

Editor tutorials

You assemble picture and sound, and produce the file that can be shown.

CONTENTS

1. Get the M&E out of an episode delivered as a mix · Voice / M&E separation · A few minutes per episode, unattended
2. Render the dubbed video · Subtitles and delivery · A few seconds per episode
3. Adjust the lips on screen · Voices · 5 minutes of setup, then the processing
4. Export what the client expects · Subtitles and delivery · 2 minutes
5. Contractor side: receive and deliver · Project management · 2 minutes

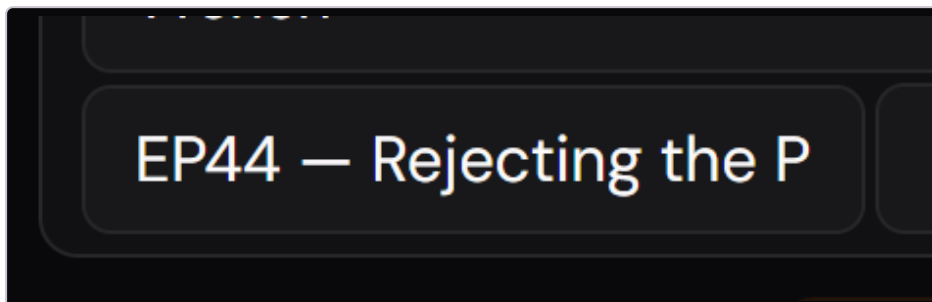
Get the M&E out of an episode delivered as a mix

The client sent a single file. You must dub over it without losing the music.

A few minutes per episode, unattended

01 Name the project first

The project name decides the episode folder. Without it, the elements land next to the sources and the final render will not find them on its own, Axeho tells you before starting, not after.



The project name field, in the top bar, filled in with the episode name.

02 Pick the sources

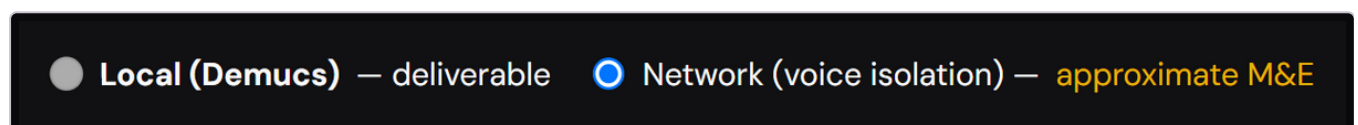
Videos or audio files, one or fifty. The sound is extracted for you: no need to prepare WAV files.



The source picker buttons, videos or audio files, and the output folder.

03 Choose the route

The **local** route returns a deliverable M&E and needs no network. The **online** route is quicker to set up, but its M&E is a subtraction: good for a mock-up, never for broadcast. The produced file name carries the difference.



The two separation routes: local, deliverable, and online, whose M&E is approximate.

04 Start it, and move on

The work continues in the background. You can switch tabs, transcribe, write the band. A final report lists the files produced and any failures.



The Separate button, which starts the job in the background.

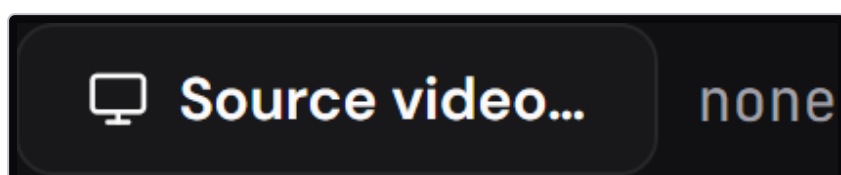
Render the dubbed video

The client wants to watch the episode, not open a DetX.

A few seconds per episode

01 Choose the picture

The **source video** is the one from the edit. It is **not re-encoded**: the video stream comes out identical to what went in, which is why an episode renders in seconds rather than half an hour.

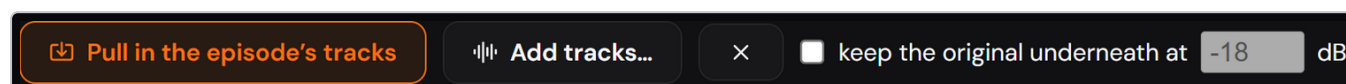


The button that picks the render's source video.

02 Pull in the episode's tracks

Pull in the episode's tracks fetches the voice from the last generation and the separated M&E, where they were filed. Everything stays editable, and you can add tracks by hand.

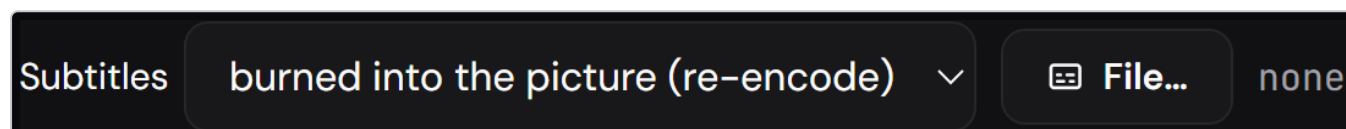
Keep the original underneath is for checking, never for delivery: you then hear both versions at once, which is exactly what a broadcaster rejects.



Pulling in the episode's tracks, adding tracks by hand, and the option to keep the original underneath.

03 Subtitles: burned in or embedded

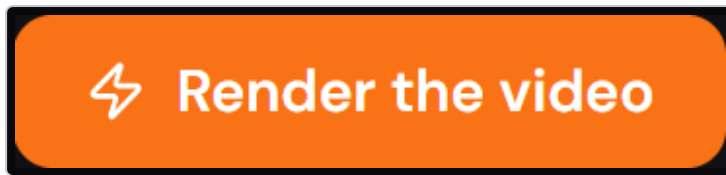
Embedded track leaves the picture untouched and can be switched off in the player. **Burned into the picture** re-encodes, so it takes longer, but the text shows up everywhere — including in editing software, which ignores text tracks.



The render's subtitle menu: none, embedded track, or burned into the picture.

04 Render

A name, a container, an output folder. The file produced is the one you show the client — and it is also the one to run through **quality control** before sending it.



The button that starts the dubbed video render.

Adjust the lips on screen

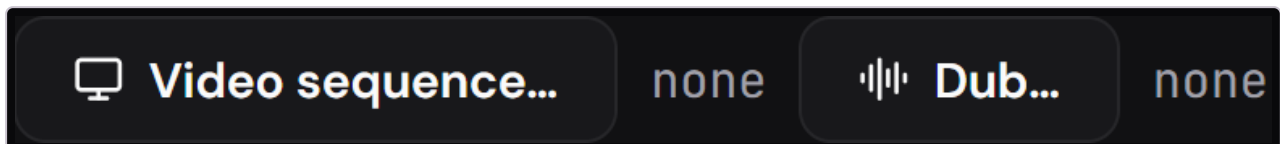
An advert, a corporate film, a training module: the picture has to match the dub.

5 minutes of setup, then the processing

01 Point at the sequence and the dub

A **video sequence** and the **dub** that goes with it. The feature works on a shot, not on a whole episode.

Not for fiction. The actor's performance is the product; distorting it damages the very thing you are paid to deliver. Axeho's reference sync remains the one in the text, words chosen to land where the mouth closes.



Picking the video sequence and the dub file.

02 Process only the part that matters

Bound the passage in seconds. Less data sent, so cheaper, and a better result: the service works on what matters instead of crossing the whole shot.

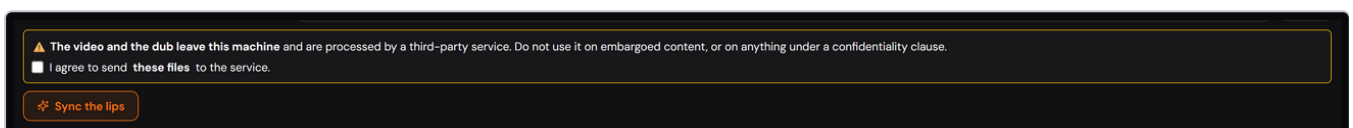


The start and end bounds of the excerpt to process, in seconds.

03 Accept the upload, then start

The video and the dub leave this machine and are processed by a third-party service. The box is ticked each time, for **those files**: it is a decision you take, not a setting you forget.

Everything else on this page works on the machine. This option is the only one that leaves, which is why it asks for explicit consent.



The third-party upload warning, its consent box and the start button.

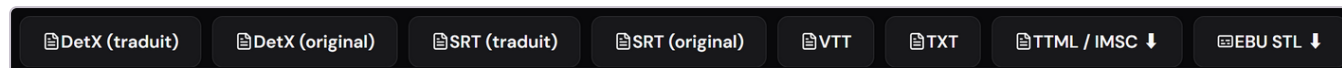
Export what the client expects

The end of the chain: every recipient expects a different format.

2 minutes

01 Pick the right format

DetX goes to the booth. **SRT** and **VTT** go to subtitling. **TXT** is for proofreading on paper, still the fastest medium for client sign-off.

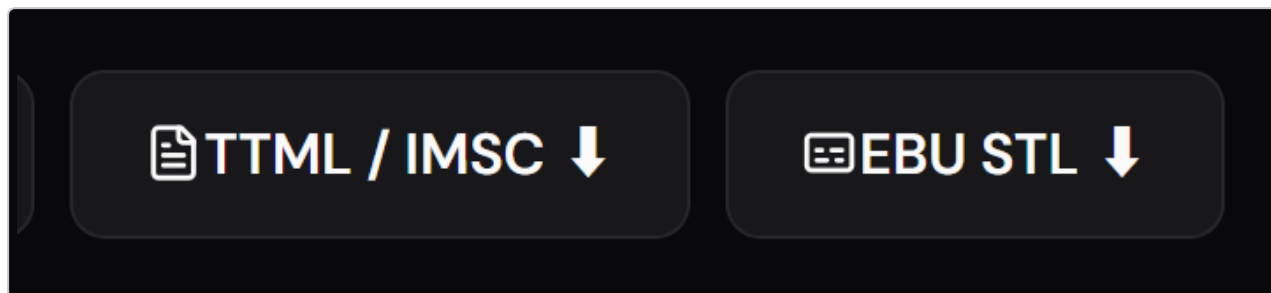


The export formats: DetX, SRT, VTT, TXT, TTML and EBU STL.

02 Export in the broadcasters' formats

EBU STL for European channels, with the French teletext conventions — an export that went through CANAL+'s own check. **TTML** for the platforms, switching to **IMSC** when the language requires it.

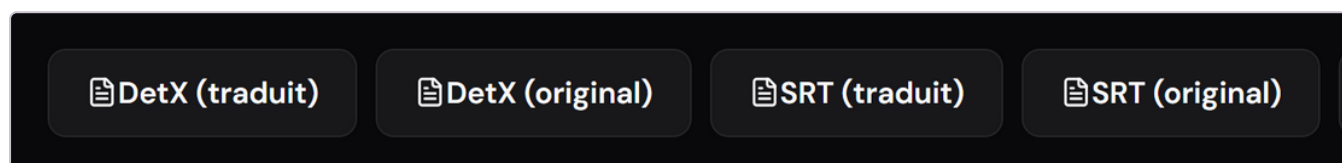
Duration and reading-rate constraints are **checked and reported**, never silently fixed: a subtitle that runs too fast shows up in the report, and you decide what gets cut.



The TTML / IMSC and EBU STL exports, next to the everyday formats.

03 Original or adapted

DetX and SRT export in **both versions**. Clients often ask for both: one to check against, one to work from.



DetX and SRT, each in an adapted and an original version.

04 Export a single role's band

For an actor in a home studio, the **role sheet** gives their lines in film order, timecoded, with the word count, useful when they are paid by the line.



The Role menu and the Role sheet button, in the rythmo band.

Contractor side: receive and deliver

The invited voice actor, mixer or adapter: what they see, and what is expected of them.

2 minutes

01 See what you have been given

A contractor has only **two tabs**: their work and their invoices. The rest of the console is none of their business, and it is removed rather than left there to refuse them.

The **working folder** holds what the studio provides — video, DetX, script, series bible — and the **instructions** say what to know before starting.

My projects 2

What the studio has given you. **Delivering is not being paid** : the amount is only settled once the studio has approved the project.

The working folder holds what the studio gives you to work with; you hand in by dropping the link to yours in turn.

PROJECT	ROLE	DUE DATE	AMOUNT	STATUS	WORKING FOLDER	DELIVERY
Voix Samara — épisode 24 FATED BRIDE — épisode 24 FB-24	Comédien voix	12/08/2026	74 900 XOF	done	open	open 11/08/2026
Instructions from the studio Bible de série jointe. Samara tutoie Finn à partir de la scène 4.						
Link to the delivered folder (Drive, WeTransfer...)						
https://drive.exemple.ci/fb24-samara		<button>Save the link</button>				
Voix Samara — épisode 23 FATED BRIDE — épisode 23 FB-23	Comédien voix	16/08/2026	to be approved	in progress	open	nothing delivered
Instructions from the studio Épisode plus court : 168 lignes.						
Link to the delivered folder (Drive, WeTransfer...)						
https://drive.google.com/...		<button>Save the link</button> <button>I've finished</button>				

The contractor's view: two assignments, their deadline, amount and the studio's instructions.

02 Submit the delivery

You deliver by pasting the **link to the delivered folder**, then marking **I've finished**. **Delivering is not being paid**: the amount is only fixed once the studio approves the work.

Link to the delivered folder (Drive, WeTransfer...)

Save the link

The delivery link field and its save button, on the contractor's side.

This tutorial online : <https://axeho.studio/en/tutorials-editor.html>