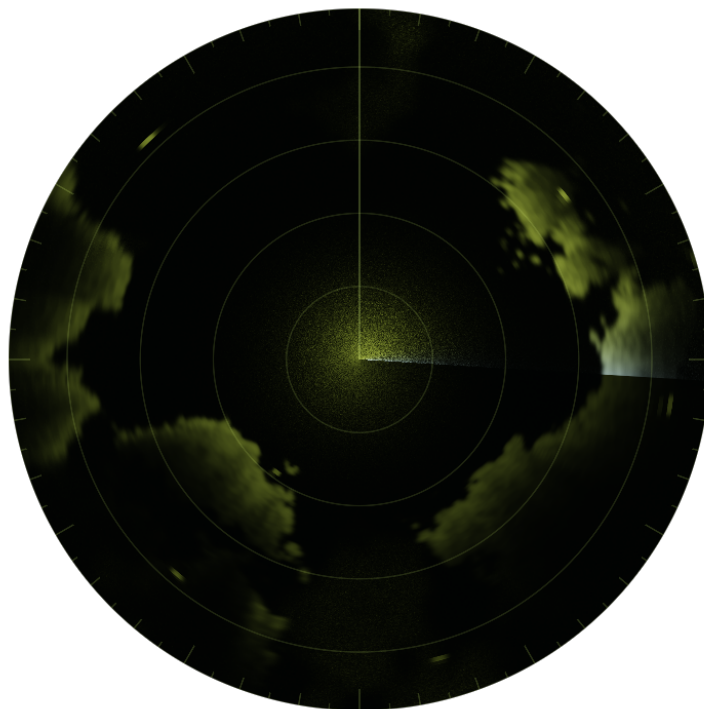


Radar user guide

Radar is a **plan-position radar scope**, for **Resolume Arena and Avenue**, as two FFGL plugins in one download: **SW Radar**, a source that is a radar watching a synthetic sea, and **SW Radar Over**, an effect that makes your clip the sea. It does not draw a picture of a radar. A reflectivity field goes through a model of a radar, and the look falls out of the model: ships painted as arcs as wide as the beam and streaks as long as the pulse, a trail behind the sweep, a bright centre that STC pushes down, clutter, rain, coasts with shadows behind them, and moving contacts that leave the plot of their track.



SW Radar at its defaults after ten seconds, rendered by the offline harness rather than captured from Resolume: a harbour approach on a 24 km scope, a bay open to the north, the sweep just past east with the trail fading behind it.

Before you rely on this: released at **v0.1.0**, and honestly early. The scope is measured rather than asserted, by a harness that drives the real plugin classes headlessly, at two rasters and on a software renderer: a point target's arc has a half-power width of Beamwidth to 0.004 degrees; its streak is $c \tau / 2$ long to 8×10^{-5} of a range bin, running outward; every bearing is painted exactly once a rotation at whole, fractional and jittered frame rates; equal targets fall with range with a slope of -3.9998 against the radar equation's -4 , and are flat with STC at $n = 4$; the phosphor holds its two-term decay to 2×10^{-7} . Eleven deliberately wrong models are each shown to fail their check, and one-character mutations of the shipped shaders are caught. All 43 parameters over both plugins are shown to change the picture. **The checks verify the stated model, not a real radar:** the phosphor's time constants are stretched to video rates on purpose, and the synthetic sea is made to look right, not measured (see Known limits). It has **never been loaded into Resolume on macOS** — the one host it has run in there is the fleet's own test host, `oxbow`, for 120 frames each. On Windows both plugins have been loaded: a build of this source (the DLLs `release.yml` built from the registered tree) loads, registers and renders in Resolume Arena 7.27.1 on software rendering (win-lab, Mesa llvmpipe, no GPU, no sound device), with every control matching what the plugins declare (29 and 27 host controls), Arena's log clean, in the fleet's Arena gate: 15 of 15 checks, the audio rows skipped. Because the sweep turns, no two grabs of the picture are alike, so the gate's noise floor is high and it could confirm only some controls one at a time: 10 of the source's 21 valued controls and 9 of the effect's 20, the rest inconclusive (none dead). The harness's own sweep (every one of the 43 parameters moves the picture) carries the rest. Software rendering says nothing about a GPU or about speed. Try it on a spare layer before you put it in a show.

This codebase was created with AI assistance, directed and reviewed by a human author.

Installing

Every download carries both plugins: **SW Radar** (a source) and **SW Radar Over** (an effect). Drop them into Resolume's effects folder and restart Resolume:

```
macOS    ~/Documents/Resolume Arena/Extra Effects/
Windows  %USERPROFILE%\Documents\Resolume Arena\Extra Effects\
```

Avenue uses the same layout under its own folder name. SW Radar appears among the sources and SW Radar Over in the effects browser.

The macOS download is a universal build (Apple silicon and Intel), as a `.dmg` or a `.zip`. It is Developer ID-signed and notarised by the release pipeline after publication, so the bundles simply load; if macOS refuses a download, it predates the signing — download it again. The Windows download is an x64 installer or a `.zip`. It is not code-signed, so the installer trips SmartScreen once: **More info** → **Run anyway**.

A scope shows the sea convolved with the radar

A plan-position indicator (PPI) is a cathode-ray tube whose trace runs out from the centre along the antenna's bearing while the antenna turns. The transmitter sends a pulse; everything the pulse hits sends an echo back; the echo from range r arrives $2r/c$ later and brightens the trace at that radius. A long-persistence phosphor holds each paint until the antenna comes round again. So what the scope shows is not the sea: it is the sea smeared by the radar that looked at it, and each smear is something you can see.

the radar	what comes out
the beam has a width	a point target paints an arc as wide as the beam: a far ship is a long arc, a near one a short one, and a coast is smeared round in bearing
the pulse has a length	every echo is a radial streak $c \tau / 2$ long (150 m per microsecond of pulse), running outward from the target's range
the antenna turns	the sweep leaves a trail : what was painted a moment ago is bright, what was painted a rotation ago is about to be repainted
the radar equation	near returns are huge : a point target's echo falls as the fourth power of range, so without correction the centre blooms and the edge is dark
sensitivity time control (STC)	the gain rises with range as R to the power n ; at $n = 4$ the radar equation is exactly undone and equal targets look equal everywhere
sea clutter, rain, noise	clutter crowds the centre , rain comes as speckled cells, noise as speckle everywhere, each different from pulse to pulse
a coast facing the radar	a bright edge with shadow behind : the land nearest returns hard and hides the land behind it
moving targets	the plot of a track : each paint of a moving contact is left fading where it was

Nothing is painted as a line per frame. Every frame paints exactly the bearings the antenna swept past since the last one, the swept wedge, into a fixed polar grid of 2048 bearings by 1024 ranges, so every bearing is painted once a rotation whatever the frame rate, and resizing the output touches none of it.

Start here

1. Put **SW Radar** in a clip slot and trigger it. The scope starts black and fills in over the first rotation, 2.5 s at the default 24 rpm: a bay open to the north on a 24 km scope, a few contacts, clutter round the centre.
2. Turn **Beamwidth** up. Every contact becomes a wide arc, and the coast smears round.
3. Turn **Pulse Length** up. Every contact grows a radial tail.

4. Turn **STC** all the way down to see the raw radar equation (the centre blooms), then all the way up ($n = 4$: flat).
 5. Change **Range**: the coast zooms, the contacts stay (see *Contacts live on the scope* below).
 6. For your own footage: put **SW Radar Over** on a clip. The clip's bright parts become echoes. Bring **Mix** down to see the clip under its own echoes.
-

The Antenna group

Resolume shows every slider as 0 to 1; the ranges below are what the ends of each slider mean.

RPM (0; 1 to 120 rpm, default 24). How fast the antenna turns. At 0 the antenna stops and paints nothing more; the picture fades by its own persistence. Faster reads as nervous, slower as a long-range search.

Beamwidth (0.5 to 20 degrees, default 2). The width of the arc a point target paints, measured at half power. This is the **two-way** width, the one you see on the scope. An antenna datasheet quotes the one-way width, which for the same antenna is about 1.4 times (the square root of 2) wider: set 1.4 degrees here to see a "2-degree" antenna. *This is a choice, made so that the number on the control is the width on the screen.* A wider beam blurs a coast; it does not brighten it (see Known limits).

Sidelobes (0 to 1, default 0.35). The shape of the beam: 0 is a Gaussian, 1 is a uniform aperture's pattern, whose sidelobes paint faint extra arcs either side of a strong target. Both have the same half-power width, so Beamwidth means the same at any setting.

Direction (*Clockwise, Anticlockwise*). Which way the antenna turns.

The Transmitter group

Pulse Length (0.05 to 20 μ s, default 2). The streak every echo leaves, $c \tau / 2$ long: 150 m per microsecond, so 2 μ s is 300 m. On a 24 km scope that is a short tail; at 12 μ s it is 1.8 km and every contact becomes a radial block.

Range (0.5 to 200 km, default 24). The scope's radius. The rings are spaced at a round number of km, four to seven of them.

Gain (−30 to +50 dB; default 0 in the source, +8 in the Over). The receiver's gain.

STC (0 to 4, default 2.6 in the source, 4 in the Over). The exponent n of the gain's rise with range. At 0 you see the radar equation raw: returns fall as the fourth power of range, the centre blooms and far targets vanish. At 4 the gain rises as fast as the returns fall and equal targets look equal at any range. Noise enters before the gain, so STC pushes the noise down near the centre too. The Over defaults to 4 so the far side of your clip is not four orders of magnitude dimmer than the near.

The Returns group

Clutter (0 to 1; default 0.4 in the source, 0.2 in the Over). Sea clutter: waves returning the pulse near the ship, strongest at the centre and falling off within a few km, speckled pulse to pulse. It is an area return, so it falls with range as R^{-3} rather than a point target's R^{-4} .

Noise (0 to 1, default 0.3). Receiver noise: speckle everywhere, before the STC gain.

The source only:

Contacts (0 to 16, default 7). Moving targets on straight tracks, respawned when they leave the scope. Each one leaves the plot of its track in its fading paints.

Rain (0 to 1, default 0.4). Rain cells: volume returns, speckled pulse to pulse, drifting with the wind. At the defaults the rain is faint behind the land and the gain; over open water (Land 0) with 10 dB more Gain it shows as speckled blobs.

Land (0 to 1, default 0.7). How much of the synthetic map is land. The coast facing the radar returns hard; the land behind it falls into shadow (1.5 km of land hides what is behind it).

Seed (0 to 9999, default 11). Which synthetic coast. The default is a bay open to the north. The map is in km, so Range zooms one map; the land is kept away from the ship itself so no seed puts the radar inside a continent.

The Over only:

Threshold (0 to 1, default 0.2). How bright a part of the clip must be to echo. The clip's **brightest channel** is used, not its luma, so a saturated blue ring echoes as well as a white one. The clip's alpha multiplies it.

The Audio group

Audio (Resolume's FFT input). **Audio Strobe** (0 to 1, default 0): each onset in the sound is a burst of interference along the next part of the sweep, a radial spoke, the way a jammer or another radar shows on a real scope. **Audio Contacts** (the source only, 0 to 1, default 0): each onset puts a strong echo 20 to 60 degrees ahead of the sweep, which the sweep then finds and which lives three rotations. Both are amounts; 0 is off. The first frame after a clip trigger fires nothing, even if the music is already loud.

A regular beat gives spokes that **stand still**: a beat every half second at 24 rpm is a fifth of a turn, so five spokes come back at the same five bearings every rotation. Change RPM, or play something less regular, to make them walk.

The Scope group

Persistence (0.1 to 30 s, default 1.6). The afterglow's time constant. At the default the trail visibly fades within the 2.5 s rotation; at 8 s several rotations linger at once.

Flash (0 to 4, default 1.5). The strength of the phosphor's fast flash at the sweep's leading edge, against the afterglow's 1.

Phosphor (*P7*, *P19*, *Green*). *P7* is the classic radar tube: a blue-white flash and a yellow-green afterglow. *P19* is orange all through. *Green* is a single green, like a general-purpose tube. Changing phosphor under a long Persistence mixes the old paints with the new until they are repainted.

Rings (0 to 1, default 0.35), **Bearing Marks** (0 to 1, default 0.5), **Heading Line** (on in the source, off in the Over). The range rings, the bearing scale round the edge, and the line at the ship's heading (north). Each is one pixel wide.

Scope Size (0 to 1; default 0 in the source, 1 in the Over). From the scope inscribed in the frame (0) to a circle covering the whole frame (1). At 1 the bearing scale is outside the frame.

Mix (the Over only, 0 to 1, default 1). The clip under the scope. At 1 the output is opaque whatever the clip's alpha; at 0 the clip is returned exactly, alpha and all.

SW Radar Over: your clip is the sea

The clip is laid under the scope in range and bearing, and its brightest channel above Threshold, times its alpha, is the reflectivity. The radar then does what it does to anything: bright shapes become echoes smeared by the beam and the pulse, painted by the sweep and fading behind it. The echoes are drawn in the **phosphor's** colour, not the clip's: a radar sees reflectivity, not colour. *This is a choice*. Bring Mix down to put the clip's own colour back under the echoes.

Point it at something with bright, separate shapes: Resolume's Trinity (rings), BattleWeapon Tank and the SpaceUniverse astronaut all read well. A clip that is bright everywhere becomes a solid disc; raise Threshold.

Contacts live on the scope

The coast, the rain and the clutter are in km: Range zooms them. The source's **contacts are in scope units**: they keep their size and their speed on the screen whatever the Range. A ship at 20 knots on a 24 km scope moves a pixel every few seconds, which does not read as traffic, and a real harbour does not fill with faster ships when you zoom out. *This is a choice, for the picture*.

How it works

- **The antenna** turns at $\text{RPM} \times \text{the host's real elapsed time}$ (the clock is Resolume's, in double precision: Resolume's clock runs to hundreds of millions of milliseconds, where a float

cannot resolve a frame). Each frame paints the bearing bins whose centres it crossed, and consecutive frames' intervals abut, so each bin is painted exactly once a rotation.

- **The echo** is, per range bin: the surface (the synthetic map, or the clip) averaged over the pulse's footprint and convolved with the beam, plus clutter, rain and point targets, times the radar equation's R^{-4} and the STC gain, plus noise; the video is $1 - e^{-P}$.
- **Point targets are analytic:** sigma times the beam at their bearing offset times the fraction of the range bin their pulse covers.
- **The beam** is a two-way power pattern, 1 on the axis and exactly $\frac{1}{2}$ at $\pm \text{Beamwidth}/2$.
- **The phosphor** is two components per texel, a flash and an afterglow. Each texel stores its value at its bearing's last paint, and the decay since then is computed exactly on the CPU for every bearing and applied when the scope is drawn, so it is exact at any frame count.
- **No sin, cos or atan** in the shaders: GLSL does not bound their error, and a software renderer's sine made a beam 3% too wide. The shaders use their own, built from bounded operations.

Performance

At the defaults, on an Apple M4 Max shared with other builds, the median frame is **0.34 ms at 720p, 0.39 at 1080p and 0.74 at 4K** for SW Radar, and **0.28, 0.43 and 1.17 ms** for SW Radar Over: a few percent of a 60 fps frame. The GPU paints only the wedge the antenna swept; the grid is fixed at 2048×1024 whatever the output size.

If it looks wrong

The scope is black. It fills in over the first rotation. At RPM 0 nothing is painted. In the Over, a dark clip under Threshold returns nothing: lower Threshold or raise Gain.

The Over is a solid disc. The clip is bright everywhere: raise Threshold.

The centre is a bright blob. STC is low. Raise it.

The far edge is empty. STC is low or Range is long: at 150 km only coastlines show, because land hides what is behind it and the sea returns little.

I cannot see the rain. Lower Land, raise Gain, and raise Rain.

The spokes stand still. A regular beat at a rotation that divides it (see the Audio group).

Old echoes linger after a change. The phosphor keeps what it was painted until the sweep comes round, and with a long Persistence for longer.

Neither plugin is in the browser. Check the folder under Installing, and that Resolume was restarted.

It does nothing at all. A shader that will not compile looks exactly like that, and the real message is in the log:

```
macOS    ~/Library/Logs/radar/radar.YYYY-MM-DD.log
Windows  %LOCALAPPDATA%\radar\logs\radar.YYYY-MM-DD.log
```

It records the GL vendor, renderer and version at load, and which shader failed if one did.

Known limits

- **The phosphor is stretched to video rates, and simplified.** A real P7 is a cascade screen: a silver-activated zinc sulphide layer gives the blue-white flash (peak near 440 nm) and a copper-activated zinc-cadmium sulphide layer the long yellow afterglow (near 558 nm), lasting over a minute in low light. The fleet's source for this is Patrick Jankowiak's compilation of the EIA/JEDEC phosphor tables (*Cathode Ray Tube Phosphors Of Interest To The Experimenter*, 2010), checked for this release; the JEDEC standard itself (TEP116-C) was not consulted. Here the flash lasts a few frames (35 ms for P7, 25 ms for P19, 60 ms for Green) so a 60 fps picture can show it, and the afterglow is an exponential with a time constant you set. The same tables list P7's long component as an inverse power law, not an exponential, so a real tube's trail has a longer, fainter tail than this one. P19's orange is (KF,MgF₂):Mn in the same tables. The colours are chosen to look like the tubes, not computed from spectra.
- **Distributed returns are normalised.** Physically the return from land, sea clutter and rain grows with the size of the resolution cell, so a wider beam or a longer pulse brightens them as well as blurring them. Here both only blur, so that Beamwidth and Pulse Length are not also gain controls. Point targets follow the radar equation exactly. *This is a choice, for the operator.*
- **Beamwidth is the two-way width** (see the Antenna group).
- **The synthetic sea is made to look right**, not measured: the fBm coast, the 1.5 km shadow rule, the clutter law, the rain, and the contacts' speeds.
- **The radar equation, STC and c tau / 2 are textbook:** a point target's return falls as R^{-4} , STC undoes it with a gain rising with range, and a pulse tau long occupies $c \tau / 2$ in range. They were checked against published summaries for this release (Wikipedia's article on sensitivity time control, Cambridge Pixel's note on STC); the textbook itself (Skolnik) was not re-read. Clutter's R^{-3} and rain's R^{-2} follow from the resolution cell's area and volume.
- **No sector scan:** Direction is clockwise or anticlockwise, never back and forth over an arc.
- **Ships and aircraft share one speed range**, in scope units.

- **Never loaded into Resolume on macOS.** Everything numeric was compiled, rendered and measured offline against the real plugin classes in a headless GL context, plus an `oxbow` load. No real audio has reached it in a host: the audio controls were checked with a synthetic spectrum only.
 - **Never seen on camera footage**, only on Resolume's bundled CG loops and the harness's card.
 - **Only ever run on an Apple M4 Max**, although the macOS build contains an Intel slice. On Windows, see the note at the top of this guide.
 - **No presets**, no OpenFX version.
 - **There is a browser demo** at radar-demo.stoatworks-labs.com. It is a port to a web page, not the plugin: the shaders run unedited in WebGL2, and the sweep's crossing rule, the per-bearing timing, the clock and the sea's contacts and rain are rewritten in JavaScript. It has no audio, so the audio controls are left off, and the page lists what else it does not reproduce.
-

About

The last group, **About**, carries the plugins' name, version, licence and maker, and buttons that open this user guide (stoatworks-labs.com/software/radar/guide/), the project page, the source on GitHub and the support page in your browser.

Reporting something

github.com/stoatworks-labs/radar/issues. A screenshot, which plugin, its Antenna and Transmitter settings, and the composition's resolution and frame rate are usually enough. If it did nothing, attach the log.

Radar · guide updated 2026-09-25 · stoatworks-labs.com/software/radar/

Latest version of this guide: stoatworks-labs.com/software/radar/guide/ · Source and issues:

<https://github.com/stoatworks-labs/radar>