (a dashed or dotted box is shown and control points are shown with circles), deselect it by clicking the drawing area outside the selected object's bounding box.

2. Set the type of spline to be created by clicking the drawing area with the secondary mouse button and choosing from the menu, as shown in figure 1.6. Do not forget this step because the spline type of an existing object cannot be changed.

3. Optionally, choose whether the new object will be a prism or lathe from the drawing area menu. This can be changed for an existing object, so it is not necessary initially.

4. Enter editing mode: with the shift-key depressed, click the canvas with the primary mouse button at a suitable place for the first control point, and a circle representing the point should appear.

5. Release the shift-key before clicking the canvas again, or the object will be

cancelled. Of course, the object may be cancelled intentionally.

6. Click to add points, in a manner suitable for the type of spline being created, until the object is complete, or at least until a curve segment is drawn between points (so that the object will not be cancelled and will remain for further editing).

7. Leave editing mode with a primary mouse button click with the shift key depressed.

1.3 Continued Editing

An idea of a form will suggest where points should be placed, but usually the curve that appears is less than ideal when first created. In fact, placing points with mouse clicks, as described so far, is just a rough start. If an idea of an object for a *POV-Ray* scene can be implemented as a *POV-Ray* prism or lathe, then epspline might help *develop* the object. That is, making any but the simplest objects will be a process of changes and tuning. In this process, the interactive graphical interface should be an advantage. The use of the facilities provided by epspline are discussed below.

1.3.1 Object Properties

To this point prisms and lathes have been referred to repeatedly, but there has been little, if any, mention of their properties. Moreover, it was stated that although an object's spline-type cannot be changed after it's created, it can be changed to a prism or lathe.

The properties of objects in epspline are changed in a *dialog* window, "Spline Properties," which is invoked through an object menu. This menu, seen in figure 1.13, will 'pop up' in response to a secondary button click on a selected object. (If no object is selected, the secondary button click will produce the drawing area menu.)

The object menu has four items (after the label, "Selection"). The last three items are equivalent to similarly named items on the edit menu (see The Edit Menu). The first item, as its name suggests, invokes the "Spline Properties" dialog, shown in figure 1.14.

The "Spline Properties" dialog fields are as follows:

(label) A label stating the type of spline.

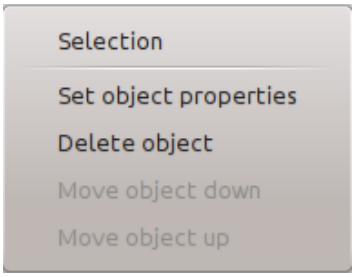

Figure 1.13: The object "pop-up" menu.

Object Name A text entry field: a name for the object, suitable for *POV-Ray* SDL, should be entered here before export. The text field will reject any characters that *POV-Ray* does not accept in names, and if the name clashes with a *POV-Ray* reserved word, a prompt for a new name is displayed.

Render Type A selection of rendering options: as lathe, as prism, or "Undefined," which means that it will not be rendered; this can be used to hide an object in a *POV-Ray* scene while keeping it in the epspline file. This "Undefined" type must be given a name too, because it is exported to SDL as a *POV-Ray* array (which might be useful).

Sweep Type This applies to prisms. The sweep type is described well in the *POV-Ray* prism documentation. Note that presently a conic sweep prism is best if moved so that its center occurs at the origin of the epspline drawing area.

Use Sturm, Open These are optional. Sturm selects a more accurate but slower algorithm; unless you know that you need this, you probably don't. Open applies to the prism: the flat faces will not be rendered, leaving an open outline.

Start, End Sweep This applies to prisms: roughly, the offset and height of the prism.

Texture, Interior, Transform Text entry fields that can accept a name for the attribute that must be '`#declare`'ed before the exported SDL is included in a scene. These fields appear with a default string (until changed) that is not applied to the object — the default is only a place-holder. These attributes can probably be applied more flexibly in the using SDL.

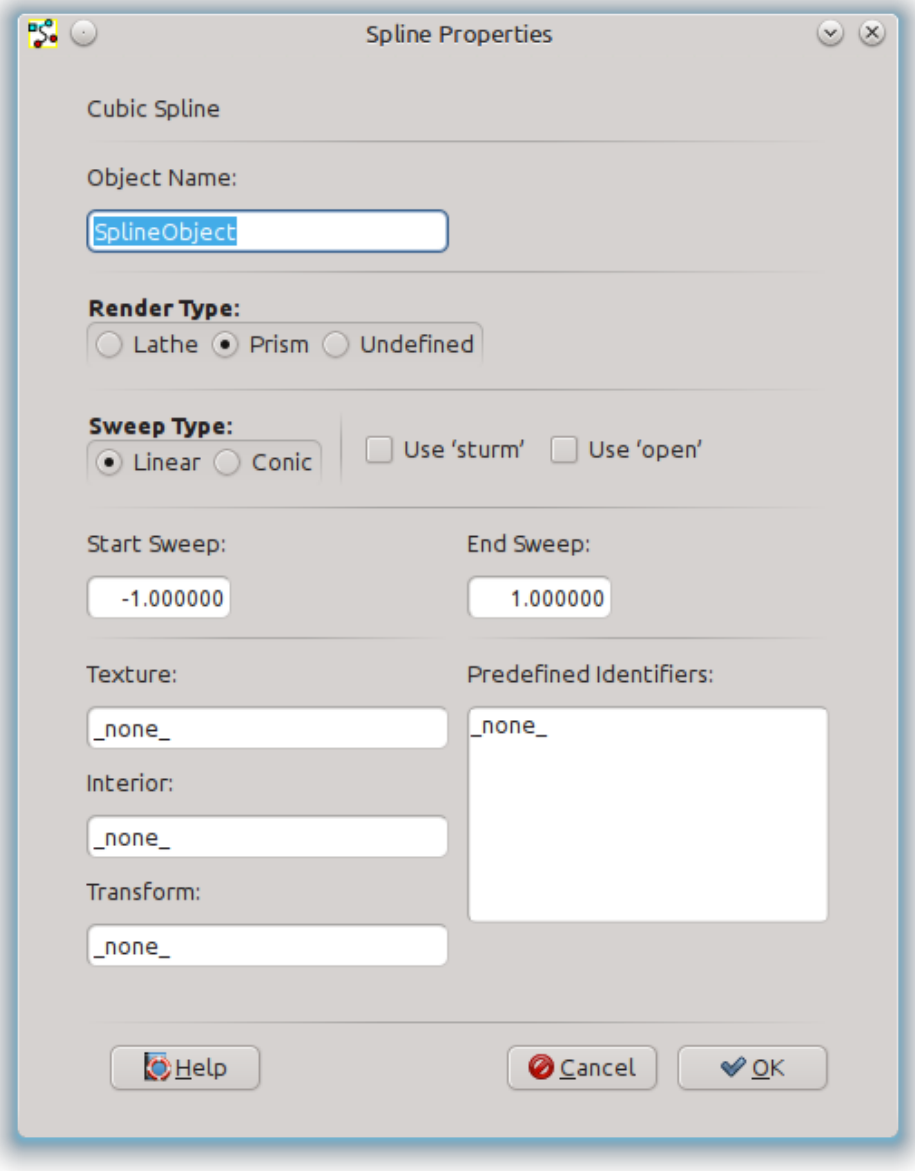

Figure 1.14: The properties window.

Predefined Identifiers This text entry field can accept more than one line. It is poorly named because it is not restricted to identifiers; for example, a comment may be added here. The most important point about this field is that it is not checked at all by epspline — the user must be certain that anything added here is valid SDL. The contents will be added near the end of the object definition.

(buttons) For "OK" and "Cancel," dismiss the dialog window, either effecting or cancelling any changes that were made. For "Help," dismiss the dialog window with changes lost and display the help viewer showing this section.

Obviously, some object properties should be set before export to SDL. In particular, each object should be given a name.

1.3.2 Epspline Transforms and Copies

The transforms provided by *POV-Ray* SDL will be necessary in any any scene, and some that pertain to epspline output are discussed in Useful Transforms. Epspline provides some interactive transforms that should be useful while developing objects. These are applied directly to the control point values rather than as exported SDL statements.

Objects may be moved. Click an object to select it, and then *drag* it with the mouse, and release it in the desired position. Single-pixel movements can be made with the arrow keys. Larger movements can be made with the page up and page down keys; if the control (or command) key is down then the page-key movements will be horizontal. Note that to move the whole object with keys, make sure that no control point is selected, or the point will be moved.

Objects may be scaled, sheared, or rotated. To enter the mode in which these transforms are done, first select the object, then hold the shift key down and click with the secondary mouse button. An object selected for these transforms is seen in figure 1.15. To leave this mode, another secondary click, but *without* the shift key down, will return the object to the simple selected state. Repeated shift-key+secondary-clicks cycle through the scale, shear, and rotate modes, in order.

In transform mode small squares are shown just inside the object's bounding box, at the corners and at the middle of the sides. These squares are *handles* for manipulating the object: drag the handles to effect the transform.

In figure 1.15 two small concentric circles can be seen at the middle of the object. These circles are shown for shearing and rotation, and may be dragged with the mouse to a new location. The transform will be centered at the circles. They are not shown in the scaling mode, which uses no center as such.

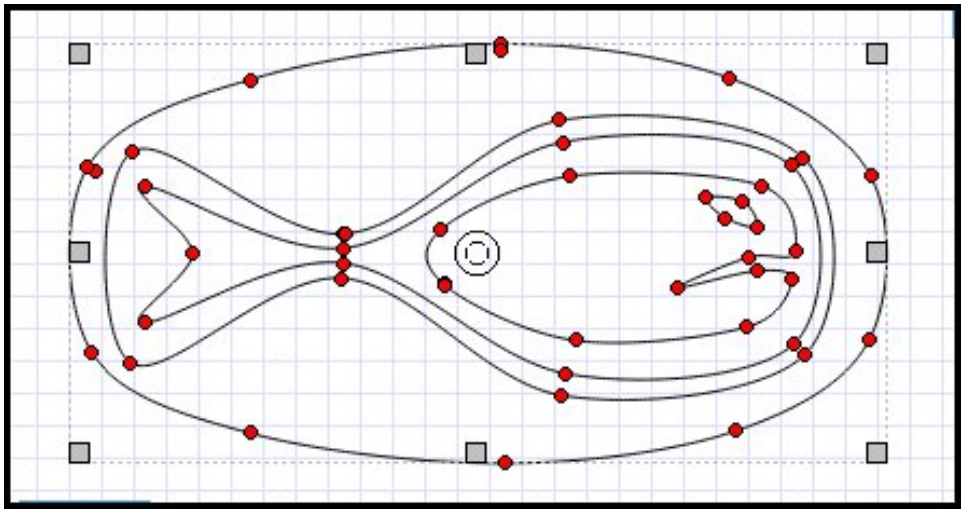

Figure 1.15: Selected object in transform mode.

For each of the scale, shear, and rotate modes the shape of the mouse pointer will change. The pointer shapes should be suggestive of the mode (in some GTK themes of the last several years the pointer shapes no longer seem to make sense, but the user should still form an association between the pointer shapes and the modes).

Each transform, effected by dragging a handle with the mouse, is modified by one or more keys. If the shift key is held down, the shear and rotate transforms will be constrained to a 15 degree angle (presently, this angle can only be changed as a compile-time macro, but may be made configurable in a future release if epspline warrants more work). The scale mode transform, with the shift key, will become instead a *flip* of the object horizontally, vertically, or diagonally according to the handle used. Also with the scale transform, if the 'alt' key is held the transform will be directly proportional when using the upper-left or lower-right handle-squares, or inversely proportional with upper-right or lower-left handles. The alt key has no effect on scaling with the middle-of–the–sides handles, or on the shear and rotate transforms.

A copy of the original, un-scaled object is left if the control (or command) key is down at the start of any transform. To make a copy that does not require a transform, simply make a clipboard copy with the edit menu (see The Edit Menu) or tool bar (see The Tool Bar) and paste the copy back. Whenever a copy is made, remember to make sure that each has a unique name (this is not done automatically, at present; see Object Properties).

1.3.3 Adding, Moving, and Deleting Points

Points can be added to an existing object for two purposes: as part of a new discontinuous sub-curve, or as part of an existing curve. The former case is discussed later in the section Discontinuous Objects.

To add a point to an existing curve, first make sure the object is selected. Place the mouse active point *on* the curve at the desired position. Hold the shift key down and click the primary mouse button. If the mouse pointer was properly on the curve, a new point will appear as a cyan colored square. If a new point appears as a red circle, the mouse active point did not lie on the curve, and the point should not be left in place. The errant new point can be removed with "Undo" on the edit menu (see The Edit Menu) or tool bar (see The Tool Bar), or the *control*+*z* key combination.

A 'point' is added to the bézier curve in the same way, but the actual result is that the current segment is divided in two at the mouse point, and several necessary points are added. Some effort is made to place new *tangent* control points sensibly, but adjustment will be needed.

Moving points should be obvious: select the point so that it appears as a cyan colored square and drag it with the mouse. Only the bézier curve needs more explanation. The section "The Points of the Bézier Type" explained that the last and first end-points of neighboring segments are (must be) coincident. Epspline actually maintains an association of four points: the last and first tangent control points and the end-points of neighboring segments. When any of these is selected the whole group will be colored cyan, and the selected point will be square. Within this group, dragging a tangent control point will also move the other tangent point onto a line that passes through the endpoints; this produces a smooth curve through the end-points. A tangent point can be moved independently by pressing the control (or command) key, making an angle at the end-points. The end-points can be dragged, and they are always kept together as one. Dragging the end-points will also make an equivalent movement of the tangent points, causing a movement of the whole group; the end-points can be moved without affecting the tangent points by pressing the control (or command) key.

Selected points may be deleted with the delete key or the edit menu "Delete" item. In the case of the bézier curve a whole segment is deleted, and remaining segments are joined at end-points. Adjustment will be needed.

1.3.4 Discontinuous Objects

Figure 1.15 was presented to show a "Selected object in transform mode." It also shows a spline object that is more elaborate than any in the earlier figures: the sin-

gle object shows several closed sub-curves comprising the whole object. *POV-Ray* handles this well, as seen in figure 1.16 (which is a *POV-Ray* preview; see The Help Menu).

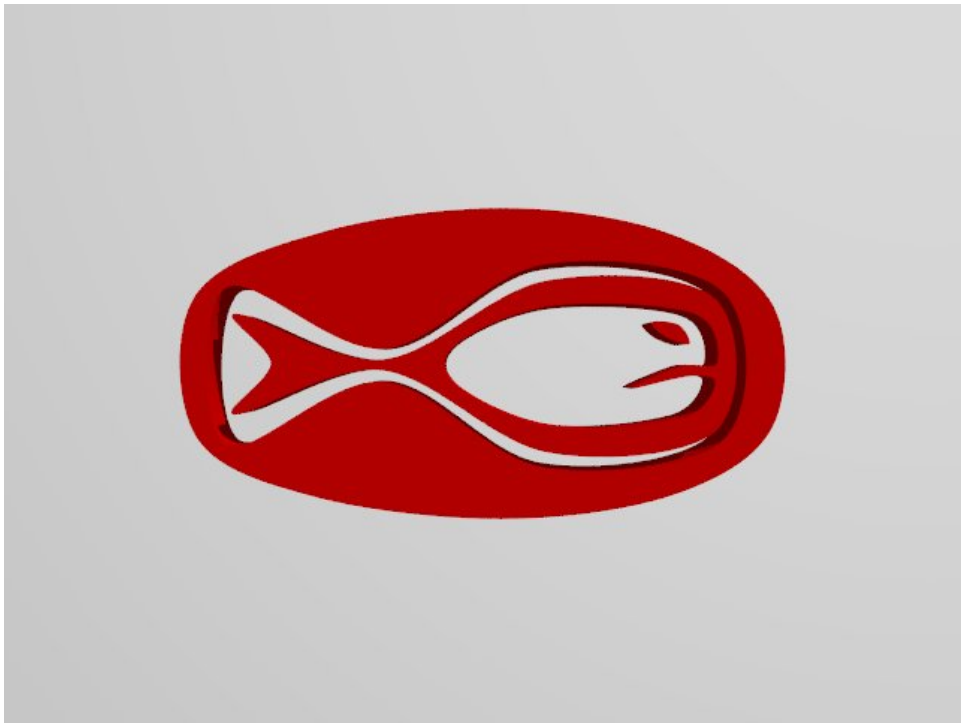

Figure 1.16: Preview image of a discontinuous object.

Creating a sub-curve in a spline object is similar to the initial creation of a new spline object. If necessary, review "Creating New Objects." A new sub-curve cannot be started while the object is already in edit mode; if it is, then leave edit mode by depressing the shift key and clicking with the primary button.

Select an existing object with a primary button click, and with the shift key depressed, another primary click will enter edit mode; unless the mouse pointer active point is very near an existing sub-curve, in which case a new point or segment is added to that sub-curve (use "Undo" if this happens). In edit mode now, points are added with clicks just as in the initial creation of an object, but they will compose a new sub-curve. Add points just as described in "Creating New Objects," and make sure the sub-curve is properly closed in the manner required for the type of spline. Leave edit mode with shift-key+primary-click.

A sub-curve may be placed within another sub-curve, in which case it will be

rendered by *POV-Ray* as a hole (and a sub-curve placed within a "hole" will be rendered as a 'fill'). A sub-curve may be placed so that it is not within any other, in which case it will be rendered as if it were a distinct object (although it will still be part of the whole). Also, a sub-curve may be placed such that its path crosses paths of other sub-curves, and *POV-Ray* will render these with 'holes' or 'fills' at the intersections. Of course, all the above may be combined.

1.3.5 Guide Lines

Epspline provides for simple *guide* lines in the drawing area, which display as thin (one pixel wide) horizontal or vertical red lines (the color may be set: see Global Preferences). These are saved with epspline's files, so they are persistent, but they are not exported as SDL and have no effect on the rendering of objects — they are only tools for editing in epspline.

To add a guide line, place the mouse pointer over one of the graduated scales at the left and top of the drawing area (see "The Drawing Area and Canvas"), hold the shift key down, and 'drag' with the mouse into the canvas (the work area with the blue grid). Do not release the mouse button until the new guide is at the desired position (the shift key may be released as soon as this action is initiated). A horizontal guide is produced from the top scale, and a vertical guide from the left scale.

If necessary, use the scroll bars to adjust the view of the canvas before adding a guide. Existing guides may be moved by dragging with the mouse. To pick up a guide, the mouse pointer active point must be placed directly over the line; since this can be difficult, it might help to place guides at incremental positions (e.g., a five pixel offset) so that the mouse point coordinates displayed on the status bar (see "The Status Bar") can assist in positioning the pointer at such intervals (irregular positions will be difficult to remember).

To remove a guide line, drag it back to the graduated scale and release it there.

When moving a selected object or point, by default the guide lines are merely a visual guide, but temporarily depressing the shift key will provide a light 'snapping' to the guide when a point is in close proximity.

1.4 Useful Transforms

Like most objects in a *POV-Ray* scene, the objects exported by epspline need to be *transformed* in various ways to be placed in the scene with a particular size and orientation. The *translate*, *scale*, and *rotate* keywords are used to accomplish this. Also, a collection of transformations may be placed in a *transform* object. This

section will suggest some useful initial transforms of prisms and lathes exported by epspline.

It was mentioned in Creating and Editing that the coordinates of the epspline drawing area, or "canvas," are not similar to the *POV-Ray coordinate system*. In fact, the SDL allows the *POV-Ray* coordinate system to be modified, and so assumptions made by a program such as epspline might not be valid. It is better that the user understand *POV-Ray* transforms and place the objects logically. Epspline declares a number of necessary values describing the (un-scaled) extremes and positions of objects, and using these declarations, the typical transforms are routine.

1.4.1 Example of Exported SDL

An example of the SDL exported by epspline is shown in listing 1.1. The name of the object is "Foo", and the file it was exported to is named "foo.inc" This "include file" begins (after an initial comment) with a guard against re-inclusion by testing and declaring "`INCLUDED_foo_inc`." The guard name will integrate the file name, so it will probably be unique. If the file is included a second time the `#ifndef` test will fail and everything will be ignored up to the matching `#end` near the end of the file. This common guard allows one include file to be used by several other files without conflict.

Following the guard, constants are declared for the dimensions and position of the the object. These values are collected by epspline as the curve is calculated (i.e., interpolated) for display, and are limited in accuracy by the resolution of the interpolation, but they should be precise enough for any scene.

Appearing next, just before the spline object declaration, is the declaration of an optional transform, named "`Foo_NORMAL_TRANSFORM`." This is only applied if "`Foo_USE_NORMAL_TRANSFORM`" is declared before the file is included. If used, the transform scales the object (not including the *sweep* of a prism) to one unit on the axis with the greater extent. It also centers the object at the origin, and inverts the object along the z-axis for the prism, or the y-axis for the lathe (because of the difference in view between *POV-Ray* and epspline).

Next, the object is declared. If there was more than one object in listing 1.1, then the sequence of constants-transform–object would be repeated for each.

Finally, constants are declared that are similar to those declared for objects but which pertain to all the objects in the file. As listing 1.1 shows only one object, these happen to be the same as the constants for "`Foo`," but if a file is composed of several objects meant to keep their relative positions and sizes, then these final constants are useful.

```
// POV include file
// Generated by epspline


#ifndef ( INCLUDED_foo_inc )
#declare INCLUDED_foo_inc = 1;

#declare Foo_width = 300.000000;
#declare Foo_height = 99.379610;
#declare Foo_left = 100.000000;
#declare Foo_right = 400.000000;
#declare Foo_top = 100.000000;
#declare Foo_bottom = 199.379610;
#declare Foo_max_extent = max(Foo_width, Foo_height);

#declare Foo_NORMAL_TRANSFORM = transform {
  translate <-Foo_left - Foo_width / 2, 0, -Foo_top - Foo_height / 2>
  scale <1 / Foo_max_extent, 1, 1 / Foo_max_extent>
  scale <1, 1, -1>
}

#declare Foo =
prism { bezier_spline linear_sweep -1.000000, 1.000000
  , 8
  , <100.000000, 100.000000>
  , <186.000000, 162.644928>
  , <315.000000, 160.717391>
  , <400.000000, 100.000000>
  , <400.000000, 100.000000>
  , <316.000000, 232.036232>
  , <186.000000, 233.000000>
  , <100.000000, 100.000000>
#ifdef ( Foo_USE_NORMAL_TRANSFORM )
  transform { Foo_NORMAL_TRANSFORM }
#end
}

#declare I_foo_inc_width = 300.000000;
#declare I_foo_inc_height = 99.379610;
#declare I_foo_inc_left = 100.000000;
#declare I_foo_inc_right = 400.000000;
#declare I_foo_inc_top = 100.000000;
#declare I_foo_inc_bottom = 199.379610;

#end // #ifdef ( INCLUDED_foo_inc )
```

SDL-Listing 1.1: One spline object named “Foo” exported as *scene description language*.

1.4.2 Example with a Lathe and Prism

Consider a scene with two objects, a lathe and a prism, that should be rendered together. The lathe is named "Cup," and the prism is named "Handle." Drawn quickly in epspline, they might look like figure 1.17. Hopefully, it is obvious which objects are assigned the names. Since lathes are rotated around the *y*-axis, "Cup" is the profile of half of a cup. Also, the left-most points of "Cup" are at the horizontal point zero. If those points had a positive offset, then the cup would have a hole in the bottom. If they had a negative offset, *POV-Ray* would reject the object and exit with an error message: lathes may not have negative values on the rotated axis.

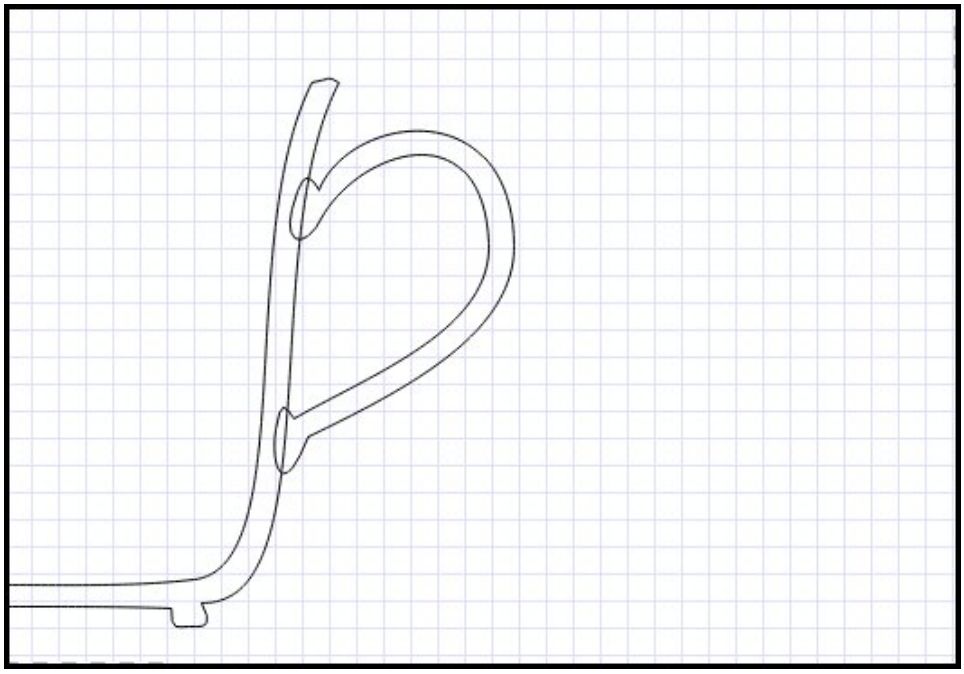

Figure 1.17: A simple cup on the epspline canvas.

After these objects are exported to an include file, "cup.inc," a main scene file is necessary to render the cup with *POV-Ray*. A minimal scene file, "cup.pov," is shown in listing 1.2. The *POV-Ray* documentation will explain the language (SDL).

Near the middle of listing 1.2 the file "cup.inc" is included. Next, the the two objects "Handle" and "Cup" are bound together in a *union* named "CoffeeCupRaw," which has been called 'Raw' because it is not yet an object ready to be presented. The first important transform is stated in the union: the handle is rotated by -90 de-

```
#version 3.5;
#include "colors.inc"

light_source { <20, 20, -20> color White }

camera {
    location <0, 0.75, -4>
    look_at <0, 0.25, 0>
    right x * image_width / image_height
    angle 45
}

#declare CupTexture = texture { pigment { color LightBlue } }
#declare BoxTexture = texture { pigment { color DarkGreen } }

#include "cup.inc"

#declare CoffeeCupRaw = union {
    object { Handle
        rotate x * -90
    }
    object { Cup }
}

#declare CoffeeCup = object {
    CoffeeCupRaw
    translate <0, -Cup_top - Cup_height / 2, 0>
    scale 1 / Cup_height
    scale <1, -1, 1>
    texture { CupTexture }
}

object { CoffeeCup translate <0, 0.5, 0> }
// somewhere to put the cup down:
box { <-1, -0.25, -0.75> <1, 0, 0.75>
    texture { BoxTexture }
}
```

SDL-Listing 1.2: *POV-Ray* SDL for a simple coffee cup scene.

grees. Recall from Creating and Editing that in the 2D view of outlines in epspline the prism is seen parallel to the y-axis, and the lathe is seen perpendicular to the y-axis. Therefore, rotating the handle brings it to the same line-of–sight as the cup.

The union CoffeeCupRaw has put the the handle in juxtaposition to the cup, but the whole is still at a size and offset according to the units used in the epspline interface. In the declaration of "CoffeeCup" the object is is centered (centering is the author's preference) and scaled to one unit on the axis which makes the most sense for this scene, the y-axis. In Example of Exported SDL the constants declared for an object were introduced. Some of these constants are used in transforming the coffee cup. The *translate* vector has zero at x and z, but at y the 'top' offset is removed and half the raw height is subtracted. This y translation is negative because the view of lathes in epspline has 0 at top and increases downward. The next statement, a *scale* transform, reduces the height to one unit, and next another *scale* inverts the cup along the y-axis for the same reason given for negative translation.

Finally, the cup is placed in the scene by adding it in an "object" statement, without "`#declare`," and shifting it again along y so that the bottom of the cup is at 0. A box is placed so that the cup rests on it. The result is shown in figure 1.18.

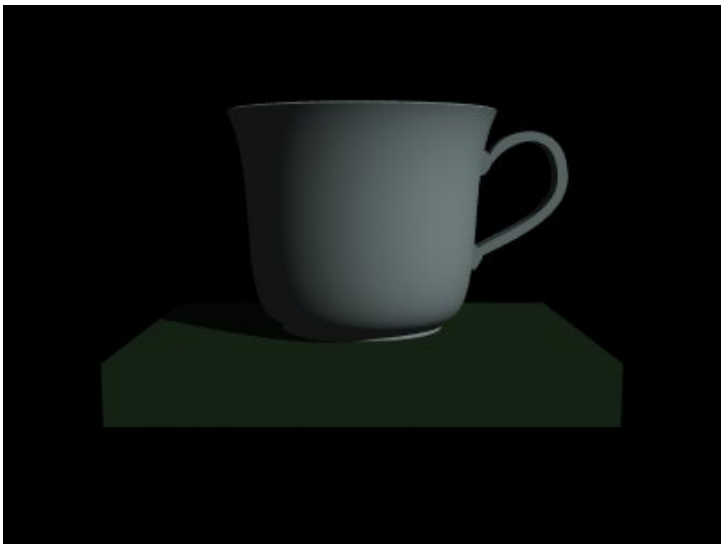

Figure 1.18: The simple cup rendered.

1.5 Preferences

Epspline provides a dialog window with which a set of *preferences* can be set. This dialog window is invoked by selecting the edit menu (see The Edit Menu), "Preferences" item. The dialog window uses a *tab* type interface, and each tab contains related settings.

Five buttons near the bottom of the dialog window affect the preferences and the window. These are explained in the table The Buttons of the Preferences Dialog Window. Note that the buttons listed in the table as "Help," "Apply," "Cancel," and "OK," might have different labels and appear in a different order depending on the current platform or theme (or translation).

BUTTON:	ACTION:
Restore Defaults	set epspline defaults in all items
Restore Configured	set values from stored configuration in all items
Help	show the help viewer window displaying this section
Apply	put current values in effect, and leave the dialog window showing
Cancel	dismiss the dialog window, and discard changed values
OK	put current values in effect, dismiss the dialog window, and save the values in stored configuration

Table 1.5: The Buttons of the Preferences Dialog Window

The dialog window is designed to allow interaction with the main window (see The Main Window Interface) while the dialog window remains present. (The system might force the dialog to remain above the main window, but it can be moved aside.) Changes to settings can be tried and changed again without dismissing the dialog using the "Apply" button. Finally, effect or discard changes with "OK" or "Cancel."

Each tab of the dialog is discussed in the following sections. (The preferences dialog was added in epspline 0.0.4.3, and is sparse. More items may be added in future releases.)

1.5.1 Global Preferences

This tab contains controls for settings that apply to epspline's appearance or behavior. Miscellaneous items appear first followed by interface color controls. The miscellaneous items are:

Default extension/suffix for exports: The suffix offered in the file name dialog displayed when the "Export As" menu item is selected to choose a name for the *POV-Ray* SDL form of your work. If this item is set, the leading *dot* ('.') should be included as it will not be added automatically. This allows a different separator character, or none at all, to be used.

Draw grid lines on canvas background: By default the drawing area will have a grid of lines meant to help with alignment, and to resemble graph paper. If a blank canvas is preferred, the grid can be disabled with this item.

The color selection controls combine a text field, and a 'picker' widget that is invoked with the button to the right of the text field. The picker will be conventional for the current platform, and will not be described here. The text field will accept a number of common color names in English (it's uncertain whether these names are subject to translation), or as red/green/blue values specified in either CSS-like or HTML-like form. For example, `rgb(255, 31, 15)` or `#FF1F0F` respectively. The color control items are:

Choose background grid color: If the background grid is enabled, it will be drawn with the color selected here.

Choose guides color: Horizontal or vertical guide lines can be drawn onto the canvas from the scales to the left and top of the drawing area (see Guide Lines). The color of these guide lines is selected here.

Choose background color: The underlying color of the drawing area is selected with this item.

1.5.2 *POV-Ray* Settings

Naturally, *POV-Ray* is used to display a preview of the current work (see The Help Menu). This tab provides settings that apply to the invocation of *POV-Ray*. The *POV-Ray* settings are:

***POV-Ray* executable file:** An alternative may be given here. If this text field contains a file system path rather than a name alone, epspline will attempt to use

the path directly. If the field contains a name only, the behavior differs for Unix-like systems and MS Windows systems. Under Unix, the executable will be searched for in `$PATH` from epspline's environment, as would be expected. Under MS Windows, the given name will be substituted in the result of a query for the '.POV' file type application. In fact, under MS Windows, the default value "pvengine.exe" is actually just a placeholder. The '.POV' type should be associated with the executable set by the *POV-Ray* installer program. If the *POV-Ray* directory contains alternative names, one can be given here. If *POV-Ray* was not formally installed (for example, a local build from source) a full path may be given here. Epspline does not attempt to search a PATH from the environment under MS Windows.

***POV-Ray* options (switches):** *POV-Ray* command options may be set in this text field. The default is '`+D +P +Q9 +A`' and at least `+D` and `+P` should be left in place. Note that if you are using a *POV-Ray* version less than 3.7 under Unix, then you might want to add '`-visual DirectColor`' to avoid unwanted transparency in the preview window, but *do not* give that option for *POV-Ray* for Unix version 3.7 or greater, or under MS Windows. This field should be most useful for `+W` and `+H` (width and height).

It is expected that more settings will be added to the preference dialog if further development of epspline is warranted.

1.6 Mouse and Key Reference

This section briefly lists mouse and key inputs and their actions. General concepts are not discussed in this section, nor is usage discussed in detail. If more explanation is needed, go to the beginning of the chapter "Using Epspline." If even *POV-Ray* is not known, go to "Introduction."

1.6.1 Mouse and Key Input Actions

Mouse Primary Button Click

- Outside the bounds of a selected object, the object is deselected
- Within the bounds of an object, when no object is selected, the object is selected
- Within the bounds of a selected object, if on a control point, selects the point

- Within the bounds of a selected object, if *not* on a control point, deselects any selected point

Shift Key and Mouse Primary Button Click

- When no object is selected, enters edit mode and begins adding points in a new object
- When an object is in edit mode, leaves edit mode
- If an object is selected, and the mouse active point lies on (or very near) a sub-curve, a new point or segment is added to the sub-curve
- If an object is selected, and the mouse active point is *not* near a sub-curve, enters edit mode and begins adding points in a new sub-curve

Mouse Secondary Button Click

- When a selected object is in transform mode, leaves transform mode
- When an object is selected but not in transform mode, the object pop-up menu is shown
- When no object is selected, the canvas (drawing area) pop-up menu is shown

Shift Key and Mouse Secondary Button Click

- When an object is selected but not in transform mode, enters transform mode
- When a selected object is in transform mode, cycles through scale, shear, and rotate modes

Mouse Primary Button Drag

- If mouse active point is on a guide line, guide line is moved, or removed if dragged over corresponding graduated scale
- On a selected control point, the point is moved; with the shift key pressed, may ‘snap’ point to a nearby guide line; in the bézier spline a group of associated points are effected unless control (command) key is depressed

- On a selected object *not* in transform mode, the object is moved; with the shift key pressed, may 'snap' an edge (of the bounding box) to a nearby guide line
- On a selected object *in* transform mode, if the mouse active point is within a 'handle' or lies on the bounding box, the transform is effected

Backspace and Delete Keys

- A selected point is deleted; in the bézier spline the group of points associated with the selected point are deleted and remaining end-points are joined
- A selected object without a selected point is deleted

Page Down and *D*, and Page Up and *U* Keys

- A selected point is moved by several pixels, horizontally if the *alt* key is down, vertically otherwise
- A selected object without a selected point is moved by several pixels, horizontally if the *alt* key is down, vertically otherwise
- With no selection the view of the canvas is scrolled by a large amount (page keys only, not *D* or *U*), horizontally if the *alt* key is down, vertically otherwise

Arrow Keys and *H*, *J*, *K*,*L* Keys

- A selected point is moved by one pixel, horizontally if the *alt* key is down, vertically otherwise
- A selected object without a selected point is moved by one pixel, horizontally if the *alt* key is down, vertically otherwise
- With no selection the view of the canvas is scrolled by a small amount (arrow keys only, not *H*, *J*, *K*, or *L*)

Chapter 2

Licenses

Epspline is licensed under the terms of the *GNU General Public License*, optionally Version 2 or a later version. The epspline documentation (this document) is released under the terms of the *GNU Free Documentation License* version 1.3.

Where there are instances in which a specific use of the wxWidgets library implies that the wxWindows/wxWidgets License should apply, that license does apply. As of this writing such instances exist in the files `wxexio.h`, `wxexio.cpp`, `lexer.l`, and `parser.y`, all under the directory `src/`.

The epspline source includes a binary installer program generated by the *Nullsoft Scriptable Install System* (*NSIS*). The files `msw_pkg/epspline.nsi.in` (from which `msw_pkg/epspline.nsi` is generated), and `msw_pkg/file_association.nsh`, are derived from *NSIS* example sources and retain the original license.

The source file `src/splines.cpp` uses, with permission granted by the POV-Team Co-ordinator on 7 April 2000, a total of 24 lines of code found in the file `prism.c` of version 3.1 of *POV-Ray*. The epspline source includes, in the top directory, the file `POV_permission.txt` containing the communication in which permission was requested and granted. It must be understood that an exception to the (then current) license was granted, and that the use of the 24 lines of code in epspline does not imply that *POV-Ray* code use is permitted. Please abide by the *POV-Ray* licenses at `http://www.povray.org/povlegal-3.6.html` and `http://www.povray.org/distribution-license-3.6.html`.

Update: on November 08, 2013 *POV-Ray* version 3.7 was announced at the *POV-Ray* web site, and it is released under the *GNU AFFERO GENERAL PUBLIC LICENSE*. (The license *URL*s in the preceding paragraph have been updated and now have a 'legacy' status.) See `http://www.povray.org/povlegal.html` for the current license. Many thanks to the *POV-Ray* copyright holders for this welcome change.

The following sections present the *GNU* licenses.

2.1 GNU General Public License

Version 3, 29 June 2007

PREAMBLE

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program–to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

   "This License" refers to version 3 of the GNU General Public License.

   "Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

   "The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

   To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

   A "covered work" means either the unmodified Program or a work based on the Program.

   To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

   The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

   A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

   The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

   The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed

to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

   All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

   You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

   Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

   No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

   When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

   You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

   You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

   You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

   (a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

   (b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

   (c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

   (d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

   A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

   You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

   (a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

   (b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

   (c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

   (d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

   (e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and

with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

   "Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

   When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

   Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

   (a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

   (b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

   (c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

   (d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

   (e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

   (f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

   You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

   However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

   Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

   Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

   You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as

a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

    Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

    An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

    You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

    A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

    A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right

to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or

copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

    If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

    Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

    The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

    Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

    THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

    IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

    If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

HOW TO APPLY THESE TERMS TO YOUR NEW PROGRAMS

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <textyear>  <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program.  If not, see <http://www.gnu.org/licenses/>.
```

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program>  Copyright (C) <year>  <name of author>

This program comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'show c' for details.
```

The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see http://www.gnu.org/licenses/.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read http://www.gnu.org/philosophy/why-not-lgpl.html.

2.2 GNU General Public License Version 2

Version 2, June 1991

51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Preamble GPL 2

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

Terms and Conditions For Copying, Distribution and Modification

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

   Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

   You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

(a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

(b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

(c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

(a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

(b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

(c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance

of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

   If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

   It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

   This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is

permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

   Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

    No Warranty

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT

NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

End of Terms and Conditions

Appendix: How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

> one line to give the program's name and a brief idea of what it does.
> Copyright (C) yyyy name of author
>
> This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.
>
> This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.
>
> You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

> Gnomovision version 69, Copyright (C) yyyy name of author
> Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.

> This is free software, and you are welcome to redistribute it under certain conditions; type ‘show c’ for details.

The hypothetical commands `show w` and `show c` should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w` and `show c`; they could even be mouse-clicks or menu items—whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a “copyright disclaimer” for the program, if necessary. Here is a sample; alter the names:

> Yoyodyne, Inc., hereby disclaims all copyright interest in the program ‘Gnomovision’ (which makes passes at compilers) written by James Hacker.
>
> signature of Ty Coon, 1 April 1989
> Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

2.3 GNU Free Documentation License

Version 1.3, 3 November 2008

<http://fsf.org/>

Preamble

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "**Document**", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "**you**". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "**Modified Version**" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "**Secondary Section**" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "**Invariant Sections**" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "**Cover Texts**" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "**Transparent**" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "**Opaque**".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "**Title Page**" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page

as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "**publisher**" means any person or entity that distributes copies of the Document to the public.

A section "**Entitled XYZ**" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "**Acknowledgements**", "**Dedications**", "**Endorsements**", or "**History**".) To "**Preserve the Title**" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they

preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

C. State on the Title page the name of the publisher of the Modified Version, as the publisher.

D. Preserve all the copyright notices of the Document.

E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.

G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.

H. Include an unaltered copy of this License.

I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.

K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.

M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.

N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.

O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and

all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See http://www.gnu.org/copyleft/.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not

as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

> Copyright © YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections,

> no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with ... Texts." line with this:

> with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

Index