Your Short Cut to Knowledge

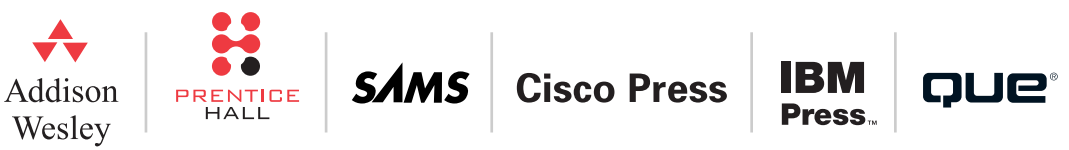
The following is an excerpt from a Short Cut published by one of the Pearson Education imprints.

Short Cuts are short, concise, PDF documents designed specifically for busy technical professionals like you.

We've provided this excerpt to help you review the product before you purchase. Please note, the hyperlinks contained within this excerpt have been deactivated.

Tap into learning—NOW!

Visit **www.informit.com/shortcuts** for a complete list of Short Cuts.



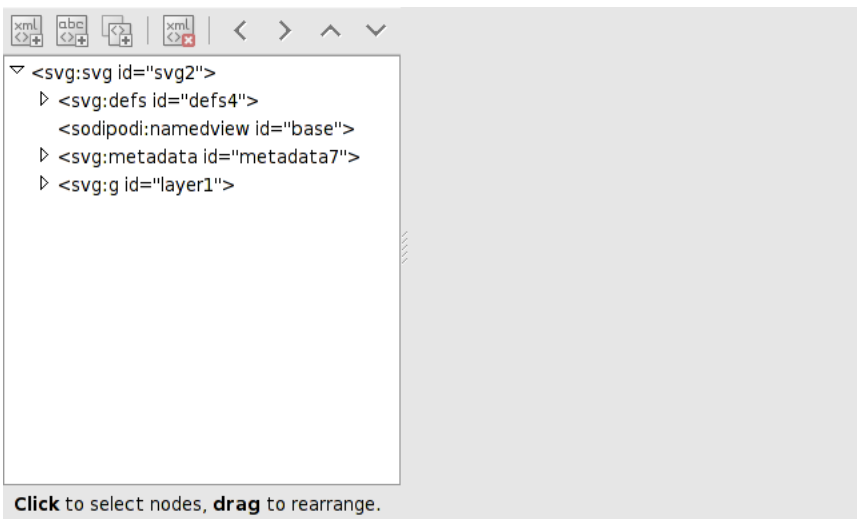
Chapter 14. XML Editor

The *XML Editor* dialog allows one to directly edit the *XML* description of an *SVG* drawing. (Recall that Inkscape is an *SVG*-based drawing program and that *SVG* is an *XML*-based file format.)

The ability to directly edit an *SVG XML* file is very powerful. It allows the user more control over objects in their drawing such as specifying the exact size or position of an object and by giving access to *SVG* parameters that are not directly or easily available through the Inkscape interface.

Basic Usage

To edit the *XML* file, open the *XML Editor* dialog (Edit → XML Editor... (**Shift+Ctrl+X**)). This will open a window like the following for an empty drawing.

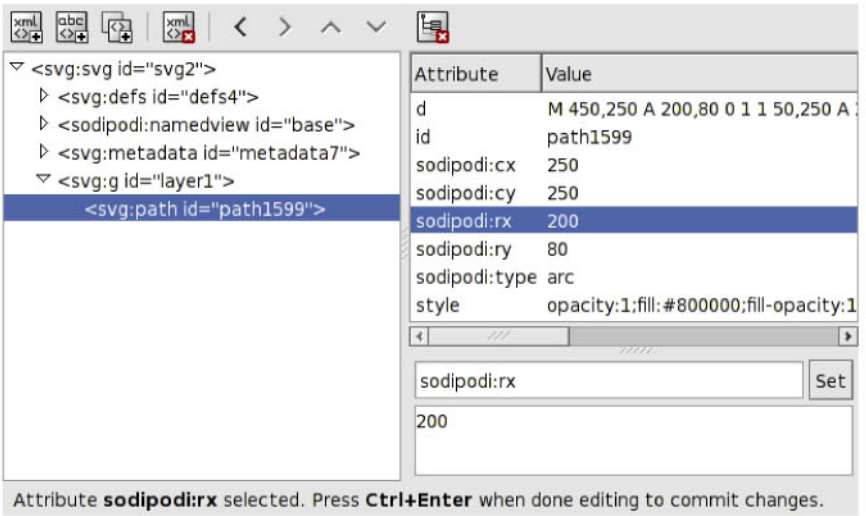


XML Editor dialog (with nothing selected).

The "tree" structure of the empty drawing is shown on the right in the dialog. Even an empty Inkscape drawing contains information including an empty *Layer* ("layer1"). The *Layer*, like all the items listed, is represented by a "node" in the tree. If the layer contained objects, they would be represented by nodes under the layer's node. The objects under a particular node can be hidden in the tree view. To hide and unhide these objects click on the small triangles just in front of a node's name.

Upon adding an ellipse to the drawing, an entry (node) is added for the new ellipse under the formerly empty *Layer* (see next figure). The line is highlighted and the ellipse's parameters are shown on the left. Note that the name of the object is given in the highlighted line (in this case, "path1599").

Now, suppose you would like the ellipse to have a width of 400 pixels (i.e., a radius of 200 pixels in the x direction). You can specify this by clicking on the "sodipodi:rx" attribute. The attribute is shown below with the current value in the attribute entry box. Change that value to 200 and then click on the *Set* button (or use **Ctrl**+**Enter**).



XML Editor dialog after changing the width of the ellipse.

The ellipse should now be 400 pixels wide. You could also change the height (sodipodi:ry) and the location of the center of the ellipse (sodipodi:cx and sodipodi:cy). And you can change attributes like the *Fill* color and *Stroke style*.

Elements in the Inkscape file fall under two categories: those included in the *SVG* standard and those that are private to Inkscape. In the above example, the attributes with the “sodipodi” tag are internal to Inkscape (the “sodipodi” tag is the result of Inkscape being branched from the Sodipodi program). These attributes are used to calculate the “real” *SVG* path definition given by the 'd' attribute. (See the section called *Paths* in Appendix B, *File Format*.)

The Inkscape internal elements should be ignored by other *SVG* rendering programs. This may not always be true thus Inkscape includes the possibility to export a drawing without the Inkscape internal elements.

Sometimes it is useful to know what the allowed attributes are for a given type of object. The *SVG* standard is described in detail at the *Official W3C SVG* [http://www.w3.org/TR/SVG/] website. Note that not all the *SVG* standard is currently supported in Inkscape. It is possible, however, to add nonsupported attributes via the *XML Editor*. These attributes may not be displayed by Inkscape but will appear in any program that supports those attributes.

Editing XML Nodes

The *XML Editor* dialog includes a number of clickable icons to manipulate the “nodes” in the *XML* tree. A description of each icon follows. Some of the things you can do with these commands may not be so sensible.

- Add XML element node. Add a node. For this to be useful, all the attributes appropriate to the type of node add must also be added.
- Add XML text node. Can also be used to edit the text in a text object.
- Duplicate XML node. Make a copy of the currently selected node including all its daughters. The new node will be placed at the same level and just after the original node. As each node must have a unique *ID*, Inkscape will assign a new *ID*. You can change the *ID* name if you wish.
- Delete XML element node. Delete a node and all its daughters.
- Unindent node. Move a node out one level. For a drawable object in a group, this is equivalent to removing that object from that group.
- Indent node. Move a node in one level. The node will be moved under the closest node above at the same level.

- Raise node. Move a node above the previous node with the same parent. Equivalent to changing the *z-order* when the two nodes are drawable objects.
- Lower node. Move a node below the next node with the same parent. Equivalent to changing the *z-order* when the two nodes are drawable objects.
- Delete the selected object attribute.

Examples

A few examples are given here to show the possibilities of “hand” editing the *XML* file.

Adding Color to a Marker Arrow

Markers on paths in Inkscape do not inherit the attributes of the path. This is most noticeable for colored paths where the markers are drawn in black. As of v0.45, Inkscape includes the *Color Markers to Match Stroke* effect for changing the color of markers to match the stroke color. This section is kept for pedagogical reasons.

To add color to a marker, open up the *XML Editor*. Select the path with the marker in the canvas window. In the “style” attribute for the line, locate the marker entry (marker-end:url(#Arrow2Lend) for example). Then expand the “<svg:defs>” line by clicking on the triangle at the beginning of the line (if not already expanded). You should see an entry for the marker. Select that entry. The attributes for the marker should be displayed on the right. Select the “style” attribute. Add “fill:#rrggbb” to the attributes in the entry box at the bottom right, where #rrggbb is the *RGB* color in *hexadecimal* form (obtainable from the attributes for the path).

The marker should change color. If is doesn't, then expand the "<svg:marker>" line. Select the path entry and remove any *Fill* and/or *Stroke paint*. For this change to show up, you must save and reopen the *SVG* file.

If you wish to have markers of the same type with different colors, then you must add copies in the <svg:defs> section. You can use the *Duplicate Node* icon to duplicate a marker entry. Rename the new entry to a suitable name and change the reference to the marker in the path object you want the marker to be associated with. Again the file must be closed and reopened for the changes to be seen.

You are not limited to changing color. You can change other attributes such as the marker size.

A red line with a red marker.

Underlined Text

Underlined text cannot be added through the normal Inkscape interface, nor will Inkscape display underlines. But you can add underlined text that will be displayed properly by another *SVG* program.

To underline text, open the *XML Editor*. Select the text you wish underlined. Go to the "<svg:tspan>" object found inside an "<svg:text>" tag. If you are selecting part of the text, you may need to add some attribute temporarily (color for example) to create a corresponding “tspan” object; the color can be removed later. Add to the style: “text-decoration: underline”.

Here is an example as rendered by the Squiggle program.

A test of underlined text.

Underlined text example.