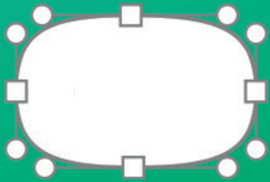
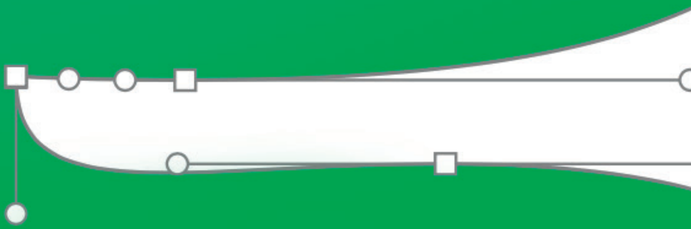
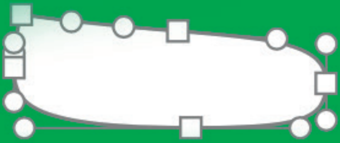
Handbook March 2017





Glyphs Mini 2



Create – Produce – Release
Affordable Font Editing for Everyone

You are reading the Glyphs Mini Handbook from March 2017 for the application version 2.0. Please download the latest version of this handbook at:
glyphsapp.com/get-started

Written by Stella Chupik, Rainer Erich Scheichelbauer and Georg Seifert.

Thanks to Jeff Kellem, Rob Keller, Toshi Omagari and Claus Eggers Sørensen for their invaluable input.

Contents

1 Glyphs

1.1 GLYPHS MINI

Glyphs Mini is a trimmed-down, light version of the application. It is primarily a tool for designing and producing new fonts. Its main principle is that you can edit glyphs in a word context. All tools are optimized for a type design workflow as natural, quick, and intuitive as possible.

We believe that you should be able to focus on your design and only be bothered with technicalities if it is really necessary. There is no need for keeping a design version next to a production version of your font. All production steps take place at export time. By default, Glyphs Mini automates a lot of technical details for you. It lacks many of the advanced features of the full application, e.g., support for plug-ins, glyph layers, Multiple Masters, Python scripting, custom OpenType feature code, custom parameters, and manual hinting.

It is intended as a simple and affordable entry-level solution for casual type design or putting together a dingbat (symbol) font. All available tools and uses are described in this handbook.

Glyphs Mini exports OpenType/CFF (Compact Font Format, suffix .otf) and Webfont (suffix .woff). Glyphs Mini can open existing fonts. However, in the import process, some information stored in the font may be lost. For details, see chapter 11, 'Import and Export' (p. 84).

1.2 INSTALLATION

Glyphs Mini is a Mac-only application. Glyphs Mini 2.0 or higher requires at least macOS 10.9.5 Mavericks. Download the trial from glyphsapp.com/mini and move the app into your Applications folder, which you can access in Finder via *Go* > *Applications* (Cmd-Shift-A). To unlock the trial, open the .glyphsmini2license file you purchased, either by simply double clicking the license file, or by dragging it onto the app icon.

1.3 COMMUNITY

If you have questions, bug reports or suggestions, you can register for free in the Glyphs forum at forum.glyphsapp.com .

1.4 UPDATES

If you downloaded Glyphs Mini from the AppStore, you can keep it up to date with AppStore app.

The application is updated on a regular basis. The latest feature additions are described on glyphsapp.com/blog. You can check which version you have by choosing *Glyphs Mini > About Glyphs Mini* and trigger an update with *Glyphs Mini > Check for Updates* or by activating automatic updates in *Glyphs Mini > Preferences > Updates*. For more details, see section 12.1.1, 'Updates' (p. 87).

1.5 KEYBOARD SHORTCUTS

To enable a workflow as efficient as possible, Glyphs Mini employs a number of keyboard shortcuts. You can set your own shortcuts in the macOS System Preferences, under *Keyboard > Shortcuts > App Shortcuts*. Click on the plus and add *Glyphs Mini*.

Some shortcuts conflict with default shortcuts for system functions like Cmd-Space for Spotlight or Ctrl-Arrows for switching between Spaces. To enable the full functionality of the application, it is advisable to change these system shortcuts in the System Preferences.

1.6 GLYPHS

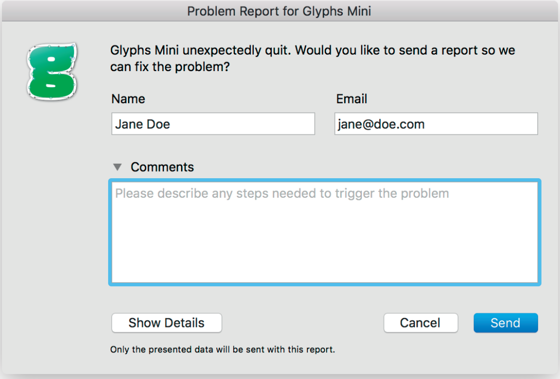
The full version of Glyphs offers many more tools and ways to design and produce your fonts. You can customize OpenType feature code, and a wide range of automatic features. Glyphs also offers a wide range of non-Latin support, including complex scripts like Arabic and Devanagari. You can add Multiple Masters and glyph layers, and extend the app functionality with third-party plug-ins and Python scripts.

The full version exports OpenType fonts of all flavors: OpenType/CFF (Compact Font Format, suffix .otf), OpenType/TT (TrueType, suffix .ttf), and the webfont formats WOFF and WOFF2 (Web Open Font Format, suffixes .woff and .woff2), and EOT (Embeddable OpenType, suffix .eot).

1.7 CRASH REPORTS

After an application crash, you will see two crash report dialogs. First, a generic one that goes to Apple, and secondly, when the app restarts, you will see a report dialog that sports the Glyphs Mini app icon in the top left. The generic crash report goes to Apple, and never reaches us. The second one (with the Glyphs Mini icon) is essential for us to fix the problem that caused the crash. So, please, in your own interest, *always* send those reports. If you do, we will receive some technical information about your hardware, your OS, and the last state of the Glyphs Mini application, and some additional information such as the exact time your crash happened.

When run for the first time, Glyphs Mini will ask for access to your contacts, in order to auto-fill your name and e-mail address in the crash reports. If you do not wish to grant that access, you can later manually enter your info when the crash report dialog appears, or simply leave the fields blank.



Problem Report for Glyphs Mini

Glyphs Mini unexpectedly quit. Would you like to send a report so we can fix the problem?

Name	Email
Jane Doe	jane@doe.com

▼ Comments

Please describe any steps needed to trigger the problem

Only the presented data will be sent with this report.

Providing your contact information, however, allows us to get back to you. In order to fix the bug that caused the crash on your machine, we may ask you for the file you were working on when the crash happened. We delete the files after we have dealt with your problem, and do not hand them over to a third party without your explicit consent. If you can reliably reproduce a crash, it is a good idea to describe the steps for reproducing it, either directly in the crash report dialog, in the forum or by email.

2 Edit View

2.1 EDITING GLYPHS

In Edit view, you can edit glyphs. To access Edit view, you need to open an Edit tab with *View* > *New Tab* (Cmd-T). To access an existing tab, either click on the respective tab title or press the number of the tab (2 through 9) while holding down Cmd-Opt. The first tab (Cmd-Opt-1) is always the Font view. For more details, see chapter 5, 'Font View' (p. 45).

The Edit view has two modes, edit mode and text mode. Access text mode by activating the Text tool (keyboard shortcut T). For more details, see section 2.8, 'Entering Text' (p. 28). Switch into edit mode by either choosing one of the other tools, double clicking one of the displayed glyphs, or pressing the Esc key when the text cursor is situated in front of a glyph.

You can prevent further changes in a glyph by enabling the *Locked* option in the context menu (right or Ctrl-click). A lock symbol will appear in edit mode to indicate its status. Disable the *Locked* option to allow editing again.

2.2 DRAWING PATHS

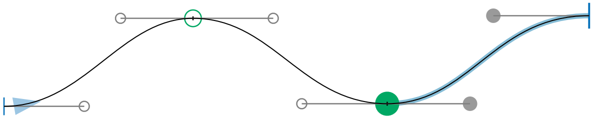
Create new paths with the Draw, Pencil or Primitives tools.

2.2.1 Draw Tool

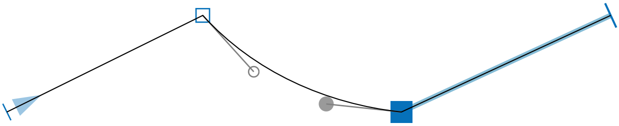
When the Draw tool ✒ (shortcut P as in 'Path' or 'Pen') is active, click anywhere in the canvas to create straight lines, or click and drag to create curve segments. If you keep your mouse button pressed, you can move the on-curve point by holding down the space bar on your keyboard. Close the outline by clicking on its first node.

This handbook uses the term 'node' to refer to on-curves, 'handle' for off-curves, and 'point' as an umbrella word for both types.

Nodes in smooth connections will appear as green circles. You can set a different color in *Glyphs Mini* > *Preferences* > *User Settings*. A smooth connection can either be a curve (an on-curve point in line with two surrounding off-curve points) or a tangent (an on-curve point in line with another on-curve point and an off-curve point). You can trigger the display of nodes with *View* > *Show Nodes*.



In order to draw a corner, hold down the Option key while dragging. Corner points are marked by blue squares. Again, you can choose a different color in *Glyphs Mini* > *Preferences* > *User Settings*. Points in a corner connection are not kept in line, so you can move them independently from each other.



Handles (Bézier control points or BCPs, off-curve points) control the curvature of the path segment and are displayed as small gray circles. In open paths, start and end points are displayed as short perpendicular blue lines. The start point also features a blue triangle, indicating the path direction, i.e., the stored order of the points. The path direction is important for certain path functions and glyph rendering (see 2.3.14, 'Controlling Path Direction', p. 20).

2.2.2 Pencil Tool

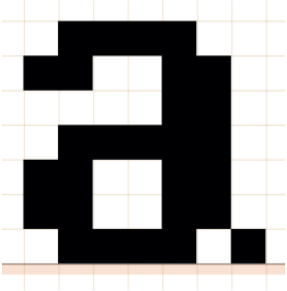
The Pencil tool (shortcut B) offers a quick way to draw freehand curves, especially for people using tablets. The resulting paths will need some cleaning up. That is because the Pencil paths will usually contain too many nodes in order to stay as close as possible to the cursor movement.

2.2.3 Pixel Tool

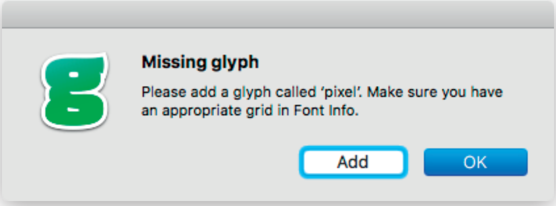
The Pixel tool (shortcut H) is a quick way to add pixel components to your glyph, which can be helpful if you want to create a pixel font. Click and hold the Pencil tool or press the shortcut key to access the tool. For the Pixel tool to work, you need a glyph named 'pixel' (see 5.2.1, 'Generating New Glyphs', p. 46), and an appropriate *Grid Spacing* setting in *File* > *Font Info …* (Cmd-I). The Pixel tool uses the grid step, the first value of the *Grid Spacing* pair, for positioning the pixel components.

Usually, best results are achieved if the grid step value is the same as the size of the pixel. So, e.g., if you have a pixel that is 50 units wide, the grid step should be 50 as well, i.e., set the *Grid Spacing* to 50/1. For details about components, see chapter 7, 'Reusing Shapes' (p. 61).

To draw pixels, simply click and drag on the canvas. To delete pixel components that have already been placed, click on an existing pixel, and start dragging.



Using the Pixel tool without a pixel glyph in your font will trigger a warning dialog that will also give you the option to generate right away with an *Add* button. Pushing the button will add a pixel glyph according to your grid settings, and set it to not export.

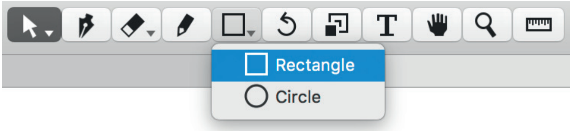


Missing glyph

Please add a glyph called 'pixel'. Make sure you have an appropriate grid in Font Info.

2.2.4 Primitives

Glyphs Mini offers rectangles and ovals as built-in primitive shapes. Click the tool or press F to activate it in its current mode. Click and hold the Primitives tool or press Shift-F to choose between the two shape options. Alternatively, you can use *Draw Circle* or *Draw Rect* from the context menu to switch between the two functions.



Click once on the canvas to create a primitive by entering its measurements with the keyboard. Or click and drag to draw it directly into the edit area. Hold down Shift for a perfect square or circle. Hold down Option to draw from the center of the shape.

2.3 EDITING PATHS

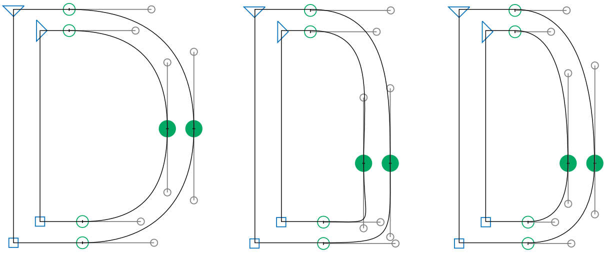
2.3.1 Selecting Nodes and Paths

Click and drag with the Select tool ↖ (shortcut V) to select nodes and handles inside a rectangular selection area. Hold down the Option key to ignore the handles and only select on-curve nodes. Double click on or near an outline segment to select complete paths. Hold down the Shift key to extend or reduce the selection.

Glyphs Mini allows you to select multiple handles independently of the nodes. You can also Shift-select any number of handles in a non-contiguous selection. When a single node or handle is selected, press the Tab key to select the following point on the path, or Shift-Tab to go to the previous one.

2.3.2 Moving Selected Nodes and Paths

Move a selection using the mouse or the cursor keys. Moving nodes will move the attached handles even if they are not selected. Hold down the Shift key for increments of 10, and the Cmd key for increments of 100. Hold down the Option key to move only explicitly selected on-curve points. While moving one or more nodes, hold down both Ctrl and Option (or add Option after you started dragging) to 'nudge' them, i.e., to proportionally adjust the surrounding unselected handles at the same time.



Left: original glyph outline with two selected nodes. Center: selected nodes moved, handles stay the same. Right: selected nodes nudged, handles are adjusted.

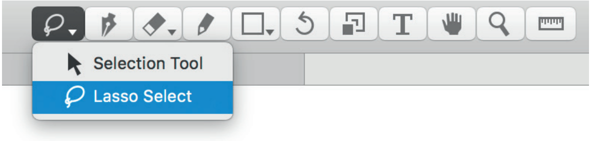
To move a handle, simply drag it with your mouse. Or select it and use your arrow keys. If more than one handle is selected, you can move them all simultaneously. Moving one or more handles while holding down the Option key preserves their angles. Again, add the Shift key for increments of 10, or the Cmd key for increments of 100.

While moving a handle next to a smooth (green) node, you can hold down Ctrl and Option simultaneously to mirror the length and angle of the adjacent handle on the other side of the node.

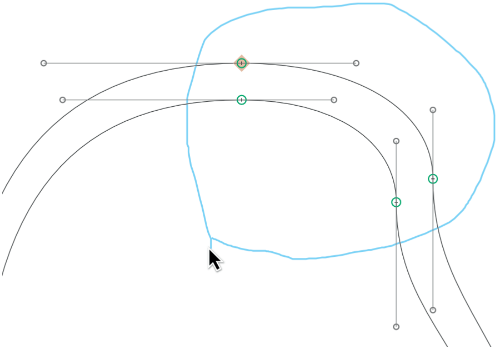
Click anywhere on the segment and drag to quickly change the shape of the segment. Option-drag a segment to change the handle lengths, but keep their respective angles.

2.3.3 Lasso Tool

Click and hold the Selection tool to switch between the Selection tool and the Lasso tool.



With the Lasso Tool, you can draw around the area in which you want to select all points. To do so, click and drag the mouse pointer. When you release, all nodes and handles inside the enclosed area will be selected.



2.3.4 Converting Nodes and Segments

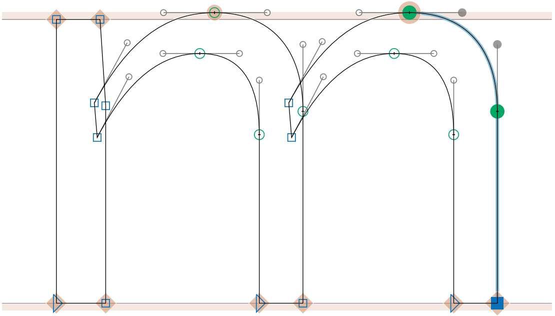
Convert between (green) smooth connections and (blue) corners by double clicking an on-curve point, or by selecting one or more nodes and pressing Return.

The *Glyph* > *Tidy up Paths* command (Cmd-Opt-Shift-T) applies heuristics to set the appropriate mode for all nodes at once, or all selected nodes if there is a selection. It also removes superfluous points, e.g., handles on a straight segment, or an on-curve point exactly in line between two others.

Tip: Quickly add handles by Option-clicking on the outline between two nodes.

Option-clicking a line segment converts it into a curve segment, i.e., adds handles. To convert a curve back into a line segment, select and delete one or both of its handles.

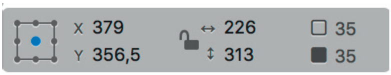
2.3.5 Nodes in Alignment Zones



Nodes located exactly on a vertical metric line are highlighted with a beige diamond. Inside an alignment zone, the highlighting assumes the shape of a circle. This helps controlling the position of nodes even at small zoom scales.

2.3.6 Scaling and Rotating

The attributes of the current node selection are shown in the gray Info box (*View* > *Show Info*, Cmd-Shift-I):



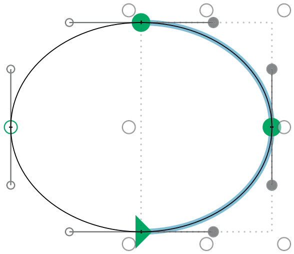
X 379
Y 356,5
↔ 226
↕ 313
□ 35
■ 35

Tip: In all number input fields throughout the application, you can use the up and down cursor keys to increase or decrease the value. Simultaneously holding down the Shift key gives you increments of 10.

When more than one node is selected, you can scale or move the selection by changing the numbers for its position (x and y) and its dimensions (↔ for width, ↕ for height) in the Info box. Set the transformation origin with the nine reference points on the left. Close the lock symbol 🔒 to scale width and height proportionally. Open the lock 🔓 to distort the selection, i.e., scale width and height independently from each other. You can use the up and down arrow keys to step through the numbers. Hold down Shift for increments of 10.

When multiple nodes are selected, the number next to the solid square will indicate how many nodes and handles are currently selected. The number next to the outlined square represents the total number of on-curve and off-curve points on the current glyph.

You can also rotate and scale your selection manually with the Rotate tool ↺ (shortcut R) and the Scale tool ⤢ (shortcut S). With one of these tools, click anywhere on the canvas to set the transformation origin, then click and drag to transform the current selection. Hold down the Shift key to rotate in steps of ninety degrees, or scale proportionally.



When more than one node is selected, you can choose to display a bounding box for the selection with *View* > *Show Bounding Box* (Cmd-Opt-Shift-B). Drag any of the white knobs to scale with the opposite end of the selection as transformation origin. Hold down the Shift key to scale proportionally, and the Option key to use the center as transformation origin.

More path transformations are possible via the Palette. See section 3.4, 'Transformations' (p. 38) for further details.

2.3.7 Aligning

Choose *Glyph* > *Align Selection* (Cmd-Shift-A) to quickly align all selected points. The command aligns both nodes and handles. Glyphs Mini will then automatically choose between horizontal and vertical alignment, whichever is smaller for the current selection. The *Align Selection* command respects the transformation origin of the gray Info box. Alternatively, you can either set the width or the height value of two or more selected points to zero in the Info Box.

You can center an anchor horizontally between to points if you run the *Align Selection* command while two points and one anchor are selected.

Running the *Align Selection* command while exactly one point and one component are selected, will align the origin point of the component to the selected node. The node keeps its position. The origin point is where the baseline crosses the left sidebearing when the italic angle is zero. If the component contains an anchor called 'origin', Glyphs Mini will use the position of that anchor instead of the origin point for aligning the component to a node.

If the italic angle is not zero, instead of the left sidebearing, an imaginary vertical line that crosses the slanted LSB at half x-height is used. In that case, the origin point is where this line crosses the baseline.

Applying *Glyph* > *Align Selection* on a single node will try to move the node over the nearest node in the background.

Glyph > *Align Selection* works on individual nodes. To also align partial paths, complete paths or components to each other, use the *Transformations* section of the Palette sidebar (Opt-Cmd-P). See chapter 3.4, 'Transformations' (p. 38), for more details.

2.3.8 Duplicating Paths

To quickly duplicate a complete path, first select it entirely by double clicking on or near it. Repeat with the Shift key to add or subtract paths to the selection. Then, hold down the Option key while you click and drag a copy of the paths into their new position.

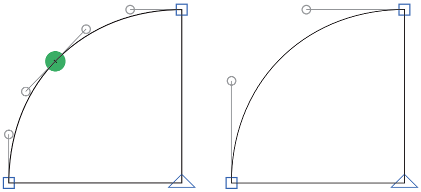
Alternatively, you can simply copy (Cmd-C) and paste (Cmd-V) the selected artwork. This method may make more sense if you plan to position the new paths with the keyboard rather than with the mouse.

Option-dragging partial paths will duplicate the selected segments. This can be helpful when replicating glyph parts like serifs or spurs.

2.3.9 Deleting Nodes

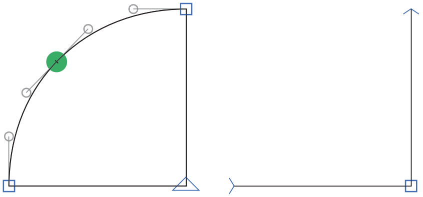
Simply select a node and press the Delete key to delete the node. Alternatively, you can use the Erase tool ◆ (shortcut E).

Glyphs Mini will keep the path closed and try to reconstruct the path segment without the node:

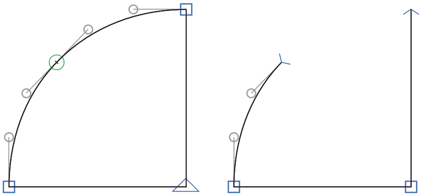


Tip: A quick way to get rid of a path segment between two nodes is to insert a point with the Draw tool (shortcut P) and immediately Option-delete it.

Hold down the Option key to break the path, i.e., remove the node and both path segments surrounding the node:

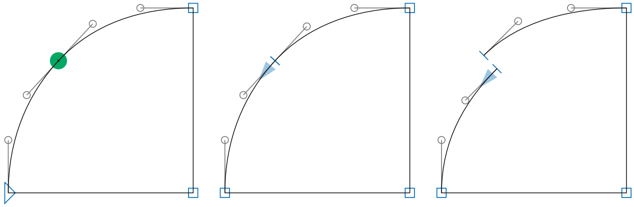


To get rid of a segment between two on-curve nodes, switch to the Erase tool (shortcut E or Shift-E), and while you hold down the Option key, click on the path segment. Alternatively, you can select a handle and press Opt-Delete. Or select the complete segment, including the on-curve nodes at the edges, and press Opt-Delete. This also works for multiple segments, adjacent or non-adjacent:



2.3.10 Opening and Closing Paths

With the Draw tool (shortcut P), click on a node to open the path in the position of the node. Open path endings are marked by short blue perpendicular lines. You can now drag them apart with the Select tool (shortcut V).

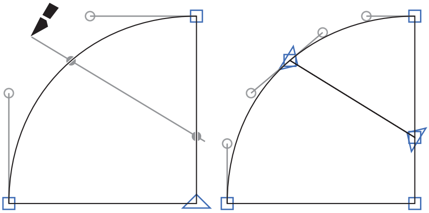


To close the path again, simply drag an open line ending on top of another with the Select tool (shortcut V). Alternatively, you can close all open paths in a glyph by right-clicking anywhere in the canvas and choosing *Close Open Paths* from the context menu. Or select two open ends, and choose *Connect Nodes* from the context menu.

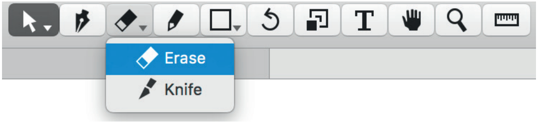
2.3.11 Cutting Paths

With the Knife tool (shortcut E or Shift-E), click and drag a line across a path, to cut the outline into two separate outlines. Glyphs Mini will close the two resulting paths along the cutting line. If you cut across several overlapping paths, it will rewire the segments with each other.

Tip: When multiple tools share one icon in the toolbar such as the Knife and Erase tools, you can add Shift to the tool shortcut to toggle between the tools.



To activate the Knife tool when it is not displayed in the toolbar, click and hold the Erase tool, and choose Knife from the pop-up menu. Alternatively, you can press Shift-E.

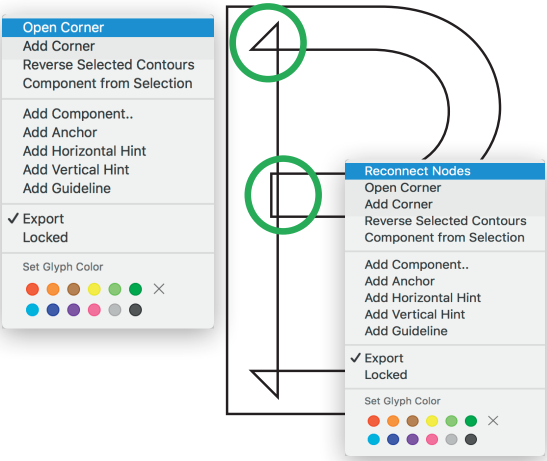


2.3.12 Remove Overlap

When *Glyph > Remove Overlap* is triggered, it immediately removes overlaps of selected paths, or all paths if none are selected. It also clears the selected glyph of all open paths and stray nodes. The function expects all outline orientations to be set correctly (see 2.3.13, 'Controlling Path Direction', p. 19). To maintain overlaps in shapes while editing, you can have overlaps removed automatically at export with the respective option in the export dialog (see 11.3, 'OpenType', p. 82).

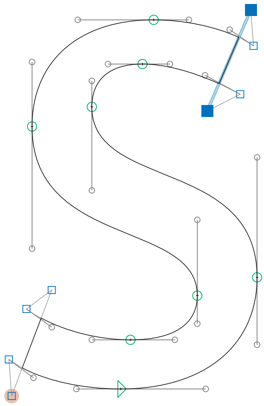
2.3.13 Resegmenting Outlines

The results of the Open Corner and Reconnect Nodes operations. Opening corners only works on (blue) corner points.



In order to recreate overlaps in a path, select an even number of nodes and choose *Reconnect Nodes* from the context menu. To open the context menu, right-click or Ctrl-click anywhere in the canvas. Glyphs Mini will then proceed to reconnect each node with its closest selected neighbor. The *Open Corner* command creates a triangular overlap at each selected point. Resegmenting your outlines this way allows you to manipulate the path segments independently from each other. It can also make interpolation significantly easier.

The size of the created overlap will be approximately half the first values entered for vertical and horizontal stems in *File > Font Info (Cmd-I)*. Thus, the created overlap should comfortably reach into the middle of your stems.



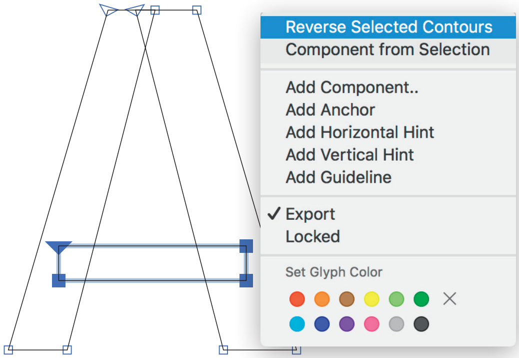
Opened corners will be considered invisible if the triangular overlaps are small enough in relation to the neighboring visible outline segments. That way, opened corners can also be placed on the outside of paths. If the overlap size goes beyond the threshold size, then they will become visible again. This is useful for editing bent finials, as in a sans-serif lowercase s.

2.3.14 Controlling Path Direction

The starting point of a closed path is displayed as a triangle indicating the path direction. Again, green and blue denote a smooth connection or a corner, respectively. On a closed path, you can make any on-curve node the first node by picking *Make Node First* from its context menu.

Tip: The easiest and quickest way to get a hole in a glyph is to draw both inner and outer shapes, and press Cmd-Shift-R (Correct Path Direction).

All outer (black) paths need to run counter-clockwise, while (white) counters must go clockwise. You can change a path's direction by selecting it and choosing *Glyph* > *Reverse Contours* or *Reverse Selected Contours* from the context menu. One node on each path will suffice as path selection in this case. When no path is selected, you can use *Glyph* > *Reverse Contours* to toggle all path directions in a glyph.



Glyph > *Correct Path Direction* (Cmd-Shift-R) will perform an informed guess and find the right path orientation for all contours in selected glyphs. This will also rearrange the contour order, and reset the starting points of all paths to the bottom left nodes.

2.3.15 Extremes and Inflections

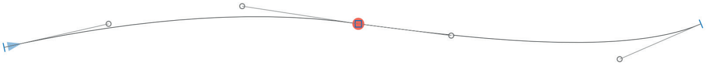
Extrema are all positions on a path with a completely horizontal or vertical tangent. An inflection is the position in some path segments where the segment changes its bend from clockwise to counterclockwise or vice versa. It is considered good practice to have nodes on extremum points. Some font technologies, like hinting, and some path operations, like nudging, see section 2.3.2, ‘Moving Selected Nodes and Paths’ (p. 12), require nodes at extremum positions. Some operations, like offsetting a curve (see 4.2.2, ‘Offset Curve’, p. 42), work better with inflection points. Also, some font renderers may behave unexpectedly if such nodes are not in place. Also, you may want to avoid additional nodes in order to keep the overall file size as small as possible, e.g., for webfont production.

You can insert nodes on extremum and inflection points by Shift-clicking a segment with the Path tool (P). A node will be inserted at the nearest extremum or inflection.

Alternatively, you can also choose *Glyph* > *Add Extremes* and nodes will be added at extremes on all paths of the active glyph. Glyphs Mini will not add an extreme if the resulting segment would be very short. In this case, it assumes that the node placement was intentional. On the other hand, if a node is only very slightly off the extremum position, Glyphs Mini will attempt to preserve the outline shape while moving the node into the extremum position and turning the surrounding handles entirely vertical or horizontal.

2.3.16 Duplicate Nodes

When two adjacent on-curve nodes share the same coordinates, thus producing a zero-length segment, they are highlighted with a red circle. You can merge these nodes with *Glyph* > *Tidy Up Paths* (Cmd-Opt-Shift-T).



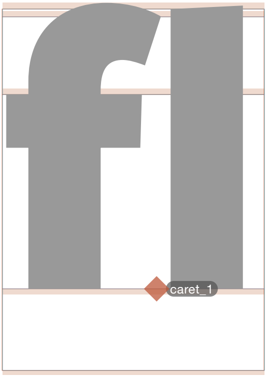
2.4 ANCHORS

2.4.1 Compound and Positioning Anchors

Anchors are special points that fulfill multiple tasks in Glyphs Mini. Primarily, they serve as a connecting pivot for automatically aligning components, corners and caps, as well as for Mark-to-Base and Mark-to-Mark Positioning (see 7, 'Reusing Shapes', p. 61). These anchors adhere to certain naming conventions. For more details on these uses, see section 7.1.5, 'Anchors' (p. 63). When an 'origin' anchor is placed inside a glyph, it can be used for aligning a component, see section 2.3.7, 'Aligning' (p. 15) to a regular path node.

2.4.2 Ligature Carets

Ligature caret positions are the positions where the text entry cursor is displayed when placed somewhere inside a ligature. In a ligature glyph, these positions are defined by special anchors on the baseline. They must be named 'caret', followed by an underscore suffix, e.g., 'caret_1', 'caret_2', etc. The suffix needs to be different for each anchor, because anchor names must be unique inside each glyph. The numbering order does not matter, the numbers are used exclusively for differentiation.



Glyph > *Set Anchors* (Cmd-U) will insert appropriate caret anchors in properly named ligature glyphs, i.e., the names of the characters in the ligature, connected by underscores, e.g., s_t or f_f_h. For the glyph naming convention employed by Glyphs Mini, see section 5.6, 'Names and Unicode' (p. 53). At export, Glyphs Mini will use the caret information to build so-called LigatureCaretByPos instructions in the GDEF OpenType table. At the time of this writing, the only known software supporting ligature caret positioning are Mac applications that make use of the Cocoa text engine. Adobe and Microsoft apps ignore this information.

2.4.3 Adding, Editing and Removing Anchors

Insert an anchor by Ctrl-clicking or right-clicking inside a glyph, and choosing *Add Anchor* from the context menu. An anchor called 'new anchor' will be placed at the click position. Its name is immediately selected for renaming, so you can enter the name right away. Confirm the new name by pressing the Return key or clicking anywhere in the canvas.

To change an anchor name, select it, press the Return key, and type a new name. Alternatively, click or double click its name in the gray Info box when the anchor is selected, and type a new name there.

The built-in glyph info database has default anchors associated with some glyphs. To have them added automatically, choose *Glyph* > *Set Anchors* (Cmd-U).

Select an anchor by clicking on the red dot that represents it. You can select the next or previous anchor by pressing the Tab key, or Shift-Tab, respectively. Select multiple anchors by Shift-clicking them. Names are only shown for selected anchors. If you want to select all anchors, you may need to run *Edit* > *Select All* (Cmd-A) twice, since selecting all will select all paths, and only select all anchors and components if all available paths have been selected already.

Move an anchor like you would move a node, i.e., either with the mouse, the arrow keys, or through the gray Info box. An anchor is diamond-shaped if it is placed exactly on a metric line such as the x-height or the baseline. It is circle-shaped in all other cases. To quickly duplicate an anchor, Option-drag it. Since anchor names must be unique inside a glyph, an underscore will be added to the end of the name.

Remove one or more selected anchors by simply pressing the Delete or Backspace key.

2.4.4 Mark to Base Positioning

Glyphs Mini can automatically build the 'mark' (Mark to Base) feature from combining (non-spacing) diacritical marks containing underscore anchors, e.g., '_bottom' or '_top', in combination with all base glyphs that carry corresponding base anchors, e.g., 'bottom' or 'top'. Latin combining diacritical marks carry a 'comb' at the end of their names, e.g., acutecomb or macroncomb.

Combining diacritical marks have their own Unicode values and thus can be typed or inserted in a text. This way, a user can place any mark on any base letter, by first typing the regular letter, and then inserting the combining mark.

Alongside the 'mark' feature, Glyphs Mini will also build the 'ccmp' (Glyph Composition and Decomposition) if glyphs like idotless and jdotless are present. See chapter 13.1, 'Automatic Feature Generation' (p. 90), for further details.

2.4.5 Mark to Mark Positioning

If both underscore and regular anchors are present in a combining diacritical mark, Glyphs Mini will also automatically build the 'mkmk' (Mark to Mark) feature. A user will then be able to stack any combining mark on any other combining mark carrying both anchors.

2.5 GUIDELINES

2.5.1 Magnetic Guidelines

If you drag a node selection across the canvas, red lines will appear, indicating when your selection is aligned with other nodes or a vertical metric. You can temporarily deactivate magnetic guidelines by holding down the Ctrl key.

Likewise, dragging a node or any other object will snap to all nodes and handles on paths, as well as in components. When you move close to a node in a component while dragging, Glyphs Mini will fade in small representations of the nodes inside the component. Again, you can disable node snapping by holding down the Ctrl key.

2.5.2 Local and Global Guidelines

To toggle the display of guidelines in Edit view, choose *View > Show Guides* (Cmd-Shift-L)

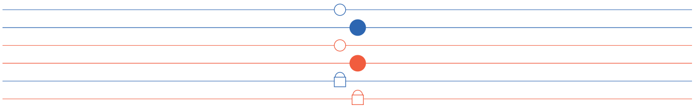
To add a static, local guideline to the currently displayed glyph, right- or Ctrl-click to open the context menu and choose *Add Guideline*. A horizontal local guideline will be added at the click position. If two nodes are selected while adding the guideline, it will be laid through the nodes. You can select it by clicking on it, alter its position by clicking and dragging its knob. Double click the circle to turn it perpendicular to its original orientation. Click and drag a selected guideline anywhere else but on the knob to rotate it.

Pro User Tip: In order to quickly create a guideline in measurement mode, temporarily activate the Measurement tool by simultaneously holding down Ctrl-Opt-Cmd, and press the G key while dragging a measurement line.

To create a global guideline, i.e., a guideline that appears in all glyphs throughout the font, hold down the Option key while navigating the context menu, and choose *Add Global Guideline*. To toggle a guideline between local and global, select it, and from the context menu, choose *Make Global Guideline* or *Make Local Guideline*, respectively.

Quickly duplicate one or more local or global guidelines by selecting them and holding down the Option key while dragging them to a new position.

You can lock one or more guidelines by choosing *Lock Guidelines* from their context menu. A locked guideline cannot be selected and will display a lock symbol instead of its knob. To unlock a guideline again, Ctrl- or right-click the knob and choose *Unlock Guide* from the context menu.

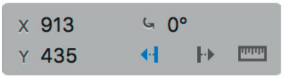


From top to bottom: unselected and selected local guides, unselected and selected global guides, locked local guide, and locked global guide.

Select a guideline by clicking anywhere on it: a filled knob indicates a selected guideline. Press the Tab key to quickly select the next, or Shift-Tab to select the previous guideline. When a guideline is selected, you can move it using your arrow keys (add Shift or Cmd for larger increments), or by dragging its knob with the mouse, just like you would move a regular node. Change the angle by dragging the guideline anywhere off the node. You can also enter values for its position and its angle in the gray Info box (Cmd-Shift-I).

You can also mirror the guidelines via the Transformations panel (see 3.4.2, 'Mirroring', p. 39).

By default, a guideline will be set relative to the left sidebearing. However, if you change the alignment of the guideline by clicking on the Alignment icon |▸ in the Info box, the guideline will stay fixed relative to the ◂| right sidebearing or ◂|▸ relative to both sidebearings. This can be useful for slanted guidelines, especially when they are global or when the right sidebearing is changed. With the measurement checkbox 📏, you can turn it into a measurement guideline. For more details, see section 2.9, 'Measuring' (p. 30).



X 913 0°
Y 435

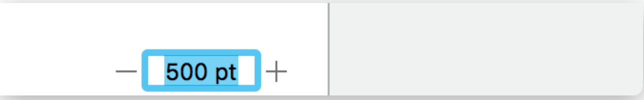
2.6 GLYPH DISPLAY

2.6.1 Zooming

There are many ways to zoom in and out in Edit view. If you are on a MacBook or have a trackpad to your disposal, you can use pinch and stretch gestures. Or hold down the Option key and use a scroll gesture or the scroll wheel of your mouse. Or activate the Zoom tool 🔍 (shortcut Z) and click in the canvas to zoom in, Opt-click to zoom out. Alternatively, click and drag across an area, and it will be zoomed to fill the window. You can temporarily activate the Zoom tool by holding down Cmd-Space for zooming in, or Opt-Cmd-Space for zooming out. If Cmd-Space collides with another shortcut on your system, you can try pressing Space before you add the Cmd key.

Tip: You may need to change your Spotlight shortcut in the System Preferences for the Cmd-Space shortcut to work in Glyphs.

Or you can use the zoom commands from the *View* menu: *Zoom In* (Cmd-plus) and *Zoom Out* (Cmd-dash). *Zoom to Active Glyph* (Cmd-zero) will maximize the area between ascender and descender in the window. *Zoom to Actual Size* (Cmd-Opt-zero) will zoom one font unit to the size of one screen point, i.e., one pixel on a classic low-resolution screen, two pixels on the new high-resolution Retina screens.



500 pt

Or you can use the zoom buttons in the bottom right corner of the window. Alternatively, you can set the zoom value numerically by entering a point height in the field between the buttons. The value specifies at which size 1000 font units will be displayed. Since macOS assumes a screen resolution of 72 ppi, one point corresponds to one actual screen pixel, so if you enter a value of 1000, one unit will zoom to one pixel. On Retina displays, zooming is done according to their higher resolution, e.g., at 144 ppi, a unit will be zoomed to two screen pixels.

2.6.2 Panning

You can use trackpad panning gestures, i.e., dragging two fingers across your trackpad, to change the canvas excerpt of the window. Use the wheel of your mouse to scroll vertically, hold down Shift to scroll horizontally. Or, drag the scroll bars on the right and bottom edges of the displayed canvas. If you have difficulties grabbing the scroll bars, consider changing the scroll bar behavior in the *General* section of System Preferences. Alternatively, you can switch to the Hand tool ✋ (shortcut H) and drag the canvas around, or simply hold down the space bar to temporarily switch to the Hand tool. If you are in text mode, simply pressing the space bar would add a space to your sample text. To avoid that, you can press Cmd-Space and subsequently release the Command key.

Tip: Cmd-Space may interfere with the system-wide shortcut for invoking Spotlight. You can change it in the System Preferences.

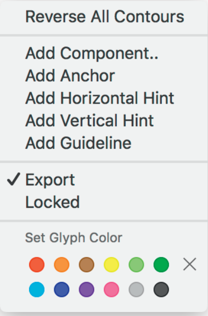
2.6.3 View Options

In the *View* menu, you can toggle several options that influence the glyph display in Edit view.

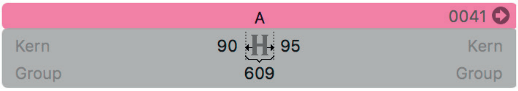
- *Show Nodes* displays on- and off-curve nodes of paths,
- *Show Metrics* shows the vertical and horizontal metrics of the glyph,
- *Fill Preview* displays closed paths with a black fill.

2.6.4 Glyph Colors

In Edit view, you can set glyph label colors via the context menu. Right- or Ctrl-click anywhere in the canvas of an active glyph, and pick the color from the context menu.



The glyph color will be displayed in the gray Info box (*View > Show Info,* Cmd Shift-I):



This corresponds with the display of label colors in Font view. See section 5.3.5, 'Color Label' (p. 48).

2.7 BACKGROUND

Each glyph has a background layer, usually simply referred to as 'background'. The background is useful for temporarily storing a path, or for tracking changes and comparing outlines before and after a manipulation.

While working in the foreground, objects on the background can be displayed as subtle gray outline with *View* > *Show Background* (Cmd-Shift-B). If this option is active while you are working in the background, the foreground objects will be displayed this way. When the background is displayed, snapping will also work with objects on the background layer.

To switch to the background, choose *Glyph* > *Edit Background* (Cmd-B). The window display will darken slightly to indicate that you are in the background layer. You can set a different color in *Glyphs Mini* > *Preferences* > *User Settings*.

Glyph > *Selection to Background* (Cmd-K) replaces the current content of the background with the active selection, this works in reverse when the background is active.

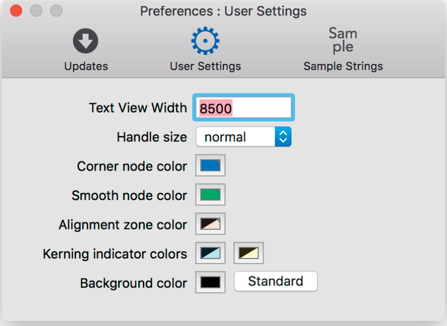
Via *Glyph* > *Assign Background*, you can copy the outlines of another font file into the background layer of all selected glyphs. You can also put the same font into its own background in order to keep track of your own changes. Selecting all glyphs and choosing *Glyph* > *Selection to Background* (Cmd-K) has the same effect.

2.8 ENTERING TEXT

Tip: To quickly switch back from text entry to editing the current glyph, press the Esc key.

The Edit view also acts as a simple text editor, and as such, allows you to edit your glyphs in the context of a whole word or even a sentence. As long as the Text tool (T) is active, it accepts input via the current input source set in the System Preferences, the Keyboard Viewer, the Character Viewer, but also via the clipboard or *Edit* > *Emojis & Symbols*. With *Edit* > *Start Dictation* (Cmd-Cmd), you can even use the speech recognition features provided by macOS and dictate your text.

The Edit view has a preset line width. You can set the maximum line width in the application preferences. Select *Glyphs Mini* > *Preferences* > *User Settings* and edit the *Text View Width* value accordingly. The value in design units. Their relation to the em size is defined by the *Units per Em* value in *File > Font Info*.



Text View Width: 8500

Handle size: normal

Corner node color

Smooth node color

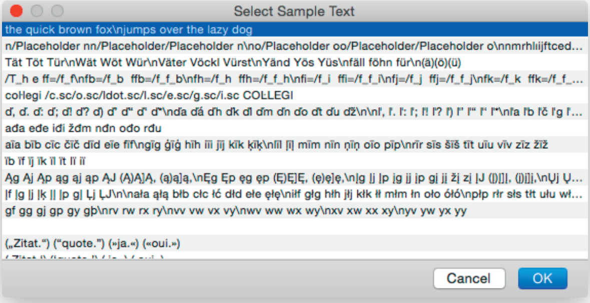
Alignment zone color

Kerning indicator colors

Background color

Keep in mind that a lot of text in the Edit area will slow down the application, especially if the Preview is displayed. If you want to test your font with large chunks of text, it is advisable to use the Adobe Fonts folder, see section 2.11.2, ‘Previewing in Adobe Applications’ (p. 36) for further details.

2.8.1 Sample Texts



```
the quick brown fox\njumps over the lazy dog
n/Placeholder nn/Placeholder/Placeholder n\no/Placeholder oo/Placeholder/Placeholder o\nnmrhlıijftced…
Tät Töt Tür\nWät Wöt Wür\nVäter Vöckl Vürst\nYänd Yös Yüs\nfäll föhn für\n(ä)(ö)(ü)
/T_h e ff=/f_f\nfb=/f_b  ffb=/f_f_b\nfh=/f_h  ffh=/f_f_h\nfi=/f_i  ffi=/f_f_i\nfj=/f_j  ffj=/f_f_j\nfk=/f_k  ffk=/f_f_…
col·legi /c.sc/o.sc/ldot.sc/l.sc/e.sc/g.sc/i.sc COL·LEGI
ď, ď. ď: ď; ď! ď? ď) ď' ď" ď' ď"\nďa ďá ďh ďk ďl ďm ďn ďo ďt ďu ďž\n\nľ, ľ. ľ: ľ; ľ! ľ? ľ) ľ' ľ" ľ' ľ"\nľa ľb ľč ľg ľ…
ađa eđe iđi žđm nđn ođo rđu
aīa bīb cīc čīč dīd eīe fīf\ngīg ġīġ hīh iīi jīj kīk ķīķ\nlīl ļīļ mīm nīn ņīņ oīo pīp\nrīr sīs šīš tīt uīu vīv zīz žīž
ïb ïf ïj ïk ïl ït lï iï
Ąg Ąj Ąp ąg ąj ąp ĄJ (Ą)Ą]Ą, (ą)ą]ą,\nĘg Ęp ęg ęp (Ę)Ę]Ę, (ę)ę]ę,\nĮg Įj Įp įg įj įp gį jį žį zį ĮJ (Į)Į]Į, (į)į]į,\nŲj Ų…
łf łg łj łķ łļ łp gł Łj ŁJ\n\nała ałą błb cłc łć dłd ełe ęłę\nıłf głg hłh jłj kłk łł młm łn oło ółó\npłp rłr słs tłt ułu wł…
gf gg gj gp gy gþ\nrv rw rx ry\nvv vw vx vy\nwv ww wx wy\nxv xw xx xy\nyv yw yx yy

(„Zitat.“) (“quote.”) (»ja.«) («oui.»)
```

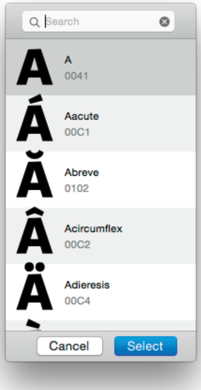
You can edit and store any number of sample texts in *Glyphs Mini > Preferences > Sample Strings* (see 12.1.3, ‘Sample Strings’, p. 88). To make an Edit tab display any of these texts, choose *Edit > Select Sample Text* (Opt-Cmd-F). In the appearing dialog, use your arrow keys or click on a line to pick the sample text and press Return or click the OK button afterwards.

2.8.2 Text Tool

Select the Text tool (shortcut T) to switch to text mode and start typing. You can enter multiple words, complete sentences, even line breaks. You can copy and paste text into and from the Edit tab. The arrow keys, the *Edit* commands and macOS Application Services work as they would in any Mac app.

The current glyph is the one to the right of the cursor. You can switch to the previous or next glyph in the font by pressing the Home and End key, respectively. Add Shift to advance through the glyphs as they are currently visible in the Font tab. This is useful when you filter glyphs in the Font view and then want to step through each one of them.

Tip: On a MacBook, you can simulate Home and End by pressing Fn-leftarrow and Fn-rightarrow.



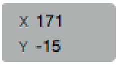
To insert one or more glyphs that you cannot or do not know how to type with the keyboard, choose *Edit* > *Find* > *Find*… (Cmd-F). In the dialog that appears, enter the glyph name, a part of it, or different parts of the glyph names separated by spaces. The dialog will show a list of available glyphs whose name contain the strings you entered. E.g., 'dier 01' will find all glyphs that have both 'dier' and 'sc' in their name, like adieresis.sc and edieresis.sc. Select the glyphs you want and press Return or click the Select button to insert them into your sample text. To select a range of glyphs, hold down the Shift key. For a non-contiguous selection, hold down the Cmd key.

2.9 MEASURING

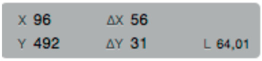
Glyphs Mini offers several ways to determine coordinates and to measure distances between points and curves.

2.9.1 Info Box

You can toggle the display of the gray Info box with *View* > *Show Info* (Cmd-Shift-I). The Info box always displays data relevant to the current selection. If there is exactly one node selected, its coordinates will be displayed.


X 171
Y -15

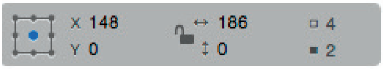
Select a handle (off-curve point, Bézier control point), and the Info box will also show its delta values (x and y difference to the on-curve point) and the total length of the handle (distance to the on-curve point).



X 96 ΔX 56
Y 492 ΔY 31 L 64,01

Tip: To quickly and precisely measure a stem or bowl width, select two nodes that indicate the width and see what the Info box displays next to the width symbol.

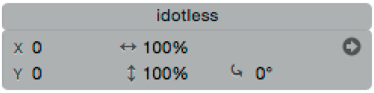
If you select more than one point, be it on- or off-curve points, the Info box will display the width (↔) and the height (↕) of the box defined by the current selection. The x and y coordinates displayed describe the position of the bounding box of the selection as indicated by the grid, i.e., the bottom center point of the grid indicates that the coordinates describe the bottom center point of the bounding box. The two numbers to the right indicate the count of nodes in the current glyph, and in the current selection, respectively.



X 148 ↔ 186 □ 4
Y 0 ↕ 0 ■ 2

In the Info box, any displayed number (except for the node counts) can be selected and changed by entering a new value. You can use the Tab key to advance to the next number slot displayed, and Shift-Tab to switch to the previous slot. Changes take effect once you press Return or tab out of a value field. Alternatively, you can use the up and down arrows for increasing and decreasing the number by one unit. Add Shift for increments of 10. If you scale a selection this way, the lock symbol toggles proportional x/y scaling, and the grid controls the scaling origin.

When a component is selected, the Info box will display the name of the original glyph the component points to, its x and y offset, its horizontal and vertical scale in percent, and its counter-clockwise rotation angle. The little arrow button in the top right corner will insert the original glyph in the Edit tab string, to the left of the current glyph, and activate it for editing. For more details on working with components, see section 7, 'Reusing Shapes' (p. 61).

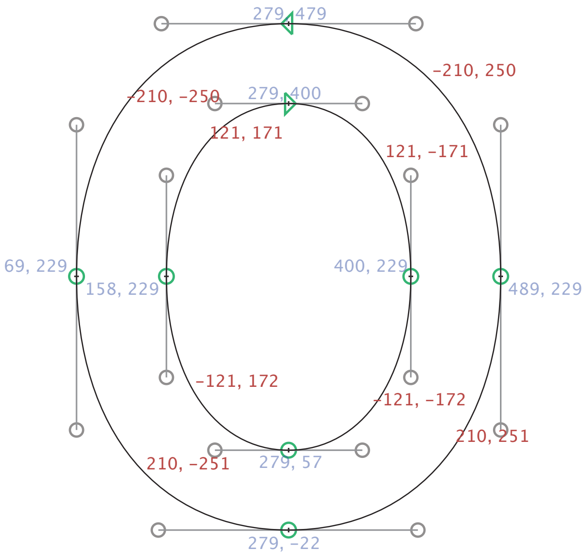


idotless
X 0 ↔ 100%
Y 0 ↕ 100% ↳ 0°

You can change the glyph the component points to by clicking on its name in the gray Info box, and choosing another glyph from the glyph list in the subsequent pop-up window. Change position, size, and rotation by manipulating the respective number values. Use the up and down arrows to step through values, and add Shift or Cmd for increments of 10 or 100, respectively. Changing the position only has an effect if the component is not automatically aligned. For more details on automatic alignment, see section 7.1.6, 'Automatic Alignment' (p. 65).

2.9.2 Measurement Tool

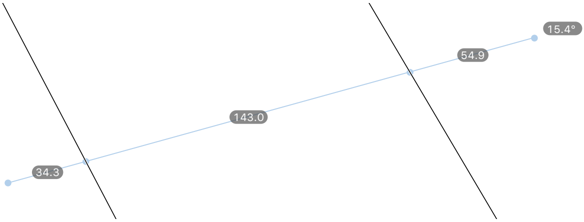
Switching to the Measurement Tool 📏 (shortcut L) allows you to see all coordinates of all nodes and anchors at once.



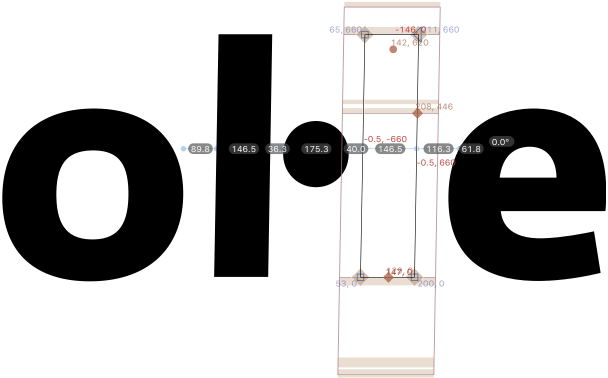
The x delta values respect the italic angle set in Font Info. So, an x delta of zero indicates a line exactly in the italic angle.

The blue numbers are the x and y coordinates of the on-curve points, the red numbers are the x and y delta values between the on-curve points. These values help you check the symmetry of your curves.

Clicking and dragging draws a ruler that displays precise measures between all of its intersections with outlines. Add Shift to drag a horizontal or vertical ruler. At the end of the ruler, its angle is displayed in counter-clockwise degrees, where zero degrees corresponds to dragging the ruler completely horizontally to the right.



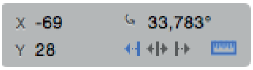
The Measurement Tool works on all visible glyphs in Edit view. So, you can drag a measurement line across multiple glyphs and see the individual distances:



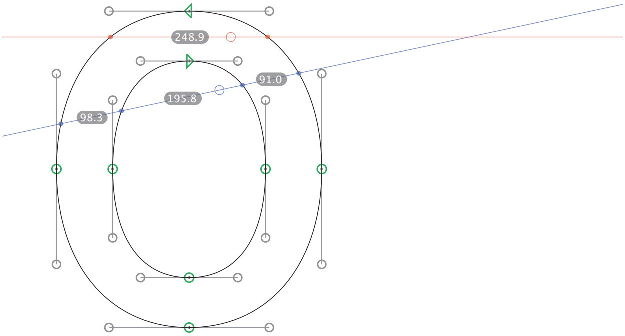
You can temporarily activate the ruler and the display of point coordinates by simultaneously holding down Ctrl, Option and Command. Pressing the G key while dragging a ruler adds a guideline in measurement mode.

2.9.3 Measurement Guidelines

Any guideline, even a global guideline, can be turned into a measurement guideline. Simply click on the guideline to select it, and then, click on the measurement symbol 📏 in the gray Info box. You can handle measurement guidelines exactly as regular guidelines, see section 2.5, 'Guidelines' (p. 24) for more details.



Similar to the Measurement tool, guidelines in measurement mode will display the distance between their intersections with outlines or components. Contrary to the tool, they always do so as long as guidelines are shown, no matter which tool is active.



2.10 IMAGES

2.10.1 Adding Images

You can add all image files supported by macOS, including PDFs, to any glyph simply by dragging them onto a glyph cell in Font view, or into the active glyph in Edit view. You can add many images at once by choosing *Glyph* > *Add Image* and selecting any number of image files in the subsequent Open File dialog. The images will be placed in the current glyphs that correspond to the file names. E.g., if your scans are called *Thorn.png* and *thorn.jpeg*, they will be placed in the glyphs *Thorn* and *thorn*, respectively. For this to work, you may have to place scans for uppercase and lowercase letters in separate folders. That is because, by default, the macOS file system is set to be case-insensitive, and thus, same names with different capitalizations cannot coexist in the same macOS folder.

In the Glyphs Mini document, only the relative path of the image is stored. Thus it is a good idea to keep them in a subfolder next to the Glyphs Mini file. If the path to the placed image is outdated or broken, it will be indicated with a missing image symbol:

You can toggle the display of images via *View* > *Show Image*. Even if this setting is off, images will be displayed as long as the glyph is empty, i.e., contains neither paths nor components.

2.10.2 Manipulating Images

By default, images will be scaled to a size where one DTP point corresponds to one font unit, and placed at the origin point of the glyph. If you plan to place a lot of images, consider preparing their size accordingly.

Move the image by dragging it to the desired position. When an image is selected, you can resize it with the bounding box (Cmd-Opt-Shift-B) or the Scale tool (S). You can rotate it with the Rotate tool (R). The *Transformations* palette also works for images.

The gray Info box (Cmd-Shift-I) allows you to numerically control position (x and y) and dimensions (↔ for width, ↕ for height) of the selected image. You can also rotate the image counter-clockwise by manipulating the degree figure next to the curved arrow symbol. A click on the right-pointing arrow symbol will reveal the original image file in Finder. A click on the lock symbol freezes the image status until you unlock it again via the context menu.

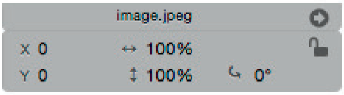


image.jpeg

X 0 ↔ 100%

Y 0 ↕ 100% 0°

Besides locking and unlocking an image, the context menu lets you crop the image to the glyph bounds (width, descender, and ascender), and reveal the image file in Finder.

2.11 PREVIEWING AND TESTING

2.11.1 Previewing in macOS

Because of the described difficulties with the macOS font cache, we do not recommend testing unfinished fonts in Font Book.

Because of the complicated font caching mechanism employed by macOS, simply overwriting a previously installed font can lead to a range of problems. Amongst other things, the font may not show the changes you made in Glyphs, or may show wrong glyphs for the characters typed, display empty or garbled glyphs, or even disappear from the font menu. In order to avoid such font cache difficulties, you can change the family name in *File* > *Font Info* > *Font* (Cmd-I) every time you export, e.g., by adding an incremental number ('MyFont 01', 'MyFont 02', 'MyFont 03' etc.) or a letter combination ('MyFont AA', 'MyFont AB' etc.). This scheme ensures a more consistent font menu ordering if you have many versions of your font installed.

If you do run into cache problems, close all running applications and remove all instances of your font from Font Book or the Fonts folder. Then, open the Terminal application and type these lines, each of them completed by pressing the Return key. The first command will prompt you for your administrator password. You will not see bullets while you type your password.

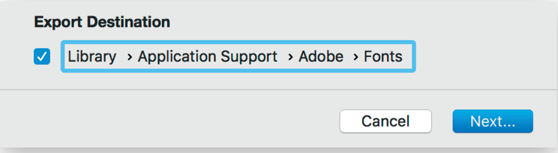
```
sudo atsutil databases -remove
atsutil server -shutdown
atsutil server -ping
```

After these Terminal commands, you need to restart the Mac for the changes to take effect. If the problems persist, boot the Mac in safe mode, i.e., hold down Cmd-S at startup until the Apple logo appears on screen. Recreate caches by holding down the Shift key while restarting and logging in.

2.11.2 Previewing in Adobe Applications

For a complete preview including things like positioning features and menu ordering, just pick *File* > *Export* (Cmd-E), pick the *OpenType* export, and choose /Library/Application Support/Adobe/Fonts/ as *Export Destination*. The font becomes immediately available in all Adobe applications. Any previously saved instance of the font in this folder will be overwritten. The font will not be available outside Adobe apps, but this is a convenient way to circumvent any font cache problems in macOS.

If the Fonts folder does not exist, you can create it right in the *Open Folder* dialog by pressing Cmd-Shift-N. Only in case they were already running while the folder was created, you have to restart Adobe apps this once for the change to take effect. After that, the Adobe font menus will update immediately every time the font is exported.



Export Destination

☑ Library › Application Support › Adobe › Fonts

2.11.3 Previewing in Web Browsers

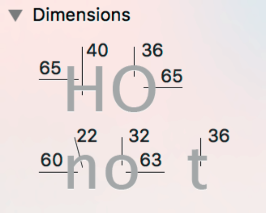
You can also export webfonts via *File* > *Export* > *Webfonts (WOFF)*. For repeated exports, you can set an *Export Destination* in the same dialog. If you set up an HTML file containing a test page that uses the webfonts, you can open the file in a web browser, and reload the page every time a new version of the font was exported. Since the webfont files are overwritten during the export, reloading the page in the web browser should show the new version of the font. Some web browsers also employ font caches, though. If a reload does not show the new font, most Mac browsers offer you to force a clean reload of the page by holding down the Shift key while clicking on the reload button, or in addition to the keyboard shortcut for reloading (usually Cmd-R).

3 Palette

3.1 PALETTE SIDEBAR

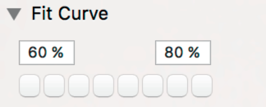
You can open the Palette sidebar with the sidebar button in the top right corner of the window, or via *Window* > *Palette* (Opt-Cmd-P). By default, the Palette has four sections. You can collapse or expand them by clicking on their title or the triangle next to it.

3.2 DIMENSIONS



The entries in the *Dimensions* section have no effect on the font. They serve as a cheat sheet for your personal design process. Simply enter values for the displayed measurements.

3.3 FIT CURVE



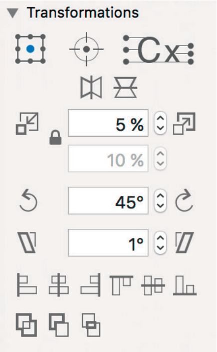
60 %   80 %

Smaller curves, e.g. counters, generally need higher curvature percentages than larger curves, e.g., on the outside of a bowl.

The *Fit Curve* panel helps creating curves with matching or corresponding curvatures. Select percentages by setting the left field to the minimum value and the right field to the maximum value. The eight buttons in between represent equal steps between these two values. Alternatively, you can use Ctrl-Opt-1 through 8 as keyboard shortcuts for the buttons. For the *Fit Curve* function to take effect, at least one handle must be selected. *Fit Curve* also works on multiple segments, provided that at least one handle of each segment is selected.

The numbers describe the percentage length of the handle. The distance from the curve point to the intersection of the handles equals 100 percent. 56 percent give you an elliptic curvature. The minimum value you can set is 1 percent, the maximum value is 100 percent. If you want a completely flat segment, select one of the handles and press the Delete key in order to turn the segment into a line.

3.4 TRANSFORMATIONS



5 %

10 %

45°

1°

The bottom section of the palette is reserved for object transformations. Once you have set the transformation origin, you can, from top to bottom:

- mirror the selection ◫ horizontally or ⊟ vertically,
- scale the selection ⇲ down or ⇱ up, horizontally and/or vertically, by percentage values,
- rotate the selection ↺ counterclockwise or ↻ clockwise by degrees,

- skew the selection left or right by degrees,
- align nodes, complete paths or components to each other,
- and execute boolean operations with two paths.

The transformations work in both the Font and the Edit view. Depending on the selection, the transformations work on paths, parts of paths, or complete glyphs, even when multiple glyphs are selected. Note that, if you execute a transformation while a unit grid is active (see 6.10, 'Grid Spacing and Subdivision', p. 59), the placement of the on-curve nodes will be subject to rounding.

3.4.1 Transformation Origin

Except for aligning, all transformations available in the palette respect the transformation origin you set in the top row. This can either be a grid point calculated relative to the bounding box of the selection, similar to the grid in the gray Info box (Cmd-Shift-I).

Or, it can be a manually set reference point. To change the reference point, you need to switch to either the Rotate (shortcut R) or the Scale (shortcut S) tool, and click once in the canvas.

Or, it can be Cx one of the metrics taken from the entries in *File* > *Font Info* (Cmd-I): baseline, (half or full) x-height and (half or full) cap height.

3.4.2 Mirroring

Tip: If you copy and paste the paths of the lowercase n into your lowercase u, you can mirror horizontally and vertically around half the x-height, rather than around the selection center. This way, the overshoot is respected.

Mirroring can be applied to point selections, complete selected paths, guidelines and selected components, or any combination thereof. Mirroring a corner component will turn a left corner into a right corner and vice versa.

3.4.3 Reversible Transformations

The corresponding buttons to the left and right of the Scale, Rotate, and Skew fields, are each other's reverse transformations. Thus you can reverse any of the transforms with their complimentary buttons, and the points will assume their previous position, even when a unit grid is in effect and rounding has been applied to point coordinates.

If you want to scale down all coordinates by half, you need to enter 100 percent and click the reverse-scale button, or enter 50 percent and click the upscale button.

Scaling takes two percentage values, for horizontal (x) and vertical (y) scaling, respectively. When the lock symbol is closed, the second value will be ignored and the first value applies to both x and y scaling. The entered values refer to positive scaling which is activated by pushing the upscale button. This means that the reverse-scale button does not scale down by the entered values, but reverses the positive transformation. This way, you can always undo a positive scale with the reverse-scale button and vice versa.

The Rotate and Slant functions work on any selection. Again, the buttons compliment each other, by reversing each others' transformations.

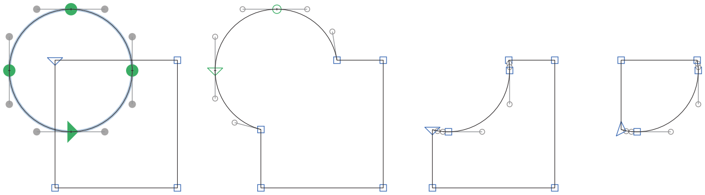
3.4.4 Aligning

The Align buttons work on complete and partial paths, as well as components. If a path is selected only partially, the selected nodes are aligned to each other. Aligning via the buttons is always done relative to the bounding box of the selection.

For most use cases, a quicker way to align selected points is *Glyph* > *Align Selection* (Cmd-Shift-A). This command respects the grid setting for the transformation origin. See section 2.3, 'Editing Paths' (p. 12), for more details.

3.4.5 Boolean Operations

The bottom row of buttons allows for boolean operations: union, subtraction, and intersection. Union and intersection work with the selected paths, or all paths if none are selected. The subtraction operation subtracts the selected paths from the unselected ones. Or, if no paths are selected, it subtracts the front-most, i.e., last, path from all others. In order to achieve consistent results, both groups of paths involved, i.e., either the selected and the unselected paths, or the front-most and all other paths, are merged before the subtraction is executed between them.



From left to right: (1) two overlapping paths before the boolean operation; (2) union; (3) subtract; (4) intersection.

4 Filters

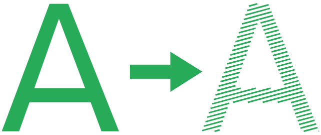
4.1 FILTERS MENU

In Edit view, or, with the Text tool (shortcut T), you can apply filters to any number of selected glyphs. In the Font view, you can apply it to all selected glyphs at once.

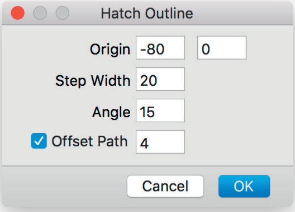
4.2 FILTERS

4.2.1 Hatch Outline

Creates hatched letters. Distance, width, and angle of the hatch-lines can be chosen. Glyphs Mini always uses the background path as source. If there is no path, Glyphs Mini will put a copy of the current path into the background. Alter the appearance of the hatching by changing the background path and applying the filter once again.



With the *Origin* option, you can change the position of the hatch strokes. The two fields represent the x and y values of the construction origin. The *Step Width* option determines the distance between individual strokes, and *Angle* their slant in degrees. The *Offset Path* value determines the thickness of the strokes, much like the *Offset Curve* filter would. Deactivate the checkbox next to it to prevent the offsetting, and only insert straight, open paths as hatch lines. This may be useful in case you plan additional subsequent transformations.



Hatch Outline

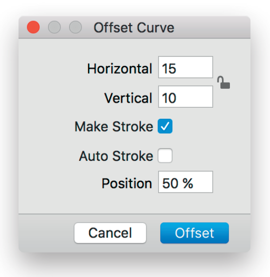
Origin -80 0

Step Width 20

Angle 15

☑ Offset Path 4

4.2.2 Offset Curve


Offset Curve

Horizontal 15

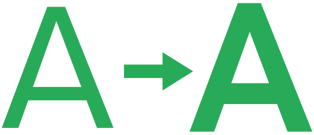
Vertical 10

Make Stroke ☑

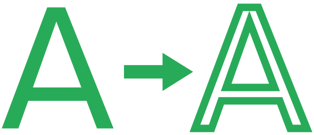
Auto Stroke □

Position 50 %

Changes the thickness of stems horizontally and / or vertically. The lock sign uses the horizontal value for both horizontal and vertical expansion. Without the *Make Stroke* option, paths will just be moved into a parallel position:

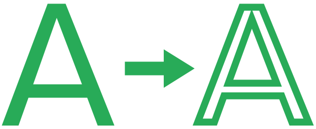


With the *Make Stroke* option, selected paths will be expanded to closed outlines:



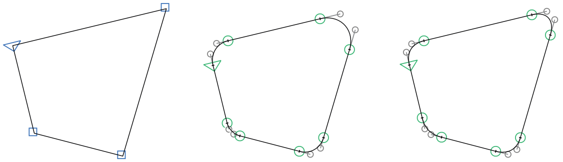
Tip: You can apply the Make Stroke option to an open path to get a thickened outline. This way, you can get an initial outline just by drawing the internal stroke.

The *Position* setting controls the distribution of the expansion. At 0%, the path will only expand to the right. At 100%, the path will only expand to the left. At 50%, the expansion will be evenly distributed to both sides of the path. Right and left sides are determined by the path orientation.
With the *Auto Stroke* option, the vertical dimensions will be kept intact. In that case, the offset position will be assumed at 50%.



4.2.3 Round Corners

Use this filter to round all selected corners of a path. To only round the outside corners, i.e., corners pointing into white background, select nothing. Supply a *Radius* value in units.

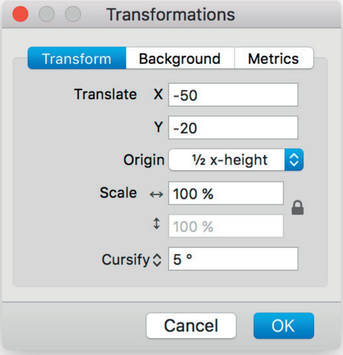


From left to right: original shape, rounded shape, and rounded shape with visual corrections.

Use the *Visual Corrections* option to create a more natural looking corner rounding. This option increases the corner radius at obtuse angles, and reduces the radius at acute angles, yielding a more natural shape.

4.2.4 Transformations

The window of the Transformations filter sports three tabs: *Transform*, *Background*, and *Metrics*.



Translate X -50

Y -20

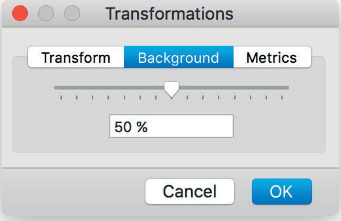
Origin ½ x-height

Scale ↔ 100 %

↕ 100 %

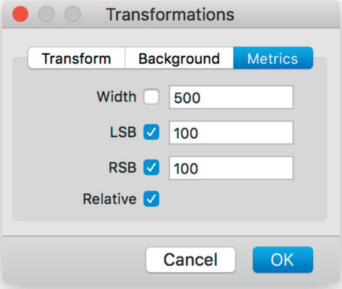
Cursify 5 °

Use *Transform* to horizontally and vertically move, scale, and skew outlines. For scaling and skewing, you can set the transformation origin to cap height, half cap-height, x-height, half x-height, and baseline. Skew without optical correction by activating the *Slant* option, or with optical correction using the *Cursify* option. Cursify requires correctly set vertical and horizontal stems in *File* > *Font Info*.



50 %

Use the slider in the *Background* tab to interpolate between front and background paths. The paths need to be compatible. This is useful if you want to experiment with a feature of a glyph, e.g., the length of a stroke. You can quickly copy a path into the background by choosing *Glyph* > *Selection to Background* (Cmd-K).



Width □ 500

LSB ☑ 100

RSB ☑ 100

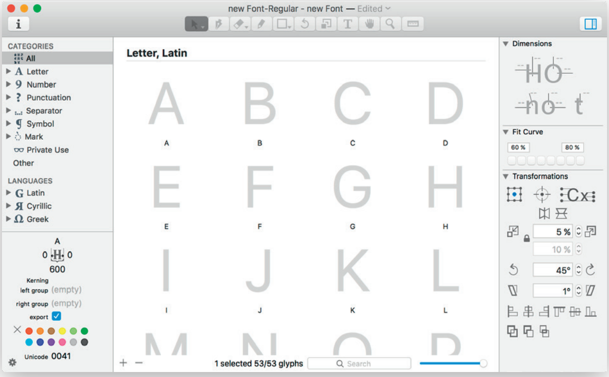
Relative ☑

In the *Metrics* tab, you can set the width and sidebearings of all selected letters at once. With the *Relative* option, the values will be added or subtracted.

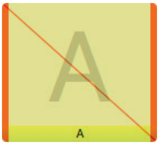
5 Font View

5.1 VIEWING GLYPHS

The Font view of the main window displays the glyphs in the font. Here, you can manage your glyph set. When mutiple tabs are open, you can activate the Font view by clicking on the first tab, or pressing Cmd-Opt-1.



With the slider in the bottom right corner of the window, you can control the zoom level of the displayed glyphs. Depending on the zoom level, Glyphs Mini will display either the glyph images only, or glyph images with glyph names, or, in large zoom stages, images, names and Unicode values, if available.

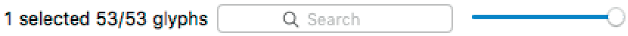


The glyph cell shows some extra information about the glyph. A vertical red stripe on the left or right edge of the cell indicates that the metric key for the respective sidebearing is out of sync (see 8.1.2, 'Metric Keys', p. 72). Likewise, a lock symbol indicates that the glyph has been locked and remains uneditable until it is unlocked again (see 2.1, 'Editing Glyphs', p. 9). A thin red diagonal line through the glyph cell indicates that the glyph in question is not exporting (see 5.3.4, 'Export', p. 48). Glyphs are shown in their glyph color (see 5.3.5, 'Color Label', p. 48). Glyphs with unsaved changes show a slightly darker glyph cell until saved.

5.1.1 Searching for Glyphs

You can choose to view different excerpts of your glyph set by selecting categories, languages or filters from the left sidebar. To further narrow down your search, you can enter a search term in the search field (Cmd-F) at the bottom right of the window. For more details, see section 5.5.1, 'Search Box' (p. 50)

The three numbers to the left of the search field indicate (from left to right) how many glyphs are selected, how many are currently displayed, and the total number of glyphs in the font file:



If you cannot see a glyph you are looking for, the current display settings may prevent the glyph from showing. If that is the case, first check if there is a selection in the left sidebar, or simply click on *All* at the top. Also, a search term may still be entered and active in the search field. Click in the search field (Cmd-F), and clear its search string.

5.2 MANAGING THE GLYPH SET

5.2.1 Generating New Glyphs

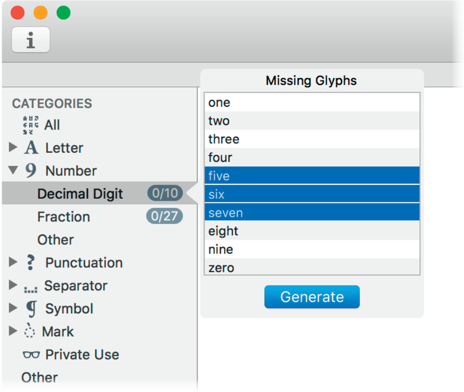
There are several ways to add new glyphs to a font:

- *Glyph* > *New Glyph* (Cmd-Opt-Shift-N) and the Plus button at the bottom in Font view add an empty glyph called 'newGlyph' to the font. You may need to scroll down to see it.
- *Glyph* > *Duplicate* (Cmd-D) duplicates the currently selected glyph(s). Because glyph names must be unique throughout a font, the new glyphs will carry an incremental number as name suffix, e.g., '.001', '.002', etc.
- Open *Window* > *Glyph Info,* then select one or more glyph descriptions, and click the *Add to Font* button.
- Some entries in the sidebar of the Font view have a number badge, indicating the number of glyphs already created versus the total number of glyphs predefined in that category, language group or filter. Right-click or Ctrl-click the badge to get a list of all glyphs missing in this category.

Adding glyphs through menu commands also works in Edit view. Newly added glyphs will then be added to the sample string in the current tab.

For the complete list of recognized glyph names, choose *Window* > *Glyphs Info*.

Select one or more glyph names, or press Cmd-A to select all of them, and click *Generate* to add them to your font:

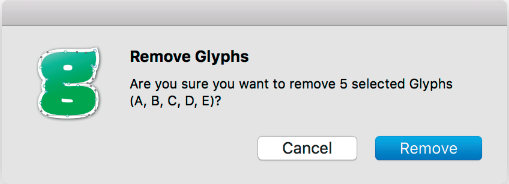


5.2.2 Copying Glyphs Between Files

You can copy any number of glyphs from one file (the source) and paste them into another file (the target). Components will be re-linked to their respective counterparts in the target font. If a glyph with the same name already exists in the target font, then the pasted glyphs will be renamed with an increasing triple-digit suffix, e.g., 'A.001', 'A.002' etc., avoiding data loss.

5.2.3 Removing Glyphs

In Font view, you can click the Minus button in the bottom section of the window. Before the respective glyphs are removed from your font, you will be asked for confirmation:

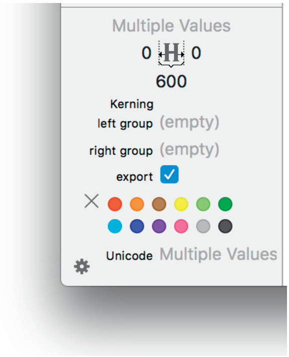


Remove Glyphs

Are you sure you want to remove 5 selected Glyphs (A, B, C, D, E)?

5.3 GLYPH PROPERTIES

Select one or more glyphs and, if necessary, click on the second button ⏏ at the bottom left of the window, and the glyph properties will be displayed.



Multiple Values

0 H 0

600

Kerning left group (empty)

right group (empty)

export ☑

Unicode Multiple Values

5.3.1 Name

The glyph name can be accessed and changed only if exactly one glyph is selected. If you want to batch-change the names of many selected glyphs, you can use *Edit* > *Find* > *Search and Replace* (Cmd-Shift-F). See section 5.6.3, ‘Renaming Glyphs’ (p. 55) for more details.

5.3.2 Width and Sidebearings

The widths as well as the left and right sidebearings can be accessed and changed for any number of selected glyphs at once. Changing only the width affects the right sidebearing.

5.3.3 Kerning Groups

You can set kerning groups for a range of glyphs, see section 8, ‘Spacing and Kerning’ (p. 72), for further details.

5.3.4 Export

The glyph will appear in the exported font only if this check box is enabled. This is useful for keeping glyph variations or glyph components from ending up in the final font. Newly created glyphs are set to export by default, unless their name starts with an underscore (‘_’).

5.3.5 Color Label

For easier sorting and filtering, or simply for keeping oversight, you can mark glyphs with one of twelve predefined colors. The colors have no effect on the exported font.

5.3.6 Unicode

The Unicode value is determined by the glyph name, which means that you cannot set the value directly. Thus, the displayed Unicode value of glyphs is read-only. Glyphs in the Private Use Area can always have custom names. For details, see section 5.5.2, 'Private Use Area' (p. 51).

5.4 BATCH-PROCESSING

5.4.1 Selecting Glyphs

Batch-processing a number of glyphs requires a selection of multiple glyphs. You can select all glyphs currently displayed in Font view via *Edit > Select All* (*Cmd-A)*.

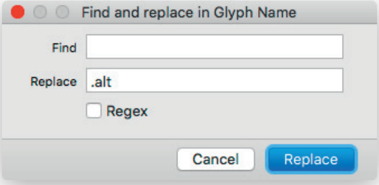
5.4.2 Menu Commands

You can apply manipulations on the selected glyphs. Select the glyphs you want to batch-edit, and choose one of these commands:

- *Glyph > Duplicate* (Cmd-D)
- *Glyph > Make Component Glyph* (Cmd-Opt-Shift-C)
- *Glyph > Decompose Components* (Cmd-Shift-D)
- *Glyph > Add Image* (see 2.10, 'Images', p. 34)
- *Glyph > Update Metrics* (Ctrl-Cmd-M)
- *Glyph > Set Anchors* (Cmd-U)
- *Glyph > Selection to Background* (Cmd-K), *Edit Background, Assign Background*
- *Glyph > Correct Path Direction* (Cmd-Shift-R)
- *Glyph > Reverse Contours* (Cmd-Opt-Shift-R)
- *Glyph > Round Coordinates*
- *Glyph > Tidy up Paths* (Cmd-Opt-Shift-T)
- *Glyph > Add Extremes*

5.4.3 Batch-Renaming Glyphs

You can search and replace in names of selected glyphs with *Edit > Find > Search and Replace* (Cmd-Shift-F). It will look for the string entered in the *Find* field and replace it with the string entered in *Replace*. You can add characters to the end of the names of selected glyphs if you leave the *Find* field empty and enter the desired suffix in the *Replace* field:


Find and replace in Glyph Name

Find

Replace .alt

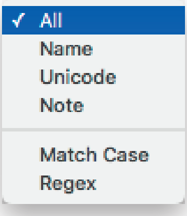
□ Regex

5.4.4 Palette Manipulations

From the Palette (Cmd-Shift-P), you can apply the mirroring, scaling, rotating, and skewing functions in the Transformations section on multiple selected glyphs. The transformation origin will be respected. Also, the boolean operations work with a multiple-glyph selection. Be careful though, it is easy to achieve an unexpected result.

5.5 FILTERING AND SORTING

5.5.1 Search Box



The search box (Cmd-F) at the bottom right provides a way to instantly narrow down the selection of glyphs displayed in the Font tab. By clicking on the search symbol (the magnifying glass) in the entry field, you can choose the following options:

- *All:* search in glyph names, glyph notes, and Unicode values
- *Name:* search in names only
- *Unicode:* search in Unicode values only
- *Note:* search in glyph notes only
- *Match Case:* if active, searches will be case-sensitive
- *Regex:* if active, the search string will be interpreted as a regular expression

The *Regex* option not only activates regular expressions, but also automatically limits the search to glyph names. Regular expressions allow you to search for a text pattern rather than a specific text. When activated, you can use search terms like [Lldt]caron(\..+)* to match Lcaron, lcaron, dcaron, tcaron, with or without dot suffixes. Some examples:

- [abc] *any of the letters a, b or c*
- [^abc] *any letter but a, b or c: d, e, f, g…*
- . *any character: a, b, c, ., -, 1, 2, 3…*
- \d *any digit: 0, 1, 2, 3, 4, 5, 6, 7, 8, or 9*

- `\.` *a literal period: .*
- `a?` *zero or one occurrence of a*
- `a+` *one or more occurrence of a: a, aa, aaa, aaaa …*
- `a*` *zero, one or more occurrences of a*
- `a{3}` *exactly three occurrences of a: aaa*
- `a{2,5}` *between 2 and 5 occurrences of a: aa, aaa, aaaa, aaaa*
- `a.*` *any glyph name with a at the beginning: a, ae, aacute …*
- `.*a` *any glyph name with a at the end: a, cedilla, ccedilla …*
- `.+-.+` *any glyph name with a hyphen in it: beh-ar, he-hb …*
- `.+\..+` *any glyph name with a period in it: a.ss01, J.titl …*
- `.+\d` *any glyph name with a digit at the end: a.001, b.x2, c.ss02*
- `[^\d]*\d{3}` *any glyph name with exactly three digits at the end, and no digits anywhere before: a.001, x.test123 …*

5.5.2 Private Use Area

In Unicode, a Private Use Area (PUA) is a range of code points that, by definition, no characters will be assigned to by the Unicode Consortium. They are intentionally left undefined so that you may define your own characters without conflicting with Unicode Consortium assignments.

Glyphs in the Private Use Area can always have custom names. You first create them with a name according to the uniXXXX (for values between U+0000 and U+FFFF) or uXXXXX scheme (for values beyond U+10000) to set their Unicode value, and then rename them to your liking. The usual glyph naming restrictions still apply, though (see 5.6.2, 'Naming Glyphs', p. 54). Make sure the custom name you choose is not already reserved by another glyph in the internal glyph database. Otherwise, the Unicode value of that specific entry will be applied.

5.5.3 Categories

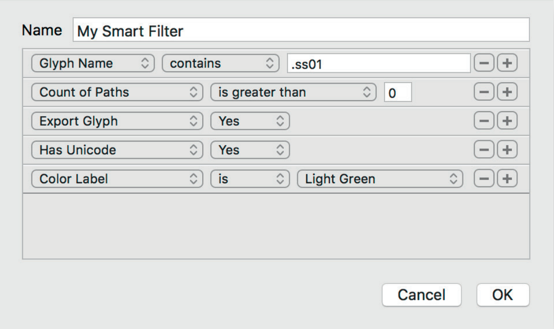
Categories give you the option to only display a predefined subset of the glyphs you have. Click the triangle next to the category name to access subcategories. You can combine more than one category by Cmd-clicking their names to display the intersection of all selected categories. You can achieve an even narrower subset by also Cmd-clicking a Filter. For instance, if you click the *Uppercase* category and then Cmd-click a Filter, Glyphs Mini will display all uppercase characters in the selected Filter.

You can cancel any subset you created and return to displaying the complete glyph set by clicking on the *All* category.

5.5.4 Languages

To only display glyphs belonging to a certain script, you can click on the name of the script in the Languages, *Latin, Cyrillic,* or *Greek*. Clicking on the triangle next to the script name will expand the view and give you more detailed subsetting options. Again, you can combine subsets by subsequently Cmd-clicking them.

5.5.5 Smart Filters



Name: My Smart Filter

Glyph Name | contains | .ss01

Count of Paths | is greater than | 0

Export Glyph | Yes

Has Unicode | Yes

Color Label | is | Light Green

To define a query-based filter, click the gear button in the bottom left of the window and choose *Add Smart Filter*. Smart filters are similar to smart playlists in iTunes. Set up the query with any of these properties:

- *Color Label:* the color assigned to the glyph, if any.
- *Count of Paths:* number of paths in the glyph.
- *Export Glyph:* whether the glyph exports or not.
- *Glyph Name:* the name of the glyph.
- *Contains Component:* whether the glyph contains a component with a specific name.
- *Has Hints:* whether manually placed hints are in the glyph.
- *Has Metrics Keys:* whether any of the glyph's metrics contains a formula such as '=n+10'.
- *Has Unicode:* whether the glyph has a Unicode value.
- *Is Auto-Aligned:* whether the glyph is automatically aligned.
- *Metrics Keys Out of Sync:* whether the metrics keys for the glyph are up to date or not.

Add additional properties by clicking on the plus button at the end of a line. By default, all properties are applied to the query (logical-and). You can add and nest logical-and (‘all of the following’), logical-or (‘any of the following’) and even logical-not (‘none of the following’) concatenations by holding down the Option key while pressing the plus button.

5.5.6 List Filters

You can define a list of glyphs to be displayed by choosing *Add List Filter* from the gear menu in the bottom left of the Font tab. By default, currently selected glyphs are added to the list automatically. You can subsequently edit the list or paste a list of glyph names. In Font view, once you apply a list filter, you will see its glyphs in the order they are listed in.

5.5.7 Manage Filters

Once you have set up a couple of filters, you can apply more than one filter by holding down the Command key and subsequently clicking the individual Filter names. Edit an existing filter by selecting it in the Font tab sidebar and subsequently choosing *Edit Filter* from the gear menu in the bottom left. Alternatively, double click its side bar entry. A dialog sheet will appear that allows you to edit the selected smart or list filter. Remove an existing filter by selecting its sidebar entry and choosing *Remove Filter* from the same gear menu. You can sort your filters by dragging them into a desired order. You can also organise your filters in sidebar folders. To add a folder, choose *Add Folder* from the gear menu. To quickly change the name of a filter or folder, click to select it, press the Return key, and type a new name.

5.6 NAMES AND UNICODE

5.6.1 Glyph Info Database

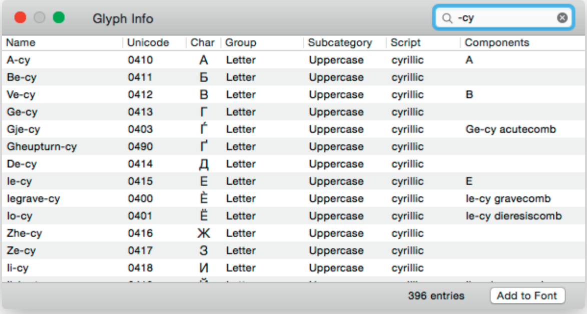
Glyphs Mini contains a glyph info database. For each glyph, the database defines:

- the human-readable (or ‘nice’) glyph name,
- the production names, i.e., the name as required for the exported installable font file,
- the associated Unicode value, if any,
- possible components,
- default anchors, if any,

- possible accents, mainly for displaying the mark cloud when you click on an anchor,
- the script such as Latin, Cyrillic or Greek,
- the category and subcategory, such as letter and uppercase, punctuation, or figure, etc.

For more information about the Adobe Glyph List, see github.com/adobe-type-tools/agl-aglfn

For the 'nice names', Glyphs Mini employs a special naming convention which is, for the most part, loosely based on the Adobe Glyph List (AGL) Specification. But in the AGL, not all glyphs have nice names. Rather, they have names like 'uni042F', which are hard to memorize. In cases like this, Glyphs Mini derives the nice name from the Unicode description and appends a script suffix, e.g., 'CYRILLIC CAPITAL LETTER A' becomes 'A-cy'. For an overview of the available data stored in the glyph info database, choose *Window* > *Glyph Info:*



Name	Unicode	Char	Group	Subcategory	Script	Components
A-cy	0410	А	Letter	Uppercase	cyrillic	A
Be-cy	0411	Б	Letter	Uppercase	cyrillic	
Ve-cy	0412	В	Letter	Uppercase	cyrillic	B
Ge-cy	0413	Г	Letter	Uppercase	cyrillic	
Gje-cy	0403	Ѓ	Letter	Uppercase	cyrillic	Ge-cy acutecomb
Gheupturn-cy	0490	Ґ	Letter	Uppercase	cyrillic	
De-cy	0414	Д	Letter	Uppercase	cyrillic	
Ie-cy	0415	Е	Letter	Uppercase	cyrillic	E
Iegrave-cy	0400	Ѐ	Letter	Uppercase	cyrillic	Ie-cy gravecomb
Io-cy	0401	Ё	Letter	Uppercase	cyrillic	Ie-cy dieresiscomb
Zhe-cy	0416	Ж	Letter	Uppercase	cyrillic	
Ze-cy	0417	З	Letter	Uppercase	cyrillic	
Ii-cy	0418	И	Letter	Uppercase	cyrillic	

Typically, suffixed glyphs will inherit their glyph info attributes from their unsuffixed counterpart. That means that, e.g., both 'X' and 'X.alt.bold' will be treated as letters, thus enable automatic alignment and receive default anchors when *Glyph* > *Add Anchors* is run. If you name the copy 'Xalternate' without the dot, it will not be recognized as a letter unless you changed the glyph data accordingly. So, if you want to handle glyph backups and copies in a way that Glyphs can still semantically connect them with their originals, make sure the glyph names are the same up until the first period in the name.

5.6.2 Naming Glyphs

The glyph name appears below the glyph in grid view, or in a separate column in List view. You can edit a glyph name by clicking once into the name. Glyphs Mini will automatically convert entries like ‘ä’ or ‘uni00E4’ to nice names, i.e., ‘adieresis’ in this case. Based on the glyph names, Glyphs Mini will automatically assign Unicode values as well as Category, Subcategory and Script attributes, and can even automatically build some OpenType features. When you name or rename a glyph, three different types of input values are accepted:

- the ‘nice name’, e.g., ‘Ia-cy’,
- the Unicode value in hex form with a ‘uni’ or ‘u’ prefix, e.g., ‘uni042F’ or ‘u10120’ (‘uni’ for four-digit hexadecimal codes, ‘u’ for five- or six-digit codes),
- the actual typed character, e.g., ‘ä’, if the character is not expected in a valid glyph name.

Exception: names of non-exporting glyphs can also start with an underscore.

Except for ‘.notdef’, a valid glyph name must begin with a character from A-Z or a-z. A glyph name can subsequently also contain figures (0-9), underscores (_), periods (.) or hyphens (-). All other characters, including whitespace characters, are not allowed. A warning will appear if you try to use invalid characters like space in a glyph name.

If you need variations of glyphs, extend your glyph names with dot suffixes, like ‘n.sc’ for a small cap n, or ‘five.numr’ for a numerator five. Some suffixes will be used by Glyphs Mini to automatically build OpenType feature code. See the appendix of this manual for a list of recognized suffixes. To name ligatures, the names of the parts need to be joined with an underscore, e.g., ‘f_f_l’ for an ffl ligature. Hyphens are used to indicate the script a glyph belongs to, e.g., ‘em-cy’ for an Cyrillic em. In ligatures, the script suffix is appended only once, e.g., em_em-cy.

You can copy the names of selected glyphs into the clipboard by opening the submenu *Copy Glyph Names* in the context menu.

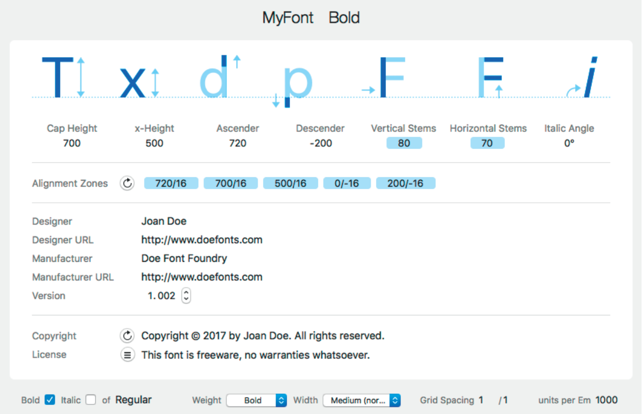
5.6.3 Renaming Glyphs

You can rename any glyph by simply clicking in its name field and editing the name as desired. Keep in mind that if you change the name before the first period, its glyph info attribution will be changed as well. This means that Unicode value, category, subcategory, script, default anchors and accents, as well as default decomposition may change.

For searching and replacing in glyph names, or batch-renaming a large number of glyphs at once, see section 5.4.3, ‘Batch-Renaming Glyphs’ (p. 49).

6 Font Info

Open the Font Info tab by choosing *File* > *Font Info* (Cmd-I), or by clicking on the Info button in the top left corner of the main window:



MyFont Bold

Cap Height	x-Height	Ascender	Descender	Vertical Stems	Horizontal Stems	Italic Angle
700	500	720	-200	80	70	0°

Alignment Zones 720/16 700/16 500/16 0/-16 200/-16

Designer	Joan Doe
Designer URL	http://www.doefonts.com
Manufacturer	Doe Font Foundry
Manufacturer URL	http://www.doefonts.com
Version	1.002
Copyright	Copyright © 2017 by Joan Doe. All rights reserved.
License	This font is freeware, no warranties whatsoever.

Bold ☑ Italic □ of Regular Weight Bold Width Medium (nor... Grid Spacing 1 /1 units per Em 1000

6.1 FAMILY AND STYLE NAMES

The family name, displayed in large letters at the left, is how the font family will appear in the font menu. Fonts with the same family name will be grouped in the same submenu. You can use uppercase, lowercase, and spaces in the font name, but non-ASCII characters prevent the font from exporting.

The style name (or subfamily name) of the font, displayed on the right, differentiates it from other members of the same font family, e.g., 'Regular' or 'Bold'.

6.2 METRICS

Setting the metrics sets the guiding lines in Edit view, and prepares the font for hinting.

The *Cap Height* is the distance from the baseline to the height of the caps, not counting the overshoots. Typically, the height of the uppercase H is a good reference.

The *x-Height* is the distance from the baseline to the height of the lowercase letters without ascenders, not counting the overshoots. Typically, the height of the lowercase x is a good reference, hence the name.

The *Ascender* is the distance from the baseline to the height of the lowercase letters with ascenders, not counting the overshoots. Lowercase letters with ascenders include b, d, f, h, k, and l. The lowercase t usually does not rise to the ascender, but stops halfway between x-height and ascender in most designs.

The *Descender* is the distance from the baseline to the depth of the lowercase letters with descenders, not counting the overshoots, and expressed as a negative number since it reaches below the baseline. Lowercase letters with descenders include g, j, p, q, and y. Also, cursive designs of the lowercase f often reach the descender line.

For a description of *Vertical* and *Horizontal Stems*, see below, section 6.3, 'Stems and Alignment Zones' (p. 57).

For more info about OpenType tables such as post and hhea, see www.microsoft.com/typography/otspec/otff.htm

The *Italic Angle* will slant the help lines in Edit view around half the x-height. All sidebearing and kerning measurements respect the italic angle. In the exported font, this value will determine the italicAngle property in the post table as well as caretOffset and caretSlopeRun in the hhea table.

6.3 STEMS AND ALIGNMENT ZONES

If you enter good average values for your standard stems, i.e., *Horizontal Stems* and *Vertical Stems,* the autohinter will find those stems in your letters and put a stem hint on them. *Alignment Zones* help create an even vertical alignment at low resolutions through overshoot suppression. The values entered here are crucial for the autohinting process when the font is being exported. An alignment zone must encompass anything that should later be aligned at a low resolution. For a detailed discussion, see section 9.2, 'Font-Wide Hints' (p. 77).

After you have set the Metrics properly, you can click the grey circle to let Glyphs Mini find the alignment zones for you. Glyphs Mini will then reduplicate the heights of the vertical metrics you entered in the Metrics field as the positions of the zones. It will try to guess the size of the alignment zones by measuring certain key glyphs in the font:

- for the ascender zone, it will take the height of the lowercase f plus one unit,
- for the cap zone, the height of the uppercase O plus one unit,
- for the small cap zone, the height of 'o.sc' plus one unit,
- for the x-height zone, it will measure the lowercase o, and add one unit,
- the baseline zone defaults to –15,
- and for the descender zone, Glyphs Mini takes the depth of the lowercase g plus one unit.

If Glyphs Mini cannot find any glyphs to measure, the sizes for top zones default to 16, and to −16 for bottom zones. It is a good idea to double check if all your important glyphs reach into, but not beyond, the alignment zones.

6.4 DESIGNER AND DESIGNER URL

Here, you can enter your name and a URL (including the protocol, e.g., http:// or ftp://), for example:

- *Designer:* Jessica Doe
- *Designer URL:* http://www.example.com/

You can check if the URL was entered correctly by clicking on the arrow next to the text entry field. Glyphs Mini will open the URL in your web browser.

6.5 MANUFACTURER AND MANUFACTURER URL

Here, you can enter name and URL (including the protocol, e.g., http:// or ftp://) of your font vendor, for example:

- *Manufacturer:* Sample Font Store
- *Manufacturer URL:* http://www.example.com/

You can check if the URL was entered correctly by clicking on the arrow next to the text entry field. The URL will then be opened in your web browser.

6.6 VERSION

Glyphs Mini derives the Version String from the *Version* entry. In the resulting font file, built-in production tools like makeotf also leave their traces in the version string, e.g., ‘Version 1.010;PS 001.010;hotconv 1.0.70;makeotf.lib2.5.58329’.

6.7 COPYRIGHT AND LICENSE

The *Copyright* is a simple copyright notice. Click the update button (the circled arrow) next to it to have Glyphs Mini fill it in automatically, based on the entry in the *Designer* field. According to the specification, the *License* is supposed to be a ‘description of how the font may be legally used, or different example scenarios for licensed use. This field should be written in plain language, not legalese.’ However, some foundries paste the complete text of their license agreement in here.

6.8 STYLE LINKING

You can style-link the font to another font with the same family name by choosing *Bold* and/or *Italic* and typing the style name of the referenced font. This enables the Bold and Italic functions of word processing software such as TextEdit, Pages and Microsoft Word. Typical style relations are as follows:

- *Thin, Light, Regular, Medium, Semibold, Extrabold, Heavy:* no style linking, i.e., leave all options blank,
- *Bold:* Bold of Regular,
- *Italic:* Italic of Regular,
- *Bold Italic:* Bold and Italic of Regular,
- *Other italics:* Italic of their respective upright style, e.g., Medium Italic is the Italic of Medium.

6.9 WEIGHT AND WIDTH

The *Weight* and *Width* options are stored as weight class and width class in the exported font, and influence the menu order of fonts in Adobe applications. More condensed fonts come before more extended fonts, and secondly, lighter fonts appear before bolder fonts in the font family submenus. If both weight and width properties are the same, the fonts are sorted alphabetically. This only applies to Adobe menus; Windows applications do not have font family submenus, and Apple submenu sorting is based on undisclosed heuristics and therefore ignores the values entered here.

6.10 GRID SPACING AND SUBDIVISION

The *Grid Spacing* entry consists of two values: the *Grid Step* before the slash, and the *Subdivision* after the slash.

The *Grid Step* value defines how coordinates get rounded. Standard is the value 1. When set to zero, a grid spacing of 1/100 is assumed. This is useful if you want to keep very fine details, e.g., after applying the *Hatch Outlines* filter or in detailed dingbat fonts, or if you plan to scale your glyphs and want to minimize rounding errors. Higher values are helpful when creating a modular design or a pixel font. The Pixel tool (T) uses the Grid Step value to determine the grid it applies for posiztioning the pixel components. For more details, see section 2.2.3, ‘Pixel Tool’ (p. 10).

Contrary to popular belief, decimal coordinates can be exported into CFFs.

The *Subdivision* value gives you the option to subdivide the *Grid Step* if you need finer steps but want the larger grid spacing as design orientation or as pixel grid for the Pixel tool. The value indicates into how many compartments the main grid is subdivided, e.g., a *Grid Step* value of 100 and a *Subdivision* value of 5 will yield a subgrid with a 20 units step. Setting *Grid Step* to 1 and *Subdivision* to 10 will give all your point coordinates one decimal.

All tools and all modifications will consequently snap to the grid determined by *Grid Step* divided by *Subdivision*. Regardless of the grid settings, you can round all point coordinates of selected nodes or glyphs to integer numbers by choosing *Glyph* > *Round Coordinates*.

6.11 UNITS PER EM

Number of units per em (UPM), 1000 is the default. One em is equivalent to the font size. You can use the value to adjust the apparent size of your font relative to others. Choose a smaller UPM to make your font appear larger at the same numerical font size, and a larger UPM to make it appear smaller.

Increasing the UPM value can improve the representation of subtle details. The OpenType specification allows values between 16 and 16,384, but values greater than 5000 can lead to problems in InDesign and Illustrator. Problems have been reported for other applications starting at 3000 UPM. Some applications expect a UPM value of 1000. Also, point coordinate values must not exceed ± 32,768, and glyph widths can be problematic beyond ± 4096. It is therefore recommended to leave it 1000, or at least not deviate far from it.

7 Reusing Shapes

7.1 COMPONENTS

Displayed as a grey glyph inside another glyph, components are an efficient way to re-use the shapes of glyphs inside multiple other glyphs. The main purpose of components is their use inside diacritic variations of the base letter, e.g., A in Ä, Ă, Ā, Á, À, Â, and Å.

The original, usually outlined, glyph that the component points back to, is referred to as the 'base glyph' of that component. A glyph built with components is usually referred to as 'compound' or 'composite glyph'. Since the component always stays in sync with its base glyph, you only need to change the original outline, and the changes will be visible in all composites. Components are converted into plain outlines before exported in OpenType/CFF fonts, since CFF does not properly support components.

7.1.1 Building Compounds

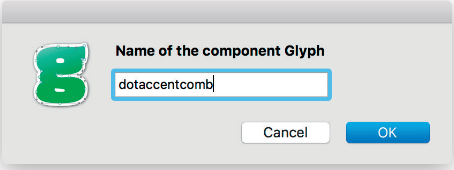
You can build compound glyphs by either creating a glyph with the proper name, e.g., 'aacute', or by selecting existing glyphs and choosing *Glyph* > *Make Component Glyph* (Cmd-Opt-Shift-C) to rebuild the glyph as a composite with its default recipe. Glyphs Mini will use a built-in database of glyph compositions to determine the components of the glyph in question. You can add components yourself to an existing glyph by choosing *Glyph* > *Add Component* (Cmd-Shift-C) and picking the base glyph in the subsequent dialog.

While multiple glyphs are selected, you can add the same component to all those glyphs at once, using the same menu command.

If you already have a letter that you want to turn into a component-based glyph, you can force the composition with *Glyph* > *Make Component Glyph*. The content of the letter will be deleted and rebuilt from scratch, then. This only works if the base glyphs of the individual components already exist in the font.

7.1.2 Turning Paths into Components

If the base glyphs are not already present in the font, you can create a new component from an existing path by selecting the path you want to turn into a component and picking *Component from Selection* from the context menu. Glyphs Mini will then suggest a name for the base glyph based on the decomposition information from the built-in glyph database:


Name of the component Glyph

dotaccentcomb

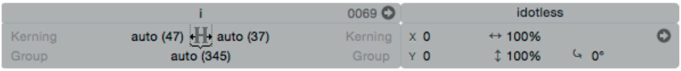
Once you confirm the dialog, Glyphs Mini will create a new glyph containing the selected path and place it as a component in the original glyph.

7.1.3 Editing Components

Select a component by clicking on it. Press the Tab key to select the next, or Shift-Tab to select the previous component. Rectangular mouse selections ignore components, unless you simultaneously hold down the Option key.

The grey Info box will display component information: the name of the original glyph the component points to, its position, scale, and rotation. Open the original glyph by either double clicking the component or clicking on the arrow next to the glyph name. Change the component by clicking on its name. Glyphs Mini will prompt you for the new glyph the component should point to.

Tip: Edit > Select All (Cmd-A) selects all paths, but ignores anchors and components. Selecting all twice in a row (hold down Cmd while you type A twice) also selects anchors and components.


i 0069

Kerning auto (47) auto (37) Kerning

Group auto (345) Group

idotless

X 0 ↔ 100%

Y 0 ↕ 100% 0°

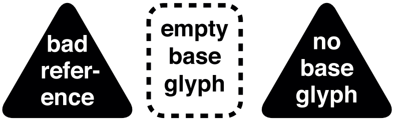
You can edit component attributes by changing the values in the grey Info box, or by applying transformations from the Transformations Palette, such as aligning, scaling, rotating, and mirroring. Flipping mark components both horizontally and vertically will switch their top and bottom anchor association. I.e., a flipped top mark will attach to a bottom anchor and vice versa. This can, for example, be useful for reusing the bottom mark commaccentcomb in gcommaaccent (i.e., instead of using the default commaabovecomb), where the comma accent is supposed to reside on top of the g.

You can move a selected component with the mouse or the arrow keys if it is not automatically aligned. Hold down the Shift key for increments of 10, and the Command key for increments of 100. Option-drag a component to duplicate it. Delete it by simply pressing the Delete or Backspace key.

7.1.4 Moving between Base Glyphs and Compounds

To edit the base glyph of a component, double click the component and Glyphs Mini will place the original letter next to the compound glyph, and activate it for immediate editing. Alternatively, you can select the component inside the composite glyph and click the arrow symbol that appears in the gray Info box (Cmd-Shift-I).

Placeholders can indicate an 'empty base glyph', 'no base glyph', or a 'bad reference'. An empty base glyph is a base glyph without paths or components. Double clicking its placeholder will open the empty glyph next to the current glyph, so you can edit it right away. 'No base glyph' means that the glyph the component is referring to does not exist in the font. Double click the placeholder to create the glyph, and insert it in the Edit tab, next to the current glyph. A 'bad reference' indicates a broken component reference, e.g., a circular reference, i.e., a component that points to the compound glyph in which it resides.



You can see in which compounds a glyph is used as a component by right-clicking or Ctrl-clicking on the glyph in Edit view to open its context menu, and choosing *Show all glyphs that use this glyph as a component*. All glyphs that contain a component pointing back to the base letter will be inserted next to it in the Edit tab. This command is also available in the context menu of placed components. All other compounds containing the same component will be inserted next to the current glyph.

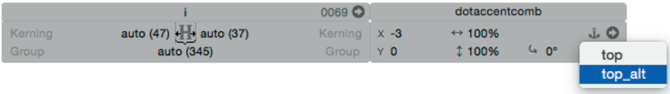
7.1.5 Anchors

In mark compounds, accents are snapped into place if corresponding anchors are placed in both the base letter and the original mark. The letter needs an anchor (e.g., 'top'), and the mark needs a corresponding mark anchor, i.e., an anchor with the same name, but preceded by a leading underscore (e.g., '_top'). In the composite, the mark will then be positioned according to the relative anchor positions.

For many glyphs, default anchors are predefined in the built-in glyph database. They can be added to selected glyphs with *Glyph* > *Set Anchors* (Cmd-U). Setting anchors also works for multiple glyphs. You can add custom anchors by choosing Add Anchor from the context menu. You will immediately be prompted for a name. Enter an ASCII-only name without whitespace and confirm by pressing Return. You can change the name of an existing anchor either by selecting it and pressing Return, or by double clicking it. Then, you can start typing the new name:

Select an anchor in the base glyph to get a preview of the most common accents that may attach to this glyph. Similarly, select an anchor in the mark glyph and this accent is shown on all other glyphs in the same Edit view. If exactly one anchor is selected, press the Tab key to select the next anchor, or Shift-Tab to go to the previous anchor. Change the position of an anchor by dragging it with the mouse, or selecting it and pressing the arrow keys. Add Shift or Command for increments of 10 or 100 units, respectively. Delete selected anchors by pressing the Delete key. Select an anchor and two points and choose *Glyph* > *Align Selection* (Cmd-Shift-A) to horizontally center the anchor between the points. Again, Glyphs Mini will respect the italic angle set in *File* > *Font Info* (Cmd-I).

Sometimes you need more than one 'top' anchor, e.g., in ligatures, or for fine-tuning positions of certain marks, or in circumflexcomb for different positions where marks can attach in Vietnamese double accents. Multiple anchors of the same kind need to be underscore-suffixed, e.g., 'top_1' and 'top_2', or something more descriptive such as 'top_viet' or 'top_acute'. If you apply this naming scheme, you can choose a different anchor for a component by selecting it, clicking on the anchor symbol in the Info Box, and choosing an alternative anchor from the pop-up menu. The anchor symbol becomes visible if there is more than one anchor that fits the currently selected component:



i 0069

Kerning auto (47) auto (37)

Group auto (345)

dotaccentcomb

Kerning X -3 ↔ 100%

Group Y 0 ↕ 100% 0°

7.1.6 Automatic Alignment

Letters built entirely from components, so-called compound or composite glyphs, can make use of automatic alignment. This means that both the placement of the components inside the compound, and the width of the compound are determined by the base glyphs and their anchors.

Single letters (‘component copies’): If the base glyph is of the category ‘Letter’, the resulting compound will be automatically aligned with the base letter, inheriting its positioning and side bearings.

Letters with accents (‘mark compounds’): If the first component points to a ‘Letter’ glyph, and the following components point to ‘Mark’ glyphs, the resulting compound will be automatically aligned with the base letter, and the marks will be automatically placed according to corresponding marks in the original glyphs.

Multiple letters (‘compound ligatures’), with or without marks: The automatic placement of the letters takes both the glyph widths and their kerning into account. For example, when building a onehalf fraction out of one.numr, fraction, and two.dnom, the fraction will look as if its parts were typed individually. To move them to a different position, you can either space and kern the individual glyphs, or you can disable the automatic alignment of a component via the context menu (see further below).

Non-letters are not automatically aligned. This, for instance, allows to re-use proportional figures for tabular figures or vice versa. Also, automatic alignment of components is disabled as soon as there are any paths in the glyph. Once those paths are removed again, the components will snap back into automatic alignment. To explicitly disable automatic alignment of a component, Ctrl- or right-click on a component, and choose *Disable Automatic Alignment* from its context menu. You can disable automatic alignment for the base, but keep it for the subsequent marks. Thus, you can move the compound and change its width, while the marks still stay aligned to the base.

7.1.7 Locking Components

For components that cannot or should not be automatically aligned, there is the option to lock them in their current position. To do that, select a component, and pick *Lock Component* from its context menu. Locked components cannot be selected, preventing accidental shifts. To unlock a component again, right click or Ctrl-click it to bring up its context menu, and choose *Unlock Component*.

7.1.8 Component Colors

Placed components indicate their status with their color. An auto-aligned component shows a subtle green, a locked component is displayed in a subdued blue, while a freely placed, unlocked and unaligned component sports a simple grey:

From left to right: auto-aligned, locked, and unaligned components.

7.1.9 Decomposing

You can turn all components inside a glyph into editable paths by choosing *Glyph > Decompose Components* (Cmd-Shift-D). All components in the current glyph will be removed, and its paths and anchors will be inserted instead. Nested components will be decomposed as well. Selecting *Decompose* from the context menu of a specific component only decomposes the selected component. In this case, nested components will not be decomposed.

7.1.10 Combining Paths and Components

As soon as there is a path in the glyph, automatic alignment is disabled. Therefore, be careful when combining components and paths, because shifts may occur, especially if the base glyph of the component is changed. If you want to prevent shifts, consider rebuilding the glyph with components and anchors, thus enabling automatic alignment.

While dragging an object over a placed component, vector points of the component become highlighted. Nodes dragged above a component snap to the compound nodes. You can align a component with an outline node by selecting both and choosing *Glyph > Align Selection*. While the outline node stays put, the component will be moved above the point so that the origin of the component will be aligned with the node. The origin is the point in the component glyph where the left side bearing intersects with the baseline, or, if present, the position of an anchor named 'origin'. This can be useful for placing serif components.

7.1.11 Nesting Components

Glyphs Mini allows you to nest components. For example, you can build the dieresiscomb glyph out of two dotaccentcomb components. Subsequently, you can use this compound dieresiscomb in higher-level compounds such as adieresis (ä).

Anchors will shine through in nested components, unless they are overridden by a specific anchor placement in a higher compound in the nesting chain. That means that you do not need to reset anchors in nesting components. E.g., if you are building oslashacute from oslash and acutecomb, and oslash is composed of o and slashlongcomb, then the acutecomb can reuse the 'top' anchor of the o. You do not need to set another 'top' anchor in oslash, unless you do want a different mark placement.

7.1.12 Preferred Marks for Glyph Composition

When automatically building compounds, Glyphs Mini will prefer marks that carry the same name suffix as the compound. For instance, when composing adieresis.sc, Glyphs Mini will prefer dieresiscomb.sc to dieresiscomb, if it is available in the font.

For building compound uppercase letters, marks with a special '.case' suffix are preferred. E.g., Glyphs Mini will prefer dieresiscomb.case over dieresiscomb for Odieresis (Ö).

Tip: You can see the default compositions of compounds in the Components column of Window > Glyph Info.

When building combinations for i and j, make sure both idotless and jdotless are in your font. This is because the dot is typically not included when i and j receive accent marks. For i and j diacritics, Glyphs Mini will prefer marks carrying a '.i' or '.narrow' suffix.

7.1.13 Underscore Components

If you are adding glyphs to your font that are not supposed to appear in the compiled font, but are solely used as parts in other letters, then it is advisable to prefix their names with an underscore character, e.g., _leftBottomSerif. This simplifies finding such glyphs through, for instance, a Smart Filter (see 5.5.5, 'Smart Filters', p. 52), or the Search function (see 5.5.1, 'Search Box', p. 50). When glyphs are generated with the *Component from Selection* function with a preceding underscore in their name, they will be created with the *Export* option off.

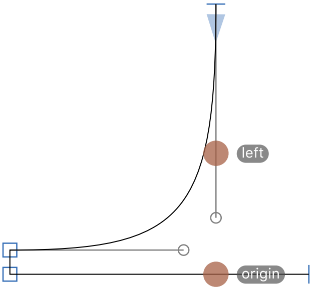
7.2 CORNER AND CAP COMPONENTS

7.2.1 Corner Components

Corner components are open path fragments that can be dynamically fitted into an outline. The main purpose for corner components is to facilitate the construction and management of serifs.

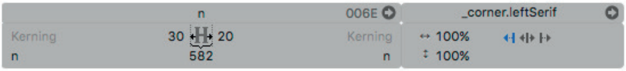
In order to set up a corner component, you need to create a glyph with a name that starts with '_corner', followed by an arbitrary dot suffix, e.g., _corner.leftSerif. Inside the glyph, you draw an outline around the origin point, or, alternatively, around an anchor named 'origin'. Optional 'left' and 'right' anchors control the amount of distortion for adjusting the path fragment to fit into the receiving path:

Tip: employing both left and right anchors in the corner component allows building of cupped serifs..



The path direction, as indicated by the arrowhead at the open start point, should match the orientation of the intended target paths. That means that if you are constructing a lower left serif, your serif path should start at the top point, bend around the origin, and end in the lower right point. To change the path direction, select the path, and choose *Reverse Selected Contours* from the context menu.

Once you are done with the corner component glyph, you can then proceed to insert the corner component into the outline of another glyph. To achieve this, select exactly one corner node in the outline of the glyph, and choose *Add Corner* from its context menu. The path fragment will immediately be bent into the corner, and also adapted to the slant of its surrounding outline segments.



n 006E _corner.leftSerif
Kerning 30 20 Kerning ↔ 100%
n 582 n ↕ 100%

You can press the Delete or Backspace key to remove it, you can copy the corner component into your clipboard and paste it on another node, or you can change its settings in the gray Info box. Via the Info box, you can switch to the original corner component glyph by clicking on the little arrow symbol. You can change which corner component is used by clicking on the name, as well as the scale, and the alignment of the corner component. The alignment controls which side of the corner component stays put while the other side gets bent into the outline of the target path. You can have a corner component:

- ◂| left-aligned, i.e., aligned with the stroke the entrance point is put upon, and the exit stroke stays put, which is typically what you want for a lower left or top right serif;
- |▸ right-aligned, i.e., aligned with the stroke the exit point is put upon, end the entry stroke stays put, which is usually what you want for top left or bottom right serifs;
- ◂|▸ center-aligned, i.e., rotated between both alignments, so neither entry nor exit stroke stays put, which makes sense for ink traps and the like.

Generally, the presence of ‘left’, ‘right’ and ‘origin’ anchors, the open path direction, and the alignment setting determine the rotation and positioning of the corner component on the target path. For simple serifs, you can influence the adjustment and positioning of the curve in slanted stems, for instance, when you need to control how far a serif can stick out horizontally at the end of a slanted stem. To do that, add an anchor called ‘left’ or ‘right’ precisely above the origin. Pick ‘left’ for a left-aligned corner component, ‘right’ for a right-aligned one, or both for a center-aligned corner component or cupped serif. The distance between the origin and the anchor controls the amount of adjustment: Moving it up will shorten the corner components on an acute position, and elongate it on a blunt one. Moving it down will achieve the opposite. Cornered slab serifs (as opposed to bracketed serifs) will need these anchors exactly on the joint between vertical and horizontal segments.

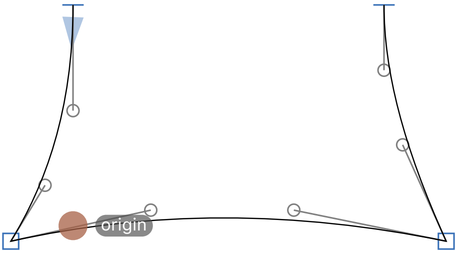
Tip: the quickest way to mirror a corner component is to select it and press the mirror buttons in the Transformations palette.

If your design permits it, you can mirror corner components and reuse them on the opposite side. I.e., you can reuse a left serif on a right corner if you set its horizontal scale to −100%.

To decompose a path with corner components into its calculated state, you can remove overlaps by choosing *Glyph > Remove Overlap* (Shift-Cmd-O).

7.2.2 Cap Components

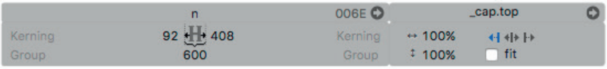
Cap components work very much like corner components except that they connect to two adjacent nodes rather than a single one. The idea behind cap components is that they can be used for spurs, finials, or head serifs like on the top of a lowercase u.



In order to create a cap component, generate a glyph with a name that starts with an underscore, followed by 'cap' and an arbitrary dot suffix, e.g., _cap.headSerif. Draw an open path around the origin point, with the connecting ends coming out on or above the baseline, i.e., a head serif would need to be drawn flipped (turned 180°) and on the baseline. Put the entry point of the open path on or above the origin. If necessary, change the path direction by selecting the open path and choosing *Reverse Selected Contours* from its context menu. Again, the origin can be the origin point of the glyph, i.e., where the baseline intersects with the left sidebearing, or it can be an anchor called ‘origin’.

In a target glyph, select exactly two adjacent nodes, e.g., the top nodes of the rectangle representing a stem. From the context menu, you can now choose *Add Cap*. The application will present a list of all available ‘_cap’ glyphs. Pick the one you want and click the *Select* button or press the Return key to confirm the dialog. The cap component will get bent into the receiving shape according to its position relative to the baseline. Cap components work best if the two receiving nodes have the same horizontal distance as the two end nodes of the open path inside the cap.

You can control the way the cap component is adapted to the target path by clicking on the gray selection knob, and choosing options in the gray Info box. You can change the scale by entering a new percentage value for the vertical or horizontal scale. The *Fit* option scales the cap horizontally so that the open ends fit on to the two nodes of the target path. The option will disable the horizontal scale value. Change alignment by clicking on the alignment symbols. Aligning to the left or right will orient the cap component to one side of the receiving path and take positions of the ‘left’ and ‘right’ anchors into account. The position of both ‘left’ and ‘right’ anchors, vertically relative to the baseline and horizontally relative to the start and end points, determines how the cap component is bent onto the receiving nodes of the target path.



To change the cap component, click on the name in the Info box, and choose a different one in the following dialog. To open the cap component for editing, click on the little arrow symbol.

To decompose all cap components present in the current glyph, choose *Glyph* > *Remove Overlap* (Cmd-Shift-O). To remove a cap component from an outline, select it by clicking on its gray knob, and press Delete or Backspace.

8 Spacing and Kerning

8.1 SPACING

Spacing is the process of adjusting the sidebearings of each letter to achieve an even rhythm in a line of text. There are no fixed rules for adjusting the white space. However, unless you are working on a monospaced font, similar or same shapes should have the same side bearing, e.g., the D will usually have the same LSB as the H, but its RSB closer to or the same as in the uppercase O.

'LSB' stands for the left, 'RSB' for the right sidebearing of a glyph.

8.1.1 Spacing Shortcuts

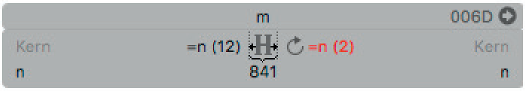
There are keyboard shortcuts to change the spacing of the current letter in Text mode (shortcut T). Hold down the Ctrl key and use the left and right arrow keys to change the LSB. Cmd and arrow keys change the RSB. Hold down both Ctrl and Cmd keys to change both LSB and RSB simultaneously, effectively moving the glyph inside its width. Add the Shift key to manipulate in increments of 10 units.

Easy to memorize: the Ctrl key is located on the left, the Cmd key on the right. The keys are associated with the left and right sidebearing, respectively.

The shortcuts for changing the LSB collide with default shortcuts for switching between macOS Spaces. You can change or deactivate the Spaces shortcuts in the System Preferences.

8.1.2 Metric Keys

In order to link metrics between glyphs, you can use so-called metric keys instead of mere numeric metric values. Put the name of the glyph you want to link to in the respective side bearing or width field and it will automatically adopt the side bearing or width of the linked glyph. Note that keyed metrics are not kept in sync automatically. You need to select the glyphs that need updates and select *Glyph* > *Update Metrics* (Ctrl-Cmd-M). Or update by clicking in one of the metric fields in the Info box and then clicking somewhere else.



m 006D

Kern =n (12) =n (2) Kern

n 841 n

You can even enter simple calculations into the sidebearing field. Calculations need to start with an equals sign (=). For instance, '=n+10' will take the same sidebearing of n and add 10 units, ‘=n-10’ subtracts 10 units, ‘=g/2’ yields half the sidebearing of g, and ‘=v*2’ doubles the sidebearing of v. Simply ‘=n’ has the same effect as writing ‘n’ into the field. And equating with a number, e.g., ‘=20’, allows to update to a predefined width or sidebearing. This can be useful for a monospaced font where the width of all glyphs must be the same, or for a connecting script font where the sidebearings must stay at a fixed value.

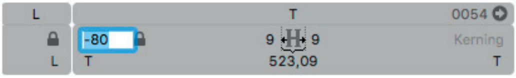
Use the pipe character (|, Shift-backslash on a U.S. keyboard, Opt-7 on a German keyboard, Opt-1 on a Spanish keyboard) to reference the opposite sidebearing. E.g., in the LSB of u, you can enter ‘=|n’ to use the RSB of n for the LSB of u.

8.2 KERNING

If the spacing has been done properly, most glyphs fit well next to each other. But some glyph pairs still need specific adjustments. Usually, glyphs with a lot of white space are problematic. E.g., V before A, L before W, or T in combination with an unaccented lowercase letter like o will look like there is too much space between them. This is where kerning comes into play. Kerning is the adjustment (increasing or decreasing) of distance between two specific letters.

8.2.1 Ways to Kern

To define these kerning pairs, switch to the Text tool (shortcut T), type both glyphs in the Edit view and place the cursor between them. The left field (labeled *Kerning*) in the Info box will show the kerning value for the combination of the previous glyph with the active glyphs. On the opposite end of the Info box, the *Kerning* field shows the kerning value for the combination of the active glyph with the following one. Just click and type a value there.



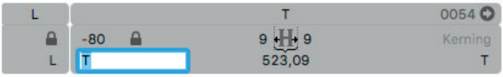
Tip: Again, Ctrl (to the left of the Option key) corresponds to the left side of the glyph, Cmd on the right to the right side.

Or use the keyboard shortcuts for much greater convenience. Ctrl-Opt-arrow keys changes the kerning towards the letter on the left side of the current letter, while Opt-Cmd-arrow keys change the kerning with the letter following on the right. Add the Shift key for increments of 10.

8.2.2 Kerning Groups

Many glyphs look similar and need the same kerning values. Kerning groups capture these similarities and help you reduce the number of pairs that need to be set manually. Kerning then applies not just to pairs of glyphs, but of groups of glyphs. This way, you can not only kern A and T with each other, but all accented A and T glyphs as well, given that they all reside in the same groups.

In Glyphs Mini, kerning classes are not edited as glyph lists. Instead, the class membership is defined as a glyph property. For example, put O in the left kerning group field for all glyphs that look like an O on the left (like C, Ccedilla, G, Odieresis, and O itself). Add a value to each glyph, even if it remains the only glyph in its group.



The small extra panel on the left of the Info box displays kerning info for the glyph to the left of the cursor: It shows the name of the left glyph, its group lock for the current pair, and its kerning group at the bottom.

You can introduce exceptions to group kerning by opening the group locks in the gray Info box. Thus you can have different kerning pairs for 'To' and 'Tö', for example. Close a group lock to remove the kerning exception for that glyph. Within one pair, the kerning can be an exception for one glyph while it applies to the whole group for the other glyph. E.g. the kerning value defined for 'Tö' can be an exception for ö, since it does not apply to the other o glyphs. Thus, for ö, the (left) lock is open. However, the value does apply to all T's including variations like Ť or Ț. So, for T, the lock remains closed.

8.2.3 Finding and Viewing Kerning Pairs

The Kerning window (*Window* > *Kerning*) lists all available kerning pairs, sorted by the first glyph or glyph class of the pairs. Groups are marked with an at sign, e.g. '@A' for the A kerning group, and displayed in dark blue. Individual glyphs are displayed in a softer beige. If the currently selected pair contains a group, you can choose *Show All Glyphs* from the gear menu to insert all possible glyph combinations for the current groups in the front-most Edit tab.

The search field on top of the Kerning window allows you to narrow down the displayed pairs. Clicking on the magnifying glass symbol gives you additional options, such as searching only for one side, only for groups or glyphs etc.

8.2.4 Deleting Kerning Pairs

In the Kerning window (*Window* > *Kerning*, Cmd-Opt-K), clicking on the minus button in the bottom left corner deletes all selected kerning pairs. For the current glyph pair (i.e., the glyphs to the left and right of the cursor), you can simply delete the kerning value in the grey Info box (Cmd-Shift-I).

8.2.5 Cleaning up Kerning

After removing glyphs or after importing, invalid or impossible kerning pairs may be left over. You can remove them with the *Clean Up* function from the gear button in the Kerning window.

8.2.6 Compressing Kerning

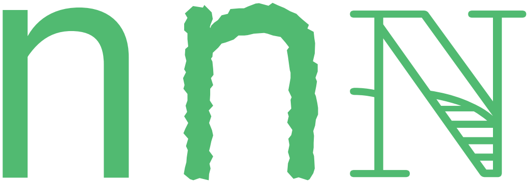
If you have kerned a few glyphs with each other before having set kerning groups, you can convert the singleton kerning pairs into group kerning with the *Compress* function of the gear button in the Kerning window. The function will delete the kerning between individual glyphs, and recreate it with their respective kerning groups wherever possible. Compressing kerning will keep explicit kerning exceptions that differ from the group kerning, but will remove exceptions that have the same value as corresponding group kernings. You may need to repeat the compressing process in order to catch singletons.

9 Hinting

9.1 WHAT IS HINTING?

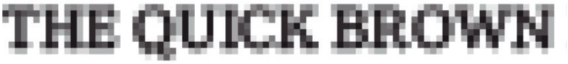
Hinting is a method to improve display at low resolutions. The eventual picture on the screen (the rendering) is created by software called the rasterizer. For fonts that are PostScript-based, such as the ones created by Glyphs Mini, the rasterizer needs to be relatively 'smart', because the fonts are said to be rather 'dumb'. But we can place 'hints' in the font, which the rasterizer can use to improve the rendering. The goal is a sharper, more consistent pixel image at low resolutions. To this end, the outlines will be distorted to achieve a better fitting on the pixel grid. In other words, hinting *does not preserve shapes,* on the contrary. If you have a font where the preservation of the shape is more important than a crisp pixel image, such as in connecting script typefaces and in icon fonts, hinting does not make sense. It only works if the font has repeated regular features. If the font is very irregular, like many handwritten, ornamental and grunge fonts, then hinting cannot improve the rendering.

If your font is intended for use in environments where hinting information is ignored, like displays with a very high resolution, or on Apple hardware with modern versions of macOS or iOS, then hinting will only make the font file larger. In cases like these, it is advisable to not hint the font.



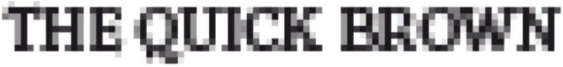
can be hinted *cannot be hinted* *cannot be hinted*

With PostScript hinting, the stems as well as the heights of the letters will appear more consistent in apps that support hinting. For example, this is what a sample text looks like in InDesign without hinting:



THE QUICK BROWN

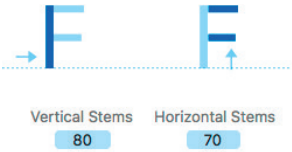
And this is what the same font in the same app would look like with hinting:



THE QUICK BROWN

9.2 FONT-WIDE HINTS

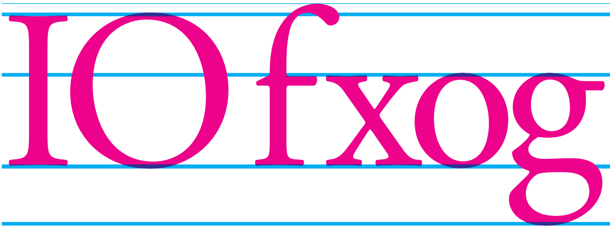
Hinting revolves around determining which part of a letter is a necessary stroke element and should not be omitted in small sizes. In order to achieve this, there are two kinds of hints. Firstly, general information that applies to the entire font. This is referred to as 'font-level hints' or 'font-wide hints', and encompasses standard stems and alignment zones (see 6.3, 'Stems and Alignment Zones', p. 57).



Measure the widths of your vertical and horizontal stems and arches and in the Font Info tab (*File > Font Info,* Cmd-I) enter them in the fields for *Vertical Stems* and *Horizontal Stems,* respectively. Horizontal stems are the thicknesses of the crossbars of lowercase e and uppercase A, and the height of the serifs. A vertical stem is the width of the stem of a lowercase n or an uppercase H, or the vertical parts of an uppercase O. Take as few values as possible, and merge values that are close to each other into one value. In the end you should have at most two H and two V values, if possible only one. Use more than one value only if it is acceptable that the respective stems are displayed with a huge difference at small screen sizes, e.g., with one and with two pixels, respectively, which amounts to a 100% difference.

Alignment zones take two values: a position and a size. The position is the vertical height of the zone, usually the vertical metrics, like x-height or ascender. The position is sometimes also referred to as the 'flat edge' of a zone. The size is the thickness of the maximum overshoot that may appear at that position. If the overshoot extends *above* the position (e.g., x-height, small cap height, cap height, ascender), the size value must be positive. Such zones are referred to as 'top zones'. If, however, the overshoots extend *below* the position (e.g., baseline, descender), the size must be negative and we call them 'bottom zones'.

A typical alignment zone setup: top zones with positive widths at ascender, cap height and x-height; bottom zones with negative widths at baseline and descender.



If your *Grid Spacing* settings amount to a grid finer than 1 unit, e.g., 1/10, 5/20, or 0/1, consider expanding your zones by one unit into both directions. E.g., change a zone at position 500, size 12 to position 499, size 14. The baseline zone must remain at position zero, though.

Alignment zones should be as small as possible, so do not try to make them larger 'to be on the safe side'. In any event, a zone must not be larger than 25 units. You can have a maximum of 5 top zones and 6 bottom zones. Zones must not overlap. There must be a minimum distance of one unit between them, the larger the better. The baseline zone must have a position value of zero.

If the font-wide parameters, i.e., alignment zones and standard stems, are set properly, you can let the autohinting algorithm do its magic by simply activating the *Autohint* option in the *File* > *Export* dialog.

9.3 GLYPH-LEVEL HINTS

The implementation of PostScript hinting in Glyphs Mini allows manual and automatic hints inside the same font. So, before resorting to manually inserting hints into your glyphs, we recommend to try to get as far as possible with autohinting. Only glyphs that do not display properly at low resolutions will need manual intervention. Manual and automatic hinting cannot complement each other *inside the same glyph*. Any manually hinted glyph is excluded from the autohinting process. Thus, if you decide to add hints manually, you must *fully* hint the glyph.

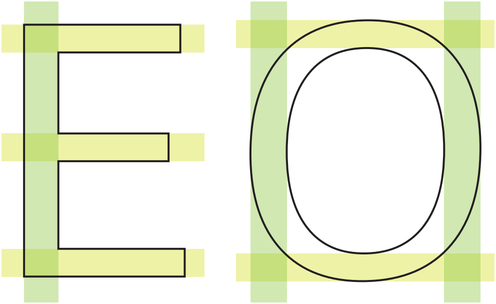
There are two types of glyph-level hints you can add manually, *stem hints* and *ghost hints*. Stem hints describe a vertical or a horizontal stem or stem-like feature of a glyph, like a serif or a crossbar. Ghost hints mark top and bottom edges when a horizontal stem hint cannot be applied. In combination with alignment zones, horizontal ghost and stem hints are important for the vertical alignment at the vertical font metrics, like the x-height or the ascender. At low resolutions, the rasterizer will try to vertically align the edges of all hinted horizontal stems that reach into an alignment zone. The horizontal hints must have their y coordinates in common with the nodes that are supposed to align. A single hint will do for all nodes it touches at its height.

9.3.1 Adding Stem Hints

To add a stem hint to a glyph, right-click and choose *Add Horizontal Hint* or *Add Vertical Hint* from the context menu. A gray bar with a number badge will appear. The numbers indicate the origin (first number) and width (second number) of the hint.

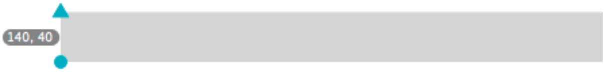
If you add a hint while two nodes are selected, the hint will be *linked* to these nodes right away. Adding linked hints this way even works on multiple node pairs at once, as long as each pair is on a separate outline. For best results, always link your hints to extremum nodes (see 2.3.15, 'Extremes and Inflections', p. 20).

Positioning of vertical stem hints (green) and horizontal stem hints (yellow).

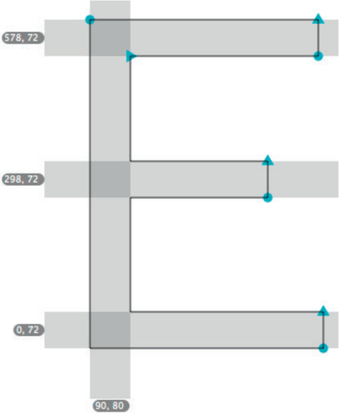


Select a hint by clicking on its gray number badge. Shift-click to select multiple hints. You can then edit the values numerically in the gray Info box (*View* > *Show Info*, Cmd-Shift-I). When exactly one hint is selected, press Tab to select the next hint, or Shift-Tab to go to the previous one.

A horizontal stem hint, its number badge indicating a position of 140 and a width of 40.



To edit a hint graphically, drag the blue marks at the edges of the hint. The blue circle indicates the hint origin, while the triangle shows the size and orientation of the hint. If you drag one of the markers onto a node, Glyphs Mini will link the hint to the position of the node. If you later move the node, the hint will adapt accordingly. You can delete one or more hints by selecting them and pressing the Backspace or Delete key.

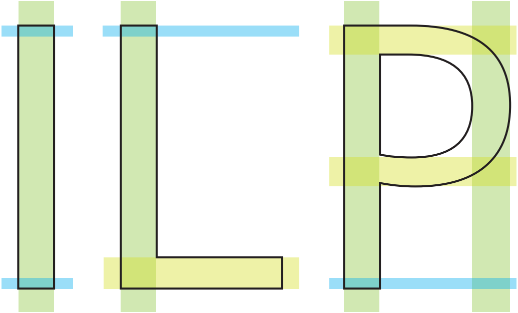


Hints linked to nodes with the blue triangle and circle.

In the final OTF, all stem hints must be positive, i.e., have a width greater than zero. But even if you mistakenly insert a negative hint, Glyphs Mini will correct its direction at export time, and all stem hints are turned positive.

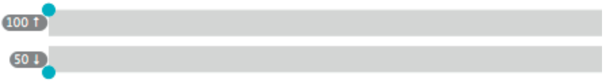
9.3.2 Adding Ghost Hints

You can use ghost hints when you need to vertically align the top or bottom of a glyph but cannot apply a horizontal stem hint. Take, for instance, a sans-serif uppercase I. The top needs to align with the cap height zone, the bottom with the baseline zone. In a serif I, you would apply horizontal hints to the serifs, but the sans-serif letter lacks the horizontal features necessary for a horizontal hint. In this case, you need to put a top ghost hint on the top of the I, and a bottom ghost hint at the bottom of the I. Similar situations occur on the top of a sans-serif uppercase L, and at the bottom of a sans-serif uppercase P:



Positioning of ghost hints (blue) alongside regular stem hints.

You can create a ghost hint by right-clicking on a single node and choosing *Add Horizontal Hint* from the context menu. Turn any existing hint into a ghost hint by right-clicking the coordinate badge of a hint and choosing *Make Ghost Hint* from the context menu. The badge of a ghost hint only displays the position and its orientation. A downward arrow indicates a bottom ghost hint, an upward arrow a top ghost hint. Attach it to a point by dragging the blue circle onto a node. You can set its vertical orientation by selecting the hint and clicking on the upward or downward icon in the gray Info box (*View > Show Info*, Cmd-Shift-I).



A top ghost hint at position 100, and a bottom ghost hint at position 50.

9.4 TESTING THE SCREEN RENDERING

Test the hinting in an Adobe application (see 2.11.2, 'Previewing in Adobe Applications', p. 36). Write a test text and zoom out far enough so that the letters are displayed with a few pixels only. Then zoom in using the operating system's Zoom function, which you can configure in the *Accessibility* settings of the System Preferences.

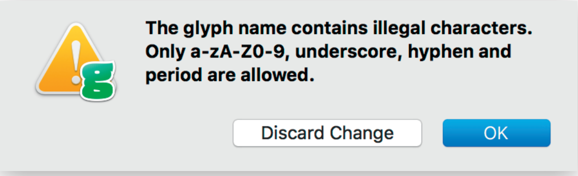
10 Error Handling

10.1 GLYPH NAMES

The most common source of problems when exporting fonts are bad glyph names. Make sure all your glyph names

- only contain letters A-Z, a-z, numerals 0-9, underscore (_), hyphen (-) or period (.),
- start with a letter (A-Z, a-z), except for '.notdef', and non-exporting glyphs, e.g., smart glyphs, that start with an underscore,
- contain no whitespace characters (space, tab, return, etc.), not even at the end,
- and contain no non-ASCII characters (like á, è, ß or ü).

Not adhering to these guidelines may yield an error message at export time like 'There is a problem with a glyph named:', followed by the bad glyph name and a descriptive explanation of the problem within brackets, e.g., 'The glyph name should not contain any space character.' Such error messages can also occur when trying to compile the OpenType feature code in *File* > *Font Info*. You can find bad names by searching for space (or other invalid characters) in the search field of the Font view (Cmd-F).



The glyph name contains illegal characters. Only a-zA-Z0-9, underscore, hyphen and period are allowed.

10.2 FONT NAMES

If your error message only entails a POSIX path to a file called 'FontMenuNameDB', set between brackets (e.g., '[/Users/<your username>/Library/Application Support/Glyphs-Mini/Temp/<your fontname>/FontMenuNameDB]'), then there probably is an invalid character in the font name or style name. Revise your entries in *File* > *Font Info* (Cmd-I). You can use spaces, but no non-ASCII characters.

10.3 DUPLICATE UNICODE VALUES

In some fonts, two or more letters erroneously carry the same Unicode. If you try to import such a font with double encodings, Glyphs Mini will warn you about it. Selecting all glyphs and choosing *Font > Update Glyph Info* will usually fix that problem. In the reverse situation, where one glyph sports two Unicodes, Glyphs Mini will silently reset the Unicode value based on the glyph name. The usual suspects for this are Delta and Omega. Delta should only be U+0394, but sometimes also has U+2206, the code for the mathematical increment operator (glyph name: increment). Omega, which should only be encoded with U+03A9, sometimes also sports U+2126, the codepoint for the Ohm symbol (glyph name: Ohm).

Tip: If you do want to keep a glyph accessible via two different codes, it is better to create a duplicate glyph.

10.4 MISSING OUTLINES

Sometimes the font does export, but some glyphs are empty or parts of glyphs are missing.

10.4.1 Open Paths

Outlines which are not closed are ignored and will not show up in the final font. So, if a glyph appears empty in the OTF, it is a good idea to check if its paths are actually closed.

10.4.2 Wrong Path Orientation

Outlines must be oriented counter-clockwise, except for counters, which need to be oriented clockwise. Otherwise, counters may appear 'closed' or missing. You can fix the orientation of selected outlines manually by choosing *Reverse Selected Contours* from their context menu, or reverse all outlines via choosing *Reverse All Contours* from the context menu when no outlines are selected. Glyphs Mini will try to apply heuristics to fix outline directions on any number of selected glyphs with *Glyph > Correct Path Direction*.

10.4.3 Multiple Paths on Top of Each Other

Two similar paths on top of each other with varying path orientations may delete each other at export. You can select exactly one whole path by double clicking it. Press the Backspace key to delete it. If it looks like nothing has changed, then you had at least two identical outlines on top of each other.

11 Import and Export

11.1 VECTOR DRAWING APPLICATIONS

11.1.1 Adobe Illustrator

Tip: To quickly get the right scale, draw a rectangle with the height of the capital letters in Glyphs, copy and paste it into an Illustrator artboard and scale the drawings to fit the height of the rectangle.

First, find the right scale for your drawings in Adobe Illustrator. One point in Illustrator corresponds to one unit in Glyphs Mini, i.e., an element that is 100 pt high will end up at a height of 100 units in Glyphs Mini. Alternatively, you can start in Illustrator and set the artboard to a height of your UPM size in points, e.g., 1000 pt if your UPM is 1000. Both ascender and descender should fit inside the artboard. If you set the page origin to the intersection of base line and LSB, the paths will have the right position. In Illustrator up to Version CS4, set the origin of the page by dragging the cross hair in the top left corner between the rulers. In Illustrator CS5 or later, set the origin of the page in the *Artboard Options*.

11.1.2 Sketch

If you are importing outlines from Sketch, set your contour to 1 pt outline without fill. This will prevent double outlines in Glyphs Mini.

11.2 FONTLAB STUDIO

11.2.1 From FontLab Studio 5 to Glyphs Mini

There is a FontLab macro available for exporting Glyphs Mini files directly out of FontLab Studio. Point your web browser to github.com/schriftgestalt/Glyphs-Scripts/ and download the script called ‘Glyphs Export.py’. To install, invoke *Go > Go to Folder* in Finder, enter ~/Library/Application Support/FontLab/Studio 5/Macros/ and place the file in the Macros folder that is displayed then. After restarting FontLab, it will become available in the macro toolbar.

11.2.2 From Glyphs Mini to FontLab Studio 5

Again, there is a Python script to import Glyphs Mini files into FontLab. You can get it from the same Github repository mentioned above. This time, look for a script named ‘Glyphs Import.py’ and place it in your FontLab Macros folder.

Keep in mind that some features are not available in FontLab, and therefore the result may not be exactly the same.

For instance, FontLab Studio cannot nest components, and therefore, the script will decompose them in the conversion.

11.3 OPENTYPE

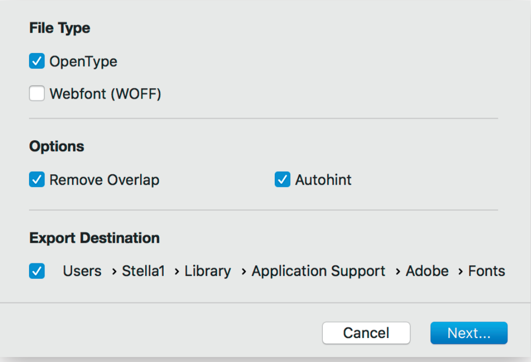
11.3.1 Opening Existing Fonts

While you can open existing PFB, OTF, TTC, and TTF fonts, Glyphs Mini cannot reverse-engineer all the information inside a compiled font file. That means that opening an OTF and exporting it again will produce a file that is different from the original. It is strongly advised to always work on a copy.

For more details on opening existing fonts, see glyphsapp.com/tutorials/importing-existing-fonts

11.3.2 Generating OpenType Fonts

Choosing *File* > *Export* (Cmd-E) will bring up the Export dialog. To generate an OpenType font for use in layout or word processing applications, pick *OpenType* as export format.



File Type

☑ OpenType

□ Webfont (WOFF)

Options

☑ Remove Overlap ☑ Autohint

Export Destination

☑ Users › Stella1 › Library › Application Support › Adobe › Fonts

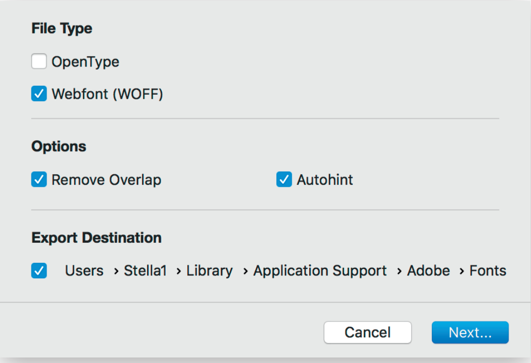
The *Remove Overlap* option applies an algorithm to remove path overlaps at export, similar to the *Remove Overlap* function in the *Glyph* menu. In a release font, overlaps must be removed. So, uncheck this only for testing purposes, to speed up the export. You can safely uncheck this option if you have already removed overlaps in the file.

The *Autohint* option applies the AFDKO autohinting algorithm to all glyphs that do not have any manually set hints. This option expects standard stems and alignment zones to be set correctly (see 9.3.1, ‘Adding Stem Hints’, p. 78). Uncheck this option for fonts that should not be hinted, or in order to speed up export for testing purposes.

The *Export Destination* option allows you to set a default location into which the fonts will get exported. If you do not set a destination, Glyphs Mini will bring up a save dialog. Be aware that exporting your font will overwrite any previous instances with the same name in the same export location. This can be useful if you use the Adobe Fonts folder as export destination (see 2.11.2, 'Previewing in Adobe Applications', p. 36).

11.3.3 Generating Webfonts

To export OpenType fonts for the web, choose *File > Export* (Cmd-E) and check the *Webfont* option.



File Type

☐ OpenType

☑ Webfont (WOFF)

Options

☑ Remove Overlap ☑ Autohint

Export Destination

☑ Users › Stella1 › Library › Application Support › Adobe › Fonts

The *Export Destination* setting works like it does for CFF-based OpenType desktop fonts. See section 11.3.2, 'Generating OpenType Fonts' (p. 85) for details.

12 Preferences

12.1 ACCESSING APPLICATION PREFERENCES

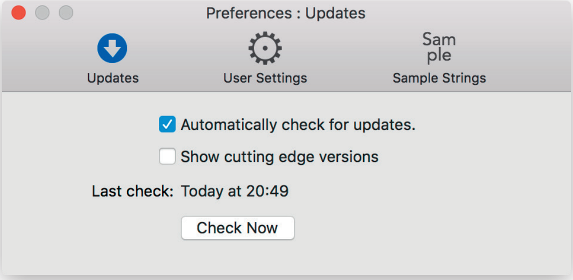
Open the Preferences window through *Glyphs Mini* > *Preferences* (Cmd-comma) to access all application-wide preferences.

12.1.1 Updates

The *Updates* section controls how Glyphs Mini handles software updates. Clicking on the *Check now* button will initiate a version check and, if available, offer you a new version for download and installation, presenting you a change log describing the latest additions and changes. The time stamp of the latest check is displayed next to the button.

If enabled, the *Automatically Check for Updates* option will periodically try to establish an internet connection and see if a new app version is available for download from the Glyphs servers. We recommend keeping this option on.

Show Cutting Edge Versions will also offer you beta versions, rather than only stable versions. Bug fixes are available much sooner this way. Enable this option if you cannot wait for the stable version or want to experiment with new app features. Although, beta versions are not as thoroughly tested as stable versions. That is why we recommend to always work with file copies if you are testing cutting edge versions.



☑ Automatically check for updates.

□ Show cutting edge versions

Last check: Today at 20:49

You can revert back to the stable version by downloading the trial from glyphsapp.com/mini.

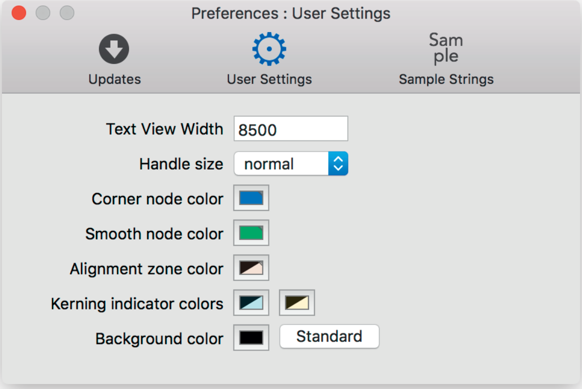
12.1.2 User Settings

Text View Width controls the line width of text entered in Edit view. The value entered is measured in thousandths of an em.

This measurement is independent from your *Units per Em* value as entered in Font Info.

Handle Size controls the size of all displayed points in Edit view, such as on-curve nodes, control handles, anchors, and position markers of manual hints. Larger nodes make it easier to click-select a point, while smaller nodes make for a cleaner user interface.

You can set the display colors of corner nodes, smooth nodes, alignment zones, positive and negative kerning indicators in the Edit view, and the canvas color. The *Standard* button resets the color values to their defaults.



Text View Width 8500

Handle size normal

Corner node color 🟦

Smooth node color 🟩

Alignment zone color

Kerning indicator colors

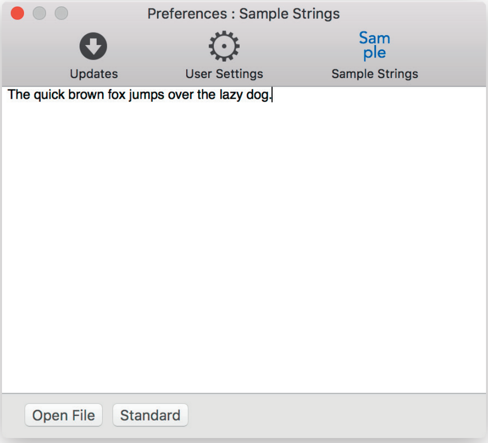
Background color ⬛

12.1.3 Sample Strings

Here, you can import, edit, reset, copy, or paste default sample strings, separated by newlines. The text field is Unicode savvy, so you can enter any diacritic and non-Latin characters. Use a backward slash followed by a lowercase n (\n) for a newline in the sample string. Use a forward slash followed by the glyph name and a separator character (either a space or another forward slash) as an escape for any glyph. This is especially useful for hard-to-type or unencoded glyphs, e.g., OpenType variations such as '/a.sc'.

Use '/Placeholder' (with a forward slash and a separator character at the end) for a placeholder function, i.e., the current glyph in Edit view. This can be useful for seeing the current glyph several times between other glyphs, e.g.: /o/Placeholder/o/Placeholder/Placeholder/o/o/Placeholder for setting the current glyph around and between lowercase o's, which may help when spacing.

The *Open File* button allows you to import an external text file, whereas the *Standard* button will reset the sample texts to the app defaults. For more details on how to use sample strings, see section 2.8.1, 'Sample Texts' (p. 29).



The quick brown fox jumps over the lazy dog.

13 Appendix

13.1 AUTOMATIC FEATURE GENERATION

Glyphs Mini can automatically generate a number of OpenType features if it finds glyphs with certain names in the font.

liga *Ligature* Join the glyph names of the components with an underscore (‘_’). Some common ligatures (f_f_i, f_f_l, f_f, fi, fl, lu_lakkhangyao-thai, ru_lakkhangyao-thai) are automatically placed in the liga feature, all others go into dlig by default. However, if you want to force a ligature into the liga feature, you can add a ‘.liga’ suffix to its name, e.g., f_b.liga or yi_yi-cy.liga.

dlig *Discretionary Ligatures* Join the glyph names of the components with an underscore (‘_’), e.g., f_odieresis.

For a complete list of language tags, see: microsoft.com/typography/otspec/languagetags.htm

Find out more about these languages and other localizations: glyphsapp.com/tutorials/tag/languages

locl *Localized Forms* Add ‘.loclXXX’ to the glyph name, where XXX represents the three letter language tag, e.g., ‘.loclENG’ for English, or ‘.loclSVE’ for Swedish. There also is a built-in list of glyphs which which trigger localizations:

- idotaccent, i.TRK or i.loclTRK: trigger i substitutions for TRK, AZE, CRT, KAZ and TAT if i is present.
- Scommaccent, Tcommaaccent, scommaaccent, and tcommaaccent: trigger substitutions for ROM and MOL if Scedilla, Tcedilla, scedilla, and tcedilla are present.
- periodcentered.loclCAT (add a .case suffix for uppercase; L_periodcentered_L.loclCAT, l_periodcentered_l.loclCAT; or the legacy Ldot, ldot: trigger ella geminada (punt volat) substitutions for CAT if L, l, and periodcentered are also present in the font.
- Iacute_J.loclNLD and iacute_j.loclNLD, or Jacute and jacute: trigger accented ij substitutions for NLD if Iacute, iacute, J, and j are present.

ordn *Ordinals* Automatically built if default figures, numero, ordfeminine and ordmasculine are found in the font.

frac *Fractions* The frac feature is generated from the .numr, .dnom and fraction glyphs. If they are not present in the font, then the feature will be composed from any pre-built fractions available in the font, like onehalf, onequarter, threequarters etc

aalt *All Alternatives* Glyphs automatically builds the aalt feature based on all features that substitute glyphs.