Resolume OMT user guide

Two FFGL plugins that put Open Media Transport (OMT) into Resolume Arena and Avenue:

- **SW OMT In**, a source. Any OMT source on the network becomes a clip.
- **SW OMT Out**, an effect. Whatever it sits on (a clip, a layer, a group or the whole composition) goes out as an OMT source, and passes through unchanged.

OMT is the open, MIT-licensed video-over-IP protocol from the vMix team. vMix, OBS and anything else that speaks it can feed Resolume through SW OMT In, and receive from it through SW OMT Out, with alpha both ways.

> **Status:** released at v0.1.0, and early. Both plugins have run in **Resolume Arena 7.29 on macOS**, checked end to end over real OMT by a receiver that reads a test card's frame counter, so a dropped, repeated or out-of-order frame shows up as well as the colour and the picture's orientation. They haven't been tried in Avenue yet. **Windows isn't built yet**, and there's no tally or audio. Try them in your own rig before you trust them in a show.

Installing

The release has a **macOS disk image** and a **zip**, each holding both plugins. Copy both bundles into Resolume's plugin folder and restart Resolume:

```
Arena    ~/Documents/Resolume Arena/Extra Effects/
Avenue   ~/Documents/Resolume Avenue/Extra Effects/
```

They appear as **SW OMT In** among the sources and **SW OMT Out** among the effects. Both are universal, so they load in Resolume on Apple silicon and on Intel Macs.

The macOS builds are Developer ID-signed and notarised, so there's nothing to clear and no `xattr` step.

Each bundle is about 18 MB because it carries its own copy of the OMT runtime (see The OMT runtime). Keep both bundles from the same release side by side.

SW OMT In: receiving

Add **SW OMT In** to a clip slot like any other source, then pick what to receive.

Parameter	What it does
Source	The OMT sources Resolume can see, found on the network by DNS-SD and updated as they come and go. Picking one fills in Address.
Address	The source to receive. A name as receivers list it, `HOST (Name)`, or an `omt://host:port` address for a source discovery can't see. This is what the composition saves.
Fit	What to do when the source isn't the clip's size: **Fit** (letterboxed, the bars transparent), **Fill** (cropped), **Stretch**, or **Native 1:1**.
No Signal	What the clip shows after a second without frames: **Hold last frame**, **Transparent**, or **Black**.
Keep Alpha	Use the source's alpha channel if it sends one. Off makes the picture opaque.
Status	What it's receiving, read-only.

Picking a source

The Source menu lists up to 30 sources. Its first two entries are fixed: **None** stops receiving, and **Typed address** is what the menu shows while Address holds an `omt://` address you typed.

Address is the one that counts. A composition saves the name, so when it's reopened the clip receives the same source by name, whatever order the network lists sources in. A saved source that isn't on the network yet stays in the menu marked **(offline)**, so you can see what the clip is waiting for.

To receive something discovery can't see (another subnet, a network that drops multicast), type its address into Address: `omt://192.168.1.20:6400`, for example.

What Status says

Status	Meaning
`Pick a source`	Address is empty.
`Starting OMT...`	The OMT runtime is loading. It takes a moment the first time in a session.
`Connecting to HOST (Name)...`	Looking for the source, or waiting for its first frame.
`1920x1080 60 fps with alpha`	Receiving: the size, the rate the sender labels its frames with, and whether it sends alpha.
`No signal for 3 s`	Frames have stopped. No Signal decides what the clip shows meanwhile.
`OMT runtime not found - see the log`	The runtime inside the bundle didn't load. See If it doesn't work.

SW OMT Out: sending

Add **SW OMT Out** as an effect wherever you want to send from: on a clip, a layer, a group, or the composition to send the whole output. It passes its input through untouched, so it can sit at the end of a chain without changing what Resolume shows.

Parameter	What it does
Name	The source name receivers see after the host name, `HOST (Name)`.
Send	Off takes the source off the network.
Quality	The VMX encoder's quality: **Receiver decides**, **Low**, **Medium** or **High**.
Send Alpha	Send the alpha channel too, un-premultiplied.
Frame Rate	**Auto**, or a fixed broadcast rate from 23.98 to 60.
Status	The name receivers see, the size, the rate, and how many receivers are connected.

Names

Two SW OMT Outs with the same Name become `Name` and `Name 2`. When the first one goes, the second takes the name back, so a composition reopened in place still sends under the name receivers expect.

Frame Rate

Resolume renders at the display's refresh rate, often 100 or 120 Hz, and a 60 fps show should carry 60 frames a second rather than relabel 100.

- **Auto** sends every frame Resolume renders, labelled with Resolume's measured render rate snapped to the nearest broadcast rate.
- **A fixed rate** paces sending to that rate: frames go out on an even timeline at exactly that rate, whatever Resolume renders at. Pick the rate the receiving end wants.

What it costs

SW OMT Out reads each frame back from the GPU asynchronously, so it costs Resolume's render thread a copy rather than a stall: about 0.75 ms at 1080p and 1.4 ms at 4K on an M-series Mac. It adds one frame of latency. The VMX encode (3 to 4 ms a frame at 1080p) runs on a thread of its own.

Sending Advanced Output

Resolume's Advanced Output device list is compiled into the app, so no plugin can add OMT to it. Two ways round that:

- **Put SW OMT Out on the composition.** That sends the whole composition.
- **Give a screen a virtual display as its device.** Frame Ferret's virtual display shows up in Arena as an ordinary display ("Display 2" and so on), and Frame Ferret can route it to an OMT

sender. That display is experimental and macOS only, and it's in Frame Ferret from v0.3.0.

The OMT runtime

OMT's runtime is two libraries, **libomt** and **libvmx** (the VMX codec), both MIT-licensed. libomt is a Native AOT .NET library, about 18 MB. Each bundle carries a copy in `Contents/Frameworks` and loads it the first time the plugin renders a frame, never at Resolume's startup scan. So having the plugins installed costs nothing until you use one. Two plugins in one Resolume share whichever copy loaded first.

For testing another build, `RESOLUME_OMT_LIBRARY` set to a `libomt.dylib` path makes the plugins load that instead.

If it doesn't work

The Source menu is empty, or a source never connects.

- Check the sender and the Mac running Resolume are on the same network and subnet. Discovery is DNS-SD, which doesn't cross routers by default. Typing the source's `omt://host:port` into Address doesn't need discovery at all.
- On recent macOS, check **System Settings → Privacy & Security → Local Network** allows Resolume.
- If the macOS firewall is on, allow Resolume to accept incoming connections, or receivers can't reach SW OMT Out.

Status says `OMT runtime not found - see the log`. The bundle's copy of libomt didn't load. Make sure both bundles came from the same release and are complete (each about 18 MB), then read the log.

Something else. Each plugin writes a plain text log:

```
macOS    ~/Library/Logs/resolume-omt/resolume-omt.YYYY-MM-DD.log
```

To add libomt's own log, which is the only window into a send that silently fails to encode, start Arena from Terminal with `RESOLUME_OMT_VERBOSE=1` set:

```
RESOLUME_OMT_VERBOSE=1 "/Applications/Resolume Arena/Arena.app/Contents/MacOS/Arena"
```

Known limits

- macOS only. Windows isn't built yet.
- No tally, and no audio in either direction.
- Not yet tried in Avenue, or in Arena versions other than 7.29.
- Resolume's Advanced Output can't list OMT (see Sending Advanced Output).

About

Both plugins have an **About** group at the end of their parameters: the version, the licence, and buttons for this guide, the project page, the source and supporting the work.

Resolume OMT is MIT-licensed. libomt and libvmx are MIT, © Open Media Transport Contributors, and the FFGL SDK is BSD-3-Clause. Everything it's built on is credited in ATTRIBUTIONS.md.

Links

- Project page: https://stoatworks-labs.com/software/resolume-omt/
- Source and releases: https://github.com/stoatworks-labs/resolume-omt
- Open Media Transport: https://github.com/openmediatransport

Reporting something

Open an issue at https://github.com/stoatworks-labs/resolume-omt/issues, and attach the log from the day it happened.
