

DETAILED INSTRUCTIONS

Smart Auto Tagging

Intelligent one-click tagging for windows and doors across whole levels.

Overview

Smart Auto Tagging is an ArchPlug plugin for Autodesk Revit that automatically places, orients, and maintains annotation tags on **windows**, **interior doors**, and **terrace doors** in floor plan views. Each tag is a Generic Annotation instance that is filled with the element's live data — Type Mark, width, height, and (for windows) parapet/sill height — and is automatically rotated to match the orientation of the host element in the plan.

The plugin is not a one-shot macro: it runs a **smart update cycle**. Tags are permanently linked to their model elements, so re-running the command refreshes the text on existing tags, creates tags for new elements, and deletes tags whose elements no longer exist — all without disturbing tags you have moved or rotated by hand.

The dialog is **modeless**: it floats above Revit while you keep working in the model, and it stays open after every run so you can change options and run again immediately.

Key Benefits & Use Cases

- **One click tags an entire floor plan** — every window and door in the view receives a correctly oriented, fully populated tag.
- **Multi-level tagging in a single run** — switch to "All levels" mode, tick the levels you want, and the plugin tags one representative architectural floor plan per level in a single Revit transaction.
- **True synchronization, not duplication** — re-running never creates duplicates. Existing tags are recognized by a persistent element key and only their text is refreshed.
- **Respects your manual adjustments** — tags you have dragged or rotated stay exactly where you put them, unless you explicitly opt in to re-orienting them.
- **Orientation-aware text** — tags on rotated elements are rotated to match, and the width/height values are swapped when needed so the readable text always reports the correct dimension.
- **Terrace door support** — doors marked with a keyword in their Comments parameter get a dedicated terrace-door tag family.
- **Bulk visibility control** — hide or unhide all Smart Auto Tagging tags in the current view or across all checked levels with one button.
- **Clean output** — Revit warning popups (e.g., overlapping annotations) are suppressed during the run, so large runs never stop to ask questions.

Typical use cases: schematic and permit drawing sets where every window/door needs a dimension-and-mark tag; keeping tag data in sync while the model evolves; re-tagging after

windows or doors are added, deleted, resized, or re-typed.

Requirements

- **Autodesk Revit 2022 – 2026** (Windows only).
- Part of the **ArchPlug suite** — installed alongside the shared ArchPlug components.
- The three ArchPlug tag families must be loaded in the project (they ship with the plugin):
 - `ArchPlug_ST_WindowTag` — window tags
 - `ArchPlug_ST_DoorTag` — interior door tags
 - `ArchPlug_ST_DoorTagGealan` — terrace door tagsIf a required family is missing, the plugin stops and tells you exactly which family it could not find.
- Tagging works in **floor plan views** (Architectural discipline).

Installation

1. Close Revit.
2. Run the Smart Auto Tagging `.msi` installer and follow the wizard.
3. Start Revit and open the **ArchPlug** ribbon tab — the Smart Auto Tagging command is there.
4. Open a project, switch to a floor plan view, and launch the command. The **ArchPlug - Smart Auto Tagging** window appears.

User Interface Walkthrough

Main window — "ArchPlug - Smart Auto Tagging"

A fixed-size, always-on-top dialog. It is modeless — you can pan, zoom, and switch views in Revit while it is open. It remembers its screen position between sessions (and safely re-centers if that position is no longer on any connected monitor).

"Levels" group (top of the window):

Control	What it does
Current level (radio)	Tag only the active view. The active view must be a floor plan. This is the default mode.
All levels (selected in level settings) (radio)	Tag one architectural floor plan per level that is ticked in the Level Settings panel. With zero levels ticked, Run is disabled until you tick at least one.
Level Settings > (button)	Opens the ArchPlug - Levels side panel (see below). Always clickable in either mode.
(N selected) (label)	Live count of levels currently ticked in the Level Settings panel.

Category options (middle of the window):

Control	What it does
Windows (checkbox)	Tag all window elements visible in the target view(s) with the window tag family.
Doors (interior) (checkbox)	Tag all doors that are <i>not</i> terrace doors with the standard door tag family.
Terrace doors (by comment) (checkbox)	Tag doors whose Comments instance parameter contains the terrace keyword with the dedicated terrace-door tag family.
Terrace door comment: (text box)	The keyword used to recognize terrace doors (default: <code>terrace door</code>). Matching is case-insensitive and matches anywhere inside the Comments text.
Update tag positions (re-orient existing) (checkbox)	Opt-in per session: when checked, existing tags are moved back to their element's center and re-oriented. Always starts unchecked each time the window opens, so your manual tag positions are safe by default.
Run (button)	Executes the tagging pass with the current options. The window stays open afterwards. Pressing Enter also triggers Run.

Keyboard in the main window:

- **Esc** — closes the window. If the Levels panel is open, the first Esc closes the panel and the second closes the main window.
- **Ctrl+Z / Ctrl+Y** — undo/redo your changes to the dialog's own inputs (checkboxes, text, radios).

All options are saved automatically when you run or close the window and are restored the next time you open it (settings are stored per Windows user).

Levels panel — "ArchPlug - Levels"

Opened with **Level Settings >**. A resizable tool window that by default docks to the right of the main window; its position and size are remembered. **Esc** closes only this panel. **Ctrl+Z /**

Ctrl+Y undo/redo changes made inside the panel.

Control	What it does
Tick levels (Shift+click for range) (label)	Instruction line above the list.
Level list	All levels that have at least one non-template Architectural floor plan view, sorted bottom-up by elevation. (MEP/structural/site plans such as Lighting or Plumbing plans are filtered out, so the list stays clean.) Tick the levels you want to tag in "All levels" mode. Shift+click toggles a whole range of levels at once. The selection is saved instantly and persists across sessions.
Check all (button)	Ticks every level in the list.
Clear (button)	Unticks every level.

"Hide / Unhide" group (bottom of the panel):

Control	What it does
Hide all tags (button)	Hides every Smart Auto Tagging tag instance (window, door, and terrace-door tags) in the selected scope, using Revit's per-view Hide Elements.
Unhide all tags (button)	Reveals previously hidden tags in the selected scope.
Current view (checkbox)	Scope: only the active floor plan view. The active view must be an architectural floor plan, otherwise you get a message and nothing changes.
Checked levels (checkbox)	Scope: all architectural floor plan views of the levels ticked in the list above. If no levels are ticked, you get a message and nothing changes.

The two scope checkboxes are mutually exclusive — exactly one is always ticked — and the choice is remembered.

Step-by-Step Workflow

1. Open a floor plan view of the level you want to tag.
2. Launch **Smart Auto Tagging** from the **ArchPlug** ribbon tab.
3. Choose the level scope: - Leave **Current level** selected to tag only the active floor plan, **or** - Select **All levels (selected in level settings)**, click **Level Settings >**, and tick the levels to process (use **Check all** or Shift+click for ranges).
4. Tick the categories to tag: **Windows, Doors (interior), Terrace doors (by comment)**.
5. If you use terrace-door tags, make sure the affected doors have the keyword (default `terrace door`) somewhere in their **Comments** parameter, and that the same keyword is

in the **Terrace door comment:** box.

6. Leave **Update tag positions (re-orient existing)** unchecked for a normal update run. Check it only when you want existing tags snapped back to their element centers and re-oriented.
7. Click **Run** (or press Enter). The plugin creates new tags, refreshes existing ones, and removes orphaned ones in the target view(s).
8. Review the result. Adjust individual tags by hand if needed — the window is still open, so you can tweak options and **Run** again at any time; later runs will not move your hand-placed tags.
9. When the model changes (elements added, deleted, resized, re-typed), simply run again — tag data resynchronizes automatically.
10. Press **Esc** or the window's X to close. All settings are saved.

Feature Reference

Categories tagged

- **Windows** — all elements of the Windows category visible in the target view. Tagged with `ArchPlug_ST_WindowTag`.
- **Interior doors** — all Doors-category elements whose Comments do *not* contain the terrace keyword. Tagged with `ArchPlug_ST_DoorTag`.
- **Terrace doors** — Doors-category elements whose Comments *do* contain the terrace keyword (case-insensitive substring match). Tagged with `ArchPlug_ST_DoorTagGealan`.

If a tag family has several types, the plugin prefers the type whose name matches the family name (the default type as shipped), so user-created experimental duplicate types don't hijack the run.

Data written to tags

Window tags receive:

- the window's **Type Mark** (with fallbacks through the type's Mark and the instance Mark if Type Mark is empty),
- **width** — taken from the type's `Width` parameter as displayed in your project units; if the type has a text parameter named `Sill` with a value, that text overrides the width,
- **height** — taken from the type's `Height` parameter; a filled `Head` text parameter on the type overrides it,
- **parapet (sill height)** — read from the instance Sill Height parameter (with fallbacks to parameters named Sill Height / Parapet / Parapet Height and localized variants); only the leading numeric part of the displayed value is written.

Door tags (both interior and terrace) receive the door's **Type Mark**, **width**, and **height**, with the same **Sill** / **Head** override logic as windows.

Dimension values are written exactly as Revit displays them, so they follow your project units.

Tag placement and orientation

- New tags are placed at the **element's center** in the view (insertion point, curve midpoint, or bounding-box center, in that order of availability).
- Each new tag is rotated so its layout matches the element's orientation in the plan. Two methods are used automatically:
- **Bounding-box method** (default): the plugin classifies which side of the element the insertion point faces (left / right / top / bottom), then trial-rotates the tag in 90° steps until the tag's own orientation matches. Trials are batched per view for speed on large runs.
- **Direction method** (fallback for special cases): used for **narrow windows (type Width under 70 cm)** and for **elements rotated off the orthogonal grid** (more than 5° away from 0°/90°/180°/270° in the view). The tag is rotated to the element's hand direction and mirrored when the element faces "down" in the view, so text stays readable.
- **Width/height swap**: when the finished tag ends up in a "bottom" or "left" orientation, the width and height values are swapped in the tag's parameters so that the dimension printed along the element is always the correct one.
- **Single-leaf terrace-door correction**: doors recognized as Gealan-manufactured (by family name or Manufacturer parameter) that are narrower than 110 cm receive an extra 90° rotation to compensate for that family's geometry. This applies whether the door carries the terrace comment or not.

Smart update (re-run behavior)

- Every tag stores a **permanent key** of its element (the element's IFC GUID if present, otherwise its Revit Uniqueld). This is how tags and elements stay linked across sessions.
- **Existing tag, element still present** → only the text parameters are refreshed. Position, rotation, and any manual adjustments are preserved.
- **New element without a tag** → a new tag is created and oriented.
- **Tag whose element is gone** → the tag is deleted as stale.
- **Unchecking a category and running** removes all of that category's tags from the processed views. Running with *all three* categories unchecked is therefore a valid "clear all tags" operation for the selected scope.
- **Update tag positions (re-orient existing)** re-enables placement logic for existing tags: each one is moved to its element's center and re-oriented, then its text is refreshed. Use it after big plan rearrangements; leave it off otherwise.

Level modes and view selection

- **Current level** — tags the active view only. The active view must be a floor plan; otherwise Run shows: *"Active view is not a Floor Plan. Switch to a floor plan, or use 'All levels' mode."*
- **All levels** — tags one architectural floor plan per ticked level, processed bottom-up by elevation. When a level has several floor plans (working views, sheet copies...), the plugin picks intelligently: 1. the **active view**, if it belongs to that level; 2. otherwise the plan **already containing the most Smart Auto Tagging tags** (continuity with where you tagged before); 3. otherwise the first plan found. Because Generic Annotation tags are owned by a specific view, this ensures updates land in the plan you actually use.
- Running in All levels mode with nothing ticked shows: *"No levels are checked in Level Settings."*

Transactions and undo

- The entire run — all views, all categories — executes as **one Revit transaction** named "Smart Auto Tagging". A single **Ctrl+Z in Revit** reverts the whole run.
- Hide/Unhide operations are likewise single transactions and can be undone in Revit.
- Inside the plugin windows, **Ctrl+Z / Ctrl+Y** undo/redo your *input* changes (option toggles, level ticks, text edits) — this is separate from Revit's model undo.
- Revit **warning** popups raised during tagging (e.g., annotation overlaps) are automatically dismissed so the run never stalls. Genuine errors roll the transaction back and are reported in an error dialog.

Hide / Unhide all tags

From the Levels panel, **Hide all tags / Unhide all tags** toggle the visibility of every Smart Auto Tagging tag (all three families) in floor plan views within the chosen scope:

- **Current view** — the active architectural floor plan only.
- **Checked levels** — all architectural floor plans of the ticked levels.

Hiding uses Revit's per-view element hiding, so it affects only the targeted views and is fully reversible with **Unhide all tags**.

Persistence

Everything you set is remembered per Windows user: category checkboxes, terrace keyword, level mode, ticked levels, hide/unhide scope, and both windows' screen positions/sizes. The only exception is **Update tag positions (re-orient existing)**, which deliberately resets to unchecked every time the window opens.

Tips & Best Practices

- **Run early, run often.** Because runs are non-destructive to manual adjustments, the safest habit is to tag as soon as the plan is roughed out and simply re-run after every modeling session.
- **Fine-tune by hand after the run, not before.** Drag or rotate individual tags freely — subsequent runs only refresh their text. Only check **Update tag positions** when you intentionally want everything reset to element centers.
- **Use one plan per level as your "tagging home".** In All-levels mode the plugin favors the plan that already holds your tags; keep tagging in the same plans and updates will always land in the right place.
- **Standardize the terrace keyword.** Put the exact keyword (e.g., `terrace door`) in the Comments of every balcony/terrace door. Matching is case-insensitive and partial, so `Terrace door - living room` also matches.
- **Override texts when needed.** Add text values to `Sill` / `Head` type parameters on a window or door type to print custom width/height strings instead of the model dimensions — useful for special assemblies.
- **Undo is one step.** If a multi-level run did something unexpected, a single Revit Ctrl+Z reverts it completely.
- **Use Hide all tags for clean presentation exports,** then Unhide when you return to documentation work.
- **Shift+click in the level list** to tick long ranges of levels in tall buildings quickly.

Troubleshooting & FAQ

"Cannot find Generic Annotation family for windows/doors/terrace doors: ..." The named tag family (`ArchPlug_ST_WindowTag`, `ArchPlug_ST_DoorTag`, or `ArchPlug_ST_DoorTagGealan`) is not loaded in the project. Load it (or transfer it from a project where it exists) and run again. Untick the corresponding category if you don't need it.

"Active view is not a Floor Plan. Switch to a floor plan, or use 'All levels' mode." In Current level mode the active view must be a floor plan. Open one, or switch to All levels mode and tick levels in Level Settings.

"No levels are checked in Level Settings." / Run button is greyed out. All levels mode requires at least one ticked level. Click **Level Settings** > and tick levels. The **Run** button is intentionally disabled while zero levels are selected.

A level is missing from the Levels list. The list shows only levels that have at least one non-template **Architectural** floor plan view. Levels whose only plans are MEP/structural/site types (or view templates) are filtered out. Create an architectural floor plan for that level.

My terrace doors got the regular door tag. Check that the door's **Comments** parameter contains the keyword from the **Terrace door comment:** box (default `terrace door`) and that **Terrace doors (by comment)** is ticked. The match is case-insensitive but the keyword must appear within the Comments text.

Tags disappeared after a run. Two intentional behaviors can remove tags: (1) the tagged element was deleted from the model — its orphaned tag is cleaned up; (2) the category checkbox was unticked — running with a category unchecked removes that category's tags from the processed views. Re-tick the category and run again to restore them, or use Revit Undo.

All my hand-positioned tags jumped back to the elements. Update tag positions (re-orient existing) was checked. It resets every existing tag to its element center and re-oriens it. Undo in Revit (one step), then re-run with the box unchecked — it always starts unchecked for exactly this reason.

A tag on a rotated/angled element looks wrong. Elements rotated off the 0°/90° grid use a direction-based orientation with mirroring; in rare geometry the plugin cannot find a matching orientation and leaves the tag in its default rotation. Rotate that tag manually — future runs will preserve your correction.

A duplicate tag type I created is being used instead of the original. The plugin prefers the tag type named identically to the family. Keep the default type name intact, or delete experimental duplicates.

Nothing happened when I clicked Hide all tags. Check the scope: with **Current view** the active view must be an architectural floor plan; with **Checked levels** at least one level must be ticked and those levels must have architectural floor plans. In each case a message explains why nothing changed.

Does re-running create duplicate tags? No. Tags are matched to elements by a permanent key, so a re-run updates rather than duplicates. This also survives closing and reopening the project.

Which views get tagged in All levels mode when a level has several plans? The active view if it's on that level; otherwise the plan already containing the most Smart Auto Tagging tags; otherwise the first available architectural plan. One plan per level per run.

Can I keep working in Revit while the dialog is open? Yes. The window is modeless and stays on top. You can change views, select elements, even switch documents — the plugin re-reads the active document and view each time you click Run.