



User Manual

Interactive 3-axis visualization of any CSV — fully local, nothing leaves your machine

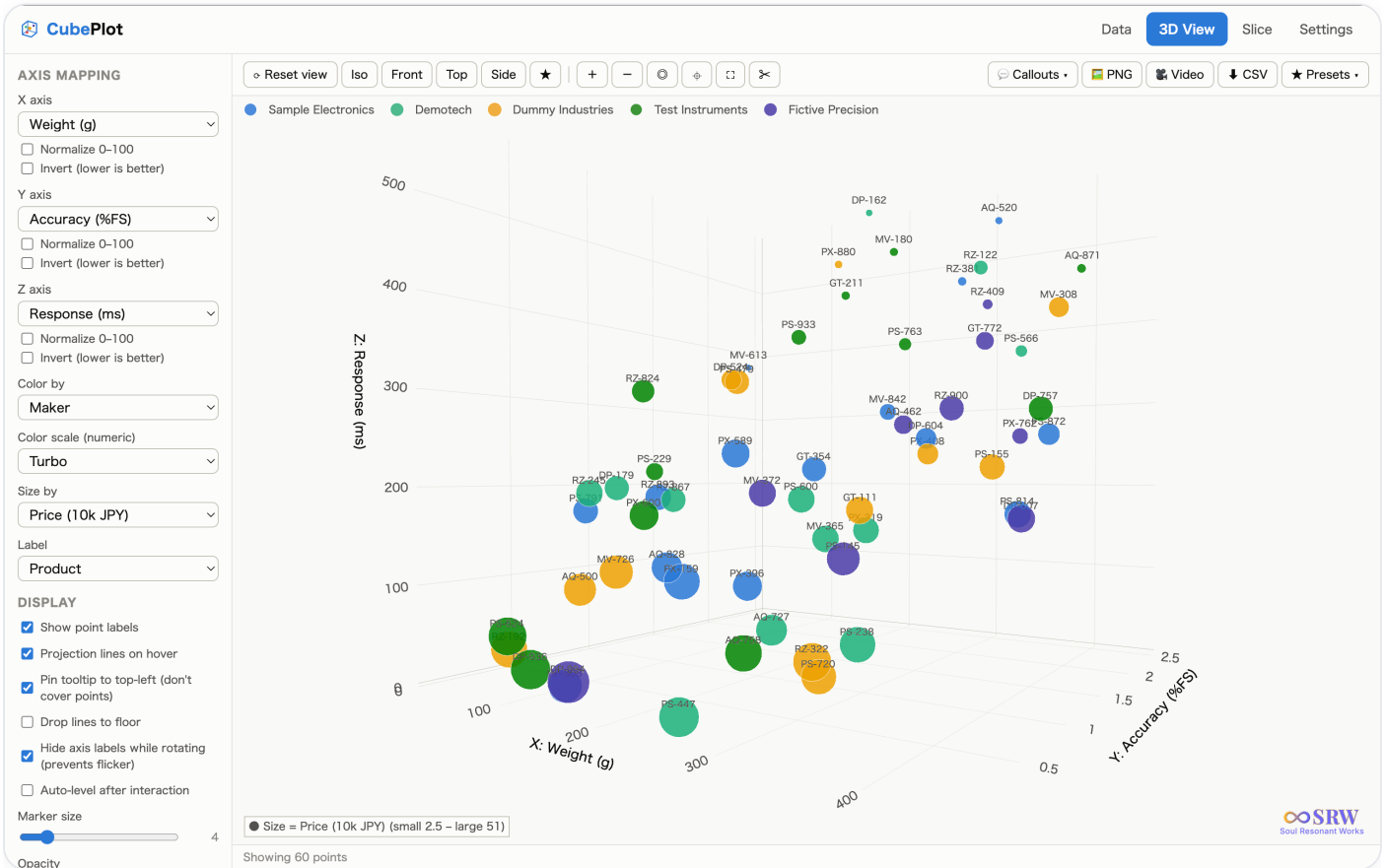
If it's a CSV, it can be 3D.

No installation

A single HTML file

Up to 5 parameters at once

PowerPoint-ready



Version 1.0.0 • built 2026-07-19 • © Soul Resonant Works

Part 1 — Quick Start

CubePlot Quick Start (5 minutes)

1. Launch

Double-click `CubePlot.html` (`CubePlot_free.html` for the free edition). It opens in your browser (Chrome / Edge recommended). No installation is required.

Put the folder anywhere (portable): it runs from any location — even a USB stick — and writes nothing outside its own folder (no registry, no hidden config). Just keep the folder's contents together. If you use it often, bookmark the opened page for one-click launch (re-bookmark if you move the folder). Uninstalling = deleting the folder.

If it does not open: right-click the file → "Open with" → Google Chrome / Microsoft Edge

2. Load a CSV

Any of the following works:

- **Drag & drop** a CSV file onto the window
- Use the "Choose file" button
- To try it first, click "**Load sample data**"

CubePlot

Data3D ViewSliceSettings

Drag & drop a CSV file here

or

Choose fileLoad sample data

Data is processed only on this PC and is never sent to any external server.

How to: 1) Load a CSV 2) Assign columns to X/Y/Z 3) Rotate & zoom in the 3D view

Recent settings

sample_tokyo_buildings.csv	東京駅から東(km) × 東京駅から北(km) × 高さ(m)	7/18/2026
sample_solar_en.csv	X (AU) × Y (AU) × Z (AU)	7/18/2026
sample_solar.csv	X(AU) × Y(AU) × Z(AU)	7/18/2026
sample_fuji.csv	山頂から東(km) × 山頂から北(km) × 標高(m)	7/18/2026
sample_fuji_en.csv	East of summit (km) × North of summit (km) × Elevation (m)	7/18/2026

SRW

Soul Resonant Works

About the sample data

The sample is a fictional comparison dataset built around a "**pressure sensor selection**" use case: 60 candidate products of the same kind (pressure sensors), compared by sensing method and maker plus price, accuracy, response time, and weight.

The goal is to **grasp multi-parameter trade-offs in 3D** — e.g. "fiber-optic: expensive but the most accurate" vs. "MEMS: cheap and light with average accuracy". Note that accuracy (%FS) is **better when smaller**, which also makes it a demo of the "Invert (lower is better)" option.

How to read it:

- Mechanical = cheap but slow and inaccurate
- Fiber-optic = best accuracy and response, but expensive
- Capacitive = balanced, leaning accurate
- Piezoresistive = middle of the road
- MEMS = small, light and cheap, with average accuracy

That is the designed pattern (fictional data imitating real-world tendencies; not actual product specs).

Product	Method	Price (10k JPY)	Accuracy (%FS)	Response (ms)	Weight (g)	Maker
DP-179	Piezoresistive	27.1	0.885	197.3	58	Demotech
RZ-867	Piezoresistive	27.1	1.251	181.8	67	Demotech
AQ-520	Mechanical	3.9	2.479	544.4	308	Sample Electronics
MV-726	Capacitive	39.5	0.703	105.9	90	Dummy Industries

(60 rows in total. All model and maker names are fictitious. Prices are in units of 10,000 Japanese yen. **The category column (Method) is designed as "variants of the same product"** — mixing different kinds of equipment would make the comparison meaningless. Your own CSV loads as-is if it follows "1 row = 1 candidate, 1 column = 1 metric".)

Supported formats:

- Encoding: UTF-8 (with/without BOM) / Shift_JIS (Japanese Excel default) / UTF-16 — **auto-detected**
- Delimiter: comma / tab / semicolon — **auto-detected**

- Header row is **auto-detected** too (a checkbox on the preview screen overrides it)

After loading, the **data preview** appears:

- Check each column's type (numeric / category / text) and its missing-value count
- Missing cells are highlighted in yellow. Rows with missing X/Y/Z values are excluded from the plot


Data
3D View
Slice
Settings

Data preview
sample_products_en.csv
60 rows / 7 columns
Encoding: UTF-8 (BOM)
Delimiter: Comma
☒ Treat first row as header
Open another CSV

✓ No missing values

Product	Method	Price (10k JPY)	Accuracy (%FS)	Response (ms)	Weight (g)	Maker
DP-179	Piezoresistive	27.1	0.885	197.3	58	Demotech
RZ-867	Piezoresistive	27.1	1.251	181.8	67	Demotech
RZ-900	Piezoresistive	27.3	1.016	274.2	371	Fictive Precision
AQ-520	Mechanical	3.9	2.479	544.4	308	Sample Electronics
MV-726	Capacitive	39.5	0.703	105.9	90	Dummy Industries
PS-791	Capacitive	27.9	0.908	169.1	13	Sample Electronics
DP-894	Fiber-optic	51	0.138	20.4	145	Fictive Precision
RZ-192	Fiber-optic	43.4	0.23	34.5	58	Dummy Industries
PX-319	Capacitive	29.3	0.884	159.7	314	Demotech
AQ-727	Fiber-optic	36.2	0.364	80.2	295	Demotech
GT-772	MEMS	18.5	1.515	352.9	361	Fictive Precision
GT-354	Capacitive	27	1.127	218.5	241	Sample Electronics
GT-111	Capacitive	31.2	0.801	179.5	317	Dummy Industries
MV-308	MEMS	21.3	1.381	373.6	428	Dummy Industries
RZ-122	MEMS	13	2.039	460.9	320	Demotech
DP-607	Piezoresistive	31.6	0.766	176.3	438	Fictive Precision
MV-372	Capacitive	31	0.807	194.1	230	Fictive Precision

Showing first 50 rows


Here's what the missing-value highlighting looks like (try it yourself with the bundled `data/sample_missing_en.csv`).

CubePlot

Data

3D View

Slice

Settings

Data preview

sample_missing_en.csv

8 rows / 7 columns

Encoding: UTF-8 (BOM)

Delimiter: Comma

☒ Treat first row as header

Open another CSV

7 missing values (highlighted). Rows with missing X/Y/Z values are excluded from the plot.

Product	Method	Price (10k JPY)	Accuracy (%FS)	Response (ms)	Weight (g)	Maker
PX-101	Piezoresistive	3.5	0.35	1.2	45	Fictive Precision
PS-202	Capacitive	—	0.15	8.0	95	Sample Electronics
DP-303	Fiber-optic	28.0	—	25.5	180	Dummy Industries
GT-404	MEMS	1.2	0.8	—	8	Test Instruments
MV-505	Mechanical	N/A	1.8	220	320	Demotech
KF-606	Piezoresistive	4.8	0.28	2.5	-	Fictive Precision
AQ-707	—	8.9	0.12	12.0	110	Sample Electronics
RZ-808	Capacitive	10.5	0.09	15.0	140	—

SRW

Soul Resonant Works

3. Assign axes

Switch to the "3D View" tab in the top menu. The left sidebar holds axis mapping (X/Y/Z), color/size/label column pickers, and display options (projection lines, tooltip, drop lines to floor, etc), from top to bottom.

AXIS MAPPING

X axis

Weight (g)

☐ Normalize 0–100
 ☐ Invert (lower is better)

Y axis

Accuracy (%FS)

☐ Normalize 0–100
 ☐ Invert (lower is better)

Z axis

Response (ms)

☐ Normalize 0–100
 ☐ Invert (lower is better)

Color by

Maker

Color scale (numeric)

Turbo

Size by

Price (10k JPY)

Label

Product

DISPLAY

☒ Show point labels
 ☒ Projection lines on hover
 ☒ Pin tooltip to top-left (don't cover points)
 ☐ Drop lines to floor
 ☒ Hide axis labels while rotating (prevents flicker)
 ☐ Auto-level after interaction

Marker size

4

Opacity

Item	Description
X / Y / Z axis	Pick numeric columns (auto-assigned on load)
Normalize 0–100	Makes columns with different units easier to compare
Invert (lower is better)	For metrics like accuracy (%FS) or response time where smaller is better. The axis direction flips while tick values stay original, so "better" points sit higher / deeper
Color by (student/corporate-license editions only)	(Auto) = colors by a category column if present, else a Z-axis gradient. Picking a column: category → discrete colors + legend / numeric → gradient
Color scale	Gradient used for numeric coloring: Blue (default) / Viridis / Turbo / Plasma . All are scientific-visualization standards designed so lightness changes monotonically with value, which makes magnitudes hard to misread (Viridis = navy→green→yellow, Turbo = navy→cyan→green→yellow→red rainbow, Plasma = navy→purple→orange→yellow)
Size by (student/corporate-license editions only)	Scales bubble size by a numeric column. While active, a size legend (column name and min–max range) appears at the bottom-left of the plot and is included in PNG export, fullscreen, and recordings
Label	The column used as the headline of tooltips and callouts (product name etc.). If you pick a numeric column it is shown as "Weight (g): 2.59", with the column name
Projection lines on hover	Hovering a point draws guide lines to the three walls, making values easier to read
Pin tooltip to top-left	Shows point details in a fixed top-left panel instead of next to the point (turn it off to show the tooltip right next to the point instead)
Drop lines to floor	Draws vertical lines from every point to the floor to convey depth (up to 2,000 points)

How much can one chart carry? — up to 5 parameters + label

The strength of a 3D scatter plot is how much information fits in one screen. CubePlot lets you assign **five channels at once: X, Y, Z, color, and size**.

A column's type determines which roles it can take:

Column type	Possible roles	Example in the sample data
Numeric	X / Y / Z axis, size, color (gradient)	Price (10k JPY), Accuracy (%FS), Response (ms), Weight (g)
Category (repeating strings)	Color only (discrete colors + legend) — no ordering, so not usable for axes/size	Method, Maker
Text (nearly unique per row)	Label (tooltip headline, callouts, point labels)	Product

"Everything on" example (sample data): X = Price, Y = Accuracy (%FS) (invert recommended), Z = Response, Color = Method, Size = Weight, Label = Product — one point carries **six pieces of information**: which product, at what price, how accurate, how fast, by which method, and how heavy.

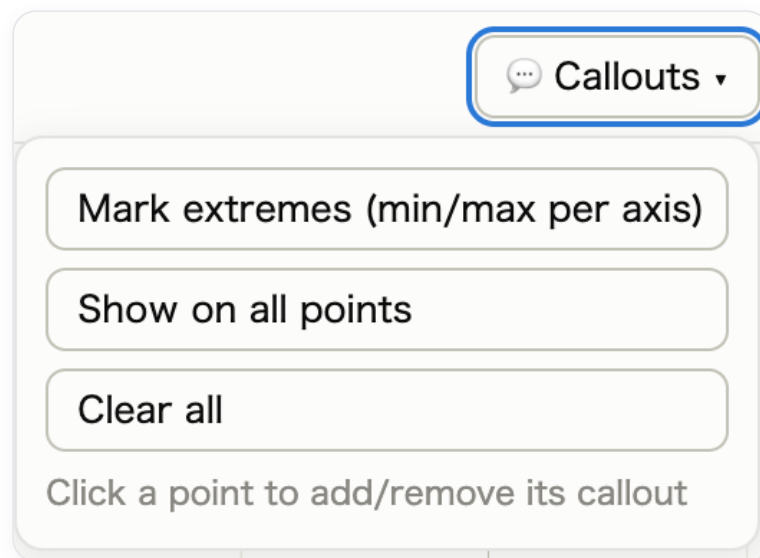
Note: only one column can drive color (you cannot color by "Method" and "Maker" at the same time). Switch between them, or narrow down with the Slice tab first.

4. Interact

Action	Button	Mouse	Trackpad
Rotate	—	Drag. Horizontal: unlimited (as many turns as you like); vertical: up to $\pm 90^\circ$ (stops at straight-up / straight-down so the view never tilts or flips). To see "past the top", rotate 180° horizontally first, then adjust the vertical angle	Click-drag. A two-finger horizontal swipe also rotates around the center
Zoom	"+" / "-" buttons	Wheel	Two-finger vertical swipe, or pinch (strong)
Pan	—	Right-drag	Press with two fingers (secondary-click state) and slide, or Ctrl+click+drag
Re-center rotation	The "⊕" button	—	—
Orbit around a point	Press "◎", then click a point	—	—
Fullscreen	The "□" button (Esc to exit)	—	—
Details (tooltip)	—	Hover a point (label plus each axis's original values)	Same
Callouts	The "☰ Callouts" menu (Feature points = min/max per axis, All points = up to 50, Clear all)	Click a point to pin a label with a leader line (click again to remove)	Same
Switch views	"Iso / Front / Top / Side" buttons (the initial view is "Iso")	—	—
Save & restore a view	"★" to remember the current angle → " Reset view " snaps back any time	—	—

Action	Button	Mouse	Trackpad
Legend	—	Click a legend entry to toggle that category's visibility	Same
Alternate pan/zoom (keyboard)	—	Ctrl+drag pans / Alt+vertical drag zooms (control / option on a Mac)	Same

Pressing the "🗨 Callouts" button opens this menu:



About "Re-center rotation" (⊕): the rotation pivot defaults to **the center of your entire dataset**.

Panning moves the pivot along with the view, which makes subsequent rotation feel like being "swung around". ⊕ returns the pivot to the data center **while keeping the current direction and zoom** — this is also how you **release a ◎ point-pivot**.

About "Orbit around a point" (◎): the point moves to the screen center and rotation then orbits around it (handy for "rotate around the Sun" in the star map, or inspecting a trimmed region). "⊕" returns to the data center. Note: **panning effectively releases the pivot** (panning moves the orbit center by definition) — press ◎ again or ⊕ to recover. **Tip for clicking a point more easily:** turn off "Projection lines on hover" — the guide lines can otherwise get in the way of the click.

Where is the rotation axis? — Rotation always orbits **the point at the center of the screen** (the look-at point). On startup that is the center of your data. **Panning shifts what sits at the screen center**, **⦿ puts a chosen point there**, and **⊕ puts the data center back**. If rotation feels like being "swung around" after panning, it is because the screen center now sits in empty space — the rotation itself is still faithfully orbiting that one point. When in doubt, press **⊕**. Also note: panning until a point *looks* centered does not put the axis on that point — the axis can still be displaced **in depth** (the screen is 2D, the axis is a 3D point). To pivot exactly on a specific point, use **⦿** rather than eyeballing with pan.

Where to interact: rotation drags work the same in the empty space around the plot. Since interacting on top of points can accidentally add callouts, **we recommend navigating from the margins.**

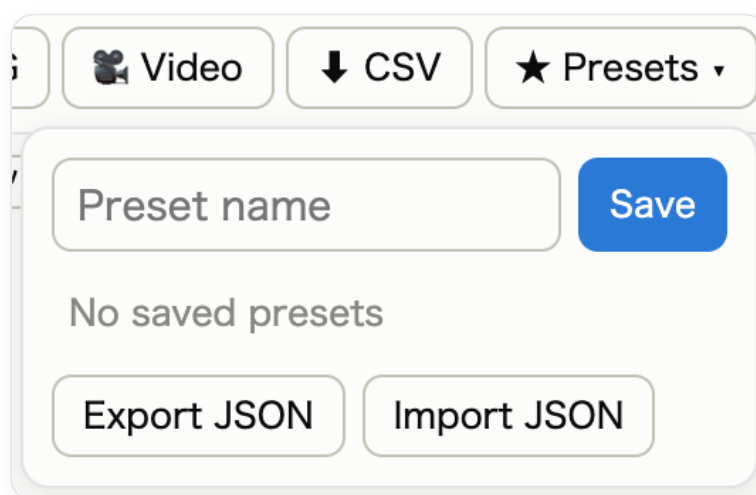
If the plot ever looks tilted (horizon off), use a view preset such as "Iso" or "Reset view" to level it. Enable "Auto-level after interaction" in the sidebar to keep it level at all times (off by default).

Due to how 3D works, axis labels jump to the most readable wall as you rotate. By default CubePlot **hides axis labels only while rotating and shows them the moment you stop**, which suppresses the flicker. Turn off "Hide axis labels while rotating" in the sidebar if you want them always visible.

5. Export

Button	What it does
Export PNG	Saves the current view as a high-resolution image (CubePlot at the top-right, SRW logo at the bottom-right. The free edition additionally adds a watermark and QR code)
Record video (student/corporate-license editions only)	Saves a rotation video (MP4) in which the camera makes a full turn starting from the current view . The rotation axis is the same as interactive rotation — the screen center (with a  point-pivot set, the video orbits that point at constant distance; press  first to orbit the whole dataset). Keep the tab visible while recording. Axis labels that would jump around are hidden, and instead the legend, X/Y/Z column names, size legend, and a rotation-synced XYZ direction gadget are burned into the video (no flicker, and you always know which axis points where). CubePlot top-right, SRW logo bottom-right. Frames are generated one by one, so the finished video is always smooth regardless of your PC's performance (on slower machines, exporting may take longer than the video's duration)
Visible data CSV	Saves the data currently shown in the 3D view (missing rows excluded, after sampling) as a new CSV. The original file is never modified. Note: setting slice ranges without "apply to 3D view" leaves the 3D view (and hence this CSV) unfiltered . For filtered data, use "Export filtered CSV" on the Slice tab, or turn on "apply to 3D view" first
Presets	Save and recall axis assignments, invert/normalize options, marker size/opacity, color scale, font size, the current camera angle and slice ranges by name (trim  is not included). Export as JSON to share with colleagues. With the local server, the  button in the list copies a ready-made URL that opens the 3D view with that preset applied (for PowerPoint links)

Pressing the "★ Presets" button opens this panel:



 Video
  CSV
 ★ Presets ▾

No saved presets

Note: the free edition (CubePlot Free) offers X/Y/Z 3D view and PNG export only. A watermark is shown on screen at all times, and exported PNGs carry a watermark plus a QR code. **Color by / Size by and video recording are student and corporate-license edition features** (their controls appear greyed out in the free edition).

Video quality and file size

Choose the bitrate under Settings → "Quality (bitrate)" (see "Conveniences" further below for the full Settings tab reference). File size is bounded by "duration × bitrate", and the Settings tab shows a live estimate (scenes that compress well, such as dark backgrounds, can come out smaller than the estimate).

About on-screen motion while exporting: recording is not a live capture — frames are **generated and encoded one by one**, so how fast the plot spins on screen during export depends on your PC (a fast machine may finish a 20-second video in a few seconds). **The finished video always has exactly the duration and turns you chose**, so judge by the output file, not the on-screen speed.

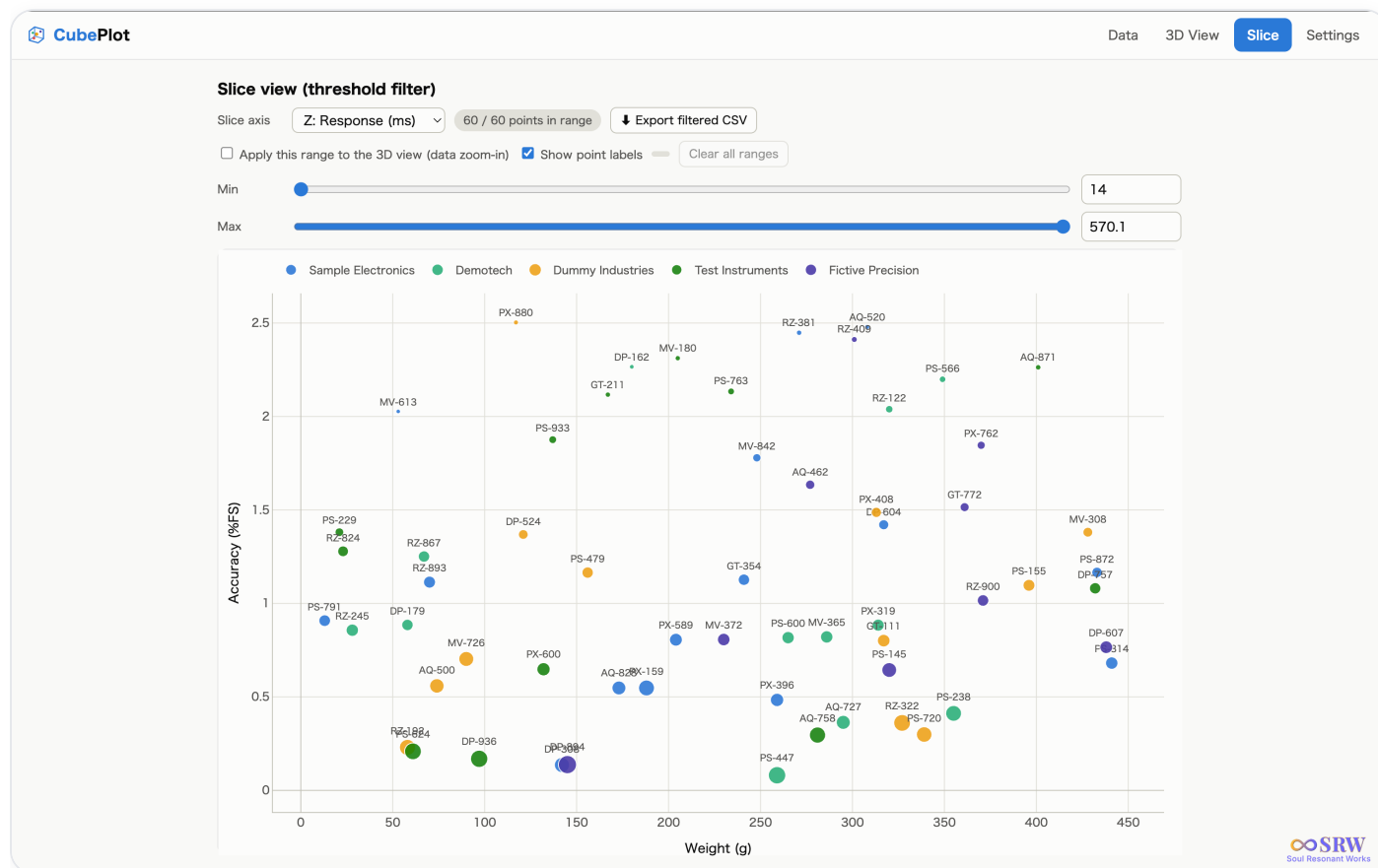
Quality	8 s	15 s	30 s
Standard (4 Mbps)	~4 MB	~7.5 MB	~15 MB
High (8 Mbps, recommended)	~8 MB	~15 MB	~30 MB
Maximum (12 Mbps)	~12 MB	~22.5 MB	~45 MB

Any of these sizes inserts fine into PowerPoint (Microsoft's own recommendation is "H.264 MP4 at 8 Mbps or below", which CubePlot's output follows).

Video resolution is normalized to **Full HD (1920 px wide) by default**, so recordings look the same on any display and quality per megabyte stays consistent (changeable in Settings).

6. Slice view and "data zoom-in"

The "Slice" tab **filters by a value range on one axis** and shows the remaining two axes as a 2D scatter. "Export filtered CSV" saves just the filtered data. Turn on "**Show point labels**" to name every point, so you can locate your targets before carving (synced with the 3D view's label setting).

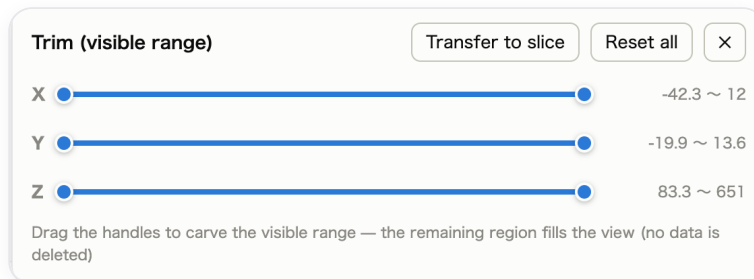


Ranges are **remembered per axis**, and **every axis you set applies simultaneously** (AND condition). Set ranges while switching the slice axis $X \rightarrow Y \rightarrow Z$ and you can carve out (in the star-map sample) "a cube of ± 20 parsecs on X, Y and Z" and similar selections. Active ranges are shown like "Active: X: 10–20 / Z: 0–5", and the "**Reset all ranges**" button clears them at once.

Turn on "**Apply to 3D view (data zoom-in)**" and the same filter applies to the 3D view, where **axes rescale to the narrowed range and the dense region fills the screen**. While camera zoom (+ button / pinch) "moves closer", this zoom **narrows the range of data being shown** (ideal for cutting a cube around the Sun in the star-map sample).

Trim — carve the visible range by dragging

Press "✂" in the 3D view toolbar and three sliders (X/Y/Z) appear at the bottom-left. Drag the handles and **the visible range is carved away in real time, with the remaining region expanding to fill the view** (on large datasets this automatically switches to applying when you release the handle).



- Cutting out "a ± 15 -parsec cube around the Sun" in the star map takes just a few drags
- Drag the panel's title bar to **move it anywhere** on screen (tuck it into a corner if it overlaps the plot)
- Handles support **keyboard fine-tuning**: click (or Tab to) a handle, then use the arrow keys (0.5% steps; Shift+arrows for 0.05%) — handy when hunting an exact edge
- Axis titles in the 3D plot carry an **X/Y/Z prefix** ("X: Accuracy (%FS)"), so even when rotation swaps the apparent positions of the X and Y axes, the mapping to the panel's X/Y/Z sliders stays obvious (hovering the X/Y/Z letters in the panel also shows the assigned column names)

Trim vs. Slice — which to use

Both narrow things down by range, but **they act on different layers**:

	✂ Trim	Slice tab
What changes	Only how the 3D view looks	The data itself is filtered
Data	Nothing is deleted (all points kept)	Out-of-range rows are excluded
CSV export	Unaffected ("Visible data CSV" still contains every point)	"Export filtered CSV" saves the carved-out subset
Range memory	Momentary ("Reset all" restores)	Remembered per axis, applied as an AND across axes
Best for	Temporarily hiding outliers, peeking inside clusters	Extracting a subset for analysis or sharing

Think of it as: ✂ **carves the view for observation**; Slice **cuts the data for export**. Explore with ✂ first, then press **"Transfer to slice"** in the panel — the same range is applied to Slice instantly (axes under 0–100 normalization are converted back to real data values automatically, so there's no need to note down numbers and retype them). Then just export the CSV from the Slice tab's "Export filtered CSV".

Callouts and trimming: "Feature points" and "All points" only target points **inside the trimmed range** while trimming is active (invisible points never consume the limit). Likewise, while the Slice tab's "apply to 3D view" is on, only the filtered points are targeted.

Conveniences

- **Recent settings:** reloading a CSV with the same column structure restores your last axis assignment

Settings tab

CubePlot

Data3D ViewSliceSettings

Settings

Language

English

Theme

Light

Max points (sampled beyond this)

20000

Tooltip decimal places

2

PNG export scale (2–5)

3

Chart text size

Standard

Video recording

Duration (seconds)

8

Rotations (1–5)

1

Quality (bitrate)

High (8 Mbps, recomr

Video resolution

Full HD (1920 px wide

容量の目安: 8秒の録画で約7.6MB

Data handling

This app is fully local. Loaded data is processed only in browser memory and never sent to external servers. Only settings such as presets are stored in this browser's local storage.

Reset settings

Version: 1.0.0

SRW

Soul Resonant Works

Item	Description
Language	Japanese / English (auto-detected from your browser on first launch)
Theme	Light / Dark / Auto (follows the OS setting)
Max points	Data beyond this is automatically sampled down to keep things responsive
Tooltip decimal places	Number of decimal digits shown in the details (tooltip)
PNG export scale	2–5×. Higher is sharper but takes longer to export
Chart text size	Standard / Large / Extra large — scales axis labels, point labels, callouts, and the legend together (handy for projector screens)
Video recording: duration	Seconds (default 8)

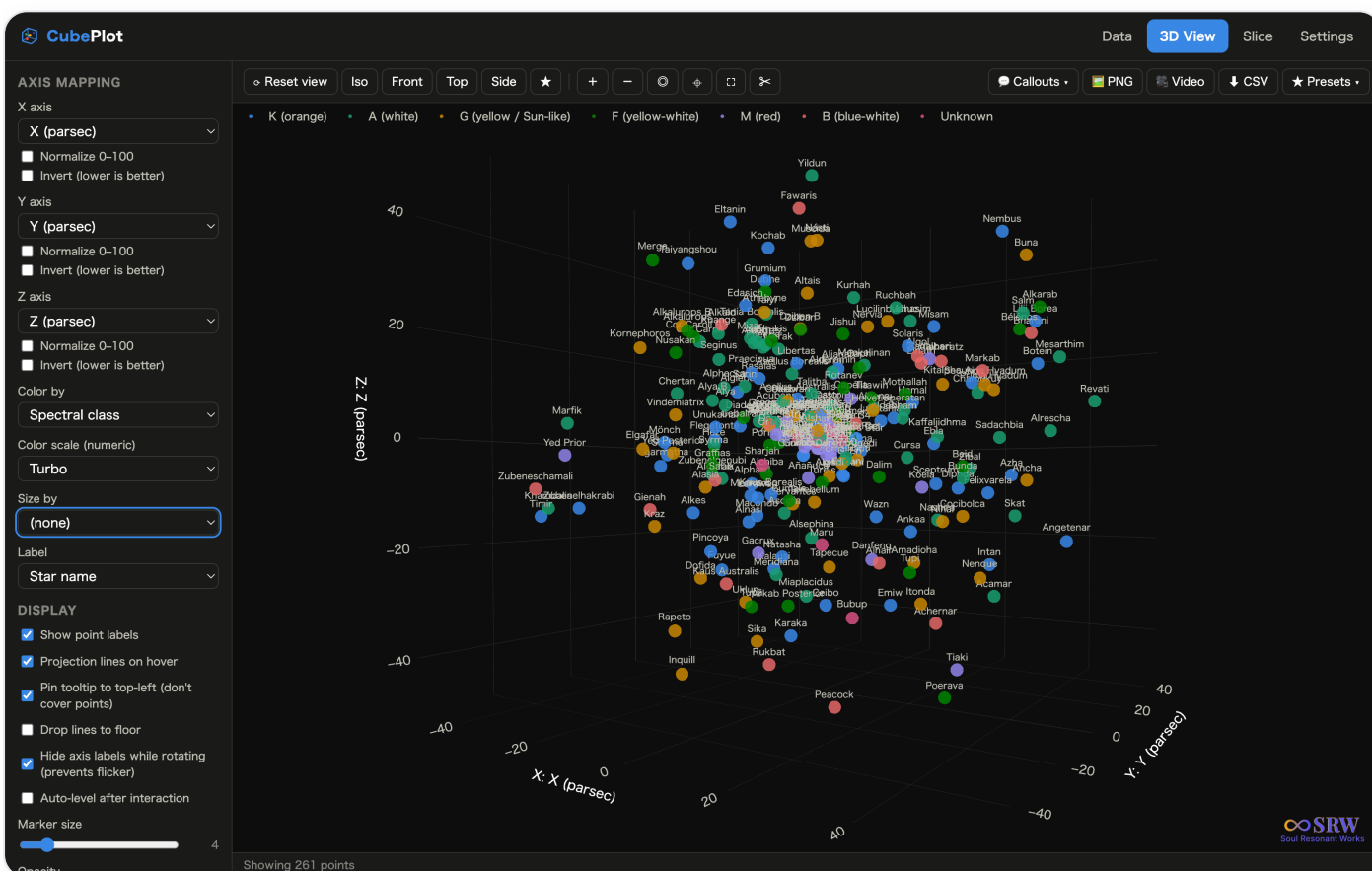
Item	Description
Video recording: rotations	1–5 turns
Video recording: quality (bitrate)	Standard (4 Mbps) / High (8 Mbps, recommended) / Maximum (12 Mbps)
Video recording: resolution	Full HD (recommended) / WQHD / Native
Data handling	This app is fully local — loaded data is processed only in browser memory and never sent to external servers. Only settings such as presets are stored in this browser's local storage
Reset settings	Restores every setting to its default

If it's a CSV, it can be 3D — gallery

CubePlot is not just for business data. **Any data becomes an interactive 3D scene as long as you shape it into a "1 row = 1 point, 1 column = 1 value" CSV.** Five bonus samples built from freely available public data are bundled in the `data/` folder (drag & drop them, or via the server use a URL like `?csv=data/sample_stars_en.csv`).

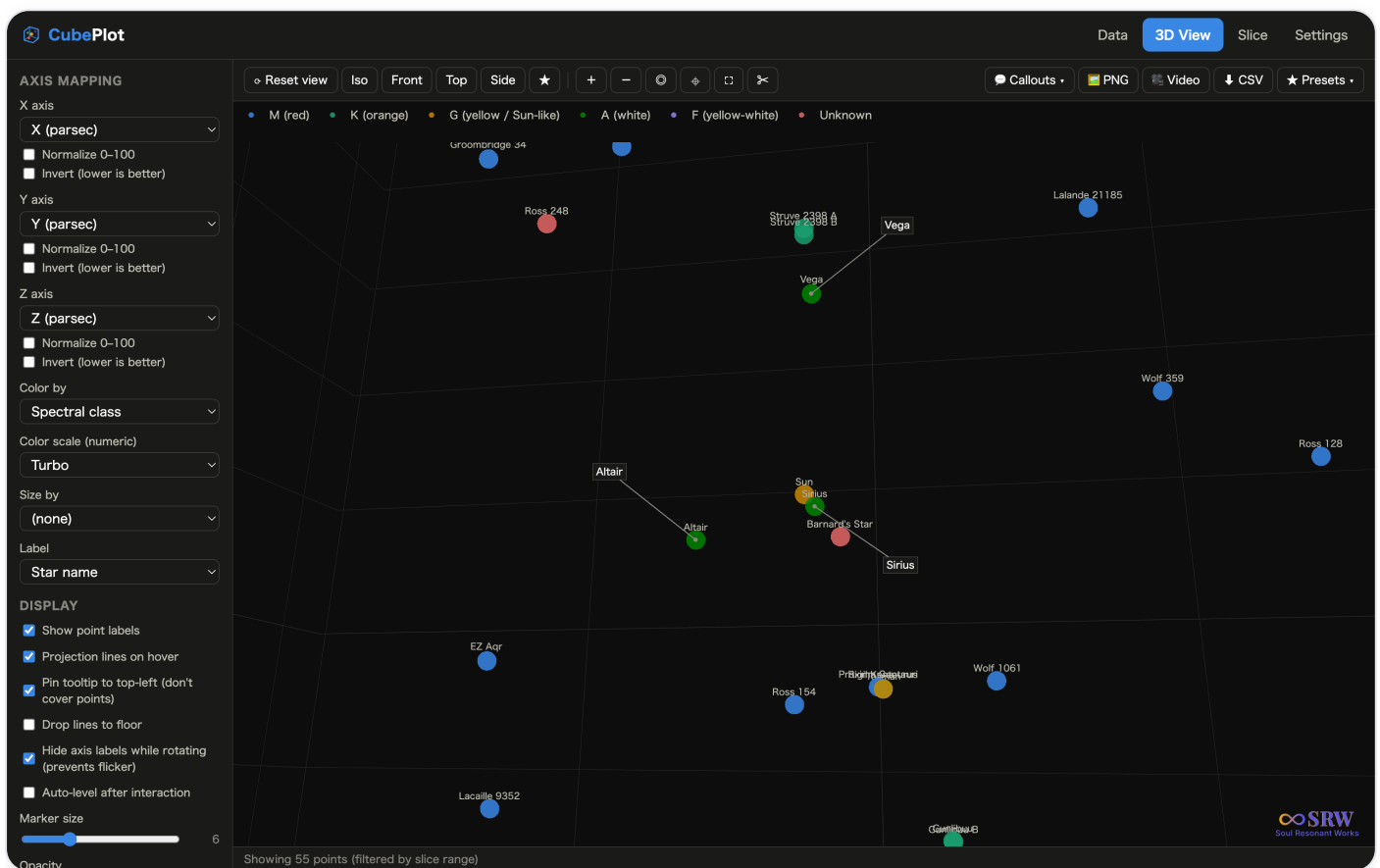
3D star map — 261 stars near the Sun (sample_stars_en.csv)

Named stars within 60 parsecs (~200 light-years) of the Sun. Assign **X/Y/Z = position in space (parsec)**, **Color = spectral class**, **Size = luminosity**, **Label = star name** and you get a 3D planetarium with Sirius and Vega floating in it (dark theme recommended). **The origin (0,0,0) is the Sun** — there is a "Sun" point in the data as a landmark. Stars appear to move nightly only because the Earth rotates; **the true 3D positions of stars do not change on human timescales** — a static 3D map is accurate.



This also explains why different constellations dominate different seasons: the Summer Triangle (Vega, Altair) and winter's signature star Sirius sit in almost exactly opposite directions from the Sun (about 160° apart). Earth's night side always faces away from the Sun, and as Earth orbits, that direction reverses by roughly 180° every six months — so stars lying in opposite directions from the Sun each take their turn dominating the night sky, six months apart. The 3D layout itself is the answer to "why do the stars change with the seasons?"

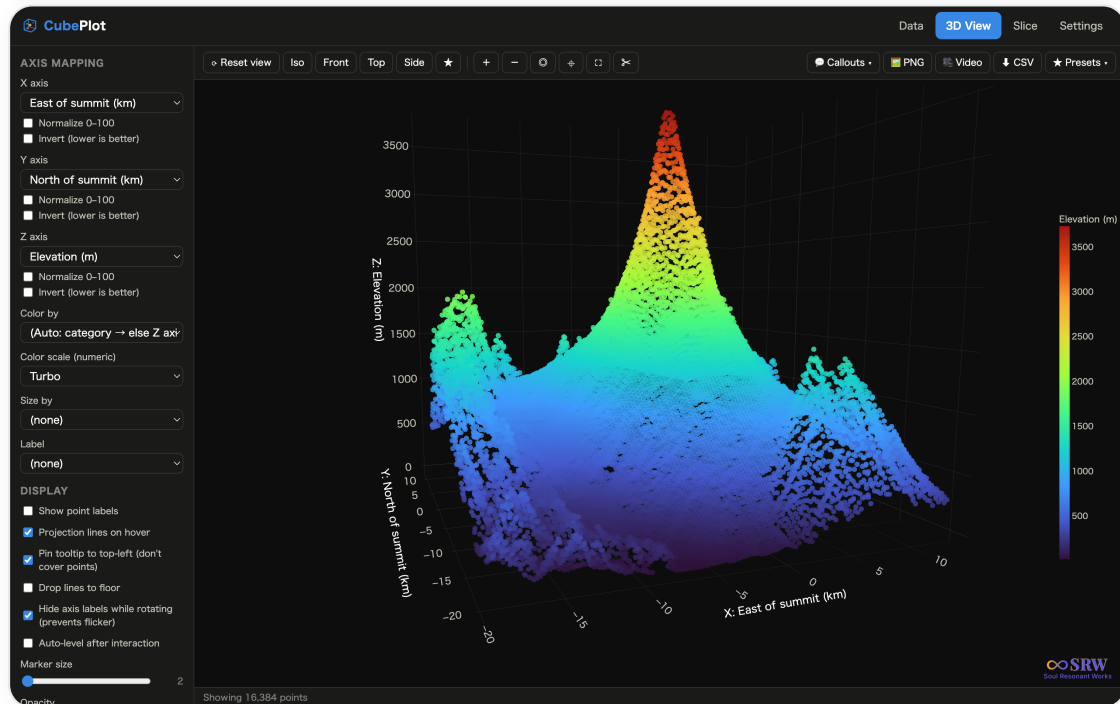
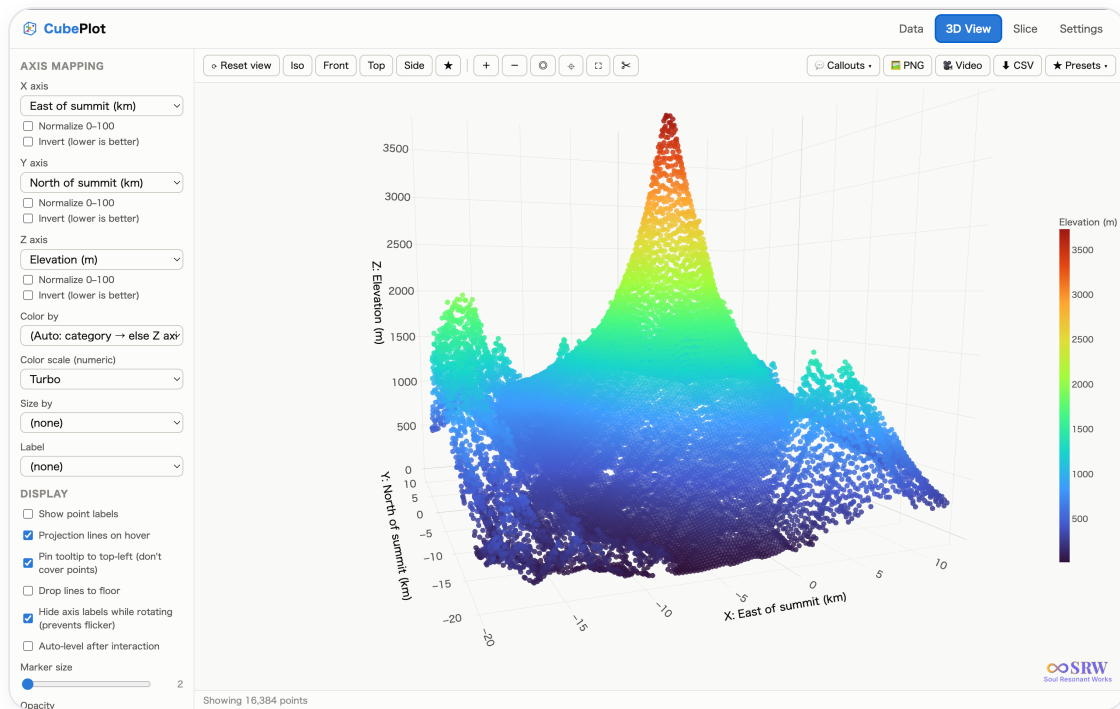
The real Summer Triangle actually has three stars — Vega, Altair, and Deneb — but this dataset only covers stars within 60 parsecs of the Sun, and Deneb sits hundreds of parsecs away, well outside that range, so it isn't included here. Three bright, familiar stars can sit at wildly different real distances — another thing 3D makes obvious that a flat sky map cannot.



3D Mt. Fuji — terrain from elevation data (sample_fuji_en.csv)

Elevation of a ~32 km square centered on the summit of Mt. Fuji, at a ~250 m mesh (16,384 points). **X = east of summit (km), Y = north of summit (km), Z = elevation (m)** — as the column names say, **the origin is the summit**; positive is east/north. Leave color on "(Auto)" for an elevation gradient. Lowering marker size to ~3 sharpens the terrain (heights are exaggerated because the plot box is a cube).

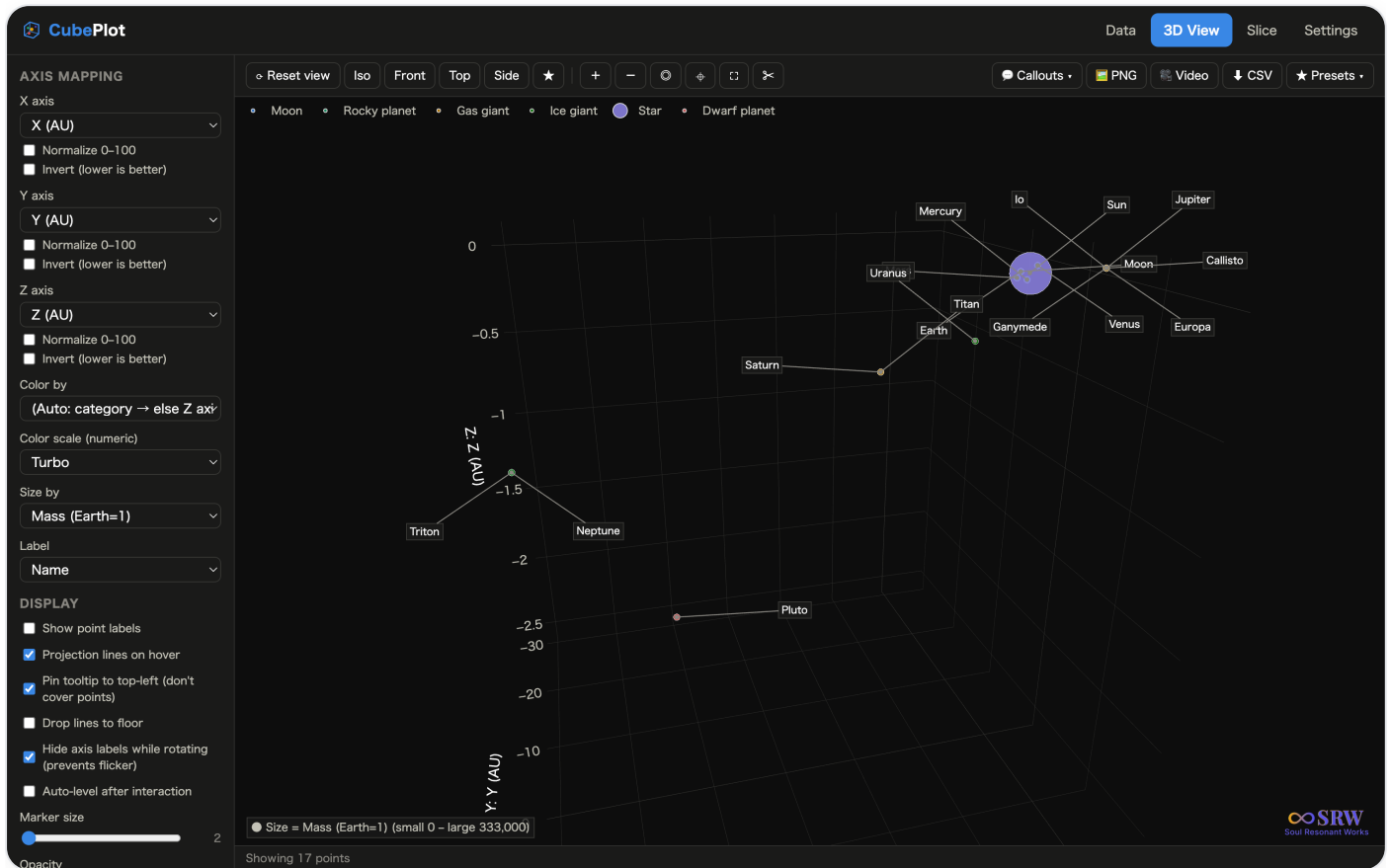
Light (left) and dark (right) — the background color changes the impression quite a bit.



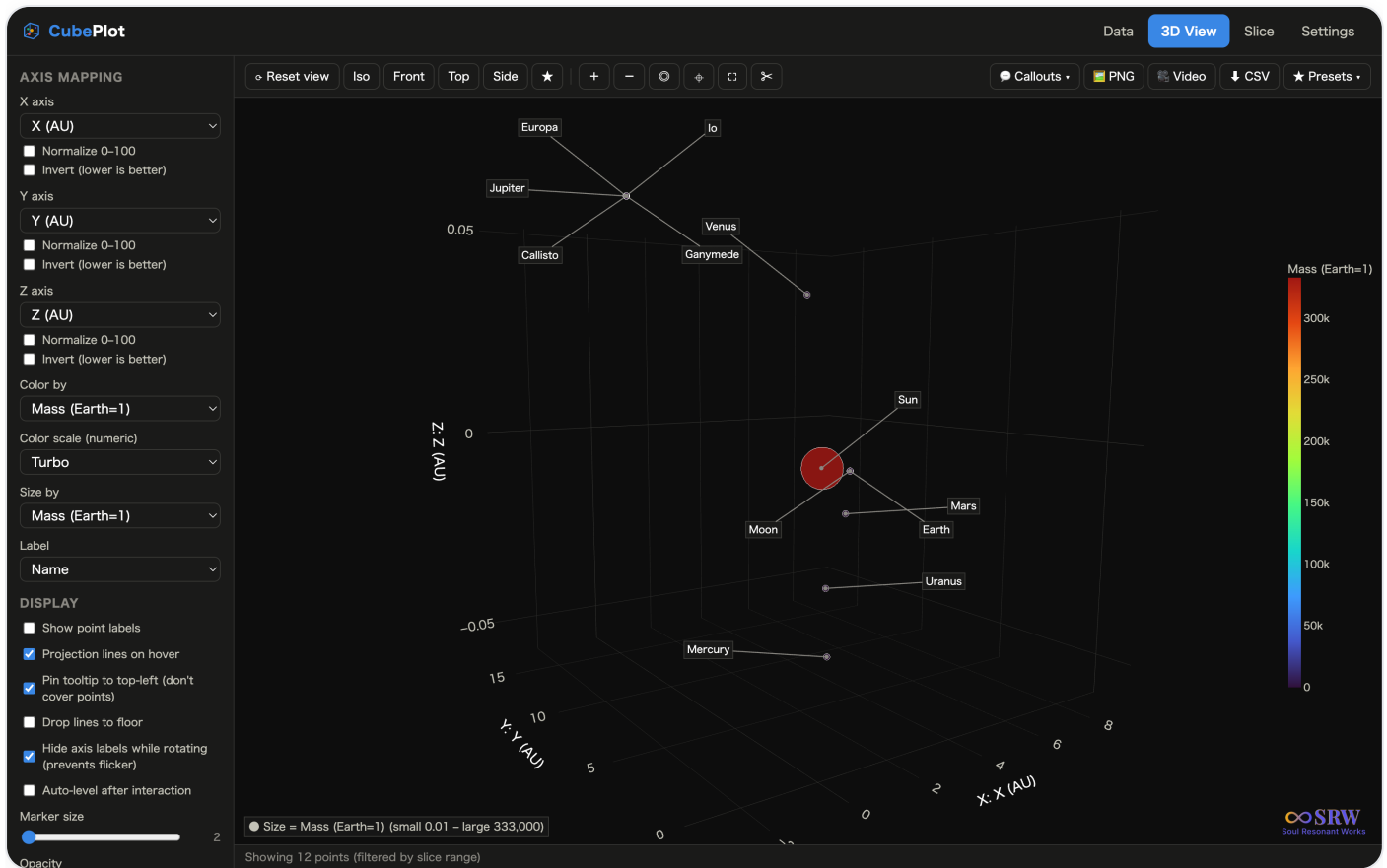
Solar system map (sample_solar_en.csv)

17 bodies: the Sun + 8 planets + Pluto + 7 famous moons (the Moon, the Galilean moons, Titan, Triton). Includes **approximate heliocentric ecliptic positions as of July 2026**, so the actual arrangement becomes a 3D scene.

Recommended: X/Y/Z = X/Y/Z (AU), Color = Class, Size = Diameter (log10 km), Label = Name.



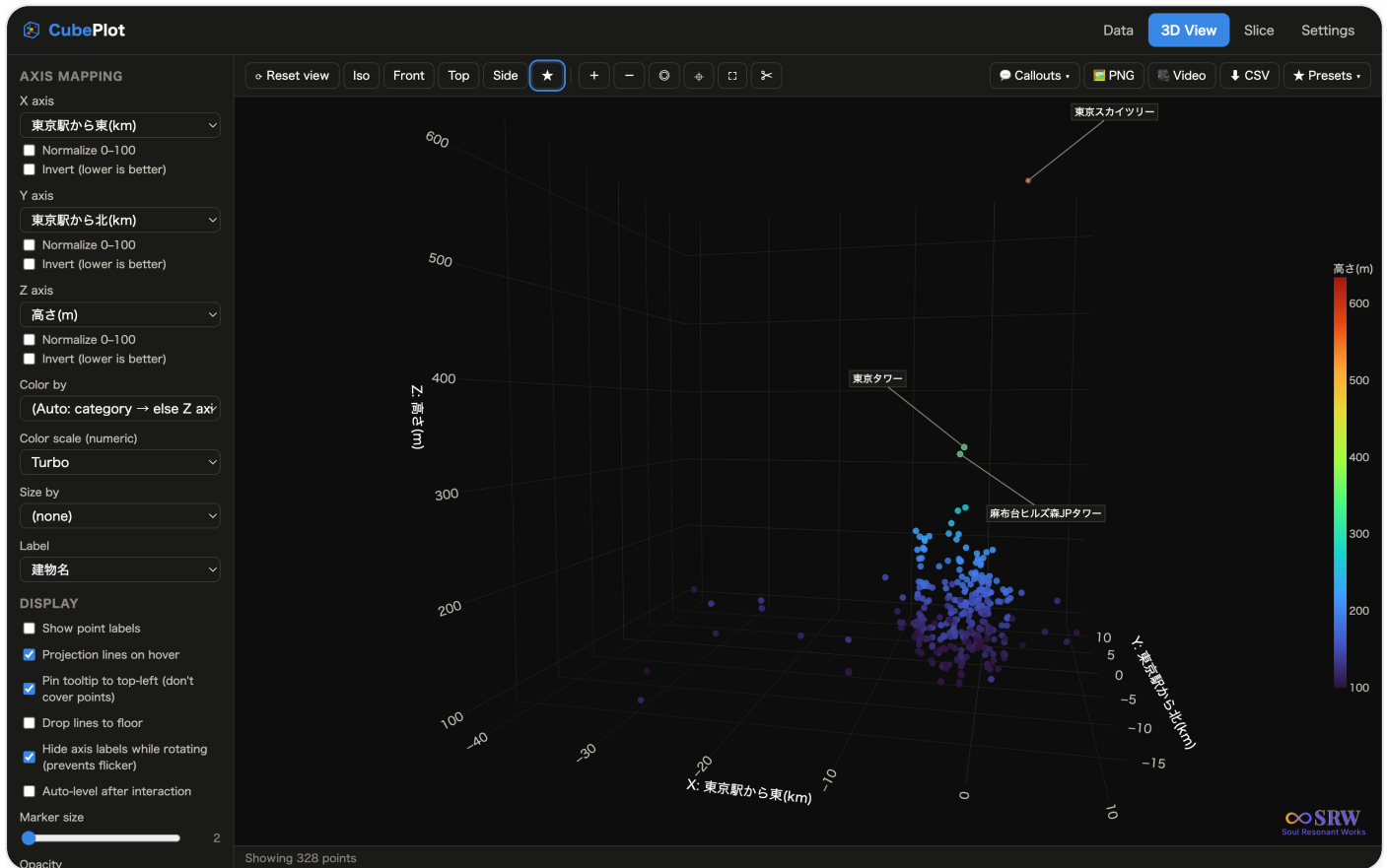
- You can see at a glance that the solar system is **nearly flat** (almost no spread in Z — a lesson in itself; only Pluto floats off the plane)
- Inner planets crowd the center — use the Slice tab's **data zoom-in with X (AU) around ± 2** to expand Mercury–Venus–Earth–Mars



- Moons are placed right next to their parent planet (offset by their real orbital radius; hover Jupiter and the Galilean moons are just beside it)
- Distances that are hard to feel from a table of numbers become obvious in 3D. **Even the Moon sits noticeably far from Earth** in this view — and by the time you reach Saturn, Uranus, Neptune and Pluto, the gaps have grown by orders of magnitude

Tokyo high-rise map (sample_tokyo_buildings.csv)

328 buildings and towers of 100 m or taller across Tokyo (from OpenStreetMap). **X = east of Tokyo Station (km)**, **Y = north of Tokyo Station (km)**, **Z = height (m)**, **Label = building name** — the origin is Tokyo Station; positive is east/north. With "(Auto)" color you get a height gradient: the skyscraper clusters of Shinjuku, Marunouchi and Toranomon–Azabudai, with Tokyo Skytree (634 m) and Tokyo Tower (332.9 m) standing head and shoulders above. Try "data zoom-in" to carve out one area. *(This sample is Japanese-labeled — building names and headers are in Japanese.)*



Data sources: Stars = HYG Database v4.1 (David Nash / astronexus.com, CC BY-SA 4.0), adapted. Terrain = GSI (Geospatial Information Authority of Japan) elevation tiles, adapted. Solar system = computed from published physical constants. Tokyo buildings = © OpenStreetMap contributors (ODbL), adapted (Tokyo Tower height corrected to the official 332.9 m). See the bundled [data/データ出典.txt](#) for details (bilingual).

Troubleshooting

Symptom	Remedy
Garbled characters	Encoding auto-detection failed. Try in order: ① open the file in Excel and "Save As" → format " CSV UTF-8 (comma delimited) ", then reload ② open in a text editor and save with UTF-8 encoding ③ if it is garbled even in Excel, the source file itself is broken — re-export it from the original system as UTF-8
No points appear	Check that X/Y/Z are numeric columns and not mostly missing values
Saved presets are missing / JSON export comes out empty	Preset storage is separate per way of opening the app (double-click = file:// and server = http:// are different stores). Look in the window opened the same way you saved them. Presentation links (🔗) embed their content and are unaffected
CSV files are greyed out in the file dialog	A known quirk of the macOS file dialog (often right after a file is saved or copied; not controllable from the app). Cancel and reopen the dialog and the files become selectable. Drag & drop always works
Video comes out as WebM	Your browser cannot record MP4. Use the latest Chrome
Sluggish performance	Lower "Max points" in the Settings tab (automatic sampling)
The 3D view behaves oddly (clicks misfire, hovering alone changes something)	A browser extension may be interfering. Reopen the file in an Incognito/Private window (<code>Cmd/Ctrl+Shift+N</code> in Chrome) and see if it improves. If it does, disable your extensions one by one to find the culprit

Part 2 — PowerPoint Integration Guide

PowerPoint Integration Guide

There are three ways to show CubePlot's 3D scatter plots in a PowerPoint presentation, explained in **recommended order**.

Important background (as of 2026) The old "Web Viewer" add-in was **discontinued in December 2024**, and the legacy "LiveWeb" add-in no longer works in current PowerPoint. A true live embed inside a slide is therefore not possible with official Microsoft means today. Methods 1 and 2 below are the practical answers.

Method 1: Insert a rotation video (MP4) ★ most reliable

Export a rotation video from CubePlot beforehand and place it on the slide as a video. If all you need is the animated view, this is sufficient — and it does not depend on the presentation environment.

1. Load your data in CubePlot and adjust the angle and colors you want to show
2. In the Settings tab, adjust "Duration" (default 8 s) and "Rotations"
3. Click "📹 Record video" in the 3D view toolbar
 - **Keep the browser tab visible while recording**
 - The file is saved as MP4 (WebM on some environments — see "WebM → MP4" below)
4. In PowerPoint: **Insert → Video → This Device**, pick the saved MP4
5. Set playback to "**Automatically**" + "**Loop until Stopped**" and the plot keeps rotating for as long as the slide is shown

WebM → MP4 conversion (if your browser cannot record MP4):

```
ffmpeg -i orbit.webm -c:v libx264 -pix_fmt yuv420p orbit.mp4
```

(If you don't have ffmpeg, use any video converter you're permitted to use.)

Method 2: One click from the slide into the browser ★ for live Q&A

If you want to rotate/zoom live or inspect arbitrary points during Q&A, put a link on the slide and open the browser in fullscreen with one click.

2-a. Single-file setup (no server, simplest)

1. Place `CubePlot.html` anywhere on the presentation PC (e.g. `C:\CubePlot\CubePlot.html`)
2. In PowerPoint, add a shape or button → right-click → **Link** → "Existing File or Web Page", point it at `CubePlot.html`
3. During the talk: click → the browser opens → drag & drop your CSV
 - **Tip:** if the CSV has the same column structure as last time, the axis assignment is restored automatically
4. When done, Alt+Tab back to PowerPoint

2-b. Local-server setup (data pre-loaded, one click to a live 3D view)

Who this is for: presentations that **switch between several different 3D plots** — say, showing marketing actions vs. prospect responses per product, one plot per product, because mixing everything into one plot would hurt readability.

With method 2-a (single file) you would open several browser tabs beforehand, arrange each view, and keep them alive in the background for the whole talk — close a tab by accident and the preparation is gone.

With the local server you **turn each shot into a preset (link) beforehand**, so no browser needs to stay open. On the day, just start the bundled server: **every click on a slide link reopens the desired plot with the desired view (axes, camera, colors, slice filter) — as many times as you like**. Closed it by mistake? Click again.

A small amount of preparation lets one click land directly on a **3D view with the data already loaded**.

1. Copy the CSV you want to show into a `data` folder next to `CubePlot.html` (e.g. `data/products.csv` ; create the folder if absent)
2. **Compose the shot and save it as a preset:** open that CSV in the app, set the axis assignments, invert/normalize options and marker size, **rotate/zoom to the exact angle you want**, then toolbar "★ Presets" → save under a name (e.g. `sensor-hero`)
3. Start the server (before the talk — and **keep it running during the talk**; the links connect to it, so nothing opens while it is stopped):
 - Windows: double-click `start_server.bat` (uses `serve.py` if Python exists, otherwise falls back to the PowerShell server that ships with Windows — **nothing to install either way**)

- Mac: double-click `start_server.command` (first time: right-click → "Open")

4. Press the "🔗" **button in the preset list** — the URL that opens exactly this state is **generated and copied to the clipboard** (no hand-typing). Paste it as the slide link (your actual URL will differ by environment and preset name).

Example link:

```
http://127.0.0.1:8347/?csv=data/products.csv&view=3d&minimal=1&preset=sensor-hero
```

(If the app was opened via the server (`?csv=`), the CSV path is taken verbatim; if the file was opened by drag & drop, the path assumes the CSV sits in the data folder — a note appears when you copy.)

5. During the talk: click → the saved angle and mapping appear instantly → F11 for fullscreen

Time to display: it takes **one to a few seconds** from click to the 3D view (depends on browser warm-up, data size and label count; datasets beyond a few thousand points take a little longer). A single connecting sentence while it opens absorbs this naturally. For extra safety, open each link once right before the talk — a warmed-up browser makes the live clicks feel instant.

What a URL can and cannot reproduce:

Reproduced	How
Which CSV opens	<code>?csv=data/FILE.csv</code>
Straight to the 3D view	<code>&view=3d</code>
Minimal presentation UI	<code>&minimal=1</code>
Axis mapping, invert/normalize, marker size/opacity, color scale, font size, camera angle, slice ranges (incl. apply-to-3D)	the  -generated link (the preset itself is embedded)
Language	<code>&lang=en</code>

Not reproduced	Workaround
Trim (✂)	Apply by hand after opening (it is an on-the-spot exploration tool). To pin a range into the link, use Slice instead — slices are included

Links are self-contained: the -generated URL embeds **the preset itself** (axis mapping, camera angle, color scale, marker size), so it reproduces the same shot **on any PC or browser**. The only requirements are that the CSV sits at the same `data/` path on that machine and the server is running. ("Export/Import JSON" is only for moving your preset collection to another PC to keep editing.)

Calling up several datasets — one server, one link per slide:

Slide 5 "Products" → `http://127.0.0.1:8347/?csv=data/products.csv&view=3d&minimal=1&preset=sensor-hero`
Slide 9 "Terrain" → `http://127.0.0.1:8347/?csv=data/terrain.csv&view=3d&minimal=1&preset=fuji-top`
Slide 14 "Products, top view" → `...?csv=data/products.csv&view=3d&minimal=1&preset=sensor-top`

Security: the server listens on 127.0.0.1 only, so it is unreachable from the venue network or other PCs. No data ever leaves the machine.

Method 3: High-resolution PNG plus video

For handouts and printing, paste a high-resolution PNG (the "Export PNG" button) as a still, followed by a Method-1 video slide as the animated supplement.

Pre-talk checklist

- ☐ Chrome or Edge is installed on the presentation PC
- ☐ (When presenting on a different PC) the CubePlot folder (HTML + serve.py/serve.ps1 + start_server.bat + data) is copied onto it — **no preset migration needed** ( links embed their content). Keep the default port 8347
- ☐ (Method 2-b) `start_server.bat` (Mac: `.command`) is running → `http://127.0.0.1:8347/health` returns `ok`
- ☐ (Method 2-b) Clicking the slide link actually shows the 3D view
- ☐ (Method 1) Autoplay and loop settings verified
- ☐ Tooltip text is readable at the projector's resolution (adjust with Settings → marker size)

FAQ

Question	Answer
Does this work on a PC without Python?	Yes — nothing to install. <code>start_server.bat</code> uses <code>serve.py</code> when Python is available, and otherwise falls back automatically to a PowerShell server (<code>serve.ps1</code>) that ships with Windows . Both behave identically (127.0.0.1 only, nothing leaves the machine)
Our locked-down PCs block .bat files	Start the server directly (run both from Command Prompt). With Python: <code>py -3 scripts\serve.py --open</code> / without: <code>powershell -NoProfile -ExecutionPolicy Bypass -File scripts\serve.ps1 -Open</code> (this command invokes PowerShell itself, so Command Prompt alone is enough for either case). If PowerShell execution is also restricted, use Method 1 (video) or Method 2-a (single file)
The server says the port (8347) is already in use	Another program is using that port — switch to a different number. Config file (permanent): open <code>scripts/config.json</code> in a text editor, change <code>{ "port": 8347 }</code> to an unused number (e.g. <code>{ "port": 9000 }</code>), save, and restart the server. One-off: start with an explicit port, e.g. <code>py -3 serve.py --port 9000</code> . Either way, update the URL in your slide link to the same number (any number in 1024–65535 works)
<code>start_server.command</code> won't launch by double-click on Mac	The first time, right-click → "Open" (Gatekeeper). If it still fails (copying via a Windows machine or USB stick can strip the executable permission), open Terminal in the folder and run <code>python3 serve.py --open --dir .</code> (or restore the permission with <code>chmod +x start_server.command</code>)
Clicking the link does nothing during the slide show	Links are active only while the slide show is running. In edit mode, Ctrl+click. First check whether the server is actually running (open <code>http://127.0.0.1:8347/health</code> in a browser — it should return <code>ok</code>). If it still fails, bookmark the URL in the browser and Alt+Tab between windows — that always works

Appendix — Verifying Your Download

You can confirm that your copy of CubePlot is identical to what Soul Resonant Works published (i.e. not tampered with or corrupted) by checking its hash value (SHA-256).

How to check

Windows: run the following in PowerShell (adjust the path to where you placed the file):

```
Get-FileHash .\CubePlot.html -Algorithm SHA256
```

Mac: run the following in Terminal:

```
shasum -a 256 CubePlot.html
```

If the value shown matches the official hash below, your file is genuine.

Edition	Official hash (SHA-256)
Corporate License / Student edition (no watermark)	a3e0e57b81e6bb02b989107629ac1483bbe8e4c760fc9e31526cdfd83cc465c8
Free edition (CubePlot Free)	f7848ff3f7430aac1c8cf3ffed3d6f552d68ebb56b71f396be032d6db56e4b76

* The hash is computed from the file's contents, so it matches even if you renamed the file (e.g. CubePlot.html and cubeplot_license.html give the same value).

* This value is for the version 1.0.0 build covered by this manual. It changes with every update — check the product page or the technical document for the current value.



CubePlot Version 1.0.0 — © Soul Resonant Works