

LINUX VOICE

TUTORIAL

MAKE YOUR OWN FONTS WITH BIRDFONT

MIKE SAUNDERS

You don't need to be a design whizz to create your own custom fonts – BirdFont makes it easy as a particularly good-looking pie.

WHY DO THIS?

- Create funky typefaces from scratch or based on existing designs
- Give your printed documents or website a unique feel
- Export to TTF, EOT and SVG formats

If you were using Linux in the late 90s (or you've seen screenshots of the desktop environments back then), you'll know that it was pretty ugly. Fonts, in particular, were a bit of a disaster area. Today we have gorgeous desktops and window managers, and distros ship with oodles of top-quality, free-as-in-freedom fonts. But have you ever considered making your own font? You can create one from scratch if you're full of ideas, or base one on an existing design – eg an old document or a logo. It's much simpler than it sounds, so we'll explain how.

To make our custom font we'll be using BirdFont (www.birdfont.org), an excellent font editor that runs on Linux, Mac OS X and Windows. Packages are available for many distros, but if you can't find it in your distro's repositories, grab **birdfont-0.37.tar.gz**



The street sign we'll be using to create the lowercase "a" character in our custom font.

from the project's website, extract it, and follow the instructions in the README. Once you have it installed, just enter **birdfont** in a terminal to start it.

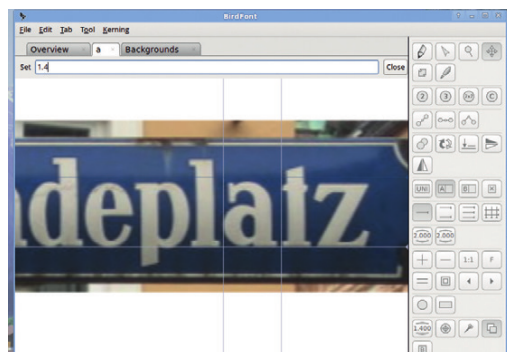
In this tutorial we'll use an existing design as the basis for a font. We'll take a street sign and create a glyph (font character) of the letter "a" from it. Of course, if you want to make a complete font then you'll need an image that contains all letters (uppercase and lowercase) along with numbers.

Step by step: create a font

1 Align image

Start BirdFont and click on File > New to create a new font. A list of glyphs will appear – scroll down and double-click on "a". In the right-hand toolbox, click the **B** button (it's shows an uppercase B) towards the bottom to insert a new background image (all of the buttons have tooltips, so hover over them with the mouse to find out what they do).

Click on the + button to add an image, and then double-click its thumbnail. Move the image using the target (🎯) tool until the image's "a" character is inside or over the box. Right-click the **100%** button to open a scale value bar, and scale the "a" until its height matches the box. Finally, grab the right-hand guide line using its small arrow at the bottom to match the "a" character's width.



We've moved and resized the image so that the "a" is inside the box, and pulled the right-hand guide in.




2 Create the outline

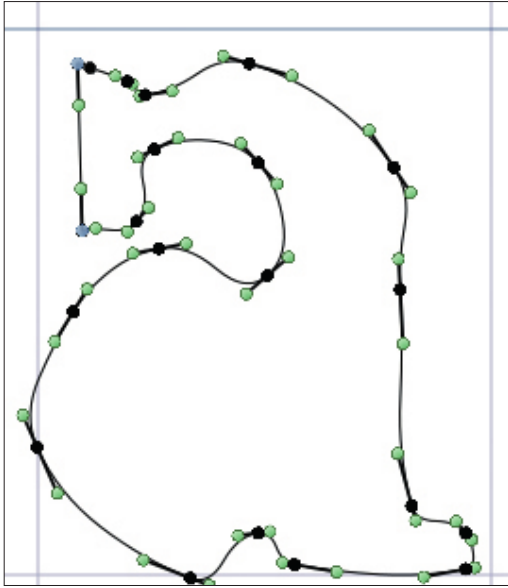
Now click the **⚡** icon (add new points) in the top-left of the toolbox, and click several times around the outside edge of the character to create an outline, eventually clicking on the first point to complete it. This outline can be pretty rough – you don't need to add points for every tiny detail. Use Shift+Ctrl+= (equals key) to zoom in.



Figure 2: The outline for our glyph. It's looking rather angular at this stage, but we'll fix that in a moment...

3 Smoothen the edges


Back in the bottom-right of the toolbox, click the  icon (show/hide background image). Then click the  button at the top. Now hold down Shift, and click on all of the blue points on the outline, going round the whole glyph clockwise. When they're all selected, click the  (tie curve handles) button in the tool pane and the edges will be rounded out.




That looks a lot better! With the edges rounded it's starting to look like a proper character.

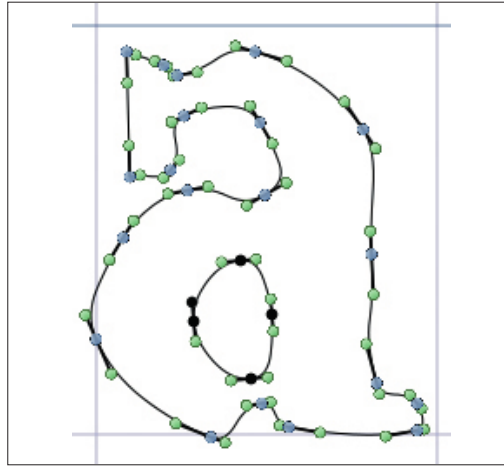
5 Preview it

When you're happy with everything, go to File > Preview (you'll be asked to enter a name for the font). Then a Preview tab will appear, showing your glyph being used in some example sentences.

If you're happy with the results, congratulations – you can now go on to do all the other letters! (It might be a long job.) If you need to fine-tune the character more, click its tab again, choose the arrow (Move Points) button in the tool pane, and fine-tune it. And if you need any help, pop by our wonderful forums at <http://forums.linuxvoice.com>. 

4 Align the paths

Click the  (show/hide background) button again. Chances are that the current paths won't be 100% perfectly in sync with the original image, so click and drag the blue points to line them up (they're Bézier curves, so you can also alter them with the green points). If you have an area that needs removing, like the hole in the bottom of the "a" character, for instance, draw a new path as per the previous instructions, and when it's complete click on Create Counter From Outline). Then smooth out the points as in step 3, to get the result shown below.



Here we've added the inside part of the character as a counter path. We won't be giving up the day job.



Our new "a" character in BirdFont's Preview. Sure, it looks rather out of place, but when we've done the others...

Exporting your design

When you've finished designing your font, click File > Export in the menu and provide a name for it. BirdFont will save your work as **<name>.bf** in your home directory (eg **/home/mike/myfont.bf**). It will also create various font files that you can install into your Linux distribution (or indeed other operating systems): **Typeface.ttf** (TrueType, the most common format), **Typeface.eot** and **Typeface.svg**.

It's also possible to include your font in your website, giving it a more personal feel than those sites that use regular Helvetica or Times fonts. During the Export process, BirdFont also generates a **Typeface.html** file. Have a look inside it, especially the **@font-face** parts of the CSS towards the top, to see how to use custom fonts in a page.

```
<style type="text/css">
  body {
    text-rendering: optimizeLegibility;
    font-feature-settings: "kern";
    -moz-font-feature-settings: "kern=1";
    -ms-font-feature-settings: "kern";
    -webkit-font-feature-settings: "kern";
    -o-font-feature-settings: "kern";
  }
  @font-face {
    font-family: 'TypefaceSVG';
    src: url('Typeface.svg#Typeface') format('svg');
  }
  @font-face {
    font-family: 'TypefaceTTF';
    src: url('Typeface.ttf') format('truetype');
  }
  @font-face {
    font-family: 'TypefaceEOT';
    src: url('Typeface.eot');
  }
</style>
```