

DETAILED INSTRUCTIONS

One Click Roof Structure

Instantly generate accurate timber roof structures and structural beams.

Overview

One Click Roof Structure turns any sloped Revit roof into a complete timber (or steel) roof frame — automatically. Select a roof, choose your beam families, click **Run**, and the plugin places four kinds of structural framing directly on the roof geometry:

- **Rafters** — sloped beams running from eave to ridge, spaced evenly across every roof plane
- **Ridge beams** — placed along every ridge line where two roof slopes meet
- **Purlins** — horizontal mid-slope beams at any number of positions you define (e.g. at 33% and 66% of the slope)
- **Cushions (wall plates / pads)** — horizontal beams along every eave edge, with vertical, inward, and orientation control

Modeling a roof structure by hand is one of the most tedious tasks in Revit: every rafter must be drawn individually on a work plane, trimmed to the roof boundary, and re-done from scratch whenever the roof changes. One Click Roof Structure reads the actual geometry of the roof — every sloped face, every edge, every ridge — and computes exact beam start/end points itself. Rafters are automatically trimmed to the true boundary of each roof plane, including irregular shapes, hips, and valleys. What takes hours manually takes seconds.

Just as important: the plugin is **non-destructive and re-runnable**. Every beam it creates is tagged as belonging to a specific roof. When you change a setting and click **Run** again, the plugin replaces only the affected beams — no duplicates, no manual cleanup, and sections whose settings did not change are left untouched.

Key Benefits & Use Cases

Who it's for

- **Architects** who need a realistic roof structure for sections, details, and visualizations without modeling every rafter by hand.
- **BIM modelers** who must deliver structural framing that stays consistent with the architectural roof as the design evolves.
- **Structural drafters** producing timber roof framing plans from an architectural model.

Concrete scenarios

- *Residential gable or hip roof*: select the roof, set rafter spacing to 80 cm, enable Ridge Beam and Cushions — a full traditional timber frame appears in seconds, correctly trimmed at hips and valleys.
- *Design iteration*: the client changes the roof pitch. Re-run the plugin — old beams for that roof are deleted and re-created on the new geometry in one step.
- *Multiple buildings*: select several roofs at once. Each roof is processed independently and keeps its own set of beams; one click updates them all.
- *Office standards*: save your firm's standard configuration (families, spacing, purlin layout) as a template file and share it with colleagues, who import it with one click.
- *Presentation views*: temporarily hide every plugin-created beam in the active view with one button while you work on the architecture, then unhide them just as quickly.

Requirements

- **Autodesk Revit 2022 – 2026** (Windows). Revit is Windows-only, and so is the plugin.
- The plugin is part of the **ArchPlug suite** and installs into the shared **ArchPlug** ribbon tab alongside other ArchPlug tools.
- Your project must contain at least one **Structural Framing** family type (any beam family — timber, steel, concrete). The plugin places instances of families already loaded in your model; it does not ship or load families itself. If no Structural Framing types exist, the plugin will tell you: "*No Structural Framing types found in the model.*"

Installation

1. Run the provided **.msi installer** and follow the prompts.
2. If Revit was **closed** during installation, no further steps are needed — the plugin is available the next time you start Revit. (A restart is only needed if Revit was running during the install.)
3. Open Revit and look for the **ArchPlug** tab on the ribbon. One Click Roof Structure appears there as its own button.
4. Click the button to open the **Place Roof Structure** window.

User Interface Walkthrough

The plugin opens a single window titled "**Place Roof Structure**". Key characteristics:

- **Modeless** — the window floats above Revit and stays open while you keep working: you can pan, zoom, change views, and (most importantly) select roofs in the model with the window open. It stays on top so it never gets lost behind Revit view windows.

- **Resizable** — drag any edge or corner, or maximize it. The Purlins section grows with the window, giving the purlin table more room.
- **No OK/Cancel** — there is one **Run** button that executes placement *without closing the window*, so you can iterate: tweak a value, Run, inspect, tweak again. Close the window with the title-bar **X** or the **Esc** key; your settings are saved automatically on close.
- The window remembers its **position and size** between sessions (including maximized state), and all field values persist between sessions as well.

The window contains four collapsible-style option groups (top to bottom), a Hide/Unhide row, and a bottom button row.

"Rafters" group

Control	Label	What it does
Checkbox	Enable	Turns rafter creation on/off. When off, the rest of the group is covered by a dimmed overlay reading " Enable Rafters first " — only the Enable checkbox stays clickable.
Drop-down	Family:	The Structural Framing type used for rafters. Lists every Structural Framing type in the model, formatted as FamilyName : TypeName , sorted alphabetically.
Text field	Spacing (cm):	Target center-to-center rafter spacing. The unit suffix in the label follows your project's length display unit — it shows <i>(mm)</i> , <i>(cm)</i> , <i>(m)</i> , <i>(in)</i> or <i>(ft)</i> depending on your project units, and the value is interpreted in that unit. Default: 80.

"Ridge Beam" group

Control	Label	What it does
Checkbox	Enable	Turns ridge beam creation on/off (overlay: " Enable Ridge Beam first ").
Drop-down	Family:	Structural Framing type used for ridge beams.
Text field	Offset (...):	Vertical offset applied to the ridge beam, in project length units. Positive values move the beam up, negative down. Default: 0.

"Purlins (mid-rafter)" group

Control	Label	What it does
Checkbox	Enable	Turns purlin creation on/off (overlay: " Enable Purlins first ").
Drop-down	Family:	Structural Framing type used for purlins.
Text field	Vertical offset (...):	Vertical (Z) offset applied to every purlin, in project length units. Default: 0.
Checkbox	Equal distances	When checked, purlin positions are computed automatically so all rows are evenly distributed along the slope (e.g. 1 row → 50%; 2 rows → 33.3% / 66.7%; 3 rows → 25% / 50% / 75%). The Position (%) column becomes read-only and updates live as you add/remove rows. Uncheck to type positions manually.
Table	columns Purlin / Position (%)	One row per purlin. Purlin is a free-text name (always editable, for your own reference). Position (%) is where the purlin sits on each roof plane: 0% = eave (bottom), 100% = ridge (top) . Values are clamped to 0–100.
Button	Add	Adds a new row (named "Purlin N", position 50.0 — or auto-recomputed if Equal distances is on).
Button	Delete	Removes the selected row; if no row is selected, removes the last row.

Each purlin row is placed **once per sloped roof face** — so on a gable roof with two planes, one row produces two purlins (one per side), automatically trimmed to each plane's boundary.

"Cushions (under rafters)" group

Cushions are the horizontal pads/wall plates placed along every **eave edge** (the lowest edge) of each roof plane.

Control	Label	What it does
Checkbox	Enable	Turns cushion creation on/off (overlay: " Enable Cushions first ").
Drop-down	Family:	Structural Framing type used for cushions.
Text field	Vertical offset (...):	Moves cushions up/down from the eave edge, in project length units. Default: 0.
Text field	Inward offset (...):	Shifts cushions horizontally toward the ridge (into the building), in project length units. Useful for placing the plate on the wall rather than at the roof edge. Default: 0.
Radio buttons	Orientation: — Horizontal / Vertical	Vertical (default) keeps the beam profile upright (Cross-Section Rotation = 0°). Horizontal lays the profile flat by setting the beam's Cross-Section Rotation to 90° — exactly the value you see in Revit's Properties palette.

Hide / Unhide row

Button	What it does
Hide all beams	Hides every beam this plugin has ever created (all roofs, all types) in the active view only — a quick declutter that does not touch other views or delete anything. The status line reports e.g. "Hid 42 beam(s) in this view".
Unhide all beams	Reverses the above in the active view.

Bottom row

Control	What it does
Save Template button	Opens a save dialog (" <i>Save Roof Structure Template</i> ", filter <i>JSON Template (*.json)</i>) and writes all current settings — every checkbox, family choice, value, and the full purlin table — to a .json file. The default location is your Documents\ArchPlug\Templates folder; the default file name is <i>roof_template.json</i> . A confirmation shows the saved path.
Import Template button	Opens a matching open dialog (" <i>Import Roof Structure Template</i> ") and loads a previously saved template, filling every field in the window. Confirmed with " <i>Template imported.</i> "
Status label	A single-line, accent-colored status area between the template buttons and Run. Shows progress ("Running...", "Running on 2 roof(s)...") and results (see Workflow step 7).
Run button	Executes placement/update on the currently selected roof(s). Does not close the window.

Keyboard

- **Esc** — closes the window (settings are saved first).
- **Ctrl+Z / Ctrl+Y** — undo / redo your edits *to the input fields in this window* (values, checkboxes, purlin rows). This is separate from Revit's own undo, which handles the model itself.

Step-by-Step Workflow

1. **Open a view** where you can see and select the roof — a 3D view is most convenient.
2. **Launch the plugin** from the **ArchPlug** ribbon tab. The **Place Roof Structure** window opens with your last-used settings already filled in.
3. **Select the roof(s)** in the model. Because the window is modeless, you can do this with the window open, before or after adjusting settings. You may select one roof or several — each is processed independently. (Other selected elements are simply ignored; only roofs count.)
4. **Enable the sections you want** — Rafters, Ridge Beam, Purlins, Cushions — and pick a **Family**: for each enabled section. Disabled sections are dimmed with an "Enable ... first" overlay.
5. **Set the values**: rafter **Spacing**, ridge **Offset**, purlin rows and offsets, cushion offsets and **Orientation**. All lengths are in your project's display unit, as shown in each label. Decimal commas are accepted (e.g. "62,5" works the same as "62.5").
6. **Click Run**. The plugin: - validates your input (a roof must be selected; at least one section must be enabled; each enabled section's family must exist; rafter spacing must be a positive number), - deletes any of its own previously created beams for the selected roof(s) in sections whose settings changed (or were disabled), - reads the roof geometry, finds all sloped faces, and places the new beams, - commits everything in **one Revit transaction** named "*Roof Structure - Update*" — a single Ctrl+Z in Revit reverses the entire run, - suppresses non-critical Revit warnings (such as overlapping framing) so the run never stalls on warning pop-ups.
7. **Read the result in the status line**, e.g. `Done (1 roof). R=24 | Ri=1 | P=4 | C=6` — R = rafters placed, Ri = ridge beams, P = purlins, C = cushions; `=del` means that section's beams were deleted because you disabled it. On success there is **no pop-up** — only errors produce a dialog.
8. **Iterate if needed**: change a value, click **Run** again. Only the sections you changed are rebuilt; everything else is left exactly as it was.
9. **Close** the window with **X** or **Esc** when done. All settings, the purlin table, and the window position are remembered for next time.

Feature Reference

Smart Update (no duplicates, ever)

Every beam the plugin creates is tagged in its **Comments** parameter with a marker identifying its roof and its role (rafter, ridge, cushion, or purlin). This drives the update logic:

- On every **Run**, the plugin compares each section's current settings against the settings used the last time that roof was processed.
- **Unchanged section + beams already present** → skipped entirely (fast, nothing touched).
- **Changed section** → that roof's existing beams of that type are deleted and re-created with the new settings.
- **Section disabled** → that roof's existing beams of that type are deleted.
- Beams belonging to *other* roofs are never touched, even when several roofs are selected together.

The per-roof settings memory persists between sessions, so re-running tomorrow with identical settings still changes nothing. After each run the plugin also verifies that its new beams carry the marker correctly and warns you if not (see Troubleshooting).

Practical consequence: **do not clear or edit the Comments parameter** on plugin-created beams. The Comments value is how the plugin recognizes its own work; if you overwrite it, the next Run will treat those beams as foreign and create fresh ones alongside them.

Multi-roof processing

Select any number of roofs before clicking Run. Each roof is processed independently with its own tagged beam set and its own change-tracking, and the whole batch is committed atomically in a single transaction. The status line reports the total, e.g. **Done (3 roofs). R=71**
| C=18.

Rafter placement logic

For each sloped face of the roof (the underside plane of each roof slope):

- Rafters run in the **slope direction** (from eave toward ridge) and are distributed along the **eave direction**.
- The requested spacing is treated as a target: the plugin divides the face width into a whole number of equal intervals as close as possible to your spacing, so the first and last rafters always sit exactly at the face edges and all gaps are identical.
- Every rafter is trimmed to the actual boundary of the roof plane — trapezoidal hip faces, notches, and irregular outlines are handled automatically (each rafter spans from the first to the last boundary intersection along its line).

Ridge beam logic

Ridge lines are detected as **edges shared between two sloped faces** — the natural ridges and hips-to-ridge lines of the roof. If the roof has no such shared edge (e.g. a single-plane shed roof), the plugin falls back to the **topmost edge of each slope**, so a shed roof still receives a beam along its high edge. Duplicates are filtered out, and the optional vertical **Offset** is applied to the whole beam.

Purlin placement logic

For each enabled purlin row, one horizontal beam is placed per sloped face at the given **Position (%)** measured up the slope: 0% = eave, 100% = ridge. Each purlin spans the full width of the face at that height, trimmed to the face boundary (so on a hip face, purlins get shorter toward the top — correctly). The **Vertical offset** raises or lowers all purlins. With **Equal distances** checked, positions are recomputed automatically from the number of rows.

Cushion placement logic

Cushions are placed along the **eave edges** — edges of each sloped face lying at that face's lowest level. Edges shared by adjacent faces are deduplicated so you never get doubled pads. Controls:

- **Vertical offset** — moves the pad up/down.
- **Inward offset** — moves the pad horizontally toward the ridge (e.g. to center it on a wall).
- **Orientation** — Vertical keeps the section upright; Horizontal writes 90° into the beam's **Cross-Section Rotation** parameter (with a geometric rotation fallback if a family exposes that parameter as read-only). The plugin even self-heals: if a previous run failed to apply the rotation, the next Run detects the mismatch and rebuilds the cushions instead of skipping them.

Families and types

- All four drop-downs list every **Structural Framing** type loaded in the current model, as **FamilyName : TypeName**.
- You can use a different family for each role — e.g. 8/14 timber for rafters, 14/16 for the ridge, 12/12 plates for cushions.
- Inactive types are activated automatically at run time; you do not need to place a dummy instance first.
- Beams are created as genuine **Structural Framing (Beam)** elements — they schedule, tag, and behave exactly like manually modeled beams.

Level assignment

Created beams are assigned the **roof's own reference level**. If the roof has no level, the plugin falls back to the nearest level at or below the roof; failing that, the lowest level in the project. If no level can be found at all, that roof is skipped with an error message.

Units and input

- Length fields follow the **project's length display unit**; each label shows the active unit (mm/cm/m/dm/in/ft).
- Both  and  are accepted as the decimal separator.
- Purlin positions are clamped to the 0–100% range.

Persistence, templates, undo

- **Session persistence:** every setting and the window geometry are saved automatically when you click Run and when you close the window, and restored on next launch. Updating the plugin does not reset your saved values.
- **Templates: Save Template / Import Template** exchange the full configuration as portable .json files (default folder: *Documents\ArchPlug\Templates*) — ideal for office standards and for moving settings between projects or machines.
- **In-window undo:** Ctrl+Z / Ctrl+Y step through your edits to the form's fields.
- **Model undo:** each Run is one Revit transaction ("*Roof Structure - Update*"), so one Revit Undo removes/restores the entire run at once.

Tips & Best Practices

- **Work in a 3D view.** Selecting roofs and judging the result is far easier in 3D; the Hide/Unhide buttons act on whichever view is active.
- **Keep the window open while iterating.** Run does not close the window by design — adjust spacing, Run, look, adjust again. Smart Update means each Run only rebuilds what changed.
- **Let spacing be a target, not a law.** If your roof plane is 9.35 m wide and you ask for 80 cm, the plugin picks the closest even division so the layout is symmetric edge-to-edge. If you need an exact modular spacing, size the roof plane accordingly.
- **Use Inward offset for cushions** to pull wall plates off the roof edge and onto the supporting wall — typically half a wall thickness.
- **Save a template per roof typology** (e.g. "Timber 80cm + double purlin", "Steel garage") and import it instead of re-entering values.
- **Select all roofs of a building at once** when settings are shared — one Run, one transaction, one Undo.
- **Leave the Comments parameter alone** on generated beams. If you need your own beam identifier, use Mark or a shared parameter instead.
- If you want to modify individual generated beams by hand (cutbacks, joins, material overrides), do it **after** the design is stable — a subsequent Run with changed settings for that section will delete and re-create those beams, discarding manual edits on them.

Troubleshooting & FAQ

"Please select one or more roofs, then click Run." Nothing in your current selection is a roof. Select the roof element itself (a footprint or extrusion roof), not walls, fascias, or gutters, then click Run again. Roofs inside groups or links cannot be selected directly.

"Please enable at least one section." All four Enable checkboxes are off. Enable at least one of Rafters, Ridge Beam, Purlins, or Cushions.

"No Structural Framing types found in the model." The project contains no beam family at all, so the drop-downs would be empty. Load any Structural Framing family (Insert → Load Family → Structural Framing), then relaunch the plugin.

"Rafter family not found: ..." (or Ridge/Cushion/Purlin family not found) The family type shown in the drop-down no longer exists in the model — typically because it was deleted or the settings came from a template saved in a different project. Re-pick a valid type from the **Family:** drop-down.

"Enter a valid rafter spacing in centimeters." The Spacing value is zero, negative, or not a number. Enter a positive number (in your project's length unit).

"Roof ...: no sloped faces — skipped." The plugin found no sloped planar underside on that roof. This happens with **flat roofs** (no slope — there is nothing for rafters to follow) and with **curved/domed roofs** (non-planar faces are not supported). Give the roof a slope, or model that structure manually.

"Roof ...: no Level found — skipped." The project has no level the beams could be assigned to. Ensure the project contains at least one level.

Beams appear doubled after a Run. Almost always caused by the Comments tag having been cleared or overwritten on previously generated beams (the plugin then cannot recognize them as its own). Delete the stray beams manually, leave Comments untouched in the future, and re-run. The plugin also warns you proactively after a run if tagging could not be applied: *"Re-running will create duplicates instead of updating."* — if you see that message, the chosen beam family is preventing the Comments parameter from being written; switch to a standard beam family.

"Cushion rotation: N of M beam(s) failed..." The selected cushion family would not accept the Cross-Section Rotation value, and the geometric fallback also failed. Try a different (standard) Structural Framing family for cushions. Note that the plugin automatically re-attempts the rotation on the next Run.

I clicked Run and nothing changed. If the status line says *"Done ... — nothing changed."*, Smart Update determined that the selected roofs already have beams matching the current settings — that is the intended no-duplicate behavior. Change a setting (or disable/re-enable a section) to force a rebuild.

Rafter spacing in the model isn't exactly what I typed. By design: the plugin uses the nearest equal division of each roof plane so rafters land exactly on both edges with uniform gaps. The actual spacing is as close to your value as the plane width allows.

Where did the pop-up with results go? There is none on success — results go to the in-window status line. Pop-ups appear only for errors (first 10 are listed).

Can I undo a run? Yes. Each Run is a single Revit transaction named "*Roof Structure - Update*" — one press of Ctrl+Z in Revit (with the model window focused) reverts everything the run did. Ctrl+Z while the plugin window is focused instead undoes your edits to the plugin's input fields.

Does it work on roofs with holes, hips, and valleys? Yes — rafters and purlins are trimmed against the true boundary of each roof plane, and hip/ridge edges shared between two planes receive ridge beams. Each sloped plane is framed independently.

The window opened on the wrong monitor / off-screen. The window restores its last position but checks that it is still visible on a connected screen; if the saved position is off-screen (e.g. a monitor was unplugged), it opens centered instead. It also adapts automatically to mixed-DPI multi-monitor setups.

Are warnings like "structural framing overlaps" suppressed? Yes. Non-critical Revit warnings raised during placement are dismissed silently so the run completes without interruption. Genuine errors are still collected and reported.