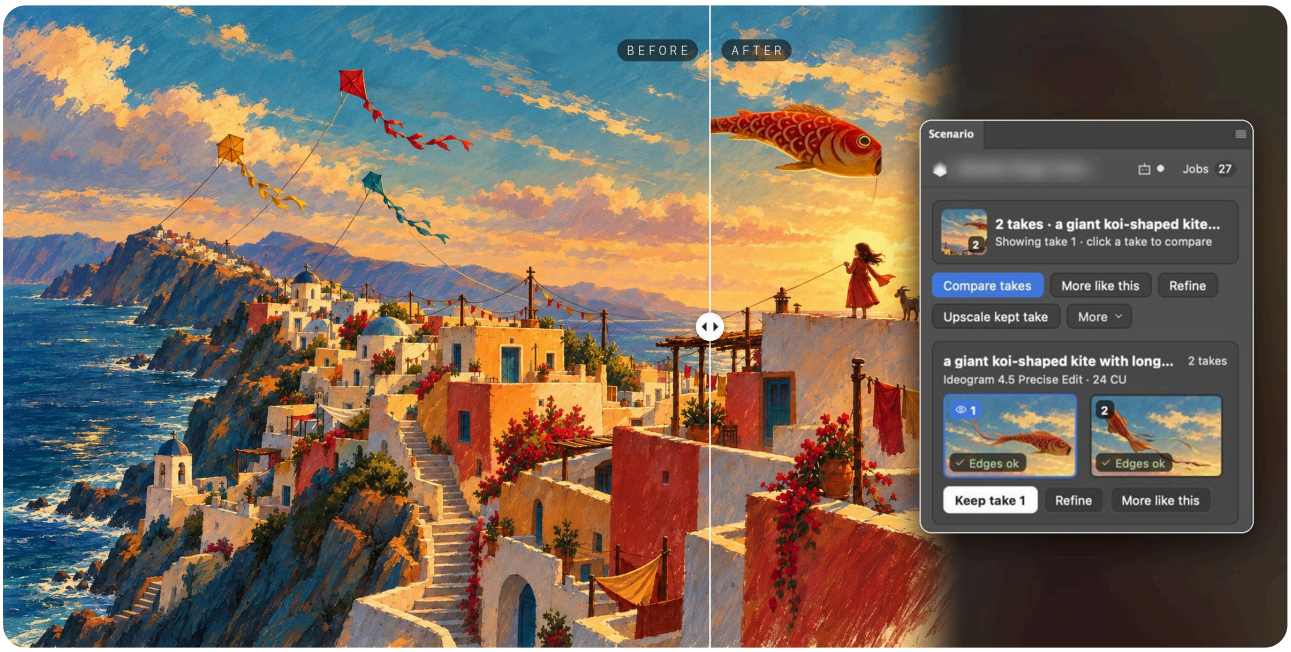
scenario

HANDBOOK

Scenario for Photoshop

Generate, edit, upscale and cut out with Scenario models without leaving Photoshop. Select something, pick an action, see its price on the button, click. The result lands as a layer you can compare, keep or undo in one step.



VERSION	DATE	PHOTOSHOP
0.8.0	10 November 2026	2026 (27.x), tested on 27.11 on macOS

Contents

01	What it does	4
02	Install	4
03	Downloads	5
04	Quick start	6
05	Find your way around	7
06	Actions	9
07	Selection edits: Inside selection or Bolder	14
08	Takes	17
09	References	18
10	Style and trained models	18
11	Library	19
12	Jobs and recovery	20
13	Settings	21
14	Agents (Scenario MCP)	22
15	Costs	29
16	What leaves your machine	29
17	Files and folders	36
18	Troubleshooting	38
19	Known limits	40
20	Security	46
21	Changelog	47
22	Licence	55

FROM YOUR DOCUMENT

A selection, drawn with any tool

The active layer or a cut-out

The whole canvas

References: a layer, a file or a Library image

Your prompt, or none for one-click actions

SCENARIO MODELS

Replace inside | Remove | Make versions | New layer | Upscale | Remove background | Extend canvas | Render sketch | New angle | Split into layers | Formats | Animate

Price on the button checked free before you click. Trained models and styles too.

BACK IN PHOTOSHOP

Smart object layers, masked to your selection

Takes as a group to compare

A mask on the layer for Remove background

One History state per run

Artboards in a new document for Formats

A clip as a new document for Animate

Scenario metadata on every layer

AGENTS Claude Code, Codex, Cursor or Claude Desktop work in this Photoshop through the Scenario MCP, with a card in the panel before any credit is spent.

Not yet on the Adobe Marketplace, and the agent helper is not on npm yet. You install the plugin from the `.ccx` package of a release.

Pictures and recordings in this guide come from real runs in Photoshop 27.11 on macOS with the panel docked at 320 px, at each action's default model and its normal quality; inputs were made with GPT Image 2.5. Claims carry how they were checked: `verified live` (in Photoshop), `tested automated` (test suite, no Photoshop) or `not tested`.

Screenshots may show earlier layouts, and the prices in them are examples, not quotes. Select a screenshot to enlarge it.

01 What it does

- **Reads what you are working on.** No document, a layer, a cut-out, a selection or a set of takes: the panel offers the actions that fit, at most six at a time, and **More** for the rest.
- **Shows the price first.** Every action button carries its price in Scenario credits (CU), checked for free before you click. Nothing is spent until you click, and a price that changes after an upload asks you again.
- **Lands results the Photoshop way.** Each run is one History state, results are smart object layers masked to your selection, takes come as a group to compare, and every layer keeps its Scenario metadata (model, prompt, settings, price).
- **Lets agents work in Photoshop.** A local Scenario MCP server, `scenario-photoshop`, lets Claude Code, Codex, Cursor or Claude Desktop read your document, run generations and land results, with an approval card in the panel before any credit is spent.

On a selection	Replace inside (Inside selection or Bolder), Remove, Make versions, Cut out subject, Select by description
On a layer or image	Make versions, New layer, Extend canvas, Upscale, Remove background, Add background, Render sketch, New angle, Relight to scene, Split into layers, Remove text, Split text, Formats, Animate
On takes	Compare takes, Keep take, More like this, Refine, Upscale kept take
From your project	Place from Library, Library images as references, trained models and styles

Before and after pictures for each action are in the User Guide.

02 Install

You need Photoshop 2026 (27.x; tested on 27.11 on macOS) and a Scenario account on a plan with API access (Starter or above).

1. Download the package of the latest release, `scenario-photoshop-<version>.ccx`, from photoshop.scenario.com/download/latest or the releases page, which also lists its `SHA256SUMS` and a build attestation (`gh attestation verify "$HOME/Downloads/scenario-photoshop-<version>.ccx" -R scenario-labs/photoshop-plugin`).
2. Install it with Creative Cloud's plugin installer, then restart Photoshop (the plugin loads at the next launch). Replace `<version>` with the one in the file name:

```
"/Library/Application Support/Adobe/Adobe Desktop Common/RemoteComponents/UPI/UnifiedPluginInstallerAgent/UnifiedPluginInstallerAgent.app/Contents/MacOS/UnifiedPluginInstallerAgent" --install "$HOME/Downloads/scenario-photoshop-<version>.ccx"
```

`● Verified live` with the release package on macOS on 2026-10-23, with Photoshop's Developer Mode on (whether the installer needs it was not tested): install, restart, sign-in and a first image took 1 min 53 s. A script did the clicks and typing, so this time has no reading in it, and 59 s of it was the generation. Double-clicking the `.ccx` should do the same through Creative Cloud; `○ not tested`.

3. In Photoshop, open **Plugins > Scenario**.

To build the package yourself instead (Node.js 20 or later), run `npm ci`, then `npm run package`, which prints the file it writes, `build/scenario-photoshop-<version>.ccx`, and install that file.

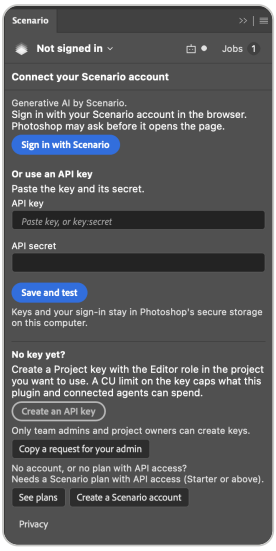
03 Downloads

- This handbook, 0.8.0: **PDF for print, A4** (`scenario-photoshop-handbook-0.8.0.pdf`) and **one HTML file that works offline** (`scenario-photoshop-handbook-0.8.0.html`).
- The plugin is not on the Adobe Marketplace yet. Download the `.ccx` package of the latest release from photoshop.scenario.com/download/latest and install it as described in Install.
- The agent helper is not on npm yet. Agent clients run it from a checkout; see Agents (Scenario MCP).
- Source code, issues and the changelog: github.com/scenario-labs/photoshop-plugin, MIT license.

04 Quick start

From install to a first image in about two minutes, once the package is built: 1 min 53 s measured live with the release package on 2026-10-23, with the clicks done by a script (see the README).

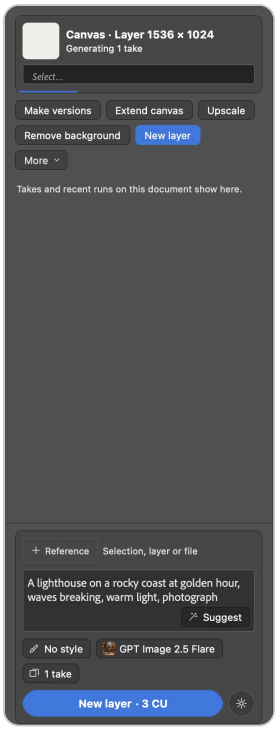
1. **Install** the `.ccx` and restart Photoshop, then open **Plugins > Scenario**. The panel opens on **Connect your Scenario account**.



The sign-in screen: Sign in with Scenario, then the API key and secret fields, Save and test, and links to create a key

2. **Sign in with Scenario.** Click **Sign in with Scenario**. Photoshop may ask before it opens Scenario's sign-in page in your browser. Approve there; if the page asks for a code, enter the one the panel shows (**Copy code**). **Open the page again**, **Copy code** and **Cancel** stay under the code while the panel waits. If your account can use several projects, **Choose a project** lists them by team; one click uses it. The project shows in the header, and **Choose another project** in Projects switches. Your sign-in stays in Photoshop's secure storage on this computer; **Sign out** in Settings removes it and asks Scenario to end it.
3. **Or sign in with an API key.** On app.scenario.com, in the project you want to use, create a **Project key** with the **Editor** role. A CU limit on the key caps what this plugin and connected agents can spend. Only team admins and project owners can create keys; **Copy a request for your admin** prepares the message for them. Paste the key and the secret (pasting `key:secret`, or the two on two lines, fills both fields) and click **Save and test**. The plugin checks the key with two free reads and keeps it in Photoshop's secure storage on this computer.
   ● `Verified live` with the release package: `key:secret` pasted into the key field, then **Save and test**, which answered in about a second.
4. **Make a new document** with **File > New** (Photoshop hides docked panels on its Home screen, so the panel shows once a document is open). On a blank canvas the panel opens on **Create image**: type a prompt and click the button, which shows its price. The image is made at the canvas's size and lands as a layer above the Background. On an image or a layer, **New layer** does the same.
5. The context card shows the progress (Queued, Generating, Downloading), then the image lands as a new layer in one History state. ● `Verified live`.

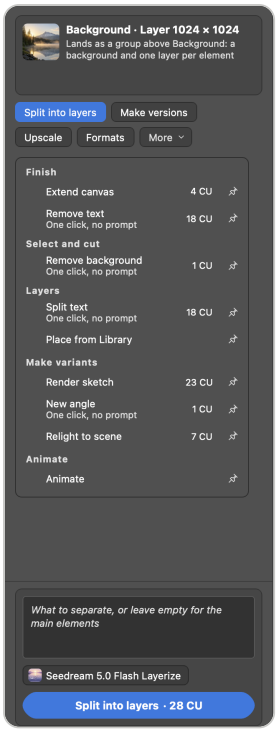
05 Find your way around



The panel while a run is generating: context card with progress, action chips, composer with prompt, chips and the priced button

From top to bottom:

- **Header**: the project name (click to switch projects), the agent icon with its dot, and **Jobs** with a count.
- **Context card**: a picture of what the panel will work on and where the result lands ("Lands inside the selection, masked", "Lands as a new layer"), with the progress during a run. Its **Select...** field is Select by description.
- **Action chips**: the actions for this context, at most six, then **More** for the others, grouped. A pin in **More** adds an action to the chips.
- **Takes and Recent**: the last run's takes, then the other runs placed in this document.
- **Composer**, pinned at the bottom: **+ Reference**, the prompt (with **Suggest**), the chips that set the model and its options (each shows its value), and the action button with its price. The gear icon holds the quality level and advanced options. To write a longer prompt, drag the grip on the composer's top edge up to make the box taller; double-click the grip to go back. The panel keeps the height for next time.



The More menu: Finish, Select and cut, Layers, Make variants and Animate groups, each action with its price

06 Actions

What the panel offers depends on what is selected. Each action below was `verified live` on 27.11 unless marked; the pictures are the before and the result of that run.

Blank document

A document of one Background of a single color, as **File > New** makes it, offers **Create image** first: made at the canvas's aspect (the model's nearest size), in up to four takes, landing as a layer at canvas size in the same document. **Place from Library** is beside it. `Verified live` (the context and its price); `tested automated` (the landing).

Selection actions

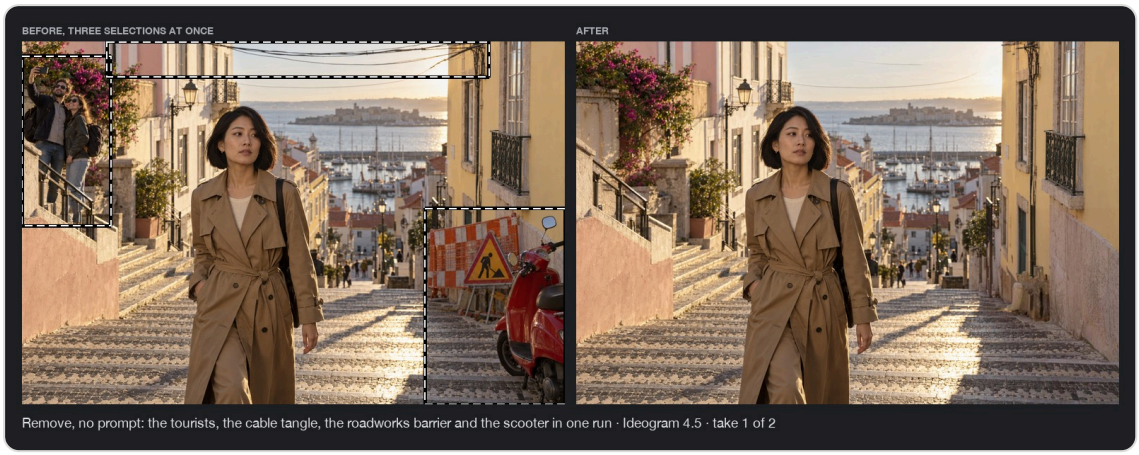
Draw a selection with any Photoshop tool. The context card reads "Selection W × H on <layer>".

Replace inside: describe what goes inside the selection. Default model Ideogram 4.5 Precise Edit (Inside selection) or GPT Image 2.5 Flare (Bolder), see the next section.



BEFORE AND AFTER A patch of empty sky selected over a painted cliff town, then a giant koi kite inside it; at 100% the town outside the selection is unchanged

Remove: removes what is inside the selection and fills it with what should be behind. No prompt needed (a built-in one is sent).



BEFORE AND AFTER Tourists, a cable tangle, a roadworks barrier and a scooter selected at once on a harbor stairway, then removed in one run

Cut out subject: the subject inside the selection, on a new layer with a transparent background. The layer stays as it was and the selection is dropped in the same History state.



BEFORE AND AFTER A man with wind-blown copper curls in front of a busy tile wall, then cut out and shown on a flat color; at 100% the flyaway strands are kept with no tile left between them

Select by description: type what to select in the context card's **Select...** field ("all the sailboats"), click **Select**. The masks the model finds become a Photoshop selection, ready for the next action.

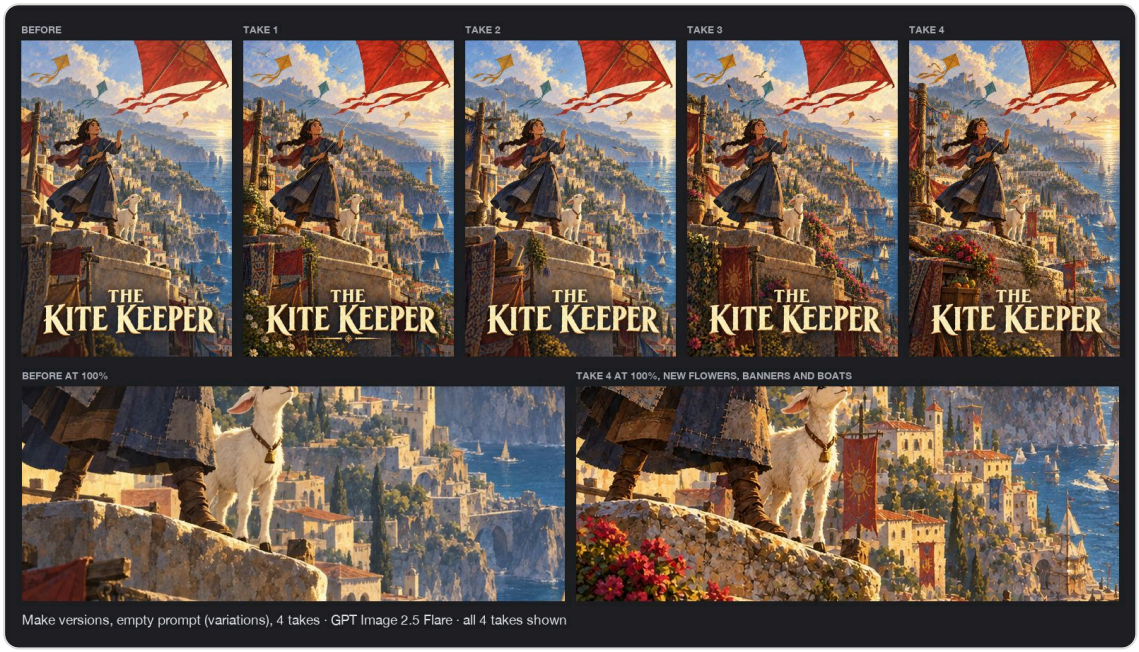
New layer in a selection: a new element drawn for the selection, sized to it and placed inside it. **Lands: Cut out** (the default) asks the model for a transparent background; when the model still returns a full picture, Photoshop's own Remove Background cuts it at placement, as a layer mask you can paint on. **Full frame** keeps the model's own background inside the selection box.

Make versions also works on a selection.

Layer and image actions

With no selection, the panel works on the active layer, or the whole image for a flat document.

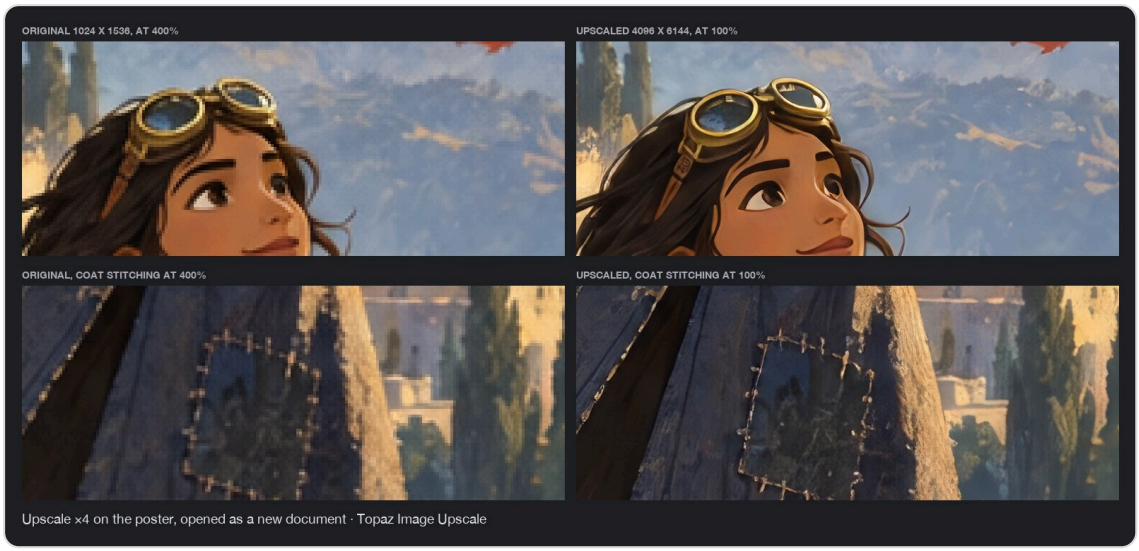
Make versions: variations of the layer, or the change you describe, as takes to compare.



BEFORE AND AFTER A painted poster for The Kite Keeper and four versions of it with the same keeper, goat and title and new flowers, banners and boats; at 100% the details differ

New layer: a new image on a layer above, from a prompt.

Upscale: 2x or the other factors the model offers (Topaz Image Upscale), from all visible layers, opened as a new document at the larger size. The result card compares the same region at 100%.



BEFORE AND AFTER The poster's goggles and coat stitching: the original at 400% looks soft and blocky, the 4x upscale at 100% shows clean brass, hair and stitches

Remove background: one click (851 Labs Background Remover); lands as a mask on the layer, so the hidden pixels stay. **Cut out to a new layer** (More) keeps the layer as it is and puts the subject on a masked copy above.

Add background: on a cut-out layer, describe the scene; it lands below the layer at canvas size.

Render sketch (More > Make variants): turns a line drawing into a finished image from your prompt (GPT Image 2.5 Sunburst).



BEFORE AND AFTER A pencil storyboard sketch of a cliff-side town with kites and a figure on a roof, rendered as painted animation concept art at sunset; at 100% the rooftops, the keeper in her indigo coat and the goat keep their places

New angle (More > Make variants): Turn left 45°, Turn right 45°, Side view, Back view or From above. No prompt.



BEFORE AND AFTER A character sheet of the kite keeper made with New layer, then her front figure cut out and turned to a side view and a back view with New angle; coat, braid, goggles and red kite string stay the same

Split into layers: rebuilds the image as a group: a clean background and one layer per element, each placed where it was.

Remove text (More > Finish): the same image without its text, as a new layer above. **Split text** (More > Layers) gives the clean base and the text on its own layer instead.

Formats: reframes the image to Story 9:16, Feed 4:5, Square 1:1 or Wide 16:9 (Scenario Smart Reframe), as an artboard in a new "<title> formats" document. Type layers stay out of the reframe: each title is laid out again on the artboard at its place in the frame, still editable. Each new format is added to the same document, beside the others. If a subject gets cut at the edge, name it in the prompt ("keep the kite and the keeper whole in frame") and run that format again.

Animate (More > Animate): a short clip from the image and your prompt (Minimax H3 Fast, 5 s by default). The clip opens as a new document and the MP4 is saved in the output folder. **Frame to layer** brings its last frame back as a layer.

Relight to scene (More > Make variants): relights a cut-out or split element to match the visible image. `Verified live` on 0.6.2; no picture in this guide.

Extend canvas (More > Finish): grows the canvas to a ratio and fills the new area. The default model is Reframe at 2K, which redraws the frame around the original; it costs more than the earlier default (the button shows the price), and it has no per-side growth (pick GPT Image 2.5 Flare for **+20% each side**). Look at the old edge at 100% before keeping the result; see Known limitations.

Place from Library (More > Layers): see Library.

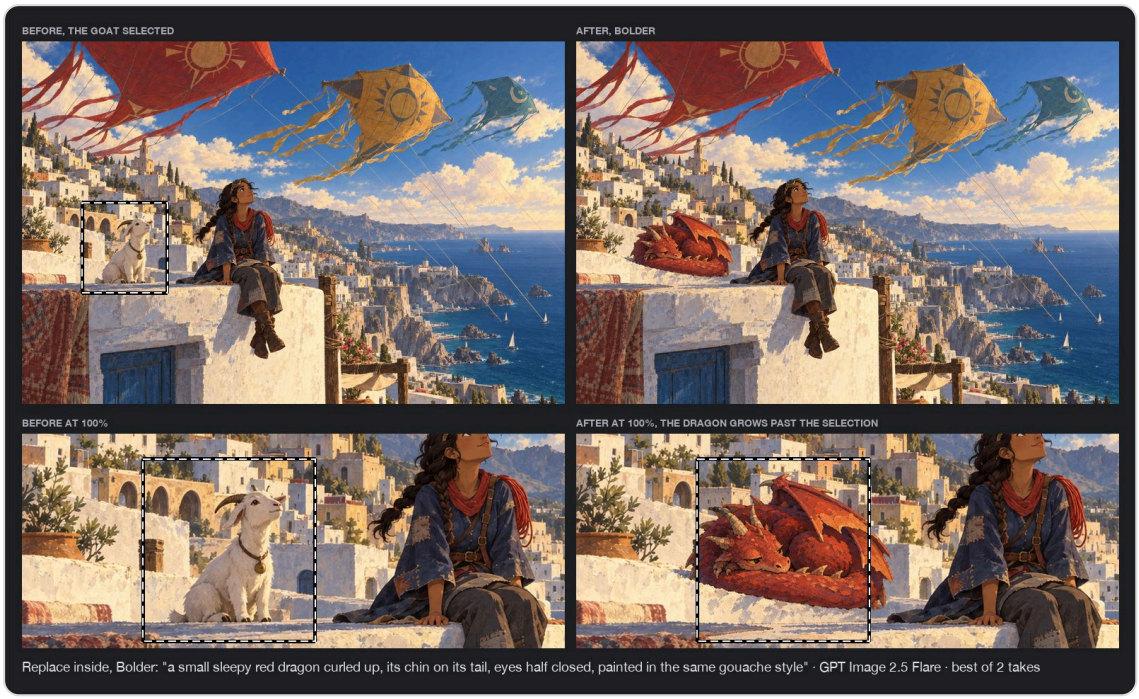
07 Selection edits: Inside selection or Bolder

Replace inside has an **Edit** chip with two modes, each with its price:



The Edit menu: Inside selection and Bolder, each with a description and a price

- **Inside selection** (Ideogram 4.5 Precise Edit): changes only what is inside the selection; everything outside stays as it is. Good for a change that fits the shape you drew.
- **Bolder** (GPT Image 2.5 Flare): a stronger change. The new subject may reach past the selection, and the layer mask grows to keep it whole. Good when the new object is taller or wider than the old one, as with the dragon below.



BEFORE AND AFTER A goat selected on a painted rooftop, then a sleepy red dragon in its place; at 100% the dragon reaches past the selection and blends into the roof

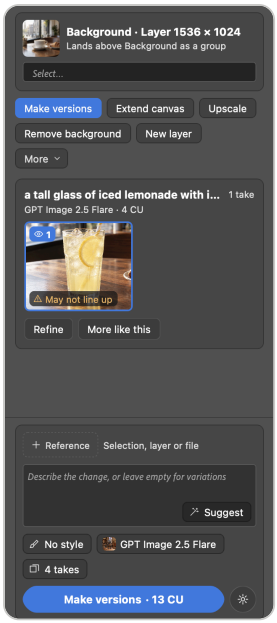


Replace inside with Edit Inside selection, Ideogram 4.5 Precise Edit, 12 CU



Replace inside with Edit Bolder, GPT Image 2.5 Flare, 4 CU

After a run, the take offers **Try Bolder** or **Try Inside selection** to run the other mode.

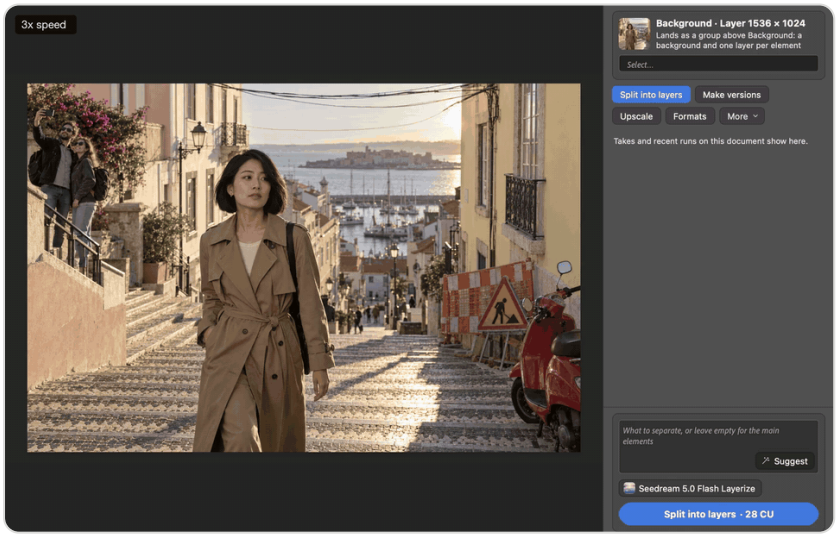


The Bolder take of the lemonade glass in Recent, with its May not line up badge, Refine and More like this

The Bolder take above carries **May not line up**: the glass reaches past the selection, as Bolder allows.

Tips:

- A feathered selection blends the edge. Leave some margin around the object.
- With Inside selection, an object taller than the selection is cut at its edge: draw the selection to the size of the new object, or use Bolder.
- A take can carry a check badge: **Edges ok**, **Edge may show** (a lighter or darker patch at the selection edge, or on Extend canvas a step or a repeated detail at the old edge), **May not line up**, **Part unchanged** or **Spills past**. The checks look at tone and alignment around the edge; they do not catch every visible patch, so look at the result.

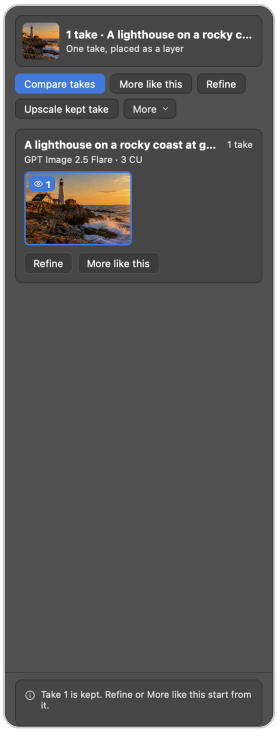


RECORDING Remove from the panel, from the selections to the clean street

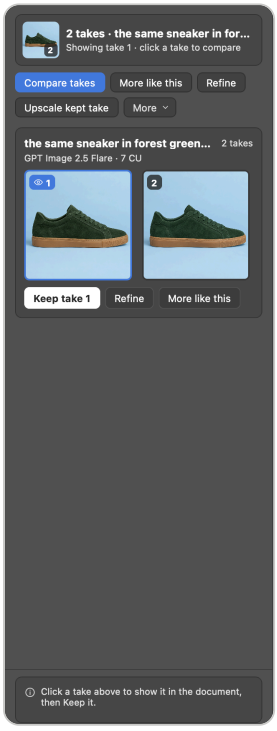
3x speed. MP4.

08 Takes

When an action makes several takes, they land as a group with take 1 visible, and the Takes card shows one tile per take. A single take lands as a layer with the same card.



One take of a lighthouse placed as a layer: Compare takes, More like this, Refine, Upscale kept take



The Takes card with two takes of a green sneaker, Keep take 1, Refine and More like this

- Click a tile to show that take in the document (one History state).
- **Keep take N** removes the other takes from the document; their files stay in the output folder and in Jobs.
- **Refine** keeps the shown take, loads its selection back and opens Replace inside on it.
- **More like this** runs again with the shown take as a reference.

- **Upscale kept take** upscales the take you kept.

Select the group in the Layers panel later, even after a restart, and the card comes back from the layers' metadata. `Verified live`.

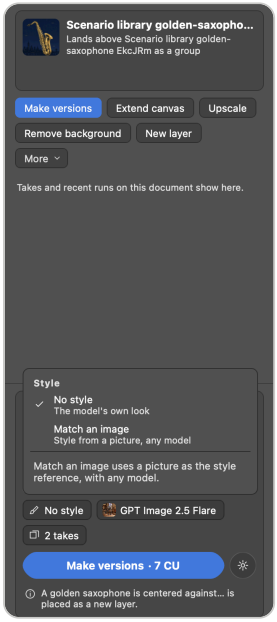
09 References

+ Reference adds an image the model uses as a reference: the selection's box, the active layer, all visible, a PNG or JPEG file, or a Library image. The hint next to it reads "Selection, layer or file". Layers pinned as document references (Character, Set, Style, Product) are kept in the document and come back with the file. References upload when you click the action button. `Verified live` (a layer reference on Replace inside, a Library reference).

On Replace inside, a reference of a whole scene may be copied into the selection: use an object or a style as the reference. The panel says so under the references row. `Tested automated`; seen live on 0.7.0, when a reference of the same scene was redrawn into the selection box.

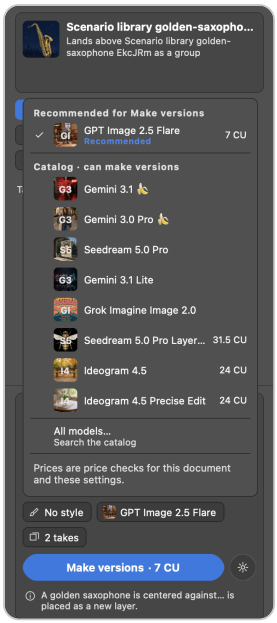
10 Style and trained models

The **Style** chip offers **No style** (the model's own look), **Match an image** (a picture goes as the style reference, with any model that takes references) and the project's trained styles.



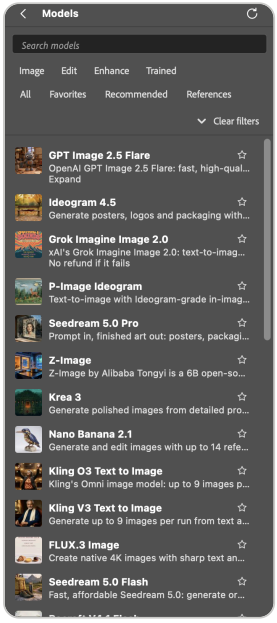
The Style menu: No style and Match an image

The **Model** chip lists the recommended model for the action and up to eight catalog models that can do it, with their prices for this document as the price checks come back (some rows can stay without a price), then **All models...**.



The Model menu: the recommended model, catalog models with prices, and All models

All models... opens the picker: search, the Image, Edit, Enhance and Trained tabs, filters (All, Favorites, Recommended, References), and a star to keep favorites.

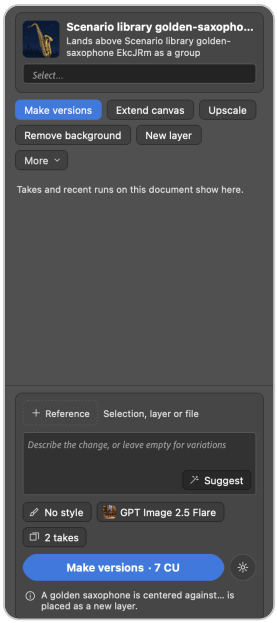


The model picker with search, tabs, filters and the model list

Trained models. Run with a trained model, the project's or a public one, runs it on its base model for the selected action, with its trigger words before your prompt. A trained style runs on the model it was trained on, and a style and a model stack on one base when it takes both. `● Verified live` (M7): Run with, Make new and a selection edit with FLUX.1 Dev and FLUX.2 Dev trained models. `○ Not tested` live: the Z-Image route and a stacked run.

11 Library

Place from Library (More > Layers) opens a grid of the project's images with a search field and the filters All, Generated, Edited and Uploaded, then **Show more**. Click an image to place it as a new layer at its own size, never enlarged, centered, in one History state. **+ Reference > Library** uses the same grid. `● Verified live`.

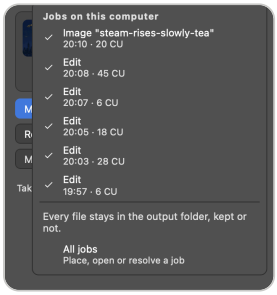


After placing a Library image: the note at the bottom says it is placed as a new layer

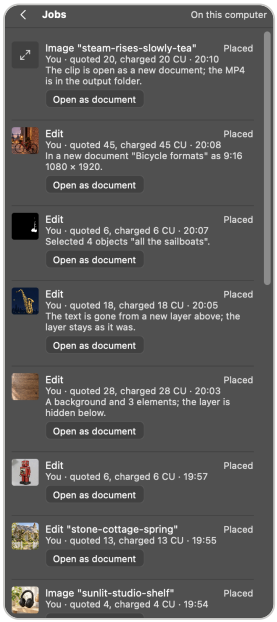
Dragging from the grid is not in this build.

12 Jobs and recovery

Jobs in the header opens the six newest jobs on this computer; **All jobs** opens the full list with each job's status, price words ("quoted 6, charged 6 CU") and its actions. `● Verified live`.



The Jobs popover with the latest jobs, their time and price



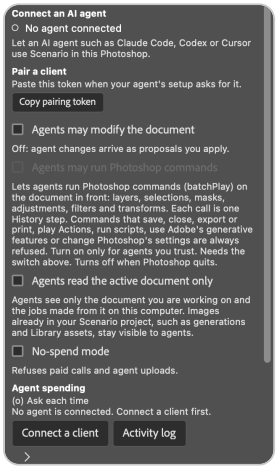
The Jobs view: each job with its picture, status, price and Open as document

- Each job is written to disk before anything is sent, so Jobs survives a restart of Photoshop. `● Verified live` (M5, six restart points).
- After a restart: a job that was uploading or waiting for approval becomes not sent (nothing charged); generating jobs resume; finished ones download again; nothing lands in a document by itself.
- A job whose request may or may not have reached Scenario (no answer, lost connection) is **Unknown**. The plugin looks it up on Scenario: **Link** continues a match; **Mark not sent** appears only when a complete look-up found none. The same request is blocked until then, so nothing is paid twice.
- **Place**, **Place in current document** and **Open as document** land a finished result where you choose. A result goes back to the document it started from, while that document is open and in front.
- Downloads can be stopped and resumed.

13 Settings

Open **Settings** from the panel menu.

- **Account**: the signed-in project, **Add a key for another project**, **Test connection**, **Sign out** (removes the key from this computer; delete it on app.scenario.com to revoke it).
- **Connect an AI agent**: link status, the pairing token, the three agent switches, No-spend mode, agent spending, **Connect a client** and the activity log. See Agents.



Settings: Connect an AI agent, pairing token, agent switches, No-spend mode and agent spending

The panel menu also has Projects, Diagnostics, Help, About and Sign Out.

14 Agents (Scenario MCP)

The Scenario plugin for Photoshop ships a small local Scenario MCP server, `scenario-photoshop`. An AI agent such as Claude Code, Codex, Cursor or Claude Desktop starts it on your computer, and through it the agent can read your open documents, price and run Scenario generations, and land the results as layers. Every credit an agent spends goes through an approval card in the panel, and every change to a document is one History step you can undo.

The server runs on this computer only. It talks to the panel over a local link (`localhost`), never over the network, and it holds no Scenario key: generations run through the plugin with the account you signed in with.

The full tool reference is Scenario MCP tool reference. Claims carry how they were checked: `● verified live` (in Photoshop 27.11 on macOS), `◐ tested automated` or `○ not tested`.

CLIENT	HOW IT WAS CHECKED
Claude Code	`● Verified live`: setup command, then real Claude Code sessions that read the document, priced, spent through the approval card and landed through the proposal card. The `claude mcp add` registration step is `◐ tested automated` (stand-in CLI); the live runs used `--config` with a temporary file
Codex	`◐ Tested headless`: the setup command through the real `codex` CLI with a temporary `CODEX_HOME`. No Codex session against Photoshop was run
Cursor	Config file writing `◐ tested automated`. Cursor itself `○ not tested` (not installed)
Claude Desktop	Config file writing `◐ tested automated`. The `.mcpb` extension `○ not tested`
Windows	`○ Not tested`

Not on npm yet. The helper is not published to npm, so the commands that read `npx -y @scenario-labs/photoshop-mcp@X.Y.Z` do not work until it is published. From a checkout of this repository, use `node helper/bin/scenario-photoshop.js` in their place and add `--local`, which registers this checkout's helper with your Node:

```
pbpaste | node helper/bin/scenario-photoshop.js setup claude-code --local --token-stdin
```

(after **Copy token** in the panel). This is the route the live runs used.

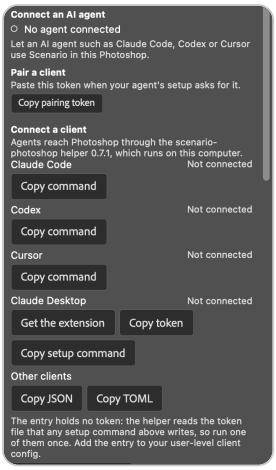
Before you start

- Photoshop 2026 (27.x) with the Scenario plugin installed and signed in.
- Node.js 20 or later for Claude Code, Codex and Cursor (`node --version`). Claude Desktop does not need it: its extension runs on the Node that ships with Claude Desktop.

- The agent client itself, installed and signed in.

Connect a client

1. In the panel, open **Settings** and go to **Connect an AI agent > Connect a client**.



Connect a client: Claude Code, Codex, Cursor and Claude Desktop rows with their buttons and status, then Other clients

2. Find your client's row and use its button (details per client below). The buttons that hold the pairing token clear the clipboard 60 seconds after copying.
3. Restart the client so it starts the helper.
4. Ask the agent something simple, such as "What document is open in Photoshop?". The row turns to **Connected, last call just now**.

Claude Code

Click **Copy command** on the Claude Code row, paste the line in a terminal and press Return. It looks like this, with your own token and the plugin's helper version:

```
SCENARIO_PHOTOSHOP_TOKEN=<token> npx -y @scenario-labs/photoshop-mcp@X.Y.Z setup cl
aude-code
```

The setup command:

- saves the pairing token in `~/.config/scenario-photoshop/token`, readable by your account only (file mode 600, folder 700);
- registers the server with `claude mcp add --scope user`, so it is available in every project. The entry it adds holds no token.

Codex

Click **Copy command** on the Codex row and run it in a terminal. It does the same as for Claude Code, with `setup codex`, and registers the server with `codex mcp add scenario-photoshop`. The entry in `~/.codex/config.toml` holds no token.

Cursor

Click **Copy command** on the Cursor row and run it in a terminal. The setup command writes the `scenario-photoshop` entry into `~/.cursor/mcp.json`:

- only that entry is replaced; the rest of the file stays as it was;

- the previous file is kept as `mcp.json.scenario-backup-<time>`;
- a new file is readable by your account only, and a rewritten file keeps its mode;
- a file it cannot parse is left untouched, and the command says so.

The entry holds no token: the helper reads the token file.

Claude Desktop

1. Click **Get the extension**. It opens the release page of `scenario-photoshop-X.Y.Z.mcpb`.
2. Download the file and double-click it. Claude Desktop installs the extension and asks for the pairing token.
3. Click **Copy token** in the panel and paste it in Claude Desktop's field. Claude Desktop keeps it as a sensitive setting.

If you prefer the token file, **Copy setup command** gives the same terminal command as the other clients, with `setup claude-desktop`. It writes the entry into Claude Desktop's config file the way it does for Cursor.

The release page and the `.mcpb` file come with the first published release; `○ not tested`.

Other clients

Copy JSON and **Copy TOML** give the server entry for a client that is not listed. Neither holds a token: run one of the setup commands above once, so the token file exists, then add the entry to your user-level client config. Do not put it in a project file such as `.mcp.json` that may be committed.

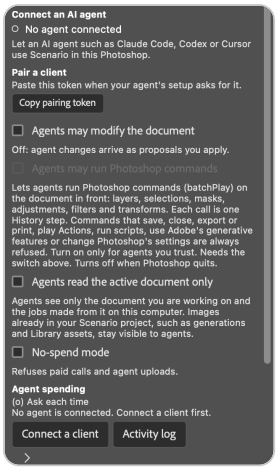
Copy skills command copies `npx skills add scenario-labs/skills`, which installs the Scenario skills for Photoshop work in your agent.

Where the token goes

- The setup command reads the token from its environment, or from standard input with `--token-stdin`, never from its arguments, which other accounts on the same computer can list.
- The pasted line stays in your shell history. To keep it out, pipe the token in instead, for example after **Copy token**: `pbpaste | npx -y @scenario-labs/photoshop-mcp@X.Y.Z setup claude-code --token-stdin`.
- Clients set up with the setup command share the token file. After **Reset** (Settings > Connect an AI agent > Details, under Pairing token), run one new setup command and every one of them reconnects.
- On Windows the buttons give the PowerShell form: `$env:SCENARIO_PHOTOSHOP_TOKEN='<token>'; npx -y @scenario-labs/photoshop-mcp@X.Y.Z setup <client>; Remove-Item Env:SCENARIO_PHOTOSHOP_TOKEN`.

What agents may do

Settings > Connect an AI agent holds three switches. All are off by default.



The agent switches, No-spend mode and Agent spending in Settings

SWITCH	OFF	ON
Agents may modify the document	Agents can read documents and run generations, but every result they want to land shows a proposal card with **Apply** and **Dismiss**	Results land at once as one History step "Scenario agent: <action>", with an **Undo** banner. The header shows "Agents can edit"
Agents may run Photoshop commands	Refused	Agents run Photoshop commands (batchPlay) on the document in front, each call one History step. Commands that save, close, export or print, play Actions, run scripts, use Adobe's generative features or change Photoshop's settings are always refused. Needs the first switch, and turns off when Photoshop quits
Agents read the active document only	Agents can read every open document	Agents see only the document you are working on and the jobs made from it on this computer

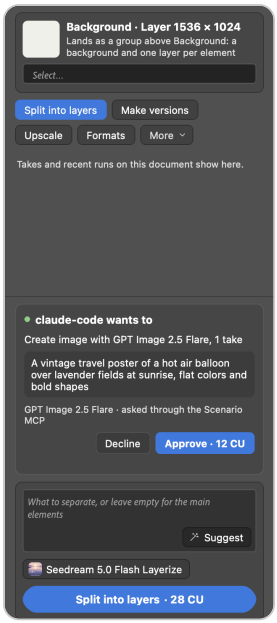
Agents change only the document in front. The plugin never brings another document forward for an agent; a proposal for another document waits until you bring it forward.

No-spend mode refuses every agent request that would spend credits, whatever the other settings say.

Approvals

Spending credits

An agent first prices a run (free), then asks to run it. The panel shows a card above the prompt bar: which client asks, what it will make with which model and how many takes, the prompt and the price on the Approve button. **Approve** runs it; **Decline** tells the agent and spends nothing. `Verified live` (approve and decline).



claude-code wants to: Create image with GPT Image 2.5 Flare, 1 take, the prompt, asked through the Scenario MCP, Decline and Approve 12 CU

The agent must send back the exact price it was quoted. If the price changed, or the quote is older than its time limit, the request is refused before anything is spent and the agent prices it again. A card left unanswered expires after 10 minutes.

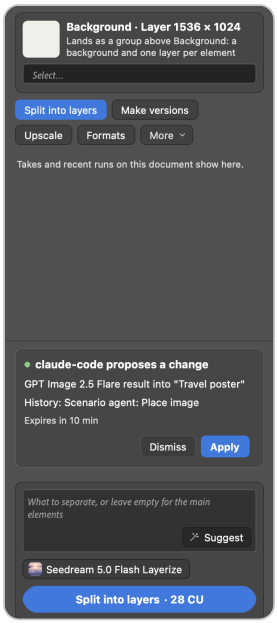
Agent spending in Settings changes this for one client session at a time: **Ask each time** (the default) or **Allow**, with a limit per call and a total for the session. **End** stops it at once. The activity log lists every agent request, approved, declined or refused.

Sending pixels to Scenario

Before an agent sends pixels from a document to Scenario for the first time in a session, the panel asks: "Allow <client> to send pixels from <document> to Scenario for this session?". Declining tells the agent, and nothing is uploaded. `◐ Tested automated`; an approved upload `● verified live` through the development loop, the card itself not seen on screen live.

Landing results

With **Agents may modify the document** off, each result the agent wants to land shows a proposal card: the client, the model and the target document, the History name and the time left. The card shows no picture of the result in this build. **Apply** lands it as one History step; **Dismiss** drops it. A proposal expires after 10 minutes or when the document changes. `● Verified live` (Apply).



claude-code proposes a change: GPT Image 2.5 Flare result into Travel poster, History Scenario agent: Place image, expires in 10 min, Dismiss and Apply

Results always land as layers with their Scenario metadata (model, prompt, job), the same as results you make yourself, and they show in **Jobs**, marked with the client's name.

Tools

GROUP	TOOLS
Document (read)	`scenario_status`, `document_summary`, `layer_detail`, `preview_document`
Scenario (read, free)	`list_models`, `model_schema`, `recommend_models`, `estimate_cost`, `job_status`, `wait_for_job`, `list_local_jobs`, `list_generations`, `list_assets`, `search_assets`, `list_collections`, `inspect_result_previews`
Inputs	`capture_reference`, `upload_reference`, `reference_upload_status`
Spending	`generate`
Recovery	`recover_local_job`, `recover_cloud_job`, `cancel_prepared_job`
Document changes (first switch)	`prepare_result_application`, `apply_result_application`, `select_layers`, `set_selection`, `remove_scenario_layers`
Photoshop commands (second switch)	`execute_batchplay`

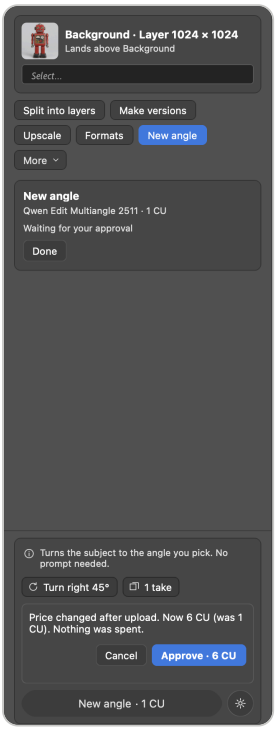
A typical edit: `document_summary`, `set_selection`, `capture_reference`, `upload_reference` (upload card), `estimate_cost`, `generate` (spend card), `wait_for_job`, `inspect_result_previews`, `prepare_result_application`, `apply_result_application`.

Troubleshooting

WHAT YOU SEE	WHAT TO DO
`NotConnected`	Open the Scenario panel in Photoshop. The helper reaches the panel only while it is open
`PairingFailed`	The token changed (for example after **Reset** under Details). Run the setup command again from Connect a client
`VersionMismatch`	The plugin and the helper versions differ. Connect a client shows the command for the new version
`SwitchOff`, or a proposal card instead of a change	Turn on **Agents may modify the document**, or use **Apply** on the card
`DocumentNotActive`	Bring the document forward in Photoshop, then ask the agent to retry
`NoSpend`	Turn off **No-spend mode** in Settings
`QuoteExpired` or `CostMismatch`	Nothing was spent. The agent prices the run again
`SpendDeclined` or `UploadDeclined`	You declined the card. Nothing was spent or sent
`npx` not found	Install Node.js 20 or later, or use Claude Desktop's extension

15 Costs

- The price is checked for free (a dry run, which creates no job) about half a second after each change, and shows on the button: exact (`12 CU`, `Free`), a range, or `from N CU`.
- An action that needs a prompt reads **Fill in the prompt to see the price** until it has one.
- **A click on a priced button is the approval.** Above 100 CU, for the top of a range and for any `from` price, a card asks first.
- The request is built again at the click. If the document, the selection or a setting changed since the price check, nothing is sent.
- If the price changes once references or captures are uploaded, the panel stops and asks again. Nothing was spent. `Verified live` (New angle: 1 CU before the upload, 6 CU after).



Price changed after upload. Now 6 CU (was 1 CU).
Nothing was spent. Cancel or Approve 6 CU

- A failed job shows "Refunded by Scenario"; a model without refunds says so before you click.
- A CU limit on your key caps what the plugin and agents can spend.

16 What leaves your machine

This page describes what the Scenario plugin for Adobe Photoshop and its local Scenario MCP server, `scenario-photoshop`, send and store. It applies to the first public release. The plugin is a client of your own Scenario account: what Scenario does with what it receives is governed by Scenario's Terms and Privacy Policy, which this page does not replace.

Status: written against the v1 specification (design specification). It is checked against the released build before each release. It sits at the root of the repository so the panel's About link points to one short path.

In the current version

This version does less than the v1 release described below. Each line is marked `● verified live` (Photoshop 27.11 on macOS), `◐ tested automated` (Node test suite with a mock of Scenario's API and a fake Photoshop) or `○ not tested`.

- **Sign in with Scenario**: the plugin asks Scenario's identity provider (`https://clerk.scenario.com`) for a sign-in code, then for tokens once you approve in your browser, then for new tokens when they expire, sending the plugin's public application id, never a password or a secret. Sign out asks it to revoke the sign-in. The tokens' answer names your account's email, shown in the panel. The plugin reads your teams and projects from Scenario's API to let you choose one. `◐ Tested automated` against a mock identity provider; `● verified live` against a development instance (see Known limitations).
- **Sent to Scenario's API** (`https://api.cloud.scenario.com`), with your key and secret as HTTP Basic, or with a browser sign-in's access token and the chosen project's id: the sign-in reads; the model list and schemas; price checks (prompt, model, settings; a blank stand-in image where a price needs one); and, when you click an action's priced button, the paid request (prompt, model, settings) and its images, uploaded through the API only, each under 4,000,000 bytes as sent. Then job status and result records while a job runs. To resolve a job left Unknown (Photoshop closed or the connection dropped while it was being sent), the plugin reads the project's jobs created from 10 minutes before the click to 60 minutes after it, and the prompt text of those jobs, to find the one that matches; it reads them only when you open that job in Jobs, ask it to check again, or mark it not sent. `◐ Tested automated`; `● verified live` for the actions listed in the User Guide.
- **Images sent with an action**, at the click and not before: the capture the action works on (`photoshop-capture-<date>-<id>.jpg` or `.png`): the selection's box plus a margin, the active layer, the document, or the grown canvas for Extend; the mask made from your selection (`photoshop-mask-<date>-<id>.png`) when the model takes one; and the references you added (`photoshop-reference-<date>-<id>`): the selection's rectangle from all visible layers, the active layer, all visible, a Library image (downloaded from your project, then sent like a file), or a PNG or JPEG file you chose with its EXIF, XMP, IPTC and text metadata removed. Captures are converted to 8-bit sRGB; no other pixels leave your document. Select by description also sends the words you typed. Frame to layer sends nothing new: it reads the clip already on Scenario. `◐ Tested automated`; `● verified live`.
- **Library**: opening the grid reads the project's asset list (names, sizes, thumbnails) from Scenario's API; the image you click downloads into `Scenario/library/<asset id>/`. `● Verified live`.
- **Downloaded** from Scenario's CDN (`https://cdn.cloud.scenario.com`, the only download host this build accepts, before and after redirects) through signed links kept in memory only, without your key or cookies. Clips download into `Scenario/video/<date>/`. `◐ Tested automated`; `● verified live`.
- **Model pictures**: the Model menu and the model picker show each model's picture from the same host's public thumbnail path (`https://cdn.cloud.scenario.com/thumbnails/<asset id>`), loaded by the panel without your key or cookies. The request tells the host which pictures the panel shows. `● Verified live`.
- **Agents**: a connected agent reads and uploads as described under "The local Scenario MCP server" below; the spend, upload and proposal cards, the switches, No-spend mode, Agent spending and the activity log are in this build. `● Verified live` with Claude Code for reads, an approved upload, the spend card and the proposal card.
- **Not in this build**: uploads to any other host than Scenario's API (so every upload stays under 4,000,000 bytes), the update check (no request to GitHub), the Output Folder setting, the Leave prompts out and Leave prices out settings (the metadata always holds the prompt and prices), and the 7-day log and capture rotation below.
- **Stored in the plugin data folder**: the model list cache, favorites, the stand-ins' asset ids, `first-run.json` (which first-run notes were shown), `scenario-link.log` (the last 200 lines; no keys, tokens, prompts or document titles), the job records in `jobs/` (prompt and settings, model, job and asset ids, prices quoted and charged, the target document's title and the position used, a 128-bit hash of the target document's file path when it was saved (so a result can find its document again after a restart; the path itself is not stored), a hash of the Scenario project id (so a job resumes only under its own project), file paths relative to the data folder), the downloaded results in `Scenario/<kind>/<date>/`, and the Edit and Enhance captures in `captures/<id>/` (the capture, the mask, and

`capture.json` with the document's title and id, the box, the sizes and the job it belongs to), kept so a result can be checked and placed again after a restart; the 7-day rotation is not in this build. Image-lane references are kept in memory until they are sent, never written to disk. `Tested automated`.

- **Development builds only**: the panel's UI loop reads `dev-ui-command.json` and writes `dev-ui-result.json` in the data folder, and its sample states write sample pictures under `dev-demo/`. A `dev-api-key` file that a developer puts in the plugin data folder is read once at startup, its key and secret go through the same Save and test as the panel, and the file is deleted whether or not Scenario accepts them. A `dev-live-run.json` file in the same folder makes a development build open a document of its own, run one lane on it, write the result and the before and after composites to `dev-live/` in the data folder, and close that document; for the restart checks it also saves that document as a PSD under `dev-live/` and opens it again after the restart. It never reads or changes the person's documents. Development builds also carry test hooks that hold a job at one point so Photoshop can be quit there; they live in memory and nothing outside the plugin can arm them. Release builds contain none of this. `Tested automated`.
- **Stored in your documents**: each result layer carries the metadata described under "Provenance in your documents", and its name starts with the first words of the prompt. `Tested automated`.

What leaves your machine, and when

WHAT	WHEN	SENT TO
Your API key and secret	With every request to Scenario's API, as HTTP Basic authentication over HTTPS. Never with downloads or uploads to other hosts	Scenario's API
A browser sign-in's access token, and the chosen project's id	With every request to Scenario's API, as a Bearer token over HTTPS (the team and project list goes without a project id). Never with downloads or uploads to other hosts	Scenario's API
A browser sign-in's refresh token and the plugin's application id	When the access token is about to expire, and at Sign out to revoke it	Scenario's identity provider
Prompt, model choice and settings	While you edit a form, for price checks (shortly after you stop typing), and when you click Generate	Scenario's API
Pixels from your document	When you click Generate, or when an agent you allowed uploads a capture, a file you chose or a saved result (below). What is sent depends on the source: the selected area plus a margin of surrounding pixels, one layer, or the visible composite. Edits also send a mask made from your selection when the model takes one. References are captured when you add them and sent when you click Generate	Scenario's API for small images (under 4 MB as sent); Scenario's upload storage for larger ones

WHAT	WHEN	SENT TO
Files you pick	When you click Generate with them as references. PNG and JPEG only, with their EXIF, XMP, IPTC and text metadata (such as GPS position, camera serial or author) removed first	As above
Catalog, Library and job lookups	When you open the model picker, the Library lane or Jobs > On Scenario, while jobs run, and once a day in the background for the model list	Scenario's API
Download requests	When a job finishes, and for thumbnails in lists	Scenario's download hosts
Update check	At most once a day, unless turned off in Settings > Updates	GitHub API
Links you click	Only when you click; Photoshop asks before opening your browser	The linked site

Price checks never upload pixels from your documents. For models whose price depends on an image input, they use a blank stand-in image, which the plugin uploads to your project and hides in the Library like its other uploads. If the exact price after the upload differs from the one shown, nothing is spent and the plugin asks you again, but the upload has already happened. Uploaded images are stored by Scenario as assets of your project, like uploads made in the Scenario web app, and the members of that project can see them.

Before sending, the plugin converts captured pixels to sRGB at 8 bits per channel. Your document's color profile is not sent. Uploads are named `photoshop-<purpose>-<date>-<short id>`; the blank stand-ins used for price checks are named `photoshop-standin-<width>x<height>.png`.

Never sent to Scenario by the plugin:

- Your PSD file, or any layer or area that was not chosen as a source.
- Document names, layer names or file paths.
- Usage analytics or telemetry. This release sends none. If a later version adds them, they will be off until you turn them on.

Never sent to anyone but Scenario's API: your key and secret. The helper, connected agents and the plugin's logs never receive them.

What the plugin keeps from the sign-in check. To show your project's name, the plugin reads the project listing from Scenario's API. That answer also lists the project's members with their email addresses and the ids of its other API keys. The plugin keeps the project name, the project id and your key's own role, and drops the rest at once: it is never stored, logged or passed to the helper.

When you connect an agent, that changes. A connected agent can read the names, layers, text and previews of every open document (only the active one, and the jobs made from it on this computer, if you turn on **Agents read the active document only**; images already in your Scenario project, such as generations and Library assets, stay

readable), and what it reads goes to the agent's model provider. After you allow it, once per document or file in each agent session, it can also upload captures from that document, files you chose and results the plugin saved to your Scenario project. See "The local Scenario MCP server" below.

Where it goes

DESTINATION	PURPOSE
`https://api.cloud.scenario.com`	Scenario's API: sign-in check, catalog, prices, uploads under 4 MB, generations, jobs, Library
`https://clerk.scenario.com`	Scenario's identity provider: Sign in with Scenario's code, tokens, refresh and revocation
Scenario's upload storage, once Scenario confirms its host as stable (it is then listed in the plugin manifest and here; until then every upload goes to Scenario's API and stays under 4 MB)	Uploads of 4 MB and more, through upload links that Scenario signs
Scenario's download hosts (listed in the plugin manifest, which never allows all hosts)	Downloads of results and thumbnails, through signed links that expire
`https://api.github.com`	Update check. GitHub receives an ordinary HTTPS request, which includes your IP address
Your browser	Pages you open from the plugin, such as the API key page on app.scenario.com, or Scenario's sign-in page
Agent clients you connect	Whatever they read through the local server (below)

The UXP plugin can reach only the hosts declared in its manifest. The local server opens only a socket on your computer's loopback interface; when your agent client starts it with `npx`, npx contacts the npm registry to fetch it. Photoshop's own handling of your data is covered by Adobe's terms, not by this page.

What Scenario does with it. Scenario runs your request with the provider of the model you chose. Scenario lists its sub-processors and model providers at trust.scenario.com. Scenario's Privacy Policy (last updated July 16, 2026) says Scenario may use your content to operate, secure and improve its services and AI models. Enterprise customers with a signed agreement get the commitments set out in that agreement instead. Read the policy for the details that apply to your plan.

Deleting your data

- **On this computer:** sign out, reset the pairing token, and delete the plugin data folder and any Output Folder you chose (paths above).

- **On Scenario:** delete uploaded images and results from your project in the Scenario web app. For anything else Scenario holds about you, including deletion of your account data, write to support@scenario.com as Scenario's Privacy Policy describes.
- **API keys:** delete them on app.scenario.com. Signing out of the plugin does not revoke a key.
- **Agent providers:** what an agent read is held under your agreement with that provider.

The local Scenario MCP server

- `scenario-photoshop` is a small program that your agent client (Claude Code, Codex, Cursor, Claude Desktop) starts. The first one listens on port 9890 of your computer's loopback interface (`127.0.0.1` and `::1`), which other computers cannot reach. The plugin connects to it. It never receives your Scenario key or secret.
- **Pairing.** The plugin and each helper prove to each other that they hold the pairing token, without sending it, and every later message carries a code that only holders of the token can produce. Web pages are refused by the origin checks. A program that does not hold the token cannot act as the plugin or as the helper; it can only take port 9890 first and block agents, which the panel reports. Messages are authenticated, not encrypted. Programs running under your own account can read the token file and client config files, so the token does not protect against them. Setup commands pass the token in an environment variable, which other standard accounts on the computer cannot list (an administrator can read any file). Agent clients that run commands on your computer (Claude Code, Codex, Cursor) run under your account: they can read the token file, change the plugin's stored settings, and may be able to read its secure storage. The switches and cards below govern what agents do through the local server; a CU limit on your API key, set on app.scenario.com, caps spending either way.
- **With all agent switches off**, a connected agent can:
  - read document information: size, mode, layers and layer names, text layers, the selection, and Scenario metadata;
  - get previews of your documents, which contain their pixels;
  - list models, get prices, and read jobs and Library items;
  - capture a selection, layer or composite, and after you allow it in the panel, once per document or file in each session, upload those captures, files you chose in the panel, or saved results to Scenario;
  - request generations, which spend your CU. By default the plugin asks you in the panel before each one. You can instead give one connected client an allowance with a limit per call and a total; prices with no upper bound still ask you; it ends when that client's session ends (it disconnects, or the agent connection restarts), when the total is used, when a charge comes in above its quote, when Photoshop quits or when you end it;
  - cancel jobs it started, and add a finished job of your project to the plugin's list.
- `SCENARIO_NO_SPEND=1` in the server's environment, or **No-spend mode** in the panel, blocks agent spending and uploads.
- Everything an agent reads goes to that agent's model provider, under your agreement with that provider.
- **Agents may modify the document**: agents can place results and change selections and layers, one undoable History step each. A selection change can run Select Subject, which follows Photoshop's Image Processing preference: if it is set to Cloud, Photoshop sends the image to Adobe for processing. With it off, their result placements arrive as proposals you apply or dismiss, and their other changes are refused.
- **Agents may run Photoshop commands** (off by default, turns off when Photoshop quits): agents can run batchPlay commands on the document in front: layers, selections, masks, adjustments, filters and transforms, one undoable History step per call. Commands outside an allow-list are refused, including any that save, close, export or print, create documents, play Actions, run scripts, use Adobe's generative features or change Photoshop's settings. Allowed commands include Select Subject (`autoCutout`), with the same note. Path values, and the document properties that can record file paths, are removed from what these commands return.

- Through the local server, agents can never read your key, secret or pairing token, save, close or export documents, resolve uncertain jobs, approve their own spending or uploads, cancel jobs they did not start, or upload files you did not choose and the plugin did not make.
- Settings > Connect an AI agent > **Activity log** lists every agent preview read, upload, spend and document change.

The website

photoshop.scenario.com counts downloads of the plugin's package. For each download of a `.ccx` from the site, it sends Scenario's analytics service (PostHog, US) one anonymous event: the version downloaded, the kind of client (browser, script, crawler or other) and the country the site's host reads from the connection. It keeps no IP address, sets no cookie and gives each event a new random identifier. The plugin itself sends no analytics.

Terms and contact

- Scenario's Privacy Policy and Terms: linked from scenario.com.
- Scenario Trust Portal: trust.scenario.com.
- Scenario documentation: docs.scenario.com.
- Privacy requests to Scenario: support@scenario.com.
- Photoshop is covered by Adobe's terms.
- The plugin is open source under the MIT license; its network calls can be checked in this repository. Report security issues privately as described in Security, and never paste keys or secrets in issues.

17 Files and folders

Every result is saved before it is placed, kept or not, in the plugin's output folder:

`~/Library/Application Support/Adobe/UXP/PluginsStorage/PHSP/27/External/com.scenario.photoshop/PluginData/Scenario/<kind>/<date>/`

Results are also in your Scenario project, so you find them again in the Library and on app.scenario.com. Each placed layer carries Scenario metadata (model, prompt, settings, job, prices) in its XMP, which survives saving the PSD.

What is stored on your machine

ITEM	WHERE	HOW TO REMOVE
API keys and secrets	Photoshop's secure storage for this plugin, one entry per project, encrypted under your operating system account. Entry names are not encrypted, so they hold a hash of the key, never the key	Settings > Account > **Sign out**. This removes the key from this computer only; delete the key on app.scenario.com to revoke it
Browser sign-in: refresh token, your account's email and the projects you chose	The same secure storage, one entry per signed-in person, named by a hash. The access token is kept in memory only	Settings > Account > **Sign out**, which also asks Scenario to end the sign-in
Pairing token	The same secure storage; the helper's token file, readable by your account only (`~/.config/scenario-photoshop/token` on macOS, `%APPDATA%\scenario-photoshop\token` on Windows); the config of clients that keep it themselves (Cursor, Claude Desktop, a manual entry); your shell history if you pasted a setup command; your clipboard after you copy it: the plugin clears it after 60 s, and clipboard managers or synced clipboards may keep a copy	Settings > Connect an AI agent > Details > **Reset** (pairing token), then delete the token file and remove the `scenario-photoshop` entry from each client

ITEM	WHERE	HOW TO REMOVE
Job records: prompts, settings, model and job ids, asset ids, prices quoted and charged, who started the job (you, or the name an agent client reported), target document name and position	The plugin data folder. On macOS: `~/Library/Application Support/Adobe/UXP/PluginsStorage/PHSP/<version>/External/<plugin id>/PluginData/`; the equivalent folder on Windows	Delete the folder. Uninstalling the plugin may not remove it
Captures sent to Scenario	The plugin's temporary folder	Deleted automatically 7 days after the job finishes
Downloaded results	The Output Folder: inside the plugin data folder by default, or the folder you chose	Delete the files; placed results are embedded in your documents and do not depend on them
Copy of the job records	`<Output Folder>/.scenario/jobs/`, only when you chose an Output Folder. It holds prompts and document names, and travels with the folder if the folder is synced or shared	Delete the folder
Logs	The plugin data folder, `logs/`: kept 7 days, at most 5 MB. Keys, secrets, tokens and request and response bodies are never written	Delete the folder
Model list cache, favorite models, the asset ids of the blank stand-ins uploaded for price checks, settings, agent activity log	The plugin data folder	Delete the folder
Provenance in your documents	Metadata on each result layer: model, prompt, settings, asset and job ids, prices, date, and who started it. Layer names include the first words of the prompt	Delete the layer. For future results, Settings > Results > **Leave prompts out of documents** removes the prompt, every free-text setting and the prompt words in layer names; **Leave prices out of documents** removes the prices

Signed download and upload links are kept in memory only, never on disk.

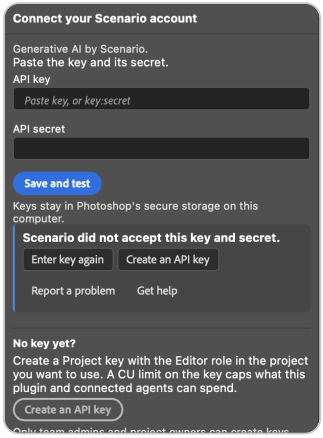
Layer metadata travels with the PSD when you share it. It never contains your key, secret, pairing token, team or project ids, or file paths. The Scenario skills for Photoshop write the same metadata, so the panel can read layers placed either way.

Copy diagnostics (Settings > Diagnostics) shows its text before copying it. It holds versions, your operating system, switch states, job counts and recent error types, and no prompts, document or project names, ids, paths, keys or tokens.

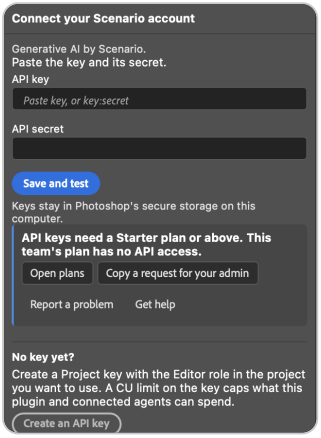
18 Troubleshooting

WHAT YOU SEE	WHAT TO DO
The panel is empty on Photoshop's Home screen	Open or create a document. Photoshop hides docked panels on Home
"Scenario did not accept this key and secret."	Check you copied both, from a Project key. **Enter key again** or **Create an API key**
"The sign-in was declined in the browser." or "The code expired before the sign-in finished."	**Sign in again**, and approve in the browser
"This account has no project it can use."	Create a project on app.scenario.com, or ask a team admin for a role in one, then **Sign in again**
"The chosen project is no longer available. Choose another one."	The project was deleted, or your role in it removed. **Choose a project**
"Your Scenario sign-in ended. Sign in again."	Scenario no longer accepts the saved sign-in (signed out elsewhere, or revoked). **Sign in again**
"API keys need a Starter plan or above. This team's plan has no API access."	Use a team on a plan with API access, or **Open plans**. **Copy a request for your admin** prepares the message
A Team (Organization) key or a Reader key	Use a Project key with the Editor role
Fill in the prompt to see the price	The action needs a prompt
A card says the price changed	Nothing was spent. **Approve** at the new price or **Cancel**
A job is Unknown	Open Jobs and let it look the job up; **Link** or **Mark not sent**

WHAT YOU SEE	WHAT TO DO
A result did not land	Bring the document it started from to the front, then **Place** in Jobs, or **Open as document**
"No connection to Scenario. Check your network."	Check the network, then **Retry**. A job that was being sent when the connection dropped shows as **Unknown** in Jobs until it is looked up
An agent says `NotConnected`	Open the Scenario panel; the helper reaches it only while it is open



Sign-in error: Scenario did not accept this key and secret, with Enter key again and Create an API key



Sign-in error: API keys need a Starter plan or above, with Open plans and Copy a request for your admin

Drawn from the panel's own failure copy; the errors themselves were not produced live in this pass.

Copy diagnostics (Settings, or **Diagnostics...** in the panel menu) shows a short report before it copies it, for a bug report: versions, switch states, job counts and the last error types, with no prompts, names, ids, paths, keys or tokens. `Tested automated`. See Security for security issues.

19 Known limits

This page lists what the plugin does not do yet, what it does imperfectly, and what was not tested. The evidence: the Node test suite (493 tests, with a mock of Scenario's API and a fake Photoshop: `tested automated`), and live passes in Photoshop 27.11 on macOS, one paid run per action at its default model, with the panel docked at 320 px (`verified live`). Everything the live passes did not reach is marked `not tested`. Each part of the plugin and how it was checked: build status.

Not in this build

- **Start a set** (several images from one brief), **Same action on each** (one action over several layers or documents) and **Run a workflow** (Scenario workflows) are not built.
- **Drag and drop**: onto the panel, or from the Library grid into a document.
- **Update check**, the **Output Folder** setting, and the **Leave prompts out** and **Leave prices out** settings described in Privacy. Results always go to the plugin data folder, and layer metadata always holds the prompt and prices.
- **Large uploads**: every upload goes through Scenario's API and stays under 4,000,000 bytes as sent. A capture over the limit is reduced in 10% steps; when the result's size follows the capture (Upscale, for example), the panel stops first and shows the new size and price. A reference file still over 4 MB without its metadata is refused.
- **Log and capture rotation** after 7 days.
- **The agent's plan card** is built, but the agent link does not send plans yet.
- **Not on the Adobe Marketplace, and the helper is not on npm.** You install the `.ccx` package of a release, from photoshop.scenario.com/download/latest, or build it from this repository; agent clients run the helper from a checkout (see Agents (Scenario MCP)).

Runtime and authentication

- **Windows**: built for it, `not tested`.
- **Photoshop versions** other than 27.11: `not tested`.
- **The panel is hidden on Photoshop's Home screen.** Photoshop hides docked panels there, so the panel shows once a document is open. The panel's own "no document" screen was never seen live.
- **Document events while Photoshop is in the background.** With Photoshop in the background, its document events did not reach the panel, so the context card can show the previous document or selection until the panel reads the document again. Documents and changes made by a script (AppleScript or ExtendScript) send no event either: the panel catches up at the person's next click on a document tab, a menu item or a layer (`verified live`). The agent tools read Photoshop at call time: `scenario_status` listed a document made by ExtendScript at once (`verified live`, 2026-10-23).
- **Sign-in**: pasting `key:secret` into the key field and **Save and test** are `verified live` with the release package (2026-10-23). Typing the key and secret by hand, and the error screens for a refused key or for a plan without API access, were `not tested` live. The pictures of those screens in the User Guide are drawn from the panel's own failure copy.
- **Sign in with Scenario**: `tested automated` against a mock identity provider and a mock API (two teams of two projects and a team without one). Against Scenario's development identity provider and the staging API, the plugin's own sign-in modules under Node signed in with the code approved in a browser, listed two teams of two projects each, used a project, passed Test connection (a Bearer read with `projectId`) and signed out with the token revoked (2026-11-03). In Photoshop 27.11 on macOS (development build against the same services, 2026-11-03), `verified live`: Photoshop showed its Request For Permission dialog for the page the first time, and later sign-ins, also after a restart, opened it in the default browser without asking; the approval page went straight to Approve and Deny without showing the code; the chooser listed both teams with their projects (Scenario's order changed between two reads, so the panel now sorts by name); a chosen project priced and ran

one Create image (Z-Image Turbo, 4 takes, 4 CU) whose takes downloaded and landed as a group; Choose another project switched and added the project's row; Test connection passed, and passed again after quitting and restarting Photoshop (the session refreshed from secure storage); Deny on the page showed "The sign-in was declined in the browser." with **Sign in again**; Sign out left no project saved. Not tested live: **Block** in Photoshop's dialog, a code that expires, `slow_down`, and a refresh refused with `invalid_grant`. A release needs the device grant on Scenario's production Photoshop application and that application accepted by the production API. Windows: ○ `not tested`.

- **Release package and first-run time** are ● `verified live` on macOS (2026-10-23): 1 min 53 s from install to a first image, with the clicks done by a script. Installing by double-clicking the `.ccx`, and installing on Windows, were ○ `not tested`.
- **One Photoshop at a time.** Two Photoshop instances on one computer sharing the helper's port: ○ `not tested`.

Creation and application

- **Color-managed documents**, one live run (2026-10-23): Upscale on an Adobe RGB (1998) 16-bit document kept its colors (RMSE 0.7% against the source, both converted to sRGB to compare). The upscaled document opens in sRGB, 8-bit. CMYK, Lab, other profiles and the other actions on such documents were ○ `not tested`.
- **Large documents**, one live run: Create image on an 8192 × 4608 canvas landed its four takes at canvas size as one History state. The model's picture is 2048 × 1152, scaled up four times to fill the canvas, and the panel does not say so; **Upscale kept take** adds the detail.
- **Undo and Redo**: Undo removes a landed run in one step and Redo brings it back; the panel follows both (● `verified live` on Create image). Undo of the other actions was ○ `not tested` live.
- **Extend canvas can show seams.** In two live runs on 0.7.1 (16:9 and 3:2), the original image was placed over a new frame the model had drawn again, and the joins showed: a doubled sun and a step where the old edge was. The seam check did not flag them. Changed after 0.7.1 (◐ `tested automated` on synthetic pictures that reproduce both): the new frame shows through an eased band inside the original (6% of its short side, at least the overlap); on models that take a mask, its tone along the old edge is matched to the original's (a `-landed.png` copy lands, the downloaded take is kept); past the band the original is untouched. The checks read across the old edge and give **Step at the old edge** and **Repeated detail past the old edge**. ● `Verified live: still broken` (2026-10-21, development build, a 2:3 poster extended to 16:9 with the default model, one take): the horizon steps at the right old edge and the model drew the main figure a second time in the new area on the left. The checks found both (the job record holds Step at the old edge, Repeated detail past the old edge and May be misaligned), but the panel did not show them: Extend canvas lands as a single result card, which reads "Placed" with the tone-match note and no badge. Changed after that re-test: Extend canvas now defaults to **Reframe** at 2K, the model that redraws the whole frame around the original. In a comparison on the same poster and the same 16:9 canvas (two takes per model, generated through Scenario outside Photoshop on the canvas and mask the plugin builds, judged at 100%), both Reframe takes continued the scene with no second figure and no step, where GPT Image 2.5 Flare drew the figure twice in 3 of 4 takes and GPT Image 2.5 Sunburst, the runner-up, left a horizon step at the old edge in every take. Asking the model in the prompt not to repeat anything from the center did not help on any of them, so the built-in prompt is unchanged. The single result card now shows the badge and every check in words (◐ `tested automated`). ● `Verified live` (2026-10-21 evening, development build at 137fd42, the same poster to 16:9 with the new default, one take): one figure and one goat, no step at either old edge at 100%, and the single result card read **Edges ok**, as the record holds (Seam ok, step 12.5 under the limit of 22). A feature the model draws twice is flagged, not removed. Look at the result at 100% before keeping it, or extend in smaller steps.
- **Extend canvas on the default model: no "+20% each side".** Reframe grows the canvas to a ratio with the original at one of nine positions; it has no per-side growth, so the Extend to menu hides **+20% each side** for it. Pick GPT Image 2.5 Flare in the Model menu for per-side growth (its mask route, with the seams described above).

- **The step check is a weak signal.** Calibrated on the 14 takes of the comparison, labelled by eye (`◐ tested automated`, scores in `test/fixtures/extend-edges/scores.json`): the step score is mostly the picture's own texture and does not separate the labels by itself, since stepped mask-route takes scored 0 to 11.7 and the clean takes 11.2 and 12.2. Its limit is now 22 (it was 10, which flagged both clean takes), which still flags a pale band past the edge (42). Horizon steps on mask-route takes are caught by the band residual instead (9.1 and up, limit 6), shown as **May not line up**; the clean takes read 5.3 there, the thinnest margin in the data. The repeat check separates as it was: clean takes 0 and 1 blocks, a figure drawn twice 3 blocks and up, limit 2. One picture only: other pictures may score otherwise.
- **The resolution note under-reads Reframe.** The note "New area at about N% of this document's resolution" reads 2K as 2048 px on the long side; Reframe's 2K at 16:9 came back 2752 x 1536, more than the note says.
- **Inside selection can leave a lighter patch.** Replace inside with Inside selection and a feathered elliptical selection left a lighter oval on the surface around the new object in two of three live takes on 0.7.1 (the new glass's reflection, cut by the selection), and the take showed no badge: the check measured a ring outside the selection, which a masked edit leaves as the original, and the Takes card had no badge for the tone change it did report. Fixed after 0.7.1, `◐ tested automated`: the check now also reads a ring just inside the selection and gives **Lighter patch at the selection edge** or **Darker patch at the selection edge** ("Edge may show" on the take). Measured on the four saved live takes of that scene, it flags three, including both rejected ones. `● Verified live` on 2026-10-21 (development build, a feathered ellipse on a sky, two takes with the default model): no lighter or darker patch in either take at 100%, and both read **Edges ok**, which matches the crops. Since no patch appeared, the new badge firing on a real patch was not seen live; that part stays `◐ tested automated`. The patch itself is still there: the 2 px feather stays, since the patch comes from what the model drew. An object taller than the selection is cut at its edge. Draw the selection to the size of the new object, or use **Bolder**.
- **The edge checks do not catch every visible patch.** The badges (Edges ok, Edge may show, May not line up, May be misaligned, Part unchanged) look at tone and alignment around the edge. Treat "Edges ok" as a hint, not a guarantee.
- **New layer's cut out depends on Photoshop's Remove Background** when the model ignores the transparent background (GPT Image 2.5 Flare did, live 2026-10-23). Remove Background picks the main subject: with several elements, or a subject it does not see, the mask can keep too much or too little; paint on the mask or choose **Full frame**. Remove Background follows Photoshop's Image Processing preference (on device or Cloud), `○ not tested` on Cloud.
- **Formats titles** are `● verified live` on 0.7.1 (2026-10-23): four artboards in one document, each live title inside its artboard at its place in the frame. Text drawn into the pixels (not a type layer) is reframed with the picture and can still be cut. Smart Reframe can crop a subject at the frame's edge (a kite cut at the top in 4:5 and 1:1); the optional prompt ("what must stay in frame") kept it whole in a re-run.
- **A reference of the same scene can be copied into the selection** on Replace inside. The panel warns under the references row; `◐ tested automated` (the note), seen live on 0.7.0.
- **A price can change after the upload.** For models whose price depends on the image, the exact price is known only once the capture is uploaded (New angle: 1 CU before, 5 to 14 CU after, live). Nothing is spent: the panel asks again. The upload itself has happened by then. Changed after the 2026-10-21 live session, `◐ tested automated`: New angle is now priced before the upload with a blank stand-in at the capture's size. `● Verified live` (2026-10-21 evening, development build at 137fd42): New angle on a 1024 x 1536 layer quoted 17 CU before the upload and was charged 17, with no approval card.
- **Trained models**: the Z-Image route and a run stacking a trained style and a trained model were `○ not tested` live.
- **Relight to scene** was `● verified live` on 0.7.1 (2026-10-23, a split figure relit for dusk). The model can redraw details: the first prompt gave a near-silhouette without her goggles and coat patches; naming what must stay ("keep every detail: the goggles, the patched coat...") kept them. A take can still change a detail (one added a satchel, another changed her hair color): compare the takes before Keep.

- **Split into layers can misplace a small element without flagging it.** Seedream Layerize returns each element redrawn larger and with no position; the plugin finds it by its colors. A small element redrawn differently from the picture (a rowboat redrawn from three tiny boats in a harbor) landed on another element's place (the keeper's chest) with no "may sit off their place" note (● verified live, 2026-10-23). Check each element layer after a split and move it if needed. A fully opaque element (a sky) is now placed across the picture (● verified live); it was placed as a small patch before.
- **Split layers are named Element 1, 2, ...**: the model returns them in its own order, not the order of the words in the prompt.
- **Select Subject with Photoshop's Image Processing preference set to Cloud** (when an agent changes a selection): Photoshop sends the image to Adobe. ○ Not tested.

Interface and capture

- **The run card could keep the first price.** On New angle (0.7.1, live), the card kept "1 CU" after the price changed to 6 CU and the job was charged 6. Fixed after 0.7.1: the card shows the charged price, else the price waiting for approval, else the quote. ● Verified live: fixed (2026-10-21, development build): on New angle the price went from 1 CU to 14 CU after the upload; the run card read 14 CU while waiting for approval, and the Takes card read the charged 14 CU after the run. The composer button under the approval card still read the first quote ("New angle · 1 CU") while the approval card showed 14 CU and during the run. Fixed after that re-test, ◐ tested automated, ○ not tested live: the button shows the price after the upload for the same action and settings; any change of settings goes back to the price check. In the evening check the price did not change after the upload, so this case did not come up. After a New angle run the Takes card is titled "Edit" and the context "2 takes · Edit", not "New angle" (live, 2026-10-21). A second run with the same settings is still quoted with the stand-in and asks for approval again after its upload.
- **Jobs labeled a video job "Image"** in the popover and in Jobs (0.7.1, live). Fixed after 0.7.1: Animate and Frame to layer jobs are named "Video". ● Verified live: fixed (2026-10-21, development build): an Animate 5 s job read "Video" in the popover and in Jobs, while generating and after the clip opened. Frame to layer was not run.
- **Photoshop busy.** While an agent drove the plugin, docked panels could not be resized and the panel showed Photoshop's raw "host is in a modal state"; the first **Keep take** right after a placement did nothing (2026-10-21, live). The audit found no modal scope holding a network wait, upload, poll, download or price check (◐ tested automated); the plugin's own scopes could collide, such as the context thumbnail read and Keep. Changed after that session, ◐ tested automated: the plugin's scopes queue one after the other, a scope refused as busy is tried again three times (about 1 s), then the panel reads "Photoshop is busy. Try again in a moment." Each scope still takes Photoshop's modal state for its own read or change, so a resize during a placement waits for it. ● Verified live (2026-10-21 evening, development build at 137fd42): **Keep take** sent about 3 s after a two-take Replace inside landed, then **Refine** 3 s later: Keep worked on the first command, and neither the panel nor the log showed the host's text or the busy line. The dev loop sends one command about every 3 s, so a click in the first second after a placement was not reproduced.
- **The second chip row could be cut** under the composer on a layer card with its Select field at about 300 px (2026-10-21, live). Changed, ◐ tested headless in Chrome: the context card and chips keep their full height and the button stays in view; on a short panel the last chips fold into **More** (down to one row), so an action can be one click further. ● Verified live: not fixed in every case (2026-10-21 evening, development build at 137fd42, panel docked at 280 x 420): the selection and takes contexts keep every row whole with the button in view; a plain layer card keeps its rows, but its top overflows by about 4 px and shows a scrollbar; with an Extend canvas result layer active, the context detail wraps to two lines, the third chip row (New layer, More) is cut in half, nothing folds into More, and the result card is pushed out of view. The composer and its button stayed in view in every case. Fixed after that check (UXP gives the chips no size, so the rows could not be counted): the fold runs again once the layout has settled. ● Verified live through the dev loop (280 x 420, a layer card with a long name and

three chip rows before): two chips went into More, both rows whole, the composer and button in view. The Extend canvas result layer itself was not re-run, and the fold may put one chip more into More than needed while Photoshop is in the background.

- **The context card named take layers by ids** ("Take 1 NfpUm8 · Layer 1536..."). Fixed, `◐ tested automated`: "Take 1 of <run name>", the id tail and size in the tooltip. `● Verified live in part` (2026-10-21 evening, development build at 137fd42): an Extend canvas result is titled "Extend canvas"; the detail line still names the raw layer ("Lands above Scenario expand extend-canvas wVaaV4 as a group"), and a selection on a result group names the raw group ("Selection 400 × 240 on Scenario edit flock-paper-lanterns (2 takes)", a name that keeps " (2 takes)" after Keep).
- **The agent's proposal card shows no picture** of the result: it names the client, the model, the target document and the History step.
- **Library thumbnails** of recent assets can stay gray for a long time or not load, and the grid can leave its third column empty at 320 px. Search matches asset names, which are often not the words in the prompt ("sneaker" found nothing in a project holding a sneaker image).
- **Model menu**: some catalog rows show no price until their price check comes back, and some stay without one. Models the plugin cannot give the layer to (on 2026-10-22: Gemini 3.1, Gemini 3.0 Pro, Gemini 3.1 Lite, Seedream 5.0 Pro, Grok Imagine Image 2.0) are left out of the Model menu of every action that sends the layer or a selection, and All models refuses them with a note. Create image is not affected.
- **Real-click coverage** (2026-10-22, real mouse events, Photoshop in front): `● verified live` on the flat image context (every action chip and every row of the Model, Style, Takes, Size, Format and Advanced menus and of More), the header, Settings (Test connection, Sign out, the pairing token, the agent switches, Activity log, Connect a client, Diagnostics, links), adding a project, Projects and the panel menu. The rest was pressed through the development build's dev loop, which calls each control's handler from code rather than with the mouse (*tested through the dev loop*): every action chip and menu row of the no document, layer, cut-out and selection contexts, All models (its four tabs), Jobs, the Library (filters, Show more, opening a tile), and one paid run per action at 1 take (Make versions, More like this, Upscale kept take, Extend canvas, Upscale, Remove background, New layer, Split into layers, Render sketch, New angle, Relight, Remove text, Split text, Replace and Remove inside a selection, Cut out, Add background), each landing in the document and charged its quote. `○ Not tested` in this pass: Formats and Animate (over the pass's 40 CU cap per run; both ran live on 2026-10-21), Create image, references, Match an image, Suggest, typing an API key or a search with real keys.
- **Links to a new site**: the first link the panel opens to a site (GitHub, the help pages) shows Photoshop's own Request For Permission card, and the panel waits until it is answered. `● Verified live`.
- **After a Photoshop relaunch** the panel shows hover states only after a first click in it. `● Verified live`.
- **Opening a clip or a new document can rearrange the panel layout.** After Animate or Formats opened a document, Photoshop moved other docked panels above the Scenario panel in one session.
- **Offline**: the development sample of the offline state did not show in the main view. Losing the connection during a run was `○ not tested` live; the recovery it leads to (an Unknown job looked up on Scenario) is `◐ tested automated`.
- **Agent model lists** (`● verified live`, 2026-10-23). `list_models` fills `accepts_mask`, `max_references` and `max_output` from a model's schema only after `model_schema` has read it in the session: before that, an inpainting model that takes a mask read `accepts_mask: false`. The `edit` lane also lists models whose schema has no input for the capture (`capture_input: null`), which the panel leaves out of its Model menus. `recommend_models` relays Scenario's ranking: for an object swap in the `edit` lane it put a material generator and a layer splitter first. Agents should choose from `model_schema`.
- **Agent progress**: an agent's `wait_for_job` call returns once the job moves on or after its timeout (45 s by default, 170 s at most). While the call is held, the client gets no progress updates.

Verification

- **Agent clients**: Claude Code was `verified live` (headless sessions through the helper over stdio, with the spend card and the proposal card approved in the panel). Codex was `tested headless` (the setup command through the real `codex` CLI), with no session against Photoshop. Cursor and Claude Desktop: config writing `tested automated`; the clients themselves `not tested`, nor the Claude Desktop `.mcpb` extension.
- **Two agent clients at once** on one Photoshop: `not tested`.
- **The upload card** (before an agent first sends pixels) was not seen on screen live: the live agent session that uploaded (M6) approved it through the development loop while a stylesheet rule hid the card, fixed since.
- **Install by double-clicking the `.ccx`** through Creative Cloud: `not tested` (the live passes installed with Creative Cloud's command-line installer).
- **Recovery after a restart** was verified live on 0.5.0 (six restart points), not repeated on 0.7.1. The look-up of Unknown jobs is `tested automated`.
- **Widths**: the live passes on 0.7.x docked the panel at 320 px. Other widths were `not tested` on 0.7.1.

20 Security

Reporting a vulnerability

Do not report security vulnerabilities through public GitHub issues, discussions or pull requests: everything posted there is public and permanent.

Report privately instead, through either channel:

- Email support@scenario.com with "security" in the subject line.
- Use in-app support at app.scenario.com.

Include what an investigation needs: the affected part (the Photoshop plugin, the local `scenario-photoshop` helper, or the Scenario platform), the plugin version (Settings > About), the Photoshop version and operating system, reproduction steps, and the impact you see. **Copy diagnostics** in Settings gives the versions and recent error types without prompts, names, paths, keys or tokens. Scenario support routes security reports to the right team and follows up through the same channel.

Supported versions

Security fixes go into the next release, published on the releases page. Only the latest release is supported. Development builds (`-dev.ccx`, built from a checkout) are not.

Scope

Reports that belong here include:

- the plugin or the helper sending your Scenario key or secret anywhere other than Scenario's API, or writing it to a file, log, layer or clipboard;
- the pairing token leaking (into a client config, a process argument, a log), or a program that does not hold it acting as the plugin or as the helper;
- an agent getting past the panel's switches or cards: spending credits, uploading pixels or changing a document without the approval this build promises (see Agents (Scenario MCP));
- a Photoshop command (batchPlay) outside the allow-list being run for an agent, such as one that saves, closes, exports, plays Actions or runs scripts;
- the plugin reaching a host not listed in Privacy, or a download accepted from a host other than Scenario's CDN;
- a web page or another computer reaching the helper's local port;
- a supply-chain concern with the repository's dependencies, build or package.

Vulnerabilities in the Scenario platform itself (app.scenario.com, the public API, the Scenario MCP server at mcp.scenario.com) go through the same private channels, not through this repository's issue tracker. Problems in Photoshop or UXP go to Adobe.

What the design relies on

These are the limits of what the plugin can protect, written out so a report can say which one failed:

- **Your key** stays in Photoshop's secure storage for this plugin and goes only to `https://api.cloud.scenario.com` over HTTPS. A browser sign-in's refresh token stays in the same storage and goes only to Scenario's identity provider; its access token stays in memory and goes only to the API. The helper and agents never receive any of them. `◐ Tested automated`.

- **The local link** listens on the loopback interface only (port 9890). The plugin and each helper prove they hold the pairing token without sending it, and every message carries a code only token holders can produce. Messages are authenticated, not encrypted. `Tested automated`; pairing `verified live`.
- **Programs running under your own account** can read the token file and client configs. Agent clients that run commands on your computer (Claude Code, Codex, Cursor) run under your account. The switches and cards govern what agents do through the helper; they cannot stop a program that already runs as you. A CU limit on your API key, set on app.scenario.com, caps spending either way.
- **Spending** needs the exact quoted price: a changed price or an expired quote is refused before anything is sent, and a price that changes after an upload asks again. `Tested automated`; `verified live` in the panel and through an agent.

Secrets in issues and pull requests

Never paste API keys, secrets, the pairing token or signed asset links (they are credentials) into issues or pull requests. If you spot a secret already exposed in this repository, report it privately rather than pointing to it in a public issue.

21 Changelog

All notable changes to Scenario for Photoshop. Entries above the "Hand-written history" heading are generated by release-please from the Conventional Commit titles in each release pull request and are never edited by hand. Versions follow Semantic Versioning and match `plugin/manifest.json`. Each release has a `scenario-photoshop-<version>.ccx` on the releases page.

0.8.0 (2026-11-10)

Features

- **auth:** sign in with Scenario in the browser, then choose a team's project (#7) (e9223d6)

Bug Fixes

- **deps:** Bump ws in / (#10) (2eb2981)

Documentation

- **docs:** describe Sign in with Scenario on photoshop.scenario.com (#11) (25d1c16)

Miscellaneous Chores

- **deps:** Bump the commitlint group (a4fbf4f)
- **deps:** Bump the commitlint group in /tools/commitlint with 2 updates (#9) (a4fbf4f)

Continuous Integration

- **ci:** add the ci-ok job the main rulesets require, as on blender-plugin and godot-plugin (#15) (75351cd)
- **ci:** count package downloads on photoshop.scenario.com (e4df1f6)
- **ci:** count site visits with Vercel Web Analytics, without cookies (25d1c16)
- **ci:** publish the site after each release and on site changes (#12) (3adb528)

- **ci:** serve photoshop.scenario.com through a Vercel proxy (e0c4ecf)
- **deps:** Bump the actions group with 2 updates (#8) (0e59e07)
- **handbook:** publish the handbook with the site under /handbook/ (#13) (3d2cfdc)
- **release:** build and publish releases with release-please (d10689f)
- **release:** publish each release's notes to www.scenario.com/changelog (#20) (91e9b90)

[Hand-written history]

Written by hand before release-please, in the Keep a Changelog format. Every version below is a development build, and the 0.3, 0.4.0 and 0.6 builds are folded into the next entry. The `photoshop-plugin-v0.7.1` tag marks commit 3132c14 (9 November 2026), which holds everything under "After the 0.7.1 build" except Sign in with Scenario: release-please lists that in the first release above.

After the 0.7.1 build

Added

- The prompt box's height: drag the grip on the composer's top edge, as between two Photoshop panels, to make the prompt taller; double-click the grip to go back. The takes area above keeps at least 64 px, and the height is kept on this computer. `◐ Tested automated`; the grip checked in place live on Photoshop 27.11, the drag not yet pressed live.
- **Sign in with Scenario**: the panel signs in with your Scenario account in the browser (the OAuth device sign-in, no key to create), then lists your teams and projects to choose one. Requests carry the chosen project; Projects can choose another; Test connection says when the chosen project is gone; Sign out asks Scenario to end the sign-in. The refresh token stays in Photoshop's secure storage. API keys keep working next to it. `◐ Tested automated`; `● verified live` in Photoshop 27.11 against Scenario's development sign-in and staging API.

Fixed

Each fix is `◐ tested automated` (Node suite, fake Photoshop, synthetic pictures). Re-tested live on 2026-10-21 (development build, one run each at the default model); the result ends each line.

- Panel: the icon of **Add a key for another project** (Projects and Settings), **Choose another project** and the Agents **Details** toggle sat on a line of its own above the label. It is now the button's icon, on the label's line. *Tested live* (Photoshop 27.11, development build).
- Agent tools: `job_status.error` named panel buttons an agent does not have ("Use Place unmasked or Open as document" after a canvas change, "Resume download", "Resolve it in Jobs"). It now names the landing purpose or the tool (`new_document`, `recover_local_job`). `◐ Tested automated`.
- Agent tools: with **Agents may modify the document** off, the proposal card did not say that a take would land unmasked (after a panel reload) or where an expand would grow the canvas, so the person approved without the facts the agent had. The card and the expand review now say both. `◐ Tested automated`.
- Agent tools: `inspect_result_previews` returned `seam_ok: false` for a take the model redrew around the edit, the same reading as a misaligned take, so an agent dropped good paid takes the panel shows as "Re-rendered around the edit". Each take now carries `shift_px` and `badges`, the checks in words. `◐ Tested automated`.
- Agent tools: `capture_reference`'s expand anchors `left`, `right` and the four corners fell back to `center`, so "original at the left" grew both sides and the result landed where the person did not ask. They now reach the shape code under its own names, and a model that sizes the canvas itself (`exact_size`) refuses an anchor other than `center`. `◐ Tested automated`.

- Agent tools: an edit landed after a panel reload or a Photoshop restart, when the selection mask is no longer in memory, went in unmasked over the capture box with no word before Apply. `prepare_result_application` now warns `lands_unmasked` and says so in its review. `◐ Tested automated`.
- Agent tools: after a restart, an edit whose unsaved document had closed landed over the document in front, unmasked, over its whole canvas. `edit_selection`, `expand` and `cutout_mask` now refuse with `DocumentClosed` and point to `new_document`. `◐ Tested automated`.
- Agent tools: `apply_result_application` refused with `PhotoshopBusy` (or another error raised before anything was placed) marked its `application_id` used, so calling it again as the message says returned `ApplicationUsed` with nothing placed. The same `application_id` now applies. `◐ Tested automated`.
- Agent tools: the `recommend_models` description offered a prompt the ranking does not read; it now says so (description only).
- Agent tools: after the canvas size changed since an edit's capture, `DocumentChanged` named `new_layer` as the way to still land the result, and `new_layer` refused the same way. It now names `new_document`, which lands it. The `SwitchOff` message names Settings > Connect an AI agent, the section's name in the panel. `◐ Tested automated`.
- Keep take and Show take touch only the group's own takes (Scenario layers of the group's job): a layer the person added to the takes group, also a Scenario layer of another run dragged into it or a copy the person made of a take (Layer > Duplicate keeps the take's metadata), is no longer deleted by Keep take (and Refine, which keeps the shown take first) or hidden by Show take, and the Takes card does not count it as a take. Before, every other layer of the group went. `◐ Tested automated`.
- No button in the panel answered a click (live 2026-10-21 evening, on the build with the chip fold): the panel redrew itself dozens of times a second, since Photoshop sends "resize" when the layout inside changes and each redraw changed the layout, so every button was replaced under the pointer and Photoshop ran at about 150% CPU with no document. The panel now redraws on "resize" only when its size changed, and the top's class is set only when it differs. `● Verified live`: 3 redraws after a reload, then none while idle, CPU back to about 4%; `◐ tested automated` (a resize with the same size draws nothing).
- A price check's quote is given back when the job record cannot be written, so the same button can send it once the disk works; it read "This quote was already used" until a new price check. `◐ Tested automated`.
- Allow without asking: when the request that used up the session total is never sent, its amount comes back and the allowance resumes; it was lost with the allowance. A charge above its quote in the meantime keeps it ended. `◐ Tested automated`.
- A job record that could not be written leaves no temporary copy behind, which after a restart read as a claimed request and blocked the same request as Unknown. `◐ Tested automated`.
- An agent that withdraws a spend request (`cancel_prepared_job`) while the person's approval is going through gets an error saying the request may be sent; it was told "not sent". `◐ Tested automated`.
- Extend canvas: the new frame shows through an eased band inside the original (6% of its short side, at least the overlap) instead of a 32 px ramp, and on models that take a mask the frame's tone along the old edge is matched to the original's (a `-landed.png` copy lands; the downloaded take is kept). Past the band the original is untouched. `● Verified live: still broken`: a horizon step at the old edge and a figure drawn twice in the new area still show.
- Extend canvas checks read across the old edge: **Step at the old edge** and **Repeated detail past the old edge** (a feature of the original drawn again, as the doubled sun of a live run). `● Verified live`: both were recorded on the live run, but Extend canvas lands as a single result card that shows no badge, so the person does not see them.
- Replace inside: a lighter or darker patch just inside the selection edge is flagged (**Lighter patch at the selection edge**, **Darker patch at the selection edge**). The seam ring outside the selection could not see it. `● Verified live`: no patch appeared on a feathered ellipse in a sky (two takes, both Edges ok); the badge on a real patch was not seen live.

- The Takes card shows **Edge may show** for those badges; before, a take with a visible edge showed no badge. *Not seen live* (no take had a patch).
- The run card shows the charged price, else the price waiting for approval after a price change on upload, else the quote. It kept the first quote. `● Verified live: fixed` (New angle, 1 CU to 14 CU); the composer button still shows the first quote during the approval.
- Jobs and its popover name video jobs (Animate, Frame to layer) "Video", not "Image". `● Verified live: fixed` on Animate 5 s.

Fixed in the release first run

The release package was installed, signed in and used live for the first time on 2026-10-23 (Photoshop 27.11, macOS, clicks and keystrokes by a script).

- Prices never loaded when Photoshop started with the panel already open, which is the usual case from the second launch on. The button stayed on "Create image · ..." and Generate stayed off. Photoshop calls the panel's show() once, while the plugin is still loading, and does not call it again when the panel is reopened. A panel shown during loading now starts its price checks when loading ends. Development builds hid the bug: their test loop starts the price checks itself. `● Verified live` on the release package: install, sign-in and a first image in 1 min 53 s. `◐ Tested automated`.
- After Undo of a landed run, the panel kept showing the takes that Undo had removed ("Showing take 1"). The panel now also follows Photoshop's History changes. `● Verified live`: Undo and Redo, twice each, and the card followed the layers every time. `◐ Tested automated`.

Fixed after the 2026-10-23 review

- New layer covered what was under it: it landed as an opaque picture at canvas size over the lettering of a character sheet. With a selection, **New layer** is now offered in the selection context, sized to the selection and fitted inside it, and lands cut out: the request asks the model for a transparent background, and a take that still covers what is under it (GPT Image 2.5 Flare returned a painted sky at alpha 0.91 to 0.98) is cut at placement by Photoshop's own Remove Background, as an editable layer mask (no credits). Photoshop takes a small layer's whole rectangle for its subject, so the take is scaled up to the canvas for the cut and back to its frame after. A **Lands** chip switches between Cut out and Full frame. `● Verified live` (a swallow kite inside a 300 x 380 selection over a painted sky: cut, no box); `◐ tested automated`.
- Formats cut painted titles: the poster's title was cropped or gone in Wide and Square. The picture is now read without its type layers (and Split text layers), and each live title is laid out again on each artboard at the same place in the frame, inside a 4% margin, scaled with the picture and smaller only when it would not fit, still editable. `● Verified live` before the placement rule changed (the title was copied and scaled into the artboard, but placed over the town); the rule now keeps the title's place in the frame, `◐ tested automated`, not re-run live.
- Cut out subject keeps the selection's own shape (a lasso, an ellipse), not only its box. `◐ Tested automated`.
- Keep take leaves the group named after its prompt: it kept "(4 takes)" with one take inside. `◐ Tested automated`.
- A Free Transform after Cut out subject or Remove background showed a 1 px line of the hidden pixels along the layer's edge: a mask written over its box revealed everything outside it. A new mask now starts as Hide All. `● Verified live` (a cut-out scaled 173% on a character sheet: no line); `◐ tested automated`.
- One-click actions on a cut-out sent what its mask hides: New angle on a Cut out subject layer got the sky and a chimney back. Photoshop's pixel read leaves a layer's own mask out, so the read now keeps the mask in the alpha (a disabled mask is left out, as Photoshop shows the layer). `● Verified live` (New angle then returned the figure alone); `◐ tested automated`.
- A kept take offered only Refine, More like this, Upscale and Animate: its **More** menu now has **Remove background** and **New angle**, both on the take that is showing. `● Verified live` (a kept side view cut out as a mask on the take); `◐ tested automated`.

- After a Free Transform or a Move, the panel kept the layer's old bounds, and New angle's takes landed where the layer had been. The panel now also follows Photoshop's transform, move, duplicate and paste events. `Tested automated`; `not tested live` (the development loop's own edits do not reach the panel's listener).
- Split into layers placed a sky (a fully opaque element) as an 84 x 47 px patch in a corner: a small plain patch of a backdrop matched better than its real place. A fully opaque element is now searched at 90% of the picture's width or more. `Verified live` (the same job placed again: the sky across the top); `tested automated`.
- Place in current document (Jobs) failed on Split into layers, Remove or Split text, Formats and the other later flows when the job's own document was gone ("Open ... to place this result there"); it now places in the document in front when its canvas size matches. `Verified live` (a split placed again after a panel reload); `tested automated`.
- Relight left a dull line of the original around a relit figure drawn a little narrower: the layer's own soft edge sat where the outline mask faded. The mask now grows a few pixels past the outline, so the take covers that edge. `Verified live`; `tested automated`.
- New layer in a selection read "Lands above ... at canvas size" until the first price; it now says "inside the selection" from the start. `Verified live`; `tested automated`.
- Formats put the title of a second artboard outside it, at the top of the Layers panel: after the first artboard, Photoshop leaves the placed layer selected, and a title copied into a document with artboards does not keep its place. The artboard is now read from the placed layer's parent and the title is laid out from where the copy really landed. After a plugin reload, the next format is added to the open "<title> formats" document instead of a new one. `Verified live` (9:16, 4:5, 1:1 and 16:9 in one document, each title live inside its artboard); `tested automated`.
- Upscale opened its new document as "Scenario upscale upscale-layer G4xH38"; it now opens as "<source title> ×2" (or ×4). The layer keeps its Scenario name. `Verified live`; `tested automated`.
- Animate warped a poster's lettering: the clip was made from the picture with its title in it. Animate now reads the picture without its type layers (as Formats does); they stay in the document, and the landing note says so. `Verified live` (the clip has no title); the note `tested automated`.
- Create image could not be reached from a new document: it lived only in the no-document context, which Photoshop hides on its Home screen, and a blank File > New document offered Split into layers first. A blank document (one Background of a single color) now opens on **Create image**, made at the canvas's aspect and landing as a layer above the Background. `Verified live` (context and price); `tested automated` (the landing).

Changed

- Button shapes follow the Scenario web app: the Generate button is a rounded rectangle (6 px corners), the chips and other buttons are full pills. `Verified live`.

Fixed in the real-click pass

A verification pass on 2026-10-22 pressed the panel's controls with real macOS mouse events (Photoshop 27.11, development build, panel docked at 280 px): the dev loop's own click calls a handler from code, so it passed while no real click could land on 2026-10-21. Each fix below is `tested automated` (a test that fails on the code before it); the live result ends each line.

- The panel could not be resized: docked, it stayed at its minimum size, 280 x 420, with empty dock space under it. The manifest now gives a maximum size. `Verified live`: after a Photoshop restart the panel fills its dock column (280 x 912); resizing by hand is for the person to confirm.
- A redraw now waits while the mouse button is down in the panel, so a click lands on the node it was pressed on. Job polls, price checks, the chip fold and menu refills redraw at any moment, and a redraw between press and release lost the click. `Tested automated`; `not tested live` yet with real clicks.

- A menu opens at its top. The menu box is reused and kept the last menu's scroll: the Model menu opened on its last rows, with Recommended out of view. `● Verified live` before the fix (the menu opened scrolled); the fix is `◐ tested automated`.
- Text that an open menu covers is hidden while it is open: a one-click action's note ("Separates the lettering...") showed through the More menu. `● Verified live` before the fix; the fix is `◐ tested automated` only for the hiding rule, `○ not tested live`.
- The Model menu lists only models that can take the layer. On Make versions it offered Gemini 3.1, Gemini 3.0 Pro, Seedream 5.0 Pro, Gemini 3.1 Lite and Grok Imagine Image 2.0, and each one blocked the button ("This model takes no image, so it can't edit an area"). All models now refuses such a model with a note. `● Verified live` before the fix (each of the five, real clicks); the fix is `◐ tested automated`.
- An approved agent spend keeps its card, reading "Approved, sending", without its buttons, until the approval ends. The card came back with both buttons live, and Decline said nothing was spent while the charge went on. `◐ Tested automated`.
- Settings > Agents and Connect a client are drawn again only when the link state changes: with no helper running, the link dials every 3 s and the whole box, its switches and its fields, was rebuilt each time. `◐ Tested automated`.
- The Library reads each project's own images (its list was kept per filter only, so a switch within a minute showed the other project's images), and a Library error has **Try again** (it stayed until another filter was clicked). `◐ Tested automated`.
- Upscale on a model with no factor setting (P-Image Upscale, Recraft Crisp Upscale) shows no factor: the chip and the button kept the factor last chosen on Topaz ("Upscale ×6") while the model upscales at its own rate, and the menu offered ×2 only. `● Verified live` before the fix; the fix is `◐ tested automated`.
- Extend canvas leaves out the ratio the canvas already has: a 3:2 document was offered 3:2, and choosing it blocked the button ("Already 3:2, pick another target"); a target chosen on another document moves to the next one. `◐ Tested automated`; found through the dev loop.
- Extend canvas moves "+20% each side" to a ratio on a model that cannot extend per side (Photoroom Expand, Reframe): the button blocked with "This model expands around the center". `◐ Tested automated`; found through the dev loop.
- A source over Scenario's 4 MB upload limit (Upscale, Remove background and the one-click tools) runs on the second Generate, read at the size that fits: the panel kept no reduced size, so each Generate said "click Generate again" and stopped again. `● Verified live` before and after the fix (Upscale on a 1536 x 1024 layer after Extend canvas: priced again at 1638 x 922, then ran).
- A reloaded panel reads the document context at once: it showed "No document open" with a document open until Photoshop sent a document event. `◐ Tested automated`.

Changed

- Extend canvas defaults to Reframe at 2K, with the resolution pinned by the lane preset; GPT Image 2.5 Sunburst at Medium was the runner-up of a two-take comparison per model on the same poster and canvas, and GPT Image 2.5 Flare stays in the Model menu (it drew the main figure a second time in 3 of 4 takes). The built-in Extend prompt is unchanged: a line asking not to repeat anything from the center did not help. Reframe at 2K costs more than Flare at Low (about seven times as much at 16:9); the button shows the price before the run. `◐ Tested automated`; `● verified live` on the same poster to 16:9: one figure, no step at either old edge.

Fixed after the re-test

`◐ Tested automated`. Checked live on the evening of 2026-10-21 (development build, one run each at the default settings); the result ends each line, and a line without one was not checked live.

- The single result card (Extend canvas, Upscale, Remove background) shows the badge the checks gave and each check in words. It showed none, so the step and the repeated detail recorded on the live Extend run were not visible. `● Verified live` on Extend canvas: the card read **Edges ok**, as the record holds.
- The composer button shows the price after the upload for the same action and settings (New angle, 1 CU to 14 CU live), where it kept the first quote during the approval and the run. *Not seen live*: in the evening check the price did not change after the upload.
- Extend canvas: the step limit is 22 (was 10), calibrated on 14 labelled real takes; at 10 it flagged clean takes. The measured scores are a fixture test. `● Verified live`: a clean Reframe take read **Edges ok** (step 12.5).
- Photoshop could stay busy, and the panel showed the host's raw "host is in a modal state". The plugin's modal scopes now run one after the other, so a **Keep take** clicked while the context thumbnail is read waits for it (the first Keep after a placement did nothing); a scope Photoshop refuses as busy is tried again three times, then the panel reads "Photoshop is busy. Try again in a moment." (`PhotoshopBusy` for agents). A test holds that no modal scope wraps a network call, an upload, a poll, a download or a price check, and that no request is sent while one is open during a whole Extend run. `● Verified live`: Keep sent about 3 s after a two-take landing worked on the first command, with no host text and no busy line (a click within the first second was not reproduced).
- The context card's second row of action chips was cut in half at the composer's edge (a layer card with its Select field and two chip rows, docked at about 300 px). The context card and the chips now sit outside the part that shrinks, so they keep their full height, and the composer with its button stays in view: when even an empty middle leaves too little room, the last chips fold into **More**, one at a time, down to one row. `◐ Tested headless` in Chrome with the plugin's own page and view code on each context at 280, 320 and 360 px: every chip row whole and the button in view from 360 to 390 px high (300 for takes), where 420 px failed before in three cases; `○ not tested live`. `● Verified live: not fixed in every case` (docked at 280 x 420): the selection and takes contexts and a plain layer card keep their rows, but with an Extend canvas result layer active the third chip row is cut in half and nothing folds into More. Fixed after that check: UXP gives the chips no size (each read 0 x 0), so the rows could not be counted; the fold now runs again once the layout has settled, one chip at a time, while the composer is out of view or the top scrolls, and two chips stay beside More. `● Verified live` through the dev loop at 280 x 420 on a layer card with a long name (the detail on two lines, three chip rows before): two chips went into More, both rows whole, the composer and its button in view; a selection card was unchanged.
- The context card names a Scenario layer in plain words: "Take 1 of A paper kite" instead of "Take 1 NfpUm8 · Layer 1536 × 1024", a single result by its run's name; the id tail and the size stay in the tooltip, and a layer the person renamed keeps its name. `● Verified live in part`: an Extend canvas result is titled "Extend canvas", but the detail line still names the raw layer, and a New angle run's card reads "Edit".
- New angle is priced before the upload with a blank stand-in at the capture's size; the 256 px stand-in quoted 1 CU where the real capture cost 5 to 14 CU. `● Verified live`: 17 CU before the upload on a 1024 x 1536 layer, charged 17, no approval card.
- Development builds: the dev loop can open Style > **Match an image** and pick its picture (selection, layer, visible, or a file from the data folder). Release builds are unchanged.

Documentation

- README rewritten for people installing the plugin: what it does, install from a built `.ccx`, a first generation, and links to the guides. The build-by-build technical status moved to `docs/STATUS.md`.
- `docs/USER_GUIDE.md`: quick start, every action with before and after pictures from live runs, selection edits (Inside selection or Bolder), takes, references, styles and trained models, Library, prices and approval, Jobs and recovery, settings, agents, troubleshooting.
- `docs/MCP.md`: real screenshots and a recording of an agent run, how each client was checked, and the from-checkout setup command while the helper is not on npm.
- New `docs/KNOWN_LIMITATIONS.md` and `SECURITY.md`; `PRIVACY.md` brought current for 0.7.1.

- The handbook: `npm run handbook` builds the docs into one web page with contents, search, light and dark themes and pictures that enlarge, a single HTML file with the pictures embedded (its fonts load from Google Fonts) and an A4 PDF with a cover and page numbers (`build/handbook/`). Made for photoshop.scenario.com; not published yet.
- The photoshop.scenario.com page in `site/`: the RED HORIZON launch film, the eight steps on two examples (RED HORIZON and THE LAST KEEPER), the gouache film, the PDF guide with a page-by-page reader, then install, prices, jobs, privacy and known limits. `.github/workflows/pages.yml` publishes it by hand from main once Pages and the DNS record are set (steps at the top of the file); `test/unit/site-page.test.js` checks that every file it links to is in `site/`.
- Panel screenshots under `docs/media/screens/`, captured in Photoshop 27.11. The before and after pictures and the recordings the guides link to follow in a later pass.

0.7.1 - 2026-10-20

Fixed

- Replace inside with a reference shows one line under the refs row: "Use an object or a style as the reference. A whole scene can be copied in."
- The small result note reads "Result is 2048 px for the 2304 px area".
- Library names, the place note and run names are cut on a word boundary with an ellipsis, never inside a word.
- Recent comes back after the panel reloads while Photoshop keeps running, also for a document that was never saved.

0.7.0 - 2026-10-20

The creation panel, the agent tools and the trained and Library work in one build.

Added

- Creation panel: the context card reads what the document holds (no document, a layer, a cut-out, a selection, a group of takes, a split element) and leads with the actions that fit; one composer with the refs row, the prompt and Suggest, chips that show their values, and the price on the button; the Takes card, Recent, and the Jobs popover.
- Later flows: Render sketch, New angle, Relight to scene, Split into layers, Remove text and Split text, Select by description, Formats, Animate with Frame to layer, the Upscale compare at 100%, and the agent's plan card (not fed by the agent link yet).
- Replace inside and Remove on Ideogram 4.5 Precise Edit and Ideogram 4.5, with an Edit chip: Inside selection, or Bolder (GPT Image 2.5 Flare with a mask that grows over what the model drew past the selection).
- Agents: the tools of the local Scenario MCP server (`scenario-photoshop`) for documents, references, prices, generation, jobs and landing results; three switches, No-spend and Agent spending; the spend, upload and proposal cards; Photoshop commands through an allow-list; Connect a client for Claude Code, Codex, Cursor and Claude Desktop.
- Trained models: Run with, stacking on one base, trigger words; the Library grid with search, filters, Show more, Place from Library and + Reference > Library.

Fixed

- The + Reference row shows again on Replace inside: its references go to Precise Edit's own reference input (3 at most with a selection).
- The upload and proposal cards are drawn in the panel; the stylesheet hid them with the F5 card's rule.

0.5.0 - 2026-10-19

Added

- Jobs: one record per job written before anything is sent, recovery after a Photoshop restart at every step, Unknown resolution against the project's jobs, downloads with fresh links, and placement in the job's own document.

0.4.1 - 2026-10-19

Added

- Edit and Enhance: selection captures with margin and mask, the three Expand routes, Upscale, Remove background, trained edits per base route, and the checks after placement.
- Image lane: references, placement as embedded smart objects, layer metadata, takes, one History state per run.

0.2.0 - 2026-10-19

Added

- Sign-in with a Project API key, projects, the model catalog and picker, schema forms and exact prices from free price checks.

0.1.0 - 2026-10-19

Added

- Plugin skeleton, the local helper over stdio, port election and pairing.

22 Licence

MIT, see LICENSE. Third-party code: third-party notices.

Adobe and Photoshop are trademarks of Adobe. This project is not affiliated with or endorsed by Adobe.

The Scenario logo is a trademark of Scenario and is not covered by the MIT license.

This handbook is generated from the Markdown docs in the repository (`npm run handbook`), version 0.8.0, commit `bc6588c`, 10 November 2026.