



AAF Everything — Quick Guide

AAF Everything converts timelines between the four exchange formats — AAF, Premiere XML, FCPXML and FCPXMLD — in every direction, one file or a whole batch at once. It is not an editor and not a player: it reads a timeline, builds a neutral model, and writes it fresh in the target format. This guide is the short tour: the workflow, the two channel switches that matter, and what each target application needs.

For AAF Everything v1.3.4 · [aafeverything.app](#)

The five steps

Before you export Final Cut sources: for deeply nested timelines, the converter cannot reliably work out a clip's timeline position. Select the affected clips — or the whole timeline — in Final Cut and choose **Clip** → **Break Apart Clip Items** before exporting; the report warns when a timeline needed it.

1. **Load timelines** — click the files row or drop them in. Several at once are fine; the list scrolls.
2. **Pick target format and frame rate** — the row reads as a sentence: *Convert to [.AAF] fps [as source]*. `as source` keeps the source's rate. The format your loaded files already have is locked in the menu — converting a format into itself is never the goal here.
3. **Decide about media** — `Link to original media` (default, nothing is copied) or `Collect next to output` (a self-contained package: output + report + `Media/` folder).
4. **Set the channel switches** if your case needs them — see below. For Pro Tools and Nuendo deliveries there is one thing to do.
5. **Convert** — the mini report appears in the window, the full report as a text file next to the output.

The output is named `<source name> (converted).<ext>` and never overwrites anything — on a collision it counts up. Settings are not saved between starts; every launch begins at the defaults.

Channels and stereo

Stereo is stored in three different shapes, and the editing world does not treat them equally:

Shape	On disk	Typical writer
1 — interleaved	one file, two channels	Resolve (default), Avid, Nuendo
2 — one file per channel	two mono files, but one stereo track	Resolve "Render one track per channel"
3 — two mono tracks	two mono files on two separate tracks	Nuendo/Cubase/Pro Tools splits, Premiere "exploded stereo"

The single most important rule: shape 3 — two mono *tracks* side by side — is recombined to stereo by **no** program. Not Pro Tools, not Nuendo, not Resolve. "I'll just deliver two mono tracks" is the most common mistake in timeline exchange, not a solution.

AAF Everything has one switch for each side of the problem, both **off by default**:

Combine split stereo tracks

Turn it **on when the source carries shape 3** — split pairs from Nuendo/Cubase/Pro Tools (same-named neighbour tracks) or Premiere's exploded stereo (`_L / _R` , `L / R` or `.L / .R` stems, hard-pan signature required). Verified against real Premiere and Nuendo exports.

- The **review window** opens as soon as you tick the switch: every audio track file by file, detected pairs as checked groups with L/R badges, everything else marked M (mono) or ST (already stereo). Untick a pair to keep it split; the combined track's name is editable. `Pair tracks manually...` guides you in two clicks — only the track directly below is offered, and tracks that cannot match say why (`layout differs` , `sources overlap` , ...).
- Combining is structural, not creative: only neighbouring tracks with identical clip layout and different sources per clip can be joined.
- **What to know before relying on it:** the upper track always becomes the **left** channel; gain, fades, automation and color of the lower track are discarded with it; and a combined pair is rendered to a new interleaved WAV in the `<output> Media` folder — **that folder must travel with the file even in Link mode**. For deliveries, `Collect next to output` is the safe way.

Separate combined channels

Turn it **on when the target is Pro Tools or Nuendo/Cubase** — it delivers every channel as its own mono WAV (`_L / _R`) while the **tracks stay stereo**; that is shape 2, the per-channel layout both work with internally (and the only shape Pro Tools reassembles into a stereo track). Resolve and Avid read the interleaved file directly — leave it off for them.

- AAF target only; for other targets the row is dimmed.
- No silent fallback: if a channel render fails, the clip shows honestly as offline (with the reason in the report) instead of slipping back to the interleaved file.
- Nothing is inflated: 16-bit and lossy sources stay 16 bit, the sample rate is never touched, and already rendered channel files are reused on re-export.

Both switches together (a Nuendo split-stereo timeline delivered to Pro Tools) work fine — the material is first combined, then split per channel; the clip name in the session carries a `_2ch` stem. In general, every file the converter writes itself marks the clip name in the target: `_2ch` , `_C1-2` , `_audio` .

What each target needs

The four formats are open exchange standards — either end of a conversion can be any application that reads or writes them. The sections below cover the applications whose quirks we have verified; they are the common cases, not the limit of what works.

Pro Tools

- Switch `Separate combined channels` **on** — everything else that keeps Pro Tools' importer stable runs automatically (the stereo track attribute only where it belongs, a unity-gain wrap around every channel combiner, canonical WAV headers on all files the converter writes).
- Stereo is the reliable ceiling for PT deliveries: surround travels completely in the AAF, but PT is not guaranteed to build those tracks as surround.
- Have all media online before converting.

Nuendo / Cubase

- Switch `Separate combined channels` **on** — like Pro Tools, Nuendo works with one file per channel internally.
- Everything else runs automatically — the four things Nuendo insists on (declared sequence lengths, sound-typed effect groups, real WAV headers in the descriptors, per-channel tape slots) are always written.
- The safe combination is WAV source material with `Transcode MXF audio` left on (the default); AIFF/FLAC/MP3 sources fall back to a weaker descriptor form that Nuendo may read as mono.

DaVinci Resolve / Avid Media Composer

- Nothing to switch; both read interleaved stereo directly.
- Resolve ignores pan on import — from any format, by design.

Premiere Pro (XML target)

- Stereo is delivered in Premiere's own "exploded" track-pair form — that is what Premiere expects from XML.
- Premiere XML has no name field for audio tracks (they arrive as A1, A2, ...) and carries no track pan; more than two channels are delivered as stereo, with a note pointing to AAF or FCPXML.

Final Cut Pro (FCPXML / FCPXMLD)

- Crossfades are written as native transitions. Final Cut blends symmetrically around the transition centre, so an asymmetric crossfade is centred there — cut point and clip content take priority; the AAF direction carries the asymmetry exactly.
- 120 fps is no Final Cut preset and is delivered as 60 fps (audio placement stays sample-accurate either way).
- Coming **from** Final Cut, remember the Break Apart note at the top of this guide.

Any other DAW or NLE

- Start with **AAF at the defaults** (interleaved stereo, linked media) — that is the shape most importers expect; use the XML family where your application prefers it.
- The three stereo shapes at the top of this guide are the vocabulary to test with: run a short timeline through once and check how your application rebuilds stereo before committing a long delivery.
- The report tells you exactly what the delivered file carries — if something looks different in your application, the difference is on the import side and usually has a setting. We are happy to look at unfamiliar dialects: support@aafeverything.app.

What transfers

Track names, sample-accurate clip positions, fades and crossfades with their real (even off-centre) cut point, constant clip gain and volume automation, track pan (constant and moving — via AAF only), markers, all video tracks, and a timecode track. Frame rates are written exactly, NTSC rates included (24000/1001, drop-frame flag) — the fps choice affects timecode and picture-edit snapping, audio placement is sample-accurate at every rate.

The **Include panel** can exclude any of it per run: crossfades, fades, clip gain, volume automation, pan, markers, colors, video, extract video audio. Excluded items are listed in the report.

No effects travel — that is the nature of these formats. AAF, Premiere XML and FCPXML carry the cut: clips, positions, fades, levels, pan, markers. Plugins, effect settings, EQs and busses are not part of any of these formats — in no converter.

Not carried, stated honestly:

- **Fade curve shapes** — fades arrive linear / standard dissolve.
- **Clip names from the source timeline** (the converter's own renders name their clips, see above).
- **Track mute and lock.**
- **Retiming** — stretched material should be rendered in the source app; the report names every affected clip.

- **Multicam, compounds and auditions are flattened and show the correct cuts** — the delivered timeline carries the active angle or take, edited exactly as in the source; only the container structure itself is not part of these formats.
-

Media

- **Link** (default) references the original files; **Collect** builds a movable folder with output, report and `Media/`.
 - **Relink** finds moved media by itself — Windows paths, renamed folders, Unicode quirks; name-only matches are flagged in the report for checking.
 - **MXF audio** is transcoded to WAV by default (multiple streams become one interleaved file).
 - In the AAF target, media is always **linked, never embedded** — the form Pro Tools requires. An AAF without its media folder is empty on arrival; for handing off to another machine, use Collect.
-

The report

Every conversion writes `<output name> – AAF Everything report.txt` next to the output. The rule behind it: *a silent change is a defect, not a feature* — everything left out, changed, decided or repaired is listed, with counts and, where one exists, the action that remains. The mini report in the window shows the important lines; the file has all of them. If nothing had to change: "No notes — nothing had to be changed or left out."

Demo and license

- The demo converts the **first 120 seconds** of every timeline — a hard cut, no watermarks, no locked features.
 - Licensing is one path (Lemon Squeezy): activate once online, run offline from then on; `Deactivate this computer` frees the seat for another machine.
 - On start the app sends one version-check ping — version, platform, demo-or-licensed. No key, no account, no device ID; it can be switched off, and without a network the start never hangs on it.
-

Good to know

- **A question, or found a bug?** We are happy to hear from you: support@aafeverything.app.