



USER MANUAL

ENGLISH

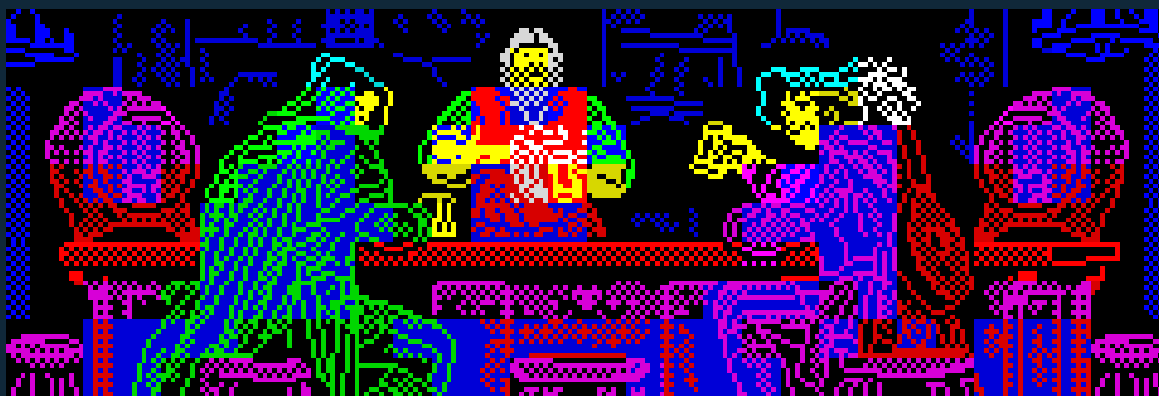
ZX SPECTRUM / GRAPHIC + VECTOR

Create Reusable Graphics for ZX-Spectrum Projects

The complete guide to SEIE

Draw, organise and reuse. From the first attribute cell to shared scenes and external game resources.

TABERN / MEDIEVAL / CREATED IN SEIE



132 pages / Editable projects / Real captures

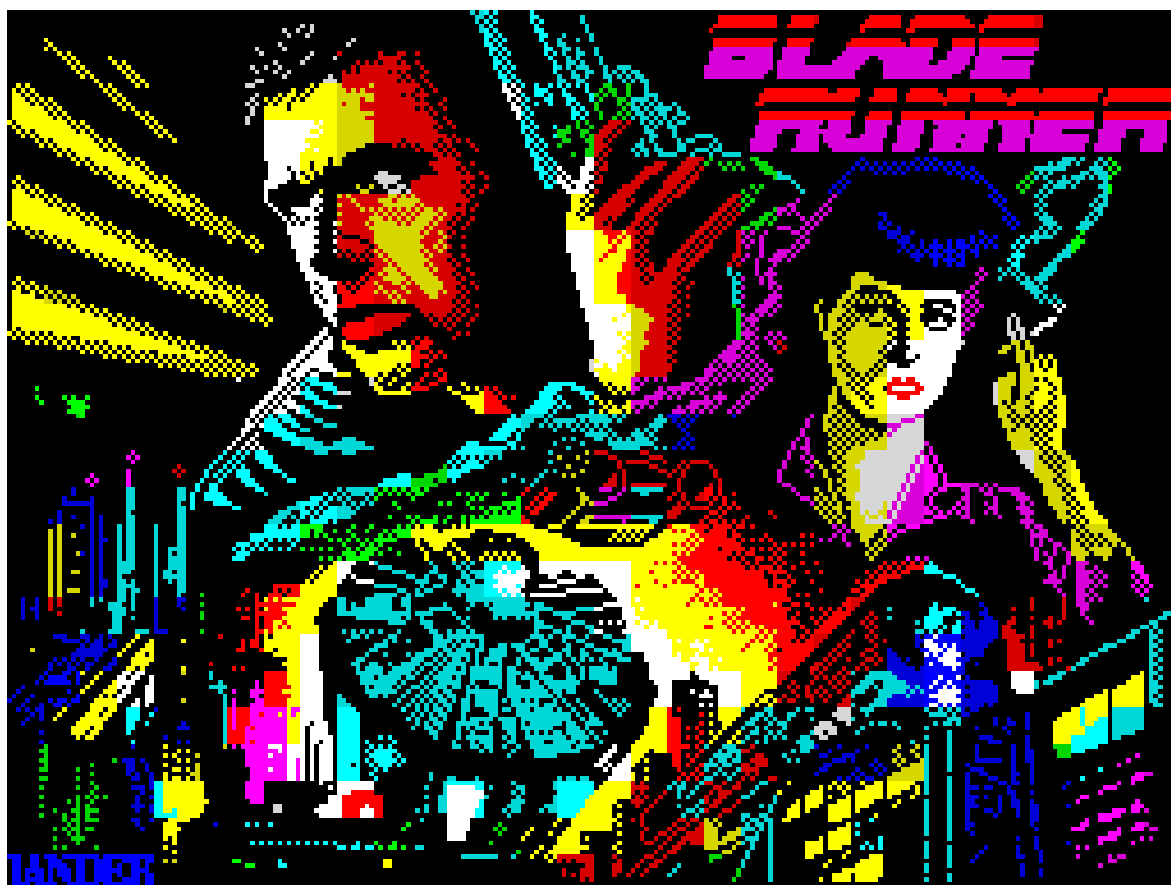
Prefabs | Groups | Animation authoring | References

English edition / SEIE 0.1.1747



Create the first impression

Create presentation, loading and title screens for ZX Spectrum games - then build the artwork that follows.



Blade Runner: a full-screen illustration created in SEIE. Direct render of the artist's original .scr file, with its pixels and colour attributes intact.

What is SEIE?

SEIE is a ZX Spectrum image editor. Start with a single screen: compose an illustration, refine its pixels, work with colour attributes and add lettering. A title screen, loading picture or presentation image is a complete and worthwhile project in its own right.

Make the screen yours

Combine Graphic pixel editing with Vector construction when useful. Draw, revise, inspect and export the result. You do not need to build an asset library, a game world or an animation to get value from the editor.

One screen. Many visual styles.

Use illustration, lettering, portraits, action and patterned shading to establish the mood before the game begins.



Big Trouble: a cinematic composition.



Ghost And Goblins: bold action and colour.



Magic VS Bird: figures, movement and contrast.



Sir Fred Tribute: character and atmosphere.

Built for the Spectrum screen

A full screen is 256 x 192 pixels. Each 8 x 8 attribute cell has INK and PAPER colours, plus BRIGHT and FLASH. SEIE puts these constraints inside the drawing workflow instead of leaving them as a conversion problem at the end.

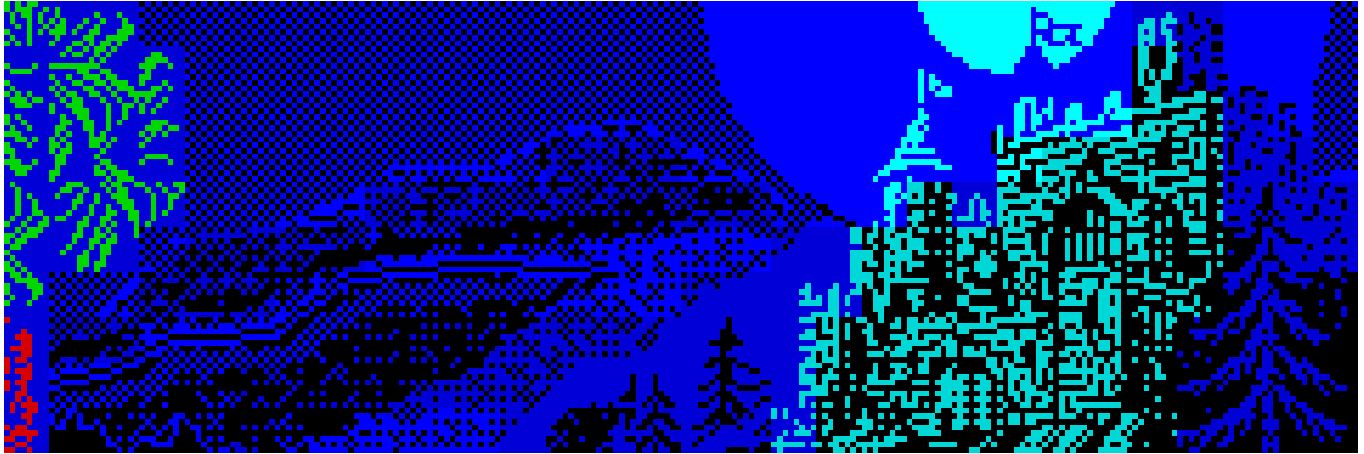
A picture first, a project when you need it

Use direct pixel tools for detail or retained vector commands for editable construction. Unlike a general image workflow, SEIE also keeps Spectrum attributes, fonts, reusable assets and scene relationships together.

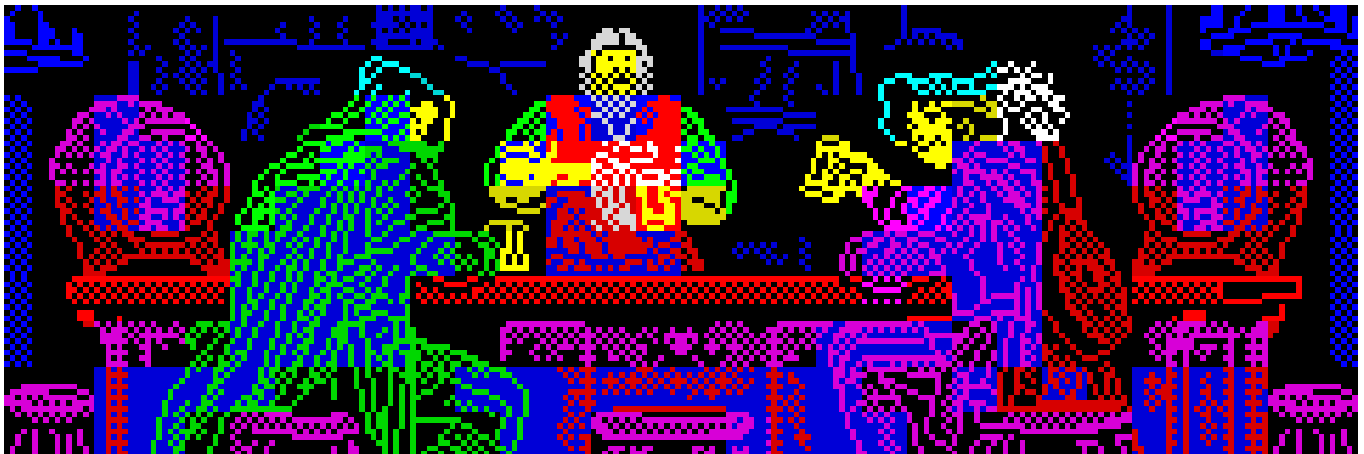
All five loading/title screen examples were created in SEIE and supplied by their artist. They are faithful renders of complete .scr screens, not mockups or cropped approximations.

Beyond the title screen

When the first image is ready, the same editor can help you build the places, characters and visual language of the game itself.



Castle_Far: distance, light and atmosphere in Medieval's 240 x 80-pixel Work Area. Actual editor render.



Tabern: expressive characters and patterned shading. Both scenes come from the artist's Medieval project in SEIE.

From one picture to connected places

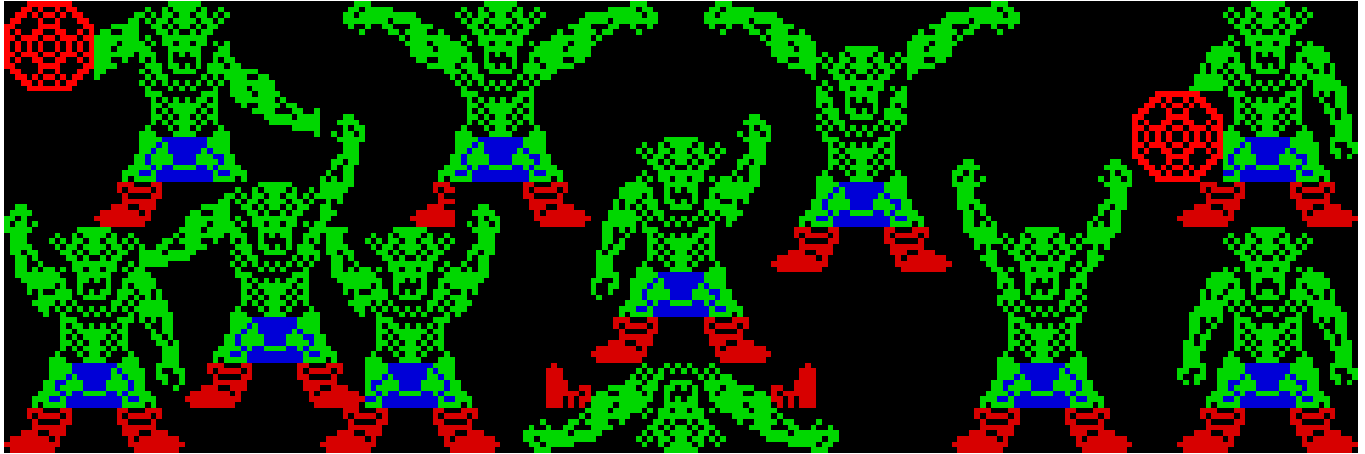
Documents let you organise separate scenes in one project. Graphic and Vector authoring are different ways to construct their local artwork; neither prevents you from using reusable assets. References can carry a shared background into another scene.

Choose the right output

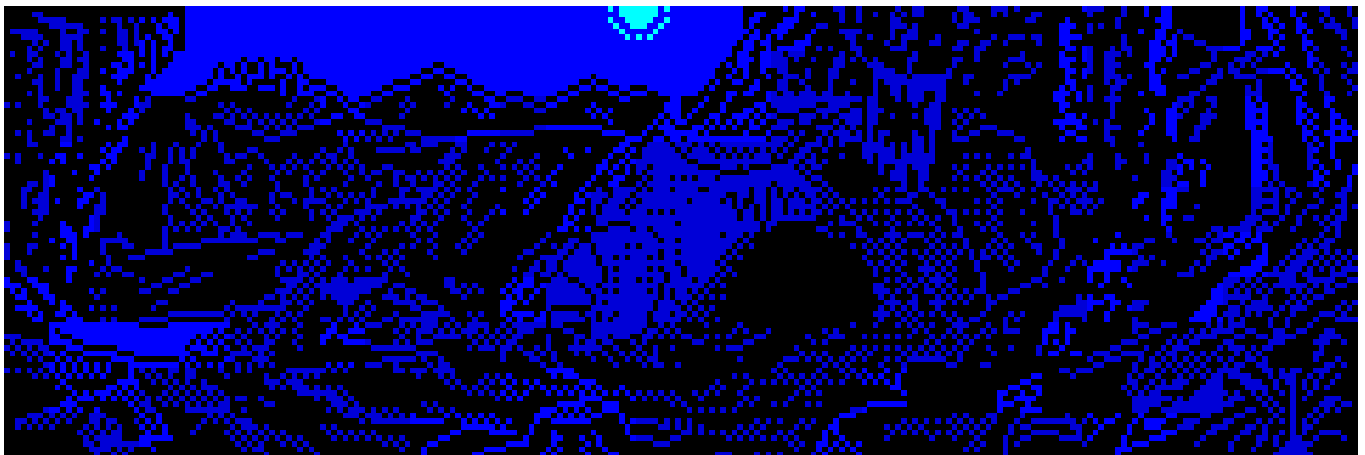
Image exports provide pictures; SNA output runs a generated presentation. Resource Packs supply graphics for an external game, whose code still owns rendering, timing and memory management. Choose the route that matches your project rather than assuming one export does everything.

Build once. Use many times.

Reuse is the next step, not a prerequisite: turn useful artwork into pieces you can place, arrange and update across scenes.



One base Orc image, shared arm/leg pieces and many poses: mirrored, cropped and repositioned prefab instances stored as groups. Actual Medieval editor render.



Orcs_Cave_Gate: 100% prefab reuse. All 14 placements use assets already used in other Medieval scenes; no new prefab source is needed for this composition.

Prefabs, groups and animation clips

A prefab stores one source image; an instance places, crops or mirrors it. The Orc poses above are not independently drawn whole-character images: shared body and limb placements form the poses, and groups remember their arrangements. Animation clips can then refer to those groups. Current static/SNA output represents the initial frame, not native animation playback.

Change the right owner

Edit a master for a shared change or an instance for a local one. Mirroring and sharing sources can save artwork memory when the output preserves shared assets; exact RAM cost depends on the export strategy. The following chapters teach prefab creation and pose grouping with the real Orc, then show how reuse scales to Orcs_Cave_Gate.

Contents and learning routes

Read straight through for a complete introduction, or use the PDF bookmarks and online contents to jump to a task.

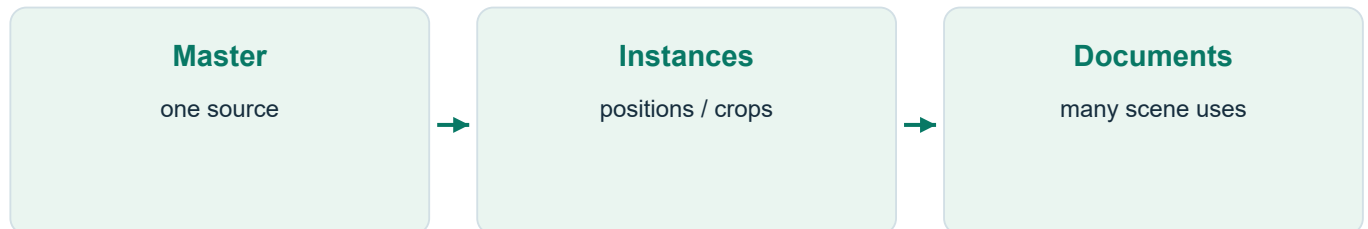
What is SEIE?	001-005
Start here	006-012
Graphic tools	013-020
Vector tools	021-028
Selection and colour	029-036
Prefabs	037-052
Groups	053-064
Animations	065-076
References	077-088
Orc reuse workshop	089-100
Text and fonts	101-108
Output and emulation	109-120
External games	121-126
Reference desk	127-132

THREE USEFUL ROUTES

New artist: start with pages 7-20. Reusable scene builder: work through pages 37-100. Game programmer: read pages 109-126, then follow the separate Prefab to Spectrum Game tutorial.

What you are editing

A scene, a reusable master and an instance are different things. Knowing the owner of a change prevents surprises later.



Teaching diagram - not an application screenshot.

Five building blocks

A document is a scene. A prefab is a reusable piece of artwork. A group is a reusable arrangement of components. An animation is a timed sequence of frame compositions. A linked document reference brings another scene into the current scene.

Local or shared?

Moving an instance changes its placement. Editing its master changes shared artwork. A local colour override belongs to a placement until you explicitly publish it to the master. A variant maintains a relationship to an original; an independent copy does not.

Before clicking Save

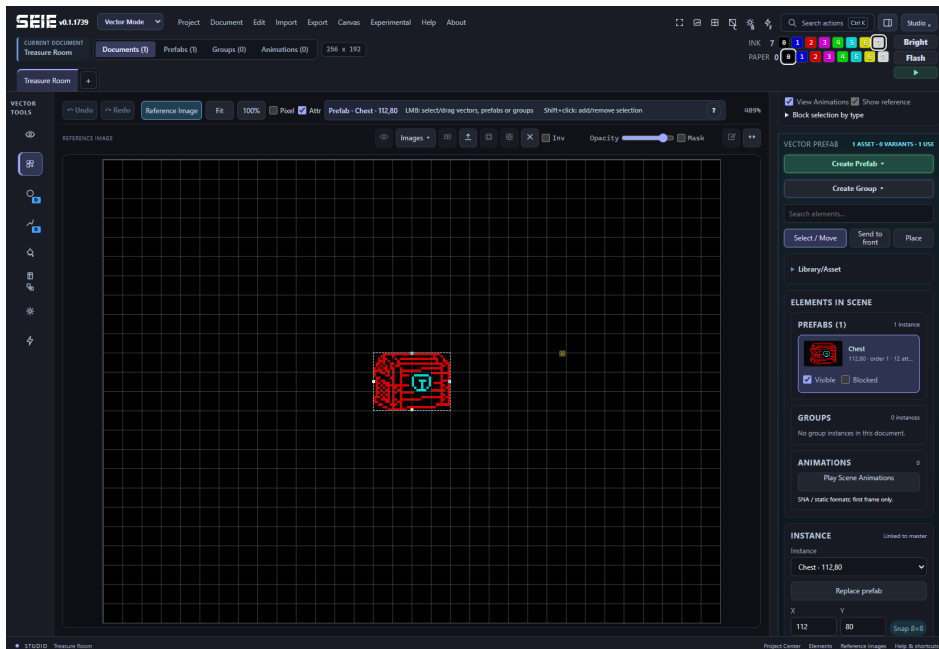
Read the active workspace tab, the Current Document/Prefab label and the editor's scope. A master editor may look like a normal canvas, but it is not the room that contains the asset.

TRY IT

Open 01-First-Adventure.siep.json. Find Treasure Room under Documents and Chest under Prefabs. They are two views of different owners, not duplicate rooms.

Read the workspace

The Studio interface puts tools on the left, artwork in the middle and task-specific controls in the right inspector.



The real workspace: menus and contexts above, tools left, canvas centre, inspector right.

Find the current context

Documents, Prefabs, Groups and Animations switch between kinds of editable content. The tabs below list only the selected kind. Counts refer to those collections, not to the number of visible objects on the canvas.

Navigate without changing artwork

Use Fit to frame the Work Area when enabled, otherwise the canvas. The mouse wheel zooms and middle-button drag pans. Pixel and Attr are inspection grids. The contextual help strip explains the active tool's controls.

Reveal the right controls

Library/Asset contains reusable definitions; Elements in Scene contains placements. In Graphic, Canvas/Elements and Select/Invert form the top section. Pixel/Rubber, Circle/Line and Square/Fill follow. Vector keeps equivalent tools in the same relative order. Studio/Classic changes layout, not the project format.

CHECK

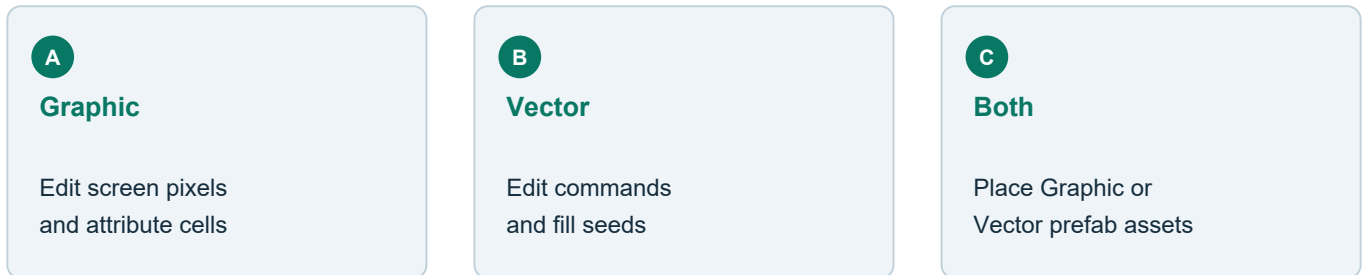
If an expected control is missing, first check mode, active tool and editing context. Do not convert a document simply to reveal an asset's controls.

Choose Graphic or Vector

The document's authoring mode describes its local artwork. It does not dictate the type of every prefab placed in it.

TEACHING DIAGRAM

Choose the local authoring model



Graphic

Graphic editing changes screen pixels and colour attributes. It is suited to direct pixel work, imported image cleanup, lettering and exact raster artwork. The new-project dialog may call this Bitmap Mode; the workspace calls it Graphic Mode.

Vector

Vector editing retains commands such as lines, circles, boundaries and seeded fills. This allows geometry to be moved and fills to be recomputed. The display is still a Spectrum-sized raster result, not a resolution-independent final image.

Mix assets deliberately

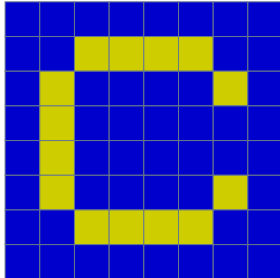
A Graphic prefab can be placed in a Vector scene, and a Vector prefab in a Graphic scene. Placement does not switch the scene's mode. Converting a scene or its masters is a separate, explicit operation with wider consequences.

CHOOSE

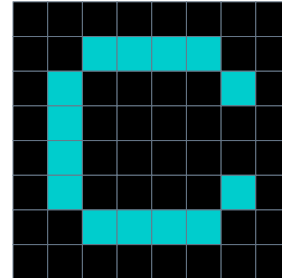
Use Graphic for direct control of pixels. Use Vector when editable construction and semantic drawing are valuable. Save a project copy before conversion; rasterizing does not preserve all vector editability.

Pixels draw. Attributes colour.

A Spectrum screen has 256 x 192 pixels and 32 x 24 colour cells. Each attribute cell covers 8 x 8 pixels.



Same bits: yellow / blue



Same bits: cyan / black

Two colours per cell

A set pixel uses INK; a cleared pixel uses PAPER. Changing a cell's INK/PAPER pair recolours all its pixels. BRIGHT affects the brightness of both colours. FLASH alternates their display; it is not a second independent colour layer.

Transparency is ownership

Black PAPER is still artwork. In a prefab, an excluded attribute region is a hole that does not belong to the asset. This is different from drawing an owned cell black, or hiding the entire instance.

Design with the grid

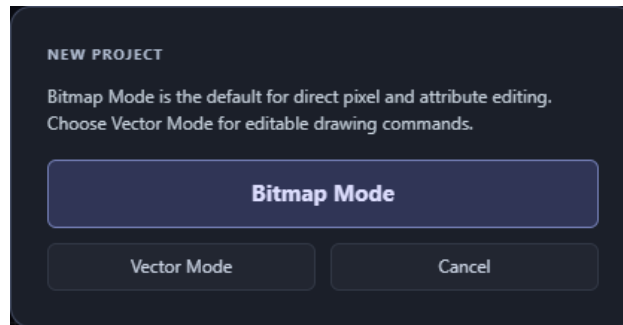
Put critical colour boundaries on cell edges. A character and a wall sharing a cell cannot independently choose arbitrary Spectrum colours. Strong silhouettes and considered dithering help, but do not remove the hardware constraint.

TRY IT

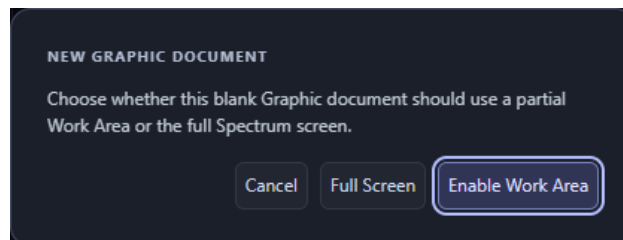
In Graphic Practice, change one cell's INK while leaving the pixels alone. The pattern stays the same; its colour changes across the complete 8 x 8 cell.

Start, frame and save a project

Save editable work before exporting pictures or opening a different project. An image export cannot restore your asset library.



Bitmap Mode is the primary default. Vector Mode remains available as a secondary alternative.



The new-document choice: a Work Area or the full screen.

Create

Choose Project > New Project. Bitmap Mode is the primary, focused default; Vector Mode is the secondary alternative. Choose Full Screen for 256 x 192 pixels, or Enable Work Area for a smaller region. Finish Work Area editing before returning to normal tools.

Save

Project > Save Project downloads the multi-document workspace with its libraries and links. Keep the downloaded file in a known folder. Use Open Project to return to it. Document saves have a narrower scope; use project saves for the exercises in this guide.

Keep recoverable checkpoints

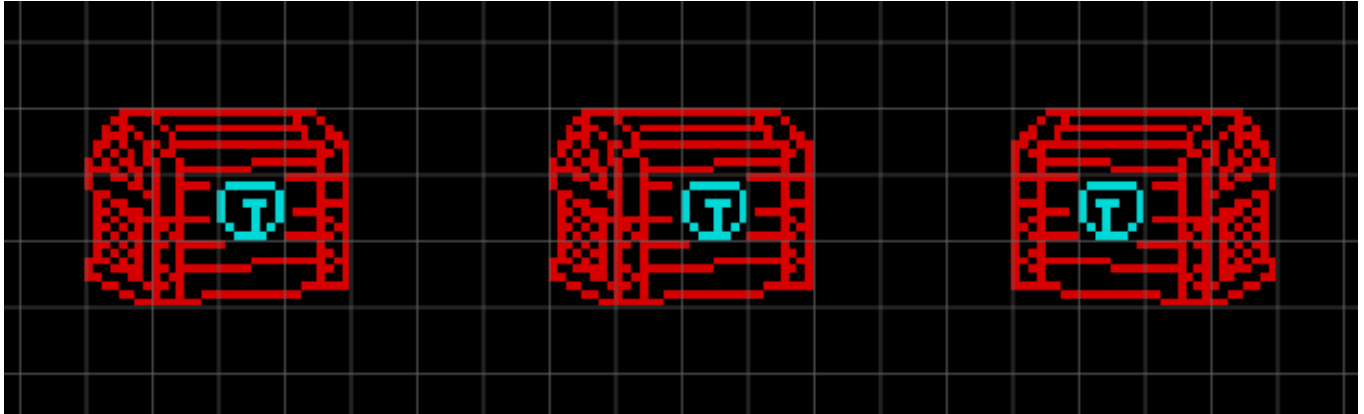
Use names such as Adventure_01 and Adventure_02 before shared edits, repair or conversion. A browser refresh is not a substitute for saving. Confirmation warnings help prevent accidental loss but are not a backup system.

CHECK

Reopen one saved file now. Confirm the document names, asset counts and a visible instance. This inexpensive check catches choosing the wrong file or export format early.

Your first reusable arrangement

The first three companion files form a complete beginner exercise. You can start at any checkpoint without rebuilding the earlier steps.



One Chest master, three placements; the right placement is mirrored horizontally.

Place

Open 01-First-Adventure.siep.json. Select Elements. Expand Library/Asset > Prefabs, choose Chest and click the canvas. Add placements at X 56, Y 80 and X 168, Y 80; the starter is at X 112, Y 80.

Transform

Select the right-hand chest and enable Mirror H in its Instance controls. Check that only this placement changes. File 02-Three-Instances.siep.json contains this result.

Group

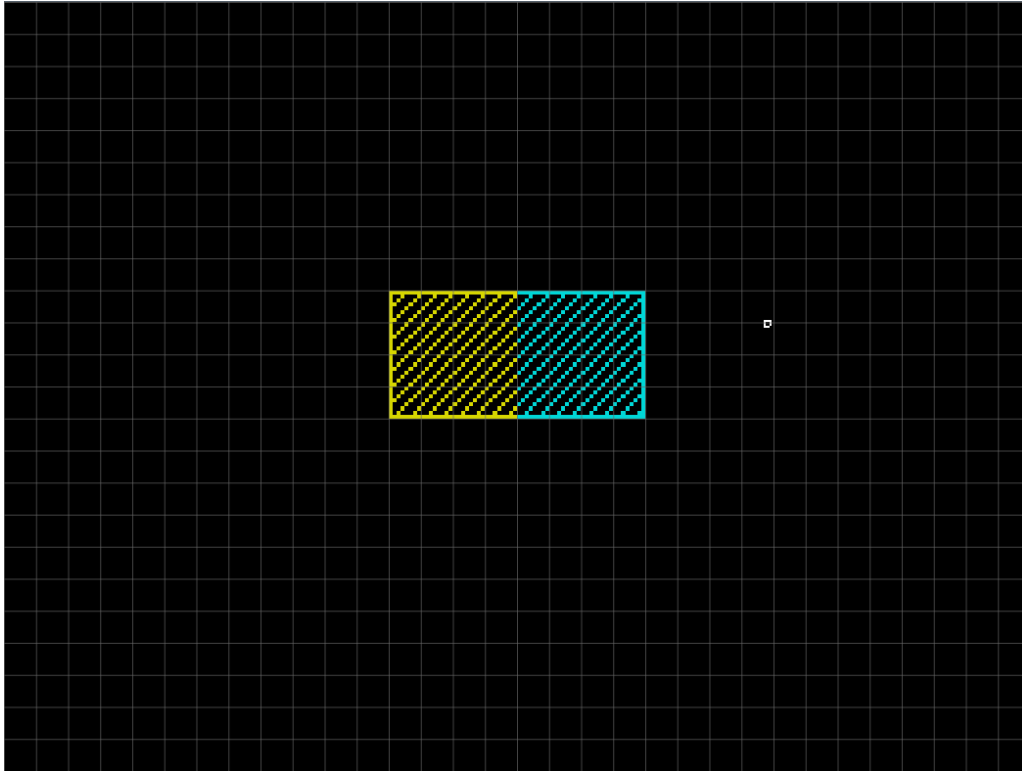
Use Edit > Select All in this small scene, then Create Group > From current selection. Name it Treasure Set and save the project. File 03-Treasure-Set.siep.json is the completed group.

BUILD NOTE

In the audited build the group-name field says Prefab name, and the Groups count may remain stale immediately after creation. Saving and reopening refreshes the count; the saved group is present. These are interface inconsistencies, not extra steps needed to define a group.

A small pixel workshop

Open 09-Graphic-Practice.siep.json. Work on the supplied pattern before trying the tools on an important scene.



A deliberately small Graphic example with two attribute colours and a repeated pixel pattern.

Prepare the view

Use Fit, then zoom in until individual pixels are comfortable to target. Enable Pixel and Attr if useful. Select Pixel and confirm the pattern density before drawing. For ordinary opaque marks, use 100%.

Make and undo a mark

Draw a short stroke in an empty cell. Undo should remove that edit; Redo should restore it. Try the same with a different brush size. Return to a small brush for detail work.

Separate two kinds of change

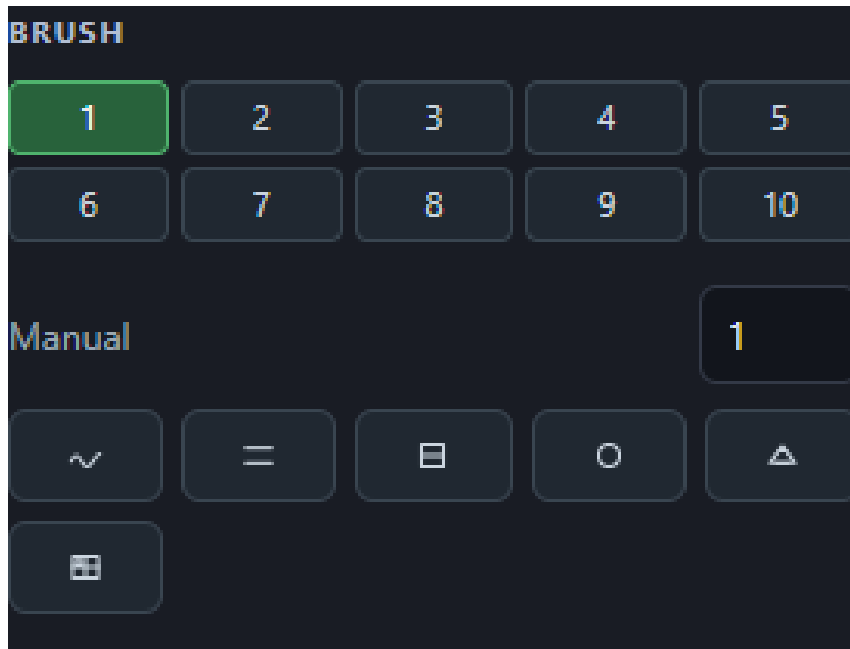
Pixel drawing changes the bit pattern. Color changes the attributes. If a new stroke seems invisible, check whether INK and PAPER are identical before drawing over it repeatedly.

CHECKPOINT

Save a new project copy rather than overwriting the supplied exercise. Keep the original practice file as a known starting point for later selections.

Pixel, strokes and brushes

Start with a predictable one-pixel brush. Increase complexity only when you can explain what the current brush will do.



Actual Pixel brush controls: start with one pixel before increasing thickness.

Pixel

A short left click paints; right click erases. Dragging draws a stroke. Holding the left button on the same pixel for more than one second starts a temporary pixel selection. Moving to another pixel before that becomes an ordinary stroke.

Alternative marks

Smooth Stroke, Art Brush and Spray produce different distributions of pixels. Inspect thickness, pattern and direction constraints in the right panel. Their percentages are not alpha transparency in the exported Spectrum screen: the result remains set or cleared pixels.

Tool memory

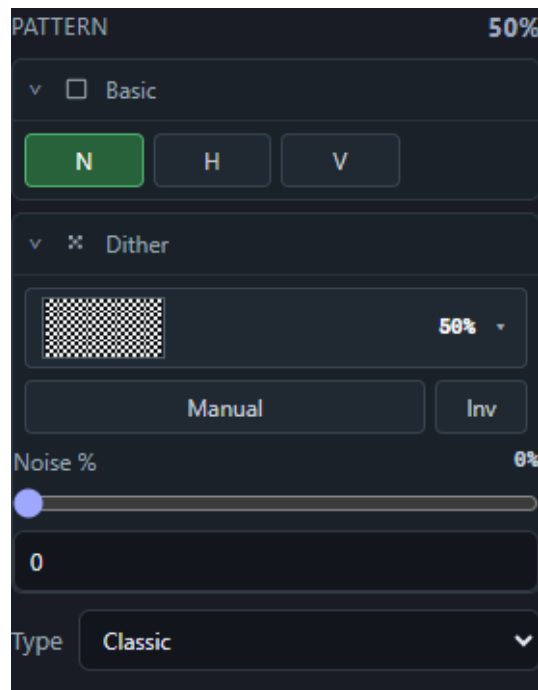
Tool activation starts ordinary painting tools at 100% and Dithering at 50%. Other tool-specific options remain independent. Always check the displayed pattern and a short test stroke when returning to an unfamiliar setup.

EXERCISE

Draw the same short diagonal with Pixel, Smooth Stroke and Spray, undoing between attempts. Compare the silhouette at 100% zoom, not only in the enlarged editor.

Dither without losing the silhouette

Dithering suggests intermediate tones by alternating INK and PAPER pixels. It does not create additional hardware colours.



Actual Dither controls with a 50% pattern and independent direction constraints.

Set the density

Select Dithering and inspect the density: it starts at 50% on activation. Other painting tools start at 100%. A 100% pattern is solid. The phase determines which positions are set in a repeated pattern.

Build an intentional texture

First establish a readable outline. Add dither inside it. Avoid changing the phase unintentionally between adjacent patches, unless the seam is part of the design. View the result at native scale to judge the apparent tone.

Direction and maps

Horizontal/vertical constraints limit the gesture, while maps and pattern options affect distribution. They are different settings. If a stroke looks unexpectedly solid, test on an empty cell and inspect the active tool's density before continuing.

TRY IT

On Graphic Practice, compare a solid area and a 50% area using the same INK/PAPER. Then change only INK. The texture is geometry; the colour belongs to the attribute cells.

Erase pixels or clean a whole cell

Removing pixels, clearing attributes and deleting an instance are separate operations. Choose the one that matches the unwanted content.



Rubber exposes its own brush and erasing controls. Clean Attribute Region is its alternate tool.

Rubber

Use Rubber to clear ordinary canvas pixels. Right click restores pixels according to the tool's behaviour. Settings such as intensity and pattern affect the stroke; use a simple solid brush for an exact cleanup.

Clean Attribute Region

Click the active Rubber tool again to switch to Clean Attribute Region. Left click clears the pixels of an 8 x 8 cell; right click fills that cell with pixels. This is useful for modular, grid-aligned corrections.

Do not confuse ownership

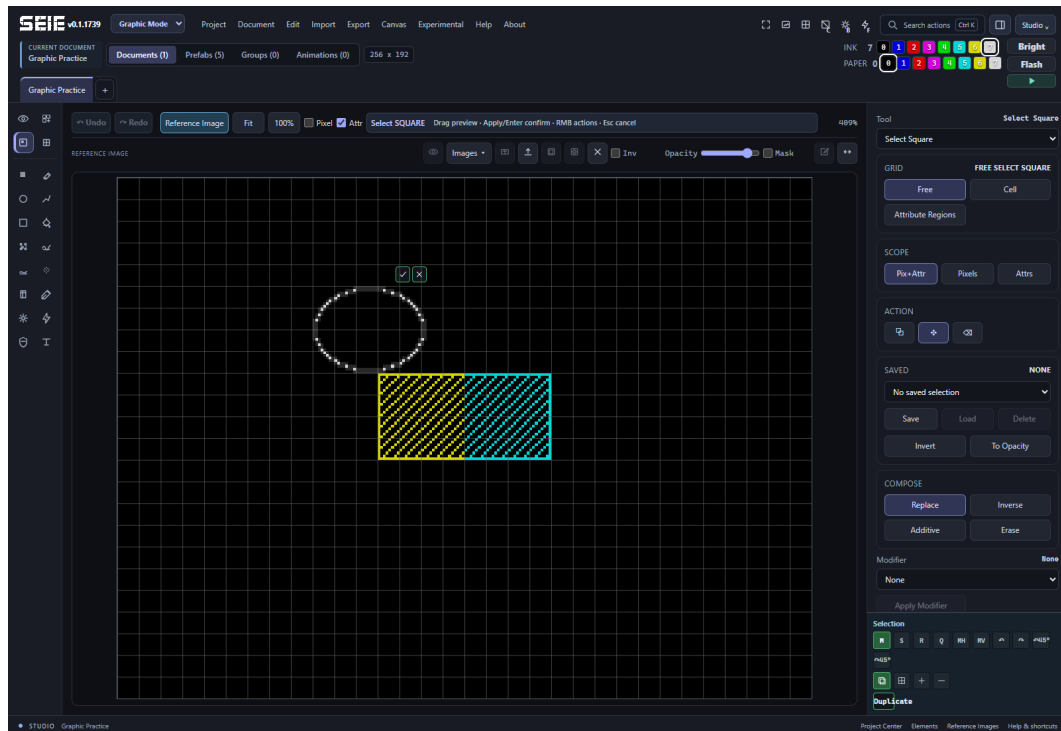
Cleaning pixels beneath a prefab does not delete the prefab instance. If the visible artwork belongs to an element, use Elements or edit the master. To create a transparent hole in a prefab, use Attribute Regions in its editor rather than merely erasing its bits.

CHECK

Hide the relevant element type temporarily if you need to inspect underlying canvas artwork. Restore visibility from the scene list; do not delete a shared master to reach the background.

Lines, squares and circles

Shape tools are fast ways to lay out precise pixel structures. Their results become bitmap artwork after confirmation.



Actual Circle preview: the tool icon identifies outline mode; Apply and Cancel sit beside the pending shape.

Guided strokes

Line Shape lets you place points for a stroke. Use its contextual help for completing or cancelling the current operation. Shift constrains the shape where supported. Clicking the active tool can cycle its related mode, such as linear/circular line.

Outline or filled

Square and Circle provide outline/fill choices. Check thickness and density before starting. A filled circle made with a dither pattern is a raster pattern, not a continuously editable Vector fill seed.

Drawing beyond the visible area

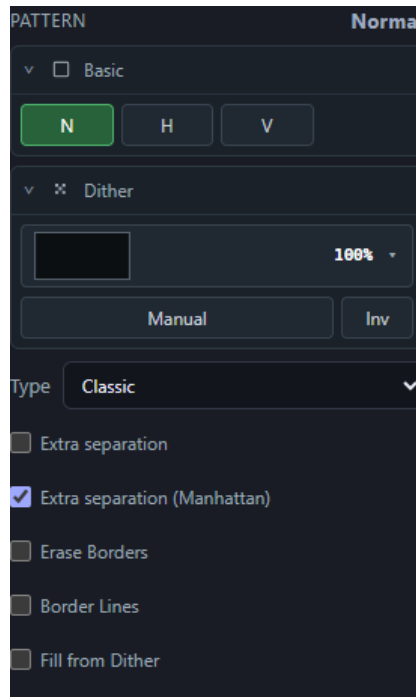
Line endpoints may be positioned in the precanvas. The actual drawing is clipped to the active Work Area or screen. Zoom and pan are navigation; they do not change those logical drawing limits.

EXERCISE

Make a frame with an outlined square, add a circle inside, then Undo both. Repeat with a different density and compare at native scale. Save only the version whose small-scale silhouette reads clearly.

Bucket fills and their limits

A Graphic Bucket fill changes a connected pixel region. It is not the same object as a Vector seeded fill command.



The Bucket pattern controls determine the pixel distribution of the fill.

Close the region first

Inspect the outline at pixel scale. A small opening can connect the intended region to a much larger area. The Work Area contains drawing when active, but it does not replace missing internal walls in your design.

Apply and inspect

Choose Bucket, set the pattern, click inside the intended area and inspect the whole result. If it floods too far, Undo, repair the outline and fill again. Repeated clicking on a broken enclosure does not repair its topology.

Separation options

Extra separation changes how a fill sits near its barriers. Use it deliberately rather than as a general cure for leaks. Inspect small corners and narrow corridors; a one-pixel difference matters at Spectrum resolution.

CHECK

After a fill, zoom out to see the entire scene. A leaked region can be much larger than the current close-up. Save the checkpoint only after checking the outline and colours together.

Paint attributes, not pixels

Color, Bright and Flash operate on Spectrum attributes. All pixels in a cell share the resulting colour settings.



The real Spectrum INK/PAPER palette, with separate Bright and Flash flags.

Color

Choose INK and PAPER in the palette, then paint the target cells with Color. Right click swaps the current INK/PAPER palette pair. To sample the displayed attributes, use Pick Color or the middle-button plus right-button shortcut.

Bright and Flash

Bright changes the brightness flag; Flash changes the flashing flag. These are properties of the complete cell, not of each visible dot. A scene with flashing cells can make a static screenshot misleading, so inspect a non-flashing reference when comparing colours.

Elements on the canvas

Painting over a visible prefab applies a local instance colour change where it owns the target cell. It should not silently recolour the master. A hole excluded from the prefab's owned regions lets you work on the content underneath.

SAFETY RULE

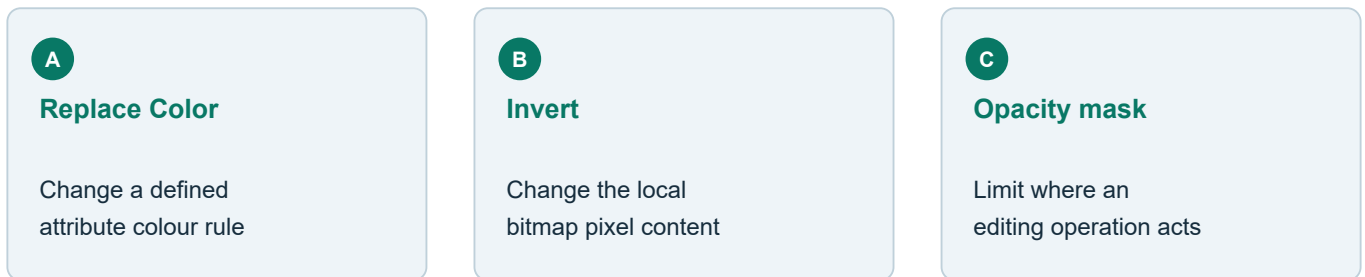
Apply Color to Prefab Master is a separate shared operation. Use it only when every linked use should inherit the change; ordinary canvas colour painting is local.

Replace colour, invert and opacity masks

These tools change different parts of the image. Test a small area before using them on an entire scene.

TEACHING DIAGRAM

Three operations, three different targets



Replace Color

Use the tool's replacement controls to define the conversion. The ordinary palette is blocked in this mode to avoid confusing a replacement rule with normal painting. Return to Color when you want to paint individual cells.

Invert

Graphic Invert works on pixel content. Do not treat it as an instance-level prefab recolouring command. For an actual prefab artwork inversion, work in its master and inspect the result before saving shared changes.

Opacity layers and pattern maps

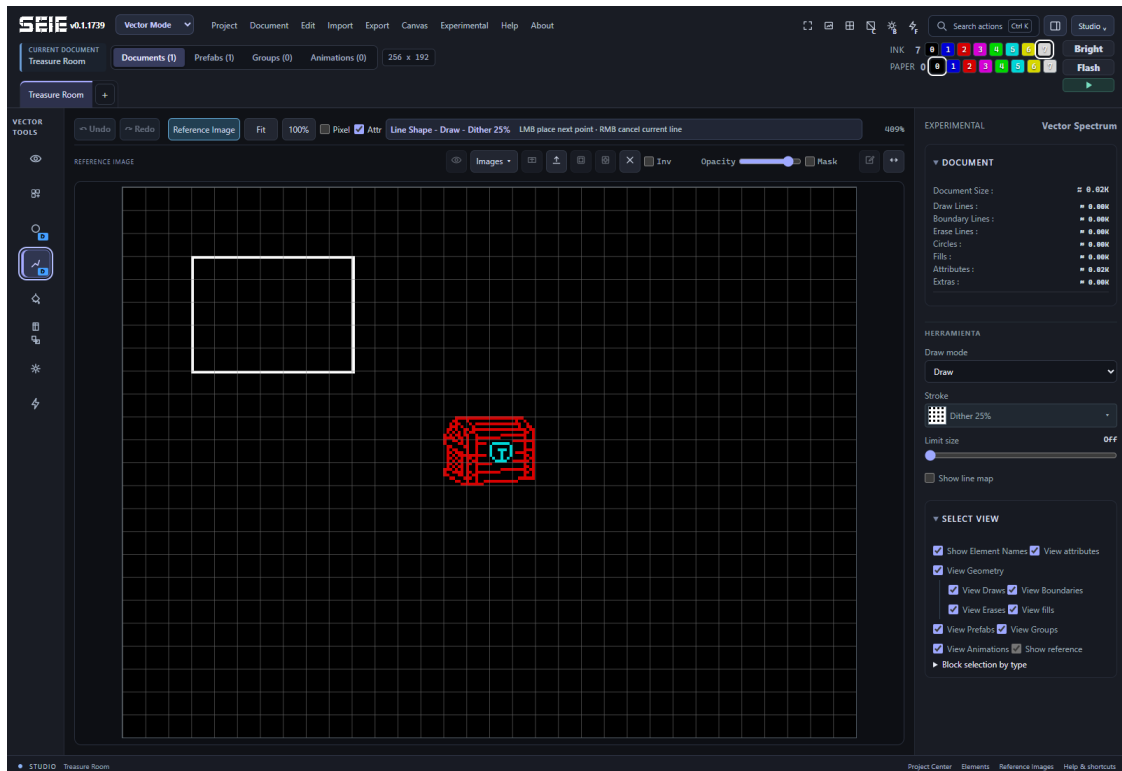
These are editor masks and distribution controls, not free-form alpha channels on a Spectrum display. Their purpose is to shape an operation. The exported screen still contains one bit per pixel and one attribute per cell.

EXERCISE

Use a saved copy, apply one operation and compare before/after at both enlarged and native scale. If a change appears invisible, inspect both pixel bits and INK/PAPER; changing both can preserve the apparent colours.

Think in commands

Vector mode keeps the construction of a scene editable. Its rendered picture is determined by geometry, boundaries, fill seeds, attributes and scene composition.



A closed enclosure drawn with the real Vector Line tool; the Chest remains a separate foreground asset.

Start with a simple enclosure

Create a Vector document, choose Line and use solid Draw lines to make a closed shape. Add a Fill seed inside. Move an edge with Elements and finish the move before evaluating the updated fill.

Inspect the right representation

Canvas shows the finished appearance. Show Line Map explains geometry types. Neither the coloured diagnostic overlay nor the seed markers are intended export artwork.

Keep ownership visible

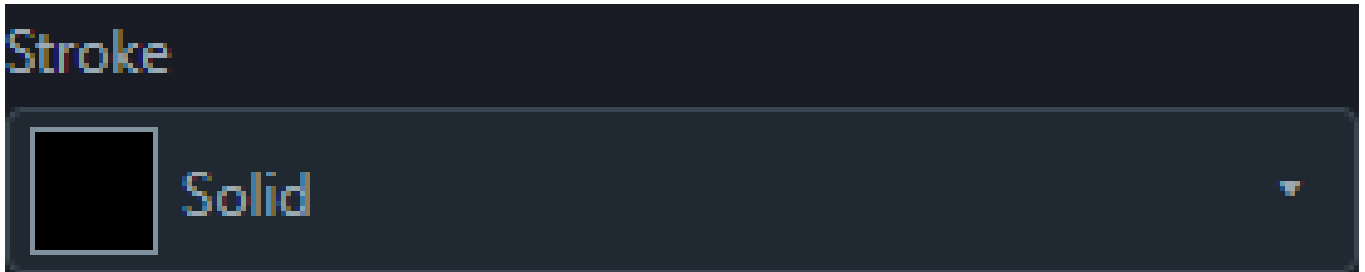
A selected command may be local, inherited through a reference, or part of a prefab. Prefab internals are edited through master/context editing, not by accidentally grabbing their individual vectors in a room.

CHECK

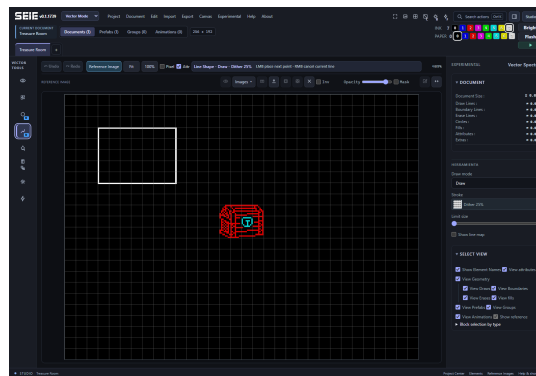
If a line is visible but cannot be selected, inspect the active context, reference protection, visibility and Block Selection settings before assuming the command is damaged.

Draw lines and stroke patterns

A line command has endpoints, a drawing mode and, when relevant, a stroke pattern. Changing one does not necessarily change the others.



The Vector stroke-pattern control selects how Draw lines mark their pixels.



Canvas example: solid Draw lines form a closed enclosure. The Stroke control above sets the line pattern.

Solid Draw

Use solid Draw for a visible contour that also blocks fills. Finish the line, then place a fill inside the enclosure. Check the corners carefully: visual closeness is not always a closed pixel boundary.

Patterned Draw

Patterned Draw creates discontinuous visible marks. In Show Line Map these are green when they lack the solid blocking behaviour. Use a separate Boundary when you need a continuous invisible fill barrier beneath a decorative pattern.

Extend across the precanvas

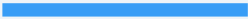
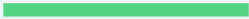
You can start or finish a line outside the active picture. Its visible result is clipped by the Work Area or screen. This helps build long diagonals without forcing both endpoints inside the final frame.

DESIGN EXAMPLE

A dashed road marking should be a patterned Draw. The edge containing a grass fill should be a solid Draw or an explicit Boundary. Do not expect decorative dashes alone to contain the fill.

Boundary is not Erase

An invisible fill barrier and a visible erasure solve different problems. This distinction is essential when constructing prefabs and mirrored scenes.

			
Solid Draw	Boundary	Patterned Draw	Erase
visible + blocks	invisible barrier	no solid boundary	removes pixels

Boundary

A Boundary blocks the spread of a fill without becoming an ordinary visible line. It is useful for separating regions where the final artwork should not show a drawn seam. Its diagnostic colour is orange.

Erase

Erase removes visible pixels in the effective drawing composition. Its diagnostic colour is purple. Replacing a Boundary with Erase can create a visible gap; replacing it with Draw can create an unwanted visible contour.

Diagnose before adding more barriers

Use Show Line Map to identify the real command type. If a fill leaks, inspect enclosure and seed context. Do not stack arbitrary duplicate boundaries without understanding the topology; it makes future edits harder to reason about.

CHECK

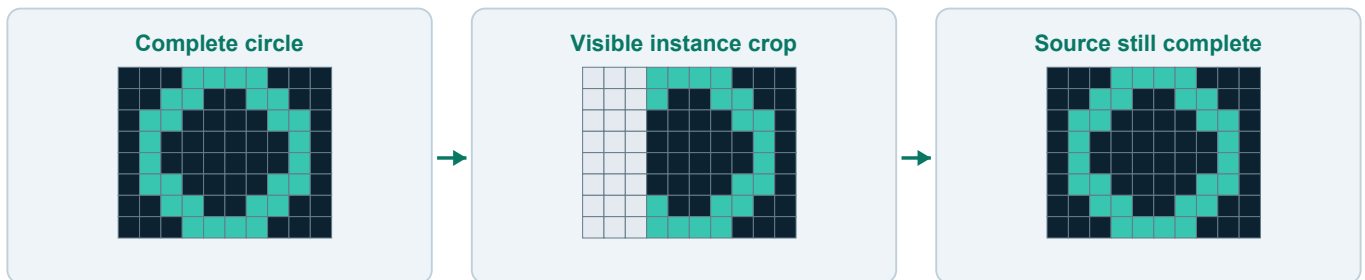
Compare Canvas and Line Map. The orange diagnostic line can be visible in the editor while the corresponding finished picture contains no visible boundary stroke.

Circles and clipped geometry

Circles remain semantic objects in Vector mode. Their behaviour differs from a collection of manually drawn pixels.

TEACHING DIAGRAM

A crop window does not rewrite the source circle



Create and adjust

Select Circle, establish its centre and radius, and check its mode. A drawn circle, a boundary circle and an erasing circle do not have interchangeable roles. Use the tool's contextual controls to finish or cancel construction.

Avoid ambiguous edges

At low resolution, a small radius changes quickly with a one-pixel adjustment. Inspect the connection between a circle and adjacent lines before using them as one fill enclosure.

Cropping and conversion

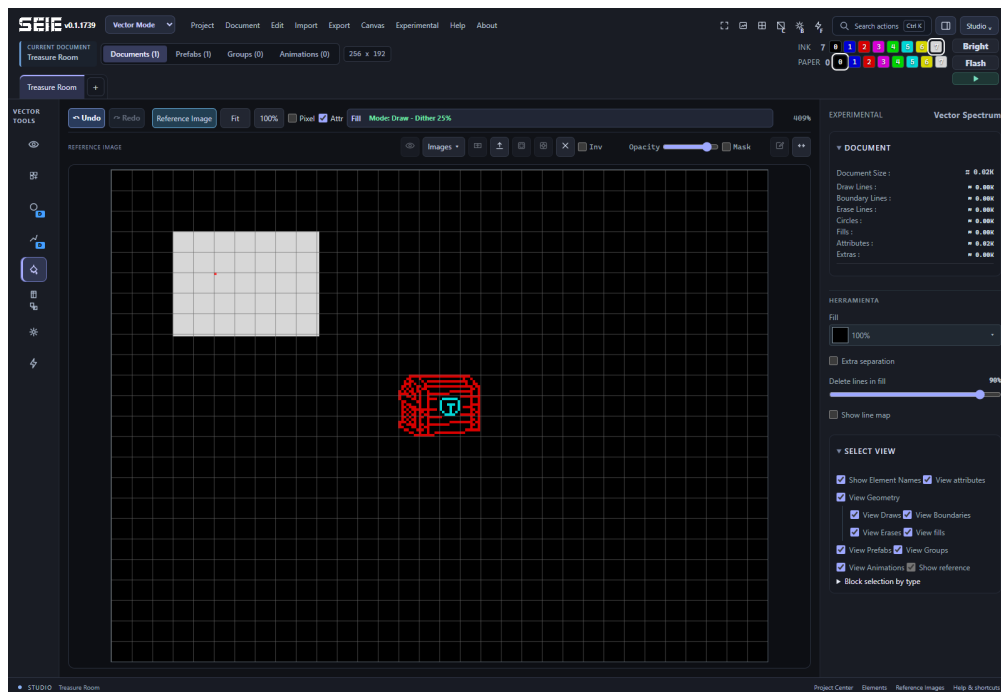
An instance crop clips its visible result without editing the original circle. Materializing or flattening an asset is a different operation. A partially outside circle may not remain a full editable circle after a conversion that enforces strict bounds; inspect the conversion result on a copy.

EXERCISE

Create a circle with a fill inside, then move its centre and change the radius. Compare the seed position and enclosure after each edit. Keep the seed in a meaningful region rather than on the outline.

Fill seeds and effective regions

A seed identifies the region to fill. The final filled area depends on the effective barriers and scene context, not only on the seed coordinates.



Actual seeded fill inside the enclosure. The seed marker is an editor overlay, not exported artwork.

Place a seed deliberately

Use Fill in an enclosed empty region. Pick a point away from narrow corners and outlines. Choose the fill pattern and separation options before evaluating the result.

Edit the enclosure

After moving Draw or Boundary geometry, finish the gesture so the editor can update the fill. Inspect the region, not just the seed marker. A marker that remains in place can legitimately produce a different region after a barrier moves.

Crops and mirrors

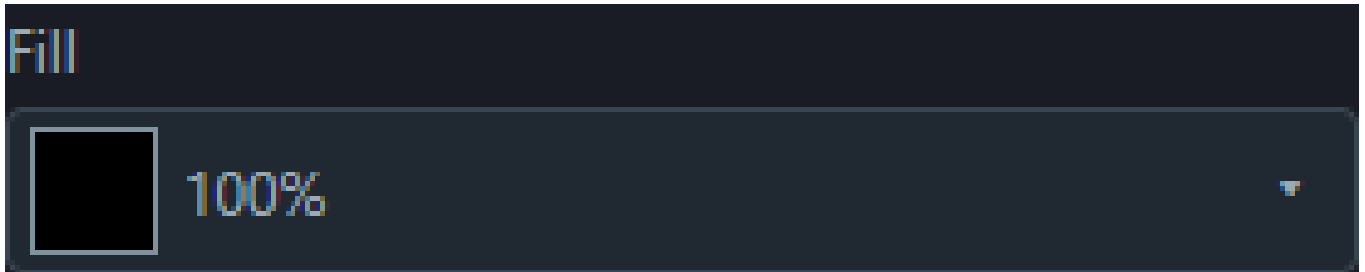
Clipping can leave a visible portion of a filled region while excluding its original seed. SEIE has recovery paths for derived contexts, but the reliable user check is the visible filled region in both editor and target emulation. Do not add a second fill blindly to hide a mismatch.

DIAGNOSE

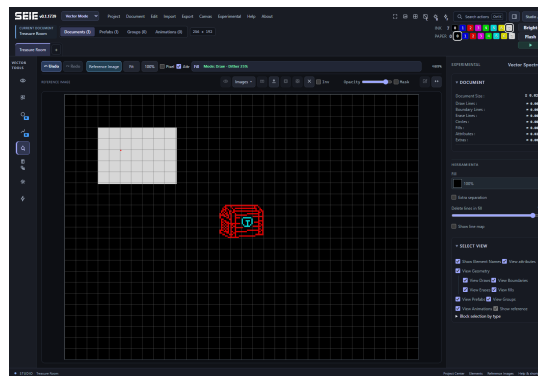
Save a small reproduction containing the enclosure, seed, crop and mirror settings. That is more useful than a screenshot alone when a region differs between editor and runtime.

Patterns and extra separation

The fill's region and the pattern applied inside it are separate decisions. A correct enclosure can still look wrong with an unintended pattern.



The Vector Fill Type control is separate from the line stroke pattern.



The 100% Fill Type shown above produces this solid region. The seed is an editor guide; it is not part of the exported picture.

Density

Use 100% for a solid fill or a lower-density pattern to alternate pixels. The fill pattern is not the attribute colour; Color changes INK/PAPER for the cells containing it.

Extra separation

Separation controls spacing near barriers. It can alter thin shapes and competing regions, so inspect the smallest details after enabling it. Treat it as a design parameter, not a universal repair switch.

Keep the original construction understandable

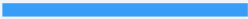
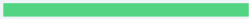
Prefer one intentional region with a clear seed over multiple overlapping fills added during troubleshooting. Preserve the original seed and settings in an editable project; a raster export alone cannot explain how the region was constructed.

EXERCISE

Duplicate a small test document. Compare solid and patterned fill, then toggle separation on one copy. Inspect the contour at pixel scale and the whole shape at native scale before choosing the final version.

Read Show Line Map

Show Line Map is a diagnostic view of command types. Its colours are editor guides and are not the colours stored in the scene's attributes.

			
Solid Draw	Boundary	Patterned Draw	Erase
visible + blocks	invisible barrier	no solid boundary	removes pixels

The four line colours

Blue identifies solid Draw lines that block fills. Green identifies patterned Draw without that blocking boundary. Orange identifies pure Boundary lines. Purple identifies Erase lines.

Use visibility in layers

Select View > View Geometry controls the geometry family. Draws, Boundaries and Erases can be shown or hidden separately. Fills and element types have their own visibility options. Hiding a category does not delete its commands.

Compare with the finished scene

Toggle Line Map off to judge final colours. If the whole scene seems blue or green, first check the diagnostic overlay before recolouring the artwork. The legend above the canvas is tied to Line Map visibility.

CHECK

Select a known Boundary and compare its inspector type with the orange overlay. If a guide is absent, check both Show Line Map and View Boundaries rather than changing the underlying command.

Edit without changing the wrong owner

Elements is the unified selection tool for Vector geometry and placed elements. Its available operations depend on the selected types.

TEACHING DIAGRAM

Find the owner before editing

A

Local command

Use Elements
in this document

B

Prefab interior

Open Edit Master
or Edit in Context

C

Referenced source

Check Read Only
and edit the source

Local geometry

Click a line, circle or fill to select it. Shift adds or removes items. The selection list can help identify overlapping or very small commands. Move Geometry is available for compatible selected geometry in the relevant editors.

Placed elements

Selecting a prefab selects its instance, not its internal vectors. Groups act as a unit unless you deliberately select a member with Ctrl or enter the group editor. Use the context menu for operations compatible with the whole selection.

Protected content

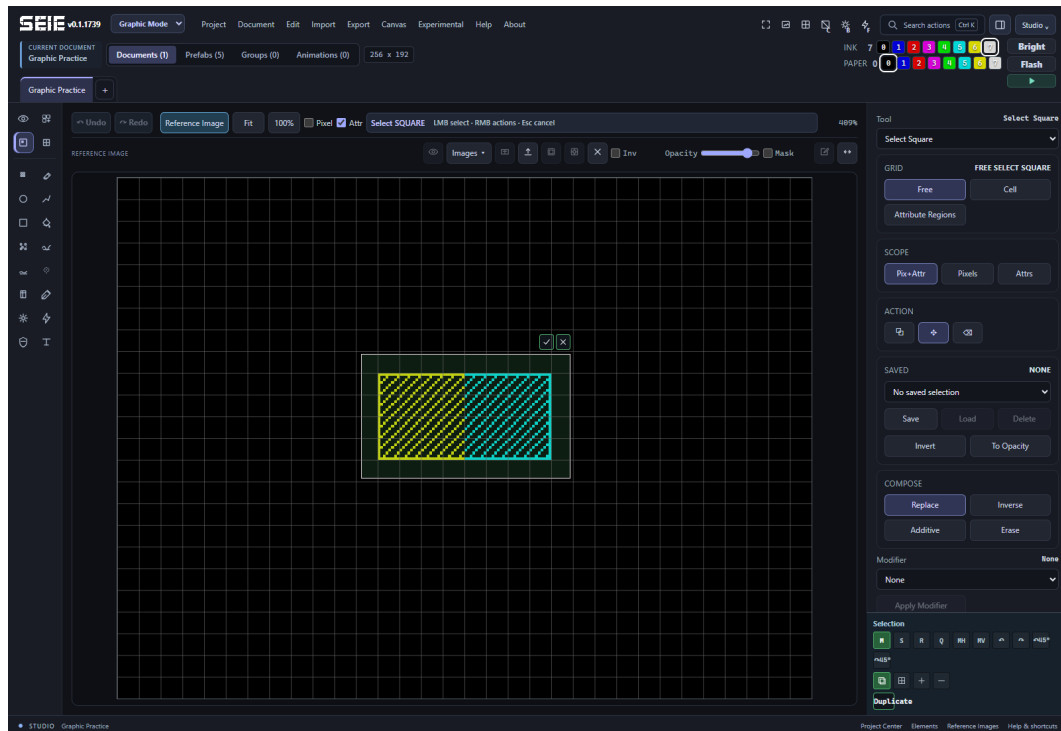
Blocked instances and blocked selection categories cannot be manipulated until unlocked. Read Only references protect inherited content while allowing local work. Use the scene list and context labels to identify which protection applies.

SAFETY RULE

If a selection spans several kinds of object, do not assume every transform applies. Crop is for compatible prefab/group selections, not an arbitrary mixture of fills, lines and instances.

Pixels or elements?

Graphic Select and Elements answer different questions: which pixels should move, or which reusable object should move?



Graphic Select stages a pixel-region operation. The selection controls differ from an element instance inspector.

Graphic Select

Use it to move, duplicate, mirror or transform an ordinary raster region. Scope decides whether pixels, attributes or both participate. The floating result remains temporary until Apply.

Elements

Use it to select prefab, group and animation instances, and local Vector geometry where applicable. Moving an instance keeps its master link, crop and orientation. Copying pixels does not create a reusable placement link.

Choose before dragging

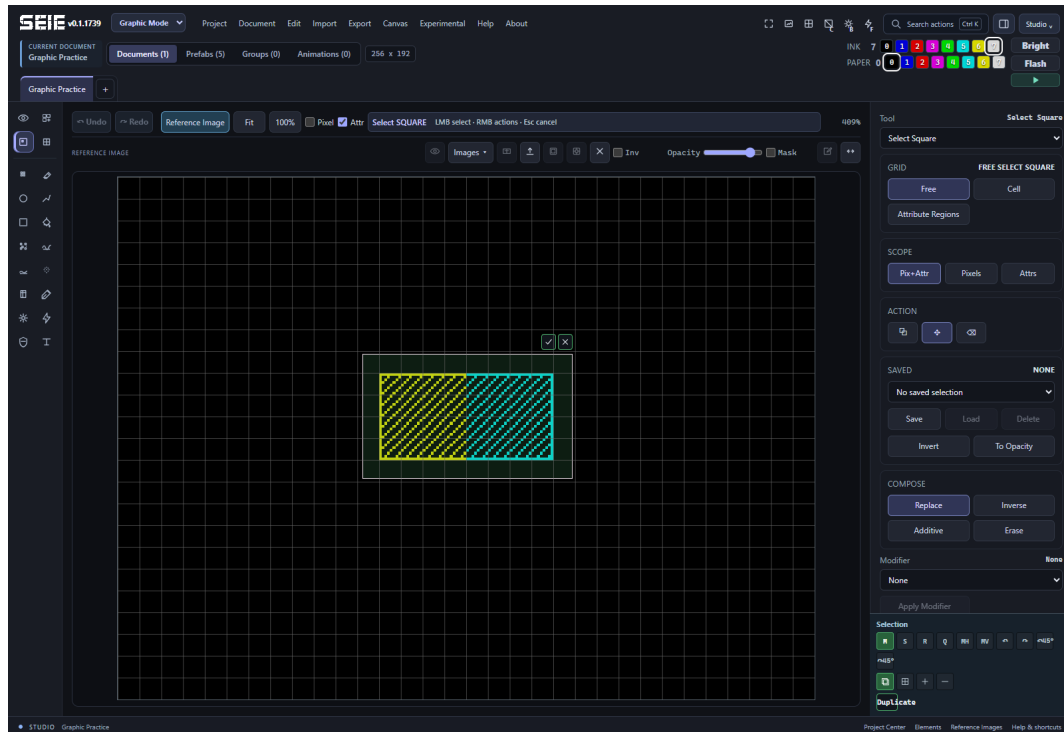
If the intended target is a placed chest, use Elements. If it is a patch of ordinary bitmap wall texture, use Graphic Select. To change the artwork inside the chest, enter its master or context editor.

CHECK

Read the selection label and controls. If you intended an instance but see pixel composition choices, cancel and change tools before committing.

Make a Graphic selection

Selection shapes identify an area; the selected content and action determine what happens to it.



Actual Graphic selection around the practice artwork; the frame identifies the pending region.

Choose the selection shape

Square, Circle, polygon Line, automatic, fill-area and Attribute Regions cover different tasks. A polygon selection closes with a quick double click. Attribute Regions selects complete 8 x 8 cells and is useful when colour alignment matters.

Choose the content

Use pixels only for a bit-pattern adjustment, attributes only for colour cells, or both when moving complete artwork. Moving only pixels can leave their original colours behind, which may be intentional or surprising.

Inspect before applying

Create the selection, drag it a small distance and look at both source and destination. In Move, the preview represents relocation. In Copy, it represents duplication. Cancel discards unapplied work; Apply commits the staged result.

EXERCISE

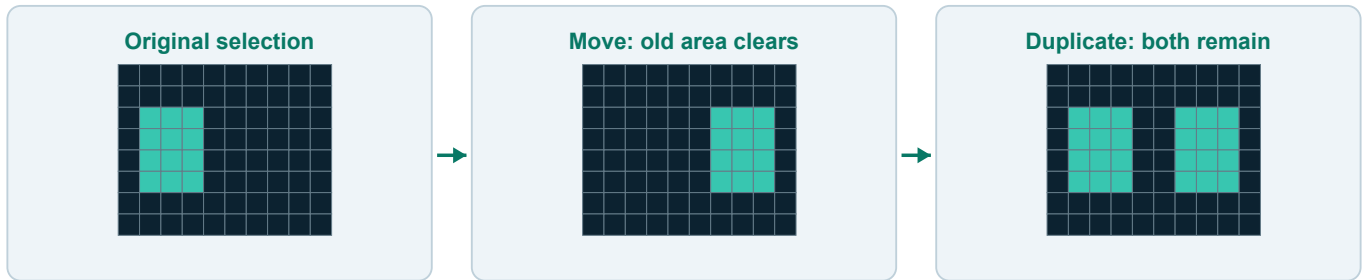
In Graphic Practice, select the left half of the pattern including attributes. Move it by one cell, Cancel, then repeat and Apply. Confirm that cancelling and applying give different, predictable results.

Move, duplicate and compose

A staged selection lets you compare the result before changing the underlying canvas permanently.

TEACHING DIAGRAM

Duplicate preserves the source; Move relocates it



Move

Drag inside the selection. Selection actions are docked compactly at the bottom of the right inspector; only Apply and Cancel remain beside the canvas selection. This keeps the artwork readable while positioning.

Duplicate

Duplicate creates a movable copy beside the original rather than exactly on top of it. The original is no longer the active floating selection. Position the copy, then Apply or Cancel.

Composition

The small composition controls in the right selection dock determine how staged data combines with the destination. They are not layer-order commands for prefab instances. Hover for their names and compare the preview on a small test area before applying.

CHECK

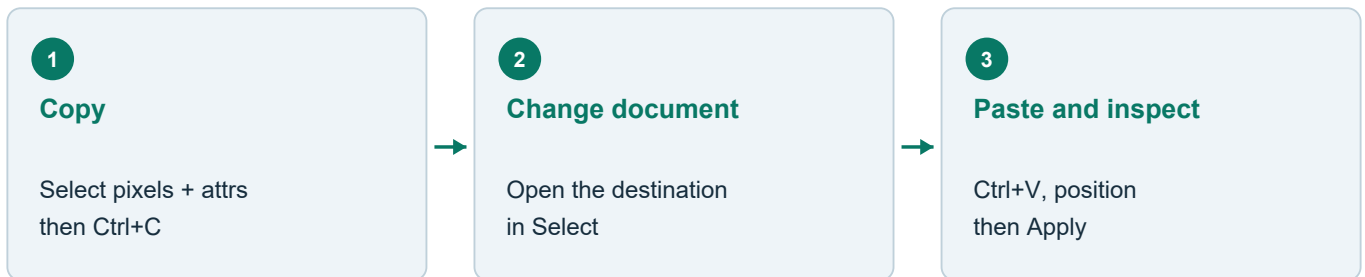
Do not switch contexts while a result is still undecided. Apply the desired operation or Cancel it, then save. This separates temporary previews from the project state you expect to reopen.

Clipboard across documents

Copy and Paste are useful for one-off transfers. Reusable assets are usually a better choice when the same artwork must stay linked.

TEACHING DIAGRAM

Clipboard transfer between workspaces



Use the normal clipboard commands

Ctrl+C copies, Ctrl+X cuts and Ctrl+V pastes the active compatible selection. Graphic selection context menus also expose Copy and Paste. Paste begins at the original canvas position rather than an arbitrary centre point.

Between browser tabs

Compatible SEIE tabs can exchange Graphic selection data through the clipboard. Browser clipboard permissions and secure-context rules can affect access. Keep the source tab open and use a direct user gesture when the browser requires one.

Verify the transfer

Check pixels, attributes and position before Apply. If only ordinary text or an unexpected image appears, cancel and repeat the copy from the intended selection. Saving both projects before a cross-project cut is safer than relying on a single clipboard payload.

PREFER REUSE

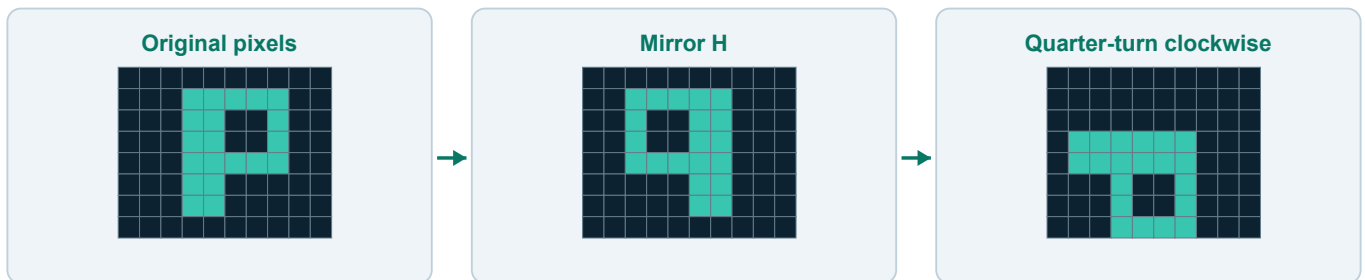
For a linked chest used in many rooms, export/import its prefab or group with dependencies. A copied raster region is not a shared master and will not receive future master edits.

Mirror and rotate selections

Transforming a Graphic selection changes raster artwork. Transforming an element changes its placement settings while retaining the asset link.

TEACHING DIAGRAM

Use an asymmetric shape to check orientation



Graphic operations

With a region selected, right click for compatible Mirror H, Mirror V and rotation actions. The Graphic selection tool includes 45-degree rotation as well as the usual right-angle rotations. Preview before applying.

Pixel-grid consequences

A 90-degree turn maps grid directions cleanly. A 45-degree turn needs raster resampling and may change thin diagonals or small lettering. Inspect at native scale; no transform can invent subpixel detail on the Spectrum screen.

Preserve colour intent

Include attributes when the colour layout should travel with the pixels. A rotated selection can cross different cell boundaries, so examine the destination cells after the transform rather than judging only the black-and-white silhouette.

EXERCISE

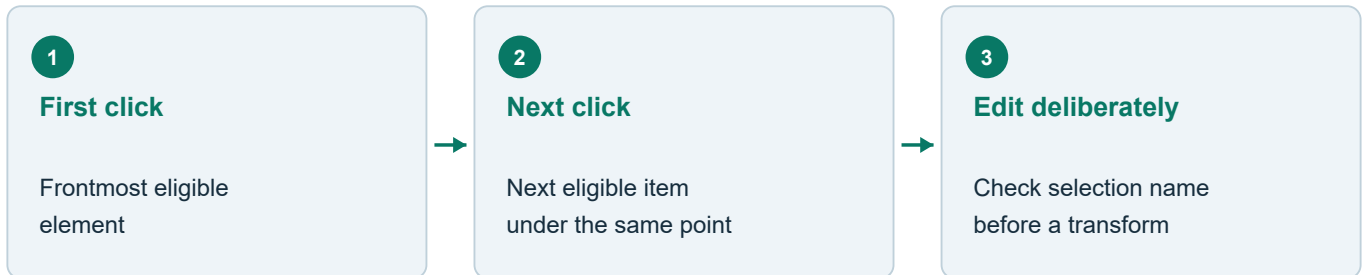
Copy a small asymmetric motif and compare H, V, 90 degrees and 45 degrees. Keep the source unchanged until you have selected the version you want.

Select through overlapping layers

Scene objects can overlap. The frontmost visible, eligible item is the natural first target, but it may not be the one you intend to edit.

TEACHING DIAGRAM

Repeated clicks walk down the overlapping stack



Cycle or use the list

Repeated clicks at the same overlap can reach deeper objects in layer order. Use Elements in Scene for an unambiguous named target when the overlap is dense. Selecting a scene element should bring its list entry into view.

Multiple selection

Shift toggles items in the current selection. Drag compatible instances together. A group is normally one unit; Ctrl intentionally targets a member where supported. Avoid interpreting an internal prefab as a separate loose scene object.

Order and protection

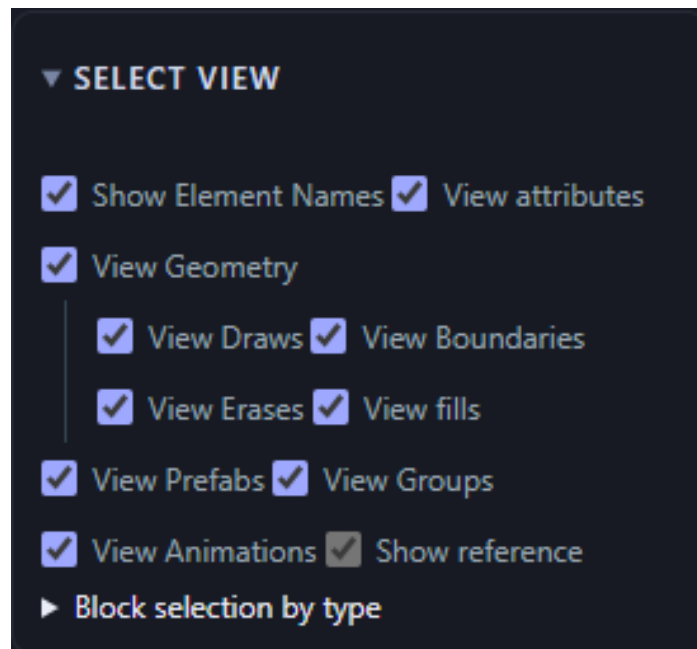
Bring to Front and Send to Back change extremes; Move Forward and Move Back change one step. Group ordering preserves its internal arrangement. Block Instance protects a placement until it is unblocked from the scene list or the applicable context action.

CHECK

When an element cannot be selected, confirm Visible, Blocked and Block Selection by Type. A hidden or protected item is not a missing asset.

View and block are different

Use visibility to inspect a composition and blocking to protect it while working. Neither is the same as deleting content.



Actual Select View controls. Visibility and blocking are separate decisions.

Select View

Graphic mode provides Attributes, Graphics, Prefabs, Groups, Animations and References. Vector mode additionally distinguishes geometry families and fills. Show Element Names controls labels and size text on the canvas.

Block Selection by Type

Blocking a category prevents accidental selection without making its artwork disappear. Instance-level Blocked settings protect individual objects. Use Edit > Block All or Unblock All for eligible objects in the active scene, not for changing library masters.

Avoid false diagnoses

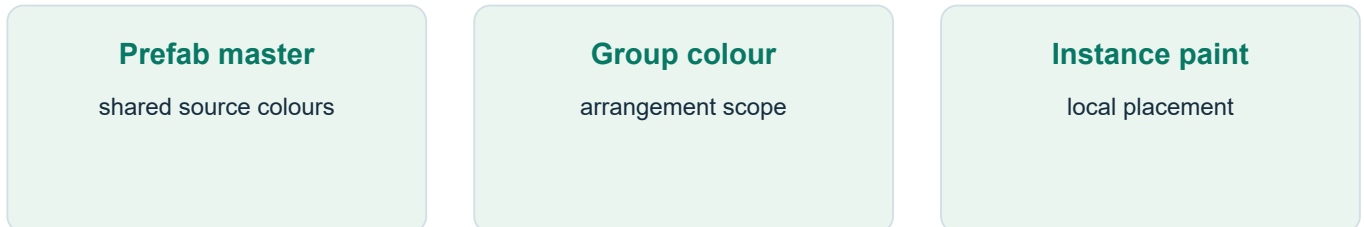
Hidden prefabs should not leave active crop handles behind. If you cannot reach an object, inspect both the scene list and view/block settings. If final output differs, verify export inclusion separately; an editor visibility control is not a universal export-selection rule.

PRACTICE

Hide Prefabs, restore them, block one instance, then unblock it from the list. Observe the difference between not seeing an object and seeing it without being able to move it.

Where does this colour live?

Colour can belong to canvas cells, a prefab master, a variant, a group or a local placement override. Scope matters more than the apparent colour on screen.



Teaching diagram - not an application screenshot.

Paint locally first

Canvas painting over an instance is for local presentation. Recover Color discards those local changes and reveals the applicable source colours again. Use this when a scene-specific tint is no longer wanted.

Publish deliberately

Apply Color to Prefab Master changes shared prefab artwork. Apply Color to Group records the group's colour arrangement without recolouring the masters of its component prefabs. Read confirmations before accepting either operation.

Transparent cells

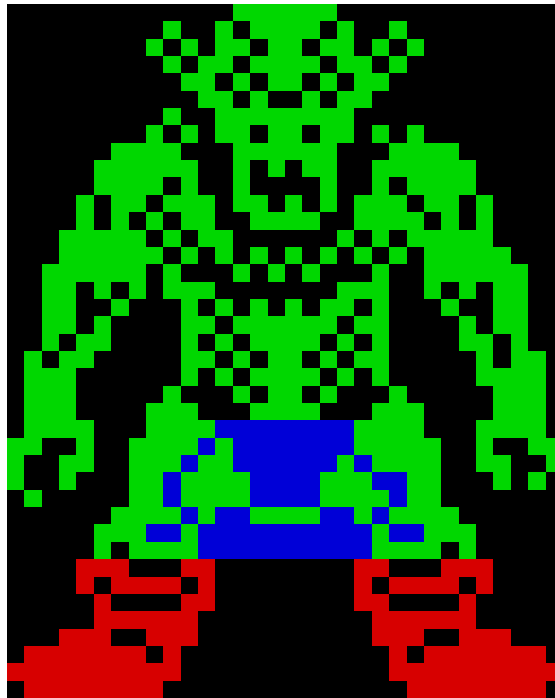
An excluded prefab cell is not a paint target owned by that prefab. Artwork beneath it remains reachable according to the active editor context and protection settings. A black owned cell is not equivalent to that hole.

CHECK

Before publishing colour, open another scene using the same asset and predict what should change. After publishing, inspect that scene to confirm the intended scope.

One asset, many placements

A prefab is the smallest reusable visual building block in this guide. Its master is stored once in the project library; scenes contain instances that refer to it.



Orc_01: one 32 x 40 source image from Medieval. A mirrored placement does not require drawing another orc.

What a placement adds

An instance has a position, orientation, crop and optional local colours. Reuse the same orc in different positions, or mirror it to face the other way. A pose can combine cropped placements of this image with shared arm and leg pieces, then store their arrangement as a group.

What reuse does not promise

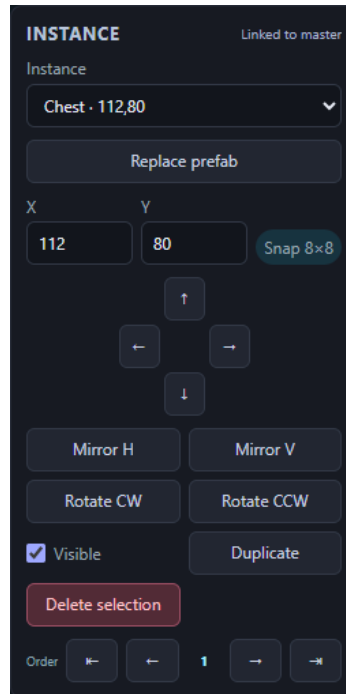
The editable project avoids unnecessary duplicated masters. An export may still materialize crops, choose a different encoding or include placement overhead. Measure the chosen export rather than assuming that three uses always cost exactly one asset plus three tiny records.

STARTING POINT

Open 04-Orc-Source.siep.json to capture the ordinary pixel image, or 04a-Orc-Created.siep.json for the saved prefab. The later Orc checkpoints show mirrored components and pose groups, not separately redrawn characters.

Place and align a prefab

Keep modular artwork on the attribute grid unless a particular design requires finer positioning.



Position, snap, mirror and quarter-turn controls belong to the selected instance.

Find the asset

Select Elements and expand Library/Asset > Prefabs. Search by name when the library is large. Click the asset card to get a placement preview, then click the desired canvas position.

Correct the position

Select the placed instance and use its X/Y controls when exact coordinates matter. Snap 8x8 keeps cell-aligned components predictable. Dragging an existing instance moves it; choosing the library card prepares another placement.

Inspect the result

Confirm the footprint, colours and layer order. A prefab may extend partly outside the canvas while a visible part remains inside; that is different from changing the source artwork's bounds. Crop and Work Area clipping can both affect what is visible.

EXERCISE

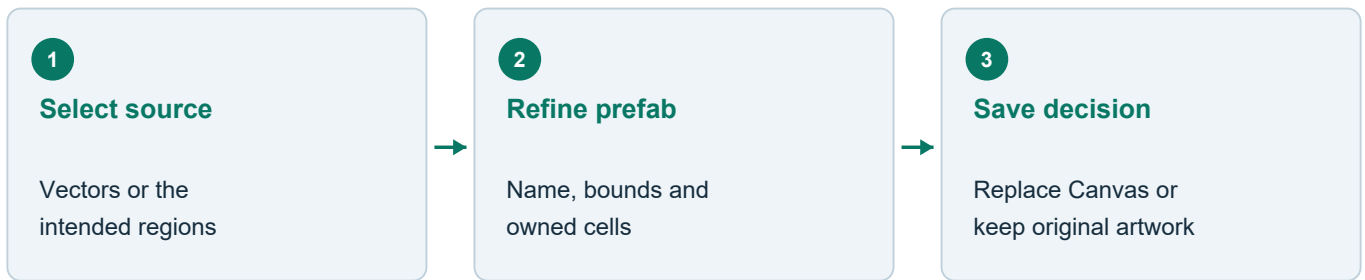
Place Orc_01 twice in an empty area, then mirror only the second instance. The library should still contain one Orc_01 master. The two placements retain independent positions and orientations.

Create from Vector artwork

Capture a meaningful reusable unit rather than turning every incidental line into its own asset.

TEACHING DIAGRAM

Create a reusable Vector asset



Choose the source

Create Prefab in Vector mode can use the current vector selection, attribute regions, all document vector content or an empty asset. Before capturing, inspect which local geometry and cells the chosen method includes.

Edit the new asset

Give it a unique name. Check the editable bounds and owned attribute regions in the prefab editor. Remove accidental distant cells; they can make the asset's footprint far larger than its useful artwork.

Replace or preserve the canvas

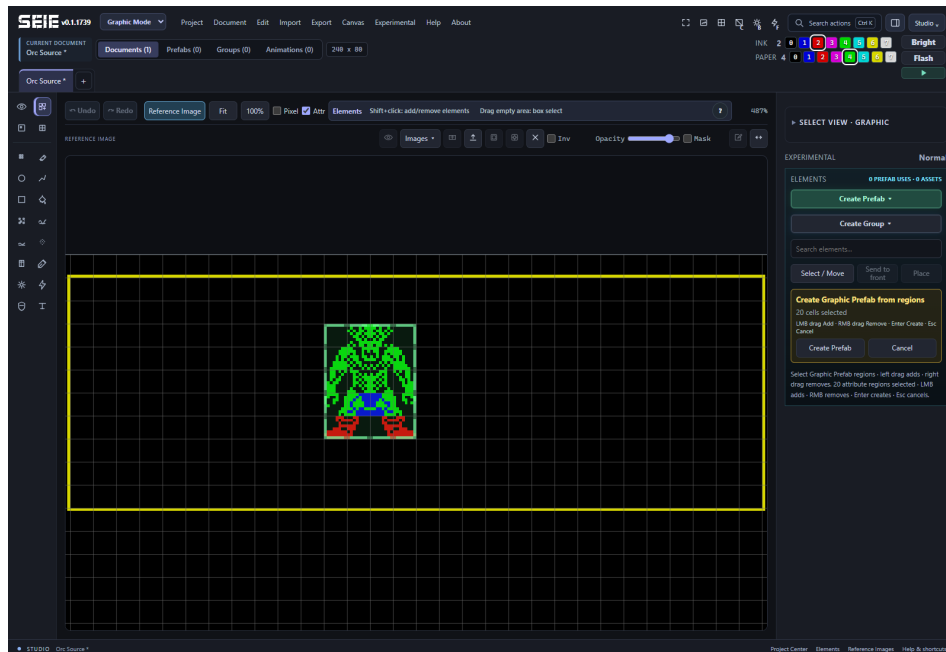
For selection/region creation, the replacement decision is made when saving the new prefab. Replace Canvas substitutes the captured content with an instance. Keeping the original leaves the source artwork available separately. Cancelling creation should not leave an unwanted replacement.

CHECK

After replacement, temporarily hide the new instance and inspect the underlying canvas. The captured source geometry should not remain as an accidental second copy. Keep a pre-capture project checkpoint for comparison.

Create from Graphic regions

Graphic capture packages exact pixels and attribute ownership into a reusable asset. Its holes and limits are as important as its visible bits.



Real creation in SEIE: the Orc source image is captured with a 4 x 5 selection of attribute regions.

Select the cells

Use Create Prefab > From Graphic Regions and mark the intended attribute cells. Include complete cells where their colours belong to the object. The selection overlay is an editing aid; inspect the actual pixels underneath it.

Name and refine

Enter a unique asset name. In the new master, check the captured colours and use Attribute Regions to remove cells that should be transparent. A black cell is not automatically a hole.

Save the intended relationship

When offered Replace Canvas, decide whether the new prefab should replace the captured artwork or leave it as independent canvas content. Save the project after checking both the new library asset and its scene placement.

EXERCISE

Open 04-Orc-Source.siep.json. Capture the 32 x 40 orc at screen X 96, Y 24: four columns by five rows of cells. Name it Orc_01 and inspect the saved master. The source checkpoint has no prefab masters, so this is a real capture exercise.

Empty and whole-area creation

Different creation routes are useful for different starting points. Do not use a whole-screen capture when a small modular asset will do.

TEACHING DIAGRAM

An empty Graphic prefab starts with a 5 x 5 footprint



Empty Graphic prefab

Choose Empty Prefab, provide a name and begin with the default 5 x 5 attribute cells: 40 x 40 pixels. The initial colours are white INK on black PAPER, with BRIGHT and FLASH off. Resize in the master editor as needed.

Entire screen or Work Area

Graphic creation offers From Entire Document Screen, or From Entire Work Area when a Work Area is enabled. This captures the active area into an independent Graphic prefab and preserves the original document.

Empty Vector prefab

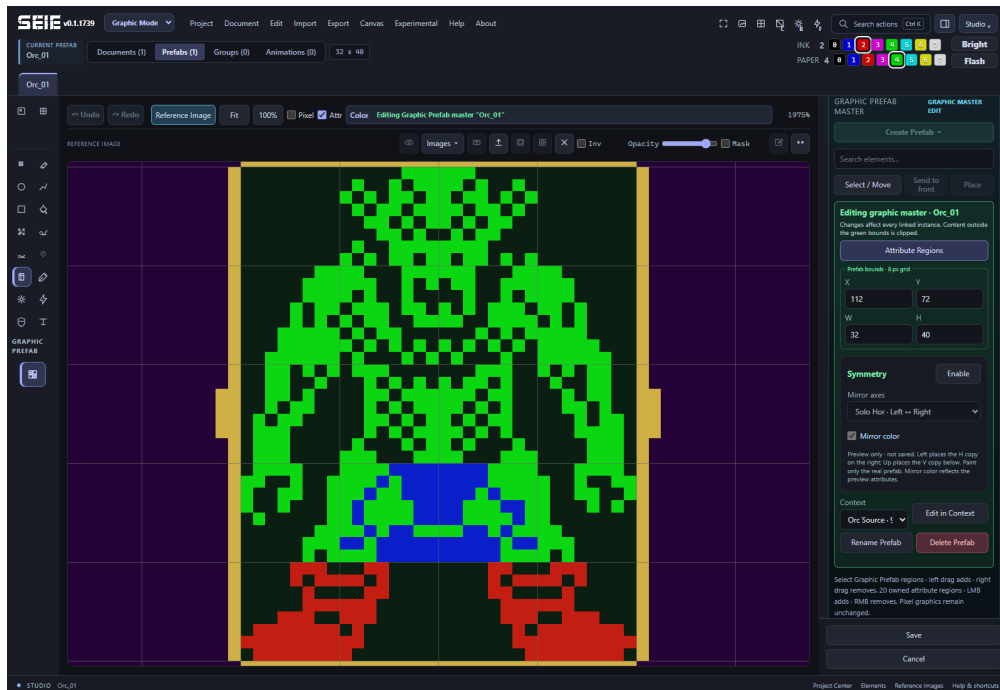
Use the empty route when building editable commands from scratch. Start with sensible bounds and owned cells, then create geometry inside them. Check limits before adding distant guides or temporary construction lines.

CHOOSE DELIBERATELY

Whole-area assets are useful for fixed panels and backgrounds. They are less flexible than a kit of small modules when only a door, window or character changes between scenes.

Master bounds and owned regions

The bounding rectangle and the set of owned attribute cells are related but not identical. A rectangular asset can contain transparent holes.



A prefab master displayed separately from its room. Background tint identifies the editing context.

Bounds

Yellow boundary handles define the editable rectangular extent when Attribute Regions is active. They are not instance crop handles. Enlarging bounds creates room for additional artwork; shrinking them can exclude content, so inspect before saving.

Ownership

Attribute Regions enables or disables individual 8 x 8 cells. Disabled cells are transparent to the prefab composition. Use this for gaps between a character's limbs, a window opening or an irregular modular silhouette.

Keep content anchored

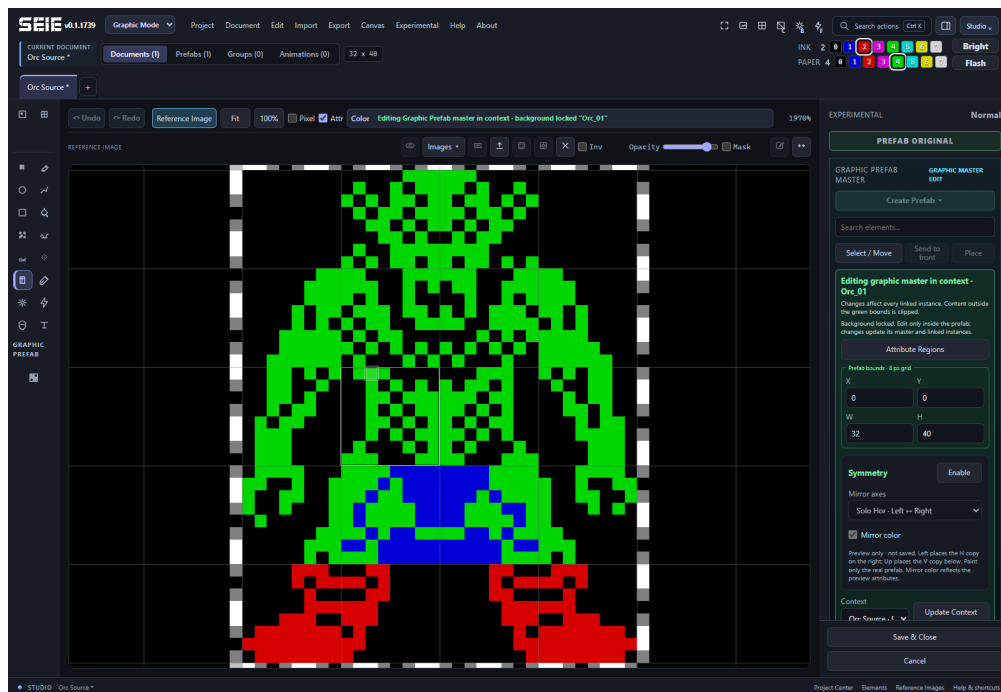
Resizing limits should not be used as a substitute for moving the artwork. After changing top/left bounds, check that the intended pixels and colours still occupy the expected cells.

CHECK

Save, return to the room and inspect all affected edges. Then reopen the master to confirm the bounds and ownership survived the round trip.

Edit Master or Edit in Context

Both routes edit asset artwork. Their difference is the visual context in which you work, not a guarantee that changes are local to one placement.



Actual Graphic Edit in Context: the surrounding document provides a locked positioning context.

Isolated master

Use the Prefabs workspace or Edit Master to focus on the asset without the room. This is the best starting point for correcting owned regions, rebuilding an outline or comparing the original with variants.

Context editing

Edit in Context shows the surrounding scene as a protected background. Work on the active prefab, then finish editing to return to the owner scene. A Graphic prefab inside a Vector document uses its appropriate editing tools and returns to the Vector document afterwards.

Finish explicitly

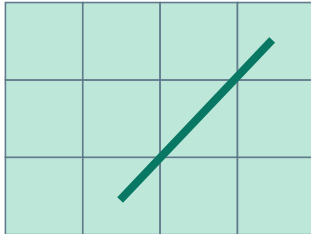
Use the fixed Save or Save & Close action appropriate to the entry context. The Prefabs workspace can save while remaining on the asset; a temporary context editor normally returns to its owner. Cancel discards the pending editing session where offered.

SAFETY RULE

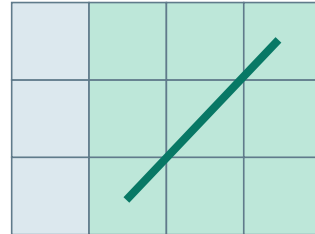
If only one instance should differ, consider local colour, crop, orientation or a variant instead of changing the shared master artwork.

Crop without moving the source

An instance crop hides source regions. It should not slide the remaining artwork inside the visible window or rewrite the master.



Source: 4 x 3 cells



Same source, left side cropped

Adjust the visible sides

Select an instance with Elements and drag the relevant crop edge. Inspect the live preview, then release and compare. Source-space crop settings travel through the instance's orientation rather than being interpreted as a new drawing origin.

Combine with orientation

After cropping, Mirror H or Mirror V should show the same retained source area, reversed. Rotation changes the displayed orientation of that retained area. If you see a different part of the asset, compare against an uncropped copy before continuing.

Keep the limits distinct

Master bounds define available source artwork. Instance crop defines the portion used here. Work Area and screen limits determine the final scene visibility. These are three different clipping stages.

EXERCISE

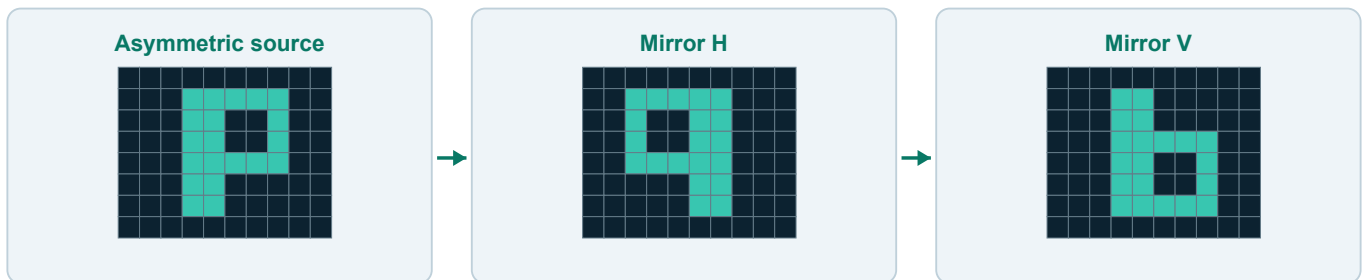
Duplicate Orc_01, crop one column from a side, then mirror the duplicate. The same visible body detail should remain, reversed. The original Orc_01 artwork should be unchanged.

Mirrors and quarter-turns

Instance transforms let asymmetric artwork serve several orientations without creating new source assets.

TEACHING DIAGRAM

The visible crop stays the subject of the transform



Mirror H and V

Mirror H reverses left/right; Mirror V reverses top/bottom. A command toggles the current mirror state rather than promising an absolute orientation independent of earlier rotations. Judge the result using an asymmetric detail.

Rotate

Use clockwise or counter-clockwise quarter-turn actions. A tall asset becomes wide at 90 degrees. Check placement and crop handles after the turn, especially close to screen edges or within groups.

Keep a useful anchor

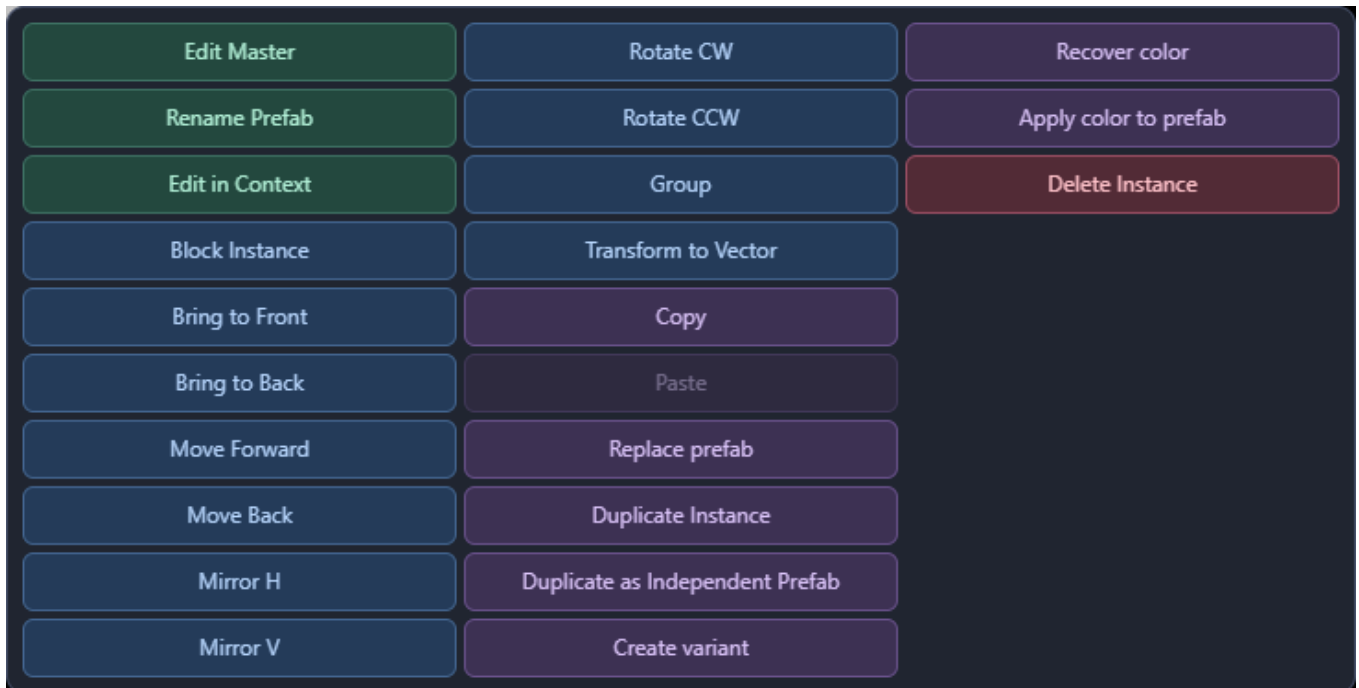
After each transform, inspect both the visible footprint and its alignment to neighbouring cells. Do not fix an unexpected shift by permanently offsetting the master; first establish whether the change belongs to placement, crop or source bounds.

TEST MOTIF

A window with a diagonal highlight is easier to verify than a perfectly symmetric square. Use a visible left/right detail whenever testing a mirror.

Local colour and recovery

Use instance colour when one placement needs a different appearance but should retain the same artwork.



Actual prefab menu: Recover Color and Apply Color to Prefab have different scopes.

Paint the placement

Return to the document canvas, choose Color and paint the visible owned cells. The local change belongs to that placement. Check another use of the same master to confirm it has not been unintentionally changed.

Recover Color

This removes local colour customisation and returns the placement to its applicable source colours. It is useful for resetting a temporary tint. It does not mean recovering deleted geometry or reversing every transform.

Apply to the master

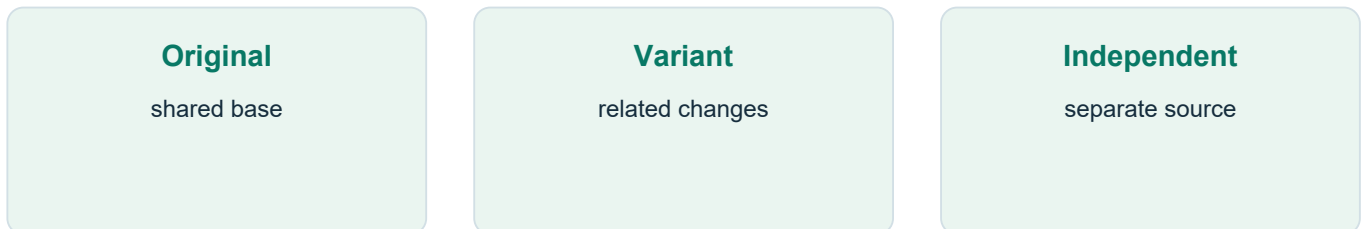
Apply Color to Prefab Master explicitly transfers the local colours to the shared source. It requires confirmation because other uses can be affected. Check the asset's orientation and crop context before publishing a partial recolouring.

PRACTICE

Recolour one of the three chests, save, reopen and Recover Color. Then repeat on a copy and publish to the master, observing the other linked placements. These are deliberately different workflows.

Variants and independent copies

Choose the relationship that matches your intention. Similar-looking assets do not necessarily have the same ownership model.



Teaching diagram - not an application screenshot.

Variant

A variant is based on an original prefab and carries its own changes. The editor identifies the original and provides navigation between it and its variants. Use variants for related designs that should remain recognisably part of one family.

Independent master

Duplicate as Independent Prefab creates unrelated artwork from the current instance, baking its crop and orientation into the new asset. Duplicate Master Prefab makes a separately named master copy. Inspect the exact operation label before accepting.

Ordinary duplicate

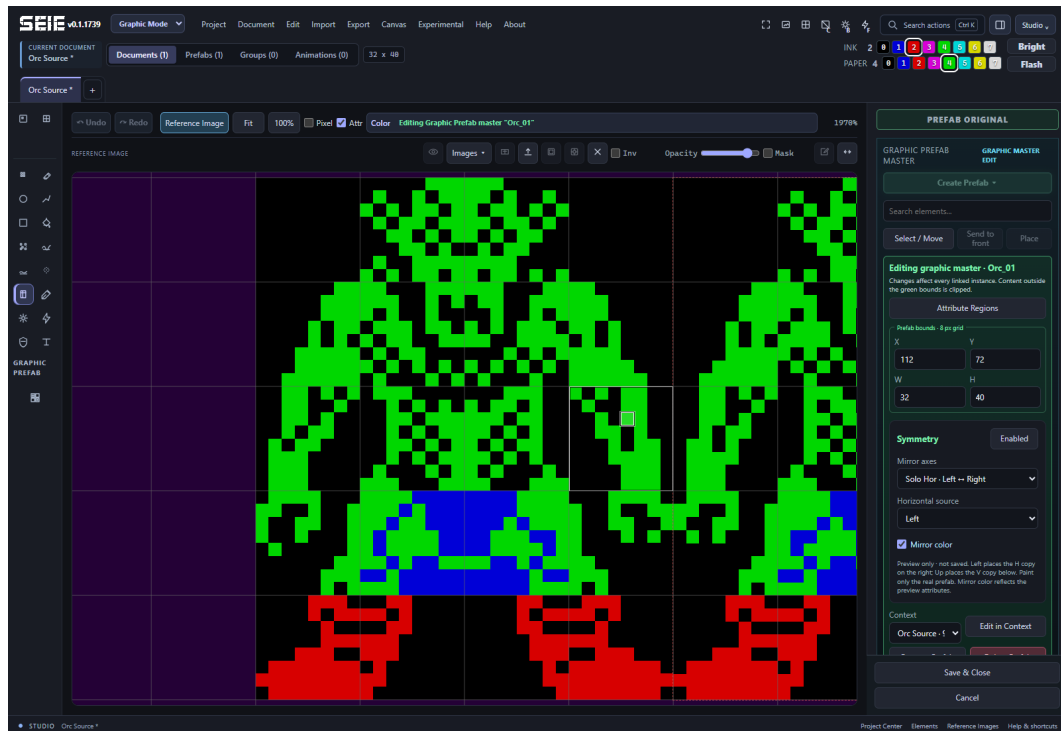
Duplicate Instance only creates another placement of the existing source. It is the inexpensive organisational choice when the artwork should remain shared.

DECISION

Another chest in the same room: instance. A damaged chest related to the original: consider a variant. An unrelated prop that merely starts from the chest's shape: independent master.

Graphic symmetry is a preview

Graphic prefab symmetry shows how reusable mirrored placements will look together. Only the actual source prefab is edited and saved.



Real Orc Graphic master with symmetry preview enabled. The reflected preview is not a second stored source image.

Set axes and source

Choose H, V or H+V. H with Left source places a mirrored preview to the right of the real prefab; Right places it to the left. V with Up source places it below; Down places it above. H+V shows three adjacent previews.

Paint the source

The external copies update while you work, including staged selection movement. Mirror Color determines whether the preview reflects attributes as well as pixels. The copies are not additional editable source regions.

Build the real arrangement later

Save the source, then place mirrored instances in a group to reproduce the desired combination. This retains one source asset instead of storing the preview as duplicated pixels.

CHECK

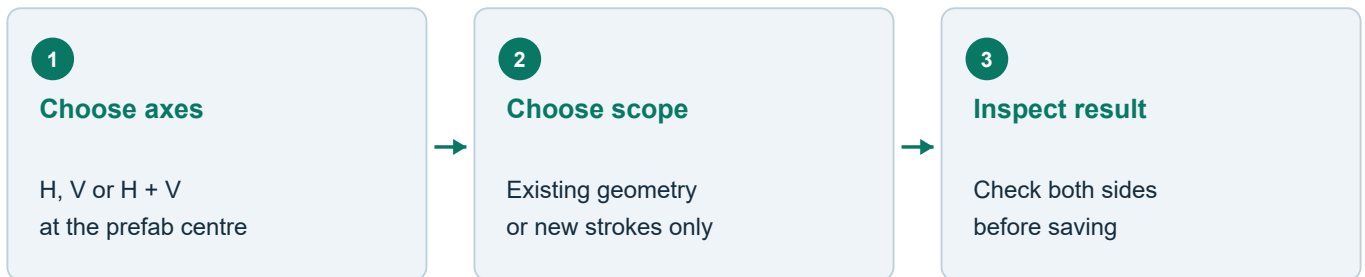
Reopen the master after saving: it should contain only the source artwork. A preview outside its bounds must not unexpectedly enlarge the asset or become a second saved copy.

Vector symmetry is a construction tool

Vector symmetry can generate mirrored commands. Do not assume it has the same read-only external-preview semantics as Graphic prefab symmetry.

TEACHING DIAGRAM

Vector symmetry constructs geometry



New geometry only

With Mirror Existing Geometry disabled, symmetry applies to newly added geometry while leaving the existing construction intact. The symmetry axes use the centre of the current prefab.

Existing geometry and source

Enable Mirror Existing Geometry when you want the existing source side reflected. Choose Left/Right or Up/Down for the enabled axes. If opposite-side geometry already exists, read the deletion confirmation carefully.

Intersections matter

Geometry wholly on the discarded side can be removed; crossing lines can be clipped at the source division. Circles are treated differently from simple crossing lines. Inspect the resulting command list and fills rather than assuming every command was split identically.

CHECK

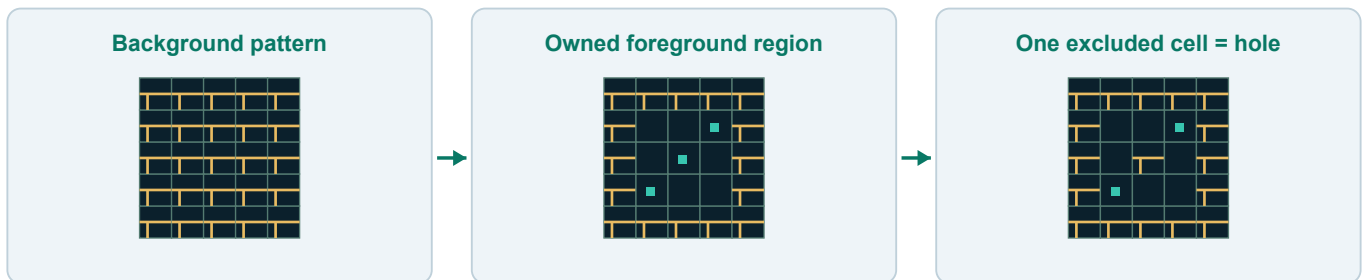
Save a checkpoint first. Use an asymmetric test shape, then inspect both the source and generated side with Line Map. Mirror Color is a separate choice from mirroring the vector construction.

Layers, holes and overlapping assets

When prefabs overlap, the front object owns the cells it covers. Correct ownership prevents unwanted background geometry leaking through.

TEACHING DIAGRAM

Owned zero pixels replace; holes preserve the background



Order first

Use Bring to Front, Send to Back or one-step Move Forward/Back commands. An instance's visual position in the scene depends on composition order, not just the order in which its master appears in the library.

Holes are deliberate

A deselected owned region leaves the content underneath visible. An owned cell with black PAPER still covers its cell according to the composition rules. If two props appear mixed, inspect order, crop and cell ownership before recolouring them.

Compare editor and output

Test at least one overlap in the target export. Cropped, rotated or mirrored assets can take different materialisation paths, so an uncropped single-prop test is not sufficient evidence for every combination.

EXERCISE

Overlap two orc placements and change their order. Inspect an excluded region in a test master. Compare transparent ownership with simply erasing its pixels: black owned cells still cover what is below.

Replace, flatten or convert

These operations change different relationships. Keep an editable project checkpoint before using them on important artwork.

TEACHING DIAGRAM

Choose how much editability to preserve

A

Replace asset

Keep an instance with a different source

B

Transform to Vector

Materialise the visible vector result

C

Rasterize

Write the visible bitmap into the canvas

Replace

Replace changes the source asset used by a placement. Position and transformation intent should be preserved, but different source dimensions can change the meaningful crop. Compare the result when replacing differently sized assets.

Flatten a placement

Graphic Rasterize Prefab/Group turns the visible placement into ordinary canvas graphics. Vector Transform to Vector materializes compatible editable geometry. You lose the placement's shared-update relationship for that flattened content.

Convert authoring mode

Converting Vector masters and variants to Graphic changes shared assets used elsewhere; the confirmation warns about those uses. It is not merely switching the container document's tool palette. Test the affected scenes and target export before discarding the vector checkpoint.

RULE

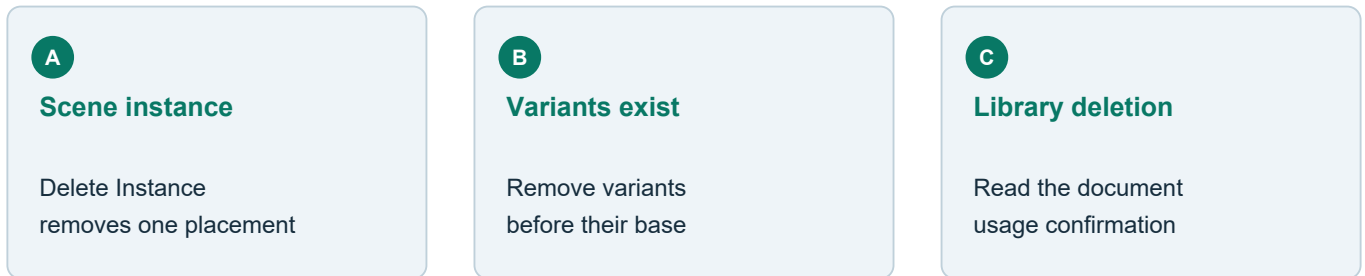
Use flattening for a deliberate one-off edit, not as the routine method of duplicating a reusable object. Duplicate Instance preserves the reusable source relationship.

Delete safely and transfer assets

Deleting a placement, deleting a library master and deleting a variant have different consequences.

TEACHING DIAGRAM

Delete an asset only after reviewing its dependants



Delete Instance

The canvas context menu removes the selected placement. The master remains available in Library/Asset. This is the right action when a prop should disappear from one room only.

Delete the library asset

Library or master deletion checks placed uses and variants. A base with variants cannot be deleted until those variants are removed. Read the listed document uses before removing the asset and its placements.

Import and export

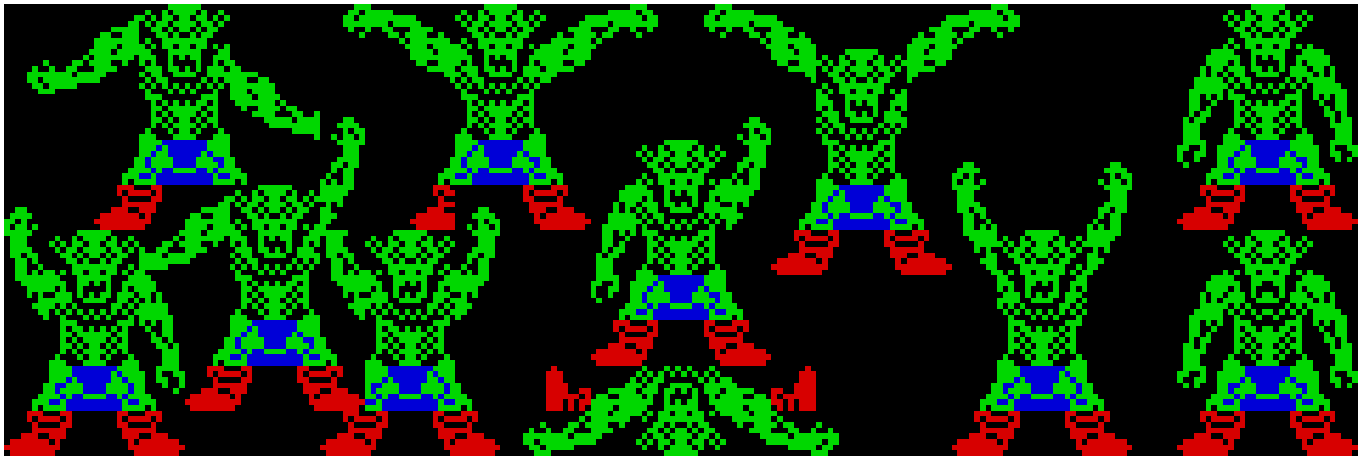
Use Import/Export Prefabs for reusable assets, not a PNG when you need editable structure. Project save preserves the complete library context. After importing, inspect names, variants, dimensions and one test placement before merging large libraries.

CHECKPOINT

Keep both the source project and exported asset file. Reimport into a small temporary project to verify the transfer instead of discovering a dependency problem in a large production scene.

Reuse an arrangement

A group combines components into a larger reusable design. Its prefabs retain their own masters; the group does not need to duplicate their artwork.



The Orc pose sheet from Medieval: the same base character and shared pieces assembled in different positions, crops and orientations.

Why not one huge prefab?

A group keeps the parts individually meaningful. Reposition an arm, mirror a leg or crop a body placement without redrawing the complete character. Save the composition as a pose group. A flattened picture of every pose would lose these shared component relationships.

Shared arrangement, local placement

The group master defines the arrangement. Each group instance positions, crops and transforms that arrangement in a scene. Local edits can override a particular instance without immediately replacing the master definition.

START HERE

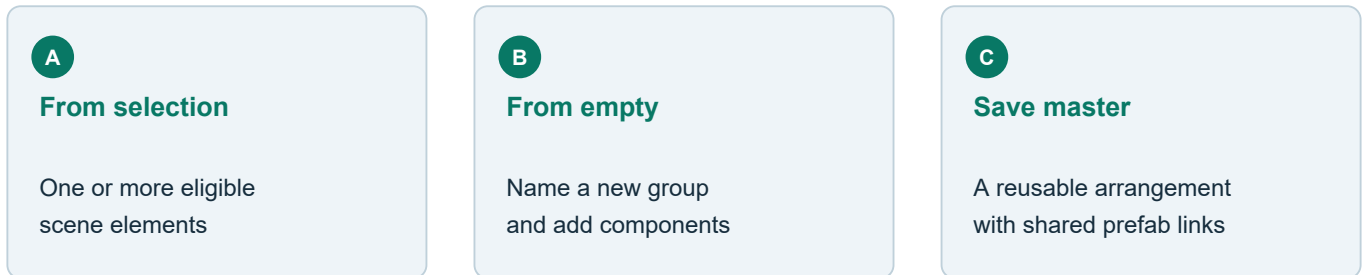
Open 05-Orc-Mirror-Pair.siep.json. Its single visible character is assembled from two cropped, oppositely oriented placements of the same Orc_01 image. Edit Group exposes the two components.

Create from selection or from empty

Create Group supports both assembling existing content and designing a reusable arrangement from scratch.

TEACHING DIAGRAM

Two ways to create a group



From current selection

Select the eligible components with Elements and Shift. A single prefab can form a group too. Choose Create Group > From current selection and give it a unique name. The selected parts become the group's members.

New empty group

Choose the new-group route to open a fresh group master canvas. Add prefab assets and local geometry there. Give the group a descriptive name before building many similar arrangements.

Keep a clean component set

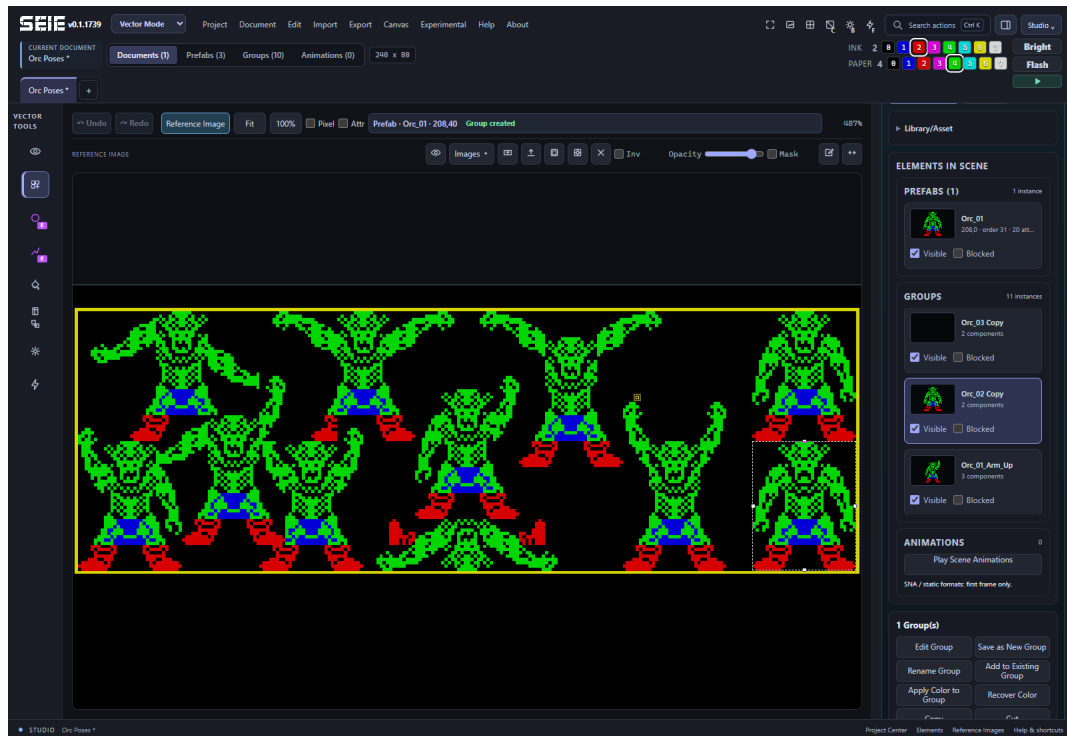
Do not include unrelated background lines merely because a box selection touched them. Inspect the group's component list, then save and reopen the project. Its library definition and scene instance should both be present.

EXERCISE

Select your two mirrored Orc_01 components and create Orc Standing. Later, arrange the shared arm and leg pieces into a new pose and save it as another group, not another full character bitmap.

Place from the library

Group placement follows the same model as prefab placement: choose a reusable definition, inspect its preview and click to create an instance.



The real Orc pose scene: reusable groups and their shared sources are distinct from placements.

Browse

Expand Library/Asset > Groups. The card shows the group's name and thumbnail. Search filters the available element libraries; it does not change which objects are present in the current scene.

Place

Click the group card, move over the canvas and click to place it. The preview should represent the entire pose, not just one internal prefab. Use exact positions and cell snapping to keep body parts aligned.

Select an existing group

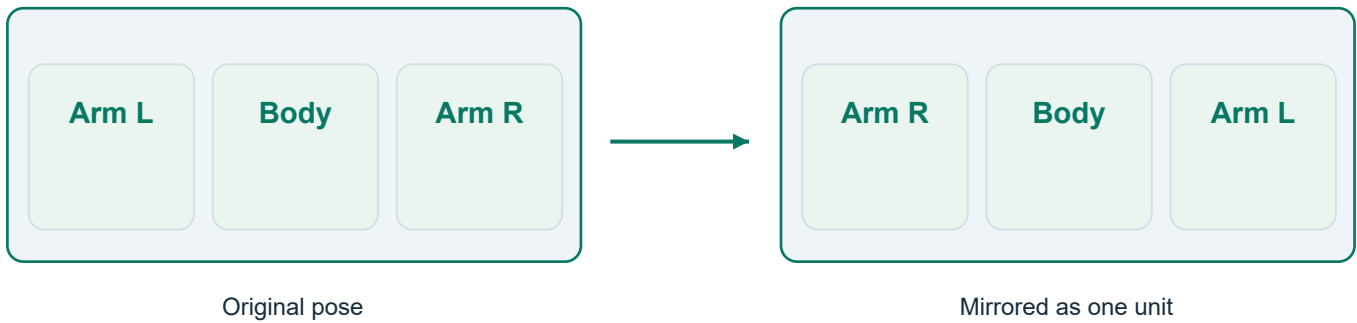
Use the canvas or Elements in Scene. Grouped prefabs appear under their group rather than as unrelated loose entries. Repeated Orc pose placements should still refer to shared component masters.

EXERCISE

Open 06-Orc-Poses.siep.json. Inspect Orc_01_Arms_Up, Orc_01_Dance and Orc_01_Dead. Different poses reuse the same Orc_01 image and arm/leg pieces; they are not a set of separately drawn whole-orc images.

Move the group as a unit

A group should retain its internal spacing during movement, rotation, mirroring and layer-order changes.



Whole-group selection

Click the group with Elements. Shift adds another eligible object. Ctrl deliberately selects an internal component where supported; without that deliberate action, ordinary manipulation targets the group as a unit.

Move and order

Drag the selection preview to the new position. Use Bring to Front/Send to Back for broad ordering and Move Forward/Back for one-step changes. Reordering the group should preserve the order of its internal components.

Verify asymmetry

When checking a mirror or 90-degree rotation, inspect a raised arm or an asymmetric pose. A symmetric standing character can hide an incorrect transform. Compare both the component positions and their individual orientations.

CHECK

After a transform, the raised arm should remain attached on the expected side and the feet should face correctly. A matching outer rectangle alone does not prove that the pose is correct.

Reversible group crop

A group crop limits the visible arrangement while keeping the source group available for other uses.

TEACHING DIAGRAM

A group crop clips the arrangement, not each source master



Select compatible content

Crop controls apply to eligible prefab/group selections. A mixed selection containing arbitrary local lines and fills does not necessarily offer the same crop operation. For several aligned prefabs in Edit Group, use a compatible multi-selection.

Inspect every member

Drag the desired edge and compare the preview with the released result. The group boundary may cut several prefabs differently. Check components at both the near and far edges, not only the largest member.

Separate crop from deletion

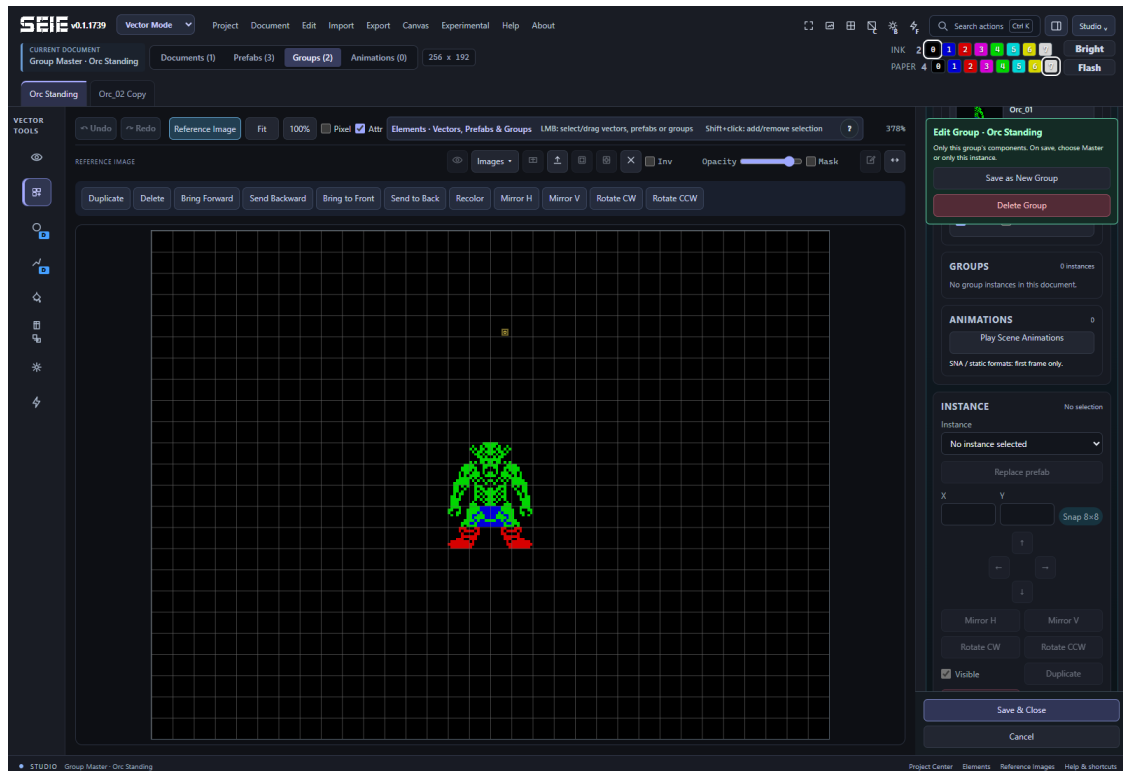
A cropped-away component is not necessarily deleted from the master. If you need a permanently different arrangement, edit an instance locally or save a new group master instead of relying on an unexplained crop.

EXERCISE

Duplicate an Orc pose, crop part of it and then restore the crop. Confirm that the source pose group, shared body image and other placements remain intact.

Edit Group and save scope

Edit Group isolates the arrangement so that you can work on components without accidentally changing the surrounding room.



Actual Edit Group for the mirrored Orc composition: edit the component arrangement, not a new full character image.

Edit the arrangement

Move components, add local geometry or place another prefab from the library. The editing view centres the group for convenience; that editing position is not a command to move every scene placement.

Save to Master

When closing a scene-entered edit, choosing Master applies the arrangement change to linked instances that follow the master. Use this for a correction that should propagate to every ordinary use.

Save only this instance

Choose the instance-only scope for a unique room variation. The placement now has local changes. It should not silently redefine the reusable library arrangement.

BEFORE CONFIRMING

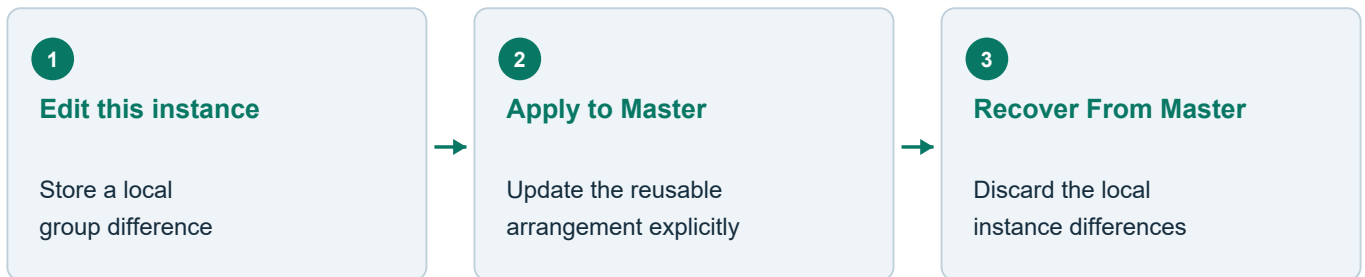
Ask whether another room using the group should change. If the answer is no, do not select Master merely because it is the first option in a dialog.

Apply and recover local changes

Local group edits are useful, but they should remain an intentional exception rather than an invisible second master.

TEACHING DIAGRAM

Local changes and the shared Group Master



Apply Changes to Master

This publishes the instance's arrangement changes to the group master. Other instances that follow the master update accordingly. Review the change in a second scene so that its shared effect is visible.

Recover From Master

This discards the instance's local arrangement changes and restores the master arrangement. It is not an undo of arbitrary unrelated document edits. Save a checkpoint if the local version may be useful later.

Save as New Group

Use a new group master when the local arrangement deserves a reusable identity of its own. Name it by purpose or variant, not by the accidental room in which it was first edited.

DECISION

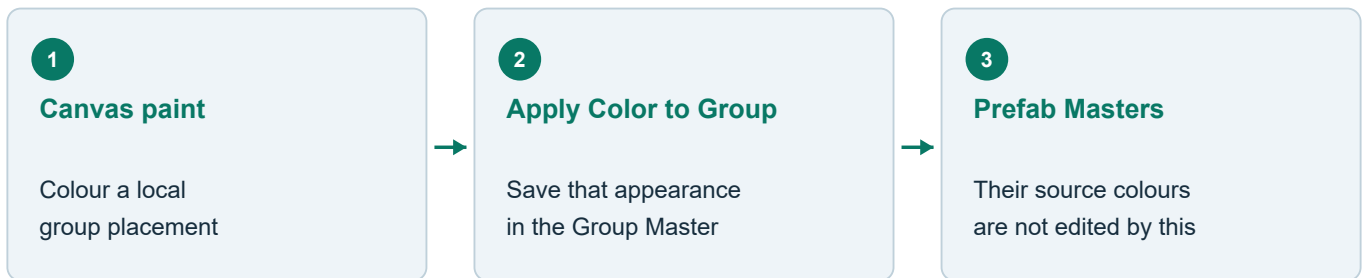
One room-specific Orc pose: local instance edit. A correction to every use of that pose: master update. A reusable new stance made from the same pieces: Save as New Group.

Group colour without master leakage

Group colouring can preserve the independent identities of its component prefabs.

TEACHING DIAGRAM

Group colours have their own ownership scope



Paint the arrangement

Apply the desired local colours in the scene or appropriate group editing context. The visible result can span several prefab components and ordinary geometry. Inspect every cell that should participate.

Apply Color to Group

This records the colour arrangement at the group level. It should not modify the prefab masters used as components. A loose Orc_01 outside the group should therefore retain its source colours.

Recover Color

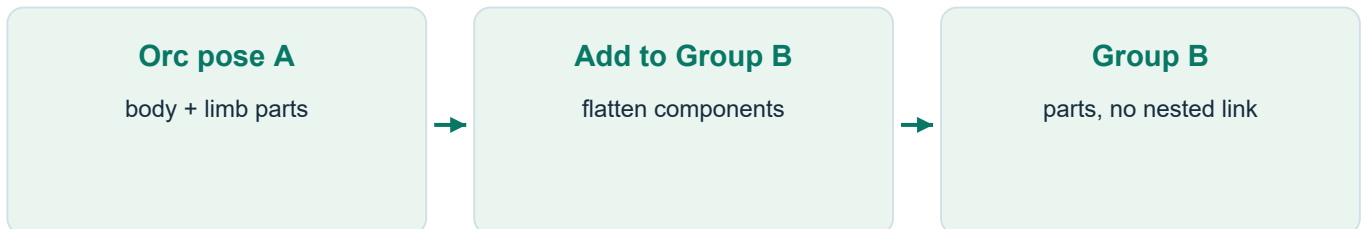
Use recovery to discard the group's local colour changes and return to the applicable group source colours. Arrangement overrides and colour overrides are different concepts; recovering colour is not a general reset of all group edits.

TEST

Place a loose Orc_01 beside a pose group, recolour the group and Apply Color to Group. The loose Orc is the control: its prefab master must not be recoloured by this operation.

Add a group into another group

SEIE flattens the added group's components into the destination group. It does not create an arbitrary nested hierarchy of linked group masters.



Teaching diagram - not an application screenshot.

Add from the library

While editing a group, choose another group for placement. Its members become components of the group being edited. Their prefab master links remain useful, but the source group itself is not a nested live dependency.

Add to Group

The context action can add eligible selected content, including a group, to another group. Choose the destination from the available list or cancel. With a group source, the same flattening rule applies.

Understand later changes

Editing the original source group does not automatically rearrange those flattened members. Editing a shared prefab master used by the members can still affect them. These are different levels of reuse.

CHECK

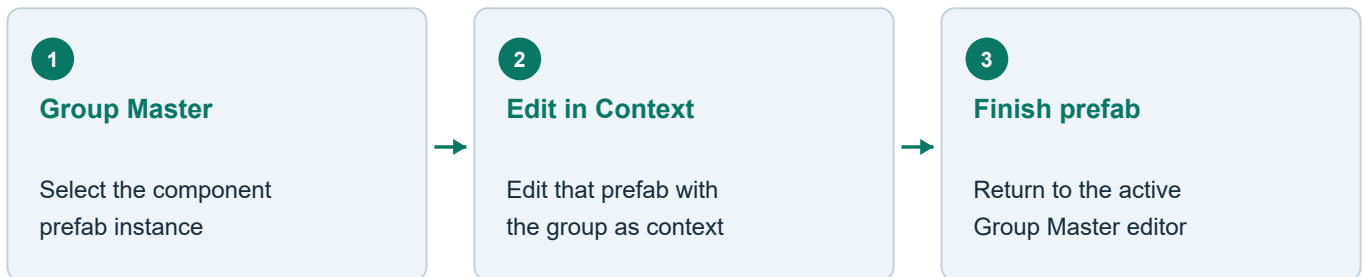
After adding an Orc pose into another group, inspect the destination's component list. Expect the pose's individual parts, not a hidden nested group dependency.

Edit a component in context

Sometimes the arrangement is correct but one prefab's artwork needs attention. Enter the component's prefab editor deliberately.

TEACHING DIAGRAM

Edit a component and return to the group



Enter from Edit Group

Select the prefab component and choose its context-edit action. The group remains the surrounding context, while the active prefab is edited using its own Graphic or Vector tools.

Keep the scope clear

The group master and prefab master are different assets. Moving a component in the group changes arrangement; repainting the prefab master can affect uses outside the group. Check the Current Prefab label before saving artwork changes.

Return to the group

Finish the prefab edit to return to Edit Group. Recheck the arrangement, then save the group at the intended scope. Do not assume that closing the component editor also finishes the group session.

EXERCISE

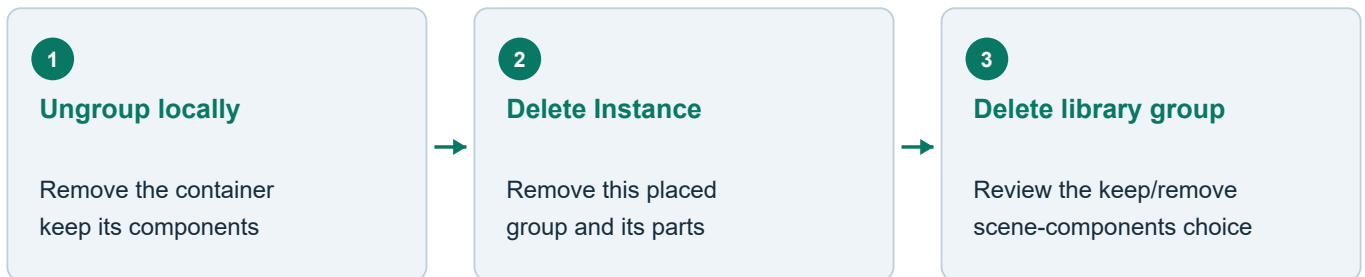
Open a copy of Orc Mirror Pair, enter an Orc_01 component in context, inspect its cells and cancel. Confirm that you return to the group editor rather than an unrelated room.

Ungroup, delete and keep the artwork

Removing organisation and removing visual content are different decisions. Read the deletion scope before accepting.

TEACHING DIAGRAM

Preserve or remove the actual artwork



Ungroup locally

Ungroup dissolves the scene grouping while leaving its component geometry and prefab instances in the canvas. It does not delete the library master or modify other uses of that master.

Delete an instance

Deleting the selected group instance removes that placement's content from the scene. The reusable master remains available for future placement. This is not the same as deleting the group from the library.

Delete a master

Library deletion warns about existing uses. Choose whether placed contents should be removed or ungrouped and retained, or cancel. When uncertain, make a project copy and inspect the usage list first.

CHECK

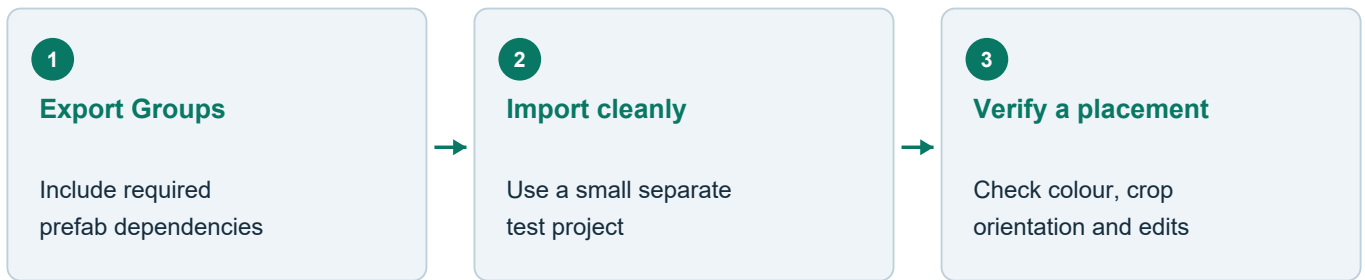
After Ungroup, the former component prefabs should become individually accessible in the scene. If your intention was merely to move one member, consider Edit Group instead of dissolving the whole arrangement.

Transfer and maintain a group library

Group exports must include the prefab dependencies needed to reconstruct their components.

TEACHING DIAGRAM

Transfer a usable arrangement



Export and import

Use Export Groups and Import Groups for reusable arrangements. Open a small test project, import the group and place it once. Inspect colours, crop and orientation before relying on a large merge.

Name for reuse

Use clear names such as Orc Standing, Orc Arms Up or Orc Dance. Unique names prevent ambiguous library entries. Do not confuse an instance's scene-specific name with the identity of its group master.

Watch dependencies

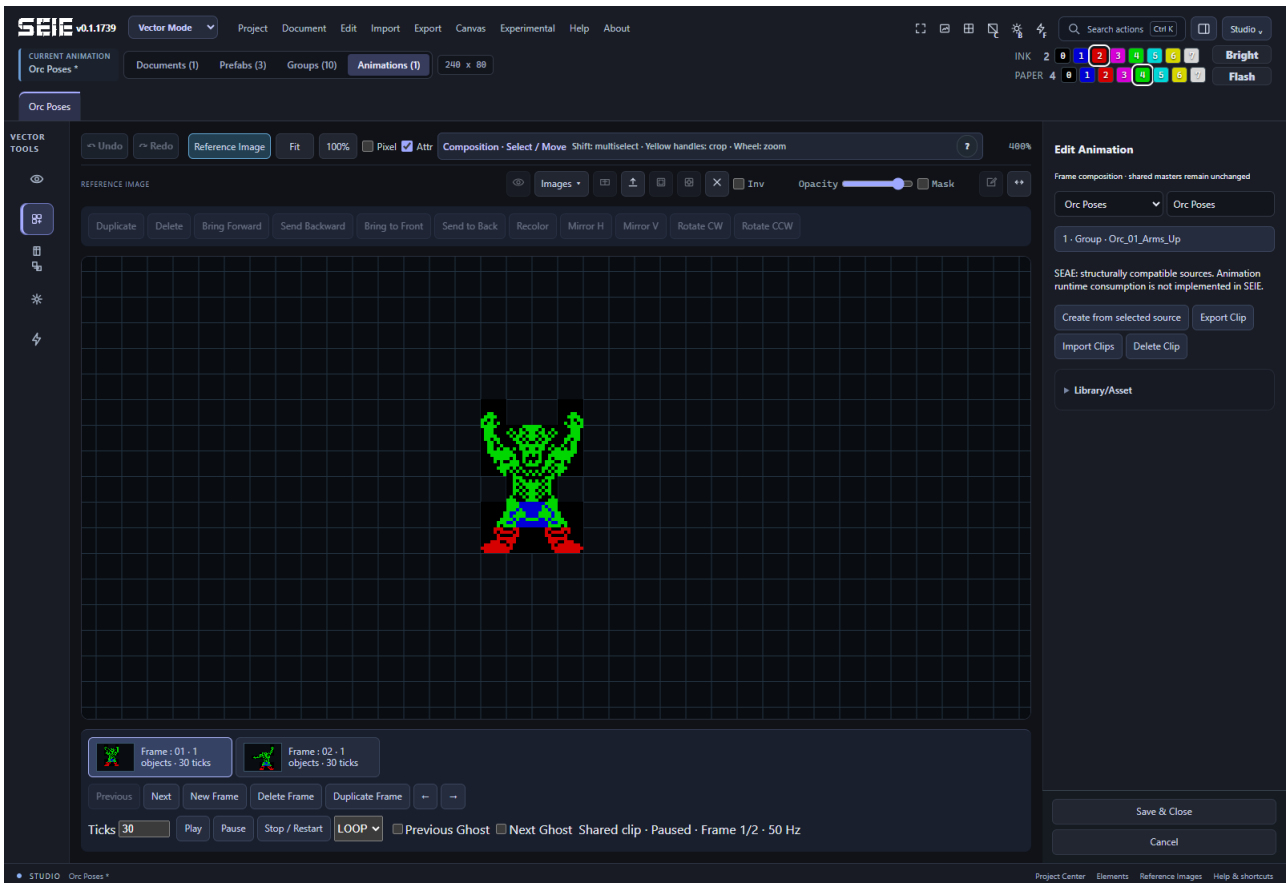
A group using several Orc_01 placements still needs that one source master. Removing or replacing it can affect the pose and animation frames that use it. Project saves preserve the complete relationship graph.

MAINTENANCE ROUTINE

Before sharing a library, save the source project, export the intended groups, import them into a clean project, then verify a placement and one master edit. Keep the clean import test as evidence of a usable package.

Clips reuse frame compositions

An animation clip is a library asset containing timed frames. A frame can arrange existing prefabs and groups rather than storing a newly duplicated master for every pose.



Orc Poses in the actual animation workspace: two frame compositions reuse existing pose groups.

Three levels

The clip defines frames and timing. Each frame contains objects that reference reusable assets. A scene animation instance positions the clip and may hold local frame-object overrides.

Current runtime boundary

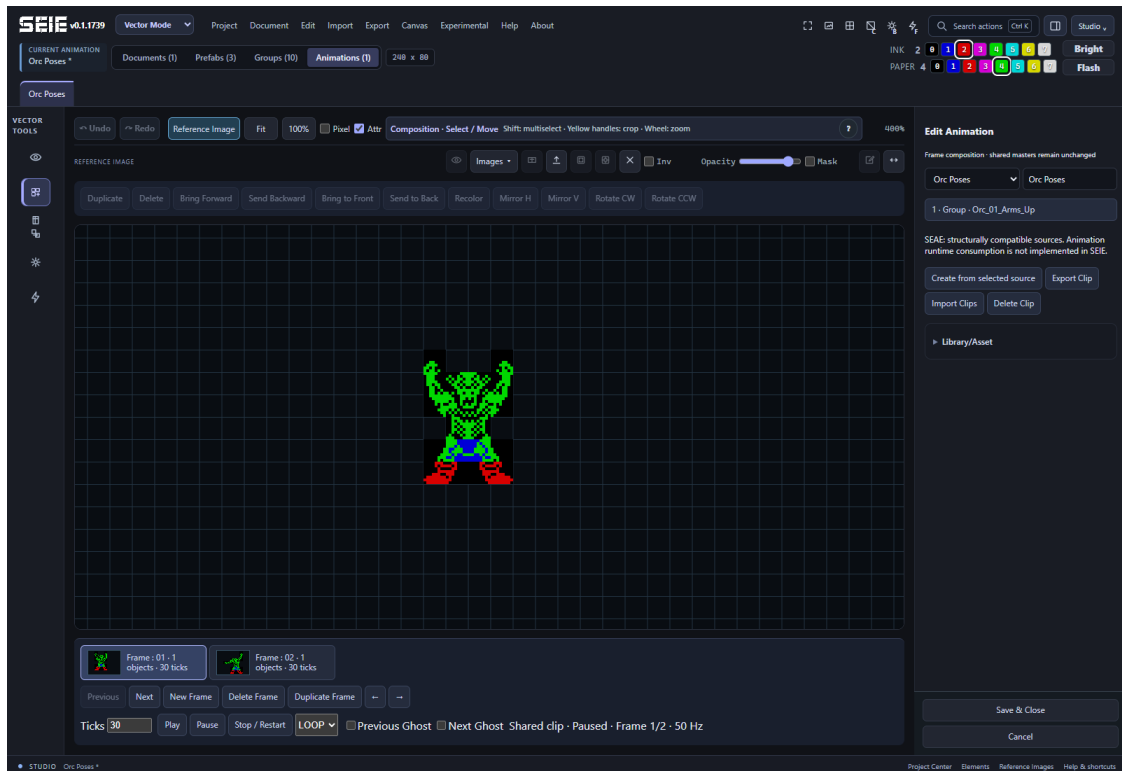
Editor playback demonstrates the clip. It is not proof of Spectrum/SNA animation playback. Formats without animation runtime support use the defined static representation, currently the first frame, rather than duplicating all frames into the scene.

START HERE

Open 07-Orc-Animation.siep.json and choose the Animations workspace. Each of its two frames refers to one existing pose group and lasts 30 ticks.

Create or open a clip

Use the animation workspace for timed compositions, not an ordinary document containing several poses side by side.



Actual clip controls identify the Orc pose group used by the current frame.

Open

Choose an animation under the Animations workspace, or use Document > Animation Clips. The editor shows the active clip, its frames and the objects of the selected frame.

Create from a source

Select a suitable prefab or group source and use Create from selected source. Give the clip a useful name. Begin with one frame so that the initial position and colour are easy to verify before adding variation.

Keep the first frame intentional

The first frame is especially important because it is the safe static representation for current non-animation outputs. Use a recognisable pose rather than a blank transition unless a blank static result is intentional.

CHECK

Verify the clip's name in the animation library and save the project. A frame preview alone is not a separate project file, and a PNG of it does not preserve the clip structure.

Build the current frame

A frame is a composition, not necessarily a single sprite. Add only the objects that belong to that frame.

TEACHING DIAGRAM

A frame is a composition of references



Add assets

Expand Library/Asset while editing the animation, choose a prefab or group and place it on the frame canvas. Existing source assets remain shared; placing them does not turn each frame into an independent prefab library.

Position and select

Use the frame-object list when several objects overlap. Positioning offsets are edited in attribute-cell steps. The stored format retains coordinates consistently; you do not need to edit raw JSON to align a frame.

Order and transforms

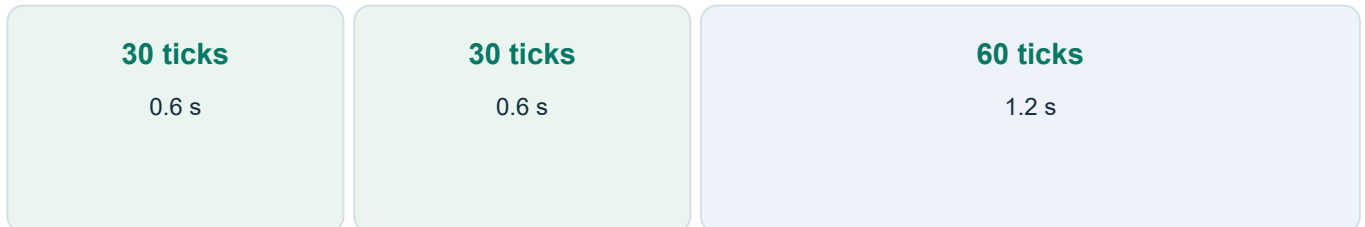
Use the composition toolbar for movement, duplication, ordering, mirror and rotation of eligible objects. Check both the object and the frame's overall footprint. A mirrored object should not accidentally change the source master.

EXERCISE

In a copy of Orc Poses, add a loose Orc_01 to one frame and preview. Then remove that frame object. The Orc_01 master should remain in the prefab library.

Timing in ticks

The animation editor displays its timing at 50 Hz. A tick is therefore 1/50 second in this authoring model.



Teaching diagram: 50 ticks per second; width represents duration.

Duration

The default frame duration is 30 ticks, equivalent to 0.6 seconds at 50 Hz. Ten ticks is 0.2 seconds; fifty ticks is one second. Different frame durations can create pauses without duplicating identical frames repeatedly.

Loop or once

Loop repeats the clip; once is for a finite action. Choose the mode that reflects the intended use, then inspect the end-to-start transition for a loop or the final pose for a one-shot action.

Playback controls

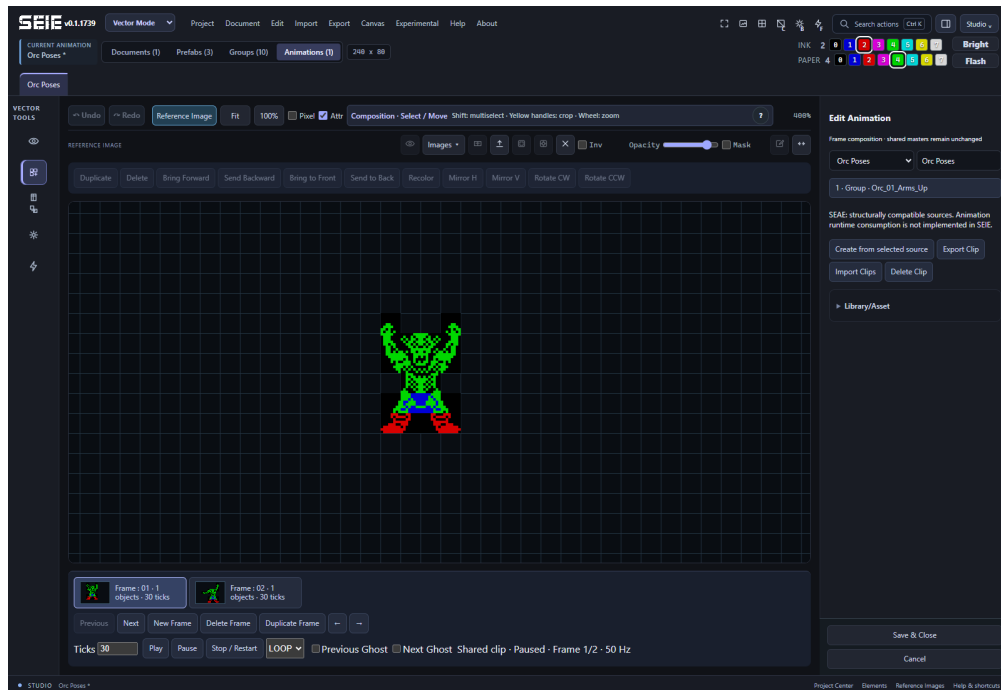
Play starts preview, Pause freezes it and Stop / Restart returns to the start. A browser preview is a design tool; an external game's timing and scheduler must deliberately implement the intended tick behaviour.

EXERCISE

Change one frame from 30 to 60 ticks and play the clip. The frame should remain visible twice as long. Restore 30 ticks before comparing with the supplied checkpoint.

Duplicate and reorder frames

Duplicate a frame when its composition is a useful starting point. Change only what needs to differ in the next pose.



Actual Orc frame thumbnails, timing controls and the paused playback state.

Duplicate Frame

Select the source frame and choose Duplicate Frame. The new frame has its own editable composition and duration while still referring to the same reusable assets. It does not require new copies of every prefab master.

Reorder

Use the frame move controls to change sequence order. Check the thumbnails and frame numbers afterwards. Moving an object within a frame is different from moving the frame in the timeline.

Remove deliberately

Delete Frame removes that step in the clip, not the source prefabs. Keep a meaningful first frame after deletion because static output uses it. A blank frame can be useful as a deliberate pause but should not become the initial output by accident.

EXERCISE

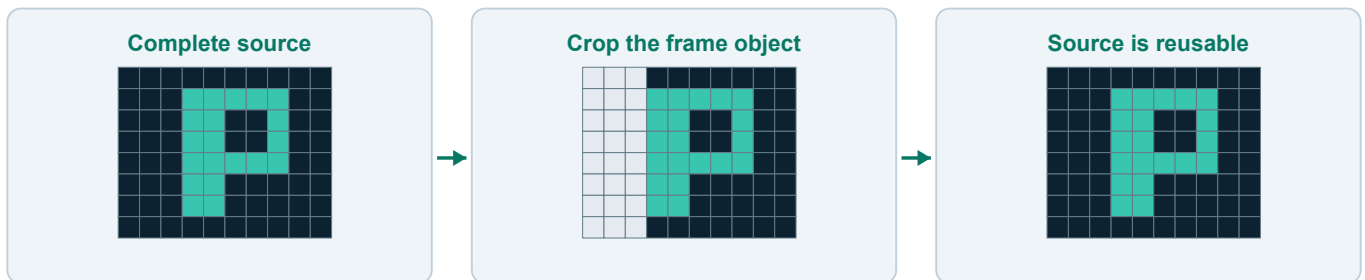
Duplicate the first Orc Poses frame, move it between the existing frames and give it a longer duration. Play the sequence, then Undo to compare the original rhythm.

Crop and transform frame objects

Frame-object transforms are composition settings. They should not rewrite the prefab or group master being reused.

TEACHING DIAGRAM

Clip the frame object without editing the source



Crop on the canvas

Select the object and drag its yellow crop handles. The retained source area should stay anchored while an edge is trimmed. Check the result after release as well as during the preview.

Mirror and rotate

Use the composition toolbar for H/V mirror and clockwise/counter-clockwise rotation. An asymmetric detail helps identify whether the correct source region remains visible after crop and orientation combine.

Maintain a common envelope

Frames can have different visible extents. Position them relative to a stable origin so a changing silhouette does not appear to jump. A scene instance uses the clip's overall coordinate context, not a separately re-centred origin for every pose.

CHECK

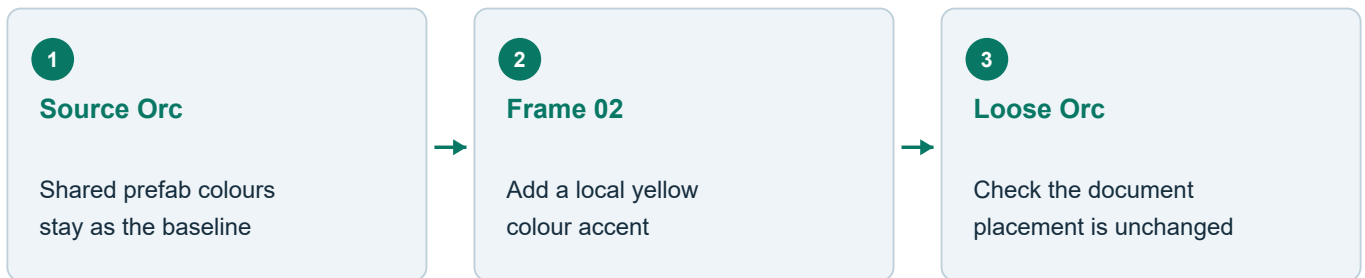
Play the clip with a fixed landmark such as the feet or a door hinge. If that landmark jumps unexpectedly, compare frame offsets and crop before moving the shared source artwork.

Paint a frame locally

An animation frame can recolour its source objects without changing the library masters they refer to.

TEACHING DIAGRAM

One yellow accent, three scopes to compare



Select the frame and object

Use the frame thumbnails and object list to make the target explicit. Set INK, PAPER, BRIGHT and FLASH as needed, then paint the intended owned cells with Color.

Compare another frame

Switch to a frame using the same source asset. The deliberate local recolouring should remain attached to the edited frame-object context rather than leaking into every use of the prefab master.

Keep geometry shared

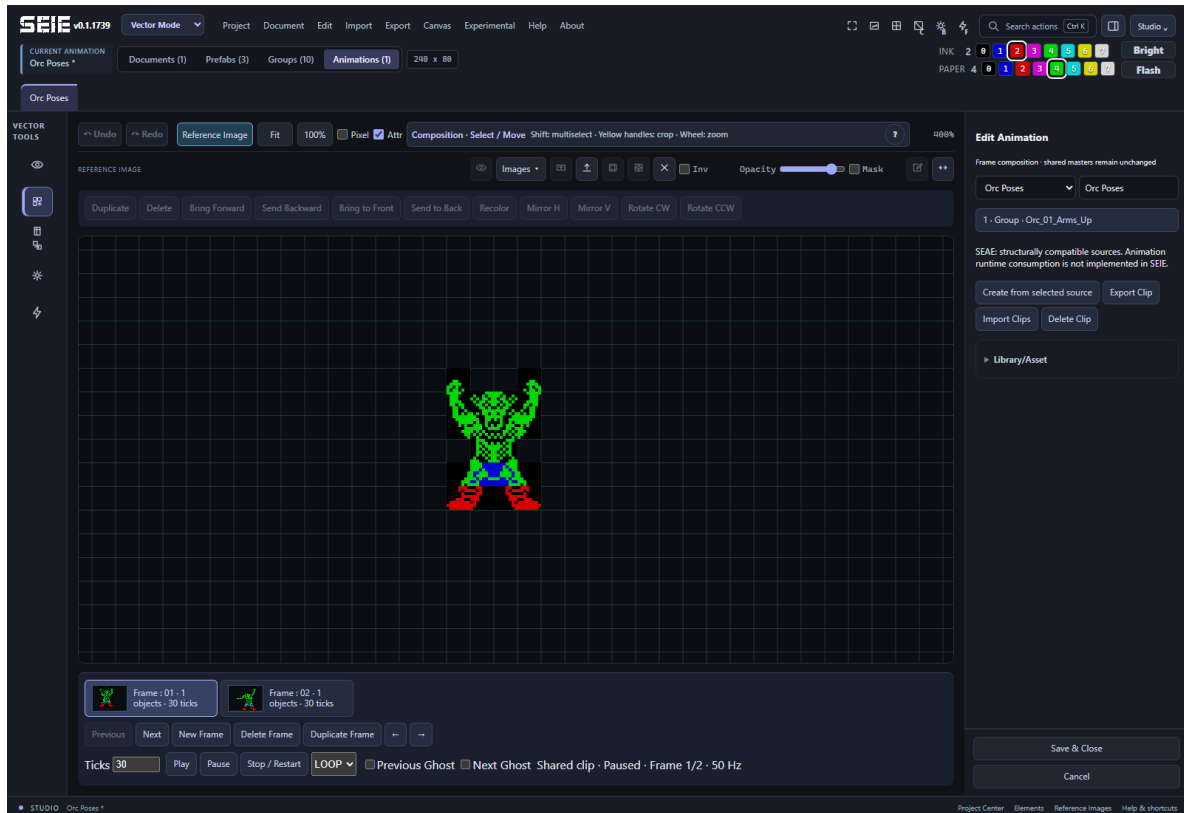
Use local colour for blinking lights, damage tints or palette changes that do not need different source geometry. If the shape genuinely changes, use another source asset, variant or composition rather than obscuring the old shape with unexplained colour patches.

EXERCISE

Give the Orc pose a yellow accent in the second frame only. Inspect both frames and a loose Orc_01 in a document. Compare frame-local paint with the shared group and prefab colours.

Ghosts and playback checks

Preview tools help compare neighbouring poses, but they are not part of the saved output artwork.



Orc Poses after playing and pausing. The frame controls expose the current playback state.

Previous and next ghosts

Use Previous Ghost and Next Ghost to compare adjacent poses. They reveal alignment and motion arcs. Turn them off when judging final colour or checking a static export.

Pause at meaningful moments

Inspect the first frame, the largest extent and the loop transition. A clip can look acceptable in continuous motion while containing one misaligned or incorrectly cropped frame.

Check timing and scope separately

First verify geometry and colours while paused. Then verify rhythm with Play. A correct-looking paused frame does not prove timing, and smooth timing does not prove that local paint stays isolated.

REVIEW ROUTINE

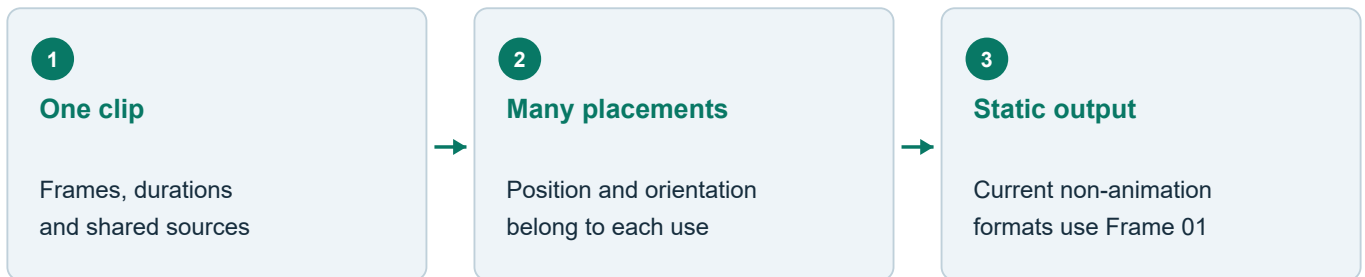
Look at normal playback, inspect each frame individually, then disable ghosts and return to Frame 01. Save only after all three views make sense.

Place a clip in a scene

An animation instance links to a clip and provides scene placement. The editable clip remains in the animation library.

TEACHING DIAGRAM

Clip definition and scene placement are distinct



Place from Library/Asset

Return to a document and choose an animation asset for placement. Position it using Elements. The scene list identifies animation instances separately from loose prefabs and groups.

View and block

Select View > View Animations controls their editor visibility. The matching block-selection option prevents accidental manipulation. Scene-level animation preview can be enabled for inspection where offered.

Static versus moving output

The editor can preview motion while a static export or current SNA output displays Frame 01. This is an intentional format boundary, not evidence that every animation frame should be rasterized into the scene.

CHECK

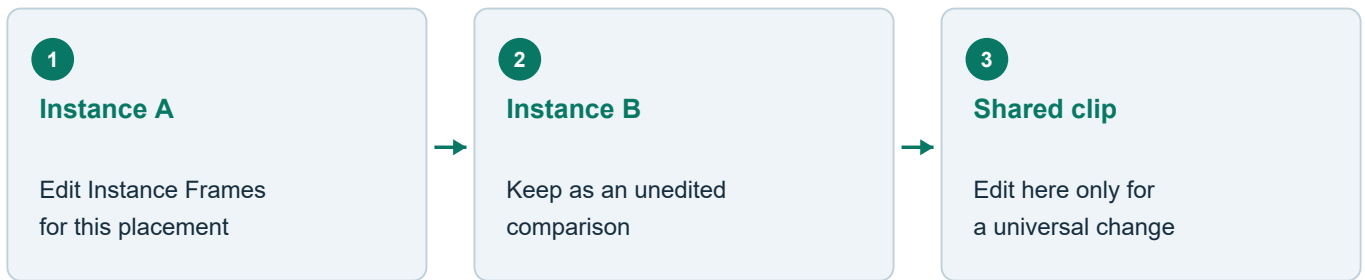
Save and reopen the project. Confirm the clip still exists and the scene placement refers to it. Then inspect the initial frame so a non-animated export remains useful.

Instance frame overrides

Use local overrides when one placed animation should differ from other uses of the same clip.

TEACHING DIAGRAM

Compare two uses before saving an instance override



Enter the right editor

Use the scene instance's Edit Instance Frames action where available. The editor identifies instance overrides rather than shared clip editing. This distinction determines who receives the changes.

Change the placement's frame objects

Local overrides can adjust compatible object settings such as colour, crop and position. They refer to existing frame/object identities; they are not an invitation to redesign the shared clip's complete frame sequence locally.

Compare a second instance

Place two copies of the same clip and edit only one. The unedited copy is a useful control case. If both change, check whether you entered shared clip editing instead of the instance override context.

SAFETY RULE

For a universally improved animation, edit the shared clip. For a special appearance of one placed actor, use its supported local overrides and keep the shared source unchanged.

Save, import and export clips

Animation data depends on its source assets. Transfer the composition and its required resources together.

TEACHING DIAGRAM

Transfer the clip together with its dependencies



Save the project

The project preserves clips, frames, source references and scene instances. Navigating out of the animation workspace can save the active editing state automatically, but a downloaded project remains your durable working copy.

Transfer a clip

Use Export Animations or Export Clip and the matching import route. Required prefab/group dependencies must remain resolvable. Test the imported clip in a clean project and inspect every frame before merging it into production work.

Avoid unnecessary duplication

Two frames using the same source should still refer to that source. Reusable assets, frame composition and playback time are separate data. Do not export a PNG per frame as your only editable backup.

CHECK

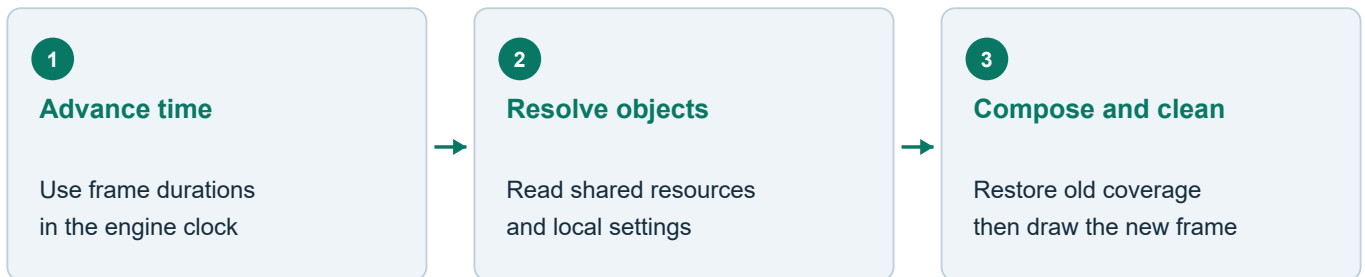
After import, count the clip's frames, inspect their durations and verify a placement. A thumbnail alone does not prove that every source dependency survived the transfer.

What an external game must provide

SEIE authors and previews animation data. A game's runtime must deliberately consume that data and implement its own playback policy.

TEACHING DIAGRAM

Playback responsibilities in an external game



Resource selection

The game loads or links the exported resources, resolves frame objects and draws them in the intended order. It must respect masks and attributes, not just copy pixel bytes indiscriminately.

Timing and cleanup

The game advances frames according to duration, handles loop/once behaviour and restores the background when the visible footprint changes. A changing frame must not leave old pixels behind simply because the next frame is smaller.

Limits to communicate

Current SNA/static output uses the first-frame representation. The animation editor's compatibility diagnostics can flag source compositions that need additional support. Do not describe a structurally compatible clip as an already implemented Spectrum playback engine.

NEXT STEP

Use the Resource Pack chapter to integrate one static component first. Once drawing, masking and background restoration are reliable, add frame scheduling in the game's own update loop.

Two meanings of reference

A reference image is a drawing guide. A linked document reference is reusable scene content. Confusing them leads to incorrect expectations about saving and exporting.

Guide image

helps you draw

Linked document

shared scene source

Image import

creates artwork

Teaching diagram - not an application screenshot.

Reference image

An imported image helps you trace, align or compare artwork. It belongs to an editing owner such as a document, prefab, group or animation frame. It is not exported as scene geometry merely because it is visible behind the canvas.

Linked document

Use Reference links another project document as a scene source. Changes to that source can affect dependants. Local additions and overrides belong to the referencing document.

Choose the right route

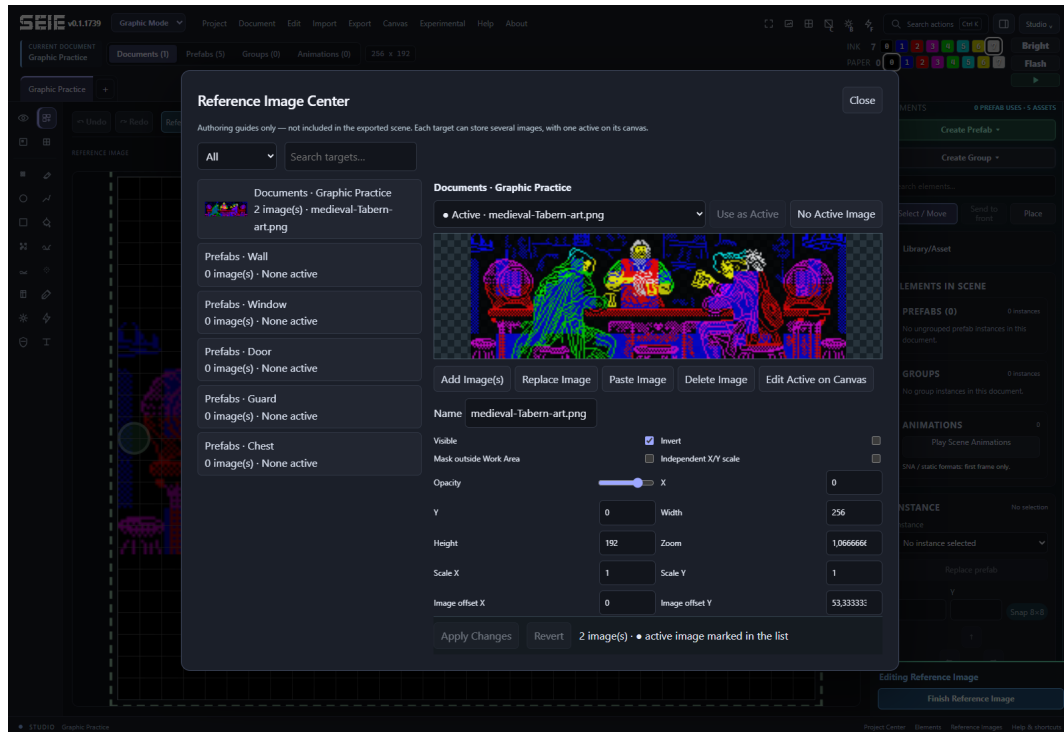
To trace a sketch, import a reference image. To share a room background between several scenes, create a linked document reference. To convert a photo into Spectrum pixels, use Image Import rather than assuming a guide has become artwork.

CHECK

Hide the guide image and inspect the actual scene. If the picture disappears entirely, you may have only a reference image, not converted or drawn graphics.

Manage reference images centrally

Project > Reference Image Center collects the project's drawing guides by owner.



Reference Image Center lists documents, prefabs and groups, with controls for the active guide.

Choose an owner

Select the document, prefab, group or animation frame that should use the image. An owner may keep several images, with one active image selected for its canvas.

Add or replace

Add Images imports another guide; Replace Image changes the selected one. The normal Import or Reload Reference Image route also distinguishes replacing from importing a new image. Use descriptive names when keeping alternatives.

Activate and inspect

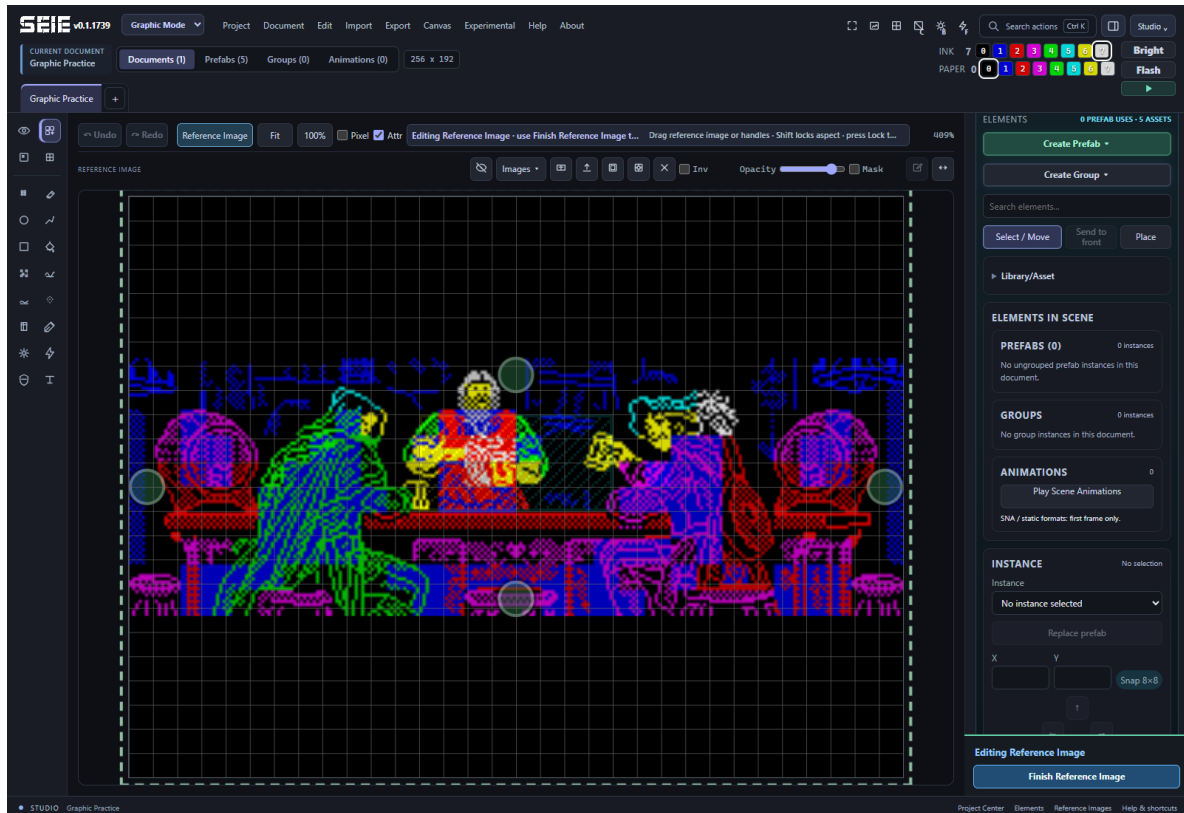
Select an image and use the active-image controls. The centre offers preview, visibility and transform settings. Deleting a guide should not delete drawn scene artwork, but keep the original source file if you may need it again.

EXERCISE

Keep a clean sketch and a colour study as separate guides for the same owner. Switch the active guide instead of repeatedly replacing one file and losing the alternative.

Edit, move and crop the guide

Reference-image editing is an explicit mode. Finish it before returning to ordinary canvas drawing.



An actual imported guide in the canvas. Image editing is separate from artwork editing.

Enter editing

Use the Edit Reference Image icon beside Mask. Entering this mode shows the image when needed and exposes its manipulation controls. An invisible guide should not move or resize accidentally.

Position and scale

Drag to position and use the handles for resize/crop. Fine Resize Sensitivity in the Canvas menu adjusts precision. Use the finer wheel adjustment when matching a small feature to the attribute grid.

Finish

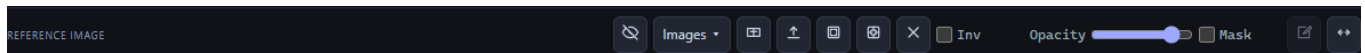
Use Finish Reference Image to leave editing and keep the image static. The collapsible Reference Image section can be hidden to recover canvas space without deleting the guide.

CHECK

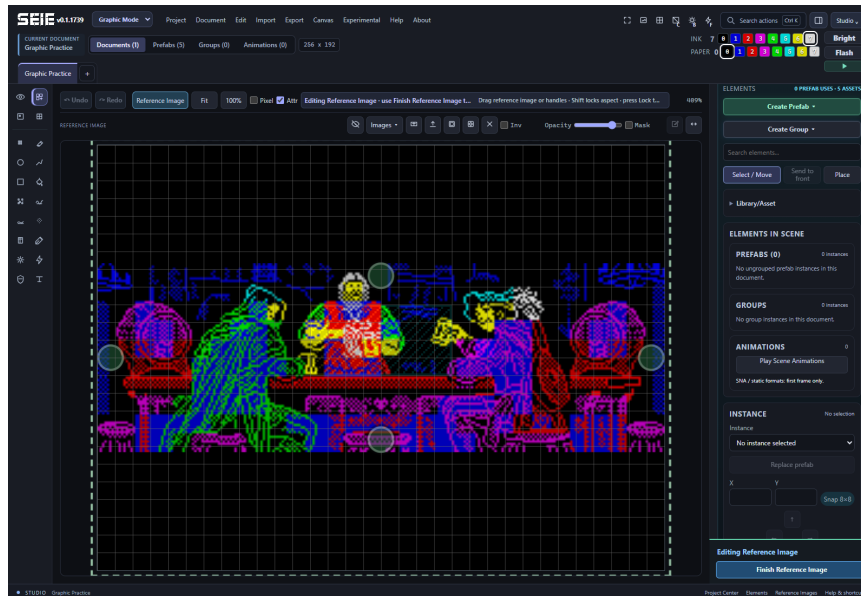
After finishing, try an ordinary drawing gesture on a safe test area. The guide should stay put. If it moves instead, verify that you actually left reference-image editing.

Mask and opacity are guide settings

Reference-image visibility settings help you draw; they are not a replacement for the scene's own clipping and transparency rules.



The real reference-image strip: visibility, active image, opacity, Mask and Edit.



The guide visible in the workspace. Mask and Opacity affect this reference image, not the exported artwork.

Mask outside Work Area

Enable Mask to hide the guide outside the active Work Area. This helps compare the usable picture region without obscuring surrounding editor space.

Opacity and inversion

Lower opacity when the guide hides your pixels. Inversion can improve visibility against dark artwork. Pixel and attribute grid lines remain useful alignment guides above the reference image.

Independent guide collections

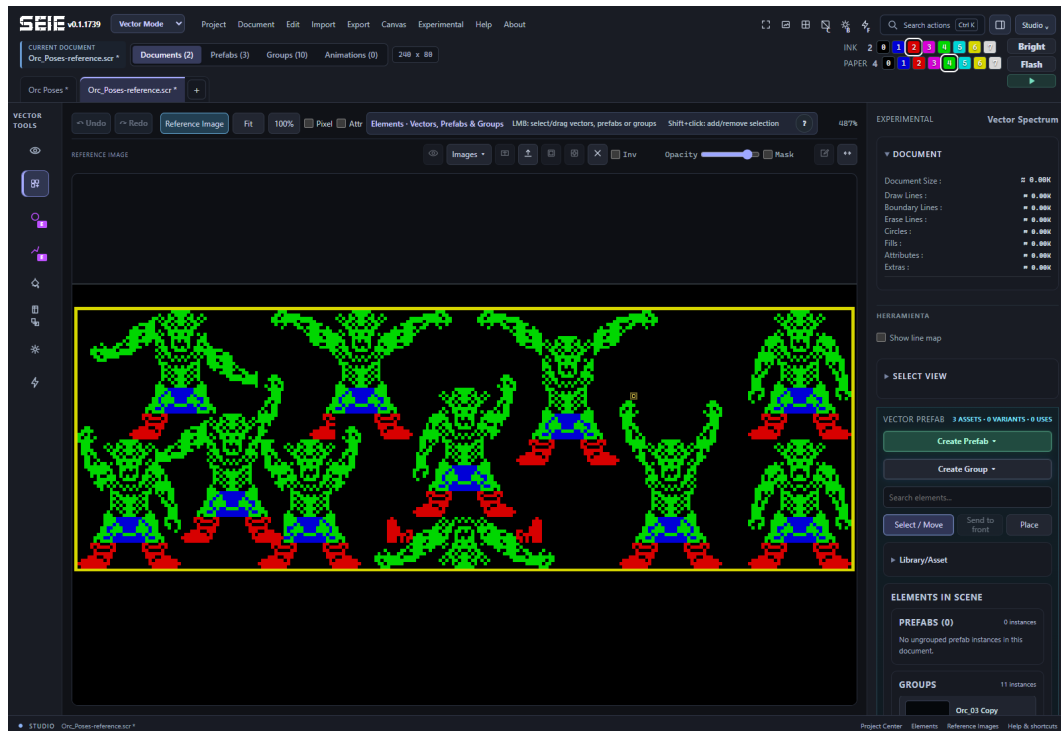
Different owners can keep different images. A prefab master guide need not be the same as its room's background sketch. An animation frame can use a pose-specific guide while other frames use their own references.

RULE

Export a normal PNG or SNA to validate artwork, not a screenshot containing a guide overlay. Screenshots are useful for documenting the editing process but can include non-exported information.

Create a linked room

Use a reference when several scenes should share a live source rather than independent duplicated artwork.



The Orc pose sheet's reference document, created through Duplicate as Reference in SEIE.

Duplicate as Reference

Open the source document and choose Document > Duplicate as Reference. Inspect Use Reference in the new document to confirm which source is linked. Save the project so both documents and their relationship remain together.

Add local content

Place a prop or add local geometry in the referencing scene. The shared background should remain reusable while the addition belongs to this room. Use meaningful names to distinguish source and dependants.

Verify the link

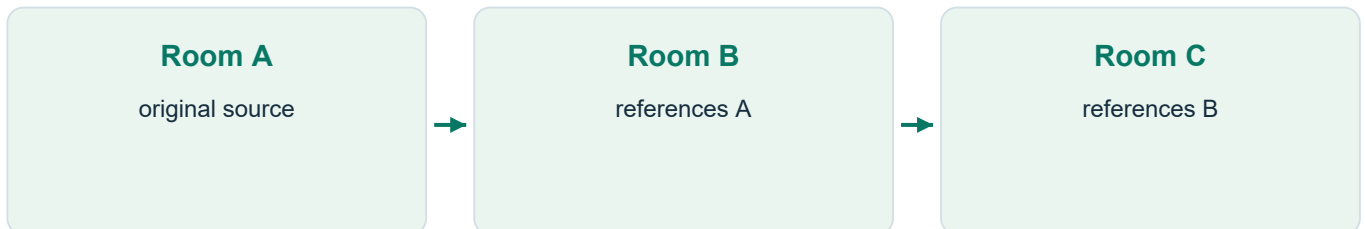
Change a small source detail on a saved copy, return to the dependant and inspect the update. Then Undo or restore the checkpoint. This demonstrates a real link rather than two similar-looking independent scenes.

STARTING POINT

08-Orc-Linked.siep.json contains Orc Poses and its linked reference, created and saved through the actual interface.

Direct and indirect dependants

A document can be referenced directly and can also influence documents farther down a reference chain.



Teaching diagram - not an application screenshot.

Direct

If Room B references Room A, B is a direct dependant of A. A source edit can therefore affect B even when B has its own local additions.

Indirect

If Room C references B, C is an indirect dependant of A. Its final appearance combines the chain and each document's relevant local context. Inspect more than the immediate child after a shared change.

Use the dependency list

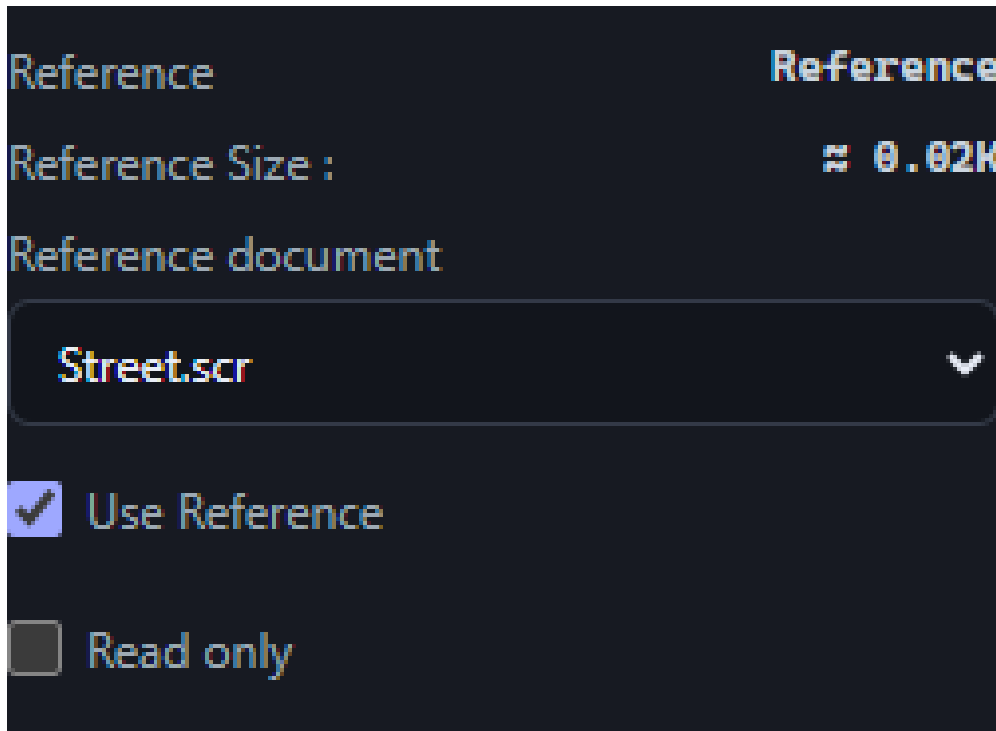
Is Referenced lists direct and indirect uses and can be collapsed to save inspector space. Navigate to representative dependants before a destructive source edit or mode conversion.

KEEP CHAINS UNDERSTANDABLE

Prefer a small number of clearly named source layers over a long chain created by repeated duplication without a plan. A deeper chain is harder to diagnose when a crop, mirror or colour patch changes its effective context.

Read Only means protected inheritance

Read Only protects the referenced background from direct selection and editing while allowing the current room to receive local work.



Actual linked-document settings, including the reference source and Read Only control.

Local drawing

Treat the room as a local overlay above the inherited background. New local geometry replaces the inherited graphics in the attribute cells it occupies, rather than indiscriminately mixing old and new pixels in those cells.

Local colour

You can change local colours in cells even when they do not yet contain local geometry. The background geometry remains until local geometry takes ownership of the relevant area according to the composition rules.

Selection

Blocked inherited content should not be grabbed while selecting local props or commands. This is useful for building room variations without accidentally moving the source backdrop.

CHECK

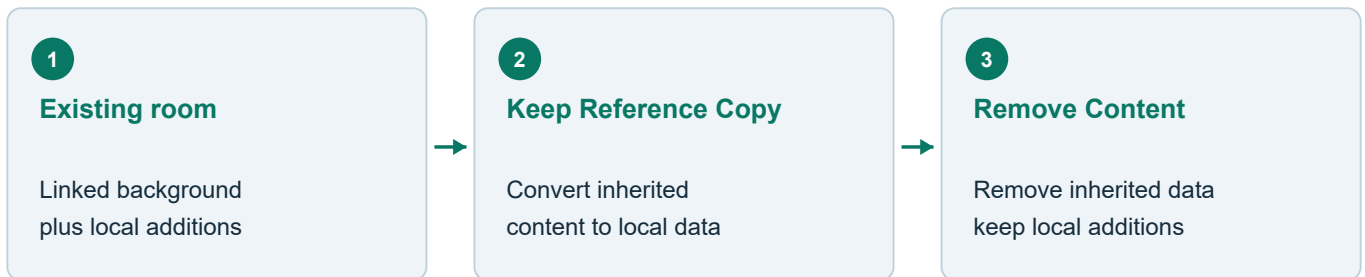
Add a small local object over a reference on a project copy. Inspect its occupied cells, then remove it and compare the background. Verify the source document itself was not edited.

Detach while preserving local work

Disconnecting a reference asks what should happen to inherited content. Local additions should not be discarded merely because the link ends.

TEACHING DIAGRAM

Detach without deleting local work



Keep Reference Copy

Choose this when the scene should become independent while retaining the visible inherited content. The result is no longer a live link to future source edits. Inspect colour-cell alignment and retained elements afterwards.

Remove Reference Content

Choose this when the inherited background should disappear and only the local scene work should remain. This is not the same as clearing the entire document.

Compare the result

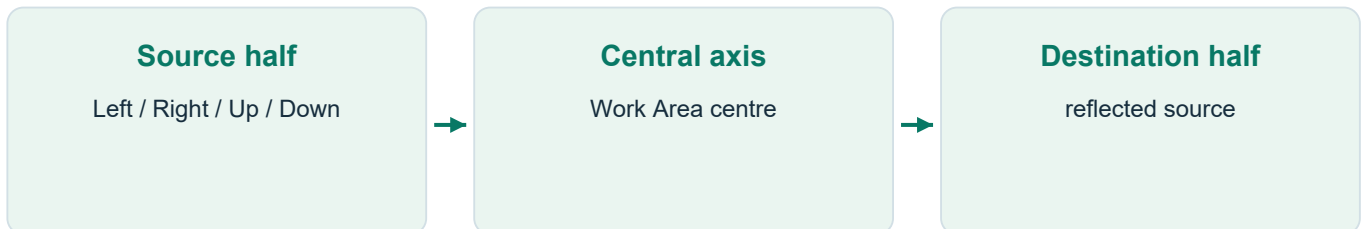
Check Use Reference, the dependency list and the actual canvas. Save and reopen a copy to make sure the intended independent state persists rather than merely appearing detached in the current view.

DECISION

Use Keep Copy before intentionally diverging a room from its source. Use Remove Content when replacing the background. Save a checkpoint before either choice if the original relationship may be needed again.

Central scene mirrors

A central mirror copies a chosen half of the scene into its opposite half. It is different from flipping the complete scene.



Teaching diagram - not an application screenshot.

Choose axes and source

Enable Mirror H and choose Left or Right, and/or enable Mirror V with Up or Down. The axis is centred on the active Work Area, or the full canvas when no Work Area is enabled.

Mirror Color

When enabled for a central mirror, attributes follow the source-side reflection. When disabled, geometry and colour treatment can differ by design. View Source tints the non-source region as an editor guide.

Edit with the source in mind

Place geometry and seeds deliberately. A visible reflected region can originate from data outside the apparent source half after clipping and fill-context processing. Check the complete rendered result rather than judging solely by a seed marker.

CHECK

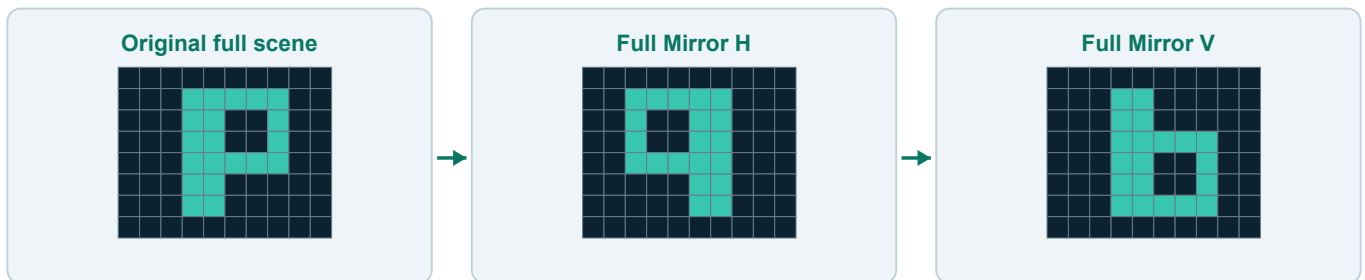
Use an asymmetric test motif and distinct colours in the source. Compare editor and target output before applying a complex H+V mirror to an important linked room.

Full mirror is a different operation

Full Mirror reverses the whole source instead of copying one side into the other. Its semantics depend on the authored representation.

TEACHING DIAGRAM

Full mirror transforms the complete picture



Graphic

A Graphic scene has raster pixels and attributes to transform. Inspect both the pixel pattern and cell colours after the full flip, especially when the Work Area is offset from the screen origin.

Vector

A full Vector mirror transforms the existing vector construction. Discontinuous stroke direction and fill/boundary interactions can produce differences from a simple pixel-for-pixel flip of a rendered bitmap. Do not use strict bitmap equality as the only test of a semantic vector transform.

Local versus inherited mirror

A mirror on a referencing document is not automatically a modification of its source. Keep the source's settings and the local mirror settings distinct when diagnosing an H/V combination.

TEST

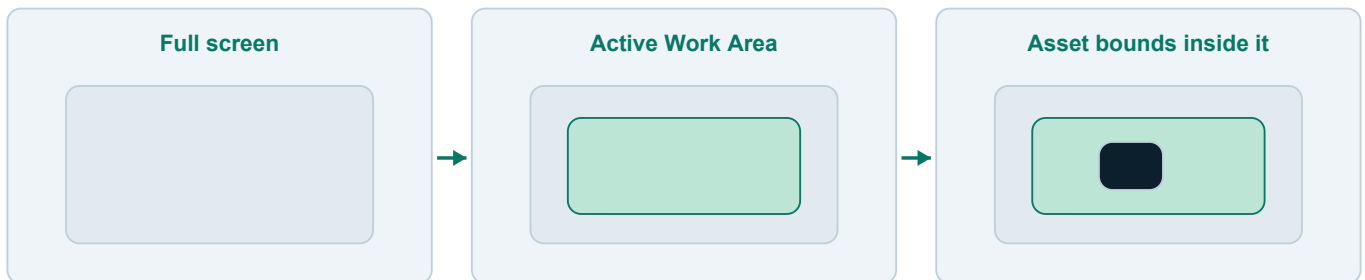
Compare a distinctive left-side detail, its seed region and its colour cells. Checking only the overall silhouette can miss an attribute transform or source-side error.

Work Area, chrome and inherited layout

Work Area limits describe the scene's active picture. Titles, descriptions and external frames can occupy a different part of the output layout.

TEACHING DIAGRAM

Three bounds with different jobs



Inherited context

A linked document inherits its source Work Area context. Local mirrors and additions must be interpreted in that coordinate system. An offset Work Area is not equivalent to a full-screen origin at zero.

Scene chrome

Use Custom Values and related controls for external frame and scene text settings. Keep those distinct from artwork inside the Work Area. The export strategy must preserve the intended outside-area presentation as well as the main picture.

Compare complete output

When validating, inspect the whole Spectrum screen, not only a crop of the Work Area. A correct room with missing title text or an incorrect external frame is still an incomplete export.

EXERCISE

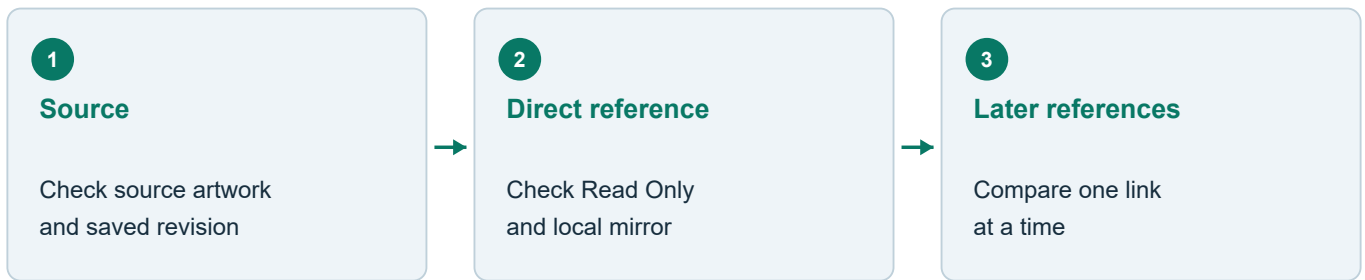
On a copy, enable a Work Area and a visible external frame. Add a short scene title, export and compare both the room and its surrounding presentation.

Troubleshoot a reference chain

Reduce the problem to a source, one dependant and one local change before testing a long chain.

TEACHING DIAGRAM

Diagnose the chain from the original outwards



Establish the first difference

Open the source and verify it independently. Then inspect the direct reference with local overlays hidden where possible. Continue down the chain until the first incorrect document appears.

Record the context

Note Work Area coordinates, H/V source choices, Full Mirror, Mirror Color, Read Only, local geometry and local colour patches. Include crop/orientation of any problematic prefab rather than recording only its asset name.

Preserve evidence

Save the small reproduction and the exact exported file. Record the SEIE version and export strategy. A bitmap-compressed result and a semantic Vector result may use different preparation paths, so compare like with like.

AVOID COMPOUNDING THE ISSUE

Do not add duplicate fills or shift masters to compensate for an unexplained reference error. Those changes can hide the symptom in one scene while damaging independent uses elsewhere.

One orc, many poses

The Orc page is not a collection of independently drawn character images. One base orc and its reusable pieces are assembled in different positions and stored as groups.



Actual Orc pose sheet from Medieval, with the unrelated shield study omitted in the teaching checkpoint.

Start with one image

Orc_01 is a single 32 x 40 Graphic source. Its placements can be mirrored, cropped and positioned without making a new full-orc master for every visible character.

Reuse the parts

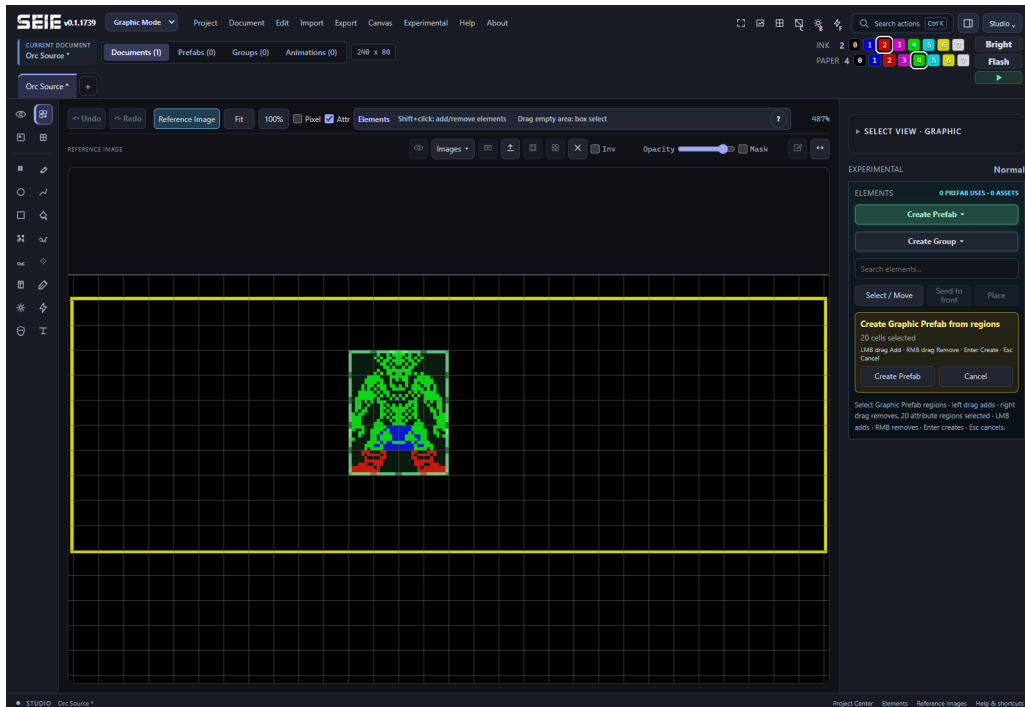
The real pose sheet also uses shared Orc_01_Arm and Orc_01_Leg pieces. Arms up, dancing and fallen poses combine these with cropped body placements. A group records each useful arrangement.

FILES

04 is the ordinary source image; 04a the captured prefab; 05 the mirrored pair; 06 the poses; 07 the clip; 08 the linked sheet. Save your own working copies.

Capture the source as a prefab

Begin with ordinary canvas pixels and turn the orc into one reusable asset.



Real Graphic region capture: four columns and five rows, enclosing the source orc.

Open the source

Open 04-Orc-Source.siep.json. The orc is ordinary Graphic canvas content, not a placed prefab. There are no prefab masters in this checkpoint.

Select the footprint

Choose Create Prefab > From graphic regions. Select screen X 96 to 127 and Y 24 to 63: a 32 x 40 image, or 4 x 5 attribute cells. These are screen coordinates, not Work Area-local coordinates.

Save and inspect

Create Orc_01, inspect its regions and check that captured pixels have not remained as an accidental duplicate below the placement. Open Edit Master to inspect the owned cells.

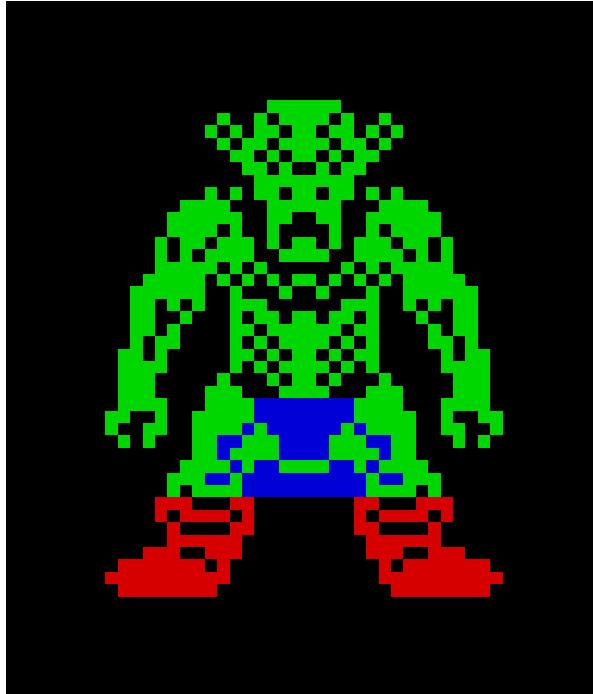
The basic capture owns all 20 cells. To match the pose kit's transparent silhouette, deselect the two empty top corner cells with Attribute Regions; black pixels alone do not make these cells transparent.

CHECKPOINT

04a-Orc-Created.siep.json is the result created and saved through the actual interface. Keep the source checkpoint for repeating the exercise.

Reuse the same image with Mirror H

A mirrored placement changes orientation, not the identity of the source artwork.



The standing Orc composition uses two cropped placements of the same Orc_01 source, one mirrored horizontally.

Duplicate a placement, not the master

Choose Orc_01 from the library to place another instance. Use Mirror H on that instance. Do not choose Duplicate Master Prefab unless you deliberately want a new independent source.

Inspect the real paired example

Open 05-Orc-Mirror-Pair.siep.json. Both components reference Orc_01. Their crops retain the useful half and its reflection; together they produce the standing character.

Keep the visual seam clean

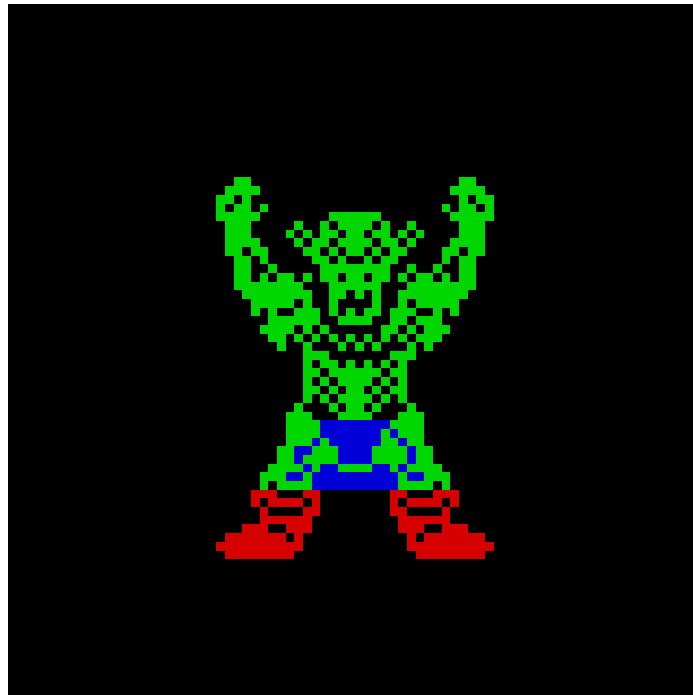
Inspect the join at native scale and with the attribute grid. Position, crop and ownership must agree. Do not redraw a full second orc to fix a misplaced instance.

MEMORY PRINCIPLE

Sharing one source avoids storing another complete drawing for the mirrored use when the target preserves shared assets. Position, crop and orientation still need data; Bitmap Compressed scene output can store the final composed image instead.

Arrange shared parts into a pose

The same source character can provide the body while shared arm and leg pieces make the pose expressive.



A real pose Group Master from Medieval, rendered by SEIE rather than redrawn for this manual.

Choose the body

Use an Orc_01 placement and crop it to the intended body region. Keep its source link. Do not include unwanted limb pixels beneath the new limb placement.

Place the limbs

Add Orc_01_Arm and Orc_01_Leg from the library. Move, mirror and rotate these placements to form the stance. Use attribute-grid alignment where colour ownership requires it.

Inspect overlaps

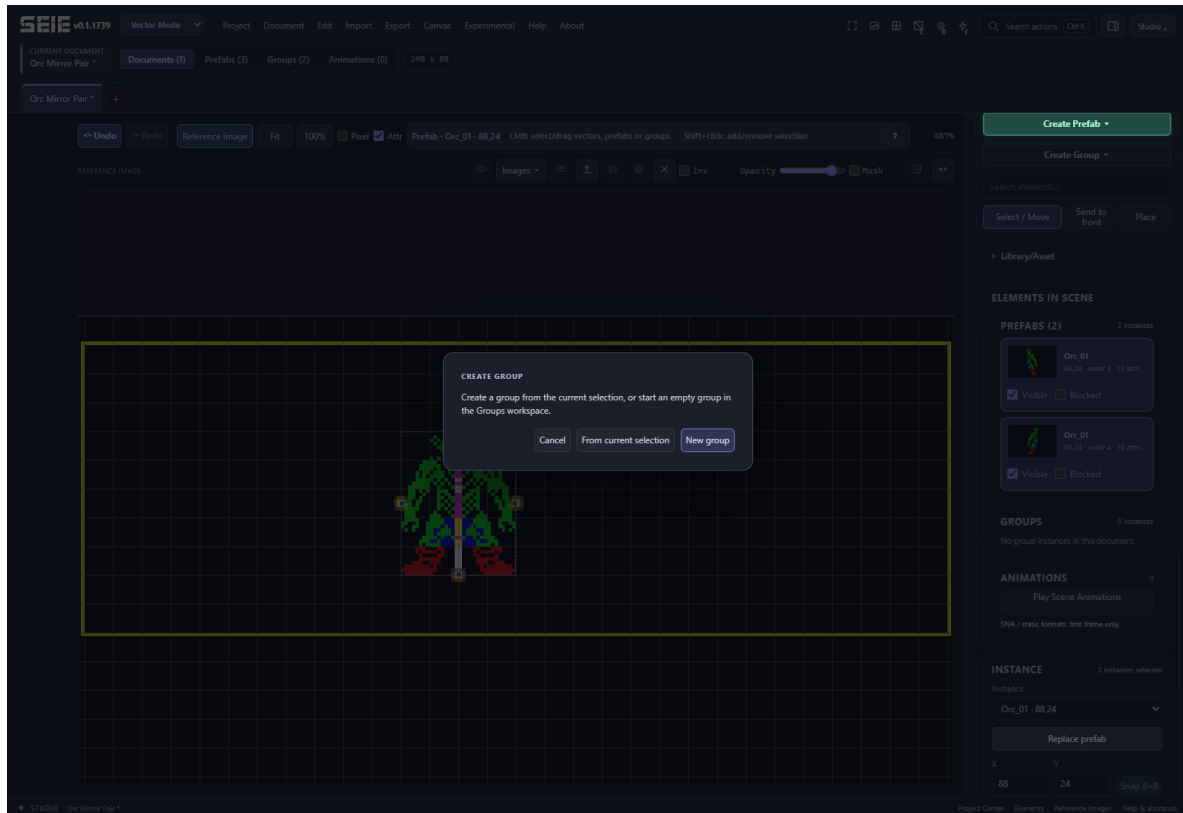
Check the shoulders, waist and feet. Owned black-paper regions cover lower artwork; excluded cells leave holes. Order the parts deliberately rather than accepting mixed pixels at a join.

REAL EXAMPLE

06-Orc-Poses.siep.json retains the source sheet's three component masters, 53 prefab placements and 11 pose groups. These are arrangements of reused artwork, not 11 separately drawn Orc bitmaps.

Save the composition as a group

A pose group stores where its parts go while leaving their source artwork shared.



Creating an Orc group through the actual menu after selecting its components.

Select the components

In a working copy, select only the placements that form the pose. Shift adds or removes members. To rebuild an existing example, Ungroup its scene instance first; the source prefab masters remain.

Create and name

Choose Create Group > From current selection. Name the arrangement Orc Standing or Orc Arms Up. The resulting group should move as one element.

Inspect Edit Group

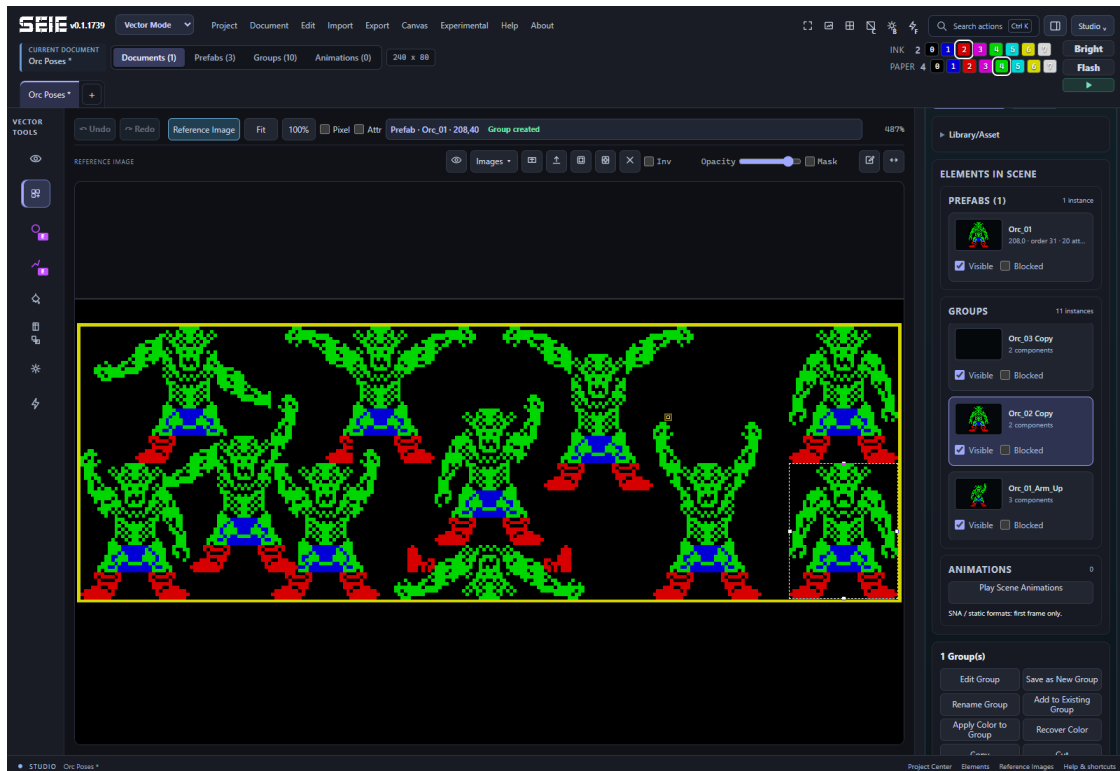
Open the group editor and inspect the component list. Two body placements should still point to Orc_01; the arm and leg placements should still point to their shared sources.

IMPORTANT

The finished Orc Mirror Pair checkpoint contains a real reusable Group Master. Its two prefab components do not become two new prefab definitions.

Read the complete pose sheet

Different arrangements create the visual variety. The source character is not redrawn for each position on the page.



Actual Elements view of the extracted Orc study, with the pose groups and shared prefab library available.

Compare poses

Find Orc_01_Arms_Up, Orc_01_Dance and Orc_01_Dead. Open their group editors in turn. Compare component positions, crops and orientations, not only the final silhouette.

Repeat a saved pose

Choose a pose from Library/Asset > Groups and place it in an empty area. Another use of the pose should not require copying its component masters into new library assets.

Recognise the scopes

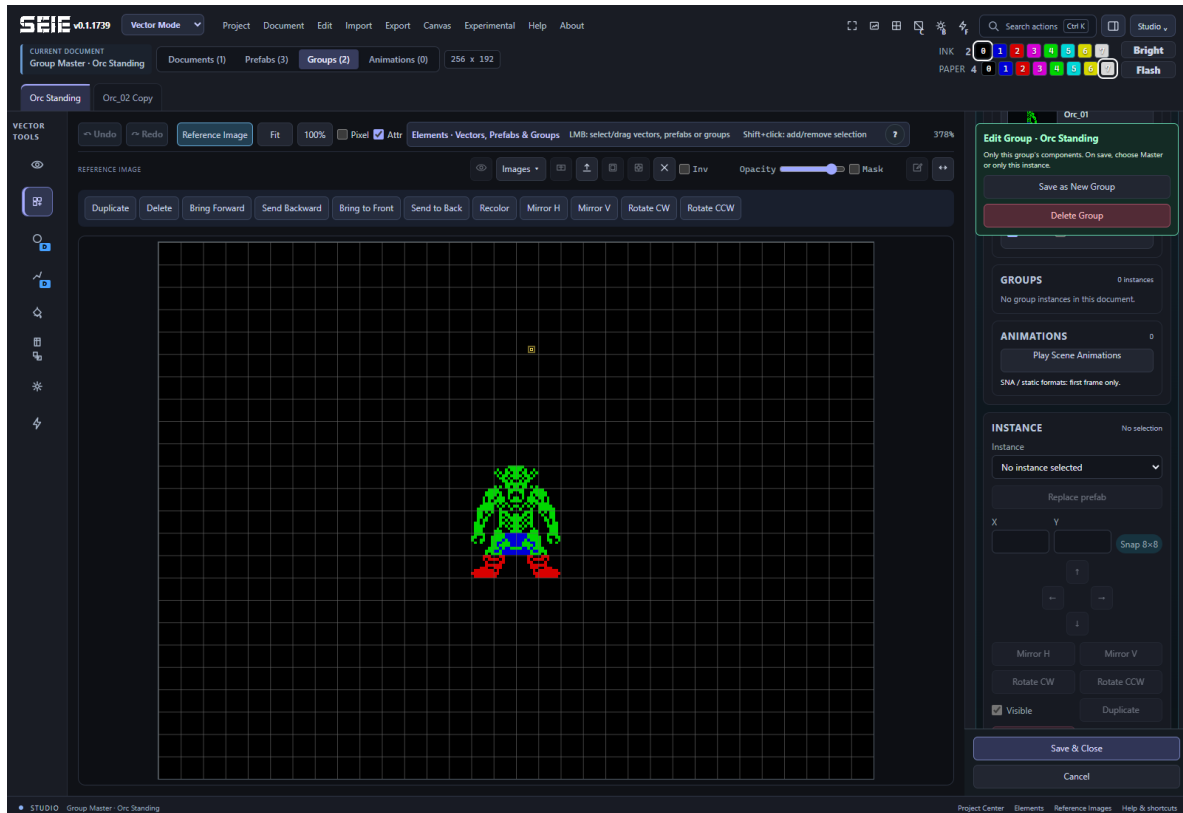
One character image supplies the base; a small set of shared parts supplies articulation; each group supplies an arrangement. Library counts describe definitions, not the number of characters visible in the scene.

CHECKPOINT

06-Orc-Poses.siep.json preserves the actual Orc arrangements from Medieval. The separate shield study was removed from this teaching copy; the original Medieval file is unchanged.

Make one pose variation

A scene-specific gesture should not silently change the pose used elsewhere.



Edit the group arrangement while its component prefab masters remain shared.

Choose a local edit

On a copy, enter Edit Group for a scene instance and move one eligible arm component. Save only this instance if the gesture belongs to this placement alone.

Save a reusable pose

Use Save as New Group for a stance worth using again. Name it by the pose. This creates a new arrangement while still referring to the same body and limb sources.

Publish deliberately

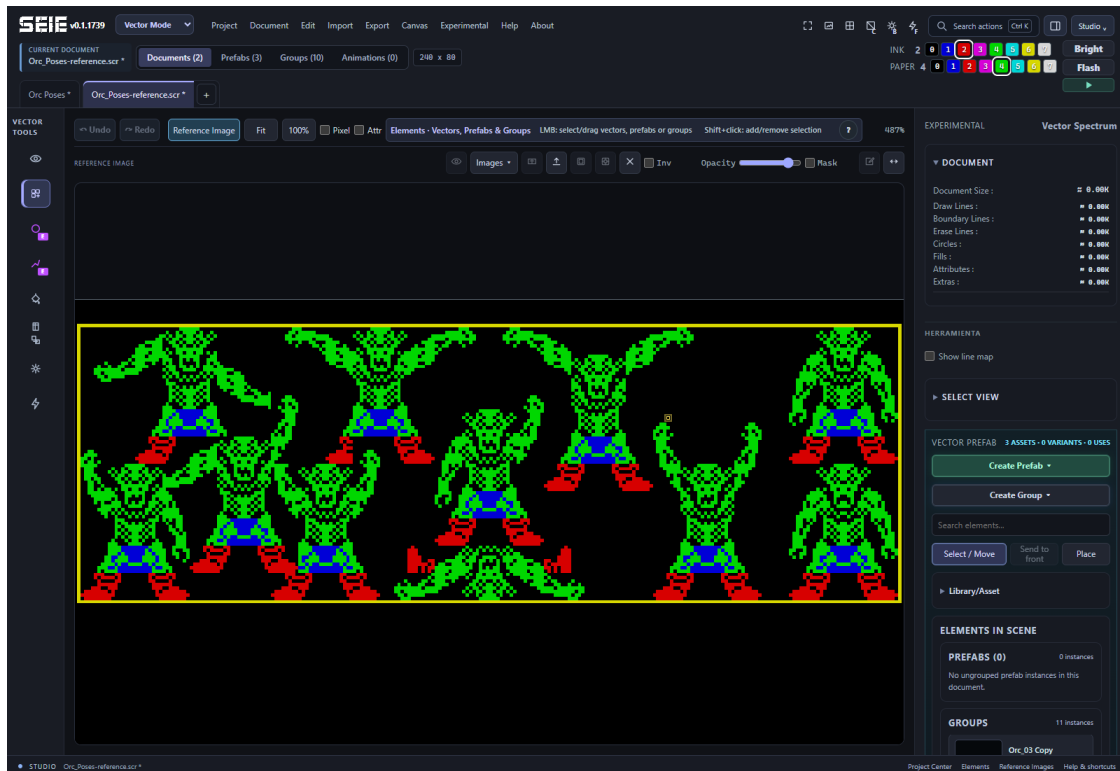
Apply Changes to Master changes ordinary linked uses of that pose. Editing Orc_01 itself has a wider effect: it can change several different pose groups that share its artwork.

CONTROL CASE

Place a second unmodified pose nearby before editing. Predict whether it should change, then verify the selected save scope actually matches that intention.

Reuse the pose sheet as a reference

Document references share a composed background; prefab and group links share the reusable pieces inside it.



A linked document made with Duplicate as Reference in the real interface.

Create the link

Open Orc Poses and choose Document > Duplicate as Reference. Inspect Use Reference in the new document and save the project before making local changes.

Keep additions local

Add a loose Orc placement in the referencing document. The original pose sheet should not gain this new placement. Read Only protects source editing, not the ability to add local content.

Compare dependency levels

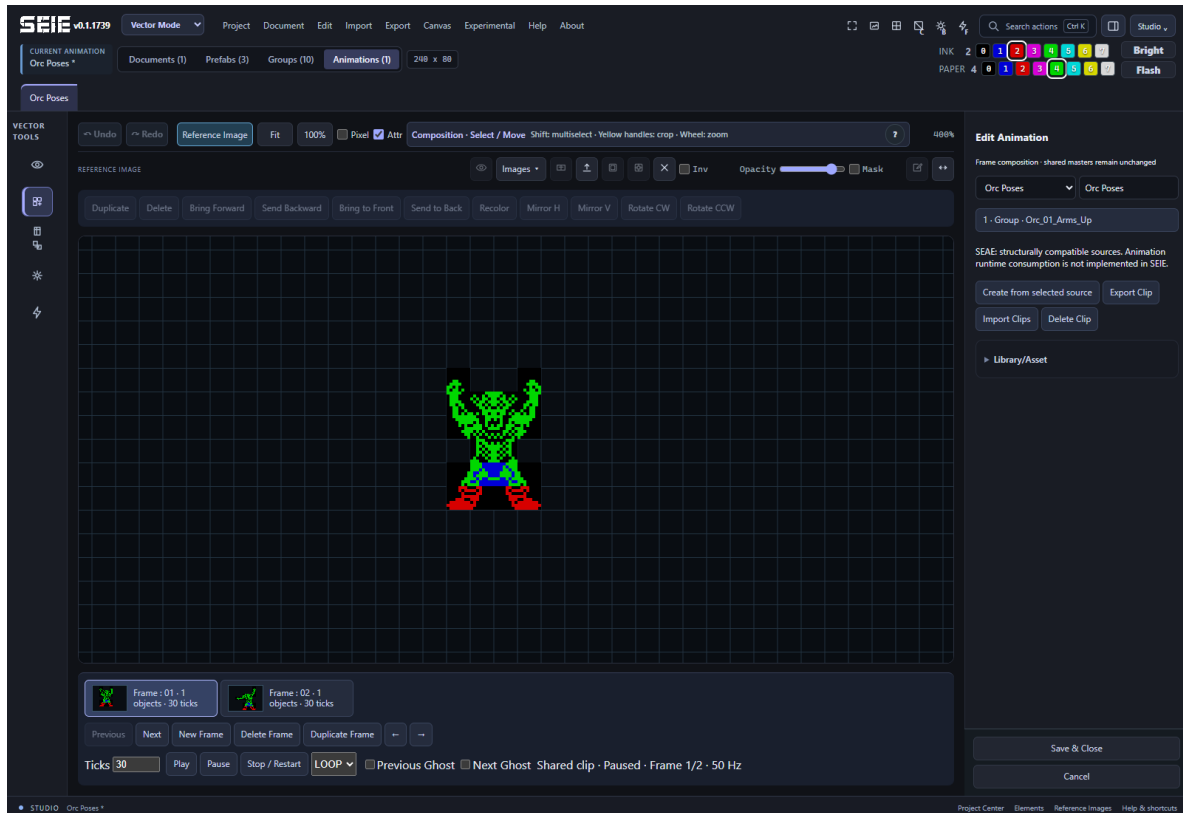
A source-document edit affects the referenced composition. A pose Group Master edit affects its linked arrangements. A prefab source edit affects uses of that part across different poses.

CHECKPOINT

08-Orc-Linked.siep.json was created through Duplicate as Reference and saved through SEIE, not assembled by inventing a private reference format.

Preview two poses as a clip

An animation frame can refer to a pose group without storing another complete character picture.



The real Orc Poses animation workspace, with two group-based frames of 30 ticks each.

Inspect each frame

Open 07-Orc-Animation.siep.json. Frame 01 refers to Orc_01_Arms_Up; Frame 02 refers to Orc_01_Dance. Their source groups still refer to the same shared body and limb assets.

Play and pause

Each frame lasts 30 ticks. The 60-tick loop lasts 1.2 seconds at the editor's 50 Hz timing. Pause on both frames to compare their component arrangements.

Keep export expectations clear

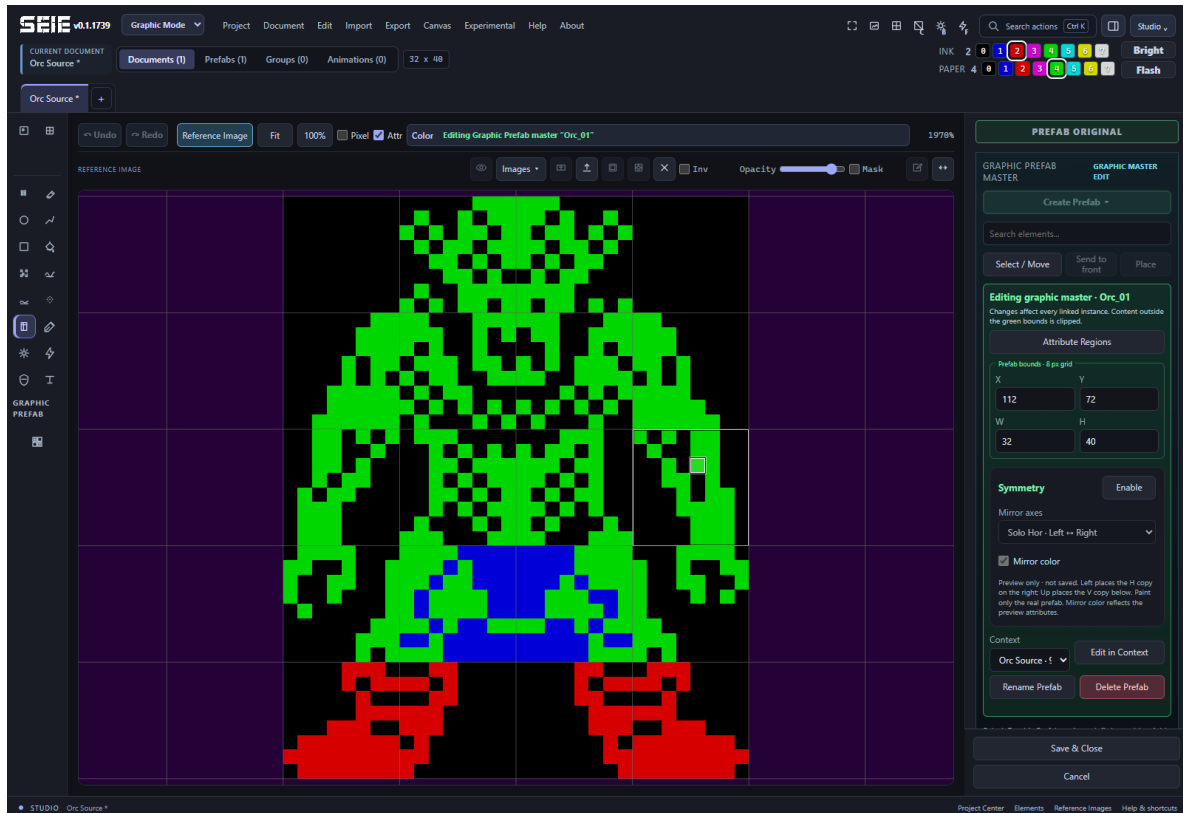
This demonstrates authoring and preview. Current static/SNA output represents the initial frame; it does not add native Spectrum animation playback. A game engine must supply its own frame timing and composition.

REUSE CHAIN

Prefab artwork is shared by pose groups; those groups are shared by frame compositions. Do not rasterize every frame simply to reuse the pose in the editor.

Verify where a shared edit travels

The same visible change can have very different consequences depending on which owner you edit.



The Orc_01 prefab master is a shared source, not one particular pose instance.

Instance transform

Mirror one loose Orc placement. Only that instance should change orientation. The library should not acquire a second Orc master.

Pose arrangement

Move a component in one pose Group Master. Ordinary linked instances of that pose should inherit the new arrangement; other pose masters retain their own layouts.

Source artwork

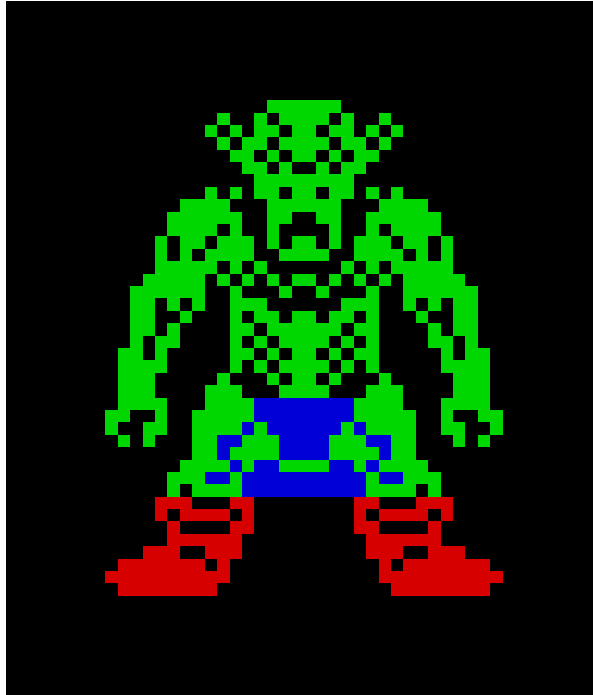
Change Orc_01 on a disposable copy. Every linked use of that source can be affected, including different pose groups and their animation frames. Undo the experiment or reopen the checkpoint.

RECORD THE SCOPE

Write down the edited owner, save action and affected placements. Reuse is useful precisely because edits can propagate; that propagation should always be intentional.

Measure the chosen output

One reused source image can save artwork storage, but no editor structure guarantees a fixed RAM cost for every export strategy.



One source, two mirrored/cropped uses. Their final picture is not evidence of a second source image.

Shared-asset output

When the target preserves asset references, repeated and mirrored uses can share the source data. Placement records, group arrangements, colour overrides and any materialized crops still have a cost.

Bitmap Compressed output

This strategy may store the final composed scene bitmap. Prefab reuse remains valuable for authoring, but do not assume that 20 placed Orcs occupy the RAM of one Orc in that representation.

Validate the result

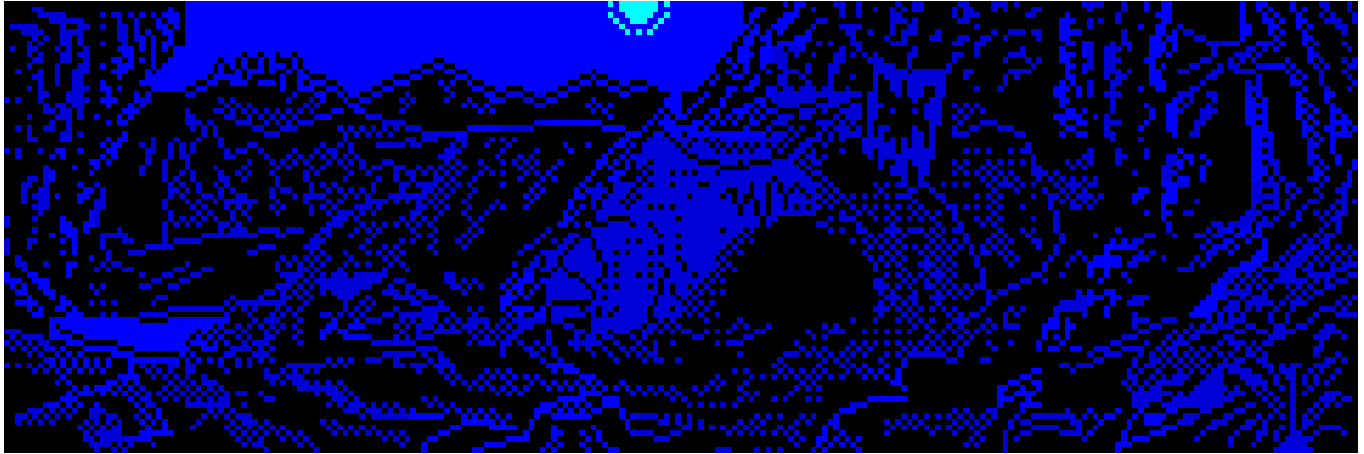
Compare the clean editor canvas with the chosen output. Inspect colours, joins, crops and scene boundaries. Use the export report for final bytes, not the editable JSON size or the number of visible characters.

SAFE CLAIM

The pose sheet reuses one base Orc image and shared pieces instead of independently drawing every pose. Exact Spectrum memory savings must be measured for the selected export plan.

A whole scene from reused prefabs

Orcs_Cave_Gate demonstrates 100% prefab reuse: all its placed assets also occur in other Medieval scenes.



Orcs_Cave_Gate, an actual Medieval scene assembled with 14 prefab placements using 13 already-shared asset definitions.

Reuse the landscape

RiverBridge also appears in TheRiver. Ground_Forest pieces, Cave_01 and Tree_Big_01 are shared with Cave Entrance. Deep_01 comes from the Forest asset set; Shore_01 and Plant_01 are also used in Forest and Castle_Far.

Reuse the accents

Estandarte_01 and Skull_01 are also used in Loading - Fantasy Lore. Moon_small is shared across multiple scenes. A new composition comes from placement, crop, orientation and layer order, not a new drawing for every component.

Apply the Orc lesson to the world

Character poses reuse body parts; this room reuses scenery parts. Store useful arrangements as groups and keep the original prefab links. The same workflow scales from one character to an entire location.

WHAT 100% MEANS

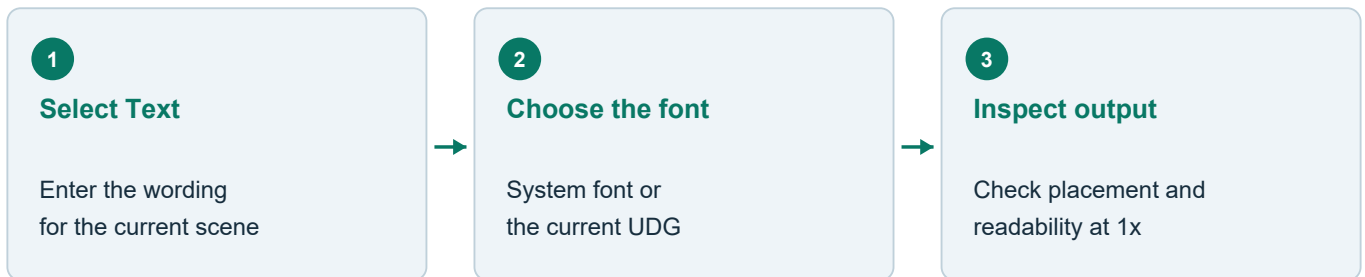
Every prefab source used by this scene is already used elsewhere in Medieval. It does not mean the new scene costs zero bytes: its placement data and selected export representation still need memory.

Place text in the artwork

Text drawn into a document and scene presentation text serve different purposes. Choose the route that matches how the text will be used.

TEACHING DIAGRAM

Place text with the intended project font



Text tool

Use Text for lettering that becomes part of the artwork. Choose its font and layout options, place the preview, then confirm it. Temporary text should be applied or cancelled before exporting.

Project and scene text

Titles, descriptions, presentation and final pages are configured through project/scene controls. They can be rendered by the export's text path rather than being ordinary editable pixel lettering.

Use the project font

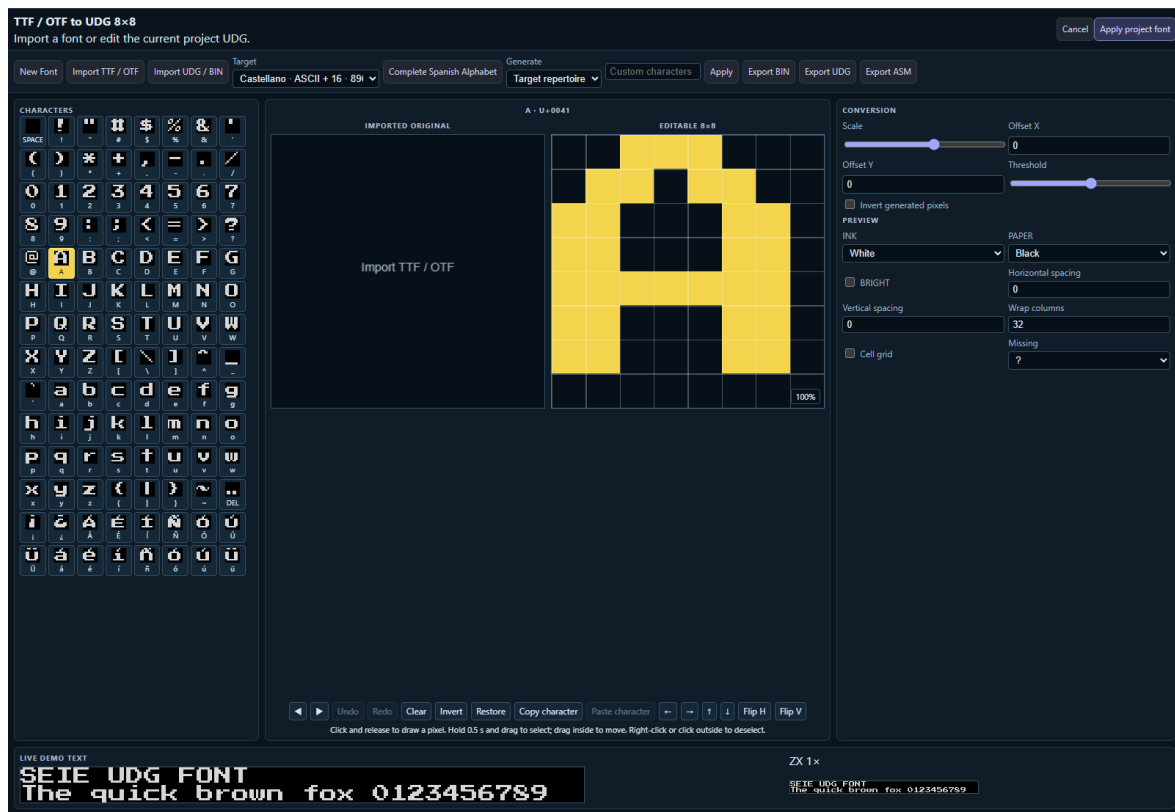
The Text tool can use the current UDG/project font. Check the selected repertoire and missing-character behaviour before relying on accented letters or custom glyphs. A beautiful imported outline font still needs a readable 8 x 8 raster result.

CHECK

Test a short word containing the widest letters and required punctuation. Inspect at native Spectrum scale; enlarged previews can make an unreadable small font look deceptively clear.

Open the font editor

Experimental > UDG Font Editor opens the dedicated project-font workspace. Project Center provides another entry for creating or editing from TTF.



The actual font editor: character repertoire, imported original, editable glyph and live demo.

Choose the output target

UDG A-U contains 21 characters and uses 168 bytes. ASCII 32-127 contains 96 characters and uses 768 bytes. Castellano contains ASCII plus 16 additional glyphs and uses 896 bytes.

Generation is a separate choice

Target defines the exported repertoire. Generate defines which glyphs to rasterize from a source font. Choosing Spanish generation alone does not expand an ASCII output target.

Apply to the project

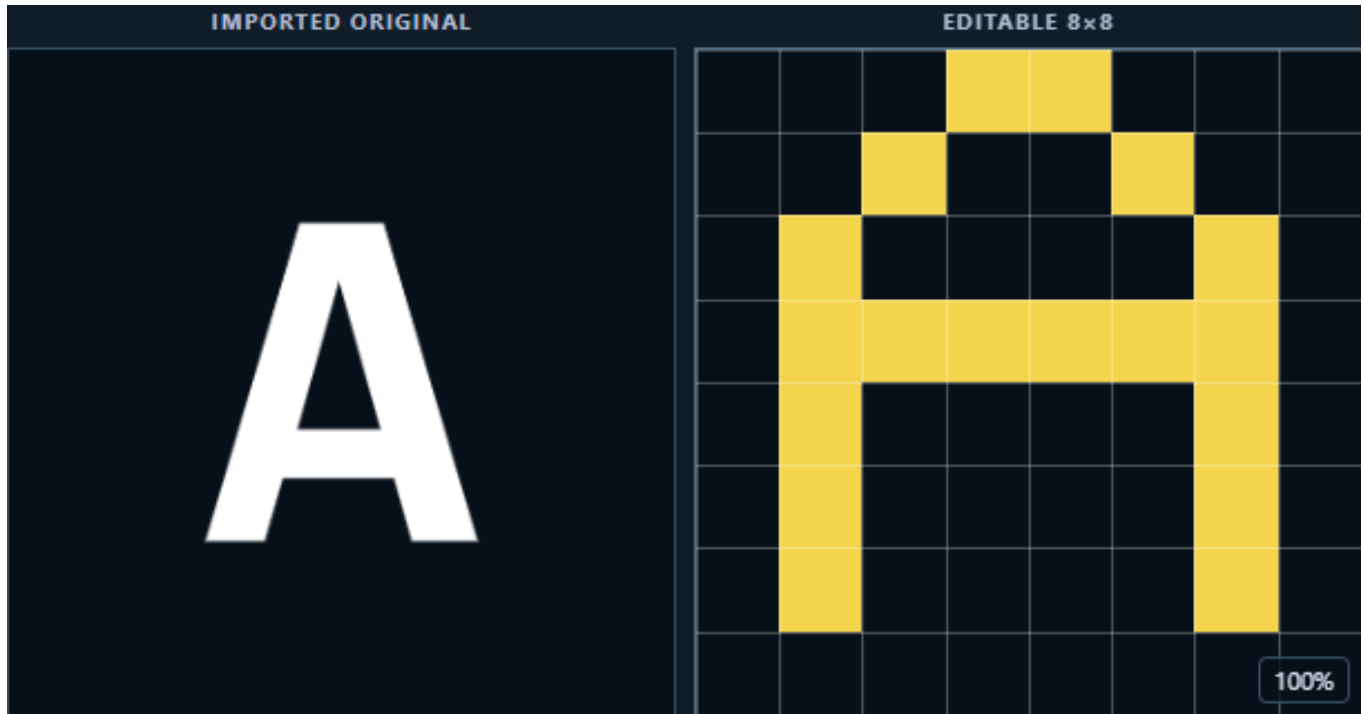
Apply project font commits the edited font and its metadata to the project. Save the project afterwards. Exporting a standalone font file is useful for external tools but does not replace that project save.

DECIDE FIRST

Choose the target repertoire before spending time editing glyphs that might not fit the intended output set.

Import and rasterize a font

Outline fonts and raw Spectrum fonts enter through different import routes.



Imported outline on the left; the 8 x 8 A on the right has been refined with actual pixel edits.

TTF or OTF

Import the source font, choose a generation range and adjust scale, X/Y offset and coverage threshold. These controls determine how the outline maps onto an 8 x 8 glyph. Compare the imported original with the editable result.

UDG or BIN

Use Import UDG / BIN for raw 168-, 768- or 896-byte font data. Confirm the target mapping rather than assuming a filename describes the correct repertoire.

Reversible preview

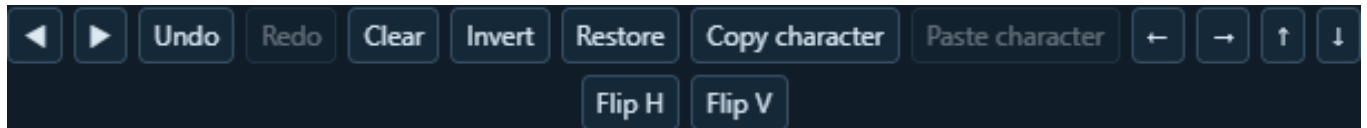
Generation controls preview their result. Apply commits the generation; Cancel restores the previous glyphs. Replacing manual edits requires care and may prompt for confirmation. Keep a standalone backup before regenerating a heavily hand-edited alphabet.

CHECK

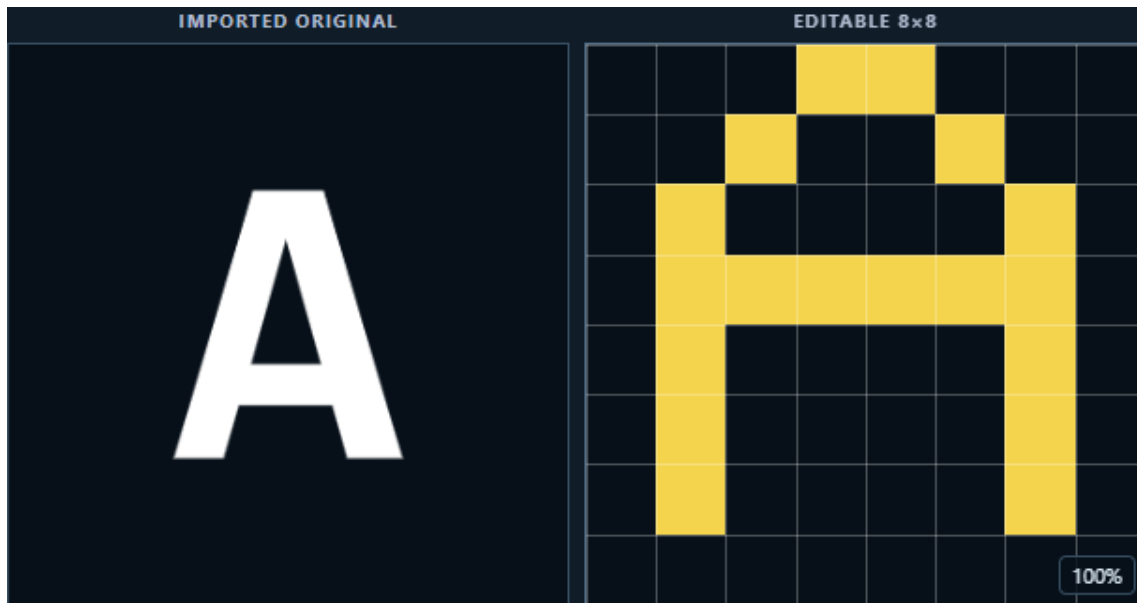
Inspect at least A, M, W, a, e, g, digits and punctuation. A setting that looks good for one letter can clip another letter's accent or descend below the intended baseline.

Edit and move glyph pixels

Each character is an 8 x 8 monochrome cell. Small changes have a large effect on readability.



Actual glyph actions include Copy/Paste character, pixel shifts and horizontal/vertical flips.



A hand-refined 8 x 8 glyph beside its imported outline. The actions above operate on the editable bitmap.

Short click or long hold

A quick left click paints a pixel on release. Hold for more than 0.5 seconds and drag to begin a selection. Drag inside the selected area to move its pixels. Right click or clicking outside deselects.

Correct with small operations

Use Undo/Redo, Clear, Invert, Restore, shifts and Flip H/V deliberately. Copy character and Paste character copy between glyph slots. They are not the same as moving a selected subregion inside one glyph.

Compare at two scales

The enlarged grid is for precise editing; the native-size preview is for judging readability. Use wheel zoom and middle-button pan if needed, but return to the whole cell before accepting the character.

EXERCISE

Copy A into an accented-letter slot, move its body down if required and draw the accent. Check that the result remains readable at native scale and does not touch the next glyph accidentally.

Complete the Spanish repertoire

Castellano adds ÁÉÍÓÚÜÑáéíóúüñ¿¡ to the ASCII repertoire. Choose that target when those characters must be exported.



The Castellano target reveals Complete Spanish Alphabet. Accent strokes still require manual editing.

Seed empty slots

Complete Spanish Alphabet copies suitable base letters into empty Spanish slots. It provides starting shapes for accented vowels and Ñ/ñ, not finished accents and tildes. You complete those marks manually.

Inverted punctuation

The helper transforms question-mark punctuation to form the starting inverted mark using both H and V inversion; the exclamation mark uses vertical inversion. Inspect their final placement on the baseline and within the 8 x 8 cell.

Verify real words

Test words and punctuation such as acción, pingüino, Ñandú and ¿Qué tal? ¡Bien! Check both uppercase and lowercase. A missing glyph should be repaired in the font rather than silently replaced in your intended text.

EXPORT BOUNDARY

A custom ASCII or UDG font needs the appropriate expanded target for Spanish output. Do not assume that selecting a Spanish test string alone changes the exported repertoire.

Use Live Demo Text

The live demo is a reading test, not just a decorative preview. Type directly in the enlarged demo region.



The actual Live Demo Text preview uses the current bitmap glyphs at enlarged and native scales.

Test the current font

The character grid, selected glyph and demo should represent the active generated or edited font. Imported-original comparison remains useful for understanding what detail was lost during 8 x 8 conversion.

Test layout

Change INK, PAPER, BRIGHT, horizontal/vertical spacing and wrapping. Use the cell grid to inspect alignment. Check the native ZX 1x preview as well as the enlarged text.

Include difficult combinations

Try narrow and wide letters, repeated vertical strokes, digits, punctuation and accented characters. A complete alphabet test is useful, but real interface phrases reveal spacing problems more clearly.

SUGGESTED TEST

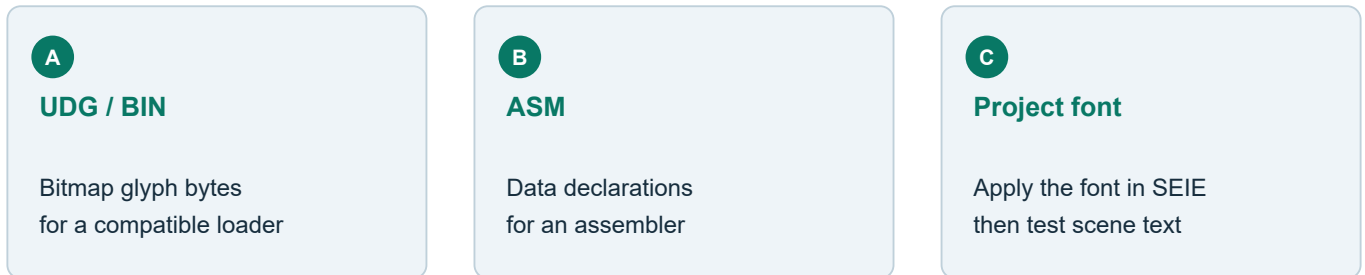
Use a short title, a two-line description and the actual game commands. Check that no glyph is missing, clipped or visually confused with a neighbour before applying the project font.

Export a font and use it in scenes

Font export produces bytes and mappings for a particular repertoire. Your target renderer must interpret that mapping correctly.

TEACHING DIAGRAM

Choose the font output for its consumer



Export formats

Export BIN gives raw data, Export ASM gives assembler-oriented data and Export UDG provides the supported UDG output route. Keep the target repertoire explicit in the filename or accompanying notes.

Apply in SEIE

Apply project font, then configure scene titles/descriptions or the Text tool to use the intended source. Save the project. A standalone font download alone does not guarantee the current scene is using it.

Check output

Inspect the title and a representative accented string in the exported target. Stock ROM text and a custom font can take different resource paths; compare the actual selected output rather than assuming all strings use one source.

MEMORY NOTE

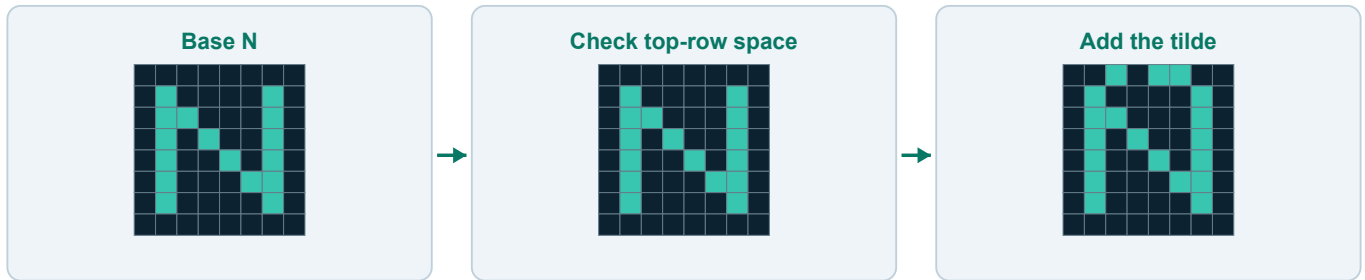
168, 768 and 896 bytes are glyph data sizes, not the total cost of a text renderer, string storage, tables or a complete SNA snapshot.

Readability and font troubleshooting

At 8 x 8 pixels, a readable alphabet is more valuable than a faithful but muddy outline-font conversion.

TEACHING DIAGRAM

Reserve space for the accent inside the glyph



Missing characters

Check the output target, generation range and source font coverage. Draw missing glyphs manually or import a source that contains them. Selecting a character in a demo does not manufacture its glyph.

Clipped or crowded letters

Adjust scale, offsets and threshold, then inspect tall accents and descenders. Leave enough separation for the words you actually use. Do not solve one clipped letter by accepting an unreadably small entire alphabet.

Unexpected old preview

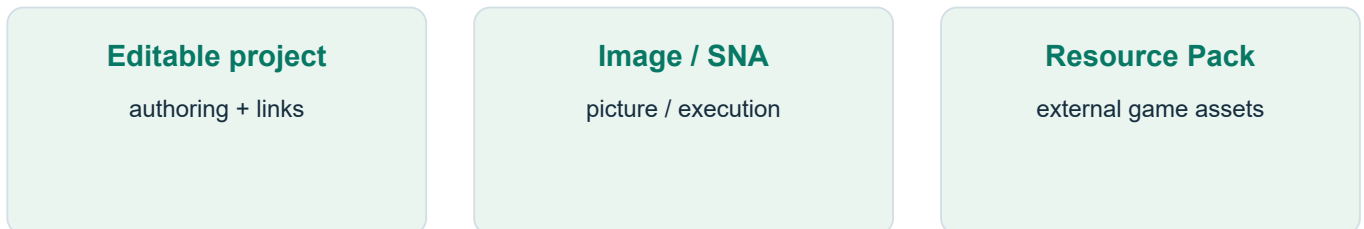
Confirm whether generation is still staged, whether Apply was used and which project font the scene selects. Compare the imported original, editable glyph and live demo as separate views of the workflow.

FINAL CHECK

Save, reopen the project, inspect a manually edited glyph and export a short text-bearing scene. This catches a font that looked correct only during an uncommitted preview.

Choose the right kind of output

There is no single export that simultaneously replaces an editable project, a game resource library and an executable Spectrum presentation.



Teaching diagram - not an application screenshot.

Continue editing

Use project/document and asset exports. They preserve the relationships and authoring data needed for future changes. A PNG or SCR is not an editable prefab library.

Share a finished picture

Use PNG, transparent PNG, JPG or SCR according to the destination. PNG is suitable for exact pixel artwork; JPG is lossy and can alter sharp pixel edges.

Run or integrate

TAP/SNA packages provide a Spectrum execution path. Resource Pack supplies graphics and metadata for an external game whose own code handles drawing, memory and playback.

FIRST QUESTION

Ask whether the recipient needs to edit the artwork, view a picture, run the generated presentation or consume assets in a game. Select the format only after answering that question.

Editable saves and asset transfer

Use a complete project as the reliable working archive, especially when references, variants, groups and animations interact.

TEACHING DIAGRAM

Keep editable data separate from final pictures



Project

Save Project retains documents, libraries and shared settings. Open Project replaces the current workspace, so save first. Merge Project adds incoming material instead of simply replacing the workspace.

Document

Document save/import/export operates at a narrower scope. Verify its dependencies when moving it between projects. Do not assume a bare scene file is a substitute for a deliberately exported shared library.

Assets

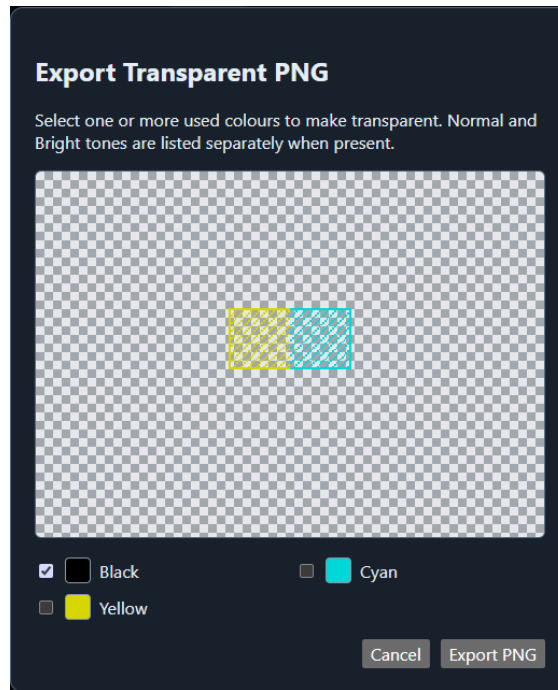
Prefab, Group and Animation import/export routes are for reusable definitions. Groups and clips need their source dependencies. After a merge, inspect name conflicts, remapped relationships and a representative placement.

TRANSFER CHECKLIST

Open in a clean project, count definitions, inspect one use of each imported kind, save, reopen and verify again. This is faster than diagnosing broken references after a large production merge.

PNG, transparency and JPG

Image exports are for viewing or importing into other graphics workflows. They do not preserve SEIE's editable object relationships.



Actual transparent-PNG dialog: used colours and a checkerboard preview before export.

Region or whole canvas

When a selection exists, PNG/JPG export can ask whether to export the selected region or the whole canvas. Read that choice carefully; an unexpectedly small image may be a selected-region export rather than a rendering failure.

Transparent PNG

The dialog lists Spectrum colours actually used, including their bright/non-bright distinctions where relevant. Select one or more entries for transparency and inspect the checkerboard preview before downloading.

Avoid accidental holes

A colour-key transparency choice removes every matching selected colour in the export, not only an object-shaped background. If black is also used inside a prop, making black transparent can remove those interior pixels too.

CHOOSE PNG FOR PIXELS

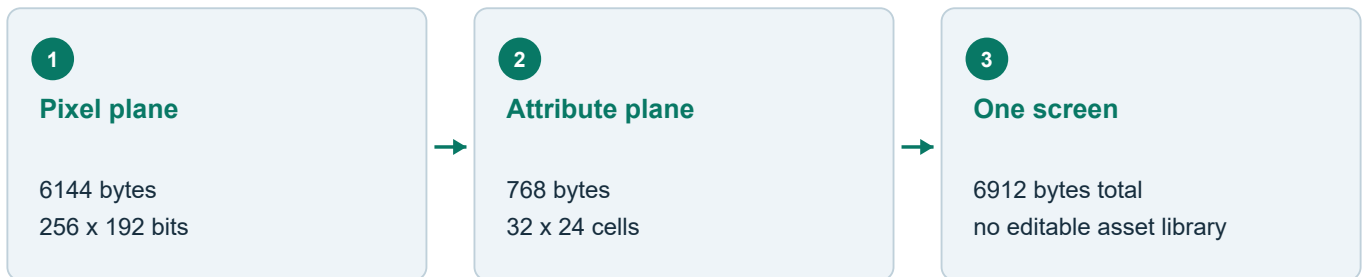
JPG compression can introduce ringing and new colours around sharp edges. Use it for convenient viewing when appropriate, not as the canonical pixel-accurate asset backup.

SCR is a screen image

A standard Spectrum SCR contains the screen's pixel bytes and attribute bytes. It is a picture representation, not a complete game or a SEIE project.

TEACHING DIAGRAM

An SCR contains the finished Spectrum screen



What is inside

A full screen uses 6144 pixel bytes and 768 attribute bytes: 6912 bytes in total. The native pixel memory arrangement is not simply a linear row-major image buffer.

What is absent

SCR does not preserve prefab masters, groups, editable vector commands, animation timing, project document order or game logic. It cannot tell you which window was an instance of which reusable source.

Use it well

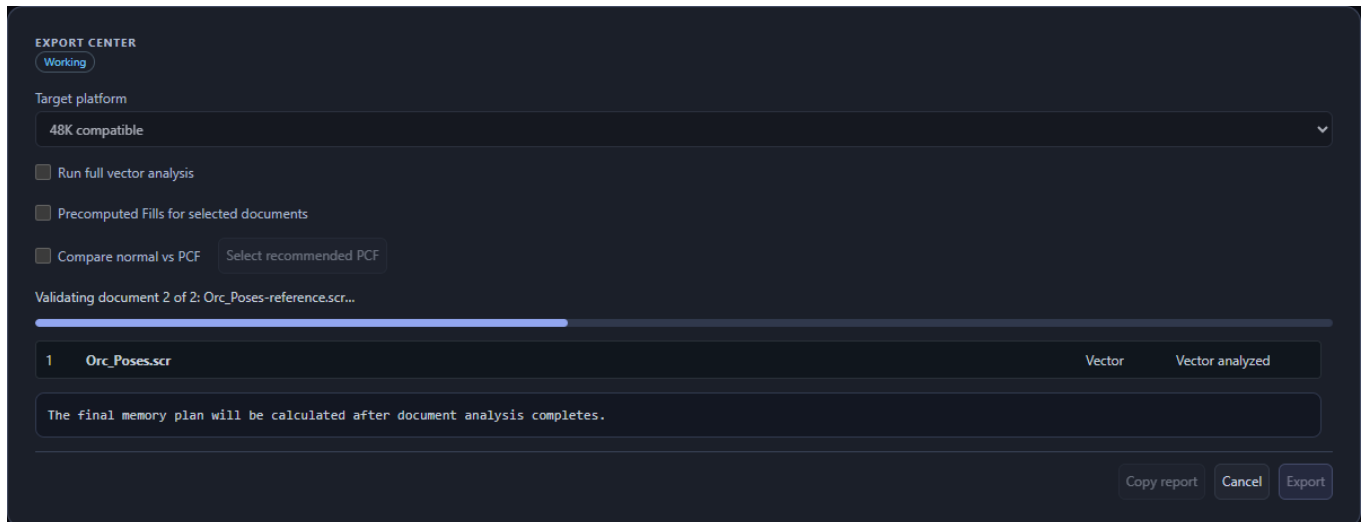
SCR is useful for a finished screen, loading-screen work and comparison with an emulator's screen memory. Keep the editable project alongside it so corrections can be made at the source.

CHECK

If an external tool scrambles an SCR, verify the native Spectrum addressing and inclusion of the attribute area. Do not assume the first 6144 bytes are ordinary scanlines in display order.

Read Export Center

Export Center prepares a plan for the selected documents, representation and target. Its completed report is more authoritative than an early provisional number.



Actual Export Center analysis of the linked Orc pose sheet. Read each selected representation and the final layout result; shared-asset authoring does not imply one universal memory cost.

Select and order

Choose included documents and inspect their thumbnails. Verify source/reference relationships and the intended playback order. A missing or excluded scene is not fixed by changing compression.

Choose representation

The project strategy and per-document mode determine the candidate representation. Inherit/Auto, Vector, Bitmap RAW and Bitmap Compressed serve different goals. Read the Result column to see what was actually selected.

Wait for completion

Preparation stages can be long for complex scenes. Progress and Cancel distinguish active work from a finished or failed plan. After a failure, read the blocking reason before retrying the same unchanged setup.

CHECK

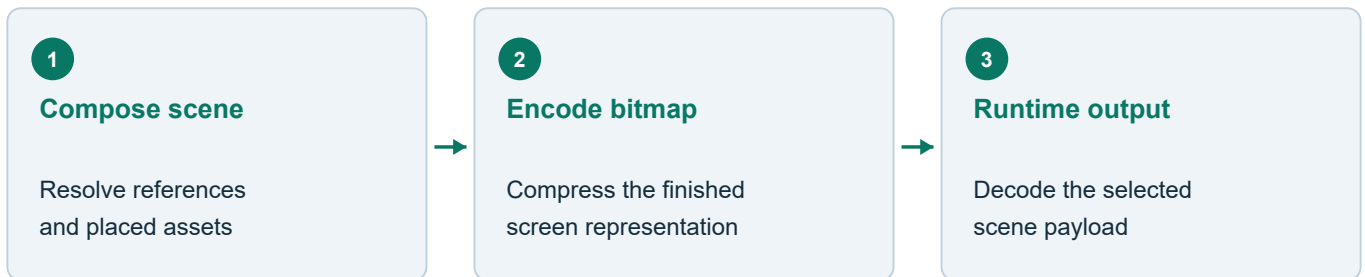
Copy the final report when diagnosing memory or rendering problems. Include the exact project version and exported SNA so the result can be reproduced.

Bitmap Compressed strategy

Bitmap Compressed exports a prepared raster representation. It is a useful starting point when exact final appearance matters more than retaining a semantic drawing stream.

TEACHING DIAGRAM

Bitmap Compressed prepares the final scene image



What changes

The editor's reusable structure remains in the saved project, but the export can encode rendered scene data. Its memory reuse and drawing cost are therefore not identical to those of a Vector command runtime.

Codec choice

The export UI offers codec choices such as ZX7 Mini, SEIE LZ4K and experimental SEAE Byte-RLE. Compression ratio, decompression code and shared overhead all contribute to the final plan. Compare complete cost rather than only one payload column.

Validate the whole picture

Inspect Work Area graphics, external frame, scene text and attributes. Compression should not be used to excuse a fill or mirror discrepancy in the prepared raster source.

PRACTICE

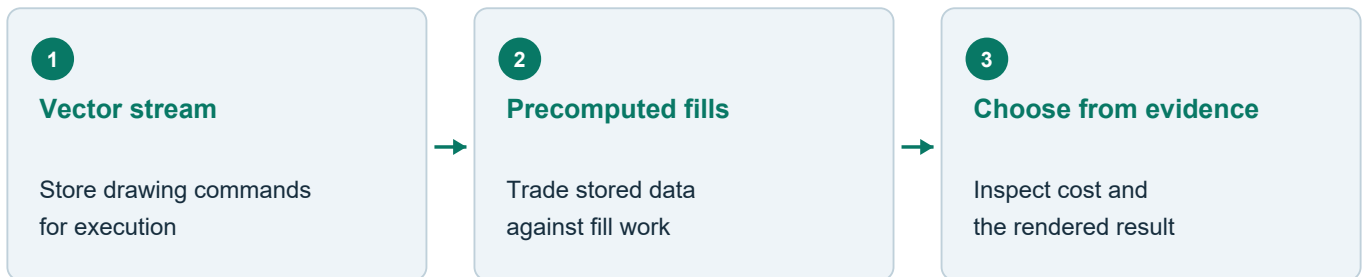
Export the small Orc Mirror Pair example first. Confirm its appearance, then compare a larger project. A successful small example establishes the workflow but does not prove that a large bank layout will fit.

Vector export and precomputed fills

Vector export aims to preserve an executable construction stream. Its constraints and cost differ from a compressed final image.

TEACHING DIAGRAM

Compare executable geometry and stored fill data



Semantic compatibility

Not every composition can be represented by every native path. Read blocking reasons and selected representations instead of assuming that a Vector-authored document guarantees a purely Vector runtime package.

Precomputed Fills

PCF options can trade stored fill data against runtime computation. Compare Normal and PCF results, inspect per-encoding costs and use recommendations as evidence to review, not as a reason to skip visual testing.

Keep authoring untouched

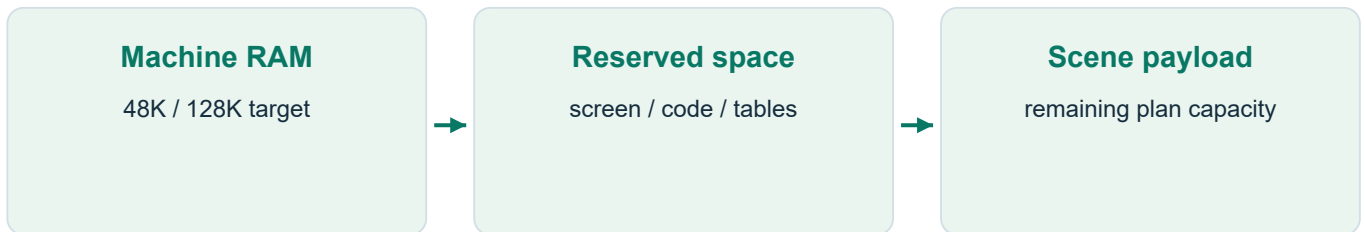
Export analysis should not require changing the project just to compare strategies. Save the editable project, compare plan results and then choose the output appropriate to the target game or presentation.

DECISION

A small stream can still be slow to execute; a fast raster result can cost more data. Optimise against the actual constraint: RAM, loading time, drawing time or visual fidelity.

48K, 128K and usable payload

Machine memory, available scene payload and the physical snapshot file size are different quantities.



Teaching diagram - not an application screenshot.

Reserved space

The display, runtime code, stack, tables, buffers and other resources consume memory. A 128K target does not mean that all 128K are freely available for scene payloads.

Bank layout

The plan allocates data within its paging and fixed-memory constraints. Total free bytes alone cannot prove that an individual asset or bootstrap region fits. Read the named layout constraint when export is blocked.

Snapshot size

A SNA file includes machine state and memory representation. Its size is not a direct measure of the cost of adding one prefab to a game. Likewise, editable JSON size is not a runtime RAM budget.

CHECK

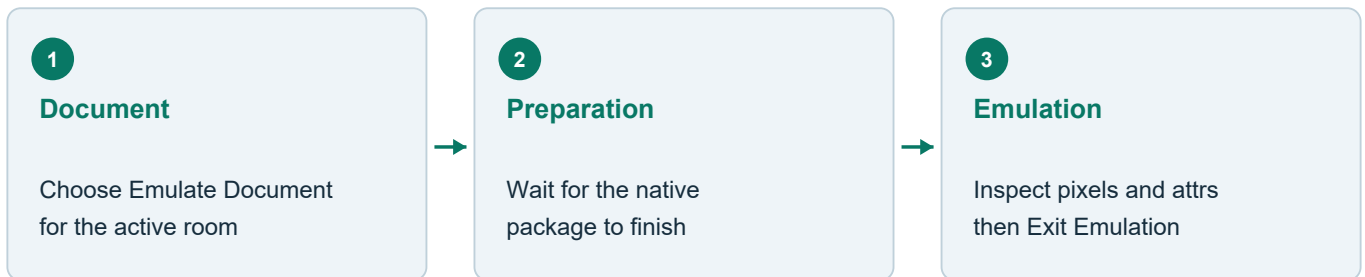
Use the completed export plan for required payload, bank usage and blockers. Treat provisional statistics as provisional until the corresponding analysis stage has completed.

Emulate the current document

Document > Emulate Document focuses on the active scene and its required dependencies. Use it for short iteration loops.

TEACHING DIAGRAM

Preview the active document deliberately



Prepare

Save the project and confirm temporary edits. Select the intended document before starting. Wait for preparation and note any reported error rather than assuming a black screen is a finished result.

Inspect

The emulation view should clearly separate runtime output from the editable canvas. Inspect pixels, attributes, cropped prefabs, mirrors and overlaps without confusing editor guides with runtime graphics.

Exit

Use Exit Emulation to return to editing. Preparation cancellation and exiting an already running emulation are different actions. A successful document preview does not automatically validate the complete project's memory layout or navigation.

TEST CHOICE

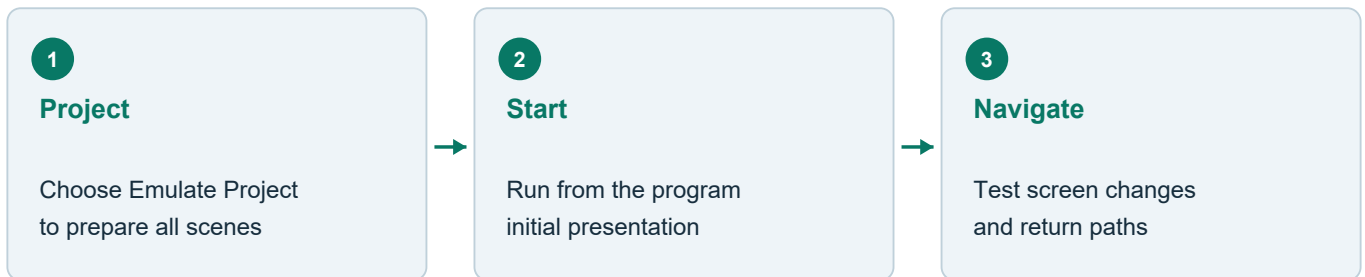
Use current-document emulation while correcting one room. Before release, also test the exported full project, because inclusion order, shared resources and bank layout can differ from the reduced preview context.

Emulate the full project

Project > Emulate Project starts the complete generated program rather than simply showing the active document's finished image.

TEACHING DIAGRAM

Test the actual project startup and navigation



Check the beginning

Inspect presentation text, initial image and start behaviour. Continue through the actual document order. A correct first room does not prove that later bank transitions or reference dependencies are correct.

Navigate

Use the controls reported for the generated SNA, commonly Space for next and Enter for previous. Check the final page and restart behaviour when those project features are enabled.

Test externally too

Save the SNA and open it in the target emulator. Compare the same scenes and navigation sequence. The internal preview is valuable but is not a universal hardware-timing certification for every external emulator or real machine.

RELEASE PASS

Inspect a simple scene, the most complex scene, a reference chain, an overlap with cropped/mirrored prefabs and the final screen. Record the exported file and report used for this pass.

Statistics, repair and diagnostics

Diagnostics help explain a project. They should not become an excuse to make destructive edits without a recoverable copy.

TEACHING DIAGRAM

Use diagnostics for different questions

A

Statistics

Understand the selected memory plan

B

Repair

Review a saved copy before applying repairs

C

Emulation

Check the visual execution result

Statistics

Read scene data, reference cost and shared assets in their stated scope. Document Size is not automatically the sum of every dependency, nor the same thing as JSON file size. Approximate editor metrics and final export planning answer different questions.

Repair

Repair Project/Document can address supported inconsistencies and duplicates. Save a separate project first. Review the report and compare representative scenes afterwards, especially where overlapping content might be intentional.

Performance symptoms

If preparation is slow, distinguish source rendering, scene payload construction and final layout. Record the stage and project, not just a total delay. Cancelling and immediately restarting unchanged work is unlikely to explain the cause.

EVIDENCE

For a bug report, include the smallest saved reproduction, SEIE version, exact operation, export strategy and full error/report. Preserve the original before using repair or conversion as a workaround.

A practical release checklist

A reliable release check is repeatable and scoped. It should cover the features the project actually uses.

Editable archive

Save and reopen the project. Confirm document order, master/variant relationships, group membership, animation frames, reference links and required font characters. Keep a clearly named release checkpoint.

Visual pass

Inspect the whole screen with diagnostic overlays off. Include Work Area edges, text, transparent owned regions, overlapping elements and transformed crops. Check local colour against master colour where relevant.

Target pass

Generate the final plan, resolve blockers, export and run that exact file. Test navigation through all included scenes. Do not substitute a previous SNA or a screenshot from a different strategy as proof of the current build.

ARCHIVE TOGETHER

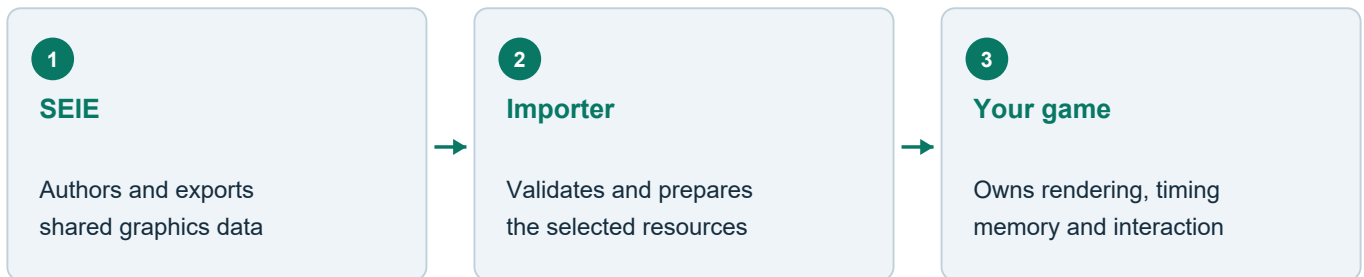
Store the editable project, export report, output file and any engine integration resources together. This makes a later regression comparison concrete rather than dependent on memory.

Resource Pack is not a game engine

A .seiepack is an importable asset package for an external game. It does not inject graphics into an arbitrary compiled game or replace the game's architecture.

TEACHING DIAGRAM

A Resource Pack needs a game-side consumer



What you need

You need the game's source code or an engine resource-import interface, a development toolchain and the SEIE importer/adapters. Node runs on your development computer, not on the Spectrum.

What it contains

The package can describe rendered graphics, masks, groups, animation frame compositions and fonts. You choose scenes, reusable components or both according to the integration workflow.

What the game owns

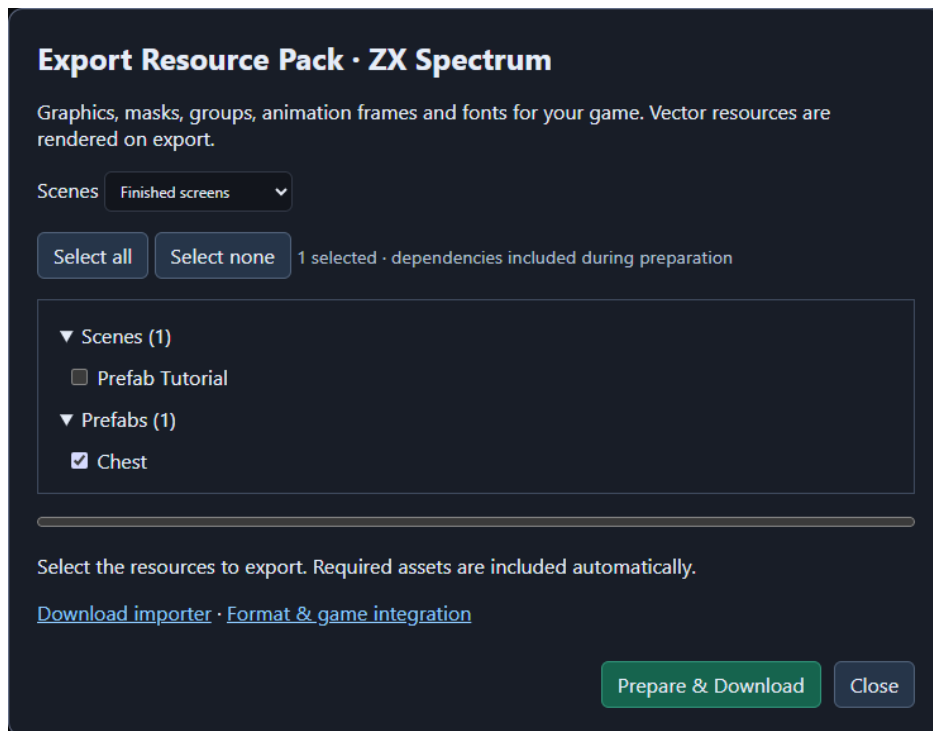
Loading, memory allocation, decompression policy, rendering order, background restoration and animation scheduling remain game responsibilities. Start with one static prefab before attempting a complete animated room system.

COMPANION TUTORIAL

Help > Prefab to Spectrum Game contains the step-by-step Chest integration guide, real export captures, source code and an executable Z80 example. Use it alongside this chapter.

Export one Chest

The first integration should have one source, one known placement and an obvious background to test masking.



Actual Resource Pack export: one Chest master selected, with the scene unchecked.

In SEIE

Open Chest-Tutorial.siep.json from the separate tutorial bundle. Finish any active edit, then choose Export > Resource Pack. Select none, expand Prefabs, tick Chest and leave the scene unchecked.

Download

Choose Prepare & Download and save Chest.seiepack. Selecting the library master exports that master, not an arbitrary scene instance's local crop, mirror or paint.

Instance-specific appearance

If you need a transformed local appearance, use an appropriate resolved scene-components export or deliberately create an independent prefab first. Keep the distinction visible in your asset names and integration notes.

CHECK

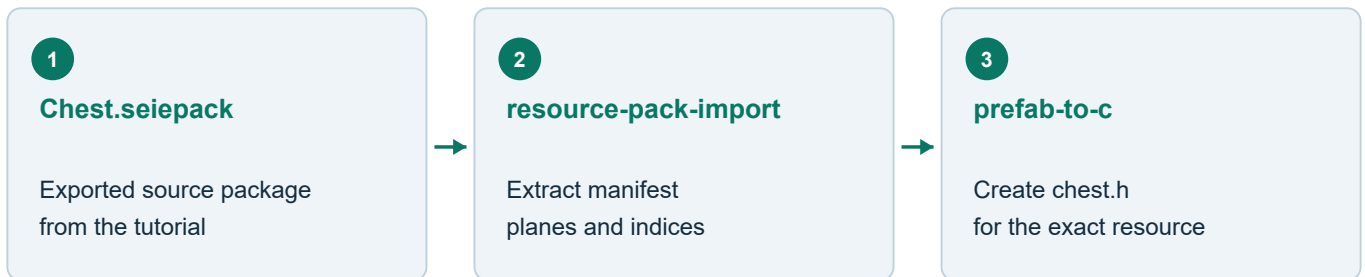
The provided Chest package is a real UI export. Use it unchanged for the first integration so that a drawing-adaptor problem is not confused with a new asset-authoring problem.

Prepare the data on your PC

Use the standalone tools from the Resource Pack tutorial bundle. Keep generated outputs separate from handwritten game code.

TEACHING DIAGRAM

Generate integration data in a new output folder



```
node resource-pack-import.mjs Chest.seiepack game-assets
```

```
node prefab-to-c.mjs game-assets "Chest" chest.h
```

What the commands do

The importer validates the package and extracts its manifest, raw planes and C/ASM indices. The helper creates a header containing the Chest pixel, attribute and mask arrays. Use Node 22 or later for the documented tools.

Avoid overwriting

The importer and helper expect new output paths. For a fresh export, choose new names or move your old generated outputs aside. Do not delete an entire game folder to satisfy a generated-output check.

Identify the resource

If the name is ambiguous, list available resources with the helper and use the exact ID, such as prefab:Chest. A prefab-only export need not contain a scene SCR; that absence is expected.

COST EXAMPLE

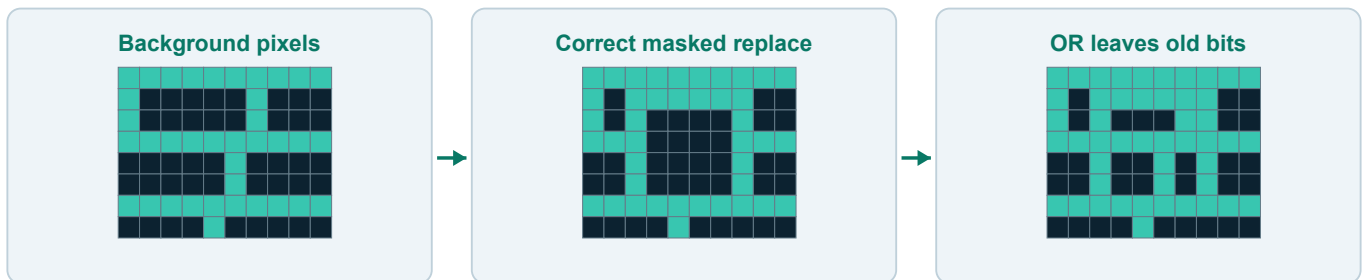
Chest's four planes total 206 bytes: 96 pixels, 12 attributes, 96 pixel mask and 2 attribute-mask bytes. This excludes drawing code, game data and any engine-specific packing.

Draw with masks and attributes

Correct integration must replace owned zero pixels as well as set pixels. Simply OR-ing the graphic over a background is not sufficient.

TEACHING DIAGRAM

A mask replaces owned zero bits as well as one bits



```
destination = (destination & ~mask) | (pixels & mask)
```

Respect ownership

The pixel mask defines which destination bits are replaced. The attribute mask defines which colour cells are written. Transparent cells preserve the background; owned black PAPER regions do not automatically become transparent.

Position and origin

Keep the exported origin offsets. The simple adapter uses pixel coordinates aligned to 8 pixels. The supplied calls draw at (64,80) and (160,80), reusing the same arrays twice.

Adapt to the engine

The sample draws to the native screen at 0x4000 and clips to the full 256 x 192 screen, not SEIE's Work Area. A back-buffer engine must adapt the destination and memory layout. Draw the prop after the background or the next redraw can erase it.

CODE SCOPE

The tutorial validates generated C data and executes the ASM counterpart. It does not claim that every C compiler or linker configuration has been tested.

Run the native integration example

The supplied static Z80 demo demonstrates real drawing of the exported resource, independently of SEIE's own scene runtime.



Native Z80 execution draws two Chest placements over a visible wall background.

Run

Open the linked browser demo or load Chest-Demo.sna in a compatible 48K SNA emulator. The program draws the background, places the two chests and holds the completed image.

Compare

Use Background only and Run Z80 demo to inspect the replacement. Brick lines should not show through owned black parts of the chest. Both placements use the same source data.

Interpret the proof

The supplied verification records 10,256 Z80 instructions and zero differing pixel or attribute bytes against the expected composition. It is a static integration proof, not a complete game loop, timing-accurate hardware test or test of SEIE's SNA runtime.

NEXT STEP

Replace the source graphic only after the unchanged example works. Then test one instance, two reused instances and an overlap with your game's actual background.

From one prop to an asset pipeline

Keep resource generation reproducible as the game grows. Do not turn a successful one-prefab example into hundreds of accidentally duplicated headers.

TEACHING DIAGRAM

Keep one static integration check as the game grows



Shared data

Use the manifest/resource IDs to organise many assets. Include generated static arrays in one translation unit where appropriate. Do not load a packed resource file and also compile the same graphic arrays unless both copies are deliberately required.

Moving objects

Restore the previous background or redraw the scene in a defined order. Drawing only the new position leaves trails. Animated objects also need cleanup when the new frame covers a smaller area.

Memory and paging

Allocate resources within the game's actual memory map. 128K paging, decompression buffers and interrupt constraints belong to the engine. A package file is not automatically a ready-to-run bank layout for any game.

INTEGRATION CHECKLIST

Validate import, dimensions, origin, masks, attributes, screen addressing, redraw order and timing in that order. Keep a minimal static test so future engine changes can be compared against a known result.

When the picture looks wrong

Start with the least destructive checks. Many apparent rendering problems are view, scope or temporary-operation problems.

TEACHING DIAGRAM

Find the scope of a visual discrepancy



Wrong colours or a diagnostic tint

Disable Show Line Map, ghost previews and reference-guide overlays. Inspect INK/PAPER, BRIGHT and FLASH. Check whether the visible colour belongs to a local instance override rather than its master.

Missing or unselectable content

Inspect Select View, instance Visible/Blocked, Block Selection by Type and Read Only references. Check that the active workspace is the document or master you intended. A hidden category may still exist in the saved project.

Residual artwork or an unexpected fill

Finish or cancel temporary selections and text. Inspect ownership, overlap order, source links and duplicate geometry. For fills, check barriers and effective context before adding replacement seeds.

STOP AND PRESERVE

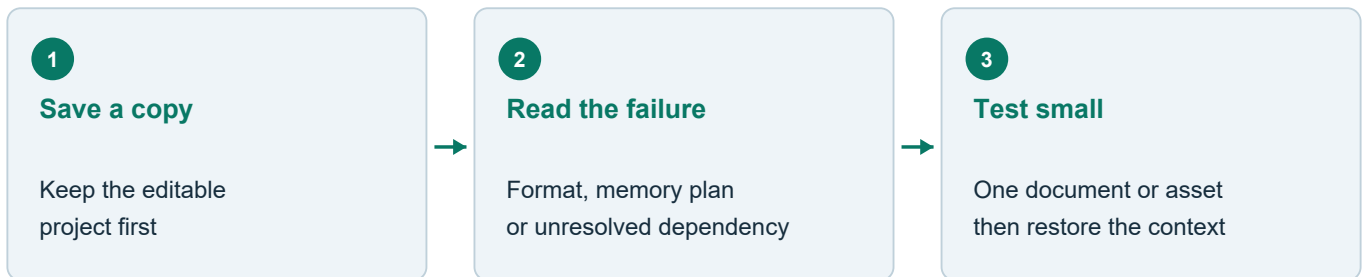
If the result still differs, save a reproduction and exported output before repair or conversion. A workaround that destroys the original evidence makes the cause harder to find.

When saving or exporting fails

Separate editing persistence, preparation errors and target execution. They are different stages with different evidence.

TEACHING DIAGRAM

Reduce a failed operation to a recoverable example



Save/load

Use a project save for libraries and shared relationships. Reopen the actual downloaded file and confirm the expected names and counts. If a browser download failed, a successful canvas edit is not proof of a saved backup.

Export preparation

Read the exact stage and error. A memory-layout blocker can be a local overlap or segmentation constraint even when total payload appears small. Copy the report rather than reducing the problem to '128K should fit'.

Runtime result

Compare the exact exported file with the same document and strategy. A stale SNA, a reduced current-document preview and a full project build are not interchangeable test cases.

BUG REPORT MINIMUM

SEIE version, saved project, document name, steps, expected result, observed result, export strategy, report and resulting file. Include a screenshot, but do not rely on it as the only reproduction.

Keyboard and pointer quick reference

Shortcuts depend on the active context. Text fields keep their normal editing behaviour; the tool HUD is the best guide to context-specific gestures.

Action	Control
Undo / Redo	Ctrl+Z / Ctrl+Y or Ctrl+Shift+Z
Copy / Cut / Paste	Ctrl+C / Ctrl+X / Ctrl+V
Search actions in Studio	Ctrl+K
Cancel operation or close menu	Escape
Zoom / pan	Wheel / middle-button drag
Elements multi-selection	Shift+click
Deliberate group member selection	Ctrl+click where supported
Quick attribute sample	Middle button + right button
Pixel temporary selection	Hold same pixel >1 second, or Shift+drag
Font pixel selection	Hold >0.5 seconds and drag
Close polygon Line selection	Quick double click

Use destructive commands carefully

Delete removes the selected eligible content; context and confirmation matter. Ctrl+D can mean a document or instance duplicate depending on the active selection. Prefer the labelled menu command when the scope is uncertain.

PLATFORMS

Ctrl corresponds to Command where supported on macOS. Browser shortcuts and permissions can also affect clipboard and fullscreen behaviour.

Glossary: artwork and ownership

These terms describe the editable model. They are more useful than treating every visible item as 'an image'.

Term	Meaning
Attribute	INK, PAPER, BRIGHT and FLASH for one 8 x 8 cell.
Boundary	Invisible barrier used to contain Vector fills; not an Erase stroke.
Document	A scene with local artwork, settings and placed elements.
Graphic	Pixel-and-attribute authoring, also called Bitmap in some dialogs.
Group	Reusable arrangement of components and optional local geometry.
Instance	A scene placement linked to a shared asset.
Master	The reusable source definition in a project library.
Owned region	A cell that belongs to a prefab's composition, even if its pixels are black.
Prefab	Reusable artwork with bounds, ownership and source data.
Variant	Related asset changes based on an original prefab.
Vector	Editable semantic construction using drawing/fill commands.

REMEMBER

Black is a colour, not a transparency rule. Visibility is not deletion. An instance is not its master. These three distinctions resolve many beginner mistakes.

Glossary: context and output

Some words describe editor state; others describe a target file or execution process. Keep the two levels separate.

Term	Meaning
Clip	Timed animation asset containing frame compositions.
Crop	Visibility trim of source content, distinct from changing master bounds.
Full Mirror	Whole-source reversal, not copying one half across a centre axis.
Local override	A placement/context change that does not silently redefine the source master.
PCF	Precomputed Fill representation; an export cost/speed trade-off.
Reference image	Editing guide, not automatically exported artwork.
Reference document	Live link to another scene with local additions/patches.
Resource Pack	Asset package for an external game's import and rendering pipeline.
SCR	6912-byte Spectrum screen image, without project relationships.
SNA / TAP	Snapshot/tape output paths for a generated Spectrum program.
Work Area	Active scene region inside the full 256 x 192 canvas.

MEMORY VOCABULARY

Editable file size, compressed payload, runtime RAM, bank capacity and snapshot file size are not interchangeable measurements.

Index, downloads and final checks

Use the linked contents or PDF bookmarks for chapters. The online edition adds text search and direct downloads.

Animation timing	68	Attributes / colour	10
Block / visibility	35	Boundaries / Line Map	23
Clipboard	32	Crop: prefab / group	44
Edit in Context	43	Export strategies	113
Font / Spanish	102	Group scope	58
Instance colour	46	Master / variant	47
PNG transparency	111	Read Only reference	83
Resource Pack	121	Runtime emulation	117
Save / reopen	11	Selection	29
Symmetry	48	Work Area	11

Companion files

The examples ZIP contains ten editable projects and an English file guide, including the Orc source image, created prefab, mirrored group, pose sheet, animation and reference. The separate Prefab to Spectrum Game tutorial retains the Chest package, importer, drawing adapters and executable example.

Before your next project

Choose a mode, establish Work Area and colours, save a checkpoint, build small reusable modules, test one representative overlap and verify the actual target output. Keep shared and local edits explicit.

DOCUMENTATION SCOPE

The guide combines source/UI audit, actual captures and tested companion workflows. It documents current limitations rather than promising unsupported runtime animation or identical memory savings for every export strategy. Recheck version-specific labels when the application changes.