PCA 3D ADVENTURE STUDIO

Cameras, Lighting and Rendering

A practical guide to shots, movement, shadows, glow, outlines and reflections

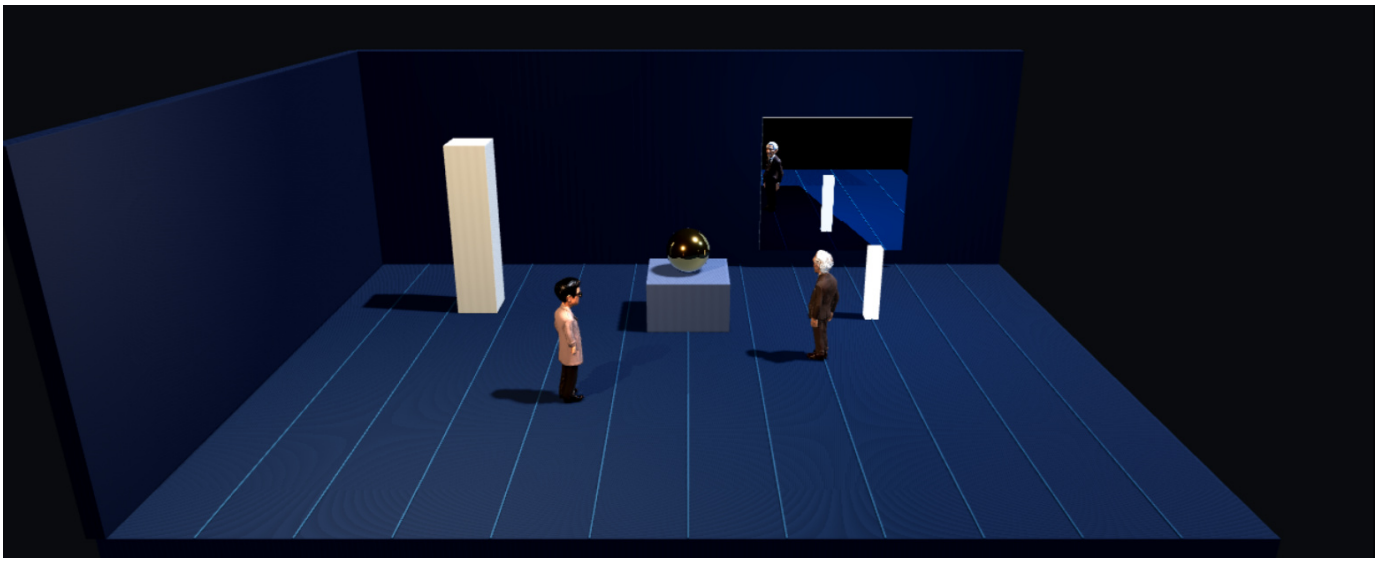

Figure 1. A small room built in PCA 3D. The same room is used throughout the guide so changes are easy to compare.

IN-DEPTH TUTORIAL • VERSION 0.86.3.4

Start with a useful room view. Build up to tracking shots, conversations, atmospheric lighting and a finished look. Each lesson explains what to change, what you should see and what to check when the result is wrong.

Before you start

You can work through this guide in an existing game or in a new game made from the Primitives template. The Clockmaker template is useful when you want two animated characters and a more detailed room. Make a separate copy of any project you want to experiment with.

The screenshots come from the native PCA 3D editor and its running game preview. The demonstration room uses bundled primitives and the Clockmaker characters. The instructions apply to the desktop editor on Windows, macOS and Linux. Mouse instructions describe the action; use your platform’s equivalent modifier key where necessary.

You do not need to write code to use the main camera and lighting features. Later lessons show the same work in action lists, visual scripts and text scripts. The short JSON examples are advanced record edits, clearly identified as such.

What to know	How it is used here
A name identifies the record on screen. An ID identifies it in actions.	Room View (wide) is a camera named Room View with the ID wide.
Apply Changes commits an Inspector edit. Save Project saves the game.	Apply, save, then Test Scene. A visible draft is not proof that the running game has received the edit.
The editor view and the game camera are separate.	Moving the free editor camera helps you work. Copy View to Camera records that composition as a game shot.
A renderer is a drawing system, not a quality slider.	Changing it may require a full editor restart. The running preview tells you which renderer is actually active.

This guide describes the controls exposed by PCA 3D 0.86.3.4. It does not assume that every feature in Godot or in the upstream Phantom Camera add-on has a matching PCA control. No project or practice files are required.

Contents

Before you start . . . 2

01 Find the right controls . . . 7

02 Position, target and the three axes . . . 8

03 Create and name your first shots . . . 9

04 Compose with Copy View to Camera . . . 10

05 Choose a useful lens . . . 11

06 Perspective and orthographic views . . . 12

07 Clipping, aspect ratio and safe framing . . . 13

08 Turn Advanced Camera on deliberately . . . 14

09 Fixed: a reliable authored shot . . . 15

10 Follow: stay with one target . . . 16

11 Group: keep the conversation together . . . 17

12 Tune group framing without guesswork . . . 18

13 Framed: move at the screen zone edges . . . 19

14 Path: a camera route tied to a target . . . 20

15 Edit path points carefully . . . 21

16 Targets, offsets and a separate aim . . . 22

17 Cuts, blends and easing . . . 23

18 Camera priorities and room zones . . . 24

19 Respect room bounds without cropping the action . . . 25

20 The original Dynamic Follow controls . . . 26

21 Dynamic Follow: speed, look-ahead and limits . . . 27

22 A two-shot conversation in action lists . . . 28

23 Build the same flow with visual nodes . . . 29
24 Camera events on a cutscene timeline . . . 30
25 Manual moves and orbiting . . . 31
26 Shake, fades and camera ownership . . . 32
27 Depth of field: focus on the subject . . . 33
28 Focus distances and transitions . . . 34
29 First-person view and headset camera ownership . . . 35
30 2D Stage and older 2D camera systems . . . 36
31 2D Stage framing and scrolling controls . . . 37
32 Camera scripts in all four languages . . . 38
33 Lua, BASIC and partial behavior updates . . . 39
34 Four jobs that often get called lighting . . . 40
35 Add and aim a light . . . 41
36 Point light: a local pool of light . . . 42
37 Spot light: aim a controlled cone . . . 43
38 Directional and hemisphere lights . . . 44
39 Ambient light and room brightness . . . 45
40 Build a warm key and cool fill . . . 46
41 Skyboxes and sky ambient . . . 47
42 Advanced fields through Raw JSON . . . 48
43 Make shadows work in the first place . . . 49
44 Choose shadow quality sensibly . . . 50
45 Bias, normal bias and detached shadows . . . 51
46 Directional shadow cascades . . . 52
47 Object shadow settings and contact patches . . . 53

48 Emission: make the surface look bright . . . 54
49 Scene glow: add the halo . . . 55
50 Advanced glow thresholds and spread . . . 56
51 Object bloom and real surrounding light . . . 57
52 Switch lights and glow from events . . . 58
53 Decorative RGB bulbs . . . 59
54 RGB lighting, animation and cost . . . 60
55 Lens flares: a visible light source . . . 61
56 God rays and cinematic visible shafts . . . 62
57 Tune fog scattering and visible beams . . . 63
58 Ordinary depth fog and height fog . . . 64
59 A flashlight reveal and 2D lighting . . . 65
60 Surface shine, room capture and mirrors . . . 66
61 Metallic, roughness and reflection strength . . . 67
62 Set up a room reflection capture . . . 68
63 Create a mirror that reflects the character . . . 69
64 Mirror orientation, blur and performance . . . 70
65 Diagnose a missing reflection . . . 71
66 Hover outlines and attention feedback . . . 72
67 Whole-scene silhouette outlines . . . 73
68 Edges Only: what it draws . . . 74
69 Choose one outline language for the game . . . 75
70 Choose the renderer for the actual game . . . 76
71 Restart the renderer, not just the preview . . . 77

72 Read Rendering details . . . 78
73 Feature availability at a glance . . . 79
74 Quality policies: Authored, Balanced and Performance . . . 80
75 Forward+ Enhanced: enable only what you need . . . 81
76 Indirect lighting and static geometry . . . 82
77 Screen-space reflections and ambient occlusion . . . 83
78 Tone mapping and cinematic grading . . . 84
79 Screen shaders and retro rendering . . . 85
80 Zoom blur for a short dramatic beat . . . 86
81 Objects, regions, overlays and particles . . . 87
82 Workshop recipe: a finished gameplay view . . . 88
83 Conversation recipe: two actors and a return . . . 89
84 Window-light recipe: beams without white fog . . . 90
85 Glowing-prop recipe: bulb, halo and spill . . . 91
86 Mirror-room recipe and a proper movement test . . . 92
87 A restrained horror or noir room . . . 93
88 Study the Clockmaker camera examples . . . 94
89 When the camera does something unexpected . . . 95
90 When the lighting looks wrong . . . 96
91 A practical performance pass . . . 97
92 Test before you distribute . . . 98
93 Advanced Camera field reference . . . 99
94 Group, frame and path field reference . . . 100
95 A short rendering glossary . . . 101
Sources and credits . . . 102

01 Find the right controls

Most of the work happens in three places. Select a camera or light in Scene Objects to edit that particular record. Select the room to edit the environment around everything in it. Open Global Settings for defaults that affect the whole game.

Place	What belongs there
Camera Inspector	Position, aim, projection, lens, clipping, Dynamic Follow, Advanced Camera and depth of field.
Light Inspector	Light kind, color, intensity, range, shadow switch, flare, shafts and transform.
Room Inspector → Camera & View	Default camera, camera pan bounds, room tracking override and default-camera focus.
Room Inspector → Lighting & Fog / Post Process	Sky, brightness, fog, shadow quality, scene glow, screen effects and whole-room outlines.
Global Settings → General	Renderer, quality policy, ambient light, tone mapping and Enhanced Rendering.
Object Inspector → Visual Effects / Pointer Feedback	Material shine, mirror mode, shader stack, hover outline, bloom or tint.

If a dock is missing, open Panels and restore Inspector or Scene Objects. Drag the divider beside the Inspector to give it more width. The tab arrows reveal headings that do not fit. A long tab may also have its own scrollbar.

> Use Test Scene to judge the final picture. The editor has its own working view and helpers. A camera thumbnail shows the authored shot; it does not demonstrate live tracking.

02 Position, target and the three axes

A camera needs a place to stand and a place to look. Position is where the camera stands. Look Target is the point it faces. These are separate XYZ coordinates. Moving one does not mean that the other should move by the same amount.



Figure 2. Position and Look Target are separate XYZ controls in Position & View.

Axis	Useful way to think about it
X	Left and right across a typical room.
Y	Height above the floor.
Z	Forward and backward through the room. The picture’s left and right depend on the camera’s angle.

For a first shot, try Position [0, 5.5, 11.6] and Look Target [0, 1, -1] in a room about twelve units wide. These are starting values for the room shown here, not a universal camera position. A character’s origin is commonly at its feet, so aim above zero if you want faces near the middle of the picture.

03 Create and name your first shots

1. Open Scene Designer and select the room you want to work on. In Scene Objects, expand Cameras.
2. Use Add to Scene → Camera to add a camera. Select it and open General in the Inspector.
3. Give it a clear name and a short stable ID. For example: Room View / wide, Conversation / conversation and Door Detail / door_close.
4. Keep Enabled at Runtime checked. Use Position & View to place and aim it. Click Apply Changes.
5. Select the room. In Camera & View, choose Room View (wide) as Default Camera. Apply Changes, save and Test Scene.

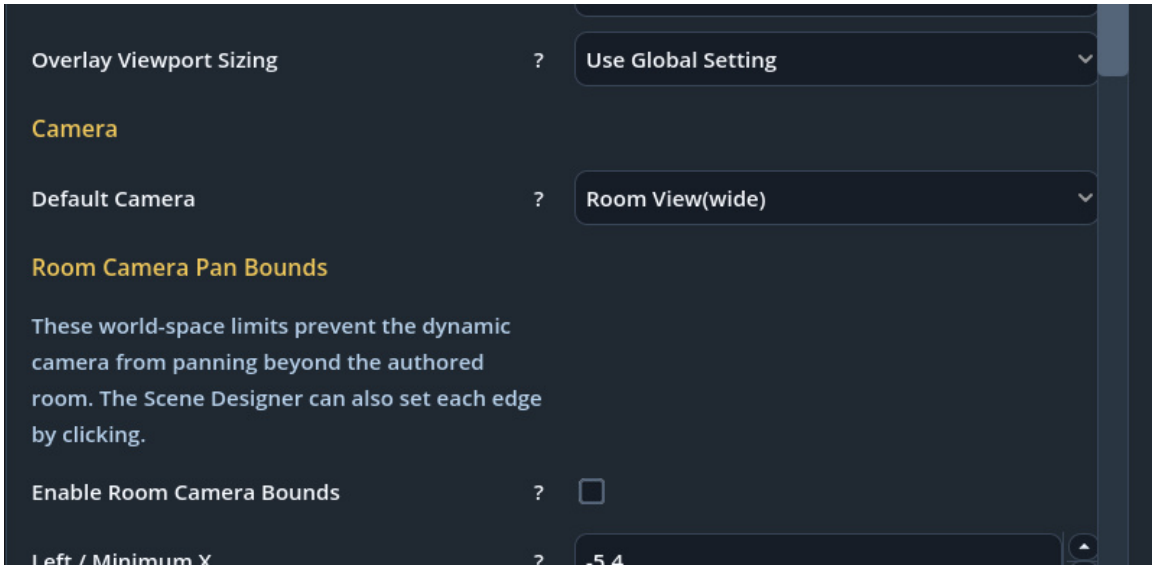

Figure 3. The room chooses its default shot separately from creating and selecting camera records.

A room’s default camera is the shot used when entering without an explicit alternative. Creating a new camera does not automatically make it the default. Nor does selecting its icon in the editor switch the running game to it.

Locked in Scene Designer protects a camera against accidental manipulation. Visible in Scene Designer controls its authoring visibility. Those checkboxes have different jobs from Enabled at Runtime.

> Keep IDs unique within the room. A camera action looks for its camera in the current room. The same ID in a different room does not make that other room’s shot available.

04 Compose with Copy View to Camera

It is often easier to find a good angle by looking around the room than by typing six numbers. Copy View to Camera transfers the free editor view’s position, aim and field of view into an existing camera.

1. Switch the Scene Designer to its 3D view. Right-drag to look around; use the editor’s WASD fly controls, middle-drag to pan and the wheel to move closer or farther away. F frames the selected item.
2. Find a view that shows the walking area, important exits and enough space around the player.
3. Click Copy View to Camera in the Scene Designer toolbar. Choose the destination camera carefully.
4. Confirm the copy. Select that camera and inspect Position & View. Test Scene to check the game frame and HUD.

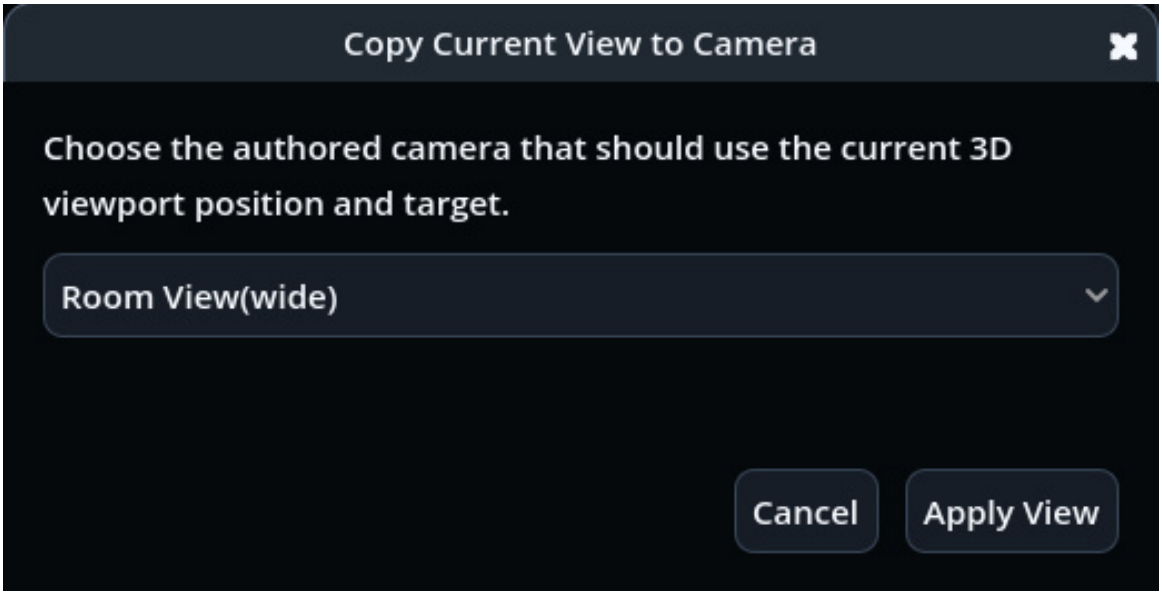

Figure 4. Copy the working view into the intended camera, rather than accidentally replacing another shot.

The copy records the view, not every behavior setting. It does not turn following on, choose targets or make the camera the room default. Set those separately. The editor viewport may have a different shape from the game, so always check the actual game frame.

05 Choose a useful lens

Field of View controls how wide a perspective camera sees. A smaller value gives a tighter view. A larger value includes more of the room and exaggerates near-to-far spacing. Very wide shots can stretch people near the edges.

Starting point	Try it for
35-45 degrees	Conversations and detail shots, with enough camera distance to avoid crowding faces.
45-60 degrees	A readable room view in a modest interior.
60-75 degrees	A broad establishing view or a deliberately spacious first-person room.

Move the camera to change perspective. Change FOV to change how much fits in the frame. These can produce the same apparent subject size with a different sense of depth. If a door looks enormous beside a character, move farther away and use a tighter lens before changing the door’s scale.

1. Start around 46 degrees. Put the player near the closest and farthest walkable points.
2. Check that the player remains recognizable and that exits stay visible.
3. Leave space for the HUD, dialogue and inventory. A face hidden behind a GUI is still hidden even if the world framing is technically correct.

> Use the same or similar lenses for a conversation unless you deliberately want a strong change in perspective. A sudden lens change can feel like the room changed size.

06 Perspective and orthographic views

Perspective makes distant objects appear smaller. Orthographic keeps their apparent size steady as they move nearer or farther from the camera. Orthographic can suit isometric adventures, diagram-like scenes and some 2D Stage arrangements.

Control	Perspective	Orthographic
Framing control	Field of View plus camera distance.	Orthographic Size, which controls the view’s vertical span.
Depth appearance	Parallel lines converge with distance.	Parallel lines remain parallel.
Group auto-framing	Adjusts camera distance.	Adjusts view height.
What moving the camera does	Changes both viewpoint and apparent scale.	Changes viewpoint; use Size to change framing scale.

1. Select the camera. In General, choose its Projection.
2. For orthographic, begin with Size 10. Reduce it to zoom in or increase it to show more of the room.
3. Test a character crossing the room. Check the edges of the floor, background and any walk-behind pieces.

Do not use orthographic just to repair an awkward perspective lens. Choose it because its appearance suits the game. A 2D Stage already constrains its own orthographic main camera; later lessons explain those limits.

07 Clipping, aspect ratio and safe framing

Near Clip removes geometry closer to the camera than the specified distance. Far Clip removes geometry beyond its distance. Neither changes navigation or collisions. A wall that disappears as you approach it may be crossing Near Clip; a distant building that vanishes may be beyond Far Clip.

Setting	Practical starting point
Near Clip	0.05 for a normal room. Lower it cautiously if a close-up cuts through a face or prop.
Far Clip	100-120 for the example room. Use enough to include the scene, rather than an unnecessarily enormous value.
Project resolution	Choose the intended game aspect ratio before polishing shots.
Room viewport override	Recheck every important shot if the room uses a different viewport shape.

A wider display can reveal the edges of a set that were hidden at another aspect ratio. Group framing uses the live viewport shape, but a fixed composition still needs checking. Letterbox bars cover the picture; they do not move actors away from the edges.

If the camera is inside a wall, lowering Near Clip may hide the immediate symptom without repairing the shot. Move the camera out of the wall. Small Near Clip values paired with huge Far Clip values can also make depth precision less reliable.

08 Turn Advanced Camera on deliberately

1. Select a camera and open Advanced Camera.
2. Check Enable Advanced Camera. Choose Camera Behavior.
3. Set targets and the settings used by that behavior. Click Apply Changes.
4. Make it the default camera or activate it with an action. Use Test Scene, then move the player.

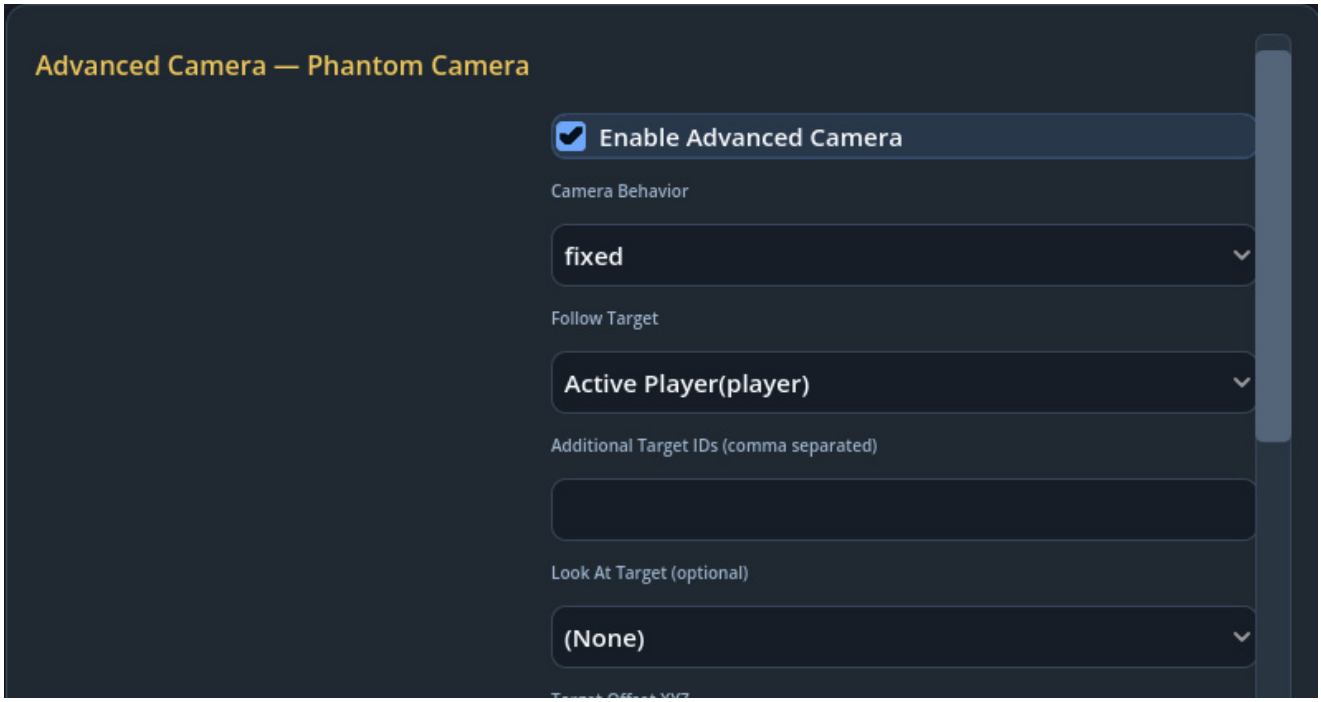

Figure 5. Advanced Camera is optional. Choose a behavior and targets for the authored shot.

To disable the optional system, uncheck Enable Advanced Camera, apply and restart Test Scene. The original camera behavior remains available. At runtime, Set Camera Behavior with enabled false turns the advanced behavior off for that room session.

The integration is powered by Phantom Camera. The ordinary Position & View settings still define the authored shot’s angle, lens and distance. Advanced Camera adds the five behaviors described next. Enabling it does not turn a poorly aimed shot into a good one.

> Use one tracking system for a shot while learning. Begin with the older Dynamic Follow disabled or inherited from an off default. Advanced behavior owns the camera while active; mixing ownership changes makes troubleshooting harder.

09 Fixed: a reliable authored shot

Fixed keeps the camera at its authored position and aim. Enable Advanced Camera with behavior fixed when you want the shot to take part in smooth transitions without following a character.

1. Create Room View (wide). Set its Position, Look Target and FOV.
2. Enable Advanced Camera and choose fixed. Set Default Camera Blend to 1 second and Blend Curve to sine.
3. Create Door Detail (door_close), also fixed. Aim it at a feature worth showing.
4. Activate Door Detail with a one-second blend, pause long enough for the player to notice the detail, then activate Room View.

A fixed camera is a good choice for a small adventure room because it gives the player a stable map of the space. It also makes clicks and exits easier to learn. Use close-ups when they reveal information or support a conversation.

Fixed means no automatic follow. An explicit Activate Camera, Camera Move or cutscene camera event can still change the view. If it moves unexpectedly, look for those actions before adjusting its position again.

> A cut is often clearer than a long journey through walls. Use duration 0 when the transition would expose the back of your set.

10 Follow: stay with one target

Follow moves with a target while keeping the camera-to-target direction established by the authored shot. It is useful for a long corridor, a large courtyard or a character crossing a room too large for one useful fixed view.

1. Choose follow. Set Follow Target to player to follow whoever the player currently controls.
2. Begin with Target Offset [0, 1.3, 0]. This tracks a point above the character’s feet.
3. Set Follow Damping to 0.2 seconds. Apply, activate the shot and walk across the room.
4. If the view lags too much, lower damping. Try 0 briefly to see the unsmoothed movement. Reintroduce a little damping for ordinary play.

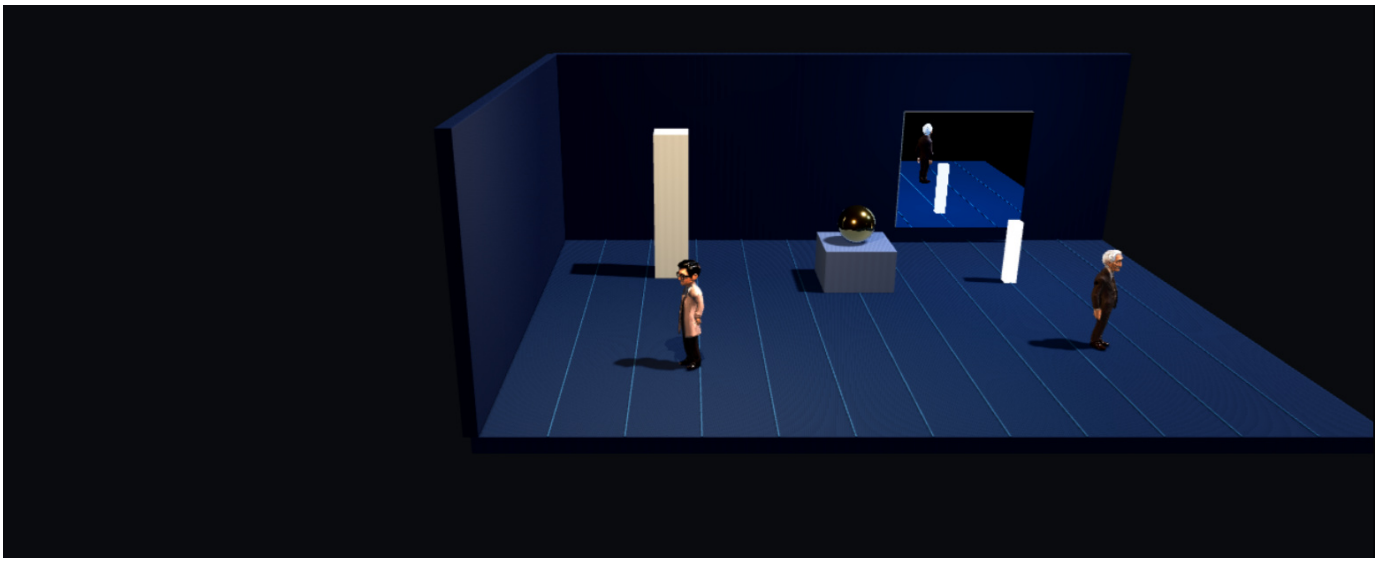

Figure 6. A live follow shot after the player moves. The camera moves with the target; the room view is no longer stationary.

Use an explicit object or character ID when the shot should remain tied to that subject after a player switch. Use player when it should follow the newly controlled character. Neither option automatically avoids walls or finds a clear line of sight.

11 Group: keep the conversation together

Group follows the center of several targets. With Automatically Frame Group on, it also adjusts distance, or orthographic size, so the targets fit. It is the useful starting point for a two-character conversation.

1. Create Conversation (conversation). Compose an angle that shows both actors without crossing the room’s walls.
2. Enable Advanced Camera and select group. Set Follow Target to player.
3. Enter elias in Additional Target IDs. For several targets, use comma-separated IDs such as elias,door.
4. Enable Automatically Frame Group. Try Minimum 3, Maximum 14 and Padding 1.3. Use damping 0.2.
5. Activate the camera. Move the actors closer together and farther apart. Check both heads and feet.

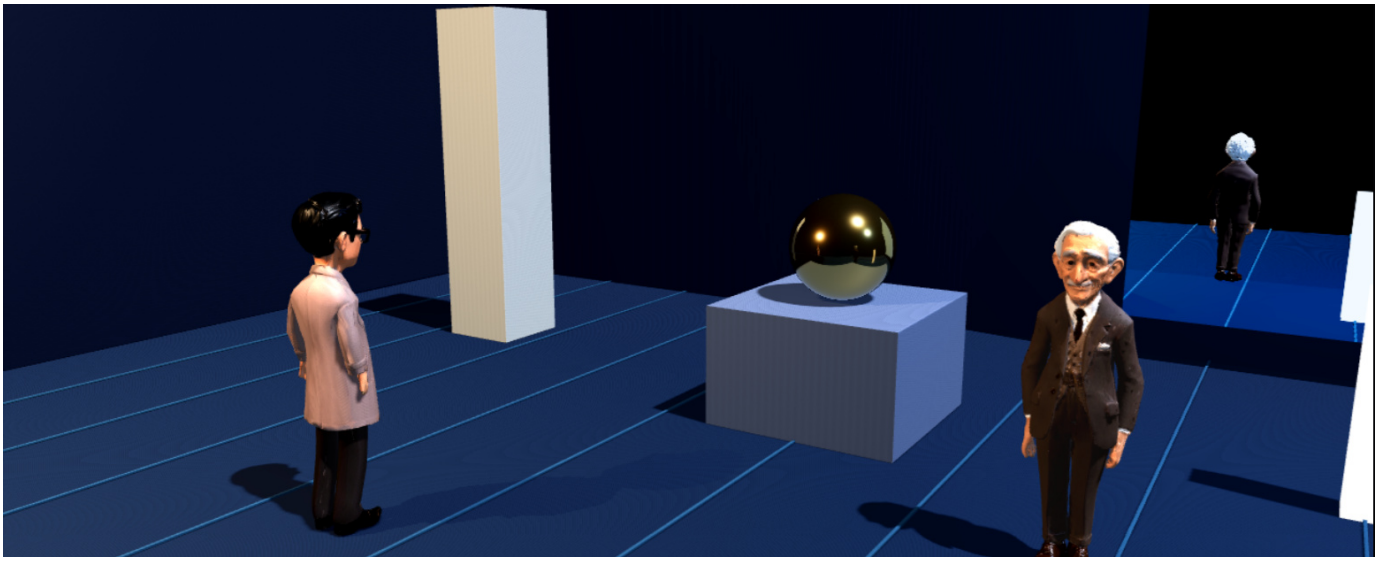

Figure 7. The group camera frames the player and Elias together. Its viewing direction comes from the authored shot.

> The group is made from the target list, not from everyone who happens to be visible. An omitted speaker can walk out of frame while the camera correctly frames the remaining targets.

12 Tune group framing without guesswork

Control	What to change and why
Automatically Frame Group	Turn it off when you want tracking around the center but a stable authored distance.
Minimum Group Distance / View Height	Prevents an uncomfortably close shot when the targets stand together.
Maximum Group Distance / View Height	Prevents an excessively distant view. Too low a maximum can leave separated actors outside the frame.
Group Framing Padding	Adds breathing room around the group. Start at 1.3; increase gently.
Target Offset Y	Raises the tracked area and contributes to the estimated character height. Start near chest height for human-scale actors.
Respect Room Camera Bounds	Bounds may stop the camera reaching the position needed to frame everyone. Disable for an unrestricted cinematic shot when appropriate.

First test with only two targets and a sensible maximum. Increase padding after the camera can fit them. Padding cannot rescue a maximum distance that is too restrictive. A narrow FOV needs more distance than a wide one to include the same spread of actors.

The automatic fit considers the viewport shape and an estimate of the actors’ height. It is not a full test of every hat, wing, carried prop or skeletal animation. Check unusually tall characters and wide gestures in Play.

13 Framed: move at the screen zone edges

Framed lets the player move within a central screen zone before the camera follows. This avoids moving the entire room every time the player takes a small step. It works well for an adventure where the camera should remain calm during local interactions.

1. Choose framed and Follow Target player. Start with offset [0, 1.3, 0].
2. Set Framing Zone Width to 0.4 and Height to 0.3.
3. Apply and activate the shot. Walk a short distance, then farther toward a picture edge.
4. Use a wider zone for more movement before following. Use a smaller zone for tighter tracking. Adjust damping after the zone feels right.

Width 0.4 means 40% of the screen width. Height 0.3 means 30% of its height. These are screen fractions between 0 and 1, not room units. This is different from the older Dynamic Follow Dead Zone, which is measured in world units.

A zone is a tracking rule, not a cage for the character. It does not stop walking and it does not change navigation. If the player cannot walk past a point, inspect the navigation and collisions separately.

14 Path: a camera route tied to a target

Path follows the part of an authored route nearest the moving target. It is useful beside a walkway or along a narrow set where a free follow camera would drift through walls. It does not automatically travel a timed rail from first point to last point.

1. Choose path and set Follow Target to player. Add at least two Camera Path Points.
2. Place the points in scene coordinates along the route the camera is allowed to use. For a simple straight route, try [-4, 3, 6], [0, 3, 6] and [4, 3, 6].
3. Use Look At Target player if you want the camera to aim at the player while its position stays on the route.
4. Apply, activate and walk across the room. Check how the camera behaves near every corner and at both ends.



Figure 8. The live camera stays on the authored route while its target moves across the room.

Point order defines the route’s segments. Three points in the wrong order can create an unwanted diagonal across the room. Add points where the permitted camera route needs to turn, not merely because the player route turns.

> For a timed opening move, blend between fixed cameras or use a timed Camera Move. Path mode responds to the target’s position; it is not the timeline itself.

15 Edit path points carefully

Camera Path Points has Add, Remove, Up and XYZ controls. Select the point you want to edit, set its coordinates, and use Up to correct its place in the sequence. The stored form is an array of XYZ arrays.

```
[[ -4.0, 3.0, 6.0 ],
 [  0.0, 3.0, 6.0 ],
 [  4.0, 3.0, 6.0 ]]
```

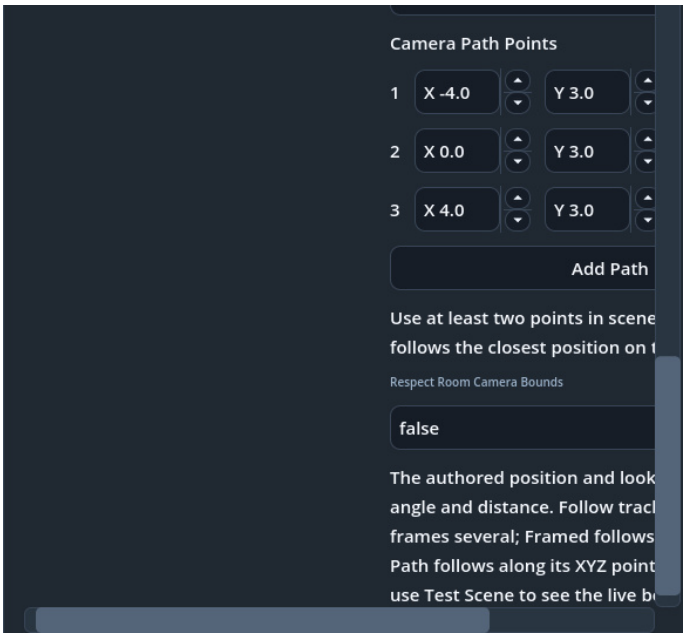

Figure 9. The path-point editor lets you add, order and edit XYZ points.

These coordinates describe camera positions. Do not copy the character’s floor path directly unless you want the camera at floor level. Give the camera enough height and sideways clearance for the subject and nearby walls.

1. Begin with a two-point straight route. Make sure it behaves correctly.
2. Add one turn. Check the view on each side of the turn.
3. If the camera chooses an unexpected nearby segment, separate overlapping route sections or simplify the route.
4. Try all legal player positions, including the extremes of the navigation area. An attractive middle section can hide poor endpoints.

With fewer than two usable XYZ points, the current implementation falls back to the fixed shot and logs a message. Do not interpret a stationary camera as proof that the route is working.

16 Targets, offsets and a separate aim

Follow Target chooses what drives movement. Look At Target can choose a different subject for aiming. For example, follow the player walking toward a clock, but aim at the clock. This can reveal the destination while retaining movement related to the actor.

Entry	Meaning
player	The current controlled character. It resolves again when control changes.
elias	That specific character or object in the current room.
Additional Target IDs	A comma-separated list for group tracking; scripts can also supply an array.
Target Offset XYZ	A point above or beside the target’s origin. The optional look-at aim also uses this offset.
Blank Look At Target	Keep the behavior’s normal aiming rather than a separate target.

A target must exist in the current room. A missing target does not create one. The shot keeps its fixed pose until it has an available follow target. Check spelling, enabled state and room placement when tracking appears inactive.

Offsets are a composition tool. Try a small X offset for space in front of a subject or a higher Y offset for a tall model. Large offsets can place the point the camera is trying to show above the ceiling or outside the room.

17 Cuts, blends and easing

Activate / Blend to Camera chooses a named camera. It works in action lists, visual scripting and cutscene camera events. Blend Seconds decides how long the transition takes.

Blend Seconds	Result
0	Cut directly to the destination shot.
Positive number	Blend over that many seconds. 1.2 is a useful starting value.
-1	Use the destination camera’s Default Camera Blend setting.

Blend Curve	What you will notice
linear	Steady interpolation; starts and stops more abruptly.
sine	Gentle start and finish; a sensible default.
cubic	A more pronounced ease in and out.
quint	Stronger acceleration and deceleration.

Wait for Blend holds a sequential action list until the transition completes. In a timeline, leave it off when other scheduled motion and speech should continue beside the transition. If a second blend interrupts the first, it starts from the current live view.

> The space between two good shots may be a bad camera route. Watch the entire blend for walls, ceilings, unwanted backs of scenery and sudden changes in actor scale.

18 Camera priorities and room zones

Set Camera Priority changes a camera’s priority and then selects the highest-priority enabled advanced camera in the room. A larger number wins. Ties retain the current camera where possible.

1. Give the normal follow camera priority 0 and a doorway detail camera priority -10.
2. On the doorway trigger’s On Enter event, add Set Camera Priority for door_close with priority 20.
3. On Exit, set door_close back to -10. The normal priority-0 camera can then win again.
4. Test entering, leaving, entering twice and switching rooms. Also test approaching from the opposite direction.

Priority is not a trigger detector. It does not create camera volumes by itself. The trigger or region supplies the event that changes a number. Room entry still begins with the room’s default or explicitly requested camera; setting priorities is a separate action.

Use Activate Camera for a direct instruction such as “show this close-up now.” Use priority when several gameplay situations compete for control. Avoid leaving a temporary camera at a high priority after its situation ends.

> Runtime priorities and target changes apply to the current room session. Change the camera Inspector if the setting should be part of the authored project.

19 Respect room bounds without cropping the action

Room Camera Pan Bounds restrict the camera’s view correction. They are separate from navigation bounds, which restrict walking. A camera can be well inside the room while its view still exposes an unwanted edge of a 2.5D background.

1. Select the room and open Camera & View. Enable Room Camera Bounds.
2. Set Minimum and Maximum X and Z around the part of the set that should be visible. Keep minimum values below maximum values.
3. Test the closest and farthest walking positions. Adjust bounds while watching the picture, rather than matching the floor dimensions blindly.
4. For an advanced camera, check Respect Room Camera Bounds. Turn it off on a cinematic shot that genuinely needs to move beyond the normal gameplay limits.

Set Room Camera Bounds is also available as an action. Supply sceneId, enabled, minX, maxX, minZ and maxZ. Use it when a camera limit must change after a story event, rather than changing the walking area.

If a group camera will not pull back far enough, temporarily turn Respect Room Camera Bounds off. If that repairs the composition, revisit the bounds or the shot angle. Increasing damping will not repair a hard spatial limit.

> A room’s pan bounds do not resize the walking grid, navigation area, event volume or collider. Those systems need their own dimensions.

20 The original Dynamic Follow controls

Dynamic Follow is the older player-tracking system. It remains useful, especially for established projects. Its settings inherit from Global Settings to the room, then to an individual camera. An override can inherit, disable tracking or use custom settings.

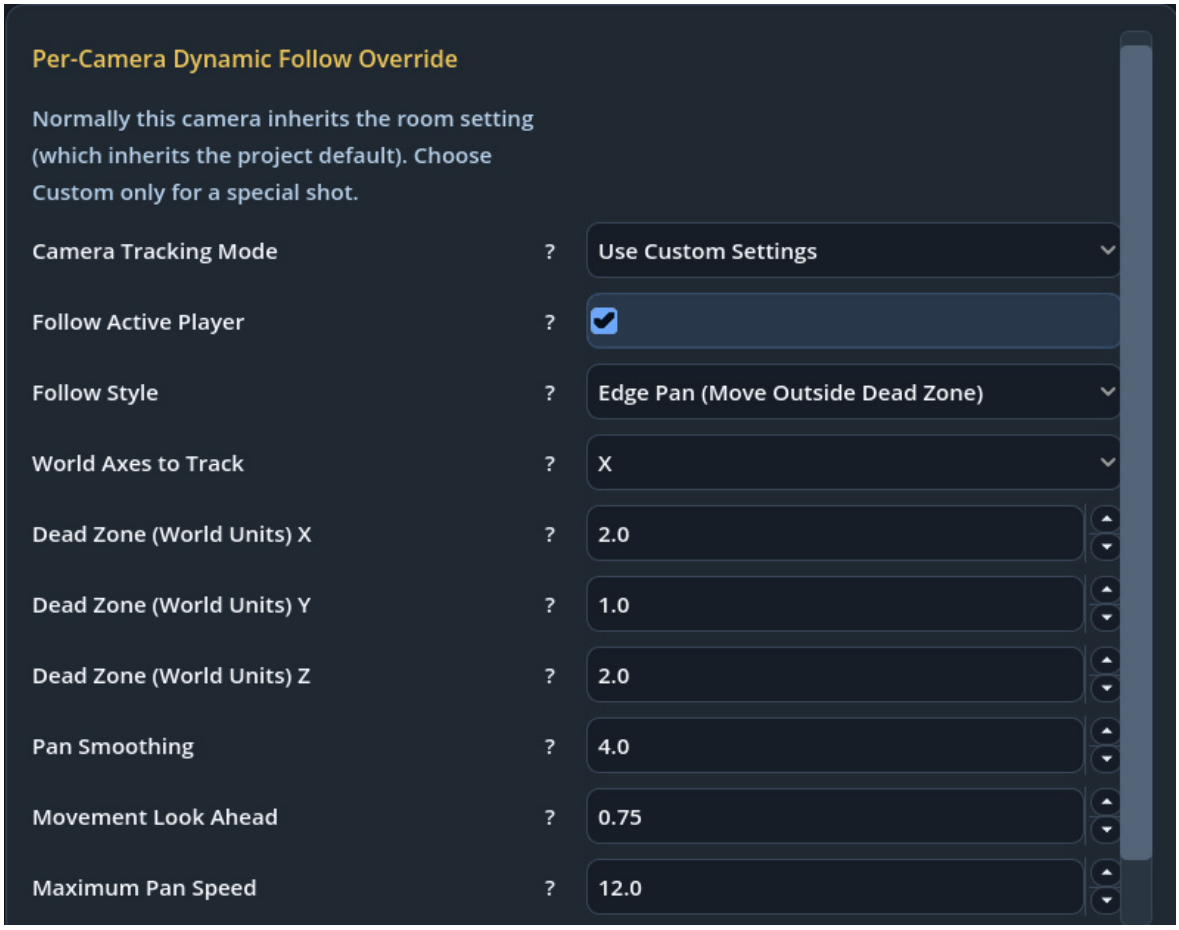

Figure 10. Dynamic Follow uses world axes and a world-unit dead zone. Its smoothing scale differs from Advanced Camera damping.

Control	How to use it
Follow Active Player	Turns this tracking system on.
Edge Pan	Moves after the player leaves the world-space dead zone.
Smooth Follow	Continuously tracks movement.
World Axes to Track	X, Z, X + Z or X + Y + Z. Match the direction the room needs to scroll.
Pan Smoothing	Higher values respond faster. This is not the seconds-based advanced damping control.

21 Dynamic Follow: speed, look-ahead and limits

Control	Starting approach
Dead Zone XYZ	Start near [2, 1, 2] world units, then enlarge it if the camera moves for every small step.
Movement Look Ahead	Start at 0.75. It adds room in the direction of movement; reduce it if direction changes make the picture wander.
Maximum Pan Speed	Start at 12 world units per second. A cap below the player’s effective screen motion may leave the camera behind.
Clamp to Room Navigation Bounds	Uses navigation-related limits in addition to other applicable camera restrictions.
Room Bounds Margin	Leaves space at the edges. Adjust while checking the actual picture.

1. Set Global Settings → Dynamic Camera to the ordinary behavior you want across the game.
2. Let rooms inherit that behavior until a particular room needs a different style.
3. Use a per-camera custom override only for an exception, such as a long hallway shot.
4. When trying Advanced Camera, disable the older custom follow on that shot so you can judge the new behavior on its own.

If the view behaves differently after a camera switch, check inheritance at all three levels. The new shot may be inheriting a room override you had forgotten about. Keep a small note of intentional exceptions while laying out the game.

22 A two-shot conversation in action lists

This setup needs two cameras in the current room: wide for normal play and conversation for the two speakers. Configure conversation as the group shot described earlier. Use IDs that match your own project.

Order	Action	Settings
1	Set Player Input Enabled	Enabled off, if the interaction should keep the player still.
2	Set Camera Targets	Camera conversation; Target player; Additional targets elias.
3	Activate / Blend to Camera	Camera conversation; Blend 1.2; Wait on.
4	Say / dialogue action	Run the conversation or show the intended speech.
5	Activate / Blend to Camera	Camera wide; Blend 0.8; Wait on.
6	Set Player Input Enabled	Enabled on.

Attach the action list to the intended Talk response, not to an unrelated hover event. Hover can start repeatedly as the pointer crosses the object. Test a short two-line exchange before connecting a long dialogue.

If your cutscene runner already manages player input and restores the camera, let it do that job. Do not add a second set of unrelated restore actions that compete with its cleanup.

> When an action returns an error about an unavailable camera, first check the current room. A camera’s attractive display name cannot compensate for using the ID of a camera in a different room.

23 Build the same flow with visual nodes

1. Create a Visual Script and keep its start event.
2. Add Set Camera Targets. Choose conversation, player and elias. Connect the start output to this node.
3. Add Activate / Blend to Camera. Choose conversation, 1.2 seconds and Wait for Blend on.
4. Connect the speech or Run Dialogue action after the blend. Finish with an activation of wide.
5. Attach the script to the object’s Talk response or call it from an action list. Test Scene and use Talk on the object.

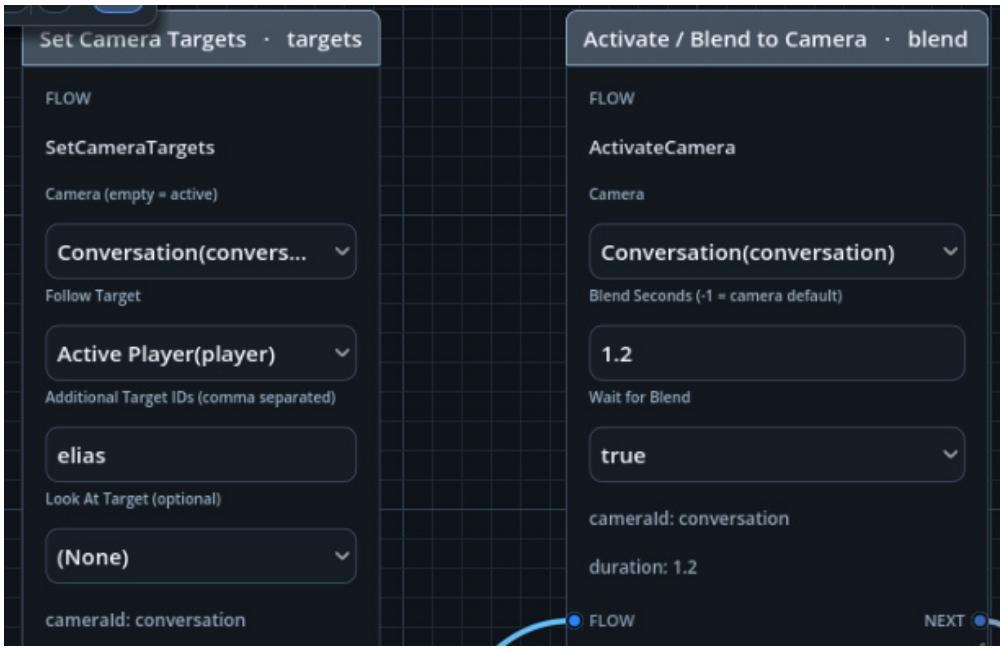

Figure 11. Camera target and activation nodes expose the same settings as their action-list counterparts.

The arrows control order. A node sitting beside the flow does nothing until the flow reaches it. With Wait for Blend off, the next connected node can run while the camera is moving. That is useful for a deliberate simultaneous effect, but it is an easy way to miss the close-up you expected to see first.

For branching conversations, return through a shared cleanup step so every branch restores the intended gameplay camera and input. Test the branch where a dialogue has no available choices as well as the ordinary successful conversation.

> A camera node edits the running room session. It does not rewrite the camera’s Inspector record into a new permanent authored shot.

24 Camera events on a cutscene timeline

Use the camera track for shot changes that must coincide with animation, movement or speech. Put Activate Camera events at the moments the viewer should see the new composition. The event’s duration is the blend duration; its timeline time is when the action begins.

Timeline time	Event	Reason
0.0	Activate wide, duration 0	Establish the room immediately.
1.5	Activate conversation, duration 1.2, wait false	Begin a two-shot as the actors settle.
4.0	Activate door_close, duration 0.7, wait false	Reveal the door during the relevant line.
6.5	Activate conversation, duration 0.8, wait false	Return to the speakers.

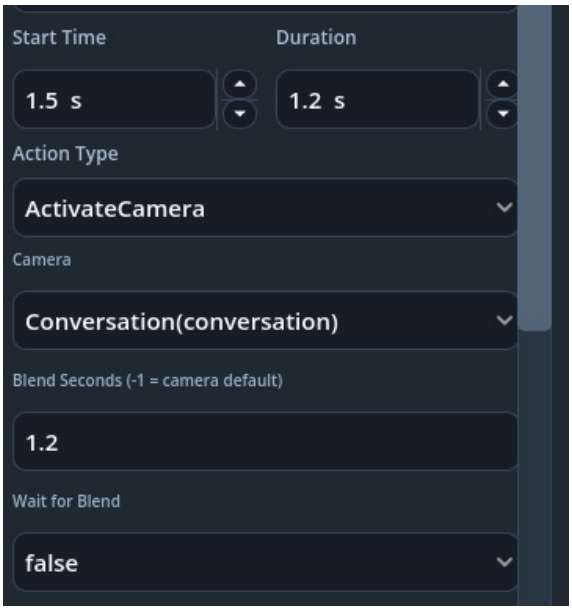

Figure 12. This camera event starts at 1.5 seconds, blends for 1.2 seconds and lets the timeline continue.

Leave Wait for Blend off for camera events that should run alongside scheduled speech and movement. Keep the entire transition before the next shot unless the interruption is intentional. Give the viewer time to read a revealed object.

Check Restore Camera on the cutscene if gameplay should resume with the pre-cutscene camera and its behavior. Test both normal completion and Skip. The view after Skip matters just as much as the carefully timed ending.

> Use the Clockmaker template to study existing camera events and named shots, then make your changes in a separate project copy. Follow target IDs and shot positions belong to that demo’s rooms.

25 Manual moves and orbiting

Named camera blends cover most adventure shots. Manual Camera Move and Camera Orbit are useful when you want a specific movement rather than a reusable tracking shot. Manual commands take ownership from advanced tracking; Activate Camera hands control back.

```
await Action("CameraMove", {
    "position": [2.0, 3.0, 7.0],
    "target": [0.0, 1.2, -1.0],
    "fov": 42.0, "duration": 2.0
})
await ActivateCamera({
    "cameraId": "wide", "duration": 0.8, "wait": true
})
```

The example is GDScript inside its normal run function. CameraMove interpolates position, aim and FOV to the supplied values over seconds. Action("SetCamera", {...}) places a manual pose immediately. If these older commands are not present in the shared visual action selector, call a script for them; the named advanced-camera actions have direct shared controls.

Camera Orbit takes a target object, degrees per second, distance, height, direction, mode and repeat count. Modes are a full turn, a half turn or a half turn and return. Stop Camera Orbit ends it. An orbit can pass through scenery, so test the full arc rather than just its opening frame.

26 Shake, fades and camera ownership

Camera Shake offsets the view briefly, then restores the original offsets. Use it for a bell strike, impact or short tremor. Begin with a small power and a short duration; large shake is tiring and can make precision clicking difficult.

```
await CameraShake({
    "direction": "horizontal",
    "duration": 0.35, "power": 0.04
})
```

Direction can be horizontal, vertical or both. Power is a camera offset amount rather than a pixel count, so compare it at the game’s actual lens and subject distance. A 2D Shake Screen effect is a different presentation tool; do not accidentally combine two shakes at full strength.

Fade Screen and Screen Transition conceal a change or mark a passage of time. They do not place the camera. If a new room begins on the wrong shot after a fade, repair its default camera or entry action.

First-person and XR normally own their gameplay camera. Activate Camera reports a failure while that ownership applies; use a cutscene or return to third-person before activating a third-person shot. A locked 2D Stage also rejects arbitrary free-camera moves.

> After a manual orbit or move, explicitly activate the intended gameplay shot. This is clearer than expecting a follow controller to guess when the cinematic is over.

27 Depth of field: focus on the subject

Depth of field blurs nearer or farther parts of a 3D view. It is an aid to composition. It is not a replacement for readable lighting, and strong blur can hide an object the player needs to click.

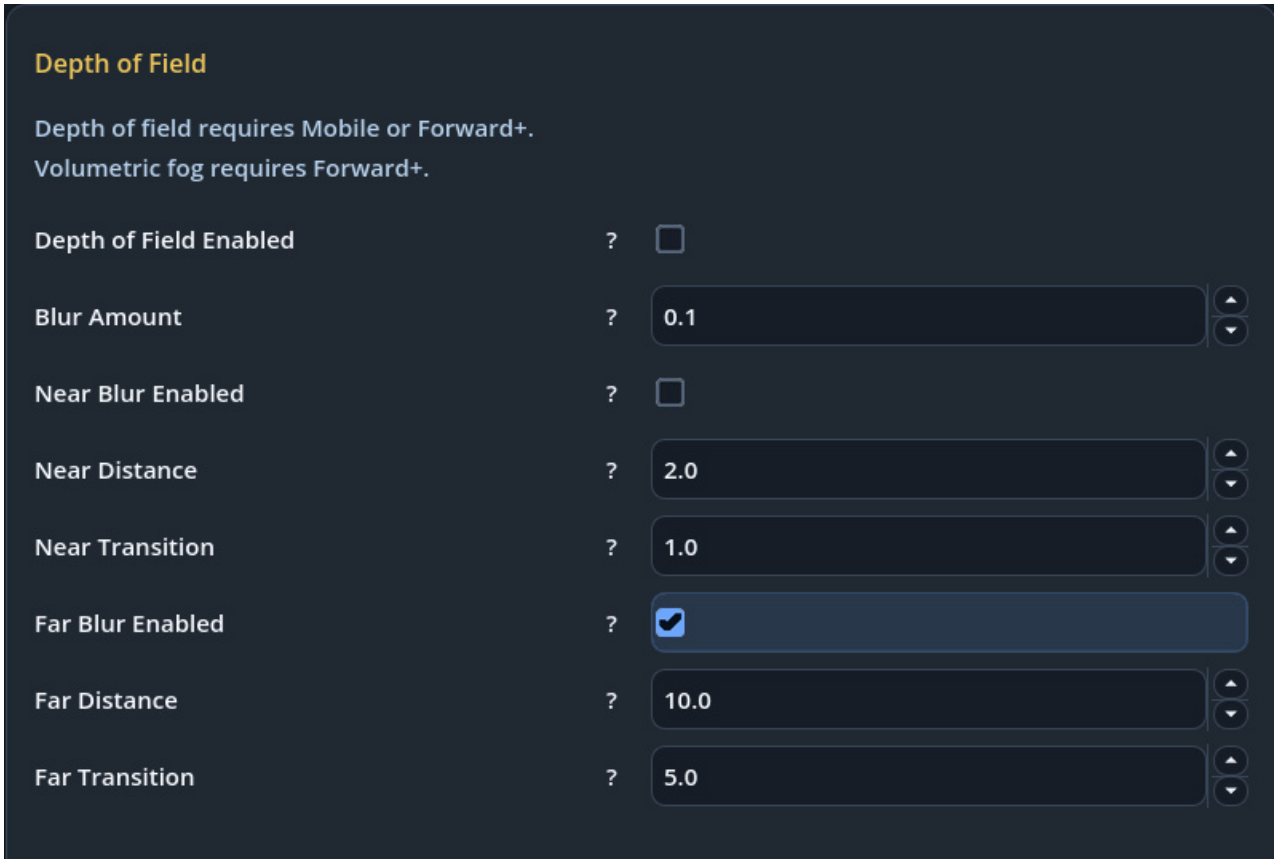

Figure 13. Each camera can have its own near and far blur distances.

1. Run the editor with Mobile or Forward+. Compatibility cannot show native depth of field.
2. Select the camera → Depth of Field. Enable it and begin with Blur Amount 0.1.
3. For a conversation, leave Near Blur off initially. Turn Far Blur on and put Far Distance beyond the speakers.
4. Set Far Transition to a few world units for a gradual blur. Test the actual camera and adjust distances by looking at the subject.

Native depth of field is unavailable in transparent FMV or layered 2D/3D views. A checked box alone does not mean that the current viewport can render it.

28 Focus distances and transitions

Control	Meaning
Blur Amount	Overall strength, 0 to 1. Start low.
Near Blur Enabled / Near Distance	Allows blur in front of the near focus boundary. Useful for a foreground object close to the lens.
Near Transition	The distance over which near blur changes. A very short transition can produce an abrupt boundary.
Far Blur Enabled / Far Distance	Allows blur beyond the far focus boundary. Keep important speakers inside the clear range.
Far Transition	Distance over which the far blur builds.
Room default-camera focus	Room Camera & View exposes the default camera’s focus settings as well. Make sure you edit the shot that is actually active.

Distances are measured from the camera, not from the floor or from the character. A high, tilted camera can be farther from a face than the horizontal floor distance suggests. When a close-up changes camera position, recheck focus rather than copying a distance from the wide shot.

For a practical first test, disable near blur, set far distance near 8 and far transition near 4 in a room of the example’s scale. Adjust for your shot. If everything is blurred, move the far boundary farther away before increasing image sharpness elsewhere.

> Changing renderer requires a full PCA 3D restart. Play / Restart restarts the preview, not the drawing backend.

29 First-person view and headset camera ownership

A first-person gameplay camera belongs to the controlled character rather than a fixed room viewpoint. Use it deliberately; enabling a third-person follow shot does not make it a first-person camera.

1. In Global Settings → Adventure & Cursor, find Runtime Control Profiles. Keep the system enabled and choose First Person as the Movement Profile.
2. Apply and Test Scene in an ordinary 3D room. Use the configured movement keys. Hold the middle mouse button and move the mouse to look around; release it to return the pointer.
3. For a legacy-controls project, Player View → First Person / FPS Style is the older view setting. A room can inherit or override it. Avoid mixing a conflicting modern profile with an old view override.
4. On touch devices, use the left stick to move and the right stick to look. Global Settings → Mobile Controls contains the first-person joystick options.

```
await SetMovementProfile({"profile":"first-person"})
# Return to the adventure view when appropriate:
await SetMovementProfile({"profile":"point-and-click"})
await ActivateCamera({"cameraId":"wide",
    "duration":0.0, "wait":true})
```

In this release, the eye position is a fixed 1.6-unit offset above the player, rather than an editable per-camera eye-height field. Advanced target offsets do not replace that first-person rule. Use a suitable character and room scale.

First-person and XR keep gameplay camera ownership. Use a cutscene or return to third-person before activating an ordinary room shot. A headset’s tracked view should not be treated as a fixed cinematic camera.

30 2D Stage and older 2D camera systems

A 2D Stage uses a single orthographic main camera and keeps it perpendicular to the stage. Select the Main Stage Camera Rig in Scene Objects, or use the room’s 2D Camera tab, to edit its framing and tracking.

Advanced tracking can scroll that stage and group framing can change its view height. It does not give the stage several independent camera identities or arbitrary rotation. Do not expect a 3D orbit to work on a deliberately locked stage.

Room style	Camera considerations
Ordinary 3D	Named shots, free angles, perspective or orthographic, advanced tracking and blends.
2D Stage	One constrained orthographic main view. Scroll, framing, background edges and walk-behinds matter.
Older Canvas2D / hybrid compositor	Retains its existing native 2D camera controls. The new advanced integration does not turn this separate compositor into the 3D camera system.
2.5D with a background image	Keep the picture’s perspective aligned with the 3D walking plane. Camera movement can expose background edges or make the alignment obvious.

For a scrolling background, test both ends of the walk area and every permitted zoom. The navigation area, visible background extent and camera limits must agree. Making only one of them larger leaves a mismatch.

31 2D Stage framing and scrolling controls

Select the Main 2D Stage Camera in Scene Objects. Its identity, projection and aim axis are managed by the stage. The framing and scrolling controls remain editable.

Control	How to use it
View Center X / Z	The part of the stage that begins at the center of the picture.
Camera Height / Distance	Distance above the stage plane. Use Orthographic View Height for apparent zoom.
Orthographic View Height	Vertical scene span in the picture. Larger values show more of the stage.
Camera Roll	Rotation within the stage picture. This does not free the camera to orbit away from the stage.
Room Scrolling	Fixed, Follow Player or Follow Near Screen Edge.
Follow Axes	Horizontal, vertical or both.
Dead-Zone Width / Height	Stage-space area before edge-follow starts. Do not confuse this with advanced framed mode’s screen fractions.
Look-Ahead X / Z	Offset for the tracked composition.
Smooth Follow / Follow Speed	Whether to smooth movement and how quickly it catches up.
Minimum / Maximum X / Z	Stage camera limits. Check the visible background edges.

First make fixed framing correct, then enable scrolling. Walk to every edge, test your largest permitted view height and check walk-behind sorting. If the camera shows beyond the background, adjust the background extent, view height or camera bounds together.

> The older Canvas2D Set 2D Camera action uses XY position, XY zoom and rotation. Those are native 2D controls, not a way to create extra independent 2D Stage camera identities.

32 Camera scripts in all four languages

All four scripting languages use the same action properties. These examples set the targets and blend to an existing conversation camera. Put the code in the language’s normal script record and call the script from an event.

```
# GDScript
func run(_context: Dictionary = {}) -> Variant:
    await SetCameraTargets({
        "cameraId":"conversation",
        "targetId":"player", "targetIds":"elias"
    })
    return await ActivateCamera({
        "cameraId":"conversation",
        "duration":1.2, "wait":true
    })
```

```
function run()
    SetCameraTargets({"cameraId":"conversation","targetId":"player","targetIds":"elias"})
    ActivateCamera({"cameraId":"conversation","duration":1.2,"wait":true})
end
```

The second example is PCA Script. Replace IDs with your own. A display label such as Conversation (conversation) is not the ID string to paste into code. Use conversation.

33 Lua, BASIC and partial behavior updates

```
local Script = {}
function Script.run(context)
    await(SetCameraTargets({cameraId="conversation",
        targetId="player", targetIds="elias"}))
    return await(ActivateCamera({cameraId="conversation",
        duration=1.2, wait=true}))
end
return Script
```

```
FUNCTION Run()
    AWAIT SetCameraTargets({"cameraId":"conversation","targetId":"player","targetIds":"elias"})
    AWAIT ActivateCamera({"cameraId":"conversation","duration":1.2,"wait":TRUE})
END FUNCTION
```

The first block is Lua. The second is BASIC. Use the editor’s generated entry-point template if a new record supplies one; keep the actions inside it.

Set Camera Behavior accepts partial settings. For example, cameraId conversation, enabled true, mode group and damping 0.15 change those settings without requiring every field again. Set Camera Targets changes targetId, targetIds and lookAtId through the same behavior update route.

For an active shot, a runtime behavior update can rebuild its controller. Do not call it every frame just to repeat the same values. Change behavior when the situation changes.

34 Four jobs that often get called lighting

A convincing lamp usually needs more than one setting. The visible bulb, the light falling on the table, the halo around the bulb and the reflection in a polished surface are different parts of the picture.

Job	PCA controls
Light nearby geometry	Point, spot or directional light; ambient and Enhanced indirect light.
Make a surface look self-lit	Object emission / glow or an appropriate shader.
Make a halo around a bright area	Scene Glow or an object bloom effect.
Show the room or a moving person in a surface	Room Capture, Enhanced screen-space reflections or Live Mirror, depending on the result required.

Set each part up separately. First make the ordinary light illuminate a neutral box. Next make the bulb look bright. Add a gentle halo. Finally adjust the material that should reflect it. This order makes failures much easier to identify.

A painted highlight in a texture does not prove that a light is working. A shiny highlight on a sphere does not prove that it can reflect the character. A bright window image does not automatically cast a patch of light on the floor.

> The practical lighting examples use normal room-scale units and relative intensity values. They are starting points, not real-world watt or lumen specifications.

35 Add and aim a light

1. In Scene Designer, use Add to Scene → Light. Give it a name and ID that describe its role, such as Warm Key Light / key_light.
2. Open Light Settings. Keep Light Starts On checked. Choose point for a first test.
3. Move it above and to one side of a neutral-colored box. Try Position [-3, 4, 3], Intensity 1.8 and Range 12 in the example room.
4. Choose a slightly warm color such as #ffd7a6. Enable Cast Shadows.
5. Apply Changes, save and Test Scene. Turn other lights off briefly if you cannot tell what this one contributes.

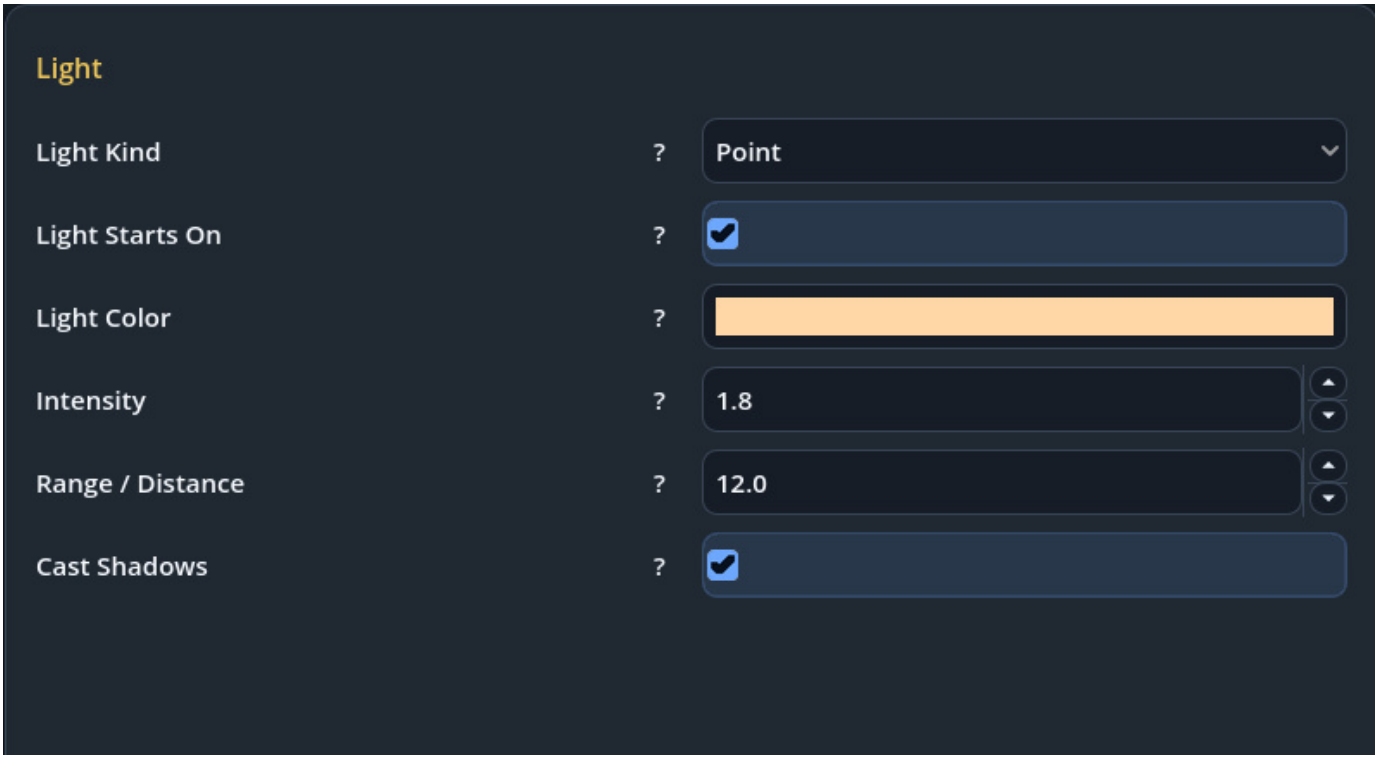

Figure 14. The everyday light controls live together in Light Settings.

Move lights with the same scene transform tools used for objects. Point lights shine in all directions, so rotating one does not aim a beam. Spot and directional lights need rotation. Light position, range and intensity solve different problems; increasing intensity is not a substitute for a range that reaches the subject.

36 Point light: a local pool of light

A point light shines outward from its position. Use it for a lamp, candle group, glowing machine or local fill. Range / Distance limits how far it reaches. The light becomes weaker away from the source according to its attenuation settings.

1. Put the point light where the source would be, or just outside its lampshade if the model hides it.
2. Begin at intensity 1 and a range large enough to reach the nearest floor and character.
3. Increase intensity until the nearby surface reads well. Then shorten the range until it stops affecting unrelated parts of the room.
4. Enable shadows only if this light needs to show occlusion. One useful shadowed key and several inexpensive unshadowed fills often work better than shadows on every lamp.

A point light inside an opaque object may be blocked by that object. If a table lamp appears bright but the table stays dark, inspect its position relative to the lampshade and base. Also check that the material is lit rather than unshaded.

Large imported rooms can be a single mesh. On renderers with per-mesh light limits, several small overlapping lights may not all contribute. Reducing overlapping ranges is often more useful than adding another light.

37 Spot light: aim a controlled cone

A spot light produces a cone. It suits a desk lamp, spotlight, shaft from a window or a deliberately narrow pool of light. Its position is the cone’s source; its rotation aims the cone along local negative Z.

1. Choose spot in Light Settings. Put it above the surface you want to light.
2. Rotate it toward the surface. In a simple room, Rotation X near -55 degrees aims it downward and forward; adjust in the scene view.
3. Use Range 8-10 and Intensity 1-2 as a first test. Keep Cast Shadows on while checking the beam.
4. If you need a different cone width, edit light.spotAngle in the light’s Raw JSON record. Try 35 degrees, Apply Changes and restart Test Scene.

The current unified Light Inspector exposes the main light controls but does not expose every advanced cone and shadow field as a separate control. The advanced record-edit lesson shows how to make those changes without hunting for a nonexistent slider.

Spot Angle Attenuation softens the cone’s edge. The light’s range still limits its reach. A cone pointing at the floor behind the camera can be working perfectly while the subject remains dark.

38 Directional and hemisphere lights

Kind	Use and limitations
directional	Parallel light across the scene, useful for sun or moon. Rotation determines its direction. Moving its icon does not move a local pool of light.
hemisphere	In this PCA version, a broad local fill implemented as an omni light with an 18-unit range and no shadows. It is not a physical upper-sky/lower-ground hemisphere lighting model.

1. For a daylight test, add a directional light. Try Rotation [-45, -30, 0], a slightly warm neutral color and Intensity 0.8.
2. Turn its Cast Shadows on. Put a tall primitive on the floor so the light direction is obvious.
3. Rotate the light until the shadow points in a useful direction. Check that it does not cover every important object.
4. Add a low-intensity point or hemisphere fill only if the unlit sides are too hard to read.

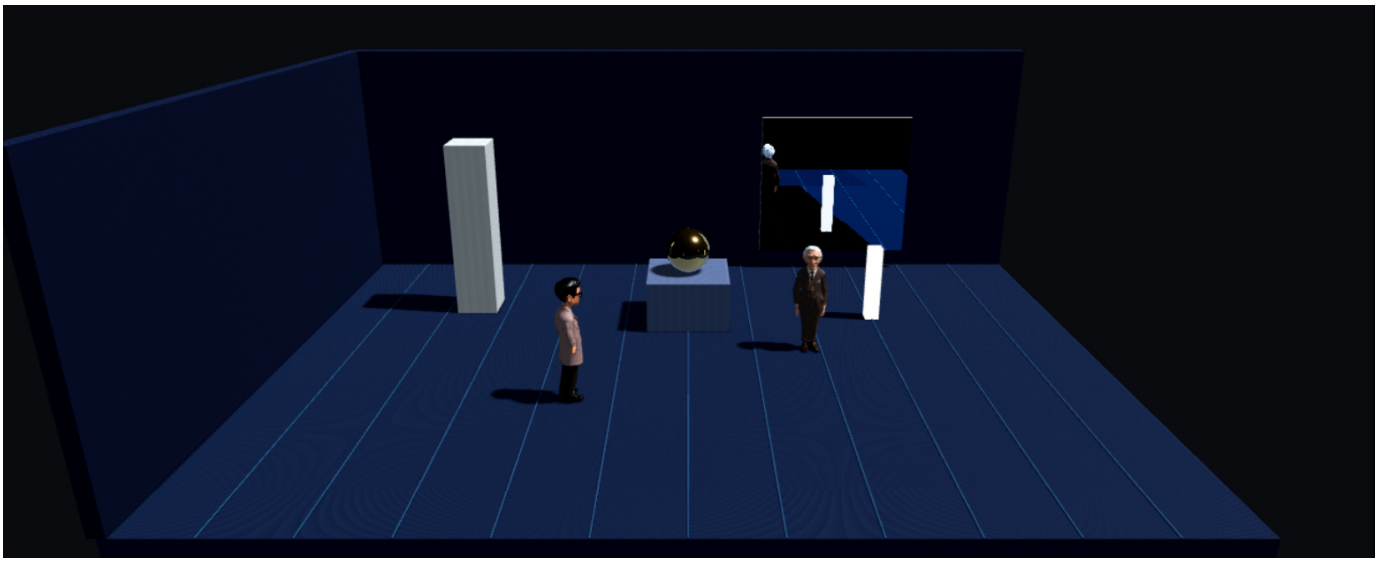

Figure 15. A single directional light establishes the main shadow direction. Additional fill is a separate decision.

Do not try to brighten the underside of a roof by moving a directional light closer to it. Change the direction or add appropriate fill.

39 Ambient light and room brightness

Global Ambient Color and Ambient Intensity provide broad light for the game. They help surfaces remain readable where direct lights do not reach. Start modestly; too much ambient light flattens shadows and makes every room feel evenly lit.

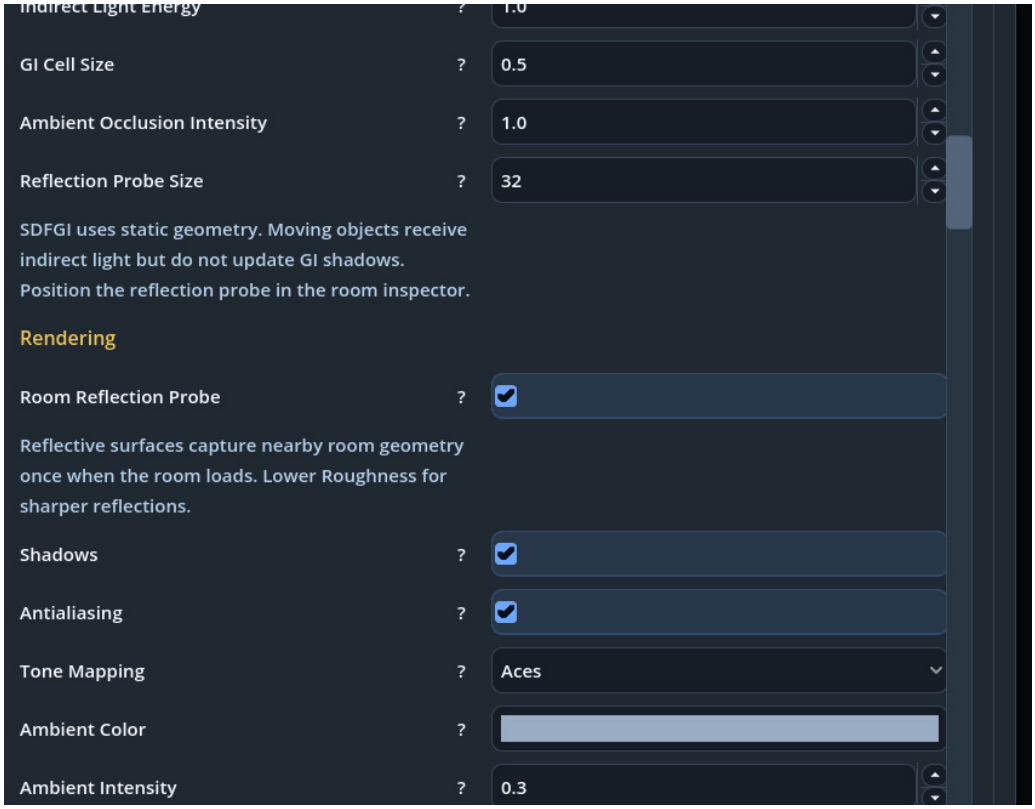

Figure 16. Ambient light and tone mapping are global rendering defaults.

For the example room, ambient color #9aabc4 and intensity 0.32 leave room for a warm key and cool fill. A lighter stylized room may need more. A night scene still needs enough separation for the player to see what can be used.

Room Brightness is a finishing adjustment on the rendered image. At 1 it is normal; 0 is dark. It does not create a new light source or change where shadows fall. If the whole picture is slightly too dark, brightness may be useful. If one face is unlit, fix the lighting there.

> Turn off grading, fog and heavy screen effects while establishing your lights. Otherwise you may keep increasing light intensity to fight a filter that is darkening the finished picture.

40 Build a warm key and cool fill

A key light establishes the main direction and mood. A fill makes the shadow side readable. A restrained rim or background light can separate a character from the wall. The labels describe jobs, not special PCA light kinds.

Example light	Starting setup
Warm key	Point at [-3, 4, 3], #ffd7a6, intensity 1.8, range 12, shadows on.
Cool fill	Point at [3, 3, -2], #7ea9ff, intensity 0.65, range 12, shadows off.
Window sun	Directional at rotation [-45, -30, 0], intensity about 0.8. Use only if the scene needs that second direction.

1. Turn the fill off and make the key alone useful.
2. Turn the fill on. Lower it until it reveals the dark side without erasing the key’s direction.
3. Walk the player through the room. Check the face against both the light and dark backgrounds.
4. Inspect the wide shot and the conversation shot. A light that looks good from one side may shine directly into the face from another.

Keep one clear visual direction. Several equally bright shadow-casting lights can produce crossed shadows that make the floor noisy. Give extra lights a clear purpose.

41 Skyboxes and sky ambient

A skybox fills the distant background. In Lighting & Fog, enable the room skybox and choose a procedural sky or an imported panorama image. Use the procedural color controls for a simple sky; use a suitable panoramic image for a specific environment.

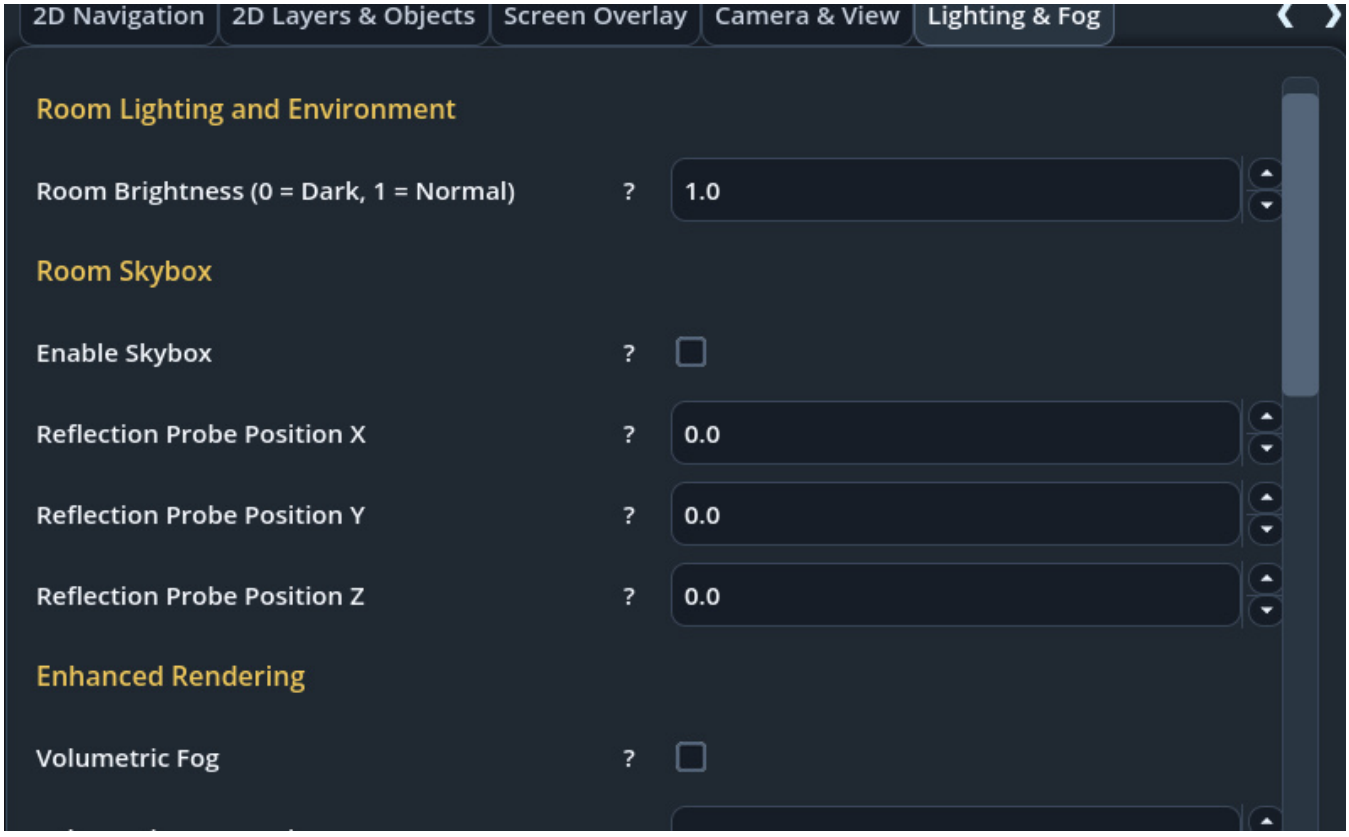

Figure 17. Room Lighting & Fog includes sky settings as well as brightness and atmosphere.

Control	What it changes
Top / Horizon colors	The upper sky and its horizon transition.
Ground Bottom / Horizon colors	The lower part of the procedural environment.
Skybox Energy	Brightness of the sky material.
Rotation Degrees	Rotates the sky around Y. Useful for aligning a panorama.
Use Sky as Ambient Light	Uses the sky for ambient illumination instead of relying only on the flat ambient color.

A sky image with a bright sun does not automatically create a matching directional light. Add and aim one if the scene needs that shadow direction. Align its direction with the bright part of the panorama.

42 Advanced fields through Raw JSON

Some supported rendering fields are stored in the record but do not have individual controls in the current unified Inspector. Use Raw JSON (Advanced) for those fields. This is optional; ordinary lights and shadows work from the main tabs.

1. Select the exact light or object. Save the project before an advanced edit.
2. Open Raw JSON (Advanced). Find the existing light, shadow or glow dictionary inside that record.
3. Add or change the named fields. Keep all other record fields and use valid JSON: double quotes, commas between entries and no trailing comma.
4. Apply the raw record using Apply Changes. Reopen the tab to confirm the values survived.
5. Restart Test Scene. A stored property change does not guarantee that an already-created light, material or capture has rebuilt itself.

```
"light": {
    "kind": "spot", "enabled": true,
    "color": "#ffe5b4", "intensity": 2.0,
    "distance": 10.0, "spotAngle": 35.0,
    "spotAngleAttenuation": 1.0,
    "castShadow": true, "shadowBias": 0.05,
    "normalBias": 0.6, "shadowOpacity": 0.8
}
```

This is a fragment inside a light record, not a replacement for the entire record or project. Do not paste it over the ID, type and transform.

43 Make shadows work in the first place

A visible shadow depends on the global switch, a shadow-casting light, a shadow-casting object and a surface that receives the shadow. Check those four conditions before changing shadow quality.

1. In Global Settings, enable Shadows and apply.
2. Select the relevant point, spot or directional light. Enable Cast Shadows and make sure it starts on.
3. Use a solid primitive on a solid floor as a simple test. For an object with advanced shadow settings, keep shadow.cast and shadow.receive true.
4. Aim the light so the shadow falls where you can see it. Shorten a huge light range or shadow distance if appropriate.
5. Restart Test Scene and compare the light on and off.

A collider is not the rendered shadow shape. Shadow casting uses the rendered geometry and its material rules. Changing an event volume does not change the visible shadow or make a model receive light.

> Hemisphere lights do not cast shadows in this implementation. Ambient light also does not create a directional cast shadow. Use a suitable direct light.

44 Choose shadow quality sensibly

Select the room → Lighting & Fog → Shadow Quality. PCA offers hard, low, medium, high and ultra. The choice controls filtering and shadow-map allocation. It does not fix an incorrect light direction, missing caster or transparent material.

Quality	Current allocation and practical use
hard	Hard filtering with a 1024 atlas. A deliberate crisp or inexpensive starting choice.
low	Soft low filtering with a 1024 atlas.
medium	Soft medium filtering with a 2048 atlas; a useful first target.
high	Soft high filtering with a 4096 atlas. Try when close-up shadows need more detail.
ultra	Soft ultra filtering with an 8192 atlas. Use only when the improvement is visible and the target hardware can afford it.

Rendering Quality is a separate global policy. Balanced reduces authored high/ultra shadow quality to medium. Performance reduces it to low unless hard was selected. Those policies also shorten directional shadow distance and simplify directional split settings.

Check shadows while the player moves. A still screenshot can hide shimmering, disappearing distance shadows and changes between directional cascade regions.

> A larger atlas uses more graphics memory. It is not a free polish switch, especially on integrated graphics or a device already struggling with the Clockmaker scene.

45 Bias, normal bias and detached shadows

Shadow maps need a small offset to avoid a surface falsely shadowing itself. Too little offset produces speckles, stripes or a dirty-looking surface. Too much makes the shadow look detached from the object.

Advanced light field	What to test
shadowBias	Start near 0.05-0.1. Increase slightly for self-shadow artifacts; reduce if the shadow floats away.
normalBias	Offsets along the surface normal. Start near 0.6 for the example’s human scale and adjust carefully.
shadowOpacity	0 to 1. Reduces the visible strength of this light’s shadow.
shadowBlur / lightSize	Softness-related settings. Renderer support and the room’s filter quality affect the result.
shadowDistance	How far relevant shadows remain available. For local lights, a distance fade also reduces their contribution near the limit.

These fields go inside the light dictionary through Raw JSON when the main Inspector does not expose them. Change one at a time. Save a baseline value so you can return to it.

If an imported model is a hundred times larger or smaller than your room scale, repair its scale first. Extreme scales make every light range, clip distance and shadow offset harder to reason about.

46 Directional shadow cascades

Directional lights cover large areas. Cascades split that coverage so nearer geometry can receive more detail. PCA supports orthogonal, two-split and four-split directional shadow modes through the light record.

```
"cascadeMode": "four",
"shadowDistance": 40.0,
"cascadeSplit1": 0.1,
"cascadeSplit2": 0.2,
"cascadeSplit3": 0.5,
"cascadeBlend": true,
"cascadeFadeStart": 0.8,
"cascadePancakeSize": 20.0,
"angularDistance": 0.0
```

Add these fields inside the existing directional light dictionary. Split values are proportions, in increasing order. Begin with the defaults rather than setting several split points to the same number. Cascade blending can hide a visible transition between regions.

For a small interior, reducing Shadow Distance can be more useful than increasing quality. You do not need distant shadow coverage behind scenery that the player can never see. For an outdoor room, check the farthest visible walking position.

Balanced and Performance quality can replace four splits with two, cap the shadow distance and remove angular softness. Judge the final selected quality mode, not only Authored Quality.

47 Object shadow settings and contact patches

The shadow dictionary belongs to the object or scene character instance. It controls that object’s casting, receiving and helper contact patch. These fields are advanced record settings in the current Inspector layout.

```
"shadow": {
    "cast": true, "receive": true,
    "doubleSided": false, "transparent": false,
    "optimization": "static",
    "contactEnabled": false,
    "contactSize": 1.0, "contactOpacity": 0.32
}
```

Use doubleSided carefully on a thin surface that needs both sides represented for casting. Check the actual model before using it everywhere. Transparent shadow handling can alter how an alpha material participates; it does not make every translucent shader behave like solid geometry.

Contact Shadow creates a soft flat patch near the object’s base. It can help a character feel grounded when full shadows are too expensive. It is not a screen-space contact-shadow ray trace, and it does not automatically conform to stairs or sloping terrain.

Optimization static/dynamic/disabled selects indirect-light geometry participation. It is not a general promise to cache all direct shadows. Mark a script-moved prop dynamic when using indirect lighting.

48 Emission: make the surface look bright

Emission gives a material a self-lit appearance. A signal light can stay visibly bright in a dark room even before you add a separate light that illuminates its surroundings.

```
"glow": {
    "enabled": true, "startOn": true,
    "useAlbedo": false, "color": "#5fe0ff",
    "energy": 3.0, "operator": "add"
}
```

1. Select a primitive or model and open Raw JSON (Advanced). Add the glow dictionary at the record’s top level, beside material and transform.
2. Use enabled true and a moderate energy such as 2-3. Apply and restart Test Scene.
3. To use the original material color as the emission color, set useAlbedo true.
4. Only after the surface looks bright, add Scene Glow for a halo and an actual light for nearby illumination if required.

An optional textureAssetId masks the emitting part with an imported image. operator can add or multiply; onUV2 chooses the second UV channel when the model has one. A shader material may implement its own emission rather than follow standard material settings.

> Do not enable emission on an entire textured character just to brighten a dark face. That removes useful shading across the whole model. Repair the lighting instead.

49 Scene glow: add the halo

1. Select the room → Lighting & Fog. Enable Scene Glow.
2. Start Scene Glow Intensity around 0.3. Test an already-emitting object.
3. Increase gently until the halo is visible around the bright area. Keep nearby text and interactive silhouettes readable.
4. Compare the same scene with glow off. If the whole picture simply looks washed out, reduce glow or repair the brightness of its source.

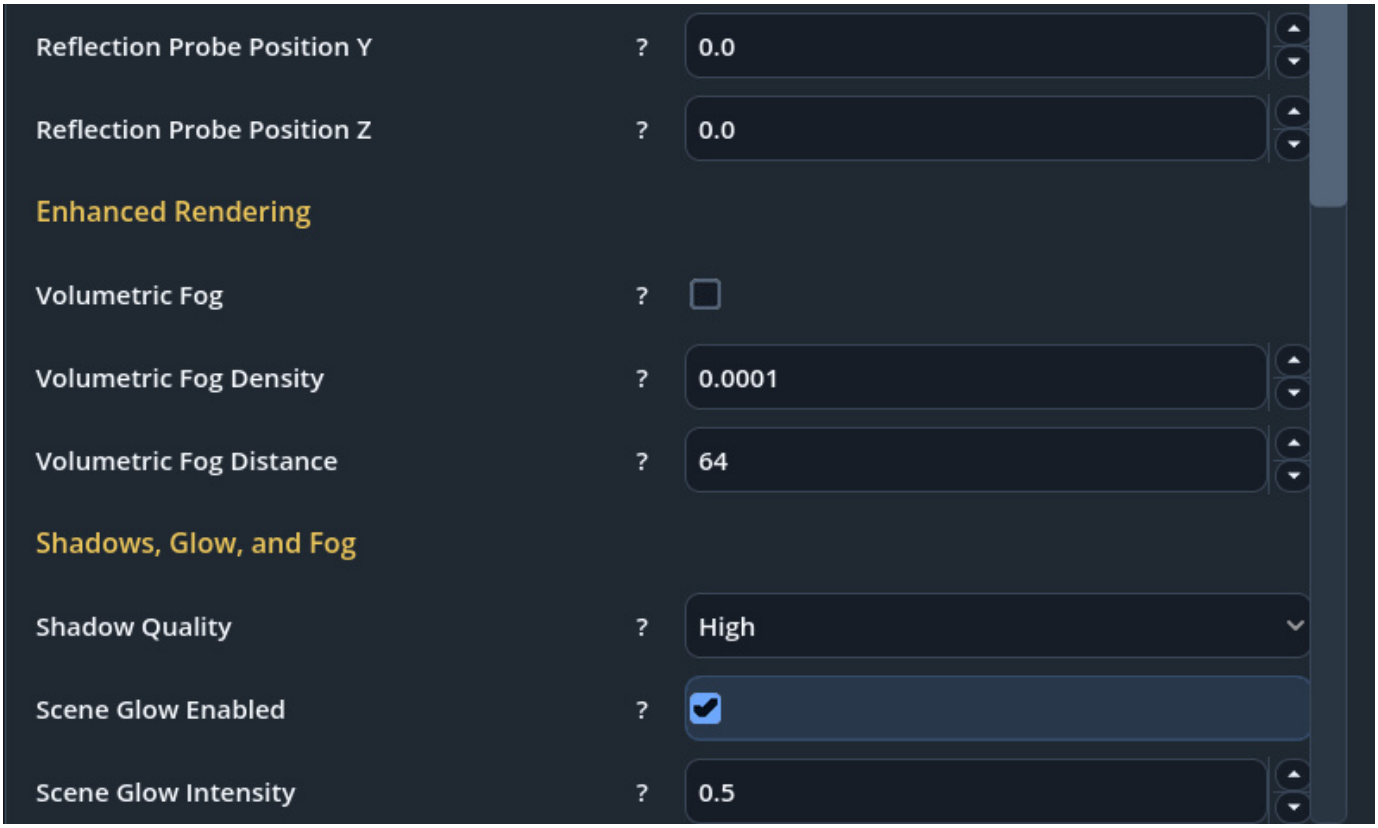

Figure 18. Shadow quality and Scene Glow are room environment controls.

Object emission and environment glow have separate switches. Turning on a room’s glow does not choose which object should be a lamp. Turning on object emission does not necessarily create a halo when the room’s glow is off.

Object bloom effects and hover bloom can request environment glow automatically. That convenience is separate from an ordinary authored emission dictionary. The final halo can also differ between renderers and quality settings.

A halo is added to the picture. It does not by itself cast warm light on the table, create a shadow or reflect a moving character.

50 Advanced glow thresholds and spread

For more control, edit the room environment dictionary in Raw JSON. Keep the ordinary room glow working before adding these fields. They tune the native glow pipeline; not every backend produces identical results.

Field	Purpose
glowHdrThreshold	Brightness threshold for contributing to glow. A lower threshold includes more of the picture.
glowHdrScale / glowHdrLuminanceCap	Control the influence and cap of very bright areas.
glowStrength	Overall spread/strength contribution.
glowBloom	Adds a broad bloom component; too much can wash out dark areas.
glowBlendMode	screen, additive, softlight, replace or mix. Start with screen.
glowMix	Mix amount when relevant to the blend mode.
glowNormalized / glowLevel1…7	Normalization and the contribution of seven spread levels. Start with defaults before adjusting individual levels.

Native default spread weights in this build emphasize levels 2, 3 and 4. The seven weights are individual fields named glowLevel1 through glowLevel7. Try one adjustment at a time and keep a before screenshot.

If a tiny source never looks bright enough for the threshold, increasing global glow intensity may only brighten everything else. Increase the emitter moderately or use an appropriate object bloom effect.

51 Object bloom and real surrounding light

In an object’s Visual Effects tab, Shader Layers and Effects can add bloom. This is a convenient way to stylize a magical object or machine. It is separate from the advanced glow dictionary used for standard material emission.

1. Select the object → Visual Effects. Choose a bloom preset in an available shader layer.
2. Start with a low layer intensity and test the object against a dark background.
3. If it should illuminate nearby 3D objects, enable Bloom Lights Surrounding 3D Objects.
4. Set Bloom Light Color, Energy and Range. Keep the range local. Enable Bloom Light Casts Shadow only if it is needed.

The surrounding-light option adds a real local light. That is why it can light the floor while an ordinary halo cannot. It has a cost like any other light and can produce unwanted overlap if every decorative object receives one.

Object shader layers also offer exclusions for scene lights, scene shadows and RGB lighting. Review them when a stylized object appears disconnected from the room. An imported custom shader can implement different lighting rules from the built-in presets.

> Use a small number of light-emitting hero props. A room full of decorative halos does not need a matching room full of shadow-casting point lights.

52 Switch lights and glow from events

Use the dedicated runtime actions when you want a switch to change visible lighting. Editing a generic stored property is not always enough to rebuild a live material or light.

```
await Action("ToggleLight", {"entityId":"key_light"})
await SetGlow({"entityId":"lamp", "enabled":true})
await SetSceneBrightness({"sceneId":"lab",
    "brightness":0.8})
```

The example is GDScript inside a run function. ToggleLight is a supported action through Action even if it is not listed by the current shared selector. SetGlow and ToggleGlow have shared action fields. Author the object’s glow before toggling it; the toggle does not invent an emission setup for an object that has none.

Action	Use
ToggleGlow	Reverse an authored object glow’s on/off state.
SetGlow	Choose on or off explicitly. Use this for reliable cleanup.
ToggleRGBString	Switch an authored decorative string.
SetSceneBrightness	Change the current or named room’s image brightness.
SetSceneTint	Apply a room tint with color and strength.

An on/off variable can link a switch to a light, glow or RGB string. Use the same variable intentionally and test both the initial state and returning to the room.

53 Decorative RGB bulbs

RGB Lighting is a scene object type. You can also attach its bulb-string settings to another object through the RGB Lighting tab. The string makes visible emissive bulbs and, optionally, a real light every few bulbs.

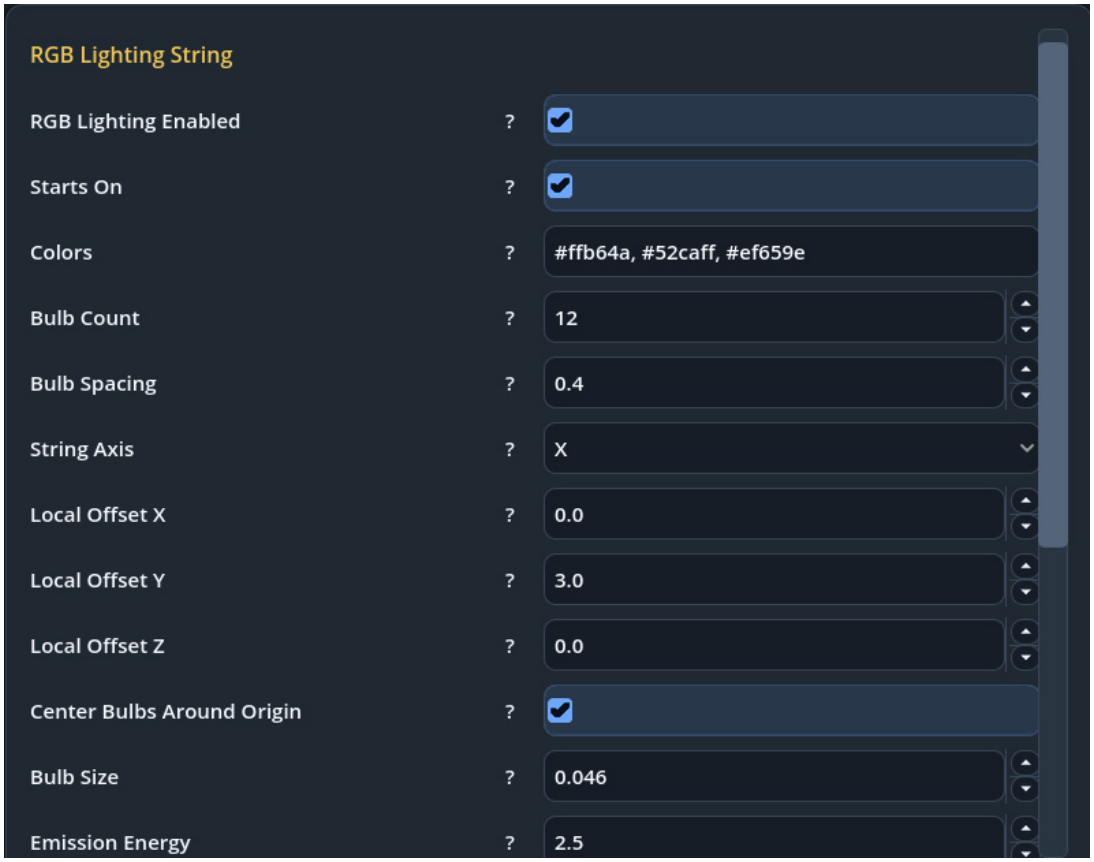

Figure 19. A string’s color sequence, count, spacing and local placement are authored together.

1. Use Add to Scene → RGB Lighting, or select an object and open RGB Lighting.
2. Enable it. Enter comma-separated colors, for example #ffb64a, #52caff, #ef659e.
3. Start with 12 bulbs, spacing 0.4 and axis X. Center them around the origin and use Local Offset to place them above a doorway.
4. Keep Bulb Size small, such as 0.045 in a human-scale room. Apply and Test Scene.

> The Inspector permits a larger bulb count, but this runtime clamps a string to 128 bulbs. Use a moderate count rather than relying on values above that cap.

54 RGB lighting, animation and cost

Control	What to choose
Emission Energy	Brightness of the bulb surfaces. Try 2.5 before raising it.
Light Energy / Range	The illumination cast by generated bulb lights. Keep range near the string.
Create A Light Every N Bulbs	0 creates no real lights. 3 makes a light on every third bulb, starting with the first.
Cycle Colors / Speed	Animates the color sequence. Use slow movement for background decoration.
Starts On / Optional On-Off Variable	Choose the initial state or link the string to a game variable.

Generated bulb lights do not cast shadows. They are decorative local illumination. Twelve glowing bulbs with a light every third bulb make four actual local lights; twelve lights are not needed just to make all twelve bulbs look bright.

1. Try Light Every N Bulbs 0 first and judge the visible string.
2. Add occasional lights only if the nearby wall needs colored illumination.
3. Turn Scene Glow on for a restrained halo. Check the string from the widest camera as well as close-up.
4. Walk past it while testing the target renderer. Watch for light popping when many lights overlap the same mesh.

Keep cycling subtle in rooms with dialogue. A rapidly changing bright object beside the speaker will compete with every line of text.

55 Lens flares: a visible light source

Lens flare adds a small camera-facing core, halo and starburst around a light’s position. It is an optical decoration. It does not mean that the surrounding air contains fog, and it does not illuminate objects by itself.

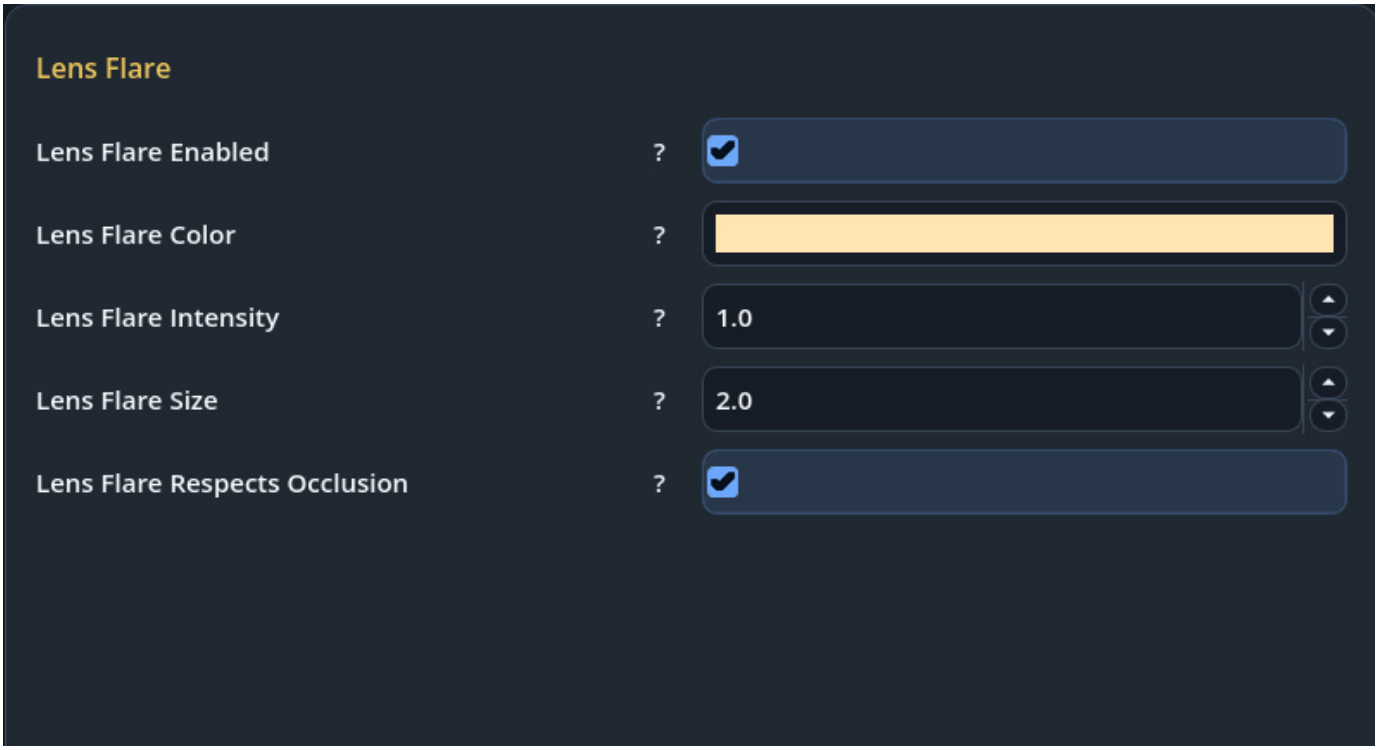

Figure 20. Flare color, strength, size and occlusion are separate from the light’s main intensity.

1. Select the light → Lens Flare. Enable the effect.
2. Begin with Intensity 1 and Size 1. Adjust color to suit the source.
3. Keep Respects Occlusion on so scenery can hide it.
4. Test from all gameplay cameras. Check the source partly hidden behind a wall or lampshade.

Use a modest flare on an exposed lamp or distant bright source. A large flare on every light quickly overwhelms a small room. If the flare appears through a wall, check occlusion before moving or deleting the light.

56 God rays and cinematic visible shafts

The God Rays tab combines two tools. Native volumetric fog scatters light through the room’s air and requires Forward+. Cinematic Visible Shafts add stylized crossed beam planes to spot and directional lights. Those planes are a separate visual layer.



Figure 21. God Rays controls include the volumetric light contribution and visible-shaft options.

Begin with a spot or directional light aimed through a useful opening. Enable God Rays. Try density 0.0001, distance 10-20 for a small room and the default directionality 0.65. Keep the rest of the room’s fog restrained.

Cinematic Visible Shafts can make beams easier to read in a stylized adventure. Reduce Shaft Opacity before adding a large count. A point light does not receive the crossed shaft geometry, even when its God Rays flag is enabled.

> A ray setting is not a replacement for a window opening. If a solid wall blocks the intended path, fix the light placement or architecture. Test visibility and depth occlusion from the game camera.

57 Tune fog scattering and visible beams

Control	Meaning
God Rays Light Energy	How strongly this light contributes to volumetric fog. The default is large because the default density is very low.
Fog Density / Distance	Amount of participating fog and its covered distance. Dense fog can erase the room.
Directionality	How strongly scattering favors a viewing direction. Check looking toward and away from the source.
Sky Affect	How the fog affects the sky contribution.
Visible Shaft Count / Width	Number and thickness of decorative beams. Start around 5 narrow shafts.
Visible Shaft Opacity / Spread	Brightness and separation of the crossed planes. Try opacity 0.1-0.2 before increasing it.

Several enabled God Rays lights can raise the room’s requested volumetric settings. The combined result is not an isolated fog box for each light. Test the complete room, not just one light at a time.

The visible planes respect scene depth, but they remain stylized geometry rather than a physical simulation of dust. At certain side angles their construction can become obvious. Tune them for the cameras the player will actually use.

> Use the Rendering details window to distinguish Enabled from Unavailable. A requested volumetric effect cannot run in Compatibility or Mobile.

58 Ordinary depth fog and height fog

Ordinary fog is available across PCA’s renderers and is much cheaper than volumetric scattering. It changes the picture with distance and can soften a large room or conceal a distant boundary.

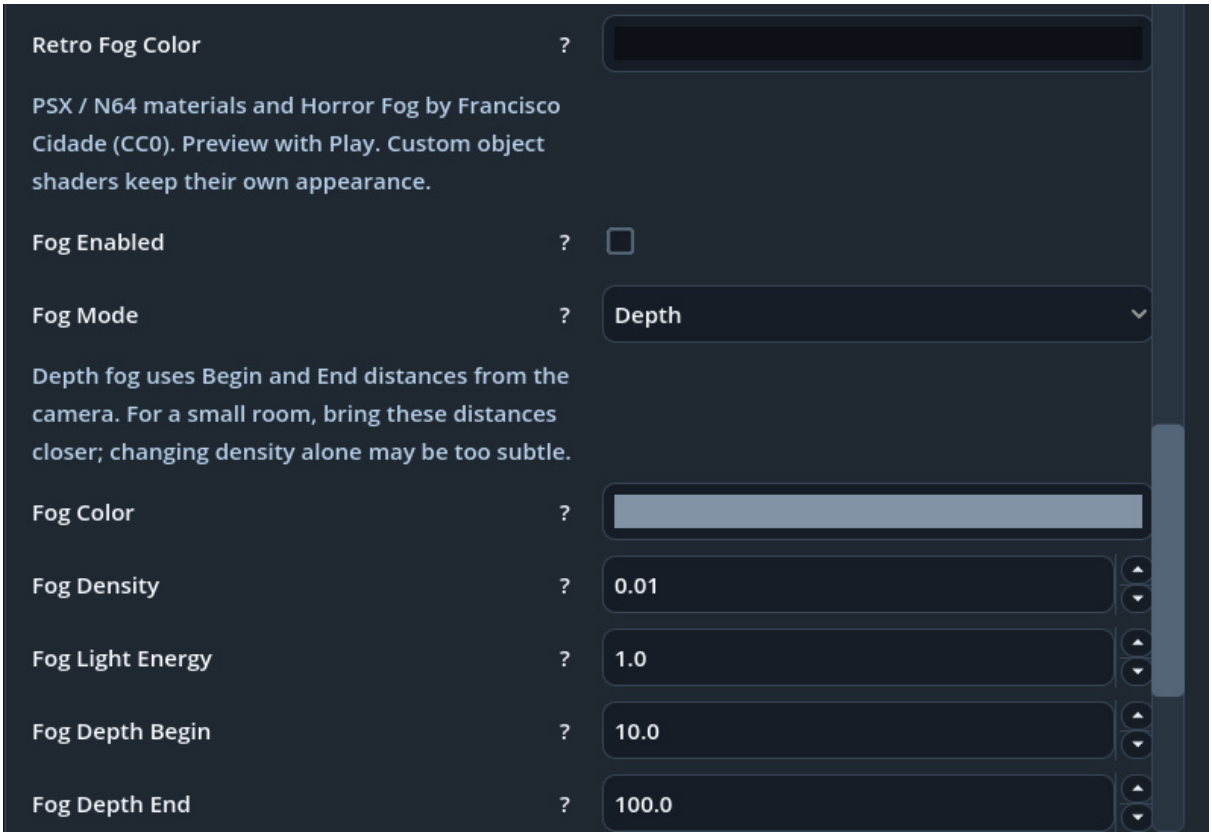

Figure 22. Bring the depth distances into the scale of your room. The default far distances can be too subtle indoors.

1. Select the room → Lighting & Fog. Enable Fog and choose depth.
2. Choose a muted fog color. For a small room, try Begin 6 and End 22, then adjust for the camera’s distance.
3. Tune Density and Depth Curve gently. Keep foreground interaction areas readable.
4. Use Fog Height and Height Density only if the scene needs a vertical change. Test above and below the chosen height.

Exponential mode uses a different distance distribution from the Begin/End depth mode. Do not expect equal-looking numeric values to produce equal pictures. Fog Light Energy, Sun Scatter, Sky Affect and Aerial Perspective adjust how fog relates to illumination and the distant background.

59 A flashlight reveal and 2D lighting

Set Flashlight Mode creates a screen-space reveal around the pointer. It affects the rendered room while leaving the higher GUI layer readable. It is a useful adventure-game reveal effect, not a physical 3D spot light attached to the character’s hand.

```
await SetFlashlightMode({
    "enabled":true, "strength":1.0,
    "radius":180.0, "shape":"circle"
})
```

Radius is a screen-space value. Shapes include circle, spherical, square, rectangle and triangle. Turn the mode off explicitly when leaving the situation that needs it. For a torch that casts light and shadows in world space, use an appropriate light and attachment workflow instead.

Older native Canvas2D rooms can use PointLight2D-style lights with color, energy, radius and a shadow switch. Their lit appearance depends on the receiving canvas items and any authored occlusion. The current native Canvas2D renderer does not automatically create light-occluder polygons from a background or navigation area. A shadow switch alone will not make painted shadows dynamic.

> 2D Stage and the older Canvas2D compositor are different room systems. Use the room’s own inspector and test its actual rendering path. Do not assume a 3D shadow-map setting will create matching 2D occluders.

60 Surface shine, room capture and mirrors

Reflections need the right method for the job. A polished door that reflects a lamp highlight is not necessarily showing the player’s reflection. Choose the method before adjusting the strength.

Method	What it can show	Main limit
Material shine	A highlight from lights and the environment.	It is not a separate live view of the character.
Room Capture	Nearby room geometry captured for reflective materials when the room loads.	It is a one-time capture, not continuous character tracking.
Enhanced screen-space reflections	Reflections derived from what the main camera can currently see.	Offscreen or hidden information is missing.
Live Mirror	A reflected camera view of the scene, including a moving character.	Adds another render and is intended for a flat mirror surface.

For a decorative polished sphere, Room Capture may be enough. For a reflective floor on supported hardware, Enhanced SSR can help. For a mirror that must show the character moving in front of it, use Live Mirror.

> Turning Metallic and Reflection Strength up cannot turn a one-time room capture into a continuously updated mirror.

61 Metallic, roughness and reflection strength

Select an object → Visual Effects → Material and Reflection. These controls tune how the surface uses the available reflection method.

Control	Practical use
Metallic	0 is a nonmetal-like surface; 1 is metal-like. Start high for brass and low for painted wood.
Roughness	Low values make sharper highlights and reflections. Higher values spread and soften them.
Reflection Strength	Controls the reflection contribution. Start near 0.6 for a deliberately shiny test surface.
Reflection Color	Tints the reflection. Begin with a pale neutral color rather than a strong saturated tint.
Reflection Mode	Choose Room Capture for ordinary environmental reflection or Live Mirror for a planar mirror.

Test on a simple sphere with a clear nearby colored object. Lower roughness to about 0.1-0.2 and make sure the scene has something to reflect. A dark featureless room naturally produces a dark reflection.

Imported materials and custom shaders can affect the result. Check whether a model has its own shader or material override before assuming that an Inspector value is controlling every visible surface.

A material can be reflective without looking like chrome. Use restrained values for varnished wood and painted metal rather than giving every surface the same mirror finish.

62 Set up a room reflection capture

1. In Global Settings → General, enable Room Reflection Probe.
2. Select the reflective object. Use Room Capture, a nonzero Reflection Strength and low Roughness.
3. Select the room → Lighting & Fog. Check the Reflection Probe Position controls. Put the capture where it can see the room, rather than inside a wall.
4. Apply, save and restart Test Scene. The capture is taken when the room loads.

In the ordinary profile, PCA fits the probe around relevant reflective surfaces and room content. A manual probe position changes the capture origin. In Enhanced Rendering, Reflection Probe Size sets a cube around the capture position.

Box projection helps an interior capture fit the room. It remains an approximation from a chosen viewpoint. A single capture cannot produce a perfect physically correct reflection for every object at every place in a large room.

On Compatibility, the capture excludes certain helper layers and shadowed positional-light contributions to avoid a blank cubemap problem. The reflected lighting can therefore differ from the main view. Judge a test with a simple unshadowed fill as well.

After changing the room arrangement or capture origin, reload the room. Moving a character after capture does not make the probe update every frame.

63 Create a mirror that reflects the character

1. Add a thin box to the room. For a wall mirror, try Size [2.7, 2.5, 0.08]. Place it vertically on the wall with its broad face toward the player.
2. Select it → Visual Effects. Set Reflection Mode to Live Mirror.
3. Set Reflection Strength to 1, Roughness to 0 for the test and Reflection Color to a pale neutral tint.
4. Leave Mirror Axis Automatic if the thin dimension is unambiguous. For this box, local Z is the thin axis.
5. Apply, restart Test Scene and walk a character across the mirror. Watch the reflected image change.

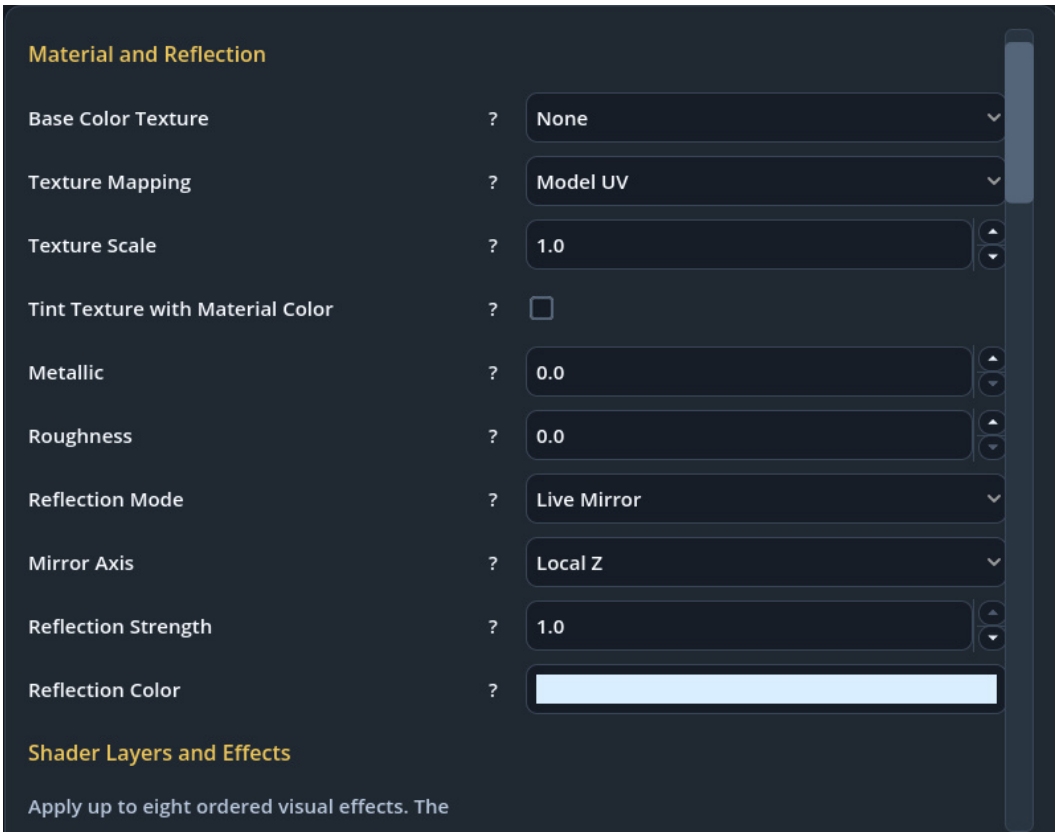

Figure 23. Live Mirror is a separate reflection mode. The axis describes the mirror plane’s local normal.

Use a primitive, procedural object or noncharacter imported model. The Live Mirror path does not turn a character model into a mirror.

64 Mirror orientation, blur and performance

Automatic chooses the thinnest local dimension as the plane’s normal. Choose X, Y or Z explicitly when the model’s dimensions make that ambiguous. For a floor mirror, Y is commonly the thin local axis; for a vertical thin box, it may be Z or X.

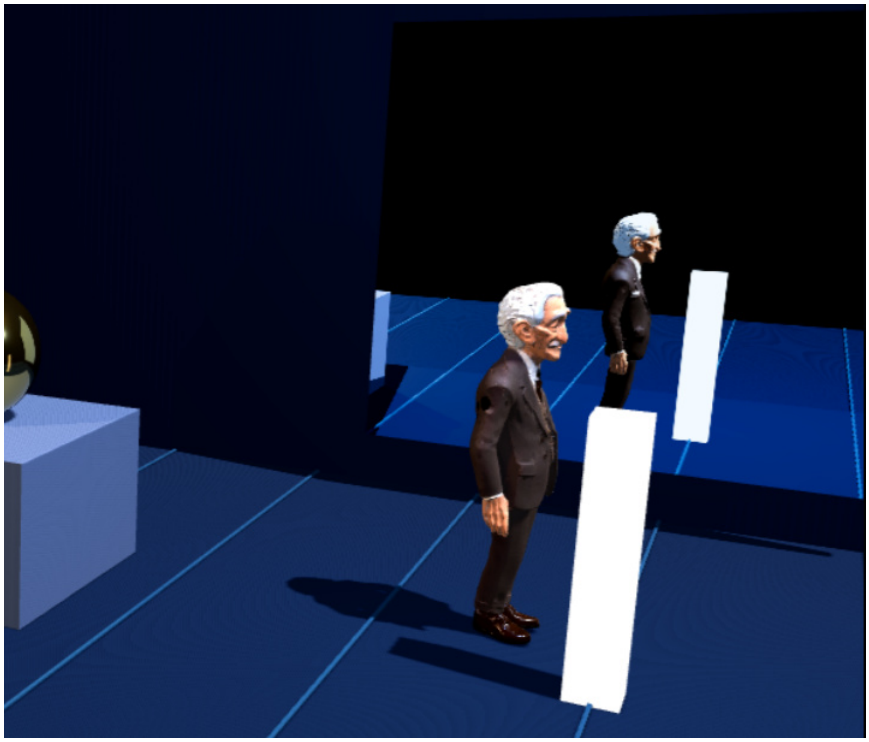

Figure 24. Elias and his live reflection after he moves toward the mirror. The reflected character changes with the live scene.

The mirror plane follows the object’s transform. Rotate the object, then check its local axis rather than treating the world’s Z direction as a universal answer. Very thick or irregular models produce a poor plane estimate; use a clean flat surface for a convincing mirror.

Roughness softens the reflected view. Reflection Strength, tint alpha and object opacity all affect visibility. If their combined contribution is zero, the reflection will disappear even though the mode is set to Live Mirror.

Each visible mirror adds a reflected render. This build sizes the capture to a maximum dimension of 512 pixels and avoids unnecessary offscreen captures. It is not unlimited-resolution ray tracing or recursive mirror-in-mirror rendering.

Use a few mirrors with a clear purpose. Test a room with every mirror visible at once before treating the feature as inexpensive decoration.

65 Diagnose a missing reflection

Symptom	Check in this order
Only a bright lamp spot appears	You may be seeing a specular highlight. Choose Live Mirror if you need a moving character reflected.
Room reflection remains unchanged	Room Capture updates when the room loads. Reload after edits; use a live method for moving subjects.
Mirror is completely dark	Strength, tint alpha, object opacity, correct thin axis, visible flat face, and whether reflected geometry is actually lit.
Reflection appears on the wrong face	Check local Mirror Axis and the model’s transform. Try a thin primitive as a control test.
Reflection is soft	Lower Roughness. Remember the live capture has a finite resolution.
SSR loses a character near the screen edge	Screen-space information is leaving the main camera view. Use Live Mirror where a stable character reflection is required.
Imported object ignores changes	Check custom materials and shaders. Compare with a standard primitive using the same settings.

Use a simple mirror and one character first. Add the decorative frame, heavy grading and surrounding props only after the basic reflection is correct.

66 Hover outlines and attention feedback

Pointer Feedback controls how an interactable object responds visually to the pointer. It is independent of the whole-room outline. Select the scene object, not only its shared character definition.

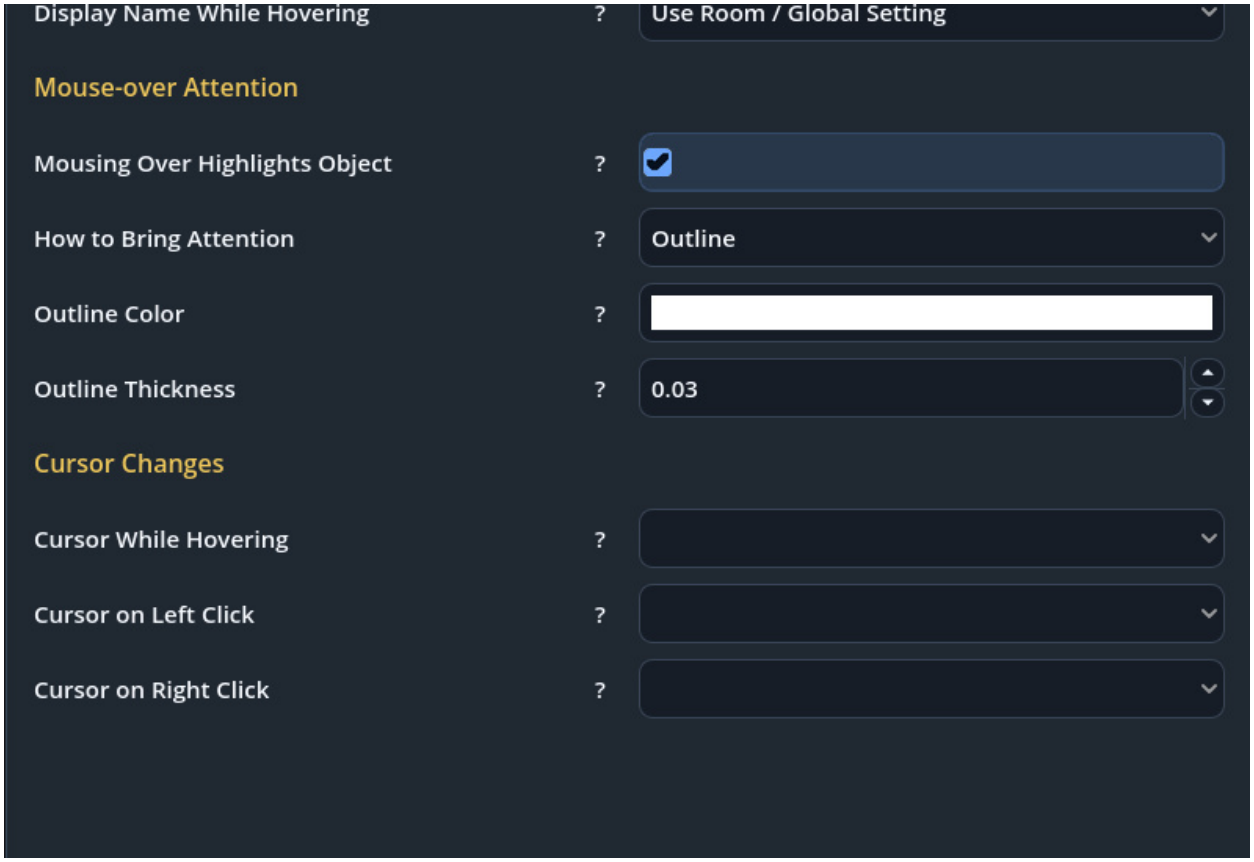

Figure 25. Choose outline, bloom or tint for hover attention on the selected scene object.

1. Open Pointer Feedback. Enable Mousing Over Highlights Object.
2. Choose Outline under How to Bring Attention. Set a clear color and begin near thickness 0.015-0.03.
3. Apply and Test Scene. Hover the actual interactable surface.
4. Try Bloom / Glow or RGB Tint if a border does not suit the artwork. Tune the controls that appear for that mode.

Keep the hotspot enabled and ensure another invisible object is not intercepting the pointer. A visible model with no usable interaction target may never receive hover feedback. Display Name While Hovering is a separate option.

67 Whole-scene silhouette outlines

1. Select the room → Post Process. Scroll to Whole-Scene Silhouette Outline.
2. Enable Outline Everything in This Room. Leave Edges Only off for the first test.
3. Choose a color that separates shapes from their background. Begin around thickness 0.015-0.03.
4. Apply, restart Test Scene and inspect both the wide shot and a close-up.

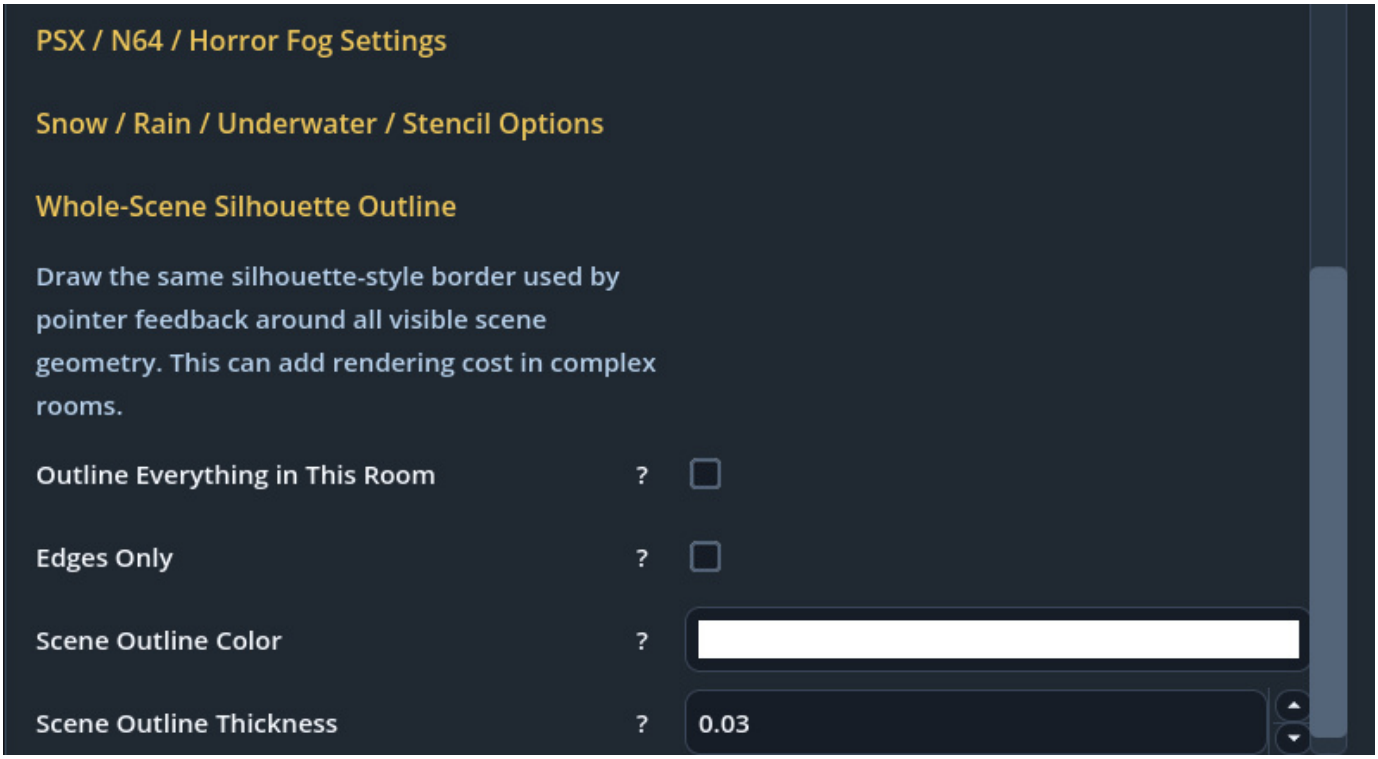

Figure 26. The room outline has its own switch, edge mode, color and thickness.

This applies a common silhouette treatment across eligible scene geometry. It excludes light, trigger and particle helpers rather than outlining editor markers as part of the game. It is a scene presentation effect, not an interaction state.

Whole-room outlines add rendering work. Dense imported meshes and several overlaid effects can make the result busy. Test thin furniture, hair, door frames and other small features before deciding on the final thickness.

68 Edges Only: what it draws

Edges Only looks for depth discontinuities around visible geometry. It can keep a border where one object’s depth differs from its surroundings while avoiding a filled-looking treatment across a broad continuous surface.

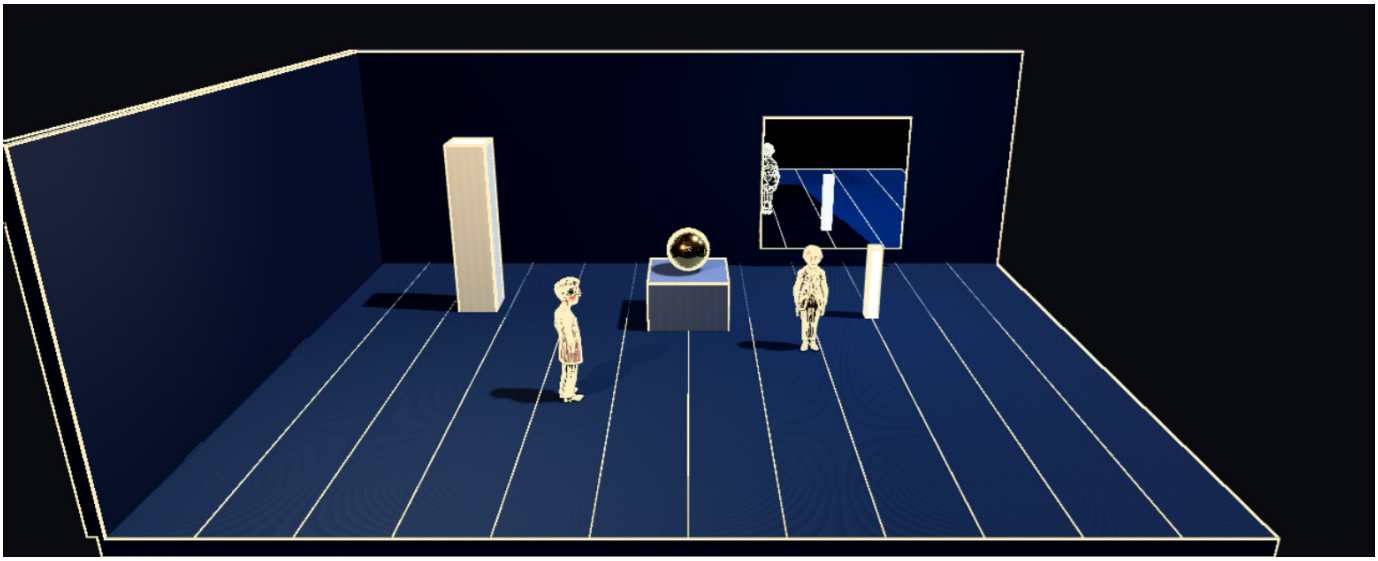

Figure 27. Edges Only in the running room. Thin separate floor strips also have their own visible depth edges.

It is not wireframe and it does not draw every triangle edge. It also does not reveal hidden edges behind walls. Coplanar adjoining surfaces may have no useful depth break for it to detect.

Transparent materials that do not write depth may produce incomplete results. Fine character geometry can create a dense border at large thickness values; reduce thickness or use hover-only attention for that artwork.

> The authored thickness is converted to a pixel-width range and clamped in the effect. Do not assume it is a literal thickness in scene meters. Always compare at the intended output resolution.

69 Choose one outline language for the game

There are several ways to create borders. Use each for a clear purpose. Layering all of them at once can turn a thin outline into a thick, flickering or inconsistent edge.

System	Good use
Pointer outline	Tell the player which object is under the pointer.
Whole-scene silhouette	Give the room a common illustrated border.
Whole-scene Edges Only	Emphasize visible depth boundaries.
Screen shader edge / toon / ink effect	Stylize the final room image using that shader’s own edge rules.
Custom object shader	Give a particular object a special rendering treatment.

1. Decide whether ordinary scenery should have outlines at all.
2. If it does, choose a restrained room border. Make the hover response different enough to remain meaningful, perhaps with a color change or mild tint.
3. Check bright and dark rooms. A white border that works at night may disappear against a bright sky.
4. Test at the smallest supported resolution, where a few pixels occupy more of a face or inventory target.

Do not use a beautiful border as the only clue for an important interaction. A meaningful cursor, name label or clear object placement can help the player understand the same object.

70 Choose the renderer for the actual game

Open Global Settings → General → Renderer. PCA offers Forward+, Forward+ Enhanced, Mobile and Compatibility / WebGL. Choose according to the required effects and the hardware you intend to support.

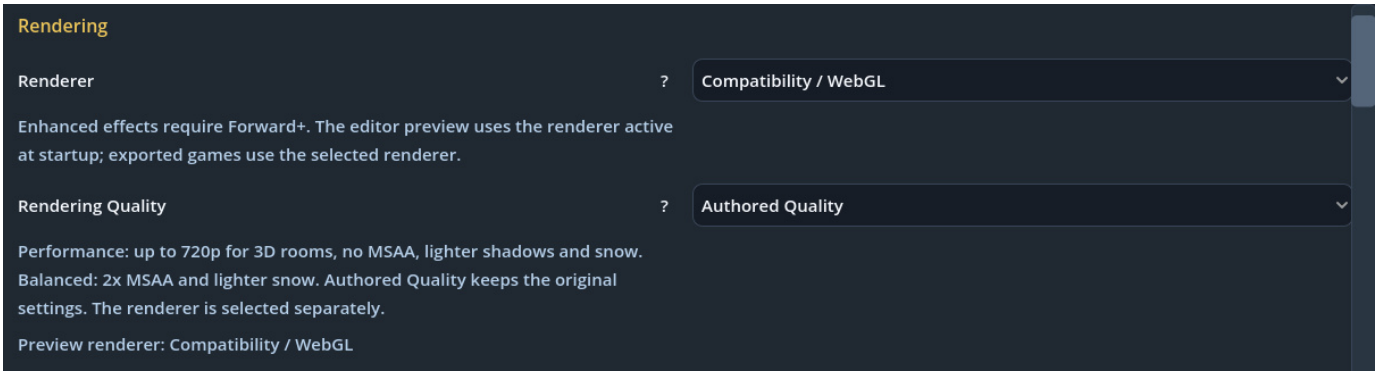

Figure 28. Renderer and Rendering Quality are separate settings; the live preview label appears below them.

Choice	Use it when
Compatibility / WebGL	You want broad hardware support or web output and can work without native DOF and volumetric fog.
Mobile	You need a lighter modern 3D renderer with depth of field, but not Forward+-only indirect and volumetric features.
Forward+	You need the more advanced rendering backend, such as volumetric fog.
Forward+ Enhanced	You want PCA’s optional SDFGI, screen-space reflections, ambient occlusion and probe configuration on Forward+.

Enhanced is a PCA profile on the Forward+ backend, not a fourth independent Godot graphics backend. It still requires Forward+ to be the renderer that actually started.

71 Restart the renderer, not just the preview

1. Select the game Renderer in Global Settings and click Apply Changes.
2. Click Use selected renderer at next editor launch.
3. Save the project, stop the preview and close PCA 3D completely.
4. Launch PCA 3D again. Open the project and inspect Preview renderer before judging effects.
5. Test an exported build too. The exported game uses its selected rendering configuration and the hardware available at launch.

Play / Restart only rebuilds the game preview inside the running editor. It cannot swap the process’s graphics backend. That is why checking a DOF or volumetric checkbox can appear to do nothing after changing Renderer.

A driver or platform limitation can make a requested renderer unavailable and cause fallback. A source or command-line launch can also explicitly select a renderer. Trust the live Preview renderer label rather than assuming that the saved game setting is what the process is using.

> Keep the full exported folder together. Its rendering configuration and supporting files are part of the build; copying only the executable is not a reliable distribution test.

72 Read Rendering details

In the game preview, click the Preview renderer button below the picture. Rendering details lists the active preview renderer, saved game renderer, active camera and relevant effect availability.

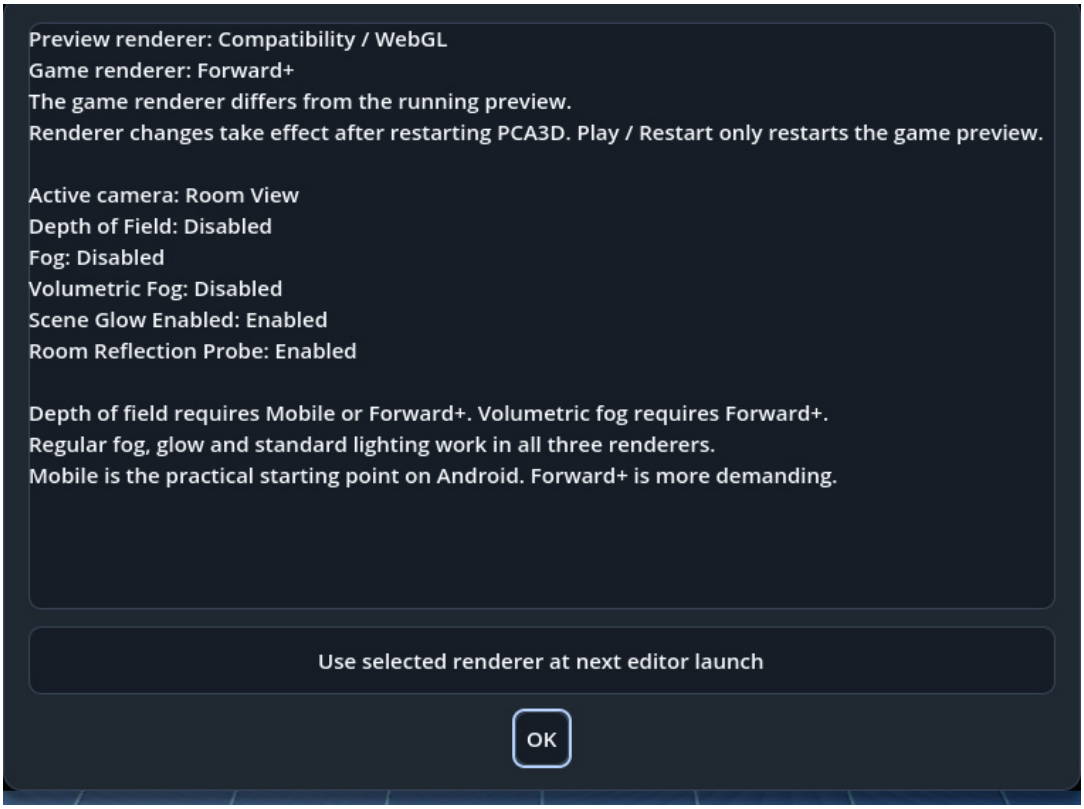

Figure 29. The saved Forward+ game setting and a live Compatibility preview are shown separately after renderer fallback.

Status or message	What to do
Disabled	The effect is off in the active settings. Enable the right camera or room setting if you want it.
Enabled	The effect is requested and supported by this rendering path. Its strength, distances and content still determine visibility.
Unavailable	Use a supported renderer or viewport arrangement; intensity alone cannot solve it.
Game renderer differs from preview	Save the startup renderer and restart the entire editor before comparing.

The example screenshots in this guide were captured with Compatibility where the live status says so. Camera motion, ordinary lighting, mirrors and outlines are shown in that renderer. Forward+-only effects should be checked on your intended hardware; a control screenshot is not presented as a rendered demonstration of an unavailable effect.

73 Feature availability at a glance

PCA feature	Compatibility	Mobile	Forward+
Ordinary lights and shadows	Yes	Yes	Yes
Regular room fog	Yes	Yes	Yes
Scene glow	Yes[1]	Yes	Yes
Room reflection probe	Yes	Yes	Yes
Live planar mirror	Yes	Yes[2]	Yes[2]
Native depth of field	No	Yes	Yes
Native volumetric fog	No	No	Yes
PCA Enhanced SDFGI	No	No	Enhanced profile
PCA Enhanced SSR	No	No	Enhanced profile
PCA Enhanced SSAO	No[3]	No	Enhanced profile

[1] The glow pipeline can look different across backends. Compare the actual target output. [2] Live mirrors add another scene render; test the device’s performance. [3] The underlying engine may support other routes, but PCA’s Enhanced SSAO configuration is gated to the Enhanced profile on Forward+.

This table describes the PCA controls in this version, not every feature that a custom Godot source project could add. A transparent layered viewport can impose an extra limit, notably for native depth of field.

> A modern renderer running successfully is only the first requirement. A device still needs enough speed and memory for the selected lights, shadows, materials and resolution.

74 Quality policies: Authored, Balanced and Performance

Rendering Quality sits beside the renderer selection. It is a set of practical performance rules. It does not delete game systems, remove actions or strip imported character animations.

Policy	Changes in this version
Authored Quality	Keeps authored shadow settings and uses 4× MSAA when antialiasing is enabled.
Balanced	Uses 2× MSAA; reduces high/ultra shadow quality to medium; uses lighter snow; reduces four directional splits to two and caps shadow distance at 65.
Performance	No MSAA; aspect-fits 3D room rendering within 1280 × 720; low shadows unless hard; lighter snow; two directional splits, distance cap 45 and no angular softness.

GUI and overlay resolution can be kept separate from the room viewport using the presentation settings. That helps text remain readable while a 3D room renders at a lower resolution.

Simplify Generated Geometry is a separate per-primitive/procedural option. It reduces generated spheres and cylinders. It does not rewrite imported meshes, skeletons or animation files.

> Test the complete exported game in the selected policy. A smaller or faster build is not evidence that camera or gameplay features were removed.

75 Forward+ Enhanced: enable only what you need

1. Select Forward+ Enhanced in Global Settings. Save the startup renderer and restart the editor as described earlier.
2. Confirm Preview renderer is Forward+. Start Enhanced Quality at low or medium.
3. Enable one Enhanced feature at a time: Indirect Lighting, Screen-space Reflections or Ambient Occlusion.
4. Test the same camera before and after. Keep only features that improve the actual game picture enough to justify their cost.

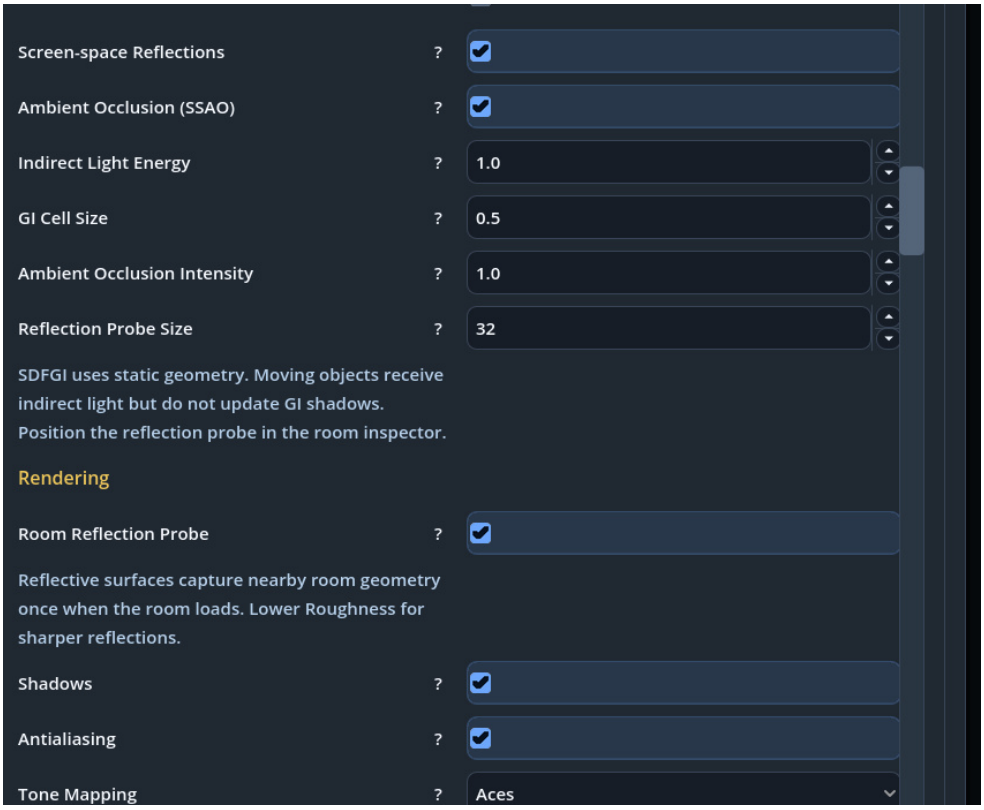

Figure 30. Enhanced Rendering has independent toggles and tuning values rather than one compulsory all-effects setting.

Enhanced Quality selects a group of internal settings. Low uses two SDFGI cascades and 32 SSR ray steps; medium uses four and 64; high uses six and 128. Higher values can improve coverage or reflection detail at a cost.

Do not raise Enhanced Quality to repair a missing mirror or an incorrectly aimed light. Begin with a clear reason for enabling each component.

76 Indirect lighting and static geometry

Indirect Lighting (SDFGI) adds bounced light from the scene. It can make an interior feel connected: a lit wall influences nearby surfaces instead of leaving every unlit face dependent on a flat fill.

Control	Use
Indirect Light Energy	Overall bounced-light strength. Start at 1.
GI Cell Size	Spatial detail of the representation. Smaller values can capture smaller features but increase the work and coverage tradeoff. Start around 0.5.
GI Uses Sky Light	Allows sky contribution. Useful outdoors; compare it off in a closed interior.
GI Occlusion	Adds occlusion handling for the GI representation. Compare carefully with thin architecture.
Enhanced Quality	Changes cascade coverage/detail settings. Begin low or medium.

SDFGI is based on static geometry. Moving characters receive indirect light but do not continuously update the static GI occlusion as though they were a new wall. PCA marks characters and its known continuous rotation/scale motion as dynamic in the Enhanced profile.

For a prop moved by your own script, set its shadow.optimization to dynamic in its authored record. Disabled excludes that geometry from GI participation. Keep solid architectural walls static.

> This does not add a PCA lightmap-baking editor, VoxelGI workflow or ray-traced moving GI. Those are separate engine concepts, not extra buttons promised by this profile.

77 Screen-space reflections and ambient occlusion

Screen-space Reflections can add useful moving reflection detail to polished surfaces. They work from information available in the main camera image, so they cannot reliably reflect a character or wall that the camera does not see.

1. In Enhanced Rendering, enable SSR. Use a low-roughness floor and a clear object visible above it.
2. Move the object and change camera angle. Watch the reflection near screen boundaries.
3. Use a room probe as supporting environment reflection where appropriate. Use Live Mirror if a particular surface must show a stable reflected character.
4. Enable SSAO separately. Start AO Intensity around 0.5-1 and check corners, feet and closely spaced surfaces.

Ambient Occlusion darkens contact and crevice-like areas to improve depth. It does not replace a cast shadow, create a light source or repair a model floating above the floor. Too much makes every corner look dirty.

SSR and SSAO settings in this PCA profile activate only when Forward+ Enhanced is selected and the actual backend is Forward+. A checked option in a Compatibility preview is not a working Enhanced effect.

78 Tone mapping and cinematic grading

Tone Mapping decides how the lighting range is mapped into the visible picture. Global Settings offers linear, reinhard, filmic and aces. Pick a baseline before polishing light intensities, because a different mapper can change the apparent highlight strength.

Cinematic Presentation is a separate finishing pass. Set it globally, then let each room inherit, disable it or use a custom override. It offers Exposure in EV, Contrast, Saturation, Warmth, Vignette, Film Grain and Letterbox Height per edge.

Starting adjustment	Try
Exposure	0 first. Adjust in small increments once lighting is correct.
Contrast / Saturation	1 is neutral. Try contrast 1.03 and saturation 0.9 for a restrained look.
Warmth	A small positive or negative adjustment, rather than a strong color wash.
Vignette / Grain	Begin near 0.1 / 0.01. Check dark objects and text readability.
Letterbox	Use only when the intended camera frame leaves room for the covered edges.

Letterbox covers the world edges while the GUI remains on its own layer. It does not automatically recompose a shot. Recheck heads and clickable objects after adding bars.

> Avoid compensating for excessive contrast or a dark vignette by adding many more lights. Disable the grading briefly and inspect the ungraded room.

79 Screen shaders and retro rendering

Room Post Process contains the screen shader stack and its room override. Global scene shaders provide the inherited default. These passes affect the rendered room; GUI and screen overlays remain separate unless their own settings request the room treatment.

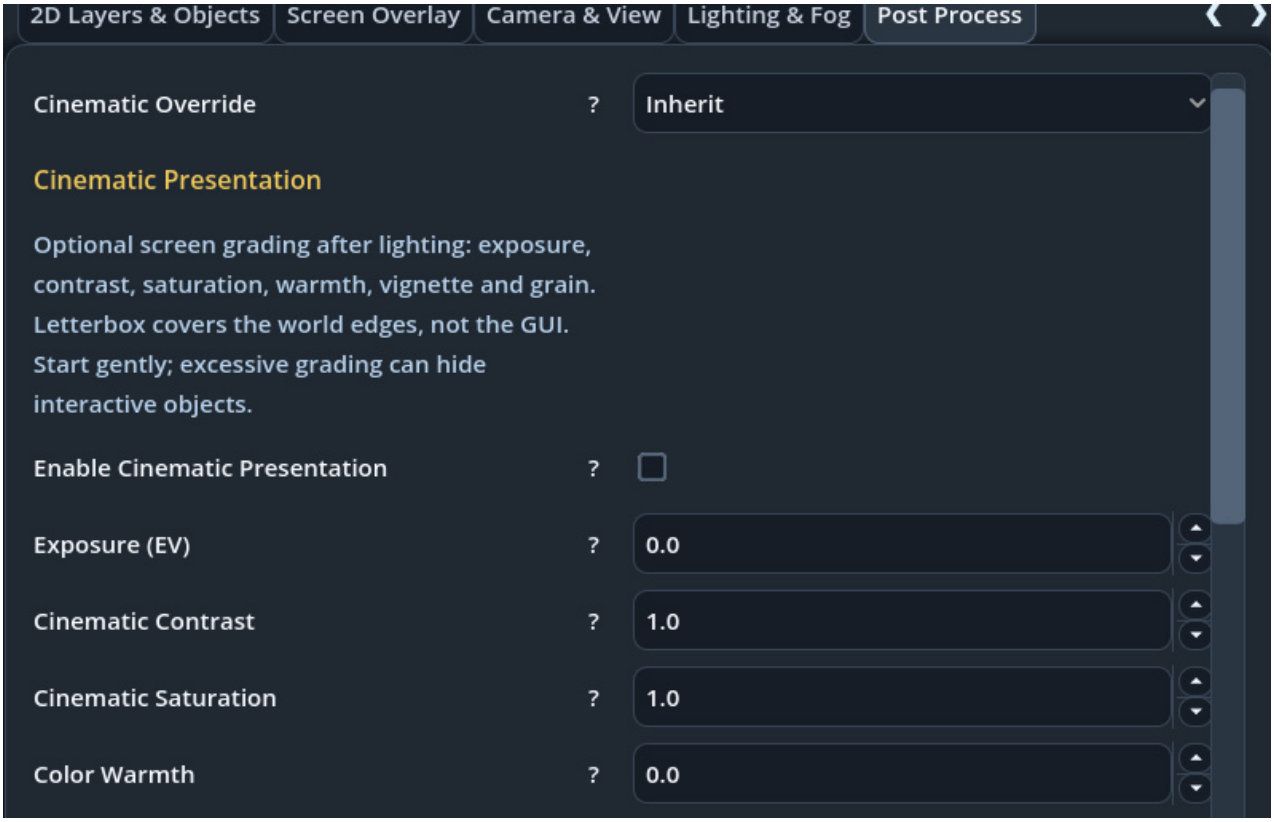

Figure 31. Cinematic grading and the room screen stack are edited in Post Process.

Use a PSX screen preset for pixelation, limited color precision and dithering; an N64 preset for its filtering treatment; or Horror Fog for its screen presentation and optional depth-related settings. Start with one preset and modest intensity before combining layers.

The separate Retro Rendering controls can change 3D material appearance. They are not the same as the final screen presets. Use the screen stack when you want a camera-wide effect. Use material treatment when the object surfaces themselves need that style.

The stack supports up to eight layers and can use an imported shader where the current source selection permits it. Layer order matters. Distortion before pixelation is not the same picture as pixelation before distortion.

> Do not use Horror Fog as a substitute for checking ordinary depth fog or native volumetric availability. Each system has its own controls and rendering rules.

80 Zoom blur for a short dramatic beat

Zoom Blur is a screen effect that streaks the picture around a chosen center. Use it briefly for a sudden realization, a transition or a dramatic event. It does not move the actual camera, change its FOV or replace ordinary depth of field.

```
await ZoomBlur({
    "strength":0.25, "duration":0.45,
    "centerX":0.5, "centerY":0.45,
    "clearRadius":0.12, "direction":"inward",
    "includeGUI":false, "wait":true
})
```

Control	Meaning
Strength	0-1. Begin around 0.2-0.3 rather than the strongest effect.
Duration	Seconds the effect runs. Keep a dramatic punctuation short.
Center X / Y	0-1 screen fractions. [0.5, 0.5] is the center of the picture.
Clear Center Radius	Keeps a central area less affected so the subject remains recognizable.
Direction	Inward or outward streaking.
Include GUI	Normally off so text and buttons stay readable.
Wait Until Finished	Hold a sequential action list until the effect ends. Use false to run other events beside it.

Use the Zoom Blur action or node, or its script wrapper as shown above. Stop Zoom Blur cancels it. Test the effect over the actual dialogue and HUD before using it during a moment that asks the player to read or click.

81 Objects, regions, overlays and particles

Several local or layered systems can contribute to the final picture. Give each one a clear role so you can find the cause when the result changes.

System	Use and boundary
Object shader stack	Stylize one object. Custom materials may change how it reacts to lighting and shadows.
Region visual effect	A local authored area can tint, emit, look metallic, mirror or use a shader. Its geometry and effect settings determine the visible area. It is not automatically a full volumetric light field.
Particles	Fire, sparks, snow and other moving visual effects. A bright particle does not automatically create a room light.
Attach points	Keep an emitter or prop related to a character’s animated hand or another named attachment. Check local offset and orientation.
Screen overlay	A separate image or effect layer. It may remain unaffected by the room’s screen stack.
Parallax	Moves layers relative to the camera to suggest depth. It does not add physical geometry or shadow casting to a painted image.

For a glowing torch, combine an attached emitter, a deliberately chosen glow appearance and a restrained real light. Test all three together. For a region mirror, use a suitable flat visible surface and verify that the chosen reflection mode supplies the result you need.

82 Workshop recipe: a finished gameplay view

Use a room you already have. The goal is a dependable playable view, not a beauty shot that hides an exit. In the example, the floor is about twelve units wide and the characters are human scale.

1. Temporarily turn off depth of field, fog, cinematic grading and room screen effects.
2. Create wide at [0, 5.5, 11.6], aimed at [0, 1, -1], FOV 46. Set it as the room’s default camera.
3. Choose fixed behavior. Use a named camera even if there will be only one gameplay view at first.
4. Set ambient intensity near 0.32. Add the warm key and cool fill from the earlier recipe, beginning with shadows on only for the key.
5. Put a tall primitive beside a low object. Check the shadow direction and the visibility of both.
6. Move the player to every navigation corner. Make sure exits, interactive props and the character remain readable behind the HUD.
7. Add mild grading last. Save, restart Test Scene and compare against the ungraded version.

If the set’s edge appears, first decide whether the camera should show less, the set should extend farther or the player should be restricted to a smaller navigation area. Those are different design choices. Do not use a camera-bound workaround to disguise a walking problem.

> Keep a simple neutral primitive in your private test room while developing the look. It is a useful reference for light color, shadow behavior and material response.

83 Conversation recipe: two actors and a return

1. Keep wide as the gameplay default. Create conversation with a different angle that shows both speakers and stays clear of walls.
2. Set conversation to group, primary target player and additional target elias. Enable auto-framing, min 3, max 14, padding 1.3 and damping 0.2.
3. Activate it with a 1.2-second sine blend. Let the camera settle before the first important line.
4. Use a close-up only for a line or gesture that benefits from it. Return to the group view before a broad interaction such as a handshake.
5. Return to wide at the end. If the conversation is a cutscene, enable Restore Camera and test Skip.
6. Repeat with the speakers at their most separated positions. Then test a different active player if your game supports switching.

For an over-the-shoulder angle, leave room between the near actor and the lens. If the foreground shoulder hides the speaker’s face, change the angle rather than trying to solve it with blur. Depth of field can soften the shoulder after the composition works.

Keep the lighting believable from both angles. A very strong frontal light may make one reverse shot look flat while the other looks dramatic. A moderate fill often produces a more consistent conversation.

A group controller follows its targets; it does not decide when a reaction shot is emotionally appropriate. Author those shot changes in the action flow or timeline.

84 Window-light recipe: beams without white fog

1. Start with a readable room and a visible opening. Add a spot or directional light whose angle matches that opening.
2. Make the direct light useful before enabling God Rays. Confirm the floor or wall receives the intended pool of light.
3. For physical volumetric fog, run Forward+. Confirm the actual renderer in Rendering details.
4. Use low density first. Enable God Rays on the light and increase its fog contribution only until a beam becomes noticeable.
5. If the adventure’s camera needs a stronger stylized beam, use Cinematic Visible Shafts. Try 5 shafts, width 0.3 and opacity 0.16.
6. Check every gameplay camera and the silhouette of the player. Reduce fog or opacity if interaction objects lose contrast.

Think of the beam as a guide for the eye. It can lead the viewer toward a door, workbench or speaking character. A beam across the entire HUD area or every corner of the room does the opposite.

If you use Compatibility, the decorative shaft layer can still provide a stylized beam, but it is not evidence that native volumetric scattering is running. Keep that distinction when preparing different platform builds.

> Avoid stacking dense ordinary fog, dense volumetric fog and strong Horror Fog until each has been tested separately. Three atmosphere systems at full strength usually produce one very confused room.

85 Glowing-prop recipe: bulb, halo and spill

1. Choose a small surface that should glow. Add its authored glow dictionary with enabled true, a chosen color and energy around 2-3.
2. Restart Test Scene and verify that the surface itself looks bright. Toggle it off and on with SetGlow to confirm that you are controlling the intended object.
3. Enable the room’s Scene Glow and begin with intensity 0.3. Keep the halo small enough to preserve the object’s shape.
4. Add a nearby point light with a similar color, modest intensity and a short range. Alternatively use an object bloom effect’s surrounding-light option.
5. Switch the real light off while keeping emission on. The surface should remain bright while nearby illumination changes.
6. Put both parts under the same intended switch logic. Use explicit SetGlow on/off states when reliable cleanup matters.

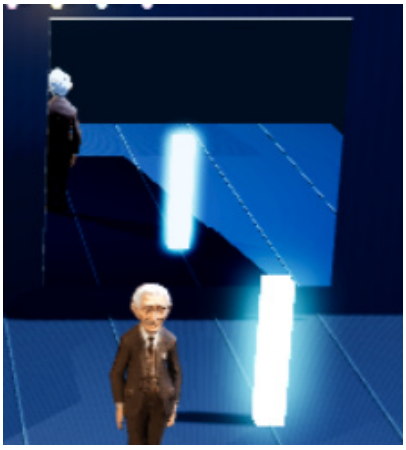

Figure 32. Scene Glow adds a halo to the cyan emitter; the mirror reflects the bright marker.

For a status lamp, the halo and surrounding light may not be necessary at all. For a magical crystal, the spill can be part of the clue. Give each component a purpose.

A glow texture is useful when only part of a model should emit. A whole machine washed in emission loses the shading that makes its shape readable. Isolate the display or bulb surface where possible.

> Emission, halo and spill are three tests. Keeping them separate while authoring saves a great deal of “why is this still dark?” clicking.

86 Mirror-room recipe and a proper movement test

1. Create a thin flat mirror surface with a standard primitive. Start without a custom shader or decorative frame.
2. Set Live Mirror, strength 1, roughness 0 and an appropriate local axis.
3. Place the player a few units in front of it. Use a fixed camera that can see both the player and the mirror.
4. Walk left and right. Check that the reflected character changes position and facing. Then walk closer and farther away.
5. Add roughness or tint only after the reflection geometry is correct.
6. Add the frame around the mirror without burying its reflective face inside the wall. Test the same walk again.
7. Measure playability with all mirrors visible and the normal HUD active.

Compare against an ordinary reflective sphere in the same room. The sphere may show a useful environmental capture while the mirror shows the character’s current movement. That comparison makes the difference between the two methods clear.

A mirror can also reveal the edge of a set that the main camera never sees. Build or conceal the reflected side deliberately. A missing ceiling in the reflection is a set-design issue, not necessarily a reflection failure.

> Use a stable fixed camera for the first reflection test. A moving group camera changes both the main view and the reflected composition, which makes simple errors harder to spot.

87 A restrained horror or noir room

A dark room still needs readable interaction. Darkness is useful when it directs attention, gives the player a reason to explore or supports a reveal. Uniform blackness merely hides the controls and scenery.

1. Begin with a working neutral room. Reduce ambient intensity gradually rather than turning every source off.
2. Use one clear key light. A cool directional source and a warmer local practical lamp can separate the room into readable areas.
3. Keep the character’s face or silhouette distinguishable against the background. Add a restrained fill if needed.
4. Use regular fog at distances suited to the room. Add Horror Fog in the screen stack only after the normal lighting and depth cues work.
5. Add mild vignette and lowered saturation. Check important props at the darkest part of the camera frame.
6. If the scene uses flashlight reveal, test it with the pointer in every corner and with the inventory open. Turn it off when leaving the scene.

Choose one strong clue treatment: a warm pool of light, a small emission accent or a restrained hover response. If every prop flashes, glows and outlines at once, the clue no longer stands out.

> Test on an ordinary display with ordinary brightness. A scene that is barely readable on your bright monitor can be effectively invisible on another player’s screen.

88 Study the Clockmaker camera examples

Create a separate game from the Clockmaker template to inspect its authored examples. Opening a new template copy avoids mixing experimental changes into a game you have already edited.

Example	What to inspect
Exterior Opening	The long blend from exterior_wide toward exterior_tower establishes the building.
intro_follow	A three-point path used while looking toward Ivo as he enters.
intro_elias / intro_ivo	Actor-related shots and softened transitions for reactions.
intro_pair	Group framing for the two actors.
intro_handshake	A composition that leaves room for both characters during the handshake.
wide	The return to the ordinary gameplay view.

Open the room’s Cameras list and select one shot at a time. Compare its Position & View with Advanced Camera. Then inspect the matching event on the cutscene’s camera track. The camera record explains the shot; the timeline explains when it is used.

Run the opening once without editing it. On a second pass, change one camera’s damping or blend duration. Compare the entire scene, not only that camera’s first frame. Timing changes can overlap speech or gestures that were designed for the original transition.

> Template updates do not overwrite an existing user-created project. If an older Clockmaker copy has different camera records, inspect a fresh template copy for these examples.

89 When the camera does something unexpected

Problem	Likely checks
The new camera never appears	Current room, camera ID, Enabled at Runtime, room Default Camera and whether an activation action actually runs.
Follow appears stationary	Advanced Camera enabled, correct active shot, available target, selected mode, and a manual move/orbit that may own the view.
Group crops a speaker	Target list, max distance/view height, padding, viewport shape and room bounds.
Path stays fixed	At least two valid XYZ points; then check the runtime log.
Path does not move on a timer	Path tracks the target’s nearest route position. Use fixed-shot blends or timed CameraMove for a scheduled move.
Camera will not reach the desired view	Room bounds and Respect Room Camera Bounds; locked 2D Stage restrictions; first-person/XR ownership.
Picture snaps after a conversation	Check Restore Camera, explicit final activation and inherited Dynamic Follow settings.
Blur has no effect	Actual renderer, active camera focus settings, transparent viewport limitations and focus distances.

Test with a fixed shot and a single target before restoring a complex setup. That gives you a clear baseline.

90 When the lighting looks wrong

Problem	Likely checks
Light has no visible effect	Starts On, intensity, range, position inside opaque geometry, spot rotation, lit material and competing ambient light.
Scene looks flat	Ambient/fill too strong, several equal key lights, high emission across ordinary surfaces or heavy grading.
Character seems to float	Model/floor position, direct shadows, detached-shadow bias and contact-patch setup.
Black speckles or stripes	Shadow bias/normal bias, model scale, thin overlapping geometry and depth precision.
Glow makes everything pale	Reduce broad bloom, intensity or low threshold. Brighten the intended emitter rather than the entire picture.
God rays missing	Forward+ for native volumetrics; suitable spot/directional type for visible shaft geometry; source direction and occlusion.
Fog barely changes the room	Depth Begin/End may be beyond the small room. Measure from the camera.
Outline makes faces unreadable	Reduce thickness; compare hover-only feedback and a simpler screen style.

Change one variable and compare the same view. If you change light intensity, ambient level, exposure and material color together, it becomes difficult to tell which change helped.

91 A practical performance pass

Use the slowest hardware you intend to support. A desktop with a dedicated GPU is not a reliable substitute for an integrated-graphics laptop or tablet. Compare the same room, camera and movement each time.

1. Begin with Balanced or Performance quality. Confirm the actual renderer.
2. Disable live mirrors briefly. If performance improves sharply, reduce their count or keep fewer visible together.
3. Turn off unnecessary Cast Shadows switches. Reduce directional shadow distance to the area actually visible.
4. Shorten overlapping local-light ranges. For RGB strings, increase Light Every N Bulbs or use 0 for decorative-only bulbs.
5. Disable whole-scene outline, heavy screen layers and large particle effects one at a time. Keep the effects that make a clear difference.
6. On Forward+ Enhanced, test SDFGI, SSR and SSAO separately. Reduce Enhanced Quality before abandoning a useful feature.
7. Test the exported build with normal input, dialogue and room changes. Check speed after several minutes, not only the first still frame.

A GPU-driver crash requires logs and hardware-specific investigation. This guide’s quality suggestions are ways to narrow the rendering workload; they are not a claim that one setting fixes every crash.

> Performance is part of the final look. A modest shadow that stays stable while walking is more useful than an elaborate effect that makes clicking uncomfortable.

92 Test before you distribute

1. Open the game fresh and confirm the intended starting player, room and default camera.
2. Visit every camera extreme: nearest and farthest walking points, left and right edges, high and low views if the game permits them.
3. Run each camera cutscene to completion, then run it again and Skip at different moments. Confirm input, HUD, camera and tracking are restored.
4. Switch the active character. Test player-alias follow and any explicit-character conversation targets.
5. Compare each supported renderer and quality profile. Check missing-effect fallback rather than assuming the same image everywhere.
6. Test mirrors with a moving character. Test outlines against transparent objects. Test glow with its switch on and off.
7. Build the game and test from the complete output folder on a clean launch. Keep the supporting configuration files with it.
8. Check every target resolution and the smallest supported screen. Verify that dialogue and GUI remain readable over the finished picture.

Keep a short list of deliberate platform differences. For example: the web build uses ordinary fog and decorative shafts, while the desktop Forward+ build uses volumetric scattering. A planned fallback is easier to maintain than an unexplained missing effect.

> Save the authored baseline before making export-specific experiments. Runtime actions and temporary test changes are not a substitute for the saved project’s intended defaults.

93 Advanced Camera field reference

Field	Default / units	Use
enabled	false	Enable optional behavior.
mode	fixed	fixed, follow, group, framed or path.
targetId	player	Primary target in the current room.
targetIds	Empty	Additional IDs as comma-separated text or a script array.
lookAtId	Empty	Optional separate aim target.
targetOffset	[0, 1.3, 0]	XYZ offset in scene units.
damping	0.2 seconds	0 is immediate; lower is faster.
blendDuration	1 second	Destination default for activation duration -1.
ease	sine	linear, sine, cubic or quint.
priority	0	Higher wins when SetCameraPriority evaluates shots.

These fields belong inside advancedCamera on a camera record. SetCameraBehavior accepts them as top-level action properties alongside cameraId. For an authored change, use the camera Inspector; for a session change, use the action.

> Enabled at Runtime is a camera-record setting. advancedCamera.enabled is the separate behavior setting. A disabled behavior does not mean that the ordinary camera record has been deleted.

94 Group, frame and path field reference

Field	Default / units	Use
autoFrame	true	Automatically fit the group.
minDistance	3 units	Minimum perspective distance or orthographic view height.
maxDistance	18 units	Maximum distance/view height; at least the minimum.
framingPadding	1.3	Group breathing room multiplier, minimum 1.
deadZoneWidth	0.4	Framed mode screen-width fraction, 0-1.
deadZoneHeight	0.3	Framed mode screen-height fraction, 0-1.
pathPoints	[]	Ordered array of XYZ points; at least two.
respectBounds	true	Allow room pan limits to correct the shot.

Action	Parameters
ActivateCamera	cameraId, duration (-1/default, 0/cut or seconds), wait.
SetCameraBehavior	cameraId plus the behavior fields you intend to change. Blank ID targets the active camera.
SetCameraTargets	cameraId, targetId, targetIds, lookAtId.
SetCameraPriority	cameraId and integer priority. Re-evaluates eligible advanced cameras.

Use explicit IDs in reusable event sequences. A blank active-camera ID is convenient when you deliberately mean “whatever shot is active,” but can change the wrong shot if an earlier action switched cameras.

95 A short rendering glossary

Term	Plain meaning
Albedo	A material’s base surface color or base-color image.
Emission	A surface that contributes its own apparent brightness.
Bloom / glow	A halo around bright parts of the picture.
Specular highlight	The bright reflection of a light on a surface.
Reflection probe	An environmental capture used by reflective materials.
Planar reflection	A reflected view for a flat surface such as a mirror.
SSR	A reflection technique using the current camera’s visible image information.
Ambient light	Broad illumination that helps surfaces outside direct light remain readable.
SDFGI	The Enhanced profile’s indirect-light system based on static geometry.
SSAO	Screen-space darkening around contact and crevice-like areas.
Shadow bias	An offset used to reduce false self-shadowing.
Tone mapping	Mapping the lighting range into a visible image.
Post process	An effect applied after the room has been drawn.
Renderer	The drawing backend; distinct from a quality policy or art style.

Sources and credits

The instructions and control names were checked against PCA 3D Adventure Studio 0.86.3.4, its bundled documentation and the native editor. The practical camera, lighting, mirror and outline screenshots were captured from the engine. Descriptions of backend-gated features follow the current PCA rendering configuration; the guide identifies them rather than presenting a fallback preview as a Forward+ demonstration.

Useful bundled reading: docs/advanced-cameras.md, docs/ENHANCED-RENDERING.md and docs/ANDROID-RENDERING.md. These files accompany the source package and explain the version-specific integration and platform notes.

Phantom Camera — project and license

Advanced camera integration: Phantom Camera v0.11.0.4 by Marcus Skov (Ramokz) and contributors, MIT license. PCA exposes its own supported camera workflow rather than every upstream feature.

Godot — environment and post-processing

Background reading on environment lighting, tone mapping, glow and camera attributes. Godot’s own UI and exposed controls differ from PCA’s editor.

Godot — 3D lights and shadows

Background reading on direct light types, shadow mapping and renderer light limits.

Godot — overview of renderers

Background reading on Compatibility, Mobile and Forward+ and their hardware requirements.

PCA’s bundled retro-rendering notes credit the PSX/N64 material and Horror Fog work by Francisco Cidade under CC0. Bundled demo assets retain the credits and licenses shipped with the engine. Screenshots here show those assets inside PCA 3D; no game practice files are included.