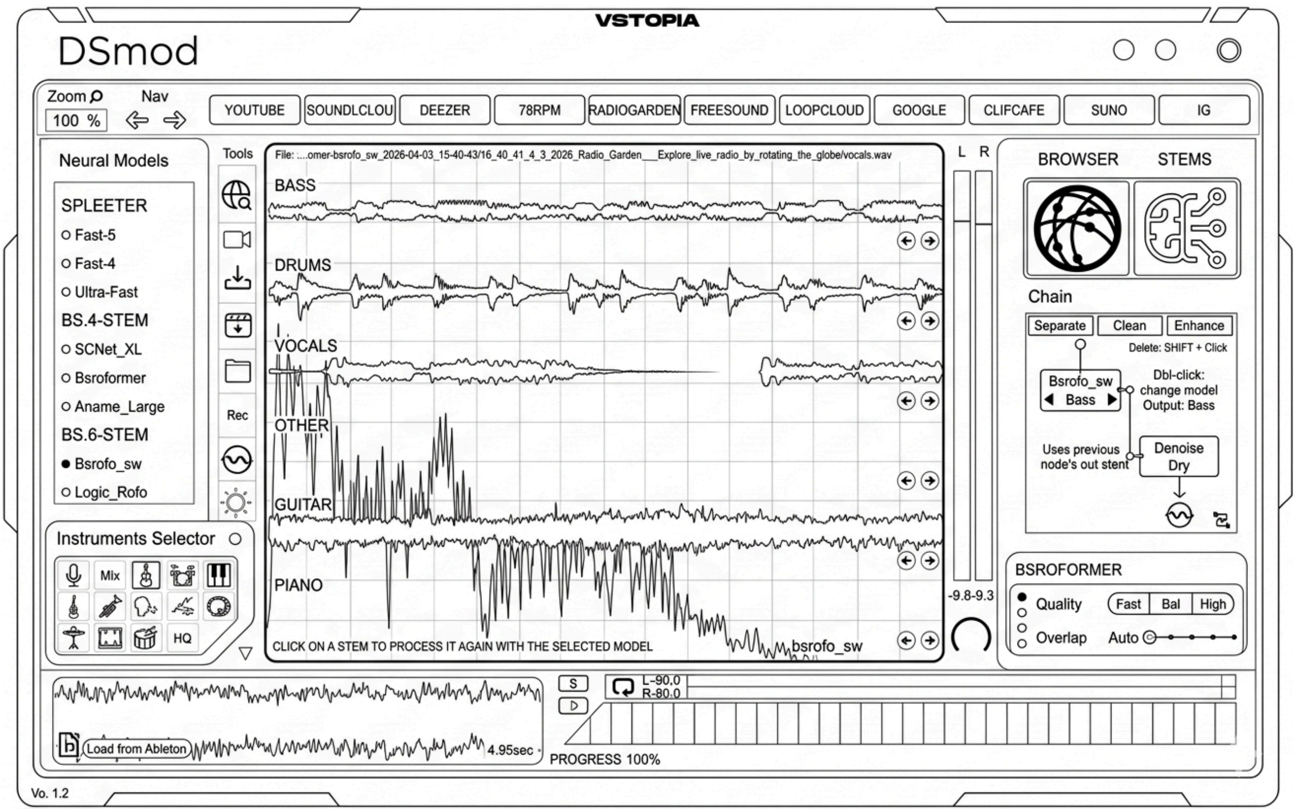
Table of Contents


VSTOPIA
DSmod
Zoom
Nav
100 %
YOUTUBE
SOUNDLCLOU
DEEZER
78RPM
RADIOGARDEN
FREESOUND
LOOPCLOUD
GOOGLE
CLIFCAFE
SUNO
IG
Neural Models
SPLEETER
Fast-5
Fast-4
Ultra-Fast
BS.4-STEM
SCNet_XL
Bsroformer
Aname_Large
BS.6-STEM
Bsrofo_sw
Logic_Rofo
Tools
Rec
BASS
DRUMS
VOCALS
OTHER
GUITAR
PIANO
CLICK ON A STEM TO PROCESS IT AGAIN WITH THE SELECTED MODEL
bsrofo_sw
L R
-9.8-9.3
BROWSER
STEMS
Chain
Separate
Clean
Enhance
Delete: SHIFT + Click
Bsrofo_sw
Bass
Dbl-click: change model
Output: Bass
Uses previous node's out stem
Denoise
Dry
Instruments Selector
Mix
HQ
BSROFORMER
Quality
Fast
Bal
High
Overlap
Auto
Load from Ableton
4.95sec
S
L-90.0
R-80.0
PROGRESS 100%
Vo. 1.2

DSM: Dynamic Split Module

A Max for Live laboratory for advanced sample extraction, separation, and neural restoration. Capture audio from anywhere, isolate stems with 63 AI models, and reshape sound directly inside Ableton Live.

DSM unifies multiple state-of-the-art separation and restoration architectures, including Demucs, RoFormer-based models, Apollo restoration, and AudioSep text-prompt separation, under a single production workflow. Instead of switching between standalone tools, everything runs locally through one controlled environment.

Designed as a modular neural processing platform, DSM can integrate new models and architectures over time, extending beyond static functionality.

What we solve

Instead of recreating sounds from scratch, you can **get the exact sound you want from the music you like**. DSM lets you capture, isolate, and pull those sounds directly into your project without leaving Live.

In one place, DSM addresses:

- **Stem isolation.** Separate vocals, drums, bass, and instruments with 63 neural models, and compare results side by side before you commit.
- **Web audio and video capture.** Capture from YouTube, SoundCloud, Deezer, Suno, Loopcloud, and other sources; get video from the internet when you need it.
- **Dialogue and film clips.** Use ClipCafe to find and grab the exact moment you want from movies and series.
- **Neural processing inside Live.** Run separation, restoration, and denoise inside Ableton; no bouncing to external apps.
- **Direct DAW integration.** Drag stems into your set (single or multiple at once); the workflow stays in the device and in Live.

So you spend less time redesigning the same sound and more time using the one you heard.

Sound design and finding the right sounds are highly time-consuming and can drain inspiration when you need to stay in the flow. Relying on **online memberships** for good-quality stems or samples is effective but **expensive and not always sustainable**. DSM is a **standalone toolbox**: no subscription, no round-trips to the cloud, capture, separate, and restore inside your DAW, ready to use whenever you need it. All processing runs locally on your system.

DSM is a spinoff of **SplitWizard+** and **YouTube4Live** by **Ostin Solo** (ostinsolo.co.uk).

What DSM does

- **Download.** Audio and video from YouTube, SoundCloud, Deezer, Suno, Loopcloud, and other supported sources. Choose audio-only or video at various resolutions.
- **Audio support.** 44.1 kHz and 48 kHz sample rates; WAV, MP3, and FLAC files for loading and processing.

- 63 stem and processing models across multiple research architectures. Vocals, drums, bass, instruments, karaoke, denoise, dereverb, de-bleed, de-breath, restoration, and text-prompt separation. Each model produces specific stem types; see Models Overview and the model-family pages for what each one outputs and which process it uses.
- MP3 to FLAC. Convert lossy material to FLAC for consistent archiving and compatibility within your workflow.
- Neural restoration. Use neural restoration models to reconstruct high-frequency content in compressed or degraded sources.

All operations are non-destructive: original files remain untouched, allowing safe experimentation and comparison.

Workflow: grab a track from the web, optionally convert to FLAC, run separation or restoration, then use the stems and loops in your set.

What makes DSM different

- Multiple independent model families in one device, not a single algorithm with variations.
- Side-by-side comparison before committing to a result.
- Integrated web capture and processing, no switching apps.
- No cloud dependency, separation and restoration run locally.
- Designed for production workflows, not just export.

Methods to use sound separation

Stem separation is a core tool for modern production, post-production, and content workflows. DSM delivers professional-grade isolation and cleanup inside your DAW. Use it for:

- Demix and stem extraction: Split full mixes into vocals, drums, bass, and other instruments for analysis, reuse, or recomposition.
- Remix and mashup: Isolate elements from existing tracks to build new arrangements; combine stems from different sources with full control.
- Sample-pack creation and editing: Pull clean one-shots and loops from any source; edit and export stems for your own packs or client deliverables.
- Karaoke and a cappella: Get instrumentals or vocal-only stems for performance, practice, or content; create backing tracks or extract acapellas for doubles and layers.
- Field recording and dialogue: Clean up location audio: reduce bleed, reverb, and room tone with denoise and de-bleed; improve dialogue clarity with de-breath and restoration.
- Registration and restoration: Restore lossy or degraded material (FLAC conversion, neural reconstruction); clean and separate before archiving or re-releasing.
- Content creation and video: Extract music, dialogue, or SFX from videos; get instrumentals for voiceovers, sync licensing, or social content.
- Music production and sound design: Speed up sound design by pulling single elements from references; compare 63 models and pick the best result before committing.
- Education and transcription: Study arrangements, train your ear, or prepare material for transcription by isolating parts clearly.

Every workflow runs inside Ableton Live: download or load, separate, then drag stems into your set. No round-trips to external apps, one device for capture, separation, and delivery.

When NOT to Use DSM

Separation is estimation, not source recovery. Knowing its limits is part of using it well.

- **Heavily mastered commercial pop.** Hard limiting, saturation, and stereo enhancement cause spectral blending that separation models cannot fully undo. Results will contain artifacts. They can still be useful—but set expectations accordingly.
- **Very short clips.** Clips shorter than the model's chunk size give the network insufficient audio context. Results may be unstable or smeared. Keep clips above the minimum chunk duration for your chosen model.
- **Complex EDM layers.** Convolution-based and lightweight models struggle with densely layered electronic material. Use transformer-based models (BS-RoFormer) for better results on complex spectral content.
- **Pristine broadcast or archival quality is required.** Neural separation introduces estimation artifacts. For legal or archival delivery where source integrity is critical, separation output is not a substitute for original recordings.

Defining limits strengthens creative control.

System requirements & processing

Separation and restoration models vary in computational demand. GPU acceleration is supported where compatible hardware is available; otherwise models run on CPU. Performance depends on model architecture, audio length, and system configuration.

Workflow efficiency

Faster processing and fewer round-trips mean:

- Quicker remix turnaround
- Faster delivery of stems to clients
- Streamlined restoration jobs
- Reduced friction in sample-based production

DSM is built for producers who prefer extracting and transforming real-world sound rather than recreating it from scratch, turning references into raw material.

DSM Philosophy

DSM is not a toy separation plugin. It is a neural sampling laboratory.

It unifies 63 architectures—spanning convolutional, transformer, restoration, and text-conditioned models—in a single controlled production environment. You do not switch tools. You switch models.

It is designed for extraction, experimentation, and transformation: pulling real-world sound apart, evaluating results, and bringing what you need directly into your set.

It replaces a scattered collection of standalone tools, research demos, and export-and-reimport workflows with one modular system that runs inside Ableton Live. No cloud. No subscription. No round-trips.

The goal is not to recreate sounds from scratch. The goal is to get the exact sound that already exists—faster, inside your DAW, with full control.

Quick links

- Download manual as PDF.
- Overview: Introduction and positioning.

Installation

The full DSM model library exceeds 31 GB. You do not need to install the entire library at once. DSM installs models, tool runtimes, and supporting components when you need them—after you choose where files should live on disk.

Installing models and shared components

Installation is driven from the DSM device in Ableton Live. There is no separate installer window: you work in the same interface you use to separate and process audio.

Destination folder. When DSM needs a location for models, outputs, and shared tools, it will prompt you to set a base folder (and output folder where applicable). Choose a drive with enough free space; you can change this later using the folder button in the device's settings.

If installation or downloads fail (file access)

DSM runs **inside Ableton Live**. Downloads and installs write through Live to the folders you chose. If a model or runtime fails to install, or files never appear on disk, your operating system may be blocking Live from writing outside its default locations.

> What to check
>
> - **macOS:** Open **System Settings → Privacy & Security**. Under **Files and Folders** (and **Full Disk Access** if your chosen path is still blocked), ensure **Ableton Live** is allowed to access the drive or folder where you want models and tools stored. If in doubt, pick a folder under your user home (for example **Documents**) and grant access when macOS prompts you.
> - **Windows:** If you use **Controlled Folder Access** or strict antivirus software, allow **Ableton Live** to write to your chosen base folder, or choose a folder that is not protected (for example a dedicated folder under your user profile). Retry the install after changing the path or the allow list.
>
> After adjusting permissions or choosing a different folder, try the download or install again from the model list.

Model browser. The model list groups models by tool or family (for example Demucs, Spleeter, BS-RoFormer). Each row shows the model name and a short description. Colour tells you the state at a glance:

- Red — not installed. The model is missing from disk. It must be downloaded before you can use it (or DSM may fetch it automatically when you process—see below).
- Green — installed. The model is on disk and ready. A single click selects it; the row turns **light blue** while selected so you can see your current choice. When you move the selection elsewhere, the row returns to **green**.
- Orange — waiting or downloading. Orange is used while a model is **queued** (another install is in progress first) and while it is **actively downloading**. During download the row text shows progress (for example a percentage) in the same list—no separate installer window.

How to install a missing model. Double-click a model that is not yet installed to start its download. Single-click only selects; it does not start a download.

Processing with a missing model. If you start separation or processing while a required model is not installed, DSM can download and install it automatically, show progress in the model list, and then continue your request when the install completes—using the same install path as when you install from the browser.

Shared components. Supporting pieces (such as shared runtimes or helper tools) are installed when DSM needs them for the task at hand, so you are not asked to manage every dependency by hand.

Shared Runtime Behavior

The DSM shared runtime, required for Demucs, BS-RoFormer, and VR/Apollo, installs automatically when the first model from those families needs it.

It is installed once and shared across supported architectures.

Some tools (for example Spleeter or AudioSep) use their own runtimes; DSM installs those when you install a model or workflow that requires them.

Download Time and Disk Usage

Ensure sufficient disk space before installation. Model sizes vary and may increase over time as updates are released. Some models may require additional temporary space during installation.

> ! **Download time.** Installing the complete model library (31+ GB) may take 40 minutes or more depending on connection speed and system performance. Select only the architectures or models you need to reduce download time and disk use.

Third-party Models and Licenses

All neural models and executables distributed through DSM are created and maintained by their respective authors and are subject to their individual licenses. DSM and VSTOPIA do not claim ownership of these models or their trained weights.

By downloading or using these models, you agree to comply with the terms specified by their authors. Compliance with applicable license terms is the responsibility of the end user.

Official repositories and license information:

- **Demucs**, https://github.com/facebookresearch/demucs (MIT License)
- **Spleeter**, https://github.com/deezer/spleeter (MIT License)
- **BS-RoFormer**, https://github.com/lucidrains/BS-RoFormer
- **Apollo**, https://github.com/JusperLee/Apollo
- **Audio Separator / VR**, https://github.com/nomadkaraoke/python-audio-separator (MIT License)

For a full list of model contributors and technologies, see Credits. For detailed license texts, see License Information, the project's **LICENSES** folder, and **LICENSES.md** where applicable. For disclaimers on recording length and legality and on using music or dialogue from films (e.g. ClipCafe), see Recording disclaimer and ClipCafe and film/actor content.

Model availability and license terms may change. Always consult the original repositories for the most up-to-date information.

User Responsibility and Copyright Notice

DSM provides tools for audio processing. Users are responsible for ensuring that their use of downloaded or processed content complies with applicable copyright and local laws.

← Introduction
Interface →
Credits →

Compatibility

DSM runs as a Max for Live device. Supported operating systems and host versions are summarised below.

Audio files

DSM supports **44.1 kHz** and **48 kHz** sample rates, and **WAV**, **MP3**, and **FLAC** files for loading, separation, and restoration.

Operating systems

Platform	Supported	Notes
macOS (Apple Silicon / ARM)	✓	M1, M2, M3, M4, etc. GPU acceleration via MPS when available.
macOS (Intel)	✓	Intel Macs supported; processing on CPU.
Windows	✓	Windows 10 / 11. CUDA used for GPU when available.

Ableton Live

Live version	Supported
Live 11	✓
Live 12	✓

DSM is a Max for Live device: file downloads and installs are performed by **Ableton Live** into the folders you configure in the device. If installs fail or files do not appear on disk, your OS may be blocking Live from writing to the chosen location. See Installation: If installation or downloads fail (file access) for macOS and Windows checks.

GPU acceleration: faster on ARM and CUDA

Neural separation and restoration run **faster when a GPU is used**. On supported hardware, DSM uses the GPU automatically; otherwise it falls back to CPU.

macOS (Apple Silicon)

On Apple Silicon Macs (M1, M2, M3, M4, etc.), DSM uses **Metal Performance Shaders (MPS)**: Apple's GPU framework. MPS is used for neural inference when available, so stem separation and restoration are significantly faster than CPU-only. You don't need to install anything extra; the device uses MPS automatically.

Windows

On Windows, DSM uses **NVIDIA CUDA** when an NVIDIA GPU and compatible drivers are present. We target **CUDA 11.x and 12.x** (e.g. CUDA 11.8 or 12.1 depending on the runtime). With a supported GPU, processing is much faster than on CPU. If no CUDA-capable GPU is found, the device runs on CPU.

CPU fallback

On Intel Macs or when no GPU is available, all processing runs on CPU. It still works; expect longer run times for large files or heavy models. See the model-family pages for options like thread count, TTA, and chunk size that can help tune performance.

Suggested file length on CPU: To avoid long waits, we suggest using **files no longer than about 30 seconds** when running on CPU. The longer the file, the longer you wait—and that increase is **more than proportional** (often called "exponential" in everyday language): doubling the file length can more than double the processing time. That's because the neural models process audio in fixed-length chunks; more audio means more chunks, and on CPU each chunk takes much longer than on a GPU. So a one-minute file can take several times longer than a 30-second one, and a full song can become impractical on CPU.

What this means for the models: Heavier models (e.g. DEMUCS, BS-RoFormer, APOLLO) are especially slow on CPU, so the 30-second guideline matters most for them. Lighter models (e.g. SPLEETER) are more forgiving but will still get noticeably slower as file length grows. If you regularly work with longer material on a machine without a usable GPU, consider cropping or trimming in DSM to a short segment (e.g. under 30 seconds) before running separation or restoration.

Performance Expectations & Hardware Guidelines

Processing speed depends on hardware, model choice, file length, and preset. The figures below are approximations for a 20-second region; actual times vary by model and system load.

> ! Separation speed scales non-linearly with file length. Doubling duration may more than double processing time.

Hardware	Recommended use	Avoid	Typical 20 s region
CPU only	Spleeter, light Demucs models; short clips ≤ 30 s	Full songs, High preset, TTA	30–120 s
4 GB VRAM GPU	4-stem Balanced preset; moderate chunk sizes	6-stem High + TTA	5–20 s
8 GB VRAM GPU	4–6 stem Balanced; BS-RoFormer at default chunk	Long files with High + TTA	3–12 s
Apple Silicon (M1–M4)	Most models at moderate chunk size; Demucs and BS-RoFormer run efficiently	Huge full-track 6-stem High with TTA	4–15 s
NVIDIA 12 GB+ GPU	High preset + TTA for export-quality stems; all architectures	Rarely limited	1–6 s

Per-hardware guidance

- **CPU only.** Do not run full songs. Crop to 10–30 seconds. Use Spleeter or lighter Demucs variants. Avoid High preset and TTA entirely.
- **4 GB VRAM.** 4-stem models at Balanced preset are reliable. 6-stem models with High preset or TTA will push VRAM limits and may fall back to CPU or fail.
- **8 GB VRAM.** Balanced preset is reliable for 4–6 stem separation. Use High preset only for short regions; avoid combining High + TTA on long files.
- **Apple Silicon (M1–M4).** BS-RoFormer and Demucs perform efficiently via MPS with moderate chunk sizes. Full-track separation at High is practical for shorter material.
- **NVIDIA 12 GB+ GPU.** High + TTA becomes practical for export-quality work. Most models and presets run without VRAM constraints on typical material.

← Introduction · Controls & Workflow · Downloading

Downloading Audio & Video

DSM can download or record from various sources. You don't need to paste anything: play the content you want, then use the download or record button and the device will capture the currently playing audio or video.

How it works

Download (YouTube and ClipCafe): start playback of the video you want, then click the **download audio** or **download video** button on the tools strip. The device automatically downloads the currently playing item to your output folder. When the file is ready the download button is replaced by a drag icon (audio: , video:); you can then **drag the file from the button** into your project.

Once the file is ready, the button shows a drag icon and you can drag the file into Live. The following show the drag action for downloaded audio and video from the tools strip.

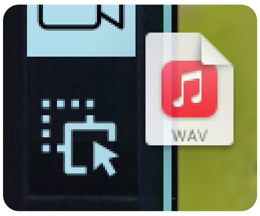


Figure: Drag downloaded audio from YouTube into Live

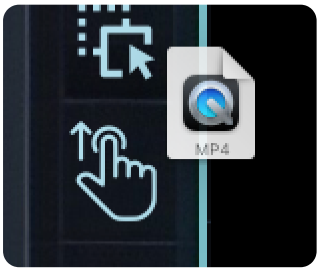


Figure: Drag downloaded video from YouTube into Live

Record streaming (all other sources): for sources that don't support direct download, DSM records the stream while it plays. Use the source and URL as usual, then use **Rec** (or the record action in the Instrument Selector) to capture the audio.

> ! **Recommended when downloading from YouTube:** Stop or pause playback before the download finishes. If audio is playing (or many other audio processes are running) when a **direct download from YouTube** completes, a short freeze can occur. To avoid this, **stop the audio when the progress bar approaches 100%**; that is, when the file is almost downloaded. Keeping playback and other DAW activity light during the download also helps.

Supported for

Source	Download (audio/video to disk)	Record streaming
YouTube	✓	✓

Source	Download (audio/video to disk)	Record streaming
ClipCafe	✓	✓
SoundCloud	—	✓
Deezer	—	✓
Suno	—	✓
78RPM	—	✓
RadioGarden	—	✓
Freesound	—	✓
Loopcloud	—	✓
Google	—	✓
Other Webpages	—	✓

Output folder

You can choose where downloaded and recorded files are saved. Click the **folder icon** 🗀 on the left tools strip to select the output folder. Files are then saved there (e.g. in subfolders such as YouTube_DSU or Clips_DSU).

Download types

- **Audio only**: Best available audio, with optional conversion to WAV or FLAC. Ideal for stem separation and production.
- **Video**: Video plus best audio, merged. For YouTube you can typically choose resolution (e.g. 360p, 480p, 720p, 1080p, or best available) in the right panel.

MP3 to FLAC

DSM offers conversion from MP3 (or other lossy formats) to FLAC. Use this when you want to archive or process material in a lossless form. Combined with restoration models, you can then improve missing top-end or overall quality.

Streaming and playback

For YouTube you can stream video for preview in the built-in player without downloading. For record-only sources (e.g. SoundCloud, Deezer, Suno), playback is the stream itself; use Rec to capture it. See Controls & Workflow for cropping, stem drag, and Compatibility for system requirements.

← Introduction · Controls & Workflow · Compatibility

Interface

DSM's layout is divided into these main areas:

- Web bar (top center), source selection
- Tools strip (left side), vertical controls
- Left panel, models and filters
- Center, webpage or stems; one shared area that switches content (controlled from the right panel)
- Sampler and waveform (bottom left), load file, crop, play, and work with stems
- Right panel, Browser/Stems switch, Instrument Selector, info panel, advanced controls
- Bottom bar, progress and overview

The core workflow revolves around three elements: the waveform/sampler, the neural models, and the source pages.

Figure: DSM interface overview

Web Bar – Source Selection (Top Center)

Use the top buttons to choose the source of audio or video:

- YouTube
- SoundCloud
- Deezer
- Suno
- 78RPM
- RadioGarden
- Freesound
- Loopcloud

- Google
- ClipCafe

Play the content you want; DSM downloads or records according to the active source. See Downloading for details.

Logging In (YouTube, SoundCloud, Deezer)

These services require login via email and password. Google redirect or popup-based authentication is not supported in this environment. Use direct email/password login when available.

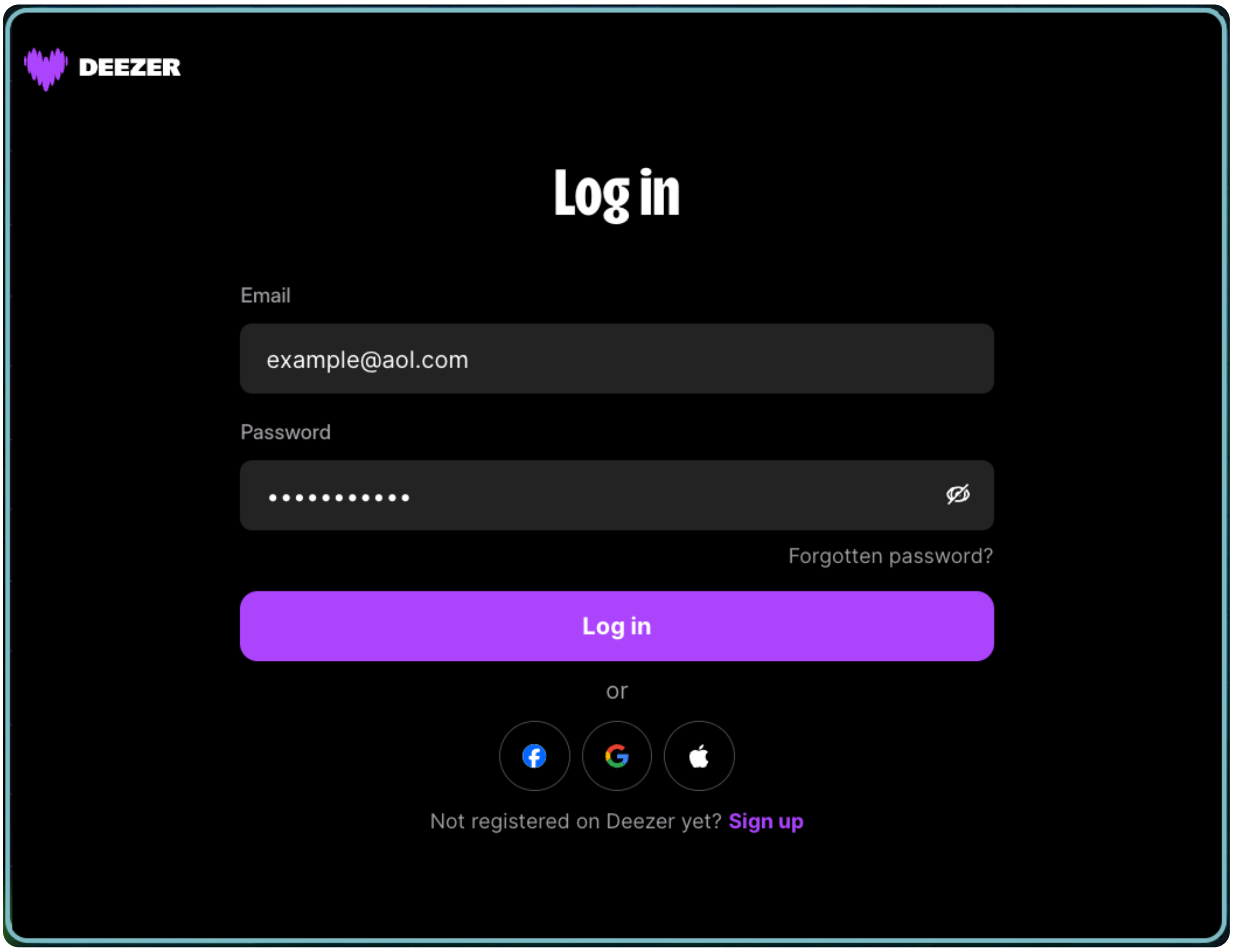


Figure: Login (email and password)

Suno

Suno uses sign-in on **suno.com** in the embedded browser (Clerk session). DSM's Suno integration proxies that session for playback and capture; you must comply with Suno's Terms of Service and Privacy Notice. If you record or reuse Suno-generated audio in your productions, **commercial rights** depend on your Suno plan (only Pro- and Premier-eligible output qualifies for commercial use under Suno's rules). See Suno's Rights & Ownership FAQs and License Information: Service terms and copyright.

Web Pages and Sources

Each source loads in the **center** area when the integrated browser view is active (see Center – Webpage or Stems).

- **RadioGarden** provides access to worldwide radio streams (subject to regional availability).
- **YouTube** playback runs within DSM's integrated browser environment.
- **78RPM** provides access to large historical vinyl archives.

Tip: For fast quality improvement, try denoise → restore (Apollo), or restore → separate. Processing order can significantly affect results.

Tools Strip (Vertical)

The tools strip provides quick-access controls:

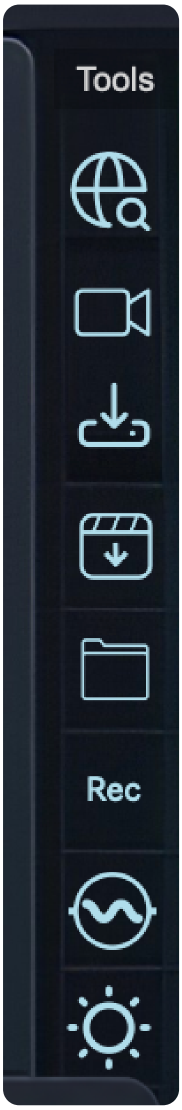


Figure: Tools Strip

- **Search**, open a website by typing a URL (example.com; no http required).
- **Video**, show the video window (YouTube or ClipCafe).
- **Download Audio**, download audio from the active source. When ready, drag the file into Live.
- **Download Video**, download video; drag into Live or Finder when complete.
- **Output Folder**, set the destination folder for downloads.
- **Record**, capture streaming audio (SoundCloud, Deezer, Suno, etc.).
- **Separate**, run separation or restoration using the selected model.
- **Theme**, toggle YouTube light/dark mode (legacy feature).

Left Panel – DSM (Models & Filters)

The left side of the device holds the models list and filters. Zoom and navigation for the browser view sit at the top of this panel.

Zoom & Navigation

Zoom adjusts the scale of the webpage shown in the center. Forward/back arrows navigate web pages.

Figure: Zoom and Navigation

Models Panel

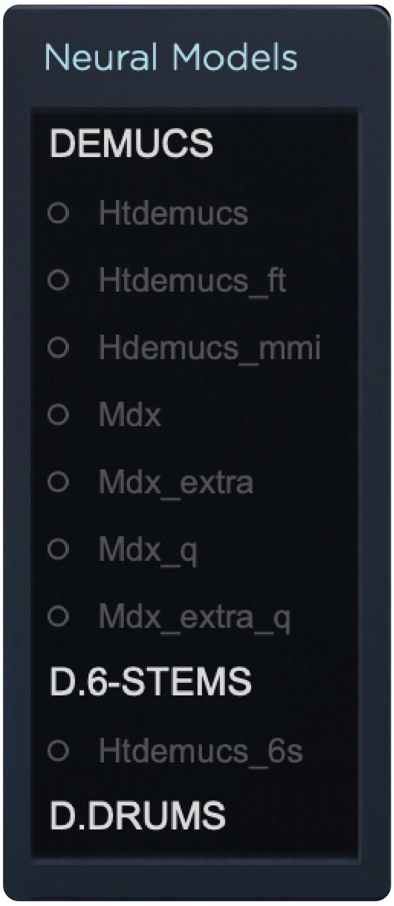


Figure: Models Panel

Scrollable list of all 63 models, grouped internally by architecture and stem type. The selected model is highlighted and its advanced parameters appear in the right panel.

Instrument Selector

A row of stem-type filter buttons (Vocals, Drums, Guitar, Piano, Bass, Wind, Mix, HQ, Denoiser, Debreath, Prompt/Text) that narrows the model list to only relevant models. The same control is available in the right panel. For how to use it, see Controls: Instrument Selector.

Figure: Instrument Selector

Center – Webpage or Stems

The **center** is one large area that displays either:

- **Webpage**, the integrated browser for the active source (YouTube, SoundCloud, Suno, RadioGarden, 78RPM, etc.)
- **Stems**, the stem filter view and separation workflow

The **Browser / Stems switch** in the **right panel** controls which view is shown here. Use Browser to see and interact with the webpage; use Stems to work with stem types and separation.

Sampler and Waveform (Bottom Left)

The sampler and waveform sit at the **bottom left**. This is where you load a file (drag & drop or folder icon), set crop markers, and run separation or restoration.

File Path, Waveform & Crop

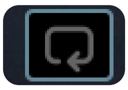


Figure: Loop button

Displays the loaded file path. The waveform shows the full audio with a selectable crop region (start/end markers). Only the cropped region is used for playback and neural processing. A **loop** button controls the loop region. Load audio via the file icon or by dropping a file onto the waveform panel. For detailed cropping and playback workflow, see Controls: Sampler and cropping.

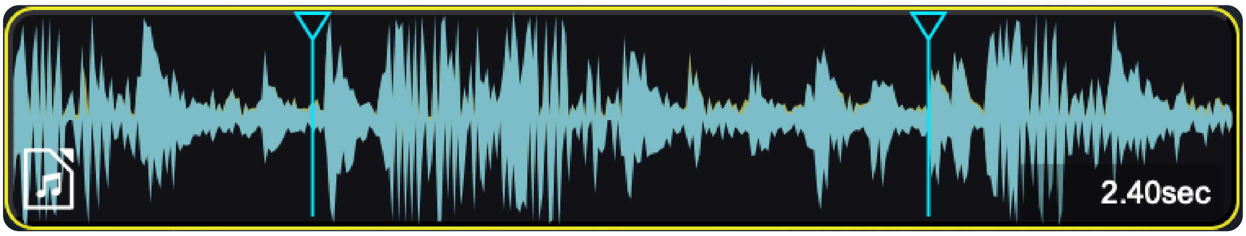


Figure: Waveform and Sampler

> ! **Note:** Web audio continues playing while the sampler plays local audio. Pause the web source to avoid overlapping playback.

Each stem has its own waveform, volume slider, and drag-to-import handle. For full stem interaction (click to solo, drag to import, double-click multi-select, version scrolling), see Controls: Stem interaction.

Right Panel – Browser/Stems Switch, Instrument Selector, Info Panel, Advanced Controls

The right panel controls the center view and provides the Instrument Selector (stem filter and recording actions), contextual help, and model-specific parameters.

Browser / Stems Switch

Figure: Browser / Stems Switch

Controls what is shown in the **center**: the integrated webpage (browser) or the stems view. Toggle to switch between the two.

Volume


Figure: Volume

Master output level control.

Info Panel

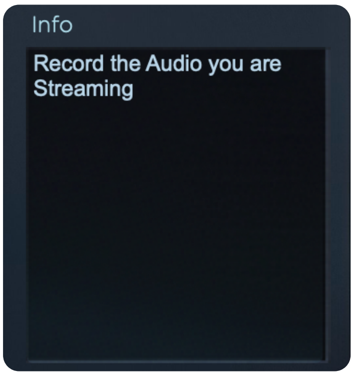


Figure: Info Panel

Context-sensitive text panel that shows control descriptions on hover, model guidance on selection, and device status messages. For details, see Controls: Info tab.

Node view

Use the **chain** button at the **bottom-right** of the **Info panel** to switch between the usual **Info panel** and **Node view**. Node view is a **chain view** (a simple **signal-flow** diagram). Each block is one step in your processing chain, for example **Separate**, **Clean**, or **Enhance**, so you can see what runs first, what comes next, and how audio flows from input to output. You can have **between two and four** steps: a new chain usually starts with **Separate** then **Clean**, and any **extra** steps you add often default to **Enhance**. With **two** steps, the chain uses a compact layout; with **three or four** steps, the blocks stack in a column so the full flow is easier to read. For which models appear under each tab, see Models Overview: Separate, Clean, and Enhance (Node view).

Figure: Node view with two processing steps

For how the Info panel behaves when you are not in Node view, see Controls: Info tab.

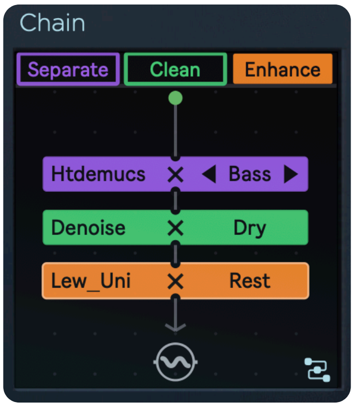


Figure: Node view with three processing steps

Advanced Controls

Architecture-specific parameter panel that appears when a model is selected. Options vary by model family (Demucs, BS-RoFormer, Spleeter, VR/Wind, Apollo, AudioSep, Denoise). Parameter values are stored per model. For a full reference of each parameter, see Controls: Advanced controls reference.

Bottom Bar

Progress Bar

Displays:

- Download progress
- Model processing progress
- Device version number

Figure: Progress Bar

← Introduction
Controls & Workflow →
Downloading →
Models Overview →

Controls & Workflow

This page explains how the sampler, cropping, stem interaction, and model filtering work, and how to get the best results from neural processing.

For the layout of all on-screen controls, **Web bar** (sources), **tools strip** (search, video, download, record, separate, theme), **left panel** (zoom, models list, instrument selector), **sampler** (waveform, crop, file path, loop, stems scroll, duration), and **right panel** (Browser/Stems switch, volume dial, Instrument Selector, Info tab), see Interface.

Sampler and cropping

DSM uses an internal sampler with region-based cropping so you can process a selected portion of the loaded file instead of the entire track. The **waveform** is the sampler: load or drop a file, set the crop, then play or process. **For playback**, click the waveform in the sampler or click on a stem.

> **Web Audio and local playback:** Web Audio from the webpage (e.g. YouTube, Suno, RadioGarden) keeps playing during local sampler playback. It is recommended to **stop or pause the webpage audio source** before using the sampler playback so you hear only the local audio. The system already avoids clicks and DC issues during audio manipulation.

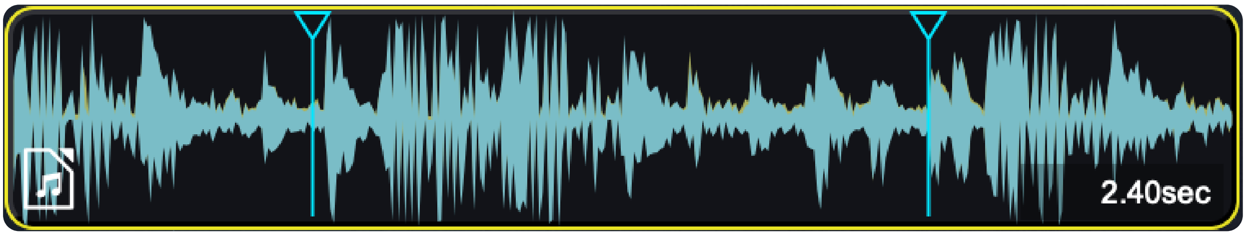


Figure: Waveform and sampler (main workspace)

Start and end markers

Use **start and end marker mode** to define the region to process or play. The **yellow button on the right** of the waveform area toggles or confirms marker mode. You set the in and out points of the crop there.

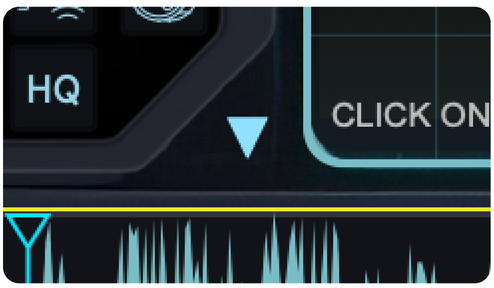


Figure: Marker mode (start/end crop)

Resizing the crop by dragging

You can also **drag to resize the crop** directly on the waveform: drag the edges of the selection to change the start and end. This is the same region used for playback and for neural processing when you run a model.

Loading time and file length

Audio is loaded in **chunks**, so longer files still take longer to prepare when cropping, but much less than before—for example, about 54–56 seconds for roughly 10.5 minutes of audio, depending on the computer.

Caps Lock: highlight section to process

With **Caps Lock enabled**, you can highlight a specific waveform region. Only the highlighted region is sent to the neural model when processing. Use this to run separation or restoration on a short loop or a specific part of the track.

Recording and loading into the sampler

Audio can reach the **sampler waveform** in several ways. Regardless of the source, all audio is loaded into the same waveform area for playback, cropping, and processing:

- **Drop a file**: Drag an audio file from your system onto the waveform panel; it loads directly into the sampler.
- **Open from system folder**: Use the **music file icon** at the bottom left of the waveform to open a folder and choose a file.
- **Record from the webpage**: With a source selected (e.g. SoundCloud, Deezer, Suno, Radio Garden), use the **record** button on the tools strip. Recording goes to the output folder; you can then open or drop the recorded file into the sampler.
- **Download from YouTube or ClipCafe**: Use the **download audio** (or **video**) button on the tools strip. When the file is ready, drag from the button into your project or open it from the output folder and load it into the sampler.

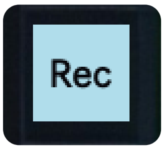


Figure: Record button (tools strip)


Figure: Download Audio (tools strip)


Figure: Download Video (tools strip)

Once loaded, the audio appears in the waveform. Set the crop if needed, then run separation or restoration with the selected model. See Downloading for details on sources and output.

Waveform: loading and selection for separation

To **load an audio file**: use the **music file icon** at the bottom left of the waveform to open a system folder and choose a file, or **drop a file onto the waveform panel** and it will load directly. The stem that is **selected for separation** has a **yellow border**. When you **hover** over a stem, it is treated as selected for separation and the border highlights yellow; when not hovered, the currently selected stem keeps the yellow highlight. To run separation, use the **separate** button on the tools strip.

Stem interaction

In the center area, each stem (Bass, Drums, Vocals, Other, etc.) has its own waveform and controls. How you interact with it:

- **Click a stem**: Listen to that stem's playback (solo audition). The selected stem turns yellow. Click again or switch stem to change what you hear. You can also **click a stem to process it again** with the currently selected model (refine that stem only).
- **Drag a stem**: **Import that stem's file directly** into your project (e.g. drag into an Ableton Live track or the Session/Arrangement view).
- **Double-click a stem**: The stem turns **red** and enters multi-select mode. You can then **click other stems** to add them to the selection (they also turn red). **Drag the selection** to import **all selected stems together** into Ableton Live (e.g. onto different tracks or the same area).

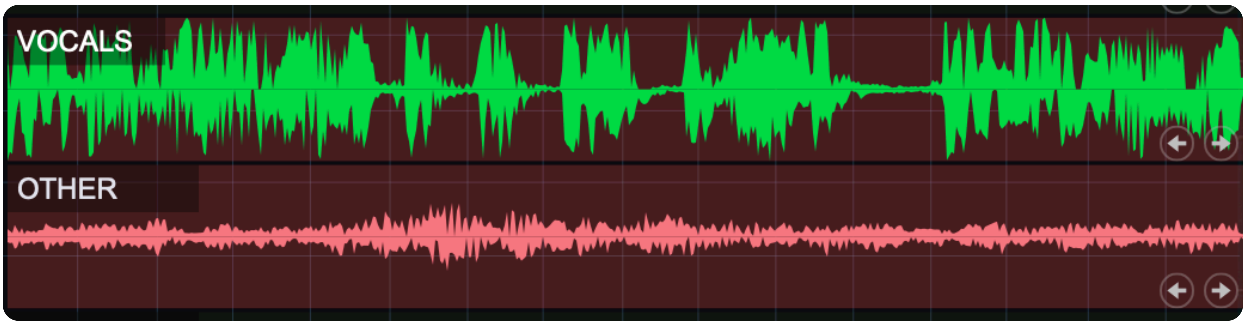


Figure: Double-click, stem turns red (multi-select)

Figure: Drag multiple selected stems into Live

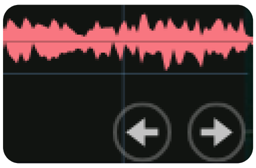


Figure: Arrow / scroll to move through stems

Use the **left and right arrow buttons** (or scroll) below the waveform to **move through stems** or change which stem is in view. When you have many stems, this lets you navigate without losing your history and compare different results versions.

Instrument Selector: filter models by stem type

In the right panel, the **Instrument Selector** lets you narrow down which models are shown in the left list. You select the **stem type(s)** you care about (e.g. vocals, drums, instrumental); the model list then shows only models that can **generate those stems**. You can select **multiple stem types in combination** (e.g. vocals + drums) to see all models that produce at least one of them. This makes it easier to pick a model when you know the output you want. The Instrument Selector here is the same stem-type filter as in the left panel (see Interface: Left panel).

Figure: Instrument Selector

Info tab

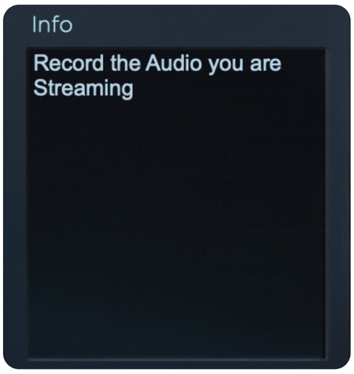


Figure: Info panel

The device has an **Info tab** in the right panel, the **Info panel**. Use it to read short help and status messages, tips for the current source or model, and other on-device information. When you hover over a control, the panel shows a short description of what it does; when you select a model, it updates with guidance for that model.

You can switch the same area to **Node view** with the **chain** button at the bottom-right of the **Info panel** to see your processing chain as blocks and flow lines; see Interface: Node view.

For how **Separate**, **Clean**, and **Enhance** relate to the model list, see Models Overview: Separate, Clean, and Enhance (Node view).

Neural processing and chunk size

Neural models process audio in **chunks**. Using **small files** (or a small cropped section) **does not reduce quality** as long as the duration is **longer than the chunk size** used by that model. Chunk sizes vary by architecture and model: typically in the **range of a few seconds to tens of seconds**. So a 5–10 second loop is fine; very short clips (shorter than a single chunk) may still process correctly but provide minimal temporal context for the model. Many models expose parameters in the right-panel **advanced controls** (chunk size, overlap, and others). When you select a model from the left list, the right panel shows that model's options. Apollo, for example, has **Chunk sec** and **Overlap / Chunk if >**; Audiosep has "Chunk if >". Other model families (DEMUCS, BS-RoFormer, Spleeter, etc.) have their own advanced controls.

Padding your crop (in and out): For the cleanest stem at the exact moment you want, leave roughly **a few seconds** of extra audio **before and after** that moment in the crop. Models analyse **fixed-length windows**; the **first and last instants** inside your selection sit at the **edges** of that context, so separation is often **less reliable there** than in the middle—by the nature of windowed processing, not a random glitch. Trim the exported stems in Live if you need a tight edit. This is different from **overlap** in advanced controls, which smooths joins *between internal chunks*; a slightly wider *user crop* still helps the sound at your region's start and end.

First run vs. same model again: The **first time** you run a model, it takes longer because the model is loaded into memory. **When you run the same model again** (without switching to another), it stays loaded and the next separation is **faster**. Switching to a different model triggers its initial load into memory, which increases processing time on first run.

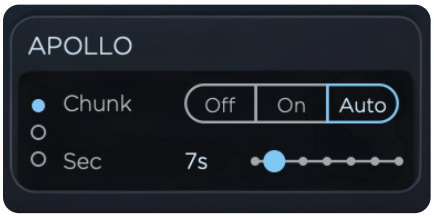


Figure: Apollo Chunk sec

Figure: Apollo Overlap / Chunk if >

For per-model options, see the model-family pages (e.g. Apollo, Audiosep).

!

File length and workflow: In most models, using longer files does not **significantly** improve separation quality once the material exceeds the model's effective chunk or context size. Because DSM is designed primarily for **grabbing and isolating sound**, we recommend using **short regions (e.g. 10–20 seconds)** when possible. Cropping to the section you need keeps processing fast on any architecture and keeps your workflow responsive.

Chunk size and context by architecture

Most models operate on fixed-size chunks. **Increasing file length alone** does not improve separation quality unless the model's chunk size or attention window is also increased. Transformer-based models (e.g. BS-RoFormer, AudioSep) can benefit from longer audio context when chunk size allows; convolutional or local-restoration models do not.

Architecture	Context benefit	Why	Recommended region
BS-RoFormer	Strong	Transformer attention across time; larger chunk or segment improves harmonic continuity and long phrases	10–30 s (longer only if increasing chunk)
AudioSep	Moderate	Text-conditioned attention over sequence; longer context can improve semantic alignment	10–30 s
Demucs	Slight	Fewer segment boundaries and blends; longer segments can improve smoothness, not core separation	10–20 s

Architecture	Context benefit	Why	Recommended region
Spleeter	None	Fixed convolution receptive field; chunking only affects memory	5–20 s
Apollo	None	Local restoration model; longer audio does not improve quality	Any length
VR / Wind	Minimal	Fixed STFT-based windows	5–20 s

Even transformer models only benefit if chunk size is actually increased and VRAM allows larger segments. For most use cases (grabbing and isolating short material), 10–20 second regions are sufficient and keep the workflow responsive.

> **Important:** If chunking is set to Auto or a low threshold, longer files will still be internally split into small segments. In that case, increasing file length alone will not increase context.

Advanced controls reference

The right panel shows different parameters depending on the selected model. Below is what each type of control does and how it affects **quality** and **time**.

TTA (test-time augmentation)

What it does: The model is run **multiple times** with small input variations (e.g. time or phase shifts), and the results are **averaged**. That reduces random artifacts and improves consistency, especially at stem boundaries and in difficult passages.

Setting	Quality	Time	When to use
Off	Good, single pass	Fastest	Previews, quick checks, realtime-style workflow
On	Higher; fewer artifacts, smoother boundaries	Much slower (e.g. ~8× on BS-RoFormer)	Final stems, exports, when quality matters most

Where it appears: BS-RoFormer, VR/Wind (and some other architectures). Not all models expose TTA.

Quality preset (BS-RoFormer)

BS-RoFormer exposes a **Quality** dropdown that sets overlap, batch size, and TTA together. This is the recommended way to trade speed for quality without adjusting each parameter separately.

Preset	Overlap	Batch	TTA	Effect
Fast	2	4	Off	Fastest; good for previews and quick checks
Bal (Balanced)	4	2	Off	Balance of speed and quality; general use
High	8	1	On	Highest quality; much slower (~8×), best for final stems

Choosing **High** enables TTA and increases overlap while reducing batch size (one chunk at a time for maximum blend quality). **Fast** and **Bal** leave TTA off and use higher batch for speed.

Chunks and chunk size

What it does: Long audio is split into **chunks** (fixed-duration segments). Each chunk is processed by the model, then chunks are **reassembled** (often with overlap blending). Chunking keeps memory use bounded and allows long files to run; chunk size is a trade-off between context (longer = more context, sometimes smoother) and memory/speed (shorter = less memory, often faster).

Concept	Effect on quality	Effect on time / memory
Chunk size (e.g. Sec in Apollo, or model default)	Larger chunks can give slightly better context at boundaries; too small may hurt continuity.	Larger = more memory per chunk, fewer chunks; smaller = less memory, more chunks (and often more overlap work).
Chunk On/Off/Auto	No direct quality change; chunking avoids OOM on long files so you can run at all.	Off = one piece (fast for short files, can OOM on long); On/Auto = split into chunks (required for long files).
Chunk if > (threshold)	None.	Only when to trigger chunking (e.g. only if file > 30 s). Lower = chunk sooner; higher = chunk only for longer files.

Where it appears: Apollo (Sec, Chunk, Chunk if >), AudioSep (Chunk, Chunk if >). Demucs/BS-RoFormer use internal segment or overlap logic rather than a single “chunk size” slider.

Batch (chunks per batch)

What it does: How many **chunks are processed at once** (e.g. on GPU). Higher batch = more parallelism and usually **faster** total time, but **higher VRAM** use. 0 = worker default (auto).

Setting	Quality	Time	Memory (VRAM)
0 (default)	Same	Worker chooses	Moderate
1–2	Same	Slower (fewer chunks at once)	Lower
4–8	Same	Faster if GPU has headroom	Higher; may OOM on small GPUs

Where it appears: BS-RoFormer (Batch 0–8). Increase if you have VRAM and want speed; reduce if you hit out-of-memory errors.

Overlap

What it does: Consecutive chunks are processed with **overlapping** regions. The overlapping parts are **blended** (e.g. crossfaded) so there are no hard seams. More overlap = smoother boundaries and fewer “chunk line” artifacts, at the cost of more computation (each overlapping region is processed and then blended).

Setting	Quality	Time
0 / default	Good; possible slight seam at chunk edges	Fastest
Higher (e.g. 1–4 or 0.1–0.5)	Smoother boundaries, fewer edge artifacts	Slower (more overlap to compute and blend)

Where it appears: Demucs (Overlap 0–50, maps to 0.10–0.50), BS-RoFormer (Overlap 0–4 segments), Apollo (Overlap 0–4, maps to 0–2 s).

Segments (Demucs)

What it does: In Demucs, the input can be split into **segments** of a given length (in seconds or samples). Segment length affects how much context the model sees per pass. 0 = worker default. Shorter segments =

less memory per segment but more segments to process and reassemble; longer segments = more context, potentially smoother, but higher memory.

Setting	Quality	Time / memory
0 (default)	Worker default behaviour	Balanced
Shorter segments	May introduce more boundaries to blend	Lower memory; can be faster or slower depending on overhead
Longer segments	Fewer boundaries; can help continuity	Higher memory; may be faster if fewer segments

Where it appears: Demucs only (Segments slider 0–60).

Shifts (Demucs)

What it does: The model is run **multiple times** with the input **shifted** in time by a small amount each time; outputs are **averaged** (with inverse shift so they align). This is a form of test-time augmentation: it reduces phase-boundary artifacts and improves separation consistency. More shifts = better quality, **linearly slower** (e.g. 5 shifts ≈ 5× time).

Setting	Quality	Time
0	Single pass	Fastest
1–5	Better; fewer boundary/phase artifacts	Roughly proportional (2 shifts ≈ 2×, etc.)
5–10	Best quality	Slowest

Where it appears: Demucs only (Shifts slider 0–10). Use 0 for previews; 5–10 for final exports when quality is priority.

Two-step separation (first-step model)

Some models use a two-step pipeline: a first model extracts a preliminary stem, and a second model refines it. **HQ** vs **normal** (and Direct Mode) choose which model is used for the first step. See the architecture pages for full tables.

Architecture	Models that use 2-step	Step 1: HQ mode	Step 1: Normal mode
Demucs	**Inaki** (when Direct Off), **Filosax** (when Direct Off), **Sax_Trick** (when Has Vocals On)	Inaki: `hdemucs_mmi`. Filosax/Sax_Trick: no HQ option (mdx_extra_q / mdx_q).	Inaki: `mdx`. Filosax: `mdx_extra_q`. Sax_Trick: `mdx_q` then `mdx_extra_q`.
BS-RoFormer	**Mdx23c_6s**, **Mdx23c_5s** (when Direct Off)	`bsrofo_sw`	`bsroformer_4stem`

Details and full flow: Demucs advanced controls, BS-RoFormer advanced controls.

Summary by architecture

Architecture	Main quality vs. time controls	Chunk / batch / overlap
Demucs	**Shifts** (more = better quality, slower). GPU for speed.	Overlap (blend at segment edges). Segments (segment length).

Architecture	Main quality vs. time controls	Chunk / batch / overlap
BS-RoFormer	**Quality preset** (Fast / Bal / High) sets overlap, batch, and TTA together: Fast = overlap 2, batch 4, no TTA; Bal = overlap 4, batch 2, no TTA; High = overlap 8, batch 1, TTA on. Manual **TTA** and Overlap/Batch also available.	Overlap (0–8 segments). Batch (0–8 chunks at once; more = faster if VRAM allows).
Apollo	Chunk Sec (chunk length). Overlap (blend at edges).	Chunk On/Off/Auto; Chunk if > (threshold in seconds).
AudioSep	Chunk On/Off/Auto; Chunk if >. STFT (Auto/On/Off) for long files.	Storage (mmap vs full load). Chunk if > (seconds).
VR / Wind	**TTA** (On = higher quality, slower). High End, Post Proc, Aggression, Window.	—
Spleeter	**Spleeter RT** is an optimized realtime variant (not general Spleeter): better quality when run offline, faster and more efficient. It trades some flexibility for speed and efficiency. CX mode adds piano (Fast-2, Fast-5).	—

For system requirements and supported platforms, see Compatibility.

← Introduction · Interface · Downloading · Compatibility

Models Overview

DSM includes 63 models for source separation and restoration. Each model produces specific stem types (e.g. vocals, drums, instrumental, restored audio) using a defined neural process. The underlying architectures are described in Architectures.

Stem types and processes

Stem types refer to the outputs generated by a model. Depending on the architecture, outputs may include:

- Vocals (single or split lead/backing)
- Drums (4–6 drum components)
- Bass
- Other instruments
- Full instrumental mix
- Karaoke (instrumental or vocal-only variants)
- Restored audio (denoise, dereverb, restoration)
- Prompt-conditioned single-stem extraction

The model list in the device shows how many stems each model outputs.

Processes define how the model transforms audio:

- Separation: splits a mix into distinct stems
- Denoise: reduces background noise
- Dereverb: reduces reverberation
- De-bleed: removes leakage between sources
- De-breath: reduces vocal aspiration artifacts
- Restoration: reconstructs degraded frequency content
- Text-prompt extraction: extracts a described source via conditioning

Model families

Models are grouped by architecture. Each architecture represents a distinct neural design with different trade-offs in quality, speed, memory usage, and specialization.

Separate, Clean, and Enhance (Node view)

In Node view, the three tabs at the top of the chain (Separate, Clean, Enhance) sort models by the kind of processing you apply. They do not line up one-to-one with a single architecture name: one family can include models under more than one tab (for example BS-RoFormer).

- Separate: stem separation and isolation. See DEMUCS, SPLEETER, BS-ROFORMER, VR / Wind, and AUDIOSEP.
- Clean: cleanup and repair, such as denoise, dereverb, bleed control, and de-breath-style processing. See DENOISE (including Loopcloud) and the process-oriented variants on the BS-ROFORMER page.
- Enhance: restoration and clarity, mainly APOLLO, with related variants listed on BS-ROFORMER where applicable.

The cards below list each architecture in full; use them for model names, stems, and advanced controls.

DEMUCS

DEMUCS is a convolutional-transformer hybrid architecture for music source separation. It provides 4- and 6-stem configurations and includes specialized instrument-focused variants. Key controls: GPU acceleration, Shifts (test-time augmentation), Segments, Overlap. Models: Htdemucs, Htdemucs_ft, Hdemucs_mmi, Mdx, Mdx_extra, Mdx_q, Mdx_extra_q, Htdemucs_6s, Inaki (drums), Filosax, Sax_Trick (saxophone).

SPLEETER

SPLEETER is a lightweight separation architecture optimized for speed. While separation quality (SDR) is generally lower than Demucs or BS-RoFormer, it is useful for rapid previews, de-bleeding, and quick stem extraction. Modes: RT (fastest) and CX (adds piano). Models: Fast-2, Fast-5, Fast-4, Ultra-Fast.

BS-ROFORMER

BS-RoFormer is a transformer-based separation architecture designed for high-quality music demixing. It supports multiple stem configurations, quality presets, and optional test-time augmentation (TTA) for improved artifact reduction. Additional models include targeted vocal, instrumental, dereverb, de-bleed, aspiration (de-breath), chorus separation, and crowd extraction variants. See the BS-ROFORMER page for the full list.

VR / Wind

VR / Wind models focus on voice and wind-instrument isolation. Output modes include vocal-only, wind-only, or complementary “other” stems. Advanced controls include TTA, High-End compensation, Aggression, Window, and post-processing options. Models: Wind, HP_Vocal, HP_Vocal3, HP_Vocal4, Karaoke_HP.

APOLLO

APOLLO is a neural restoration architecture designed to reconstruct high-frequency content and improve clarity in compressed or degraded recordings. Chunking for long files. Models: Lew_Uni (universal), Official, Lew_V2 (voice), EDM_Big (electronic).

AUDIOSEP

AudioSep is a text-conditioned separation model. Users describe the desired source (e.g. “vocals”, “snare drum”), and the model extracts the corresponding component. This enables flexible single-stem extraction beyond fixed stem configurations. Main model: AudioSep. Controls: STFT, Storage, Chunk.

DENOISE

DENOISE provides envelope-matched noise reduction using a bundled noise profile. It is intended for cleanup rather than stem separation. Output: restored audio. Model: Loopcloud (dsu-denoise). See DENOISE for a best-use recommendation (e.g. 78 RPM / vinyl).

Choosing a model

When choosing a model:

- Select by stem type (vocals, drums, instrumental, restoration, etc.)
- Match the process to your goal (separation vs restoration vs prompt-based extraction)
- For maximum quality, enable TTA or higher-quality presets where supported

- For speed or previews, use lightweight models (e.g. Spleeter or low-shift settings)

For architectural details and processing trade-offs, see Architectures.

← Introduction · Architectures · Credits

DEMUCS

Neural network–based source separation with GPU support, shifts, segments, and speed controls. Use the GPU toggle for speed; Shifts for quality.

4-stem models

Htdemucs 4 stems

Htdemucs_ft 4 stems: Fine-tuned, better quality, slower

Hdemucs_mmi 4 stems: Multi-domain, good for mixed content

Mdx 4 stems: Fast, lower quality

Mdx_extra 4 stems: Extended training

Mdx_q 4 stems: Quantized, smallest memory

Mdx_extra_q 4 stems: Best quantized, fast and good quality

6-stem (D.6-STEMS)

Htdemucs_6s 6 stems: Keys and guitar included

Drums (D.DRUMS)

Inaki 4 stems: Kick, snare, toms, cymbals. Direct Mode skips pre-separation

Instrument (D.INST)

Filosax 2 stems: Saxophone. Direct Mode skips vocal removal

Sax_Trick 2 stems: Best sax via two-pass extraction

Advanced controls

When a DEMUCS model is selected, the right panel shows these parameters (paginated if needed). GPU is handled automatically (MPS on ARM Mac, CUDA on Windows/Linux, else CPU).

Control	Type	Range / options	Description
Threads	Tabs	1, 2, 4, 8	CPU threads (worker speed).
Overlap	Slider	0–50	0 = worker default 0.25; 10–50 = 0.10–0.50 overlap.
Shifts	Slider	0–10	More shifts = higher quality, slower.

Control	Type	Range / options	Description
Segments	Slider	0–60	Segment length (0 = default).
Hard Clip	Toggle	Off / On	Hard clip output.
Storage	Tabs	SSD, HDD	SSD = faster I/O; HDD = full load strategy.

Two-step separation: which models, which first step

Some Demucs models run in two steps: a first pass extracts a stem (e.g. drums or "other"), then a second model runs on that stem. The table below shows **which models** use two-step, **when** it applies, and **which model is used for step 1** (HQ vs normal).

Model	When 2-step runs	Step 1 (first pass)	Step 2
Inaki (D.DRUMS)	When **Direct Mode** is Off (default)	**HQ Source Off**: `mdx` (fast). **HQ Source On**: `hdemucs_mmi` (best quality). Both extract drums only.	Inaki on the drums stem (kick, snare, toms, cymbals).
Filosax (D.INST)	When **Direct Mode** is Off (default)	Always `mdx_extra_q` (full 4-stem). No HQ/normal option.	Filosax on the "other" stem (sax lives there).
Sax_Trick (D.INST)	Only when **Has Vocals** is On	Step 1: `mdx_q` with two-stems vocals (sax + vocals go to "vocals").	Step 2: `mdx_extra_q` on that "vocals" stem → sax to "other", real vocals stay.
Sax_Trick (D.INST)	When **Has Vocals** is Off	Single step only: `mdx_extra_q` (sax ends up in "vocals" stem).	—

Model-specific (Inaki: D.DRUMS)

Control	Type	Range / options	Description
Direct Mode	Toggle	Off / On	Off = two-step (extract drums with mdx or hdemucs_mmi, then Inaki). On = skip step 1; use when source is already drums.
HQ Source	Toggle	Off / On	Only for step 1 when Direct is Off. **Off** = `mdx` (fast). **On** = `hdemucs_mmi` (higher quality drum extraction).

Model-specific (Filosax: D.INST)

Control	Type	Range / options	Description
Direct Mode	Toggle	Off / On	Off = two-step (mdx_extra_q full separation, then Filosax on "other"). On = skip step 1; run Filosax on the full mix.

Model-specific (Sax_Trick: D.INST)

Control	Type	Range / options	Description
Has Vocals	Toggle	Off / On	On = two-step (mdx_q → mdx_extra_q) when sax is mixed with vocals. Off = single pass mdx_extra_q (sax in “vocals” stem).

← Introduction · Interface · Models Overview

SPLEETER

The fastest separation architecture in DSM, ready for realtime. Spleeter RT is an optimized realtime variant (not general Spleeter): when run offline it produces better quality and is faster and more efficient than before. CX mode adds a piano track where available.

Why use Spleeter

Spleeter RT in DSM is an **optimized realtime** implementation: it is the **fastest model** in the device and best suited to **realtime or near-realtime** use. When used **offline**, it has been optimized to produce **better quality** and to run **faster and more efficiently** than the original. Separation quality (SDR) is still lower than Demucs or BS-RoFormer, but it remains **valuable for quick previews**, auditioning, debleeding, and cleanup and for offline runs when you want speed and efficiency.

Models

Fast-2 2 stems: Vocals + instrumental (CX mode)

Fast-5 5 stems: With dedicated piano (CX only)

Fast-4 4 stems: Works in RT and CX

Ultra-Fast 4 stems: Realtime preview quality (use RT mode)

Modes

RT: Realtime-optimized variant; fastest, good for previews and for offline use (better quality and efficiency than before).

CX: Adds piano as a separate stem where the model supports it (e.g. Fast-5, Fast-2).

Advanced controls

SPLEETER has **no adjustable options** in the right panel. Mode (RT/CX) and stem count are determined by the model you choose (Fast-2, Fast-4, Fast-5, Ultra-Fast). Select the model and run; no extra parameters to set.

← Introduction · Interface · Models Overview

BS-ROFORMER

State-of-the-art separation with Quality presets and TTA (test-time augmentation). TTA is slower but gives higher quality; use Fast preset for previews.

BS.4-STEM & BS.6-STEM

SCNet_XL 4: Highest rated 4-track; TTA for max quality

Bsroformer 4: Standard; Fast for previews, TTA for finals

Aname_Large 4: Best drum isolation; TTA recommended

Bsrofo_sw 6: Full band; High + TTA for best results

Logic_Rofo 6: Great low end; TTA improves bass/kick

BS.VOCALS

Melband 1: Cleanest voice isolation

Viperx 1: Excellent clarity

Mdx23c 2: Fast voice extraction

Resurrect_V 1: Recovers buried voices

Revive2 1: Minimal instrument bleed

Revive3e 1: Natural tone

Gabox_Fv7 1: Newer, lighter, faster

BS.INST (instrumental)

Resurrect_I 1: Clean instrumental

Gabox_Fv7z, Fv8, Fv4, Fv9 1: Full instrumentals

FlowersV10 1: Newest v10

V1e_Plus 1: Low noise floor

BS.KARAOKE

Becruily 2: Lead and backing

Aufr33 1: Clean karaoke

BS.DNR (speech / music / effects)

BandIt_Plus 3: Speech, music, effects

BS.PROCESS (cleanup)

Denoise 1: Background noise

Dereverb 1: Room ambience

BleedSupp 1: Leaked voice

DenoiseDB 1: Artifacts from other models

BS.SPECIAL

Aspiration 2: De-breath

Apollo_MSST, Apollo_Full, Apollo_Off 2: Vocal restore

Chorus_MF 2: Male/female duets

Crowd 1: Audience extract/remove

BS.DRUMS & BS.GUITAR

Mdx23c_6s 6: Full kit

Mdx23c_5s 5: Five-way kit

Guitar (Guitar) 1: Isolate guitar

Advanced controls

When a BS-ROFORMER model is selected, the right panel shows these parameters. Overlap: 0 = worker default; 1–4 = num_overlap. Batch: 0 = worker default; 1–8 = chunks at once (VRAM).

Control	Type	Range / options	Description
Quality	Tabs	Fast, Bal, High	Fast = preview; High = best quality.
Overlap	Slider	0–4	Overlap segments (0 = default).
Batch	Slider	0–8	Chunks per batch (0 = default).
TTA	Toggle	Off / On	Test-time augmentation; 8× slower, higher quality.
Threads	Tabs	Auto, 4, 8, 16	CPU threads.
FLAC	Toggle	Off / On	Output FLAC instead of WAV.
Depth	Tabs	Float, 24b, 16b	PCM bit depth (Float, PCM_24, PCM_16).
Storage	Tabs	SSD, HDD	SSD = faster; HDD = full load.

Two-step separation: drum models (Mdx23c_6s, Mdx23c_5s)

Only **Mdx23c_6s** and **Mdx23c_5s** use two-step separation. When **Direct** is Off, step 1 extracts drums from the full mix; step 2 runs the selected drum model on that drums stem. The first step uses either a normal or an HQ extraction model:

Model	When 2-step runs	Step 1 (first pass)	Step 2
Mdx23c_6s	When **Direct** is Off (default)	**HQ Source Off**: `bsroformer_4stem` (fast). **HQ Source On**: `bsrofo_sw` (best quality).	Mdx23c_6s on the drums stem (full kit split).
Mdx23c_5s	When **Direct** is Off (default)	**HQ Source Off**: `bsroformer_4stem` (fast). **HQ Source On**: `bsrofo_sw` (best quality).	Mdx23c_5s on the drums stem (five-way split).

Overlap and Batch are hidden for these models. Instead:

Control	Type	Range / options	Description
Direct	Toggle	Off / On	Off = two-step (extract drums with bsroformer_4stem or bsrofo_sw, then run Mdx23c_6s/5s). On = skip step 1; use when source is already drums.
HQ Source	Toggle	Off / On	Only for step 1 when Direct is Off. **Off** = `bsroformer_4stem` (normal, faster). **On** = `bsrofo_sw` (HQ, best quality drum extraction).

← Introduction · Interface · Models Overview

VR / Wind

Voice and wind-instrument separation. Output can be Both, Wind only, or Other. Advanced options: TTA, High End, Post Proc, Aggression, Window.

Models

Wind 2: Flute, sax, clarinet, brass

HP_Vocal 2: Voice and music split, fast

HP_Vocal3 2: Improved voice

HP_Vocal4 2: Best voice extraction

Karaoke_HP 2: Remove lead for karaoke

Advanced controls

When a VR/Wind model is selected, the right panel shows:

Control	Type	Range / options	Description
Output	Tabs	Both, Wind, Other	Multi / Woodwinds / No Woodwinds: which stem(s) to output.
TTA	Toggle	Off / On	Test-time augmentation for higher quality.
High End	Toggle	Off / On	Better high-frequency preservation.
Post Proc	Toggle	Off / On	Cleaner output, fewer artifacts.
Aggression	Slider	0–20	Default 5. Higher = more aggressive separation; may cause artifacts.
Window	Tabs	320, 512, 1024	Window size (default 512).

← Introduction · Interface · Models Overview

APOLLO

Restore quality of compressed or lossy audio. Long files can be split into chunks to avoid memory limits and speed up processing; short files stay in one piece.

Models

Lew_Uni 1: Best universal restoration, any lossy file

Official 1: General-purpose enhancement

Lew_V2 1: Lightweight voice, fast

EDM_Big 1: Optimized for electronic music

Advanced controls

When an APOLLO model is selected, the right panel shows chunk and CPU options. Worker defaults: chunk_seconds=7.0, chunk_overlap=0.5. Overlap slider 0–4 maps to 0–2.0 s.

Control	Type	Range / options	Description
Chunk	Tabs	Off, On, Auto	Off = no chunking; On = always chunk; Auto = chunk when file > threshold.
Sec	Slider	5–20	Chunk size in seconds (default 7).
Overlap	Slider	0–4	Blend at edges (0.5 s per step; display as 0.0–2.0 s).
Chunk if >	Slider	0–60	Auto chunk only when file length exceeds this many seconds (0 = Auto).
CPU	Tabs	Auto, 4, 8, 16	CPU threads.

← Introduction · Interface · Models Overview

AUDIOSEP

Text-conditioned separation: you enter a prompt describing what you want (e.g. “vocals”, “drums”, “guitar”), and the model extracts that source.

Model

AudioSep 1: Prompt-conditioned separation

Tips and behaviour

Using words like **isolate** or **extract** in the prompt often gives predictable, useful results. For some material (e.g. harmonica) you may hear leaks or artefacts; we are working toward adding more models and more advanced architectures. For now, AudioSep is one of the first lightweight text-conditioned architectures that still produces meaningful separation—worth trying for creative or quick extraction tasks.

Advanced controls

Enter the **prompt** (text description of what to isolate) in the device’s prompt field. When AudioSep is selected, the right panel shows:

Control	Type	Range / options	Description
STFT	Tabs	Auto, On, Off	Auto = choose by duration; On = force torch STFT; Off = disable.
AutoSec	Slider	10–120	Threshold (seconds) for STFT Auto; file duration ≥ this uses torch STFT (default 60).
Storage	Tabs	SSD, HDD	SSD = mmap (faster load); HDD = full load.
Chunk	Tabs	Off, On, Auto	Off = no chunk; On = always chunk; Auto = chunk only for long files.
Chunk if >	Slider	5–60	Auto chunk threshold in seconds (default 30).

← Introduction · Interface · Models Overview

DENOISE

Envelope-matched noise reduction using a bundled noise profile (stored in the device's media folder). Output stem: restored.

Model

Loopcloud (denoise_loopcloud) 1: dsu-denoise with bundled profile

Use this when you want to reduce consistent background noise (hiss, room tone) using the built-in profile. For learning a custom noise profile or more advanced denoising, see the BS-ROFORMER Denoise and related process models.

> **Best use:** DENOISE is particularly well suited to removing background noise and vinyl crackle from 78 RPM and other vinyl sources while preserving tonal character. Use it in professional transfer and restoration workflows for best results.

Advanced controls

Mode sets the release time for the envelope: Default = vocals/melodic; Drums = kicks/percussive (reduces pumping); Slow = pads/sustained. Subtract = noise subtraction strength (50–100).

Control	Type	Range / options	Description
Mode	Tabs	Default, Drums, Slow	Default: 0.1 s release (vocals/melodic). Drums: 0.3 s (percussive). Slow: 0.5 s (pads/sustained).
Subtract	Slider	50–100	Noise subtraction strength; maps to 0–1 internally. Higher = more aggressive (default 90).

← Introduction · Interface · Models Overview

Architectures

DSM integrates multiple neural and signal-processing architectures. Each architecture represents a distinct design philosophy (convolutional, transformer-based, prompt-conditioned, or spectral) and determines how audio is analyzed, separated, or restored.

Architecture comparison

Architecture	Type	Strength	Best use
Demucs	Hybrid CNN/Transformer	Balanced quality	General music separation
BS-RoFormer	Transformer (mel-band)	Highest separation quality, extensive specialized variants	Final exports
Spleeter	U-Net	Fastest	Previews, quick stems
VR	Vocal-focused networks	Voice isolation	Karaoke, wind isolation
Apollo	Restoration network	Perceptual detail enhancement	Compressed audio
AudioSep	Text-conditioned model	Flexible target extraction	Specific instrument queries
Denoise	Spectral process	Noise reduction	Broadband noise reduction

DEMUCS

What it is: A hybrid convolutional architecture operating in the time and frequency domains, originally developed by Facebook Research (Meta AI). Later versions incorporate transformer components for improved context modeling.

How it's used in DSM: Multiple variants (Htdemucs, Mdx, etc.) are trained to separate 4 or 6 stems (vocals, drums, bass, other) or dedicated targets (e.g. drums only, sax only). You can run with GPU, adjust shifts and segments, and trade speed for quality. Outputs are time-aligned discrete stem tracks.

Demucs (GitHub)

BS-RoFormer (Mel-Band RoFormer)

What it is: A transformer-based architecture operating in mel-band frequency space, designed to model long-range temporal dependencies in music. Originally proposed by ByteDance AI Labs; open implementations and training ecosystems maintained by Phil Wang (lucidrains) and Roman Solovyev (ZFTurbo).

How it's used in DSM: Many of the 63 models are BS-RoFormer–based. Specialized variants include targeted vocal processing (aspiration removal, dereverb, de-bleed), chorus separation, crowd extraction, and restoration-oriented configurations. Quality and TTA (test-time augmentation) controls let you balance speed and quality. Each model configuration is trained for a specific stem or process type.

BS-RoFormer (GitHub) · ZFTurbo training

Spleeter

What it is: A U-Net–style model in the time–frequency domain, from Deezer Research. Trained on MusDB for 2, 4, or 5 stems.

How it's used in DSM: DSM includes Spleeter RT, an optimized real-time-oriented variant. It provides the fastest processing among supported architectures. Separation quality (SDR) is generally lower than Demucs or BS-RoFormer but is suitable for previews, rapid auditioning, de-bleeding, and lightweight workflows. Memory footprint and CPU usage are lower than transformer-based models. RT and CX modes; Fast-2 (vocals + instrumental), Fast-4, Fast-5 (with piano), Ultra-Fast. Outputs are stem tracks.

Spleeter (GitHub)

Audio Separator (VR architecture)

What it is: The "VR" (Vocal Remover) family used in Ultimate Vocal Remover (UVR), distributed via python-audio-separator. These models employ different backbone networks specialized for vocal isolation and complementary source extraction.

How it's used in DSM: Wind-instrument isolation, voice/music split, karaoke (lead removal). Models: Wind, HP_Vocal, HP_Vocal3, HP_Vocal4, Karaoke_HP. You choose output (Both / Wind / Other) and advanced options (TTA, High End, Post Proc, Aggression, Window).

python-audio-separator (GitHub)

Apollo

What it is: A neural audio restoration model (Jusper Lee) designed to reconstruct high-frequency detail and enhance perceived clarity in compressed or degraded recordings.

How it's used in DSM: Restore perceived quality of compressed material. Models: Lew_Uni (universal), Official, Lew_V2 (voice), EDM_Big (electronic). Long files can be processed in chunks. Used in workflows where compressed material is first restored, then archived in lossless format for consistency.

Apollo (GitHub)

AudioSep

What it is: Text-conditioned source separation: a model that takes a natural-language prompt (e.g. "vocals", "drums") and extracts that source from the mix.

How it's used in DSM: One main model (AudioSep). You type a prompt describing what you want to isolate. Controls for STFT, storage (mmap vs full load), and chunking for long files. Output is a single stem corresponding to the semantic prompt.

Denoise (dsu-denoise)

What it is: Envelope-matched noise reduction using a fixed or learned noise profile (e.g. from Loopcloud). Not a neural separation model; it uses spectral/statistical noise reduction.

How it's used in DSM: One bundled profile (Loopcloud). Reduces consistent hiss or room tone. Output stem: "restored". Useful before or after other models.

> **Best use:** DENOISE is particularly well suited to removing background noise and vinyl crackle from 78 RPM and other vinyl sources while preserving tonal character. Use it in professional transfer and restoration workflows for best results.

← Introduction · Models Overview · Credits

Credits

DSM (Dynamic Split Module) builds upon prior projects including SplitWizard+ and YouTube4Live. The following projects, researchers, and contributors made this work possible. DSM does not claim authorship of the underlying research models and integrates publicly available architectures and trained weights under their respective licenses.

Core technologies

Demucs: Music Source Separation

Facebook Research (Meta AI). Authors: Alexandre Défossez, Nicolas Usunier, Léon Bottou, Francis Bach.
github.com/facebookresearch/demucs · MIT License

BS-RoFormer / Mel-Band RoFormer

Original research: ByteDance AI Labs. Implementation: Phil Wang (lucidrains). Training & models: Roman Solovyev (ZFTurbo).
github.com/lucidrains/BS-RoFormer
github.com/ZFTurbo/Music-Source-Separation-Training

Spleeter: Music Source Separation

Deezer Research. Authors: Romain Hennequin, Anis Khlif, Félix Voituret, Manuel Moussallam.
github.com/deezer/spleeter · MIT License

Apollo: Audio Restoration

Jusper Lee (Apollo project). Perceptual restoration of lossy audio.
github.com/JusperLee/Apollo

Audio Separator (VR Architecture)

Ultimate Vocal Remover: Anjok07. python-audio-separator: nomadkaraoke.
github.com/nomadkaraoke/python-audio-separator · MIT

AudioSep (text-conditioned separation)

Audio-AGI: “Separate Anything You Describe”. DSM integrates the published implementation under its respective license and acknowledges the original authors. github.com/Audio-AGI/AudioSep · MIT.

MDX / MDX-Net

MDX / MDX-Net architectures and trained weights from community repositories including TRvlvr and KUIELab SDX23. Community-developed variants and training ecosystems.

Model contributors

The following individuals and communities have contributed trained models, experiments, or improvements used within DSM:

GaboxR67 · pcunwa · jarredou · becruily · aufr33 · KimberleyJSN · baicai1145 · ChenTechnology · Aname-Tommy · anvuew · xavriley · Politrees · deton24 (Lew) · essid · ZFTurbo · TRvlvr · MVSep · kuielab · Hugging Face community.

Integration & development

The DSM Max for Live integration, user interface, orchestration logic, and runtime system were developed by **Ostin Solo** and **VSTOPIA**.

Ostin Solo · Dynamic Split Module · SplitWizard

ostinsolo.co.uk

VSTOPIA

www.vstopia.com

Externals

Third-party Max/MSP externals used in the device.

11olsen.de

11dragfiles, external used in the device. 11olsen.de
11globalForegroundWindow, external for the foreground application and window. On macOS, advanced window access may require Accessibility permission for Max or Live (see the object's help patcher).
11olsen.de

Platform and tools used

The device was built using the following software. We credit them here as a matter of professional practice.

Ableton Live Suite

Ableton Live was used to create and run the Max for Live device. Ableton and Live are trademarks of Ableton AG.

ableton.com

Cycling '74 (Max)

Max was used for the device's patching and integration. Max for Live is a collaboration between Ableton and Cycling '74.

cycling74.com

Distribution

This software bundles third-party executables and libraries; model weights are downloaded by the user from their original repositories or distribution endpoints. Each component remains subject to its original license terms. See the project License Information page, the **LICENSES** folder, **LICENSES.md**, and **LEGAL_POSITION.md** (in the project docs) for details.

Contact

For all contact details and emails, see Contacts.

← Introduction

Contacts

For support, technical inquiries, licensing questions, or collaboration proposals, contact the DSM team below. DSM is actively maintained and updated; feedback and technical reports are welcome.

Email

- **VSTOPIA**, support, distribution, licensing: contacts.vstopia@gmail.com
- **Ostin Solo**, development and technical matters: contact@ostinsolo.co.uk

Web

- www.vstopia.com
- ostinsolo.co.uk

← Introduction

License Information

DSM bundles third-party executables, libraries, and model pipelines. Each component is used under its own license. This page lists the main components and where to find their license terms. For full license texts, see the project's **LICENSES** folder and **LICENSES.md** where applicable. For the project's legal position and distribution notice, see **docs/LEGAL_POSITION.md**.

DSM is distributed as software infrastructure and does not include or provide copyrighted media content.

- DSM UI, integration and orchestration code: © VSTOPIA
- Separation/restoration engines: third-party, licensed individually
- Model weights: downloaded by user
- Media sources: subject to external terms
- Full licenses: see LICENSES folder

Legal position and distribution

This section summarises what we create and distribute, and how we comply with third-party licenses. For the full text, see the project's **LEGAL_POSITION.md** (in the docs folder).

What we create and distribute: The Max for Live device (UI, patching, integration), the JavaScript/Node.js orchestration code, architecture-specific builds and frozen executables, and the sampling/model-download structure. We bundle **executables** (frozen Python binaries) built from open-source repos. We do **not** bundle **model weights**, users choose which models to download from their original sources (Hugging Face, GitHub, etc.); DSM does not host, mirror, or directly redistribute third-party model weight files.

Compliance: We credit all third-party components (Credits UI and LICENSES folder), include or reference full license text where required, and avoid GPL/AGPL dependencies where possible. Users who download models are responsible for complying with each model's terms. For licensing questions: VSTOPIA (contacts.vstopia@gmail.com) or Ostin Solo (contact@ostinsolo.co.uk).

Core separation & restoration

Demucs

Source: github.com/facebookresearch/demucs
Authors: Alexandre Défossez, Nicolas Usunier, Léon Bottou, Francis Bach (Facebook Research / Meta AI)
License: MIT License
Full text: see **LICENSES/DEMUCS.txt** and Demucs LICENSE.

Full texts for other core components (Spleeter, BS-RoFormer, Apollo, Audio Separator) are in the project's **LICENSES/** folder and in **LICENSES.md**. The same citation-and-inclusion approach is used for all bundled components.

Spleeter

Source: github.com/deezer/spleeter
Authors: Romain Hennequin, Anis Khlif, Félix Voituret, Manuel Moussallam (Deezer Research)
License: MIT License
Full text: Spleeter LICENSE

BS-RoFormer / Mel-Band RoFormer

Implementation: github.com/lucidrains/BS-RoFormer (Phil Wang / lucidrains)
Training: github.com/ZFTurbo/Music-Source-Separation-Training (Roman Solovyev / ZFTurbo)
License: MIT (see project LICENSES/BSROFORMER.txt and BS-RoFormer LICENSE)

Apollo (audio restoration)

Source: github.com/JusperLee/Apollo
Author: Jusper Lee
License: Attribution-ShareAlike 4.0 International (CC BY-SA 4.0)
Full text: see project LICENSES/APOLLO.txt and Apollo LICENSE (JusperLee/Apollo)

Apollo is included under CC BY-SA 4.0. Where applicable, attribution is preserved and license text is included in the LICENSES folder. No modifications to Apollo's core source are redistributed without preserving its original license terms.

Audio Separator (VR architecture)

Source: github.com/nomadkaraoke/python-audio-separator
Author: nomadkaraoke (Andrew Beveridge). Based on Ultimate Vocal Remover (Anjok07).
License: MIT License
Full text: see project LICENSES/AUDIO_SEPARATOR.txt and python-audio-separator LICENSE

AudioSep (text-conditioned separation)

Source: github.com/Audio-AGI/AudioSep
Authors: Audio-AGI (Liu, Kong, Zhao, Liu, Yuan, Liu, Xia, Wang, Plumbley, Wang). Paper: "Separate Anything You Describe" (arXiv:2308.05037, 2023).
License: MIT License.
We use AudioSep in DSM and acknowledge their work.

Models and weights

Neural model weights are not bundled by default; users obtain them through DSM when models are installed. Each model has its own license (Hugging Face, GitHub releases, TRvlvr repository, etc.). Check the model card or repository for each model's terms. For a summary of model sources, see the project's **LICENSES/MODELS.txt** (if present) and Credits.

Executables and Python runtime

DSM distributes frozen Python executables (Demucs, BS-RoFormer, Audio Separator, etc.) built with cx_Freeze. Those executables embed Python, PyTorch, and many bundled site-packages. The following apply.

Python

Source: python.org
License: Python Software Foundation (PSF) License
Python is distributed as part of the frozen executables. Full text: Python 3 license.

cx_Freeze

Source: github.com/marcelotduarte/cx_Freeze
License: Python Software Foundation (PSF) License (PSF-2.0)
Used to build the standalone executables. cx_Freeze licensing.

PyTorch

Source: pytorch.org / github.com/pytorch/pytorch
License: BSD 3-Clause
PyTorch is bundled inside the separation/restoration executables. Full text: see PyTorch LICENSE and project LICENSES folder.

Bundled Python site-packages

The frozen executables include many Python packages (e.g. numpy, torch, torchaudio, onnxruntime, and others). Each has its own license. Full license texts for bundled dependencies are typically included in the project's **LICENSES** folder or in the build artifacts. You are responsible for complying with each package's terms when distributing or using the executables.

Bundled dynamic libraries and runtimes: The distribution may also include a large number of dynamic libraries (.so, .dll, etc.) and runtime dependencies. A full list can be maintained in **LICENSES/THIRD_PARTY.txt** (or equivalent) in the project distribution. We do not reproduce every library's full license in this manual; citing the component and pointing to the LICENSES folder (and to that list) satisfies common license requirements. When in doubt, check the project's **LICENSES.md** and **LEGAL_POSITION.md**.

Download and media tools

yt-dlp

Source: github.com/yt-dlp/yt-dlp
License: Unlicense (public domain dedication)
Used for downloading audio and video from supported sites. yt-dlp LICENSE.

FFmpeg

FFmpeg is not bundled with DSM. During installation, users may download FFmpeg directly from its official distribution source. FFmpeg is licensed under LGPL 2.1+ (with optional GPL components depending on build configuration). DSM invokes FFmpeg as an external command-line tool and does not link against FFmpeg libraries. Users are responsible for complying with FFmpeg's license terms. FFmpeg legal.

Node.js modules

The DSM project submits and distributes code that depends on Node.js packages (e.g. in `code/node_modules`). Each package has its own license. A dependency license report can be generated using tools such as `license-checker` or `npm-license-report` and may be included in the distribution or project documentation. The project does not claim rights over third-party npm packages; you must comply with their terms when using or redistributing the project.

Service terms and copyright

DSM can download or record from various sources. Those sources are owned by third parties and are subject to their terms of service, copyright, and trademark policies. By using DSM with a given source, you must comply with that service's rules and applicable law. The following are for reference only; always check the current terms on each service.

- **YouTube**: Google's Terms of Service, YouTube Terms of Service, and copyright policies (e.g. fair use, download restrictions) apply. YouTube and related marks are owned by Google.
- **Deezer**: Deezer's Terms of Use and copyright licensing apply. Deezer is a trademark of Deezer.
- **SoundCloud**: SoundCloud's Terms of Use and community guidelines apply. Content may be subject to creators' licenses.

- **Suno**: Suno's Terms of Service and Privacy Notice apply. Use of the service, your account, Submissions, and generated Output are governed by Suno; DSM and VSTOPIA do not grant any rights to Suno content. **Commercial use** of music made with Suno is limited to material created under eligible paid plans (e.g. Pro or Premier); free-tier output is for personal, non-commercial use unless Suno's terms say otherwise. See Suno's Rights & Ownership help category for current rules. Suno is a trademark of Suno, Inc.
- **Freesound**: Freesound content is licensed under Creative Commons or similar; check each sound's license. Freesound terms apply.
- **Radio Garden**: Radio Garden's terms and the terms of each radio station apply. Recording may be subject to broadcast and copyright law.
- **Loopcloud**: Loopcloud's terms and license agreements apply to any content accessed or downloaded via the service.
- **Google**: Where Google services are used, Google's Terms of Service and applicable policies apply.

DSM and VSTOPIA are not affiliated with these services. We do not grant any rights to their content; users are solely responsible for complying with each service's terms and applicable copyright law.

Recording and distribution disclaimer

Recording length and legality: Laws on recording streaming audio (e.g. maximum duration, personal vs. commercial use, redistribution) vary by country and jurisdiction. You are responsible for knowing and respecting the laws that apply to you (e.g. private copying levies, broadcast rights, performer rights) and for complying with each source's terms of service. DSM does not specify or enforce how long you may record; any such limits are imposed by law and by the services themselves, not by DSM. Use the recording feature only in a way that complies with applicable law and the relevant platform's terms.

ClipCafe and use of film / actor content

Music and dialogue from films and series: ClipCafe (and similar features) may allow access to music, dialogue, or other audio from movies, series, or other copyrighted works. Using such content in your own productions may require rights clearance (e.g. from rightsholders, collecting societies, or distributors). Unauthorized use of music or actor performances from films can lead to **legal action, including lawsuits**, for copyright or related rights infringement. DSM and VSTOPIA do not grant any rights to third-party content. You are solely responsible for ensuring that your use of any content obtained via DSM (including from ClipCafe) is lawful and properly licensed where required.

Other components

Any other runtime or build dependencies (e.g. additional Node or Python packages, dynamic libraries not listed above) each have their own licenses. See the project's **LICENSES** folder, **LICENSES.md**, **LICENSES/THIRD_PARTY.txt** (if present) for a list of bundled libraries, `package.json`, and the `node_modules` directory for the full dependency tree and individual license files. This manual cites the main components and points to those locations; including every license in full here is not required for compliance, provided the project ships or references the relevant notices elsewhere.

For distribution and attribution guidance, see the project's **LICENSES.md**. For contact and credits, see Contacts and Credits.

← Introduction