

XGlass

GLASS FOR UI TOOLKIT



UNITY

6.3 or newer

WORKS IN

UI Toolkit runtime panels,
editor windows and UI
Builder

SETUP

No GameObjects, cameras
or render textures

EDIT WITH

UI Builder, UXML, USS and
C#

XGlass User Guide

01	Introduction				03
	What XGlass does	03	How it works	04	Requirements
02	Getting started				05
	Installation	05	Quick start	06	
	Folders	05	In UXML	07	
03	Glass elements				08
	XGlass	08	Glass on glass	09	XGlassGroup
04	Glass Profiles				11
	Creating a profile	11	Normal Map	13	Highlights
	The default profile	12	Color	14	Dark Edge
	The preview	12	Ink	14	Shadow
	Glass	13	Reflections	15	
05	Normal maps, reflections and ink				19
	Normal maps	19	Reflections	20	Ink
06	Glass Shapes				22
	Making a shape	22	Shape settings	23	Good to know
07	Overriding single values				24
	In UI Builder	24	In USS	25	USS property reference
	In UXML	24	What wins	25	
08	Scripting				27
	Creating glass from code	27	API reference	28	
09	The demo				30
	The five pages	30	Liquid dragging	31	
10	Limitations and troubleshooting				32
	Limitations	32	Troubleshooting	33	

Glass for UI Toolkit.

XGlass adds glass to UI Toolkit. A glass element bends, blurs and tints whatever is behind it, lights its rim and casts a soft shadow. Glass in a group melts together where pieces come close.

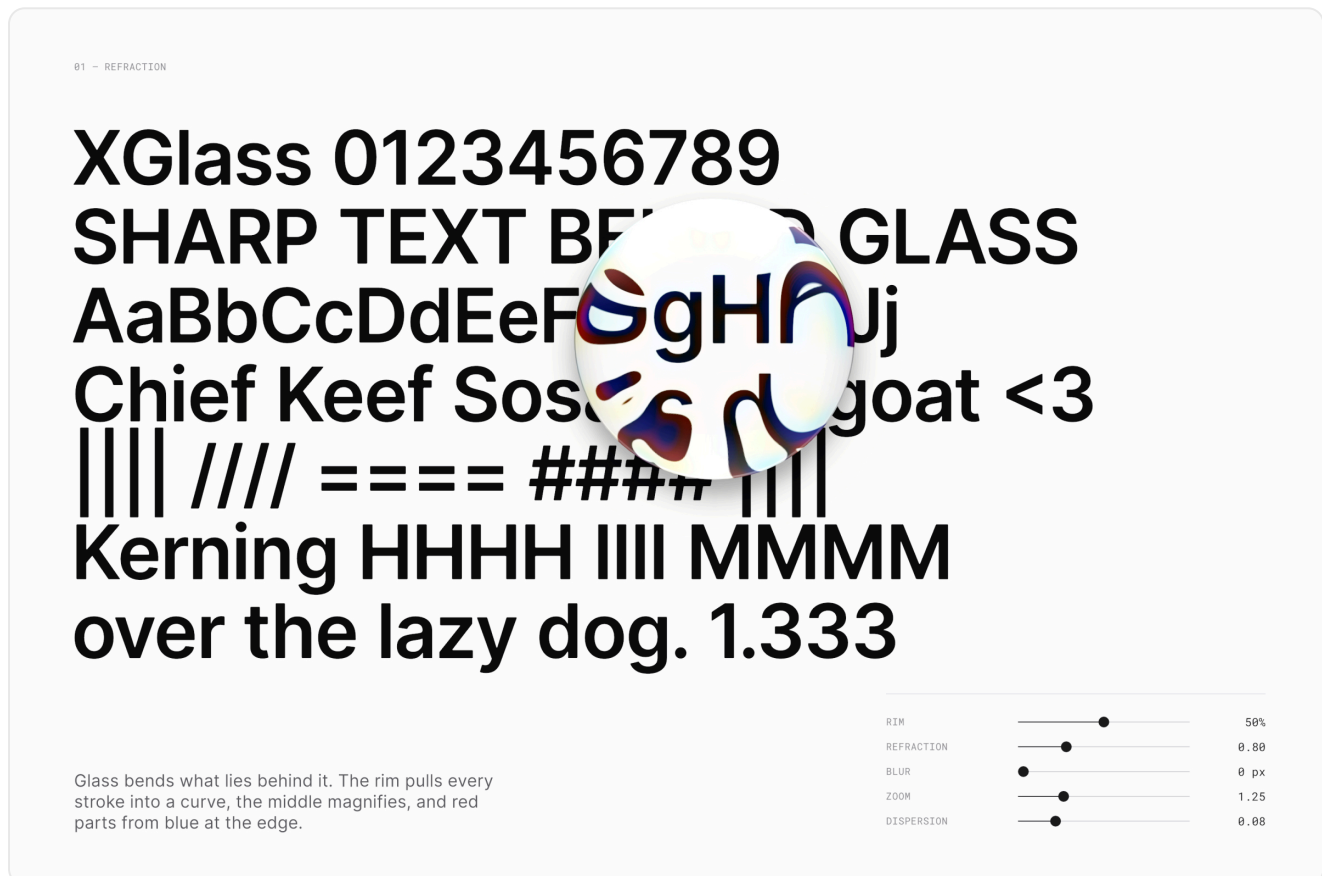


FIG. 1 A clear lens over sharp type, from the demo's Refraction page. The rim pulls every stroke into a curve, the middle magnifies, and red parts from blue at the edge.

1.1

What XGlass does

Refraction

The rim bends what lies behind it, with zoom, blur and colour fringes at the edge.

Glass on glass

Glass inside glass, as deep as you like. Every pane bends the panes beneath it.

Merging

Glass in a group flows into one surface where pieces come close.

Single values

Override any value per element in UI Builder, or per class with USS.

Normal maps

A surface texture whose bumps work as tiny lenses.

Ink

Colour suspended in the glass, drifting like paint in water.

Glass Profiles

A complete look in one asset, with a live preview in its Inspector.

Glass Shapes

Any outline, baked from a PNG or SVG. The rim, light and shadow follow it.

Reflections

An HDRI, a skybox or a live probe: faint face-on, nearly a mirror at the edge.

Rim lighting

Highlights, a glint and a dark edge, lit from any angle.

1.2

How it works

Glass is a UI Toolkit element. It runs a backdrop filter: UI Toolkit hands it what's behind the element, and XGlass's shader bends, blurs, tints and lights it. There are no cameras, render textures or scene objects to set up, so the same element works in runtime panels, in editor windows and on UI Builder's canvas.

Three things make up every piece of glass:

- **Elements.** XGlass is a pane of glass and XGlassGroup melts the glass inside it together. Drag them in from UI Builder's Library like a Button.
- **Glass Profiles.** Assets that hold a complete look, shared by any number of elements. The project has a default one.
- **Single values.** Any value can be overridden per class with USS custom properties, or per element in UI Builder, UXML or C#.

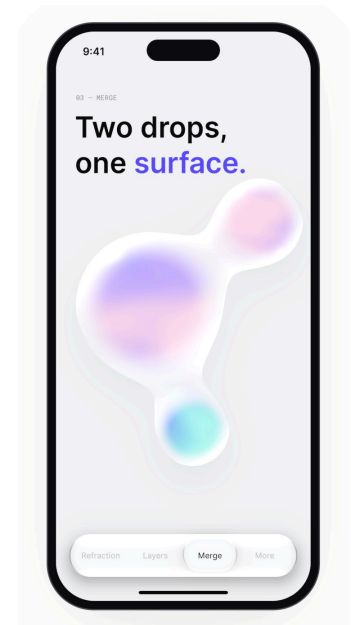


FIG. 2 The demo's Merge page laid out for a phone, with a glass tab bar.

1.3

Requirements

- **Unity 6.3 or newer.** XGlass is built on UI Toolkit's custom filters, which older versions don't have.
- **UI Toolkit** runtime panels or editor windows. No GameObject is needed for the glass itself.
- **Vector Graphics module** (**Package Manager** > **Built-in** > **Vector Graphics**), only to bake Glass Shapes from SVG files. New projects include it; PNGs work without it.

Your first pane.

Drag a glass element into a UI Builder document, give it a size and rounded corners, and put something colourful behind it. That's all it takes.

2.1

Installation

Open **Window** › **Package Manager**, select XGlass under **My Assets**, then download and import it. Everything lands in `Assets/XGlass`.

XGlass loads its filter definition, settings and default profile from `XGlass/Core/Resources`. If the console ever reports the filter definition missing, run **Tools** › **XGlass** › **Repair Setup**: it recreates whatever is missing.

2.2

Folders

XGlass/	
Core/	Everything XGlass runs on.
Runtime/	The code that runs in builds.
Elements/	XGlass and XGlassGroup.
Profiles/	Glass Profiles and the project settings.
Styling/	How USS and an element's own values combine with its profile.
Shapes/	Glass Shapes and their baking.
Rendering/	The filters, the layers merged glass draws on, the shadows.
Editor/	The inspectors, the profile editor and Repair Setup.
Shaders/	The glass shader.
Resources/	The filter definition, the settings asset and the default profile, loaded by name.
Documentation/	This guide.
Samples/	The demo. Nothing in Core depends on it, so you can delete it.

Quick start

- 01 **Open a UXML document in UI Builder, or make one with Create**
 › UI Toolkit › UI Document.
- 02 **Add a glass element.** In the Library, choose **Project**, open **Custom Controls (C#)** › **XGlass** and drag **XGlass** into the Hierarchy.
- 03 **Give it a size and a border-radius (Inspector › Border › Radius;** expand it to set each corner). The glass follows every corner. A percentage is taken of the width across and the height down, so 50% makes a square box a circle.
- 04 **Put something colourful behind it,** such as an image or big type, to see the glass bend it.

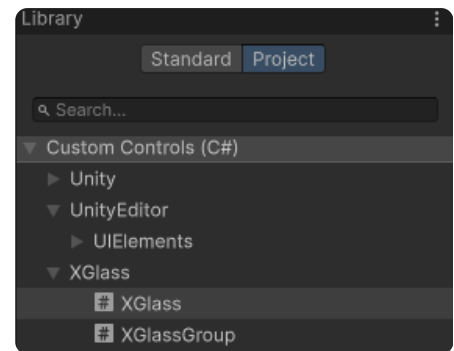


FIG. 3 XGlass and XGlassGroup in UI Builder's Library.

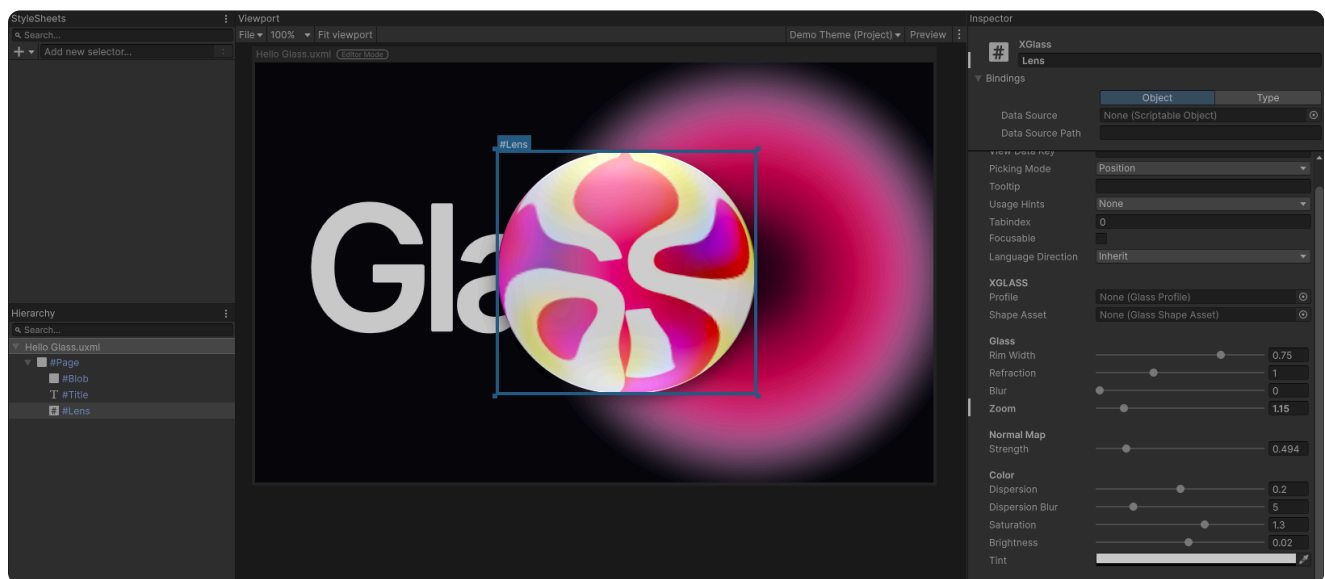


FIG. 4 A glass lens on UI Builder's canvas, over type and a colour blob. It draws the project's default profile; the Inspector on the right lists its values (chapter 7).

The same lens written by hand. The element draws the project's default profile until you give it one of its own.

```
<ui:UXML xmlns:ui="UnityEngine.UIElements" xmlns:xg="XGlass">
  <xg:XGlass name="Lens"
    style="width: 320px; height: 300px; border-radius: 150px;" />
</ui:UXML>
```

To see the whole tour, open `XGlass/Samples/Demo/XGlassDemo.unity` and press Play (chapter 9).

Panes and groups.

Two elements do all the work. **XGlass** is a pane of glass. **XGlassGroup** melts the glass inside it into one surface.

3.1

XGlass

A pane of glass. In UXML: `<XGlass.XGlass />`, or `<xg:XGlass />` with `xmlns:xg="XGlass"`. Every XGlass has the USS class `xglass`.

ATTRIBUTE	MEANING
profile	The Glass Profile it draws with. Empty uses the project's default profile.
shape-asset	A Glass Shape baked from a PNG or SVG (chapter 6). When set, the glass takes its outline instead of its rounded box.
rim-width, refraction, blur, zoom	How the glass bends: single values the element sets for itself.
normal-strength	How far the profile's normal map bends the light.
dispersion, dispersion-blur, saturation, brightness, tint	Colour fringes at the rim, and the colour of what's seen through the glass.
ink	How much the profile's ink colours the glass.
highlight, dark-edge	Overall strength of the rim lighting.

Values the element doesn't set come from USS and its profile; see chapter 7. The shadow has no attribute: set it with USS or the profile.

Shape and corners

The glass follows the element's `border-radius` and its transform the way UI Toolkit draws the element. A round lens is a square box with `border-radius: 50%`: scaled and turned by a transform (`scale` , `rotate`), it stays a smooth ellipse. Clicks only land on the glass, inside its rounded corners.

3.2

Glass on glass

Glass inside other glass, for example a selection bubble inside a tab bar, is drawn on top of it as glass on glass. Nesting needs no setup, and every pane bends the panes beneath it, rims and highlights included.

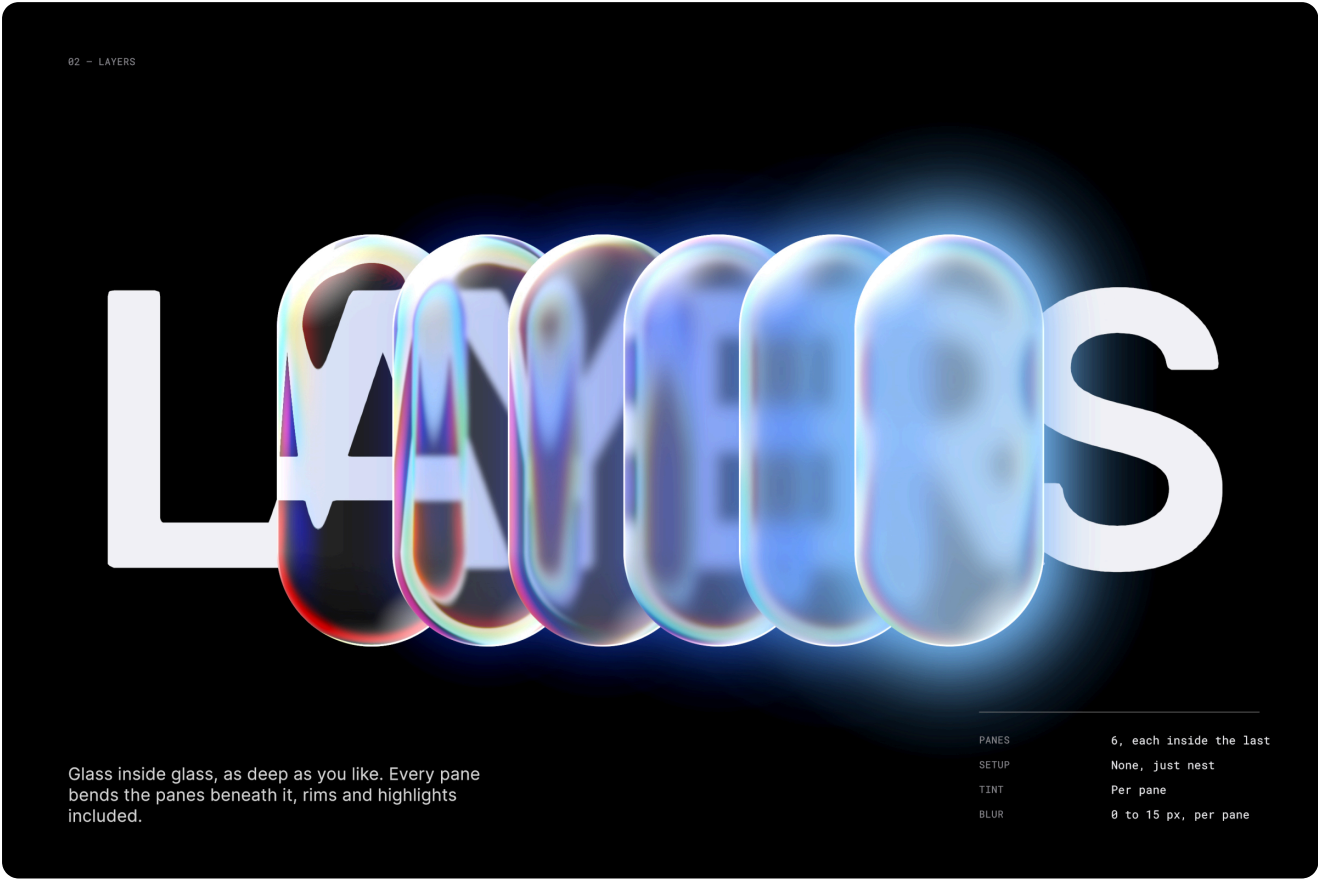


FIG. 5 Six panes, each inside the last, from the demo's Layers page. Each pane has its own tint and blur.

3.3

XGlassGroup

A container whose glass melts together, like Apple's `GlassEffectContainer`. Put `XGlass` elements directly inside it, and pieces that come closer than the merge distance flow into one surface. Every `XGlassGroup` has the `USS` class `xglass-group`.

ATTRIBUTE	MEANING
merge-distance	Glass closer than this, in px, flows together. 0 keeps every piece separate. Unset, it uses <code>--glass-merge-distance</code> , or 16.
profile	Sets the merged glass's shadow, rim lighting, reflections, normal map texture and ink (its colours, size and flow). Empty uses its first glass's profile, so glass keeps its look when it joins a group.

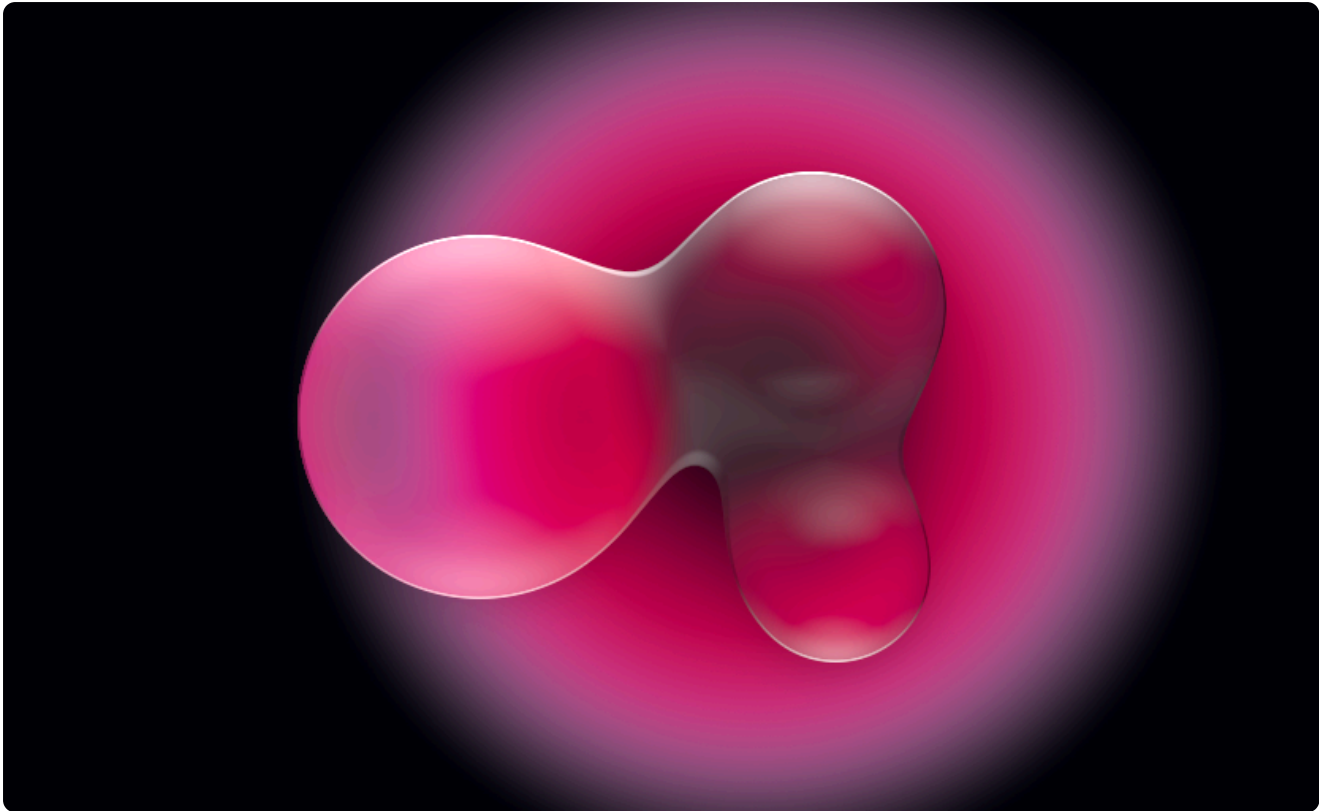


FIG. 6 Three round XGlass elements in an XGlassGroup with a merge distance of 48 px, on UI Builder's canvas.

How groups merge

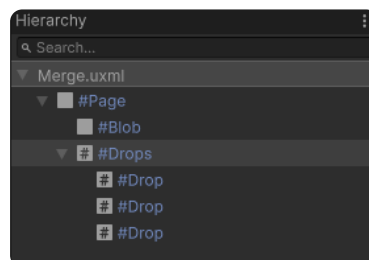


FIG. 7 The group and its glass.

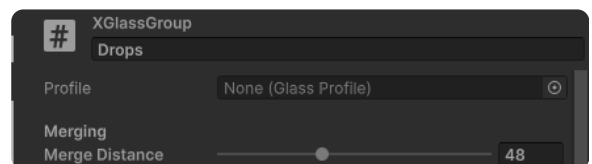


FIG. 8 The group's own fields in the Inspector.

- Only glass **directly** inside the group merges. Glass outside any group stays separate.
- Each piece keeps its own shape (a Glass Shape too), rim width, refraction, zoom, blur, colour, dispersion blur, highlight and dark-edge strength, normal map strength and ink strength. Where pieces meet, these blend, and their inks mix.
- The group draws its glass with one layer, the first thing inside it. The layer covers the group's box and its glass, plus room for the shadow, so glass can reach past the group's edge or be dragged out of it. Keep the group's `overflow: visible` (the default), or its own clip cuts the glass off at its edge.
- A group merges up to **16** pieces of glass. Any more are drawn separately.

One asset, a whole look.

A Glass Profile holds a complete look: how the glass bends, its colour and surface, what it reflects, how its rim catches the light and the shadow it casts. Elements share profiles, and changes show live in play mode, edit mode and UI Builder.

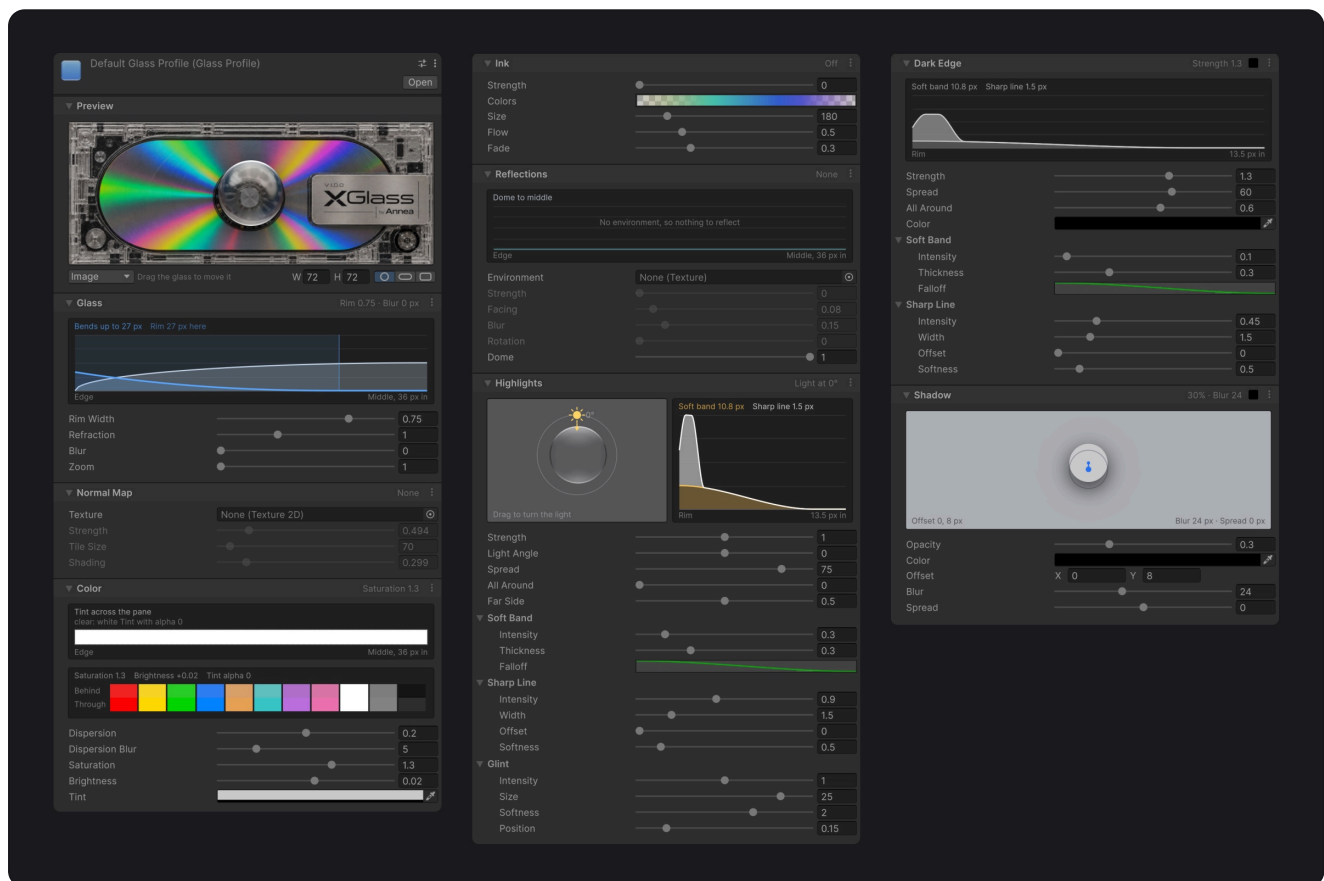


FIG. 9 The Inspector of a Glass Profile, laid out in three columns: a live preview, then a section for each part of the look.

4.1

Creating a profile

Choose **Create** > **XGlass** > **Glass Profile** in the Project window. Drag the profile onto an element's or a group's **Profile** field in UI Builder, or make it the project's default.

XGlass ships with **DefaultGlassProfile** in `XGlass/Core/Resources/XGlass`, the project default: no blur and a thick rim. The demo's profiles in `XGlass/Samples/Demo/Profiles` are good starting points too.

The default profile

Every glass element and group without a profile of its own draws with the project's default profile. Pick it under **Edit › Project Settings › XGlass**. The page shows the profile's Inspector right there, so you can tune the default without leaving it.

Clear the field and glass falls back to XGlass's built-in values.

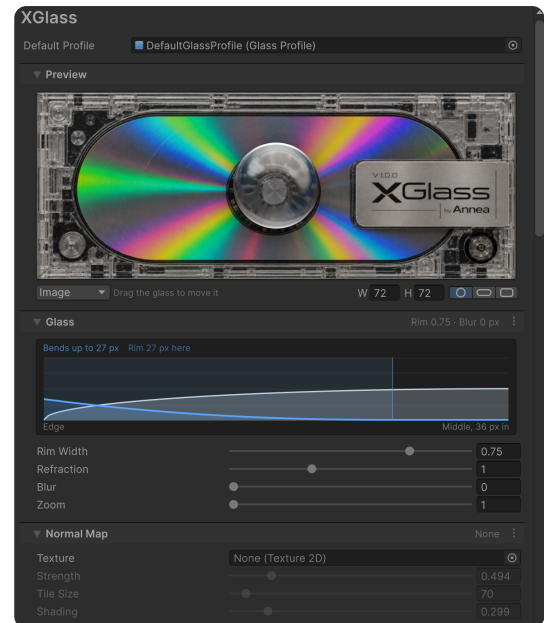


FIG. 10 Project Settings › XGlass.

The preview



FIG. 11 The preview: live glass over a test backdrop.

The profile's Inspector shows it on a pane of live glass over a test backdrop (an image, text or a grid), which you can drag around, as a circle, pill or rounded box. **W** and **H** set the pane's size in points of the preview's stage, which is 400 points wide and scales to fit the Inspector, the glass with it, so the glass keeps its size against the backdrop however wide the Inspector is.

Most sections open with a drawing of what their values do: the rim in cross-section, colours seen through the glass, how much it reflects from its edge to its middle, the rim lighting, the lighting layers across the rim, and the shadow. Drag the light around the rim lighting drawing to set **Light Angle**, and drag in the shadow drawing to set its **Offset**. The drawings measure from the preview's pane, so their px readouts are for a pane of that size.

Each section's menu (:) resets it to XGlass's built-in values. Those are the values listed with each setting on the following pages.

Glass

How the glass bends. Rim width is a share of the way from the glass's edge to its middle (half its shorter side, or a Glass Shape's deepest point), so the same profile looks alike on a small button and a big card.

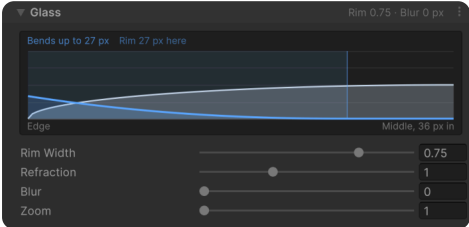
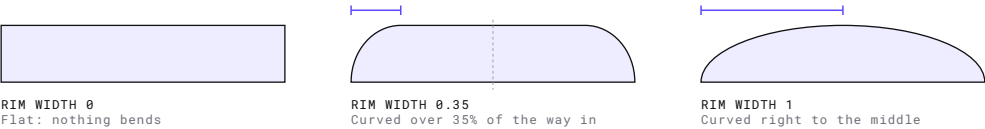


FIG. 12 The Glass section. The drawing shows how far the rim bends the light, edge to middle.

Rim Width 0–1 · 0.35	Width of the curved rim, as a share of the way from the edge to the middle. 1 curves right to the middle; 0 makes the glass flat, so nothing bends.
Refraction 0–3 · 0.9	How far the rim bends the light, in rim widths.
Blur 0–24 px · 3	Blur radius behind the glass.
Zoom 1–2 · 1	Magnification of everything seen through the glass. 1 leaves it unscaled.

Normal Map

A surface texture for the glass, like hammered or frosted glass. See chapter 5 for how it shapes the glass.

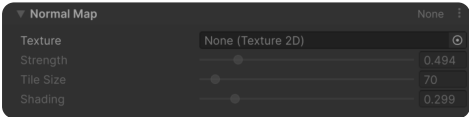


FIG. 13 The Normal Map section.

Texture None	The normal map. None turns it off. A texture not imported as a normal map shows a warning with a Fix Now button.
Strength 0–3 · 1	How far its bumps bend the light. 1 takes the map as it is, 2 doubles its bumps, 0 flattens them.
Tile Size 16–1024 px · 256	Size of one repeat. It tiles out from each element's centre, and the bumps scale with it.
Shading 0–2 · 0.5	How much the bumps catch the light, scaled by the highlight and dark-edge strengths. 0 leaves only their refraction and reflections.

Color

Colour fringes at the rim and the colour of what's seen through the glass. The drawings show the tint across the pane and a row of colours behind and through the glass.

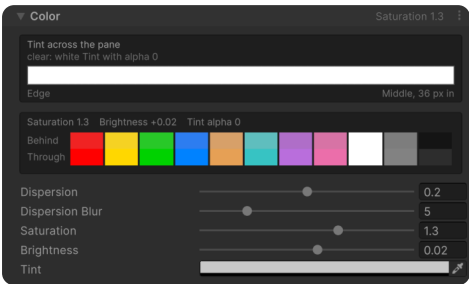


FIG. 14 The Color section.

Dispersion 0-0.4 · 0.05	How much more red bends than blue at the rim.
Dispersion Blur 0-24 px · 5	Blur on the colour fringes alone, so they blend softly even over a sharp backdrop. The image itself stays sharp. 0 splits the colours crisply.
Saturation 0-2 · 1.3	Colour saturation of what's seen through the glass. 1 leaves it unchanged.
Brightness -0.2-0.2 · 0.02	Brightness added to what's seen through the glass.
Tint White, alpha 0	Colour of the glass itself, lighter where it thins out towards the rim. White doesn't tint, and a lighter colour tints less. Its alpha is the opacity: it fills the glass evenly, like milky or painted glass. 0 is clear, 1 is solid.

Ink

Colour suspended in the glass, swirling slowly like paint in water. See chapter 5.

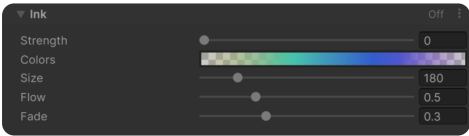


FIG. 15 The Ink section.

Strength 0-1 · 0	How much the ink colours the glass. 0 turns it off.
Colors Gradient	The colours the ink runs through. Alpha is how much ink there is: transparent keys leave the glass clear.
Size 16-1024 px · 180	Size of the swirls.
Flow 0-2 · 0.5	How fast the ink flows. 0 holds it still.
Fade 0-1 · 0.3	How far in from the rim the ink fades in, as a share of the way to the middle.

Reflections

An environment the glass reflects, like a skybox, and the dome it reflects with. See chapter 5.

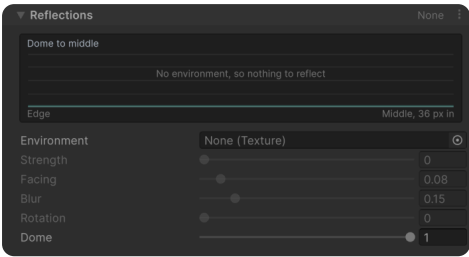


FIG. 16 The Reflections section, with no environment set.

Environment None	A cube map, or a 360° panorama or HDRI imported with Texture Shape set to Cube. None turns reflections off.
Strength 0-2 · 1	Strength of the reflections. 1 is real glass; more makes it shinier.
Facing 0-1 · 0.08	How much the glass reflects where it faces you, as a share of what it reflects at its very edge. Real glass: about 4%.
Blur 0-1 · 0.15	How blurry the reflections are, as if the glass were less polished. 0 is a sharp mirror, 1 the environment's softest mip.
Rotation 0-360° · 0	Turns the environment around the vertical axis.
Dome 0-1 · 0.8	How far in from the rim the surface bulges for the reflections, the glint and the tint's fade. Never less than the rim width.

Highlights

Light on the rim: a soft band, a sharp line and a glint where the curved rim reflects the light. They centre on the side facing the light and, weaker, the side across from it, where the light comes back out through the glass.

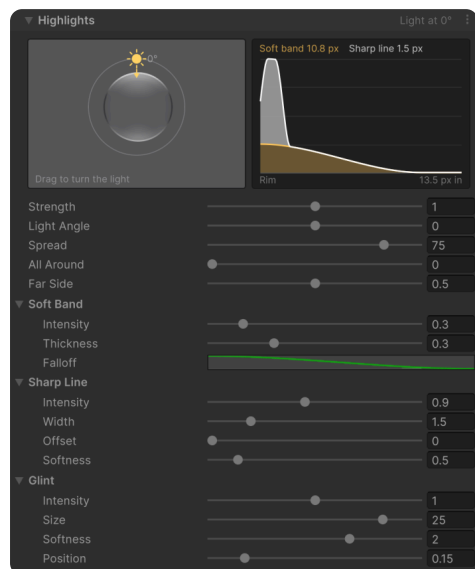


FIG. 17 The Highlights section. Drag the light around the dial to set Light Angle.

Strength 0-2 · 1	Overall strength, scaling all three layers. 0 turns them off.
Light Angle -180-180° · -45	Where the light comes from, clockwise from straight up: -45 is the upper left.
Spread 0-90° · 75	How far the highlights reach around the rim before fading out.
All Around 0-1 · 0	Share of the highlights kept all around the rim, beyond their spread.
Far Side 0-1 · 0.5	Strength on the side away from the light, as a share of the lit side's.
SOFT BAND	
Intensity 0-2 · 0.3	Strength of the band. 0 turns it off.
Thickness 0-1 · 0.3	Width of the band, as a share of the way to the middle.
Falloff Curve	Strength across the band, from the rim (left) to its inner side (right).
SHARP LINE	
Intensity 0-2 · 0.9	Strength of the line. 0 turns it off.
Width 0-8 px · 1.5	Width of the line.
Offset 0-12 px · 1.5	How far in from the rim it starts. 0 puts it right on the edge.
Softness 0-4 px · 0.5	How far each side of the line fades out.
GLINT	
Intensity 0-2 · 1.2	Brightness of the glint. 0 turns it off.
Size 1-30° · 7	Size of the light: a bigger light makes a broader glint.
Softness 0-3 · 0.6	How far the glint fades towards its edge, as a share of its size.
Position 0-0.95 · 0.2	Where across the dome it sits: 0 on the rim, 1 at the dome's inner side.

Dark Edge

Shade on the two sides of the rim between the highlights, like the thick side of real glass.

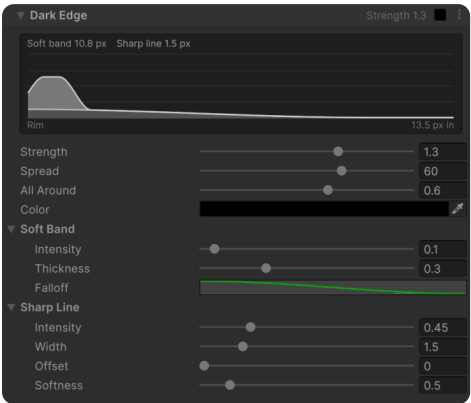


FIG. 18 The Dark Edge section.

Strength 0-2 · 1	Overall strength, scaling both layers. 0 turns it off.
Spread 0-90° · 60	How far the dark edge reaches around the rim before fading out.
All Around 0-1 · 0.6	Share kept all around the rim. 1 shades it evenly, like a glass outline.
Color Slate blue	The colour the edge shades towards. Black just darkens.
SOFT BAND	
Intensity 0-2 · 0.1	Strength of the band.
Thickness 0-1 · 0.3	Width of the band, as a share of the way to the middle.
Falloff Curve	Strength across the band.
SHARP LINE	
Intensity 0-2 · 0.45	Strength of the line.
Width · Offset · Softness 1.5 · 0 · 0.5 px	As for the highlights' sharp line.

Shadow

The shadow around each element, set up like a CSS `box-shadow` . It only shows outside the glass.

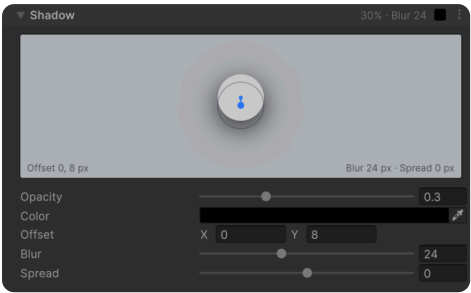


FIG . 19 The Shadow section.

Opacity 0-1 · 0.3	Darkness of the shadow. 0 turns it off.
Color Black	Colour of the shadow.
Offset px · 0, 8	How far it falls from its element. Positive y is down. Drag in the drawing to set it.
Blur 0-64 px · 24	How soft its edge is, like a CSS blur radius.
Spread -32-32 px · 0	How much bigger than its element it is. Negative makes it smaller.

NOTE

The values after each range are XGlass's built-in values, which a section's menu resets to. The shipped DefaultGlassProfile has its own: a thicker rim (0.75), no blur and stronger dispersion (0.2).

Surface, light and colour.

Three profile features go beyond bending light: a surface texture, an environment to reflect, and colour that flows inside the glass.

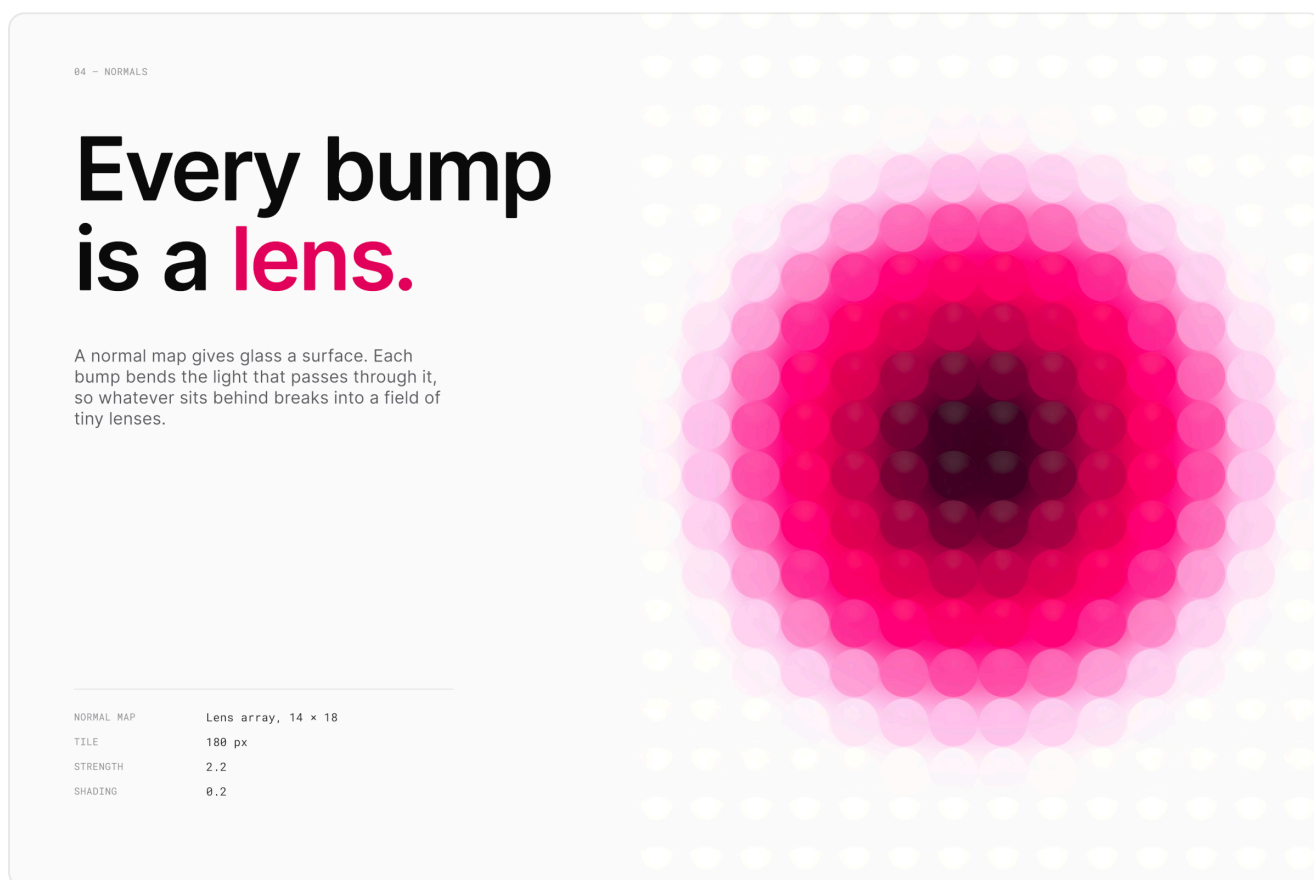


FIG. 20 A lens-array normal map over a colour blob, from the demo's Normals page.

5.1

Normal maps

A normal map gives glass a surface, like hammered or frosted glass. It's wrapped over the glass's dome like a decal on a ball, so it crowds together towards the rim, and its bumps tilt the surface itself: they bend what's seen through the glass and break up the reflections, the glint and the rim lighting.

With **Shading** the bumps also catch the light themselves, brighter where they turn towards it and darker where they turn away. The bumps scale with the tile, so a bigger tile makes them bigger without flattening them.

Import the texture with **Texture Type: Normal map**. If it isn't, the profile shows a warning with a **Fix Now** button that re-imports it. The demo's `Normal Lens Array` and `Normal Reeded` textures are in `XGlass/Samples/Demo/Textures`.

5.2

Reflections

Give a profile an **Environment** and the glass reflects it, like real glass: faintly where it faces you, almost like a mirror at its very edge. The middle of a pane shows what's behind the viewer, the top of its rim the sky and the bottom the floor, as a Unity camera looking into the screen would see the same texture as its skybox.

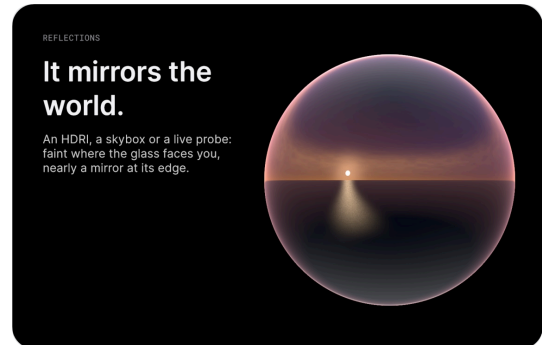


FIG. 21 An HDRI sunset reflected in a glass sphere.

- The environment is a cube map: a skybox's, a cube render texture (a realtime reflection probe's gives live reflections), or a 360° panorama or HDRI imported with **Texture Shape: Cube**. A panorama imported as an ordinary 2D texture shows a warning with a **Fix Now** button that re-imports it as a cube map.
- An HDR environment gives bright lights the punch of real studio lights. The demo has two, `Environment White Studio` and `Environment Sunset Sea`, in `XGlass/Samples/Demo/Textures`. The default profile has no environment, so it doesn't reflect.
- **Strength** scales the reflections, **Facing** sets how much the glass reflects where it faces you (real glass: about 4%), **Blur** reads softer mips as if the glass were less polished, and **Rotation** turns the environment.
- **Dome** sets how far in from the rim the surface curves for the reflections, the glint and the tint's fade, as a share of the way to the middle like rim width, and never less than the rim width. 1 domes the glass right to its middle.

5.3

Ink

Ink is colour inside the glass: a soft field that drifts and folds over itself, picking its colours from the profile's **Colors** gradient. Give a profile an Ink strength and the glass carries it, or set `ink` on single elements.

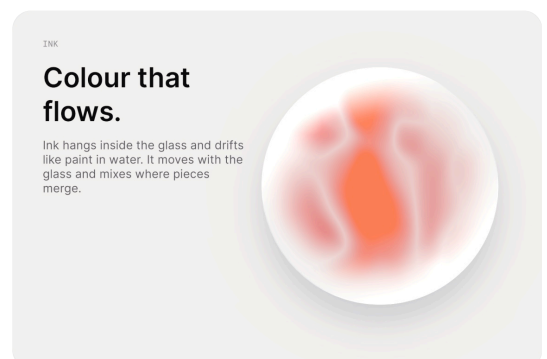


FIG. 22 Ink drifting inside a glass drop.

- The gradient's alpha is how much ink there is. Transparent keys leave clear glass between the swirls; opaque ones fill it. The colours cover what's seen through the glass like the tint's opacity does, under the rim lighting and reflections.

- **Size** is the swirls' size in px. The ink drifts out from each piece's centre, so it moves with the glass, and each piece starts from its own place in the field, so glass side by side doesn't swirl alike.
- **Flow** is how fast it moves; 0 holds it still. Flowing ink changes the glass every frame, so that glass is redrawn every frame, in UI Builder too.
- **Fade** gathers the ink in the middle: how far in from the rim it fades in, as a share of the way to the middle.
- In a group each piece keeps its own ink, and where two melt together their inks mix across the bridge.

Any outline.

A Glass Shape gives glass the outline of an image. Its rim, highlights, dark edge, refraction and shadow all follow the outline, and in a group it melts into the other glass like a rounded box would.

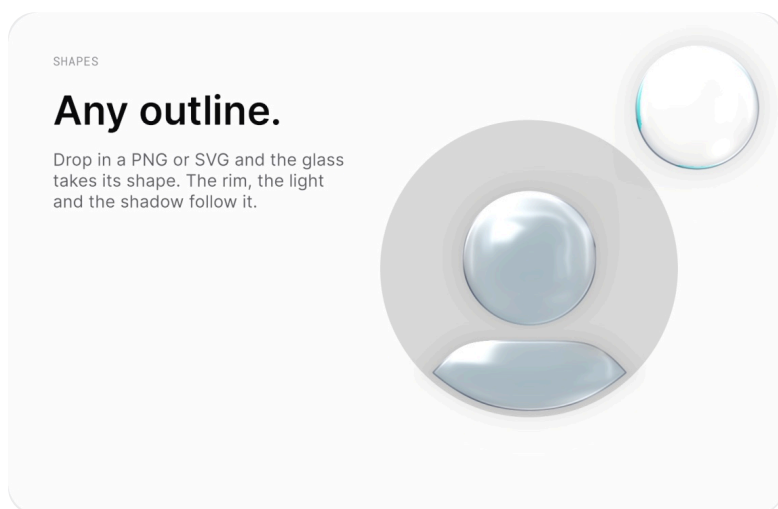


FIG. 23 A user icon baked from an SVG, from the demo's More page.

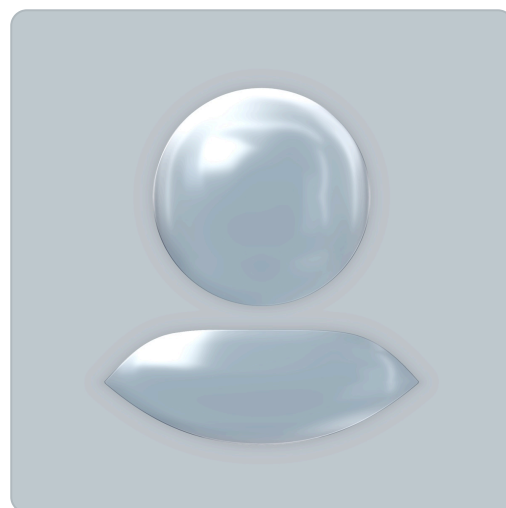


FIG. 24 The same shape with a clear, domed profile.

6.1

Making a shape

- 01 **Import a PNG or SVG.** A PNG gives its opaque part (or, without any transparency, its bright part). An SVG gives the area its shapes fill, strokes included.
- 02 **Select it** in the Project window and choose **Create** › **XGlass** › **Glass Shape**. This makes a Glass Shape next to the image and bakes it.
- 03 **Drag the Glass Shape** onto an XGlass's **Shape Asset** field in UI Builder.

The shape is stretched to fill the element's box. Give the element the shape's proportions, which the Glass Shape's Inspector shows as its aspect ratio, so the shape isn't distorted. Clicks only land on the glass inside the outline.

The shape bakes again whenever its image or a setting changes. A build only includes the baked shape, not the image.

Shape settings

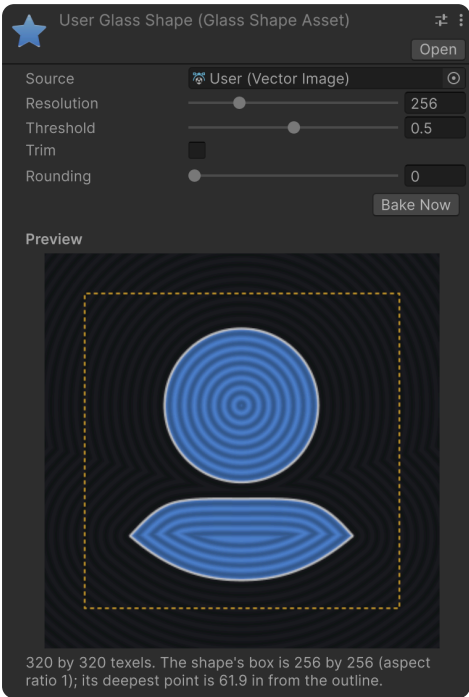


FIG. 25 A Glass Shape's Inspector. Blue is glass, white is the outline, and the yellow frame is the box that fills the element.

Source PNG or SVG	The image it bakes from. Any texture format works.
Resolution 32–1024 · 256	Texels across the shape's longer side. Higher keeps finer detail and sharper corners; 256 suits most icons.
Threshold 0.01–0.99 · 0.5	The opacity where the outline runs, for images with soft edges. 0.5 follows an anti-aliased edge.
Trim On	On: the glass fits the image's opaque part, which is how UI Toolkit draws an SVG by default. Off: the glass fills its box with the image's whole canvas, transparent margins included.
Rounding 0–0.1 · 0	Rounds the outline's corners, as a share of the shape's size. It also removes parts and closes gaps narrower than twice the radius.
Bake Now Button	Bakes the shape again. It also bakes by itself whenever its image or a setting changes.

Good to know

- Rim width is a share of the shape's deepest point, so 1 curves it all the way in. Parts narrower than twice the rim are curved all the way across, so a thin star arm bends the light everywhere. A lower `rim-width` keeps them flatter.
- Where the outline turns a sharp corner, the light bends smoothly around it rather than creasing.
- SVG shapes need Unity's Vector Graphics module (**Package Manager** › **Built-in** › **Vector Graphics**), which new projects include. PNGs work without it.
- Glass with a Glass Shape outside a group draws itself the way a group does, with a layer that reaches past it for the shadow. Keep its `overflow: visible` (the default) so the shadow isn't cut off.

One value, one element.

A profile sets every value. Override single values for one element in UI Builder's Inspector, or for many elements with USS. Anything not overridden comes from the profile.

7.1 In UI Builder

Select a glass element. Under **Attributes**, the **Glass**, **Normal Map**, **Color**, **Ink**, **Highlights** and **Dark Edge** groups list every value with a slider, in the Glass Profile's order and with its labels.

Change one to set it on this element: like UI Builder's own style fields, it gets a **white bar** on the left. Right-click it and choose **Unset** to hand it back to USS and the profile.

A field without the bar shows the value the element draws with, from its USS rules or its profile. Change it to set it on the element from there.

The shadow only takes USS and the profile.

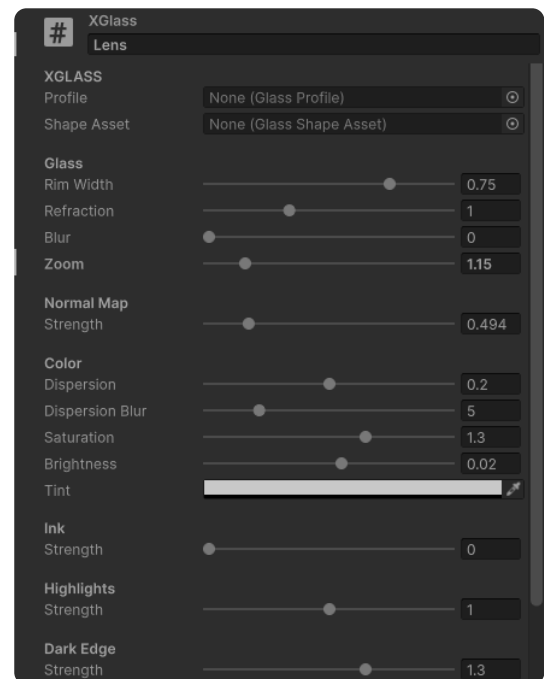


FIG. 26 An XGlass in UI Builder's Inspector. **Zoom** is set on this element (white bar); the rest come from its profile.

7.2 In UXML

A set value is a plain attribute:

```
<XGlass.XGlass zoom="1.3" dispersion="0" tint="#BFE3FF00" />
```

Lengths can be written with or without `px`, in attributes and in USS.

7.3

In USS

USS custom properties override values per class, or per element in an inline style:

```
/* All buttons heavier */
.my-button { --glass-refraction: 1.8; --glass-rim-width: 0.6; }
```

```
<XGlass.XGlass style="--glass-blur: 0px;" />
```

UI Builder has no fields for USS custom properties, so a shared look across many elements is easiest to edit as a profile.

7.4

What wins

From highest to lowest:

01	The element's own values	Set in the Inspector, in UXML attributes or from code.
02	The element's inline style	<code>style="--glass-blur: 0px;"</code>
03	USS rules	More specific rules win.
04	The element's profile	Its Profile field.
05	The project's default profile	Project Settings › XGlass .

7.5

USS property reference

PROPERTY	UNIT	MEANING
--glass-rim-width	0–1	Width of the curved rim, as a share of the way from the edge to the middle. 0 makes the glass flat, 1 curves it right to the middle.
--glass-refraction		How far the rim bends the light, in rim widths.
--glass-zoom		Magnification. 1 leaves it unscaled.
--glass-blur	px	Blur behind the glass.
--glass-dispersion		How much more red bends than blue at the rim.
--glass-dispersion-blur	px	Blur on the colour fringes only.
--glass-saturation		Colour saturation. 1 leaves it unchanged.
--glass-brightness		Brightness added.
--glass-tint	colour	Colour of the glass itself; white doesn't tint. Its alpha fills the glass with the colour: <code>rgba(191, 227, 255, 0)</code> tints light blue and stays clear, alpha 1 is solid.
--glass-highlight		Overall highlight strength.

PROPERTY	UNIT	MEANING
<code>--glass-dark-edge</code>		Overall dark-edge strength.
<code>--glass-normal-strength</code>	0-3	How far the normal map's bumps bend the light. 1 takes the map as it is, 2 doubles its bumps.
<code>--glass-ink</code>	0-1	How much the profile's ink colours the glass. 0 turns it off.
<code>--glass-shadow-opacity</code>		Shadow darkness. 0 turns it off.
<code>--glass-shadow-color</code>	colour	Shadow colour.
<code>--glass-shadow-offset-x</code> <code>--glass-shadow-offset-y</code>	px	Shadow offset. Positive y is down.
<code>--glass-shadow-blur</code>	px	Shadow softness, like a CSS blur radius.
<code>--glass-shadow-spread</code>	px	How much bigger than the glass the shadow is.
<code>--glass-merge-distance</code>	px	On an XGlassGroup: how close glass must be to merge.

OLDER NAMES

Blur used to be called frost. UXML written with `frost` and `dispersion-frost`, and USS with `--glass-frost` and `--glass-dispersion-frost`, still works. In USS the new names win where both are set.

Glass from code.

Everything you can do in UI Builder, you can do from C#: create glass, give it a profile or a shape, override single values and change profiles at runtime.

8.1

Creating glass from code

The glass element has the same name as its namespace, so after `using XGlass;` C# still reads `XGlass` as the namespace: write `XGlass.XGlass`, or add `using Glass = XGlass.XGlass;` and write `Glass`. Everything else, such as `XGlassGroup` and `GlassProfile`, works with the `using` alone.

```
using XGlass;

// A lens. USS: .my-lens { width: 200px; height: 200px; border-radius: 50%; }
var glass = new XGlass.XGlass { profile = myProfile };
glass.AddToClassList("my-lens");
root.Add(glass);

// Take a Glass Shape's outline, or bake one from a texture or SVG text.
glass.shapeAsset = myGlassShape;
var star = GlassShapeAsset.Create(starTexture);
var logo = GlassShapeAsset.CreateFromSvg(svgText, 512);
float aspect = star.aspectRatio; // width over height of the shape

// Every element using the profile updates.
myProfile.blur = 6f;

// This element only, over USS and its profile.
glass.zoom = 1.3f;
glass.zoom = GlassOverride.none; // back to USS and the profile
glass.ClearOverrides(); // every value it set
bool own = glass.zoom.enabled; // whether it sets its zoom itself
```

Group glass the same way: add XGlass elements to an `XGlassGroup` and set its `mergeDistance`.

```
var group = new XGlassGroup { mergeDistance = 32f };
group.Add(new XGlass.XGlass());
group.Add(new XGlass.XGlass());
root.Add(group);
```

API reference

XGlass

profile	The Glass Profile it draws with; null uses the project's default.
shapeAsset	A Glass Shape whose outline it takes, or null for its rounded box.
rimWidth, refraction, blur, zoom, normalStrength, dispersion, dispersionBlur, saturation, brightness, ink, highlight, darkEdge	Values the element sets for itself, as <code>GlassOverride</code> : assign a float to set one, <code>GlassOverride.none</code> to hand it back to USS and the profile.
tint	The tint as a <code>GlassColorOverride</code> : assign a <code>Color</code> , or <code>GlassColorOverride.none</code> .
ClearOverrides()	Unsets every value the element set.
resolvedProfile	The profile it draws with.
group, merged	The group it belongs to, and whether that group draws it.
imageShaped	Whether it takes a Glass Shape's outline.

XGlassGroup

profile, resolvedProfile	The group's own profile, and the one it draws with (its own, or its first glass's).
mergeDistance	A <code>GlassOverride</code> ; <code>resolvedMergeDistance</code> is the distance in use.
members	The glass the group melts together.
maxMergedGlass, defaultMergeDistance	16 pieces per group, and 16 px.

GlassProfile

rimWidth, dome, refraction, zoom, blur, ...	Every setting of chapter 4 as a property. Setting one updates every element using the profile.
NotifyChanged()	Call after changing a falloff curve's or the ink gradient's keys in place.

GlassShapeAsset

Create(texture, resolution = 256) CreateFromSvg(svg, resolution = 256)	Bake a new shape at runtime. Runtime textures don't need to be readable.
Bake(texture), BakeSvg(svg)	Bake an existing shape again with its current settings.

resolution, threshold, trim, rounding	The bake settings of chapter 6.
aspectRatio, baked	Width over height of the shape's box, and whether it holds a shape.

XGlassSettings

XGlassSettings.instance.defaultProfile	The project's default profile; setting it redraws every glass element that uses it.
XGlassSettings.DefaultProfile	The default profile in use: the project's, or one with the built-in values when none is set.

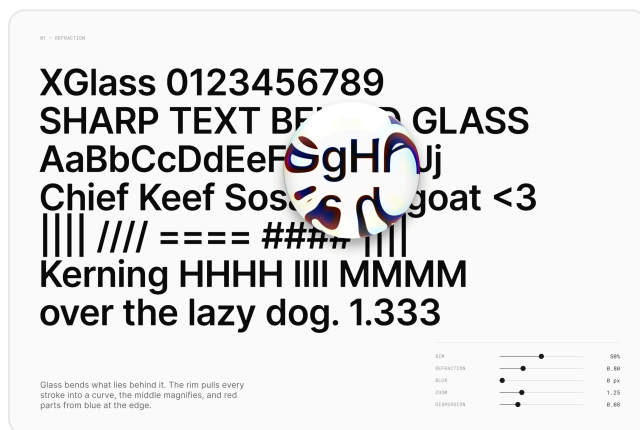
A tour in five pages.

The demo in `XGlass/Samples/Demo` tours the features page by page, from a sidebar drawn in thin lines where a glass pill slides over the page you pick. Open `XGlassDemo` and press Play.

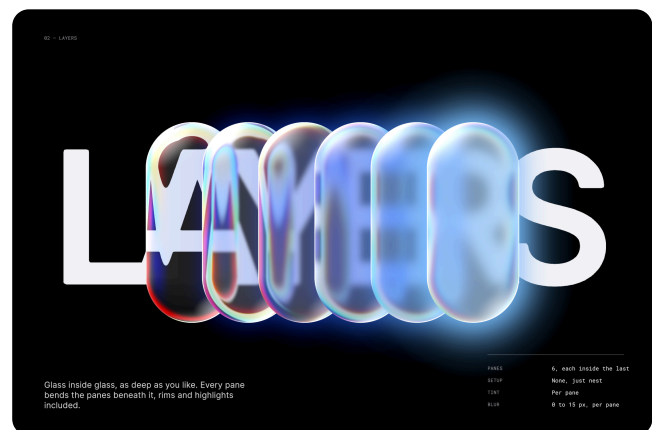
9.1

The five pages

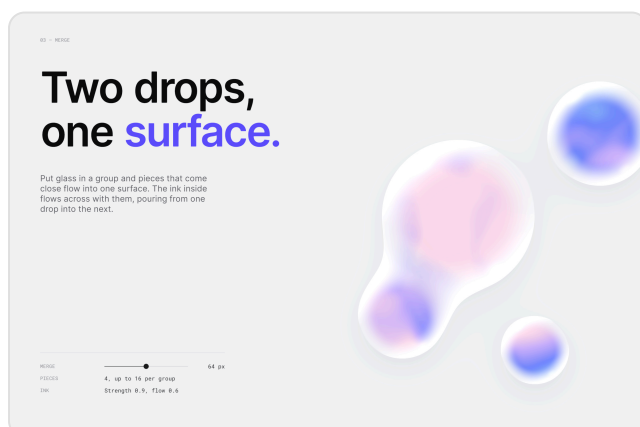
Each page's text rises in line by line as the page opens. Keys **1–5** (or the arrow keys) switch pages, and **H** hides the tips for clean screenshots.



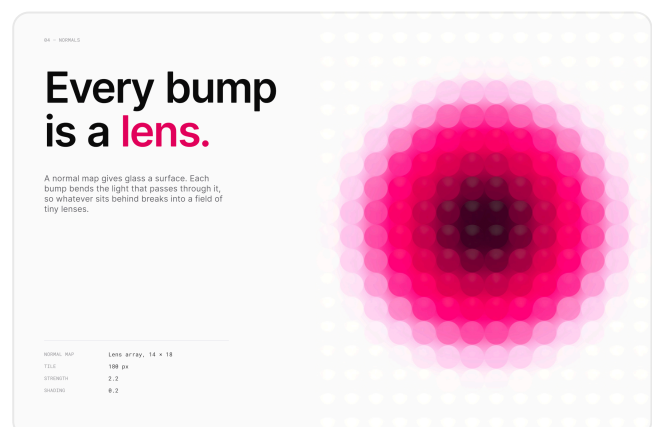
01 REFRACTION A clear lens you drag across sharp type, with live sliders for rim, refraction, blur, zoom and dispersion.



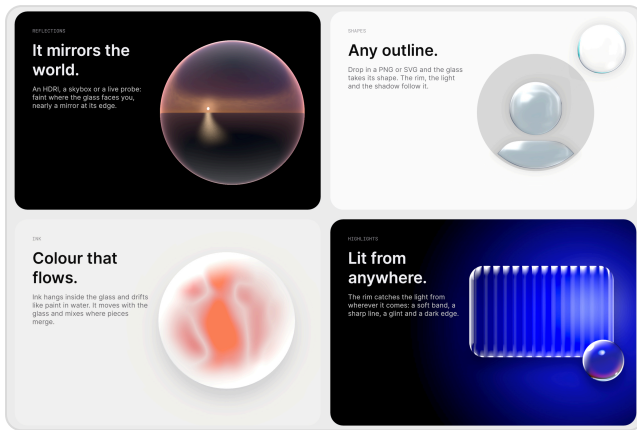
02 LAYERS Nested glass fanned over big type. Drag to fan the layers.



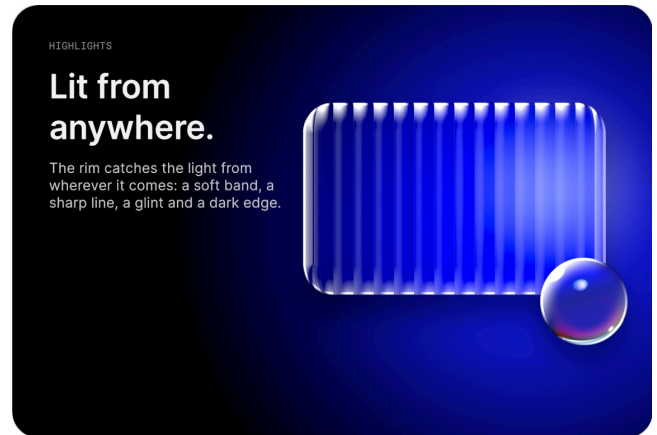
03 MERGE Drops with flowing ink that melt together, with a slider for how close they must come.



04 NORMALS A lens-array normal map over a colour you drag.



05 MORE Reflections, shapes, ink and highlights that follow the pointer.



05 MORE – HIGHLIGHTS Reeded glass lit from wherever the pointer is.

9.2

Liquid dragging

The demo drags glass with `LiquidDrag`. The glass follows the pointer with a little lag, stretches along its direction of travel and swells while held, like a drop of liquid.

```
// Every element with the "draggable" class in a tree:
LiquidDrag.MakeDraggable(root);

// Or a single element:
element.AddManipulator(new LiquidDrag());
```

- The drag moves the element with its translate, rotate and scale, and only brings absolutely placed elements to the front, since reordering one in the layout's flow would move it and its siblings.
- Give an ancestor the `draggable-bounds` class to keep dragged elements inside it: an element's box as laid out can't leave the ancestor's, whatever the swell and stretch do.
- `LiquidDrag` and `TextReveal` live in the samples' `XGlass.Samples` namespace. Copy them into your project if you delete the samples.

Good to know.

How XGlass fits UI Toolkit's rules, where it leans on Unity internals, and what to check when glass doesn't look right.

10.1

Limitations

Corners

Corners are rounded as UI Toolkit draws them: each reaches at most half the width across and half the height down, and a percentage is taken of the width across and the height down. So a radius bigger than half the box's shorter side makes the corner oval, and `border-radius: 50%` or `9999px` on a wide box gives an ellipse, not a pill. For a pill, use half the height in px.

Zoom and transforms

Glass looks the same at any UI Builder zoom: its lengths (highlight lines, shadow, blur, normal map, merge distance) are points of its parent, so they scale with the canvas, or with any container scaled by a transform. The glass's own transform doesn't scale them. Rim width, dome and soft band thickness are shares of the glass's size on screen, so they follow every transform, its own included, and its corners follow its own transform too, like UI Toolkit's clip.

Dispersion blur

Dispersion blur blurs the backdrop with two extra passes and works the colour fringes out on that. Glass in a group, or with a Glass Shape, takes two more when any of its pieces is blurred as well, and there the blur stops at the glass's outline: blurred glass with a zoom below 1 (only reachable from USS or code) shows what it draws in from just past its edge unblurred.

Unity internals

XGlass uses four Unity features that aren't public. If a future Unity version removes one, XGlass falls back and warns once:

- Groups and dispersion blur read the unfiltered content through a pass input name. Without it, merged glass blurs its surroundings too and glass samples its dispersion blur around each pixel, which a wide dispersion blur breaks into separate coloured copies of sharp detail behind it.
- Glass reads an element's clip area to place itself when it hangs past a ScrollView, an element with `overflow: hidden` or UI Builder's canvas. Without it, that glass draws offset.

- Glass asks UI Toolkit how it resolves percentage radii. Without that, a percentage radius is read as points.
- Glass checks how the target it draws into stores colour as UI Toolkit captures the backdrop. Without that, glass drawn into a float target (URP with HDR on) comes out a level or two darker than its surroundings, which shows as a faint edge around a group on a flat background.

Colour and memory

- UI Toolkit blends in linear colour space in linear projects, so translucent shadows look lighter than the same alpha in CSS. XGlass compensates for mesh shadows.
- UI Toolkit gives each backdrop filter textures the size of its element's whole box, even the part that's clipped away. A group keeps its layer to the visible part, but a single very large XGlass element at a high UI Builder zoom uses a lot of GPU memory.
- In play mode with no Game view in front (another window maximized, UI Builder for one), Unity keeps drawing the runtime panels without showing them, and the GPU memory behind their filters piles up until the editor goes black or crashes. That's a UI Toolkit bug that Unity's own filters have too. XGlass hides its runtime panels with glass while no Game view shows, and brings them back as soon as one does. Filters of your own on those panels are still affected.

10.2

Troubleshooting

PROBLEM	WHAT TO DO
Glass is invisible, or the console says the filter definition is missing	Run Tools > XGlass > Repair Setup . It recreates the filter assets, the settings asset and the default profile, if they're missing. A recreated default profile has XGlass's built-in values.
A USS override has no effect	Check the property name, and that the rule matches the element. UI Builder's Matching Selectors list helps.
Glass in a group isn't merging	Check <code>merge-distance</code> is above 0, and that no other XGlass sits between the glass and the group: only glass directly inside the group merges.
Merged glass is cut off at the group's edge	Give the group <code>overflow: visible</code> , the default.
A Glass Shape looks stretched	Give the element the shape's proportions: its Inspector shows the aspect ratio.
The glass doesn't reflect	Give its profile an Environment that's a cube map. A panorama imported as a 2D texture shows a Fix Now button.
A normal map bends the light the wrong way	Import it with Texture Type: Normal map, or press Fix Now in the profile.
A pill comes out as an ellipse	Use half the height in px for <code>border-radius</code> , not <code>50%</code> .

XGlass

GLASS FOR UI TOOLKIT - VERSION 1.0.0

ANNEA