Signal Rooms

Signal Rooms

AWC Field Manuals · No. 3

Abdul Kavungal (AK)

AWARD WINNING CREATIONS

Signal Rooms
AWC Field Manuals, No. 3

Copyright © 2026 Abdul Jaleel Kavungal. All rights reserved.

Published by Award Winning Creations.

No part of this book may be reproduced or transmitted in any form without written permission from the publisher, except for brief quotations in reviews and the templates in Appendix A, which readers may adapt for their own use.

Statements about products, models, laws and court cases describe them as reported in public sources up to November 2026. These fields change quickly; each such statement is dated in the text, and readers should check current terms, specifications and law before relying on them.

Lantern Street Sound, used throughout as a running example, is a composite, illustrative studio. It does not exist, and its sessions, clients, settings and outcomes are illustrative.

Product and company names mentioned are trademarks of their respective owners. The information in this book is provided without warranty. Legal, licensing and consent decisions should be made with appropriate professional and legal advice. Protect your hearing: monitor at moderate levels.

Typography by TyAK: Keyslate Display and TyCode, under the SIL Open Font License 1.1. Body text set in Source Serif Pro.

Contents

Preface vii

PART I. THE ROOM AND THE SIGNAL 1

1. The Control Room Moves Into the Model 3
2. Sound, Hearing and Numbers 11
3. Signal Flow and Gain Staging 19
4. Rooms, Monitoring and Acoustics 27
5. The Core Processors 35
6. Loudness and Delivery 43

PART II. THE AI TOOLKIT 51

7. How Audio Models Hear and Speak 53
8. Repair, Restoration and Dialogue Cleanup 63
9. Taking Mixes Apart: Source Separation 73
10. Voices: Synthesis, Cloning and Conversion 81
11. Generative Music and Sound Effects 91
12. Machines at the Mix Bus 101

PART III. PIPELINES IN PRACTICE 109

13. From Bedroom to First Release 111
14. Dialogue, Podcast and Post-Production Sound 119
15. Game and XR Audio 127

PART IV. THE FRONTIER 137

16. Neural DSP and Real-Time AI 139
17. Machine Listening 147
18. Spatial, Immersive and Object-Based Audio 155
19. Provenance, Watermarks and Detection 163

PART V. RIGHTS, OPERATIONS AND WHAT COMES NEXT 171

20. Rights, Consent and Copyright 173
21. Designing and Costing the Pipeline 183
22. Research Directions to Watch 191

Appendix A. Templates and Checklists 199
Appendix B. Ninety-Day Plan 205
Glossary 211
Notes and References 219

Index 229

Preface

The first machine that could keep a sound and give it back was a cylinder wrapped in tinfoil, turned by hand. A person shouted into a mouthpiece, a needle cut a groove that rose and fell with the voice, and when the cylinder was turned again the needle traced the groove and the voice came back, thin and scratchy and unmistakably human. Thomas Edison's phonograph, patented in the United States in early 1878, is on the cover of this book because it is the simplest possible picture of what sound engineering is: a signal, a medium, a way in and a way out, and a person listening at the end who decides whether it worked.

Almost everything else has changed. The signal is now a stream of numbers. The medium is a hard drive, a network and, increasingly, a neural network that has learned to turn noise into speech, a stereo mix into separate instruments, a sentence into a voice and a paragraph into a song. Some of those systems are astonishing. Some of them are quietly wrong in ways that only a trained ear will catch. Many of them raise questions that the tinfoil cylinder never did: whose voice is this, what was the model trained on, and will a listener be told that a machine made it?

This book is a field manual for working in that world. It is written for two readers at once. The first is starting out: a home studio, a laptop, a microphone and a first song, podcast or short film to finish. The second already does this for a living, in a music studio, a broadcast facility, a post-production house or a game studio, and needs to decide which AI tools belong in a professional pipeline, how far to trust them, and how to keep clients, performers and the law on side. The two readers need the same foundations. A professional who skips them ends up trusting a model's output because it sounds plausible. A beginner who skips them ends up with a mix that falls apart on a phone speaker.

How the book is organised

Part I covers the room and the signal: how sound behaves, how hearing works, how a signal travels from a microphone to a file, why rooms lie to you, what the core processors do and how loudness is measured and delivered. None of it is about AI, and all of it is what you will use to judge AI.

Part II opens the AI toolkit. It explains how audio models represent sound, then works through repair and dialogue cleanup, source separation, voice synthesis and conversion, generative music and sound effects, and assisted mixing and mastering. Each chapter says what the tools do well, how they fail and how to check them.

Part III puts the tools to work in three pipelines: a music release from a home studio, dialogue-led work from podcasts to film and television post-production, and interactive audio for games and extended reality.

Part IV moves to the frontier: neural signal processing that runs in real time, machines that listen and describe what they hear, spatial and object-based audio, and the provenance systems that are starting to label and trace synthetic sound.

Part V deals with rights, consent and copyright, with designing and costing a pipeline, and with the research directions most likely to change the work in the next few years. The appendices collect templates, checklists and a ninety-day plan.

If you are new, read Part I in order and do the listening exercises; then read Part II in order. If you are experienced, skim Part I for the loudness and delivery material in Chapter 6, then jump to whichever part of the toolkit you need. Everyone should read Chapter 20 before putting a cloned voice or a generated track in front of the public.

A note on dates and claims

AI audio moves faster than book production. Model names, features, prices, court cases and bills change month to month. Wherever this book describes a specific product, model, ruling or law, it says when the information applies, and everything of that kind reflects public sources

available up to November 2026. Treat those passages as a dated snapshot and check the current state before you rely on them. Standards such as ITU-R BS.1770 and EBU R 128 change slowly and are described as they stood at that date.

The book does not invent statistics or quotations. Where a figure comes from a company's own announcement, the text says so, because a company reporting on its own product is a source with an interest. Lantern Street Sound, the studio that appears throughout as a running example, is a composite: it does not exist, and its sessions, clients and settings are illustrative.

A note on your ears

Every technique here depends on hearing, and hearing damaged by loud monitoring does not come back. Work at moderate levels, take breaks and check mixes quietly. A model can measure loudness for you; it cannot replace the ears you will need for the next thirty years.

Abdul Kavungal (AK) Melbourne, November 2026

PART I

The Room and the Signal

CHAPTER 1

The Control Room Moves Into the Model

On a Thursday afternoon at Lantern Street Sound, a composite two-room studio that will appear throughout this book, three jobs arrive within an hour. A singer-songwriter sends a phone recording of a song she wrote in 2014 and asks whether the vocal can be lifted out and used on a new release, because she can no longer sing it the way she did at twenty-two. A documentary producer sends an interview recorded beside a busy road and asks whether the traffic can be removed without the subject sounding like a robot. A small game studio asks for three hundred variations of footsteps on gravel, wood and metal, and wants to know whether "the AI" can make them by Monday.

Ten years ago each of those requests would have produced a short, honest answer. The phone vocal could not be separated from the guitar under it; at best you could filter and hope. The roadside interview could be improved a little with spectral repair, at the cost of hours and some damage to the voice. The footsteps would be recorded by a Foley artist and edited by hand, which was expensive but predictable. Today every one of those jobs has an AI-assisted path, and each path is better than it was. Each one also carries a new set of questions. Is a separated vocal good enough for a commercial release, or will a listener on headphones hear the ghost of the guitar? Has the noise-reduction model invented consonants that the speaker never said? Were the footsteps generated by a model trained on sound libraries whose owners never agreed to that use?

This chapter sets out the ground the rest of the book stands on. It describes what has changed, what has not, and a simple set of questions you can ask of any AI processor before you let it near a client's audio.

From groove to number to model

Recorded sound has passed through three broad eras, and each one changed what a sound engineer did all day.

In the first, the medium was physical and the signal was an analogue of the air pressure itself. Edison's tinfoil phonograph cut a groove whose depth followed the vibrations of a diaphragm. Later came wax cylinders, shellac and vinyl discs, optical film soundtracks and magnetic tape. The engineer's craft was mechanical and electrical: placing microphones, setting levels so the signal stayed above the noise of the medium and below the point where it distorted, cutting and splicing tape, and aligning machines. Every copy lost something, so the work was organised around keeping generations to a minimum.

In the second era the signal became a sequence of numbers. Digital recording, the compact disc in the early 1980s and then the digital audio workstation, or DAW, turned audio into data that could be copied without loss, edited without a razor blade and processed by software. The engineer's tools became plug-ins that modelled consoles, compressors and reverberation chambers, and the main limits became the converters, the room and the engineer's ears. Most of the knowledge from the first era survived. Gain staging still mattered, rooms still lied and a microphone in the wrong place still sounded wrong.

The third era, which this book is about, adds a new kind of processor. A neural network is trained on large amounts of audio and learns patterns that no engineer wrote down explicitly: what a voice sounds like apart from the room around it, how a bass guitar differs from a kick drum in the same frequency range, what a snare sounds like when it is hit near the rim, or how a spoken sentence should rise and fall. Once trained, the network can be used as a processor that does things no equaliser or compressor can do. It can remove a sound that overlaps another in both time and frequency. It can produce a sound that was never recorded. It can describe what it hears in words.

The important point is that the third era sits on top of the second, as the second sat on top of the first. A neural vocal isolator still receives a stream of samples at a sample rate and bit depth, still outputs a stream

that can clip, and still depends on what happened at the microphone. A model that generates music still produces a file that has to meet a loudness target and survive a lossy codec. You cannot judge the new processors without understanding the old signal chain, which is why the first part of this book is about rooms and signals rather than models.

What AI changes

It helps to be specific about where AI is genuinely new and where it is a faster or cheaper version of something that already existed.

The first genuinely new capability is separation and isolation. Classical tools could split a signal by frequency, by level or by stereo position. They could not reliably split two sources that occupy the same frequencies at the same time in the same place, such as a voice over a guitar or speech over traffic. Learned models can, often well enough for professional use and sometimes not. Chapter 8 covers dialogue and noise, and Chapter 9 covers music stems.

The second is synthesis that sounds recorded. Synthesisers have made sound from nothing for decades, but they made it from oscillators, samples and physical models that a human designed. Generative models make speech, singing, music and effects that resemble recordings of real performances, from a description or an example. Chapters 10 and 11 deal with voices and with music and sound effects.

The third is description. A machine-listening model can tag a recording as containing applause, a dog or a siren; transcribe speech; estimate tempo and key; or answer a question about what it heard. This turns large archives of audio into something you can search and check automatically. Chapter 17 covers it.

The fourth is assistance with judgement. Mixing and mastering assistants listen to a track, compare it with targets or references, and propose settings. They have been on the market for years in various forms. What changed recently is that more of them can work on separated parts of a finished mix and explain their suggestions. Chapter 12 looks at how far to trust them.

Everything else is mostly acceleration. Automatic transcription makes dialogue editing faster. Generative tools make placeholder music and sound effects cheaper. Agents that can operate audio software through an interface, a development that reached game-audio middleware in 2026, as Chapter 15 describes, make repetitive project work faster. Speed matters, but it does not change what good sound is.

What AI does not change

A few things stay put however good the models get, and a professional pipeline is built around them.

Physics does not change. Sound in a small room still builds up at certain frequencies and cancels at others. Speakers still have a frequency response. A signal that exceeds full scale on a digital converter still clips. A model trained on clean studio speech cannot recover information that was never captured, though it can make a convincing guess, and a convincing guess is not the same thing as the original.

Hearing does not change. Human sensitivity to frequency varies with level, louder sounds mask quieter ones and listeners judge "better" partly by "louder". These facts shape every loudness standard and every mixing decision. They also explain why a demonstration of an AI tool played loudly over good speakers in a quiet room may sound more impressive than the same output will on a phone in a train.

Responsibility does not change. When a client's audio goes out with your name on it, you are answerable for what is in it. If a noise-reduction model invented a word, if a separation left an artefact in the second chorus or if a generated sound effect turns out to contain a recognisable fragment of someone else's recording, "the model did it" is not a defence a client will accept, and it may not be one a court accepts either.

Consent matters more, not less. A recording of a voice was always a recording of a person, but until recently you could not make that person say something new. Now you can. The ethics and, increasingly, the law of voice and likeness have moved to the centre of sound work, and Chapter 20 treats them in detail.

The four questions

Throughout this book every AI processor is treated as a processor: something you put in the signal chain, whose behaviour you need to understand before you trust it. Four questions are enough to start.

What does it change? Every processor has an effect on the signal, intended and unintended. An equaliser changes frequency balance and, depending on its design, phase. A compressor changes dynamics and, if pushed, adds distortion. A neural denoiser changes the noise floor, but it may also change the timbre of the voice, smear transients, remove breaths and room tone, or alter the way consonants sound. Find out what is changed beyond the thing you asked for. The simplest way is to null-test where you can: invert the processed signal, sum it with the original and listen to what is left.

How does it fail? Classical processors fail gradually and audibly. Push an equaliser too far and it sounds harsh; push a limiter too far and it pumps and distorts. Neural processors often fail differently. They may work beautifully on material like their training data and badly on anything unusual: a whispered line, a non-English language, an instrument they rarely saw, a recording with heavy reverberation. They can fail plausibly, producing output that sounds clean and confident but is wrong. The usual sign of plausible failure in speech is a word that sounds slightly different from what was said, or a breath that has become a phoneme. In music it is a part that wanders into another instrument's timbre for a bar. You only find these failures by listening on more than one system and comparing with the source.

What does it cost? Cost includes money, time, compute and latency. Some models run in real time on a laptop and can be used while tracking. Others need a powerful graphics processor or a cloud service and take longer than real time. Some cloud services charge by the minute or by the character and keep copies of what you upload. For live work, latency of more than a few milliseconds becomes audible to a performer, which rules out many models. Chapter 21 shows how to put these costs into a pipeline budget.

What travels with it? Every output now carries a history. Who recorded the source and what did they agree to? What was the model trained on, and does its licence allow commercial use of the output? Does the output contain a watermark, and does your client need to disclose that AI was used? Is a voice in the output identifiable as a real person, and do you have that person's written permission? Some of these questions have legal answers that depend on where you and your audience are; Chapter 20 maps them as of late 2026. Others are matters of professional practice. Either way, the answers should be recorded alongside the audio, not reconstructed from memory when someone asks.

Lantern Street's three jobs

Return to the three jobs that arrived on that Thursday and run each through the four questions.

The old phone vocal is a separation job. A modern separation model will probably lift the voice cleanly enough to judge the performance, and may be good enough for a lo-fi release. What it changes, beyond removing the guitar, is likely to include some loss of high-frequency air and occasional bleed where guitar and voice overlap closely. It may fail in quiet passages, where it can leave a watery residue. It costs little to try. What travels with it is easier than it might seem: the singer owns the recording and her own voice, so the rights question is simply whether she is happy with the result. The engineer's job is to tell her honestly what can and cannot be fixed.

The roadside interview is a repair job. A dialogue isolator will remove most of the traffic. The risk is that it changes the speaker's voice and, at the worst moments, when a truck passes, invents or swallows syllables. For a documentary that risk is serious: the words are evidence of what someone said. The pipeline therefore keeps the original, processes conservatively, checks every flagged passage against the raw recording and, where the words are unclear, chooses honest noise over confident fiction. Chapter 8 describes the technique.

The footsteps are a generation job, and the most complicated. A text-to-sound model could produce three hundred variations quickly. Whether they are good enough depends on the game; whether they can be used

depends on the model's licence and training data, and on the studio's policy on generated assets. In practice Lantern Street might record a small set of real footsteps, use procedural variation in the game engine for most of the variety, and use generation only where it adds something that recording cannot. Chapter 15 explains why that hybrid is common.

None of these answers is "use AI" or "don't use AI". Each one is a decision about where in the chain a model belongs, what it should be trusted to do and how its output will be checked.

The shape of a professional AI pipeline

The later parts of this book develop the idea that an AI-assisted audio pipeline is a sequence of rooms, each with a job, an input, an output and a check. A typical post-production pipeline might move from capture, to ingest and logging, to transcription, to edit, to repair, to separation where needed, to mix, to loudness and delivery checks, to archive. At each stage some processing may be done by a model and some by a person, and at each stage the output is compared with something: the original recording, a reference mix, a specification or a listener's judgement.

Three habits keep such a pipeline honest. Keep the originals untouched, and make every AI process a non-destructive step that can be undone or redone with a different model. Log what was done, with which model and version and settings, so that a result can be reproduced or explained. And listen at every stage on more than one playback system, because the failure modes of neural processors are often invisible on a meter and inaudible on a single pair of speakers.

How to use this book

Each chapter ends with practical guidance rather than a summary. Some chapters include listening exercises; do them, because the skill this book is trying to build is mostly in your ears. Where a chapter names a product or model, treat the name as an example of a category with a date attached, not as a recommendation that will stay true. The standards and principles are more durable than any product. A beginner who understands gain staging, masking and loudness will be able to judge a

tool released in 2029. A professional who has learned to ask the four questions will be able to bring it into a pipeline without being surprised by it.

In practice

- Before adopting any AI processor, run the four questions: what it changes, how it fails, what it costs and what travels with its output. Write the answers down.
- Keep every original recording untouched. Treat every AI step as a reversible, logged process.
- Null-test processors where possible, and listen to what they removed as well as what they left.
- Check AI output on at least two playback systems, one of them small and cheap, before signing it off.
- For any job involving a real person's voice, settle consent in writing before you start, not after the client likes the result.

CHAPTER 2

Sound, Hearing and Numbers

Before a single model enters the picture, a sound engineer needs a working sense of three things: what sound is physically, how people hear it and how a computer stores it. This chapter covers each in turn, with as little mathematics as the job allows. If you have trained as an engineer you can skim it, but read the section on hearing, because almost every judgement about AI-processed audio comes back to how ears and brains behave.

Sound as pressure

Sound is a travelling disturbance in air pressure. A vibrating object, such as a guitar string, a vocal fold or a loudspeaker cone, pushes and pulls on the air next to it, and the resulting compressions and rarefactions move outward at the speed of sound, roughly 343 metres per second in air at room temperature. The pressure changes involved are tiny compared with atmospheric pressure, but the ear is extraordinarily sensitive to them.

Two properties of the disturbance matter most. Its frequency is how many times per second the pressure cycles, measured in hertz. Its amplitude is how large the pressure swings are. A pure tone has a single frequency, but almost every real sound is a mixture of many frequencies whose amplitudes change over time. The shape of that mixture is what makes a violin sound like a violin and a vowel sound like an “ee” rather than an “oo”.

Frequency and wavelength are linked by the speed of sound. A 100 Hz tone has a wavelength of about 3.4 metres; a 1 kHz tone, about 34 centimetres; a 10 kHz tone, about 3.4 centimetres. That relationship explains a great deal of practical acoustics. Low frequencies have wavelengths comparable to the dimensions of a room, so they interact

with the room as a whole and are hard to absorb. High frequencies have short wavelengths, so they are easily blocked by a head, a music stand or a badly placed foam panel, and they beam strongly from loudspeakers and instruments. It also explains why moving a microphone a few centimetres can change its tone: you are moving it relative to interference patterns whose spacing depends on frequency.

Sound spreads as it travels. From a small source in open space, the level falls by about 6 decibels for each doubling of distance, a relationship known as the inverse square law. Indoors, reflections from walls, floor and ceiling add to the direct sound, and beyond a certain distance the reflected sound dominates. That boundary, sometimes called the critical distance, is where a recording starts to sound roomy rather than close.

Decibels

Engineers measure level in decibels because hearing responds to ratios rather than differences. A decibel is a logarithmic unit: it expresses one quantity relative to another. That means a decibel value always has a reference, even when the reference is not stated.

Sound pressure level, written dB SPL, uses a reference of 20 micropascals, close to the quietest sound a young healthy ear can detect at around 1 kHz. Normal conversation at a metre is roughly 60 dB SPL; a loud rock concert can exceed 100. Inside a digital system the reference is full scale, the largest value the system can represent, and levels are written in dBFS. Zero dBFS is the ceiling, so all practical digital levels are negative. Loudness meters use LUFS or LKFS, which are decibel-like units relative to full scale with frequency weighting and time averaging, described in Chapter 6.

A few rules of thumb make decibels easier to live with. Doubling the amplitude of a signal adds about 6 dB. Doubling its power adds about 3 dB. A change of around 1 dB is close to the smallest difference most listeners can reliably detect in a direct comparison, and listeners tend to judge a sound roughly twice as loud when it rises by about 10 dB. Two uncorrelated sources of equal level, such as two different noise signals, sum about 3 dB louder than either alone; two identical signals in phase sum 6 dB louder; two identical signals with one inverted cancel

completely. That last fact is the basis of the null test mentioned in Chapter 1.

How people hear

The ear converts pressure changes into nerve signals in three stages. The outer ear collects sound and, through the shape of the pinna and the ear canal, colours it in ways that depend on direction. The middle ear transfers the vibration from air to fluid. The inner ear, in the cochlea, separates the sound by frequency along a membrane whose different regions respond to different frequencies, and hair cells turn the motion into neural signals. The brain then does an enormous amount of work: grouping components into objects, locating them in space, recognising them and deciding which to pay attention to.

Several consequences matter every day in a studio.

Sensitivity depends on frequency and level. The ear is most sensitive between about 2 and 5 kHz, a region that matters for the intelligibility of speech, and much less sensitive at very low and very high frequencies, especially at low listening levels. The curves that show which levels at different frequencies sound equally loud are called equal-loudness contours; the work of Fletcher and Munson in the 1930s established them and they are now standardised in ISO 226. The practical lesson is that the balance of a mix changes with playback level. A mix that sounds full at high volume can sound thin at low volume because the bass seems to disappear, and a mix balanced only at low volume can sound boomy when turned up. This is also why louder sounds "better" in a quick comparison, a fact that skews almost every untested before-and-after demonstration, including demonstrations of AI tools. Always level-match before comparing.

Loud sounds hide quiet ones. Masking is the reduction in audibility of one sound caused by the presence of another. It is strongest when the two sounds are close in frequency, and it spreads more upward in frequency than downward, so a loud low sound can mask quieter sounds above it. Masking also works in time: a loud sound can hide a quieter one that happens just after it and, to a smaller degree, just before it. Masking is the main reason mixes become muddy, and the main reason

lossy codecs such as MP3 and AAC work at all: they discard what masking would hide. It is also central to how listeners perceive AI artefacts. A faint residue of a removed instrument may be masked in a dense chorus and painfully obvious in a sparse verse.

Hearing groups frequencies into bands. The cochlea behaves roughly like a bank of overlapping filters, called critical bands, that are narrow at low frequencies and wider at high frequencies. Many perceptual scales used in audio and in machine learning, such as the mel scale and the Bark scale, approximate this. When you see a "mel spectrogram" in Chapter 7, it is a picture of sound warped to match this property of the ear.

Direction comes from differences between the ears. For low frequencies the brain uses the difference in arrival time between the two ears; for higher frequencies it uses differences in level caused by the shadow of the head; and for elevation and front-back distinctions it relies on the filtering effect of the pinna, which differs from person to person. Together these cues are summarised in a head-related transfer function, or HRTF, a concept central to binaural and spatial audio in Chapter 18.

The first arrival wins. When the same sound arrives from two directions within a short interval, typically a few milliseconds up to a few tens of milliseconds, listeners hear a single sound coming from the direction of the first arrival. This is the precedence effect. It is why a short delay can widen a sound without making it seem to come from two places, and why reflections from a desk or a nearby wall can shift the stereo image of a pair of monitors.

Ears tire. After long exposure to loud sound, hearing becomes temporarily less sensitive and judgements of brightness and balance drift. Long sessions at high levels cause permanent damage. The practical rule for engineers is to monitor at moderate levels, take regular breaks and make critical decisions early in a session, not at the end.

From air to numbers

A microphone turns pressure changes into a varying voltage. To store that voltage on a computer, an analogue-to-digital converter measures it

at regular intervals and records each measurement as a number. Two settings govern the process: how often the measurement is taken, which is the sample rate, and how precisely each measurement is recorded, which is the bit depth.

Sample rate

The sampling theorem associated with Nyquist and Shannon says that a signal can be captured and reconstructed exactly from its samples if it contains no frequencies at or above half the sample rate. That half is called the Nyquist frequency. Compact disc audio uses 44.1 kHz, which puts the Nyquist frequency at 22.05 kHz, just above the upper limit of human hearing. Video and broadcast work standardised on 48 kHz. Higher rates such as 96 kHz are used in some production work, mainly to give filters and nonlinear processors more room above the audible band.

Any content above the Nyquist frequency that reaches the converter is not simply lost: it folds back into the audible range as aliasing, a form of distortion that is not harmonically related to the original sound. Converters prevent this with steep low-pass filters before sampling. Inside software, processors that create new harmonics, such as saturators, distortion and some neural models, can generate aliasing too, which is why many plug-ins oversample internally.

Sample rate matters for AI work in a specific way. Many speech models are trained and run at 16 kHz or 24 kHz, which caps their output at 8 or 12 kHz of bandwidth. Music models often run at 32, 44.1 or 48 kHz. If you pass 48 kHz dialogue through a model that works at 16 kHz, the result will have lost its top end, even if the file you get back is labelled 48 kHz. Always check what rate a model actually runs at, not only what rate it accepts.

Bit depth

Each sample is stored as a number of fixed precision. The difference between the true value and the stored value is quantisation error, which behaves like a low-level noise. Each additional bit adds about 6 dB of dynamic range. Sixteen-bit audio, the CD standard, offers a theoretical range of about 96 dB; twenty-four-bit audio offers about 144 dB in theory,

far beyond what any real converter achieves, which is why 24-bit recording gives you generous headroom with no practical penalty.

Many DAWs and plug-ins now process internally in 32-bit or 64-bit floating point. Floating-point numbers can represent values far above 0 dBFS without clipping, so an internal bus that goes "into the red" in a floating-point mixer can often be recovered simply by turning it down later. That safety ends the moment the signal reaches a fixed-point stage: a converter, a 16-bit or 24-bit file, or a plug-in that clips internally. Treat floating-point headroom as a safety net, not a working method.

Dither

When you reduce bit depth, for example from a 24-bit master to a 16-bit file, the quantisation error becomes correlated with the signal and can sound like distortion on quiet material such as fades and reverb tails. Dither is a small amount of noise added before the reduction that decorrelates the error and turns it into a benign, steady hiss far below the music. Some dither types also shape the noise spectrum to push it into frequencies where the ear is less sensitive. The rule is simple: dither once, at the final reduction to a lower bit depth, and not at every intermediate stage.

Lossy and lossless files

A WAV or AIFF file stores samples directly. Lossless compression formats such as FLAC and ALAC reduce file size without changing a single sample. Lossy codecs such as MP3, AAC and Opus reduce size much further by removing information that a perceptual model predicts will be masked or inaudible. Lossy encoding can cause a signal that peaked just below 0 dBFS to exceed it after decoding, which is one reason delivery specifications set true-peak limits below zero, as Chapter 6 explains.

A newer category, neural audio codecs, compresses audio into a small number of learned tokens and reconstructs it with a neural decoder. These are discussed in Chapter 7 because they now sit at the heart of many generative models.

What the numbers do not tell you

It is tempting to treat everything in this chapter as a set of specifications to meet and forget. The numbers are necessary but not sufficient. A file can have the right sample rate and bit depth, peak safely below full scale and still sound bad because the microphone was in the wrong place, the room was boomy or the performance was flat. Equally, an AI-processed file can measure cleaner than the original on every meter and sound worse, because the noise reduction has flattened the voice or the separation has hollowed out the midrange. Meters describe signals; only listening describes sound.

Listening exercises

1. Play a well-mixed song at a comfortable level, then at a very low level, without changing anything else. Note what seems to disappear first. That is the equal-loudness effect at work.
2. Take two copies of the same file. Turn one up by 1 dB and compare them blind with a friend switching. Then try 0.5 dB. Find the level at which you can no longer reliably tell them apart; it tells you how carefully you must level-match comparisons.
3. Take a mix and a processed version of it, invert the polarity of one and sum them. Listen to what remains. Do this later with an AI-denoised file and the original; the residue is everything the model removed, including anything it should not have.
4. Export a quiet piano fade at 16 bits with and without dither and listen at high gain (carefully, then turn back down). The undithered version will sound grittier as it fades.

In practice

- Know your references: dB SPL in the air, dBFS in the computer, LUFS on the loudness meter.
- Record at 24-bit, at 48 kHz for anything involving picture and at 44.1 or 48 kHz for music, unless a client specifies otherwise.
- Check the internal sample rate of any AI model before you use it on full-bandwidth material.

- Level-match every comparison, including every comparison between AI-processed and unprocessed audio.
- Dither once, at the end, when reducing bit depth.

CHAPTER 3

Signal Flow and Gain Staging

Signal flow is the route a sound takes from its source to its destination, and gain staging is the practice of keeping its level sensible at every point along that route. Neither topic is glamorous. Both decide whether the rest of your work is possible. A noisy, clipped or badly routed recording limits every later process, and AI processes are no exception: a dialogue isolator fed a clipped recording will isolate clipped dialogue, and a separation model fed a mix that was squashed by a limiter will struggle to find the drums.

This chapter follows the signal from the microphone into the computer and through a mixing session, and ends with the habits that keep files usable for both people and models.

Microphones

A microphone converts pressure changes into voltage. The three main families behave differently, and choosing between them is the first and most important tonal decision you make.

Dynamic microphones use a diaphragm attached to a coil moving in a magnetic field. They are robust, handle very loud sources and are relatively insensitive to distant sound, which makes them forgiving in untreated rooms. Many podcasters and live vocalists use them for exactly that reason.

Condenser microphones use a lightweight diaphragm as one plate of a capacitor. They are more sensitive and capture more detail and high-frequency air, but they also capture more of the room. Most need power, usually supplied by the interface as 48-volt phantom power.

Ribbon microphones use a thin metal ribbon suspended in a magnetic field. They tend to sound smooth and dark and usually have a figure-of-

eight pickup pattern. Some older ribbons can be damaged by phantom power or by blasts of air, so check before you plug one in.

Every microphone also has a polar pattern, which describes how sensitivity varies with direction. Cardioid microphones pick up mainly from the front and reject sound from the rear. Omnidirectional microphones pick up equally in all directions and have no proximity effect. Figure-of-eight microphones pick up front and back and reject the sides. Directional microphones exhibit proximity effect: as the source comes closer, bass increases. Singers use it deliberately; podcasters fight it.

Placement matters more than model. A cheap microphone a hand's width from a singer in a quiet, dead corner will usually beat an expensive one two metres away in a reflective room. For speech, a distance of roughly 10 to 20 centimetres, slightly off-axis to avoid plosives, with a pop filter, is a reliable starting point. For acoustic instruments, move the microphone while listening on headphones until the sound is right; aim for the part of the instrument that sounds best rather than the part that looks most obvious.

From microphone to converter

A microphone produces a very small voltage, called mic level. A preamplifier raises it to line level, the working level of most audio equipment. Professional line level is nominally +4 dBu; consumer equipment often uses −10 dBV. Electric guitars and basses produce a high-impedance instrument level and should go into an instrument input or a direct box, which converts the signal for a microphone input and lets you run long cables without loss of tone.

The preamplifier's gain control is the most important level control in the chain, because it sets the relationship between the wanted signal and the noise of everything after it. Too little gain and the converter's noise and the noise of later digital gain will be audible. Too much and the preamp or converter clips, which no later process can truly undo.

In a modern home studio the preamp and converter usually live together in an audio interface connected to the computer by USB or Thunderbolt. Interfaces vary in noise, headroom and converter quality, but even

inexpensive current models are good enough to make professional-sounding recordings. The bigger differences are in the room, the microphone and the performance.

Setting record levels

At 24 bits there is no reason to record hot. A reasonable target is for typical peaks to sit somewhere between −18 and −10 dBFS, with the loudest moments of the performance safely below −6 dBFS. Average level will often sit around −20 to −18 dBFS. Many engineers align their systems so that an average level of about −18 or −20 dBFS corresponds to the nominal operating level of analogue gear, which is why you will often see −18 dBFS mentioned as a reference. Treat it as a convention, not a law.

The reason for leaving this headroom is that performers are unpredictable. A singer who is quiet in the soundcheck may be loud in the take that matters. A podcast guest may laugh. A drummer always plays harder when the red light is on. Recording at moderate levels with 24-bit resolution costs nothing in noise and saves you from the one problem that nothing in post-production can repair cleanly.

AI repair tools have made that last statement slightly less absolute. Declipping models, covered in Chapter 8, can reconstruct clipped peaks surprisingly well. But a reconstruction is an estimate, not the original performance, and on a clipped vocal it will often sound slightly different from what was sung. Prevention remains cheaper and more honest than cure.

Latency and monitoring

Every digital system introduces delay. The converter takes time, the computer processes audio in blocks called buffers, and each block has to be filled before it can be processed and emptied before it can be heard. At a sample rate of 48 kHz, a buffer of 128 samples represents about 2.7 milliseconds, and the total round trip from microphone to headphones is typically several times that once input, output and converter delays are added.

Performers start to notice latency in their own monitoring at a few milliseconds, and many find more than about 10 milliseconds distracting, particularly singers and drummers. There are three common answers. Lower the buffer size while recording and raise it while mixing. Use the interface's direct monitoring, which sends the input straight to the headphones before it reaches the computer. Or avoid heavy plug-ins on record-enabled tracks.

Latency is the single biggest constraint on using AI processors live. Many neural models need to see a window of audio, sometimes tens or hundreds of milliseconds, before they can make a decision, and they may need a graphics processor to run fast enough. Chapter 16 explains how real-time neural effects get around this and what latency figures they reach. For tracking, the rule is simple: if a model adds audible delay, keep it off the performer's monitor path and apply it afterwards.

Clocking

Digital devices must agree on exactly when each sample is taken. A single interface uses its own internal clock. When several digital devices are connected, one must be the master and the others must follow it, either through the digital audio connection or through a dedicated word-clock cable. Mismatched clocks cause clicks, dropouts or slow drift. In a home studio with one interface you rarely need to think about this. In a larger studio with several converters and digital consoles, clocking problems are one of the most common causes of mysterious noises.

Inside the DAW

Once audio is in the computer, the DAW becomes a virtual console. Its routing follows the same logic as an analogue desk, and learning that logic will make every DAW easier to use.

A track holds audio or MIDI. Each track has inserts, which are processors placed directly in its signal path, such as an equaliser or compressor. It has sends, which copy some of its signal to another destination, typically a bus carrying a reverb or delay; sends can be taken before the fader, so their level stays fixed, or after it, so they follow

the fader. Each track has a fader and a pan control, and its output goes to a bus or directly to the master.

A bus combines several signals. Common buses include a drum bus, a vocal bus and effects returns. Buses let you process groups together and keep a session organised. The master bus is the final mix.

The order of processing matters. An equaliser before a compressor changes what the compressor reacts to; after it, it shapes the compressed result. A reverb on a send is shared and keeps its tail independent of the dry signal's processing; a reverb on an insert affects only that track. When you add AI processors to a session, put them where their job belongs in this chain. A dialogue isolator belongs early, on the source track, before compression and equalisation that would make its task harder. A mastering assistant belongs on the master bus, after the mix is finished.

Gain staging through the mix

Gain staging inside the DAW is about keeping each stage at a level where processors behave as designed. Floating-point mixers will not clip internally, but many plug-ins, particularly those modelled on analogue equipment, respond differently depending on input level. A compressor model calibrated around −18 dBFS will behave very differently if you feed it peaks at −2.

A practical approach is to keep track levels in the same broad range as your recording targets, adjust clip gain or a trim plug-in at the start of each chain to bring sources into that range, and leave faders near their nominal position so you have room to move them. Keep the master bus peaking well below 0 dBFS before any final limiting, often around −6 dBFS, so that the mastering stage has room to work.

AI tools often have their own expectations. Some separation and enhancement models normalise their input internally; others do not, and produce better results when the input is at a sensible level. Some cloud services accept only certain formats or downsample silently. When you use any model, check its output level and format against its input, and adjust gain before the next stage.

File and session hygiene

Good sound engineers are tidy, not because tidiness is a virtue but because untidy sessions lose work. The habits that matter most are simple.

Name tracks and files clearly and consistently, with the project, the source and the take, before you record, not after. "LS_SongTitle_LeadVox_T03.wav" is better than "Audio 1_07.wav", and an assistant, a client or a future you will be able to find it.

Keep a session sheet. Note the date, the people present, the microphones and positions, the interface settings, the sample rate and bit depth and any problems. Appendix A includes a template.

Keep the raw recordings separate and untouched. Every edit, process and AI treatment should happen on copies or in non-destructive form in the session. When you render a processed file, give it a name that shows what was done, for example "LeadVox_T03_dialogueIsolate_v1.wav", and keep a record of the model, version and settings used.

Export stems with a common start point. When you send material to another engineer, a mastering house or a separation model, export every stem from the same start time and at the same length so that they line up exactly when imported. Many problems in collaborative work come from stems that drift by a few milliseconds.

Back up in at least two places, one of them away from the studio. Hard drives fail, laptops are stolen and cloud services change their terms.

Cables, grounding and noise

Many of the noises that people later try to remove with AI begin as wiring problems. Balanced connections, used by professional microphones and most line-level equipment, carry the signal on two conductors in opposite polarity, so that interference picked up along the cable cancels at the receiving end. Unbalanced connections, used by guitars and much consumer equipment, have no such protection and should be kept short. Hum at the mains frequency and its harmonics often comes from ground loops, where equipment connected to different power outlets creates a path for current through the audio cables.

Powering the whole audio system from one outlet, using balanced connections and, where necessary, isolation transformers or ground-lift switches on direct boxes usually cures it. Never remove the safety earth from a mains plug to fix hum.

Computers add their own noise. USB-powered interfaces can pick up whine from laptop chargers and graphics activity; trying a different port, a powered hub or running on battery during a quiet take is quicker than any repair afterwards. Phones near microphones and cables produce bursts of interference when they communicate with the network; put them in flight mode or out of the room. Lighting dimmers and some LED power supplies buzz into nearby cables. Five minutes spent listening to the noise floor with nothing playing, headphones turned up carefully, before a session starts will catch most of these.

A Lantern Street tracking session

At Lantern Street, the head engineer sets up a vocal session for the singer-songwriter from Chapter 1, who has decided to re-record her old song rather than rely on the separated phone vocal. The vocal booth is small and well treated. A large-diaphragm condenser in cardioid sits about 15 centimetres from her mouth, behind a pop filter and angled slightly down. Phantom power is on. The interface runs at 48 kHz and 24 bits with a 64-sample buffer for tracking, and her headphone mix comes through the interface's direct monitoring with a little reverb from the DAW on a send.

The engineer asks her to sing the loudest line of the chorus twice, sets the preamp so that the peaks sit around −10 dBFS, and leaves it there. Takes are named as they are recorded. Between takes the engineer listens back on headphones and on the main monitors and makes notes on the session sheet: which lines are strongest, where she was off-mic, where a breath is noisy. Nothing is processed destructively. After the session the raw takes go to two backup drives and to the studio's off-site storage before anyone starts editing.

That session contains no AI at all. It is the foundation on which every later process, human or machine, will depend.

Where AI services sit in the chain

Many AI tools are not plug-ins at all. They are standalone applications or cloud services that receive files and return files. Each round trip outside the DAW is a point where signal flow can quietly break. Files may be resampled, converted to a different bit depth, changed from stereo to mono or trimmed. Start times may shift, so that a processed file no longer lines up with the original. Some services add a short silence or fade at the start and end. Metadata such as timecode may be stripped.

Treat every round trip as an interface between two systems and check it. Export with a known start point and handles of silence or room tone at each end. When the file comes back, check its sample rate, bit depth, channel count, length and alignment against the original, ideally by null-testing an unprocessed region. Keep a note in the session of which file went where and what came back. These checks take a minute and prevent the slow, confusing problems that come from files that are almost, but not quite, what you sent.

In practice

- Choose microphone and position before reaching for any processor, AI or otherwise.
- Record at 24-bit with typical peaks around −18 to −10 dBFS and the loudest moments below −6 dBFS.
- Keep latency-heavy processors, including most neural models, off performers' monitor paths.
- Place AI processors where their job belongs in the chain: repair early, mastering last.
- Name, log and back up every take. Keep originals untouched and record every model, version and setting used on a copy.

CHAPTER 4

Rooms, Monitoring and Acoustics

Every recording is a recording of a room, and every mix is made through one. The room you record in adds reflections and resonances to the sound that reaches the microphone. The room you mix in adds its own reflections and resonances to the sound that reaches your ears, and you then adjust the mix to compensate for problems that exist only in your room. The mix that results sounds right where you made it and wrong everywhere else. Learning to recognise, reduce and work around room effects is one of the most valuable skills in audio, and it is one that AI has not made obsolete. A model can remove some reverberation from a recording, but it cannot tell you that your mixing room has a hole at 90 Hz.

What a room does to sound

When a sound is made in a room, the listener first hears the direct sound, then a series of early reflections from nearby surfaces within the first few tens of milliseconds, then a dense, decaying wash of later reflections called reverberation. Each part affects perception differently. The direct sound carries most of the information about what the source is and where it is. Early reflections colour the tone and can blur the stereo image. Reverberation gives a sense of space and, if there is too much of it, makes speech hard to follow and music muddy.

Reverberation time

The standard measure of a room's reverberation is RT60, the time it takes for sound to decay by 60 dB after the source stops. Wallace Clement Sabine, working at Harvard around 1900, showed that reverberation time depends on the room's volume and the total absorption of its surfaces. His formula, still used for first estimates, says that RT60 is roughly

proportional to volume divided by absorption: big rooms with hard surfaces ring for a long time; small rooms with soft surfaces do not.

Different uses want different reverberation. Concert halls for orchestral music often have reverberation times around two seconds. Control rooms and voice-over booths are much drier, typically well under half a second. A typical untreated bedroom is not especially reverberant, but its reverberation is uneven: it may decay quickly at high frequencies, where curtains and furniture absorb, and slowly at low frequencies, where almost nothing absorbs.

Room modes

At low frequencies a small room behaves less like a space full of random reflections and more like a resonator. Sound reflecting between two parallel walls reinforces itself at frequencies whose half-wavelength fits the distance between them, and at multiples of those frequencies. These are room modes. A room 4 metres long, for example, has its first lengthwise mode at about 43 Hz, with further modes at about 86, 129 Hz and so on. Width and height produce their own series, and combinations of all three produce more.

Modes cause the bass response of a room to vary dramatically from place to place. At some positions a mode produces a peak, so a bass note at that frequency booms; at others it produces a null, so the same note almost vanishes. Peaks and nulls of 10 to 20 dB at low frequencies are common in untreated small rooms. If your listening position sits in a null at 80 Hz, you will add bass that is not needed and your mix will be boomy everywhere else.

Above a certain frequency, often called the Schroeder frequency, modes become so dense that the room behaves statistically, and reflections and absorption matter more than individual resonances. In typical small rooms this transition happens somewhere in the low hundreds of hertz.

Treating a room

Acoustic treatment uses absorption, diffusion and placement to make a room more neutral. It is not soundproofing, which stops sound from

getting in or out and requires mass and isolation. Treatment changes how sound behaves inside the room.

Absorb the first reflections. The surfaces between you and your speakers, typically the side walls, the ceiling above the desk and sometimes the desk itself, send early reflections to your ears that blur the image and colour the sound. A simple way to find these points is the mirror method: sit at the listening position while someone slides a mirror along the side wall, and mark every spot where you can see a speaker in the mirror. Panels of dense mineral wool or similar material, a few centimetres to ten centimetres thick and covered with fabric, at those points make a clear improvement.

Trap the bass. Thin foam panels absorb high frequencies but do almost nothing below a few hundred hertz. Controlling modes needs thick, porous absorbers placed where low-frequency pressure is high, usually in corners, including wall-ceiling corners. Purpose-built bass traps, or thick panels spaced from the wall, can smooth the low end substantially. They will not eliminate deep nulls entirely; nothing short of moving the listening position or the speakers will.

Diffuse the back. In larger rooms, scattering reflections from the rear wall with diffusers rather than absorbing them all keeps a sense of space without strong discrete echoes. In very small rooms, absorption is usually more useful.

Avoid over-treatment. A room covered entirely in thin foam becomes dead at high frequencies and still boomy at low ones, which is often worse than an untreated room. Aim for balance across the frequency range.

Placing monitors and the listener

Monitor placement has as much effect as treatment and costs nothing.

Set the speakers and your head at the corners of an equilateral triangle, with the tweeters pointing at your ears or just behind your head, at ear height. Position the setup symmetrically in the room, so that the left and right speakers see similar boundaries; asymmetry shifts the stereo image. Avoid putting your head exactly in the middle of the room's

length, where several modes have nulls, and avoid placing it right against the back wall, where bass builds up. Many engineers start around a third of the room length from the front wall and adjust by measurement and listening.

Distance from the wall behind the speakers changes their bass response. Close to the wall, bass is reinforced, which some speakers compensate for with a switch. At intermediate distances, reflections from the wall can cancel certain low frequencies, a problem known as speaker boundary interference. Try several positions and measure.

Use stands or isolation pads to decouple speakers from the desk, which otherwise resonates, and keep the desk surface clear between you and the speakers where possible.

Measuring

Measurement turns guesswork into evidence. A measurement microphone, such as an inexpensive omnidirectional calibrated model, and free software such as Room EQ Wizard can show you the frequency response at your listening position, the decay time at different frequencies and the location of the worst peaks and nulls. Measure with the microphone at ear position, then move it a little and measure again: the variation you see at low frequencies is the room's modal behaviour.

Measurement also keeps you honest about improvements. A new bass trap should flatten a peak or shorten a decay; if the measurement does not change, the trap is in the wrong place or not thick enough.

Room correction and its limits

Digital room correction uses measurements to build a correction filter that is applied to the monitor signal. Many products and some speakers now include it, and some describe their analysis as machine learning or AI. Whatever the algorithm, correction can do some things well and some things not at all.

It can reduce peaks. If a mode adds 10 dB at 60 Hz at your listening position, cutting 60 Hz in the monitor path makes the room sound more

even there. It can correct broad tonal tilts caused by speakers and boundaries. It can align timing between speakers and subwoofers.

It cannot fill nulls. A null is caused by cancellation; adding more energy at that frequency mostly adds more cancellation and wastes amplifier power. It cannot fix problems that vary rapidly with position, because a correction measured at one spot can make things worse a few centimetres away. And it cannot remove reverberation or early reflections; it can only change the overall frequency balance.

The sound approach is to treat and position first, then use correction for what remains, with modest cuts rather than large boosts. Treat any correction system's claims, AI-based or not, as hypotheses to be measured.

Headphones

Many beginners mix on headphones because they have no treated room, and many professionals check mixes on headphones because so many listeners use them. Headphones remove the room from the equation, which is a strength and a weakness.

The strength is consistency. A good pair of headphones sounds the same in a bedroom, an office or a hotel, and reveals small details such as clicks, edits and noise that speakers in an untreated room can hide. Headphones are essential for checking AI processing, because artefacts such as watery separation residue or phase smearing are often more audible in them.

The weakness is that headphones present each channel to one ear only, without the natural crosstalk you hear from speakers, so panning and stereo width sound exaggerated and the centre image sits inside your head. Low-frequency response varies with fit and seal. Many headphones also have pronounced tonal colourations. Tools exist that simulate speakers in a room on headphones, or correct a particular model's frequency response from measurements of that model, and they can help. Spatial rendering based on head-related transfer functions, covered in Chapter 18, is the same idea taken further.

If you mix mainly on headphones, check regularly on speakers, even modest ones, and on earbuds and a phone speaker.

Reference listening and translation

A mix "translates" when it sounds balanced on many different systems. No single room or pair of speakers can guarantee that. Professionals improve their odds in three ways.

They learn their room by listening to commercial recordings they know well, in the same genre as their work, at the same level they mix at. If those references sound bass-light in your room, your mixes will tend to be bass-heavy elsewhere.

They check on multiple systems: main monitors, small speakers or a single mono speaker, headphones, earbuds, a phone and often a car. Each exposes different problems. A phone reveals whether the vocal and the key elements survive without bass. A car reveals low-frequency balance and how the mix copes with noise.

They compare at matched loudness. Streaming services normalise loudness, as Chapter 6 explains, so the right comparison is between your mix and a reference both played at the level the listener will actually hear.

AI mastering and mixing assistants increasingly use references as targets, analysing a reference track and suggesting processing to move your mix toward its spectrum or dynamics. That can be useful, but a reference chosen in a room that misleads you will produce a target that misleads you too. Translation is a property of your judgement, and the room is where that judgement is trained.

Recording in imperfect rooms

Most people record outside ideal studios: in bedrooms, offices, on location. The guidance is the same as for mixing, applied to the microphone instead of the ears.

Get close. The closer the microphone is to the source, the larger the direct sound is relative to the room. Use directional microphones and aim their dead side at the noisiest or most reflective part of the room.

Put soft material around and behind the source: a duvet, a sofa or a portable vocal shield. Record in the smallest dead space available, such as a wardrobe full of clothes, rather than an empty room.

AI dereverberation and dialogue isolation tools, covered in Chapter 8, can remove a surprising amount of room sound afterwards. They work best when there is less to remove. A recording that is mostly direct sound with a little room will clean up naturally; a recording that is mostly room will clean up into something that sounds processed. Treat the room first and the model second.

A Lantern Street room check

When Lantern Street took over its second room, a former office about 4.5 metres by 3.5 metres, the engineers measured before buying anything. The response at the listening position showed a peak of roughly 12 dB near 40 Hz and a deep null near 75 Hz, along with an uneven decay below 200 Hz. Moving the listening position 40 centimetres forward reduced the null noticeably. Thick panels in the four vertical corners and at the first-reflection points on the side walls and ceiling reduced the peak and shortened the low-frequency decay. A modest correction filter then trimmed the remaining peak by a few decibels. The room was still not perfect, so the engineers spent a week mixing familiar references in it before using it for client work, and kept a pair of headphones and a small mono speaker on the desk as permanent second opinions.

The numbers in this example are illustrative; your room will have its own. The sequence is what matters: measure, move, treat, then correct, and only then trust your ears.

In practice

- Measure your listening position before buying treatment, and measure again after every change.
- Place speakers and listener in an equilateral triangle, symmetrical in the room, away from the exact centre of the room's length.
- Treat first reflections and corners first; avoid covering the room in thin foam.

- Use room correction for modest cuts, never to fill nulls.
- Check every mix on headphones, a small speaker, earbuds and a phone at matched loudness.
- When recording in imperfect spaces, get close and add absorption around the source before relying on AI cleanup.

CHAPTER 5

The Core Processors

Most of what happens in a mix can be described with a small set of tools: equalisers, which change balance across frequency; dynamics processors, which change balance across time; saturation, which adds harmonics; and time-based effects, which add space. Every AI mixing assistant you will meet in Chapter 12 ultimately proposes settings for these same tools, and many neural effects in Chapter 16 learn to imitate them. Understanding what each one does to a signal, and what it costs, is how you judge whether a machine's suggestion makes sense.

Equalisation

An equaliser boosts or cuts a range of frequencies. The name comes from early telephone and broadcast engineering, where filters were used to make a line's response flat, or equal, again. In music production, equalisers are used both to correct problems and to shape tone creatively.

Filter types

A high-pass filter removes energy below a chosen frequency and lets higher frequencies pass. It is the most used equaliser in mixing because many sources carry low-frequency rumble, handling noise or proximity-effect boom that contributes nothing but mud. A low-pass filter does the opposite. Both are described by their cutoff frequency and their slope, in decibels per octave: a gentle 6 or 12 dB per octave slope sounds natural; steeper slopes are more surgical.

A shelving filter raises or lowers everything above or below a frequency by a fixed amount, like tilting the end of a shelf. Shelves are good for broad changes: adding air to a vocal above 10 kHz, or warmth to a mix below 200 Hz.

A peaking or bell filter raises or lowers a band around a centre frequency. Its width is set by Q: high Q values give narrow bands for surgical work such as removing a resonance; low Q values give broad, musical changes. A parametric equaliser offers several bell bands with adjustable frequency, gain and Q.

Phase and linear-phase equalisers

Ordinary equalisers, like their analogue ancestors, shift the phase of frequencies near where they act. This is usually inaudible and often part of their character, but it matters when the same signal is processed differently on parallel paths, such as multiple microphones on a drum kit, because the phase shifts can cause partial cancellation when the paths are combined. Linear-phase equalisers avoid phase shift at the cost of latency and a characteristic pre-ringing on transients, which can make drums sound softened. Use them where phase coherence matters, such as on mastering, parallel processing or separated stems that will be recombined, and ordinary designs elsewhere.

Dynamic equalisation

A dynamic equaliser band acts only when the signal in that band exceeds a threshold, combining equalisation and compression. It is ideal for problems that come and go: a singer's harsh resonance on loud notes, a boomy note on a bass guitar, or sibilance. Many mastering and mixing assistants rely heavily on dynamic equalisation, because it lets them correct problems without changing the sound when the problem is absent.

Where frequencies live

Rough frequency regions help you name what you hear. These are guides, not rules, and real instruments overlap them freely. Below about 60 Hz is sub-bass, felt as much as heard, and the home of kick drum weight and the lowest bass notes. Between about 60 and 250 Hz is bass and warmth; too much here is the most common cause of muddy mixes. From about 250 Hz to 2 kHz is the midrange, where most instruments' fundamentals and the body of the voice sit; excess around 300 to 500 Hz sounds boxy, and excess around 1 kHz can sound nasal. From about 2 to 6 kHz is the presence region, where the ear is most sensitive and where

intelligibility and attack live; too much sounds harsh. Above about 6 kHz is brilliance and air: sibilance, cymbals and the sense of space.

Using equalisers well

Cut before boosting where you can, because removing what is wrong often makes what is right audible. Use narrow cuts to remove resonances and broad boosts to shape tone. Equalise in the context of the mix, not in solo; a guitar that sounds thin alone may sit perfectly against the bass. Level-match before and after: equalisers change loudness, and louder always sounds better at first. And remember masking from Chapter 2. Much of mixing is giving each source its own space in frequency, cutting a little from one where another needs to be heard.

Compression

A compressor reduces the level of a signal when it exceeds a threshold, narrowing the gap between loud and quiet parts. Used well, it makes a performance more even, adds sustain and punch and helps a source sit consistently in a mix. Used badly, it flattens life out of the music.

The controls

The threshold sets the level above which compression begins. The ratio sets how strongly the level is reduced above the threshold: at 4:1, a signal 8 dB over the threshold comes out only 2 dB over. The attack time sets how quickly the compressor responds once the threshold is crossed, and the release time sets how quickly it recovers once the signal drops. The knee sets whether compression starts abruptly at the threshold or gradually around it. Make-up gain raises the output to compensate for the reduction.

Attack and release shape the sound most. A slow attack lets the initial transient of a drum or a plucked note through before the compressor clamps down, emphasising punch. A fast attack catches the transient and softens it. A fast release recovers quickly and can make a sound denser and louder, but may cause audible pumping or distortion on low frequencies. A slow release sounds smoother but can hold the level down into the next phrase.

Variations

Sidechain compression uses a different signal to control the compressor. The classic example is ducking: compressing a bass guitar or a music bed whenever the kick drum or a voice-over occurs, so that the more important sound comes through. In podcast and broadcast work, automatic ducking of music under speech is a form of sidechain compression.

Parallel compression blends a heavily compressed copy of a signal with the uncompressed original. The compressed copy lifts the quiet detail, while the original preserves the transients. It is widely used on drums and vocals.

Multiband compression splits the signal into frequency bands and compresses each separately, so that a boomy bass note can be controlled without affecting the vocal. It is powerful and easy to overuse; in mastering, many engineers prefer dynamic equalisation for the same job.

De-essers, gates and expanders

A de-esser is a compressor that reacts only to sibilant frequencies, typically somewhere between 4 and 10 kHz depending on the voice, and reduces them when they become harsh. Gates and expanders do the opposite of compressors: they reduce the level of a signal when it falls below a threshold, which can remove background noise between phrases or tighten drum spill. Gates are blunt instruments for noise compared with the AI noise reduction in Chapter 8, but they remain useful because they are predictable and transparent when set well.

Limiting

A limiter is a compressor with a very high ratio and fast attack, used to stop a signal from exceeding a ceiling. On the master bus, a limiter controls the final peak level and allows the overall loudness to be raised. Modern limiters look ahead at the incoming signal to anticipate peaks, and true-peak limiters also account for the peaks that can appear between samples when audio is converted back to analogue or encoded to a lossy format.

Limiting is where the loudness war was fought. For many years, the commercial pressure to sound louder than competitors drove masters to extreme limiting, sacrificing dynamics and adding distortion. Loudness normalisation on streaming services has reduced that pressure, as Chapter 6 explains, but the habit lingers. A good rule is to limit only as far as the music sounds better, not as far as the meter allows.

Saturation and distortion

Saturation adds harmonics to a signal by shaping its waveform nonlinearly, imitating the behaviour of tape, valves and transformers driven hard. Gentle saturation can make a sound feel warmer, denser or more present, and can make a bass audible on small speakers by adding harmonics that those speakers can reproduce even when they cannot reproduce the fundamental. Heavier distortion is a creative effect in its own right.

Because saturation creates new frequencies, it can create aliasing in digital systems, as Chapter 2 explained. Better plug-ins oversample to avoid it. Saturation is also one of the main targets of neural modelling: neural amp and pedal models, discussed in Chapter 16, learn the nonlinear behaviour of real hardware from recordings of its input and output.

Reverberation

Reverberation effects put a sound in a space. Algorithmic reverbs build that space from networks of delays and filters designed to imitate rooms, halls, plates and springs; they are flexible and efficient. Convolution reverbs use an impulse response, a recording of how a real space or device responds to a short, sharp sound, and apply it to the signal mathematically. Convolution reproduces real spaces precisely and is widely used in post-production to match dialogue recorded in a studio to the location seen on screen. Machine-learning research now extends both approaches: estimating a room's impulse response from a recording, or generating reverberation that matches a video scene.

The key controls are pre-delay, which sets the gap between the dry sound and the start of the reverb and helps keep the source clear; decay time;

tone, usually controlled by damping or equalisation; and the balance of early reflections against late reverberation. Feeding reverbs from sends rather than inserts lets several sources share a space and keeps the mix coherent. Equalising the reverb return, often with a high-pass filter around a few hundred hertz, stops it from muddying the low end.

Delay and modulation

A delay repeats a signal after a set time. Short delays, under about 30 milliseconds, are heard as thickening or colouration rather than echoes, because of the precedence effect. Longer delays are heard as distinct repeats and are often synchronised to the tempo. Filtering the repeats, so each is darker than the last, makes them sit behind the dry sound.

Modulation effects, such as chorus, flanging and phasing, vary short delays or filters over time to create movement and width. They are common creative tools and are among the effects that neural models have learned to imitate from examples.

Stereo and mid/side

A stereo signal can be described as left and right, or as mid and side. Mid is the sum of the two channels, the part common to both; side is the difference, the part that differs between them. Processing mid and side separately lets you, for example, add brightness to the sides to widen a mix without affecting the centred vocal, or remove low frequencies from the sides so the bass stays centred and mono-compatible.

Mono compatibility matters because many playback systems sum to mono, including some phone speakers, smart speakers and public address systems. Wide stereo effects that rely on phase differences between channels can partially cancel when summed. Check every mix in mono.

Automation

Automation records changes to faders, pans and plug-in parameters over time, so that a mix can respond to the music: raising a vocal phrase that drops under the band, opening a filter into a chorus, riding the level of a

podcast guest who leans away from the microphone. Much of what distinguishes a professional mix is detailed automation rather than elaborate processing. Some AI tools now generate automation, for example by riding vocal or dialogue levels to keep them at a consistent loudness. That is useful, but check that the automation follows the meaning of the performance as well as its level.

Processing for small speakers and earbuds

Most music is now heard on small speakers, earbuds and phones, which reproduce little or nothing below about 100 Hz and often emphasise the upper midrange. A mix that relies on sub-bass for its energy can sound empty on them. Engineers respond in three ways. They make sure that bass parts have harmonic content in the low midrange, through saturation, layering or the choice of sound, so that the ear perceives the bass line even when its fundamental is missing. They watch the presence region, because small speakers exaggerate harshness there. And they check mixes on small speakers throughout the process, not only at the end. None of this means mixing for phones alone; it means making sure that the essential elements of the music survive on the systems most people use.

Processing order and restraint

There is no single correct order for processors, but there are sensible defaults. Clean up first, with high-pass filters, repair and noise reduction. Correct tone next, with equalisation. Control dynamics after that, with compression. Then colour with saturation, and send to shared reverbs and delays. Finally balance with faders and automation.

The most useful principle is restraint. Every processor has costs: changes to phase, transients, noise, distortion or space. A source that was well recorded often needs very little. When an AI assistant proposes a chain of eight processors with confident settings, ask of each one what problem it solves, and remove any that you cannot hear doing something useful at matched loudness.

Listening exercises

1. Sweep a narrow boost slowly through a vocal from 200 Hz to 8 kHz. Note where it sounds boxy, nasal, harsh and sibilant. Then try narrow cuts at the worst spots.
2. Compress a drum loop at 4:1 with fast attack and then with slow attack. Describe what happens to the hit of each drum.
3. Put a reverb on a send and change the pre-delay from 0 to 80 milliseconds. Listen to how the vocal moves forward as the gap grows.
4. Switch a wide stereo mix to mono. Note which elements get quieter or change tone.

In practice

- High-pass most sources, but set the cutoff by ear, not by habit.
- Cut resonances narrowly, shape tone broadly and always compare at matched loudness.
- Choose compressor attack and release by listening to what happens to transients and phrase endings.
- Use linear-phase equalisation where phase coherence between paths matters, including recombined separated stems.
- Feed reverbs from sends, high-pass their returns and check the mix in mono.
- Judge every AI-proposed processing chain one processor at a time.

CHAPTER 6

Loudness and Delivery

Finishing a piece of audio means delivering it: handing a file to a distributor, a broadcaster, a streaming service, a podcast host, a game engine or a cinema in a form that meets their requirements and survives their playback chain. For most of recorded history the requirement that mattered most was peak level, because the medium distorted above a certain point. Since the late 2000s the industry has moved to measuring perceived loudness instead, and that change has altered how music is mastered and how dialogue is mixed. It also gives AI tools a clear, measurable target, which is one reason loudness was among the first parts of the workflow to be automated.

This chapter explains how loudness is measured, what the main delivery targets were as of late 2026 and how to check a file before it leaves your hands.

Why peaks were not enough

A peak meter shows the highest sample value in a signal. It tells you whether the signal will clip, but almost nothing about how loud it sounds. A recording of a single clap can peak at full scale and sound quiet overall; a heavily limited song can peak at the same value and sound very loud for its whole length. Broadcasters who normalised programmes by peak level found that adverts, which were compressed and limited aggressively, sounded far louder than the programmes around them. Listeners complained, and in several countries regulators intervened. In the United States, the CALM Act of 2010 led the Federal Communications Commission to require broadcasters to follow the ATSC A/85 recommended practice for loudness.

Measuring loudness: BS.1770

The international basis for loudness measurement is Recommendation ITU-R BS.1770, published by the International Telecommunication Union and revised several times; the fifth revision, BS.1770-5, dates from November 2023. It specifies an algorithm that approximates how loud a programme sounds to a typical listener.

The algorithm works in four steps. First, each channel is filtered with a "K-weighting" curve, which reduces very low frequencies and boosts high frequencies slightly to approximate the ear's sensitivity and the acoustic effect of the head. Second, the mean square energy of each channel is measured over short blocks of 400 milliseconds, overlapping by 75 percent, and the channels are summed with weights; surround channels are weighted slightly higher than front channels. Third, gating removes quiet passages from the calculation: blocks below an absolute threshold of −70 LKFS are discarded, then blocks more than 10 LU below the loudness of the remaining blocks are discarded too. Fourth, the gated blocks are averaged to give the integrated loudness of the whole programme.

The result is expressed in LKFS, loudness K-weighted relative to full scale, which the European Broadcasting Union calls LUFS. The two terms mean the same thing. Differences in loudness are expressed in LU, loudness units, where one LU equals one decibel.

BS.1770 also defines how to measure true peak. Because the waveform between samples can exceed the highest sample value, especially after conversion to analogue or after lossy encoding, true-peak meters oversample the signal to estimate those inter-sample peaks. True peak is written in dBTP.

Related measures

The EBU added two further measures in its supporting documents. Momentary loudness uses a 400-millisecond window and short-term loudness a 3-second window; both are useful for watching a mix in real time. Loudness range, or LRA, describes the variation in loudness across a programme, ignoring the very quietest and loudest moments. A heavily compressed pop song might have an LRA of a few LU; a film soundtrack

may have a much larger one. LRA is useful for checking whether dialogue will be comfortable to listen to on a television in a living room.

Dialogue-gated loudness, used by some broadcasters and streaming services, measures loudness only where speech is detected. It reflects the fact that viewers set their volume by dialogue, not by explosions. Speech detection in loudness meters is itself a form of machine listening, and the details of the detector can affect the result.

Broadcast targets

The EBU's recommendation R 128 sets a target programme loudness of −23.0 LUFS for European broadcast, with a small tolerance and a wider one for live programmes where exact normalisation is impractical, and a maximum true peak of −1 dBTP. The accompanying production guidelines, EBU Tech 3343, explain how to work to the target.

ATSC A/85, used in the United States, sets a target of −24 LKFS for television. Australian free-to-air broadcasters follow Free TV Australia's operational practice OP-59, also based on −24 LKFS. Japan, other regions and individual broadcasters have their own specifications, usually within a decibel or so of these figures. Always ask for the actual delivery specification for a job; broadcasters attach requirements for true peak, loudness range, channel layout and file format that matter as much as the target itself.

Streaming and normalisation

Music streaming services normalise playback so that tracks play at similar perceived loudness, which removes much of the incentive to master louder than everyone else. Each service has its own approach, and not all publish it.

Spotify documents its approach on its artist support pages. As of November 2026 it normalises by default to −14 LUFS integrated, with listener settings for louder (−11 LUFS) and quieter (−19 LUFS) playback on premium accounts. Tracks louder than the target are turned down. Tracks quieter than the target are turned up only as far as their true-peak headroom allows, so a quiet, dynamic master with peaks near 0 dBTP

will play quieter than its neighbours rather than being limited by the platform. Spotify also normalises at album level when an album is played in sequence, preserving the intended differences between tracks.

Apple Music applies its Sound Check normalisation, and YouTube applies normalisation to music videos, but as of late 2026 neither publishes an official integrated loudness target in the way Spotify does. Figures commonly quoted, such as around −16 LUFS for Apple Music and around −14 LUFS for YouTube, come from third-party measurements. Treat them as observations, not specifications.

The practical consequence is that there is no single "streaming loudness" to master to. Mastering much louder than about −14 LUFS integrated buys nothing on services that normalise; the track is simply turned down, and if the loudness came from heavy limiting, it sounds flatter than less limited tracks at the same playback level. Mastering much quieter can cost you on services that do not raise quiet tracks fully. Many mastering engineers therefore aim for a master that sounds right for the genre, typically somewhere between about −14 and −9 LUFS integrated for loud contemporary styles and lower for acoustic and classical music, with true peaks at or below −1 dBTP. That is a judgement, not a rule.

Film and series delivery

Streaming platforms for film and television publish detailed sound specifications for their partners. Netflix's partner documentation, as of 2026, asks for near-field 5.1 and stereo mixes with dialogue-gated loudness of −27 LKFS, with a tolerance of plus or minus 2 LU, measured over the full programme, and with true peaks no higher than −2 dBTP. For Dolby Atmos deliverables it describes checking loudness and peaks through a 5.1 re-render. The figure is lower than broadcast targets partly because it measures dialogue only and partly to leave room for wide dynamics in effects and music.

Theatrical cinema works differently. Cinema mixes are calibrated to a reference playback level in the dubbing stage rather than to an integrated loudness target, and are then re-mixed or adjusted for home and streaming release, a process often called a near-field or home mix. Chapter 14 covers these deliverables in more detail.

Podcasts and spoken word

Podcasts have no regulator, but there are widely used recommendations. The Audio Engineering Society's technical documents on loudness for streaming and on-demand distribution, AES TD1004 from 2015 and its successor TD1008 from 2021, recommend integrated targets in roughly the range of −16 to −20 LUFS for internet audio, with true peaks no higher than −1 dBTP, and discuss dialogue-based measurement for speech content. Some podcast platforms publish their own guidance; Apple's podcast production guidance has long recommended around −16 LUFS. Check the current documents before a job, because platform guidance changes.

Speech-heavy content benefits from moderate loudness range. A listener in a car or on a train cannot hear a guest who is 10 dB quieter than the host. Levelling tools, which ride the gain of each speaker to a consistent loudness, are among the most useful AI-assisted features in podcast tools. Check their work: a levelling algorithm that boosts every breath and cough to the same loudness as speech is not helping.

Codecs and true peak

Most listeners hear lossy-encoded audio: AAC, Opus, Ogg Vorbis or MP3. Encoding removes information and changes the waveform, and the decoded signal can have peaks higher than the original. A master that peaks at −0.1 dBTP can clip after encoding. That is why most delivery specifications set a true-peak ceiling of −1 dBTP or lower, and why some mastering engineers go lower still for very loud masters.

The best way to check is to listen to and measure your master after encoding. Several tools can preview a file through common codecs at typical bitrates. Listen for added harshness in cymbals and sibilance, smearing of transients and artefacts in reverb tails. AI-generated music and heavily processed AI-separated stems can contain unusual high-frequency content that codecs handle poorly, so this check matters especially for them.

Reading a loudness meter

A loudness meter usually shows several numbers at once, and beginners often watch the wrong one. Momentary loudness jumps around with every word and drum hit; it is useful for seeing whether a section is broadly where you expect, not for hitting a target. Short-term loudness, over three seconds, is the most useful number to watch while mixing: it shows how loud a passage feels. Integrated loudness accumulates over the whole programme and is the number specifications refer to; it only means something once the meter has measured the entire file, from the first second to the last, so reset it at the start and play or analyse the whole piece. True peak shows the highest estimated inter-sample peak so far. Loudness range is calculated over the whole programme too.

Two habits prevent most loudness mistakes. Measure the final file, not the session: plug-ins on the master bus, dither and export settings can all change the result. And measure offline where possible, using the meter's file analysis, so that the whole programme is included and nothing depends on real-time playback. For dialogue-gated specifications, use a meter whose dialogue detection matches the specification, because different detectors can give different results on the same programme.

Metadata and file formats

Delivery is not only audio. Music distribution requires identifiers and metadata: an ISRC code for each recording, songwriter and publisher information, credits and, increasingly, disclosure of AI use. Chapter 13 covers music metadata in detail, including the DDEX fields that allow AI involvement to be declared.

Broadcast and post-production often require Broadcast Wave Format files, which carry timecode and descriptive metadata, at 48 kHz and 24 bits, with specific channel layouts and naming conventions. Immersive deliverables, discussed in Chapter 18, use formats such as ADM Broadcast Wave files that carry object positions and rendering metadata alongside the audio.

Always deliver what the specification asks for, named as it asks, with a written report of measured loudness, true peak and, where relevant, loudness range.

Automating loudness with care

Loudness normalisation is easy to automate because the measurement is standardised. Many DAWs can render to a target, many podcast tools normalise automatically, and almost every AI mastering service asks for a target LUFS value. Automation is safe for the measurement. It is less safe for the judgement about how to reach the target.

There are two ways to reach a loudness target. One is simple gain: turning the whole programme up or down. It changes nothing else, but it can push peaks above the true-peak limit. The other is dynamics processing: compression and limiting to raise loudness without raising peaks. It changes the sound. An automated tool that hits −14 LUFS on a quiet folk song by limiting it hard has met the number and damaged the music. Check how a tool reached the target, and listen.

A delivery check

Before any file leaves Lantern Street, an engineer runs the same check. Measure integrated loudness, true peak and loudness range with a BS.1770-compliant meter and compare them with the specification. Listen to the whole file from start to finish at a moderate level, on headphones, without looking at the screen, noting the time of anything odd. Listen to the beginning and end for clicks, cut-off reverb tails and stray noise. Check the file format, sample rate, bit depth, channel layout and naming. Encode a copy to a typical lossy format, measure its true peak and listen to the loudest and most detailed passages. Write the measurements into the delivery note. Appendix A contains the checklist.

That process takes time, and it is exactly the kind of work that machine-listening tools can increasingly help with by flagging clicks, dropouts and loudness problems automatically. They are good at finding candidates for a human to check. The final listen remains a human job.

Loudness and AI-processed material

Two kinds of AI-processed material deserve particular care at delivery. Generated music often arrives already limited to a high loudness, sometimes with an unstable stereo image and high-frequency content that behaves badly in codecs. Treat it like a client master of unknown quality: measure it, listen through a codec and, if it is part of a larger programme, bring it to the programme's loudness with gain rather than further limiting where possible.

Material reassembled from separated stems may measure differently from the original mix even when it sounds similar, because separation can change peak structure and low-frequency content. Remeasure after any separation-based remix or upmix, and check true peak especially, because recombined stems with small timing or phase differences can produce unexpected peaks.

Synthetic speech is often more consistent in level than recorded speech, so its dialogue-gated loudness can be close to the target while sounding flat. Meeting the number is necessary; it is not sufficient.

In practice

- Use a BS.1770-5-compliant meter and know which loudness measure a specification uses: integrated, dialogue-gated or short-term.
- Ask for the actual delivery specification for every broadcast, streaming or platform job.
- For music, master to what sounds right for the genre; very loud masters gain nothing on normalised services.
- Keep true peaks at or below −1 dBTP for most streaming deliveries, lower for very loud masters, and at or below −2 dBTP where specified.
- Check masters after lossy encoding, especially AI-generated or heavily processed material.
- Automate measurement freely; check how any tool reached the target before trusting it.

PART II

The AI Toolkit

CHAPTER 7

How Audio Models Hear and Speak

You do not need to build neural networks to use them well, but you do need a working picture of what is happening inside them. Without that picture, every model is a black box whose failures arrive as surprises. With it, you can predict where a model is likely to struggle, why it might run at 16 kHz, why it adds latency, why two tools built on the same idea sound different and why a benchmark score may not predict how it will handle your material.

This chapter explains how audio is represented for machine learning, the main families of model architecture, the neural audio codecs that underpin much of modern generative audio, the main ways models generate sound and how they are evaluated. Specific model names are given with dates as examples; the field moves quickly and the categories last longer than the names.

Representing sound for a model

A neural network operates on arrays of numbers. Audio can be given to it in several forms, and the choice shapes what the model can do.

Waveforms

The simplest representation is the raw waveform: the sequence of samples. It contains everything, including phase, but it is long: one second at 48 kHz is 48,000 numbers, and the patterns that matter to hearing unfold over thousands of samples. Models that work directly on waveforms need to capture structure at many time scales at once. WaveNet, published by DeepMind in 2016, was a landmark because it generated speech sample by sample from a waveform model with an unusually natural quality, though at a very high computational cost.

Spectrograms

More often, audio is converted into a spectrogram using the short-time Fourier transform, or STFT. The signal is cut into short overlapping frames, typically tens of milliseconds long, and each frame is analysed into frequency components. The result is a two-dimensional picture of energy over time and frequency, much like the display in a spectral editor. Each point also has a phase, which describes the timing of that component within the frame. Many models work only on the magnitude and either reuse the original phase, estimate it or hand the job to a separate network.

A mel spectrogram warps the frequency axis to approximate the ear's frequency resolution, discussed in Chapter 2, and usually reduces it to between 64 and 128 bands. Mel spectrograms are compact and perceptually meaningful, and for years they were the standard input to speech and sound-recognition models and the standard intermediate representation in text-to-speech systems. Their weakness is that they discard information. A mel spectrogram cannot be turned back into audio exactly; something has to invent the missing detail.

Vocoders

That something is a vocoder: a model that turns a spectrogram or another compact representation back into a waveform. Neural vocoders such as HiFi-GAN, published in 2020, and later designs including BigVGAN and Vocos made this step fast and high in quality. Many systems you will use, including text-to-speech and some enhancement tools, consist of one model that predicts a spectrogram and a vocoder that renders it. When such a system produces a metallic, buzzy or phasey artefact, the vocoder is a common culprit.

Learned representations

The most important shift of the early 2020s was toward representations learned by the models themselves. Self-supervised speech models such as wav2vec 2.0, HuBERT and WavLM learned from large amounts of untranscribed speech to produce internal features that capture phonetic content, speaker identity and other properties, which could then be used for recognition, separation or synthesis. In general audio, models

trained on huge labelled or paired datasets learned embeddings in which similar sounds lie close together. Chapter 17 discusses those embeddings for machine listening; the next section discusses the learned representations that generative systems depend on.

Neural audio codecs

A neural audio codec compresses audio into a compact sequence of discrete codes, or tokens, using a learned encoder, and reconstructs it with a learned decoder. It is a lossy codec like MP3, but where MP3's compression is designed by hand around a model of masking, a neural codec's compression is learned by training the encoder and decoder together to reconstruct audio that sounds right.

The modern line began with Google's SoundStream in 2021, followed by Meta's EnCodec in 2022 and the Descript Audio Codec, known as DAC, in 2023. These systems introduced and refined residual vector quantisation: the encoder produces a vector for each short frame of audio, a first codebook picks the closest entry, a second codebook encodes the remaining error, and so on, so each frame is represented by a stack of tokens. Using more codebooks raises quality and bitrate. These codecs can reach quality close to the original at a few kilobits per second for speech and somewhat more for music.

Their significance for this book is not mainly file compression. It is that they turn audio into tokens, and tokens are what language-model-style networks are good at predicting. Once audio is a sequence of tokens, the techniques that made text models powerful can be applied to sound. Many of the music, speech and sound generators in Chapters 10 and 11 work by predicting codec tokens and then decoding them.

That has driven intense research into codecs designed for generation rather than storage. Kyutai's Mimi codec, released with its Moshi dialogue model in 2024, runs at a low frame rate of 12.5 frames per second and distils semantic information from a self-supervised speech model into its first codebook, so that the first token of each frame carries meaning and later tokens carry acoustic detail. Research through 2025 and 2026 pushed bitrates lower still and toward single-codebook designs that are simpler for generators to model. FocalCodec-Stream, described

in a September 2025 paper, compresses speech to a single binary codebook at 0.55 to 0.80 kilobits per second with a theoretical latency of 80 milliseconds. HH-Codec, from mid-2025, reports operation at 24 tokens per second for 24 kHz speech, at about 0.3 kilobits per second. ContextCodec, posted in June 2026, reports a useful balance of quality and intelligibility down to 500 bits per second and real-time operation on a typical mobile processor. A low-resource audio codec challenge held in 2025, LRAC, gave such systems a shared test. These are research results reported by their authors, mostly for speech; the general direction, toward fewer tokens per second that carry more meaning, matters more than any one number.

For working engineers, codecs matter in three ways. They set a quality ceiling on any generator built on them: a music model whose codec smears cymbals will always produce smeared cymbals. They define the sample rate and bandwidth of the output. And they are starting to appear in communication tools, where very low-bitrate neural speech codecs make calls workable on poor connections at the cost of occasional strange artefacts, such as words that sound subtly mispronounced, which matter if you are recording a remote interview.

Model architectures in brief

Several network architectures recur in audio.

Convolutional networks apply small learned filters across time and frequency. They are efficient and good at local patterns such as onsets and harmonics. The U-Net, a convolutional design that compresses its input and then expands it back with shortcuts between matching levels, has been widely used for separation and enhancement because its output has the same shape as its input.

Recurrent networks, including LSTMs, process sequences step by step and carry a memory forward. They suit streaming and low-latency processing and remain common in real-time noise suppression and neural amp modelling.

Transformers use attention, letting each position in a sequence draw on every other position. They model long-range structure well, such as the relationship between a chorus and its repeat, and they dominate recent

separation, recognition and generation models. Their cost grows with sequence length, which is one reason compact codec tokens matter.

State-space models and other efficient sequence models are increasingly used where long sequences and low latency are both needed.

Hybrid designs combine these elements, and some, discussed in Chapter 16, combine neural networks with classical signal processing such as filters and oscillators.

How models generate sound

Generative audio models fall into a few families. Knowing which family a tool belongs to helps explain its behaviour.

Autoregressive models

An autoregressive model generates a sequence one step at a time, each step predicted from everything before it, like a text model predicting the next word. Applied to codec tokens, this approach produced AudioLM in 2022, MusicGen in 2023 and the VALL-E family of text-to-speech models. Autoregressive generation is good at long-term coherence and natural continuation, and it can stream output as it goes. It tends to be slow because steps are sequential, and errors can accumulate: a model that drifts slightly off pitch or tempo may keep drifting. In speech, autoregressive models occasionally repeat or skip words.

Diffusion and latent diffusion

A diffusion model learns to reverse a process that gradually adds noise. To generate, it starts from noise and removes it step by step, guided by a condition such as a text description. Latent diffusion does this not on raw audio but on a compressed latent representation produced by an autoencoder, which is far cheaper. AudioLDM in 2023 and Stability AI's Stable Audio models are examples. Diffusion models generate a whole segment at once, which makes them good at producing a clip of a fixed length and at inpainting, filling a gap in existing audio. They can be slower than autoregressive models for long outputs unless distilled to fewer steps.

Flow matching

Flow matching is a related technique that learns a direct path from noise to data, often needing fewer steps than diffusion. It has been prominent in speech synthesis since about 2023, including Meta's Voicebox research and the F5-TTS model released in 2024, and it is increasingly used in music and effects generation.

Masked and parallel generation

Masked generative models predict many tokens in parallel, filling in masked positions over several passes. They trade some of the sequential coherence of autoregressive models for speed. Many current systems combine approaches, for example an autoregressive model for coarse structure and a diffusion or flow model for detail.

Conditioning

Every generator is steered by its conditioning: text descriptions encoded by a language model or a joint audio-text model such as CLAP, a reference recording to imitate, a melody, chords, a tempo, lyrics, a transcript, a video, or a time range to fill. The richer and more precise the conditioning a tool accepts, the more control you have. Many practical frustrations with generative tools are conditioning problems: the model cannot be told that the snare should be quieter because nothing in its conditioning represents the snare. Chapter 11 returns to this.

Training data

A model can only learn what is in its data. Speech models trained mostly on English studio recordings handle English studio recordings best. Separation models trained on popular music stems may struggle with orchestral music or field recordings. Enhancement models trained on synthetic mixtures of clean speech and noise may behave strangely on real reverberant rooms. Where possible, find out what a model was trained on, and test it on material like yours before committing a project to it.

Training data also carries legal and ethical weight. Whether training on copyrighted recordings without permission is lawful was being litigated in several countries as of late 2026, and some vendors now advertise models trained only on licensed or owned data. Chapter 20 covers the state of play.

How models are evaluated

Audio research reports results with a mix of objective metrics and listening tests. Knowing what each measures helps you read claims sensibly.

For separation, signal-to-distortion ratio, or SDR, measures how close a separated stem is to the true stem, in decibels. Higher is better, but SDR does not correspond perfectly to how a stem sounds; two outputs with similar SDR can differ in how annoying their artefacts are.

For speech enhancement, metrics such as PESQ and STOI estimate quality and intelligibility, and newer learned predictors estimate listener ratings directly.

For speech synthesis, papers report word error rate, measured by running a recogniser on the output, speaker similarity, measured with a speaker-verification model, and mean opinion scores from listeners.

For music and effects generation, the Fréchet Audio Distance compares the statistics of generated audio with real audio in an embedding space, and text-audio alignment scores measure how well output matches its prompt. Both are rough. Listening tests remain the most trusted evidence.

Formal listening tests such as MUSHRA, standardised by the ITU, present listeners with hidden references and anchors to make ratings comparable. When a vendor reports that listeners preferred its output, ask how many listeners, on what material, against what, with what playback and whether the comparisons were level-matched.

Speed, latency and compute

Two numbers describe how practical a model is. The real-time factor is processing time divided by audio duration: a factor of 0.1 means ten

seconds of audio take one second to process. Latency is the delay between input and output for streaming use, which depends on how much future audio the model needs to see and how long each block takes.

Offline tools such as separation and generation can afford real-time factors near or above one if the result is good. Live tools need latency within a few milliseconds for performers or within tens of milliseconds for calls and broadcasts. Many models can run either on a graphics processor or more slowly on a central processor; some vendors run models only in the cloud. Chapter 21 turns these figures into costs.

Running models yourself

Many of the models named in this book are available as open weights that anyone can download and run. Running a model locally has real advantages for a studio: client audio stays on your machines, the model does not change unless you change it and there is no per-minute fee. It also has costs. You need a computer that can run the model at a useful speed, which for large models usually means a graphics processor with plenty of memory; you need to install and maintain the software environment, which can be fiddly; and you need to read the model's licence, because "open weights" does not always mean free for commercial use. Some open models are released under permissive licences, some only for research or non-commercial use and some under custom terms with conditions.

A middle path is software that packages open models inside a familiar interface, such as a plug-in, a DAW feature or a desktop application. These hide the technical work, at the cost of some control over versions and settings. Whatever route you take, record exactly which model and version a tool uses, so that a result can be reproduced or explained later, a practice Chapter 21 turns into a routine.

In practice

- For any model, find out its internal sample rate, its representation and, where possible, its codec. They set its ceiling.

- Expect autoregressive generators to drift or repeat; expect diffusion generators to be good at fixed-length clips and gap-filling.
- Read benchmark numbers as rough guides; listen on your own material at matched loudness.
- Ask what a model was trained on, both to predict its failures and to understand its rights position.
- Separate offline and live uses: measure real-time factor and latency before putting any model in a live path.

CHAPTER 8

Repair, Restoration and Dialogue Cleanup

Repair is where most engineers first meet AI in their daily work, and where it has had the clearest effect. Problems that used to take hours of careful spectral editing, such as traffic under an interview, air conditioning in a classroom recording or a hum from a bad cable, can now often be reduced in seconds. Repair is also where AI's most dangerous failure mode, the plausible invention, does the most damage, because repair is usually applied to recordings of real people saying real things.

This chapter covers the problems repair tools address, how classical and neural approaches differ, and a working method that gets the benefits without the risks.

The problems

Recordings go wrong in a limited number of ways, and it helps to name the problem precisely before choosing a tool.

Broadband noise is a steady hiss or rush across many frequencies: preamp noise, air conditioning, distant traffic, wind in trees. Tonal noise is concentrated at particular frequencies: mains hum at 50 or 60 Hz and its harmonics, the whine of a computer fan, a buzz from a lighting dimmer. Intermittent noise comes and goes: a door, a cough in the audience, a phone vibrating, a chair creak, clicks from a damaged record. Reverberation is the room's contribution, which makes speech distant and less intelligible. Clipping is the flattening of peaks when a signal exceeded the maximum level somewhere in the chain. Bandwidth loss is missing high or low frequencies, as in a phone recording or an old transfer. Plosives, mouth clicks, sibilance and breath noise come from

the speaker or singer. Dropouts and codec artefacts come from digital transmission, as in remote interviews over the internet.

Each has different classical and AI remedies, and many recordings have several problems at once.

Classical repair

Before neural networks, engineers had three main families of repair tools, and all remain useful.

Filters and equalisers remove noise that sits outside the wanted signal's range, such as rumble below a voice, and notch filters remove narrow tonal noise such as hum. They are transparent and predictable but cannot touch noise that overlaps the voice.

Gates and expanders reduce noise between phrases. They do nothing during the phrases, where noise is masked by speech, and set badly they chop off word endings.

Spectral noise reduction learns a profile of the noise from a section where only noise is present, then subtracts that profile from each frame of the spectrogram. It works well on steady noise. Pushed too far, it produces "musical noise", a watery, chirping residue of random spectral peaks, and dulls the voice. Spectral editors let an engineer see the spectrogram and paint out individual events such as a cough or a siren; skilled spectral editing remains one of the most precise forms of repair, if slow.

These tools share a property worth valuing: they remove, but they do not invent. A spectral subtraction can damage a voice, but it cannot make it say a different word.

Neural enhancement

Neural enhancement models learn from large numbers of examples of clean speech mixed with noise, reverberation and other degradations, and are trained to recover the clean speech. Once trained, they can separate speech from noise that overlaps it in time and frequency, because they have learned what speech sounds like rather than what the noise profile is.

The modern history runs through a few landmarks. Jean-Marc Valin's RNNoise, published in 2018, combined classical signal processing with a small recurrent network to suppress noise in real time on ordinary processors, and it is still embedded in many communication tools. Microsoft's Deep Noise Suppression challenges, run from 2020, gave researchers shared datasets and listening-test evaluations. DeepFilterNet, published in 2022, showed full-band, low-latency enhancement suitable for real-time use. In parallel, commercial tools brought neural enhancement to the mainstream: real-time noise suppression in video-call software and in graphics-card driver suites, dialogue isolation modules in professional repair suites, and one-click "enhance speech" services aimed at podcasters.

Neural enhancement tools fall into two broad types, and the difference matters.

Discriminative enhancement

Most enhancement models are discriminative. They estimate a mask, or a direct transformation, that keeps the parts of the input they judge to be speech and suppresses the rest. Their output is built from the input. When they fail, they typically fail by removing too much, leaving speech that sounds thin, muffled, gated or watery, or by removing too little, leaving residual noise. Like spectral subtraction, they mostly remove rather than invent, although aggressive models can distort consonants enough to change how a word is heard.

Generative restoration

A newer group of tools is generative. Instead of filtering the input, they use it as conditioning and generate new speech that matches what they infer was said, in the speaker's voice, often with studio-quality tone and bandwidth. The results can be startling: a muffled phone recording comes back sounding as if it were recorded in a booth. The risk is in the word "infer". Where the input is ambiguous, because a truck covered a syllable or the speaker mumbled, a generative model will produce its best guess, and the guess will sound as clean and confident as everything around it. It may change a word, drop a hesitation, smooth an accent or

add a breath that becomes a phoneme. It may also alter the speaker's timbre toward the voices it was trained on.

Generative restoration is enormously useful for some jobs, such as a narrator's pickup recorded on a laptop that needs to match the studio session, provided the narrator approves the result. It is inappropriate for others, such as an interview whose words are evidence, unless every word is checked against the original and the processing is disclosed where it matters.

Other repair tasks

Neural approaches now exist for almost every repair task. A few deserve specific notes.

Dereverberation. Removing reverberation was close to impossible with classical tools. Neural dereverb can dry a roomy recording substantially. It tends to work well on moderate reverberation and to sound processed on heavy reverberation, where the direct sound is weak. Removing reverb also removes the sense of place, which may be wanted in a documentary; consider reducing it rather than eliminating it.

Declipping. Classical declippers interpolate the missing peaks from the surrounding waveform. Neural declippers do the same with learned models and can reconstruct heavily clipped material more convincingly. Both are estimates. On a clipped vocal, compare the declipped peak with what you remember of the performance.

Hum and buzz. Tonal noise is still often best removed with dedicated dehum tools that track the fundamental and harmonics precisely; neural tools help where the hum varies or is mixed with other noise.

Clicks, crackle and mouth noise. Declickers and de-crackle tools have been effective for years; neural versions are better at telling clicks from transients such as consonants and percussion. Mouth-click removal is one of the most time-saving tools in voice-over editing.

Bandwidth extension. Some models predict missing high frequencies, turning narrowband phone audio into something that sounds full-band. This is invention by definition: the high frequencies were never

recorded. It can make a phone interview less fatiguing to listen to; it cannot recover what was not there.

Packet-loss concealment. Remote interviews recorded over internet calls suffer dropouts. Concealment models fill short gaps plausibly. Wherever possible, record each participant locally and transfer the files afterwards, which avoids the problem.

Music restoration. Archival transfers of records and tapes involve declicking, decrackling, denoising, speed correction and sometimes separation of instruments for remixing. Neural tools have made restoration of very damaged recordings practical. Archivists generally keep a flat, unprocessed transfer alongside any restored version, and so should you.

Repairing music

Music repair follows the same principles as dialogue repair, with two differences. First, the line between noise and music is less clear. The breath before a sung phrase, the squeak of fingers on guitar strings, the sound of a piano's pedal and the hiss of an old tape can all be part of a recording's character. Removing them makes a recording cleaner and often less alive. Decide with the artist what counts as a problem before treating it.

Second, music is more sensitive to the side effects of neural enhancement. Speech-trained enhancement models can treat sustained instruments as noise, smear reverb tails and pump the background between phrases. Use tools designed for music where they exist, and apply speech enhancement to vocals only after separating them, if at all. For tonal problems in music, such as a buzzing amplifier, dedicated dehum tools that track the fundamental are usually more transparent than broadband neural processing.

Common music repair jobs include removing a click from an edit, reducing amplifier hiss on a quiet guitar part, taming a single harsh resonance that a microphone picked up, recovering a clipped drum hit and reducing spill from one instrument into another's microphone. The last of these is a separation problem in disguise: spill reduction tools trained to recognise, for example, hi-hat in a snare microphone are a

specialised form of the separation in Chapter 9 and work best on recordings similar to their training material.

For archival music restoration, archivists work to a principle of minimal intervention and reversibility. They keep a flat transfer of the original carrier, document every process and avoid "improving" a recording beyond what its makers would have heard. Commercial remastering may go further, but it should start from the same records.

The risks, named

Four risks recur across repair tools.

Changed words. Any model that reconstructs speech can change what is heard. This matters for journalism, documentary, legal, medical and academic recordings, and anywhere a transcript is treated as fact.

Changed identity. Strong enhancement and generative restoration can alter a speaker's timbre, accent or age impression. People notice when their own voice has been changed and often dislike it.

Lost context. Noise and room are information. Removing all traffic from a street interview, or all crowd from a protest recording, changes what the listener understands about where it was recorded.

Over-processing. The quietest artefact can become the most tiring one over an hour-long podcast. Overprocessed speech is hard to listen to even when no single moment sounds wrong.

A working method

Lantern Street uses the following method for dialogue repair. It is conservative by design.

Start by preserving the original. Never process the only copy. Work on a duplicate track or a rendered copy, and keep the original aligned beneath it.

Diagnose before treating. Listen to the whole recording, look at the spectrogram and write down the problems and where they occur. Separate steady problems from intermittent ones.

Treat in order, lightly. Remove rumble and hum with filters first, because they are transparent. Use spectral repair for isolated events. Then use neural enhancement for broadband noise and reverberation, at the lowest strength that makes the speech clear. Many tools let you blend the processed signal with the original, or set how much noise to leave; leaving a little natural noise usually sounds better than leaving none, and masks small artefacts.

Choose the tool type for the job. Use discriminative enhancement by default. Use generative restoration only when the brief allows reconstruction, the speaker or client agrees, and you can check every word.

Check against the source. Play processed and original in alternation, level-matched, especially at the worst passages. Read along with a transcript made from the original, not the processed file. Mark any word whose identity changed.

Listen as the audience will. Check on headphones, on a laptop speaker and in mono. Listen to a long stretch without stopping, to judge fatigue.

Record what you did. Note the tools, versions, settings and the passages treated. If the work is journalism or documentary, tell the producer what kind of processing was used, and in particular whether any of it was generative.

The roadside interview

Return to the documentary interview from Chapter 1, recorded beside a busy road. The diagnosis finds steady traffic rumble, intermittent truck passes that briefly cover the voice, some wind noise and a lavalier microphone that rubbed against a jacket twice.

The engineer high-passes the voice around 80 Hz, removes the jacket rubs with spectral repair, and runs a discriminative dialogue isolator with the noise-retention setting high enough to leave a quiet bed of traffic. That bed keeps the sense of location and hides small artefacts. Two truck passes cover parts of words. A generative restoration tool produces a lovely clean version of those passages, and in one of them the reconstructed word differs from what the producer remembers being

said. The engineer discards the generative pass for those passages and leaves them noisy but honest, with a note to the producer suggesting subtitles there. The final delivery note lists the tools used and states that no generative reconstruction was applied.

That is slower than a single click on “enhance”. It is also the version that a journalist can defend.

Real-time cleanup

Not all repair happens after the fact. Live streams, broadcasts, video calls and remote interviews increasingly run real-time neural noise suppression on the way in. It is often the right choice for live delivery, where a clean signal now is worth more than a perfect one later. It is usually the wrong choice for recording, for the reason given in Chapter 3: whatever is printed cannot be undone.

The practical answer for anything that will be edited later is to record two versions where the tools allow: the processed feed for the live audience and a clean, unprocessed feed for post-production. Many streaming and recording setups can do this with a split before the suppressor. If only the processed signal is available, as with many call recordings, note it in the session sheet, because later repair tools behave differently on audio that has already been through a suppressor, and artefacts from two neural processes in series can combine unpredictably.

Call software also applies automatic gain control and echo cancellation, which can pump levels and chop the start of words when two people talk at once. Ask remote guests to wear headphones, which reduces the need for echo cancellation, and record locally wherever possible.

In practice

- Name each problem precisely before choosing a tool, and remove what classical tools can remove transparently first.
- Prefer discriminative enhancement; reserve generative restoration for jobs where reconstruction is acceptable and approved.

- Leave a little natural noise and room where possible; it sounds better and preserves context.
- Check every processed passage against the original and against a transcript of the original.
- For journalism, documentary, legal and academic work, disclose the type of processing and never let a model decide what was said.
- Record locally on each end of remote interviews rather than relying on concealment.

CHAPTER 9

Taking Mixes Apart: Source Separation

For most of the history of recording, a mix was a one-way door. Once the multitrack tapes were lost, or the session was bounced to stereo, the parts could not be recovered. You could equalise the mix, compress it or filter it, but you could not turn up the vocal without turning up everything that shared its frequencies. Source separation, the task of recovering individual sources from a mixture, was a research problem with modest results until the late 2010s. Within a few years neural networks turned it into a routine tool. By 2024 a stem splitter was built into mainstream DAWs, and by 2025 mastering suites could rebalance a stereo master by separating it internally.

This chapter covers how separation evolved, what current systems can and cannot do, how to judge their output and how to use it in real work.

The problem

A stereo mix gives you two channels. A pop song might contain thirty or more sources: several drums, bass, guitars, keyboards, lead and backing vocals and effects. Recovering them from two channels is mathematically underdetermined; there are infinitely many sets of sources that would sum to the same mix. A separator must therefore rely on knowledge of what sources usually sound like. That knowledge is exactly what neural networks learn from data.

Researchers usually frame music separation around a small number of stems. The standard benchmark, MUSDB18, released in 2017, provides 150 full-length tracks with four stems each: vocals, drums, bass and "other", which contains everything else. A high-quality version,

MUSDB18-HQ, provides uncompressed audio. Its test set of 50 tracks has been the common yardstick for most of the systems discussed here.

Before neural networks

Classical separation used whatever structure a mix happened to have. Centre-channel extraction exploited the fact that lead vocals were usually panned to the centre, subtracting one channel from the other to remove them for karaoke, which also removed bass and drums panned centre and left everything else. Harmonic-percussive separation split sustained tones from transient hits using their different shapes on a spectrogram. Repetition-based methods separated repeating accompaniment from non-repeating vocals. Non-negative matrix factorisation decomposed spectrograms into recurring spectral templates. These methods are still occasionally useful, especially harmonic-percussive separation as a creative effect, but none came close to clean stems.

The neural era

The step change came when supervised networks were trained on many mixtures with known stems.

Deezer's Spleeter, released as open source in 2019, made two-, four- and five-stem separation available to anyone with a laptop. Its quality was limited by later standards, but it changed expectations overnight and spawned a wave of remixes, karaoke tracks and educational tools. Open-Unmix, released the same year as a reference implementation by researchers led by Fabian-Robert Stöter, gave the research community a transparent baseline.

Meta's Demucs, developed by Alexandre Défossez and colleagues, worked directly on the waveform rather than on a spectrogram, and a hybrid version that combined waveform and spectrogram processing did well in the Music Demixing Challenge of 2021. Hybrid Transformer Demucs, published in 2022, added transformer layers and became one of the most widely used open separators.

Band-split models then pushed quality further by splitting the spectrogram into frequency bands and modelling the relationships within and between bands. The Band-Split RNN of 2022 was followed by BS-RoFormer from ByteDance's research team, which replaced the recurrent layers with transformers using rotary position embeddings. BS-RoFormer, trained on MUSDB18-HQ plus 500 additional songs, won the music separation track of the Sound Demixing Challenge in 2023. Its authors reported an average SDR of 11.99 dB on the MUSDB18-HQ test set for that larger system, and 9.80 dB for a smaller version trained without extra data. A 2024 model, SCNet, reported a similar figure of about 9.7 dB for its large version without extra data. These numbers are medians over the four stems; vocals and bass separate better than the "other" stem in almost all systems.

An active community of independent developers fine-tunes and combines these architectures, publishing models specialised for vocals, instrumentals, drums, de-reverberation or particular genres, and comparing them in public leaderboards. Many commercial products are built on such models.

Prompt-based separation

The next shift was away from fixed stems. In December 2025 Meta released SAM Audio, named after its Segment Anything image models, as a single model that separates whatever sound you describe. It accepts three kinds of prompts: text, such as "dog barking" or "man speaking"; a visual prompt, where you mark the object in a video frame that is making the sound; and a time-span prompt, where you indicate an interval in which the target sound occurs. It returns the target and the residual. Systems like this move separation from four fixed categories toward any sound you can name, see or point to, and they are especially relevant to post-production, where the target might be a particular car, a crowd or a single voice. As with any recent model, test it on your material before relying on it.

Separation for picture

Film and television have their own separation problem: splitting a finished soundtrack into dialogue, music and effects. The cinematic

track of the 2023 Sound Demixing Challenge and datasets such as Divide and Remaster, published in 2022, addressed it. Dialogue-music-effects separation lets archives create music-and-effects tracks for dubbing older programmes whose original stems are lost, lets broadcasters raise dialogue relative to background for viewers who find it hard to follow and lets restorers remix old soundtracks. Some televisions and streaming players now offer dialogue enhancement that applies a form of this separation at playback.

What separation sounds like when it fails

Separation artefacts have a recognisable vocabulary, and naming them helps you hear them.

Bleed is the residue of other sources in a separated stem: a ghost of hi-hat in the vocal, a shadow of vocal in the guitar. It is usually most audible when the stem is soloed and least audible in a full mix, where masking hides it.

Phasiness, sometimes described as an underwater or watery sound, comes from the separator assigning parts of the spectrum to the wrong source frame by frame. It is most noticeable on sustained sounds and reverb tails.

Transient smearing softens the attack of drums and plucked instruments when the separator cannot decide where a sharp event belongs.

Timbre loss happens when parts of a source's spectrum that overlap strongly with another source are removed: a vocal that loses its chest resonance where the bass guitar plays, or a guitar that loses its top end under cymbals.

Hallucinated content is less common but real: a separator, especially a generative one, may fill in what it expects a source to contain, adding a breath or a cymbal decay that was not there.

Different architectures fail differently. The BS-RoFormer authors observed that their outputs looked sharper and less foggy on a spectrogram than those of older convolutional models, and that in the 2023 challenge's listening test their outputs were preferred more by musicians and educators than by music producers. That is a useful

reminder that "better" depends on the use. A clean vocal for practice and a usable vocal for a remix are different targets.

Judging separated stems

Lantern Street uses a three-stage check.

First, solo each stem and listen for bleed, phasiness and smearing on headphones. Note where problems occur; they are often concentrated in dense passages.

Second, listen in context. Put the stems back together and rebalance them as the project requires, then compare with the original mix at matched loudness. Many artefacts that are obvious in solo disappear in context, and some that are invisible in solo, such as a loss of cohesion or a change in stereo image, appear only when the stems are recombined.

Third, null-test the recombination. Sum all the stems unchanged and subtract the original mix. In an ideal separation the residue would be silence. In practice it reveals what the separator lost or invented. Some separators guarantee that their stems sum to the input; many do not. If you need perfect recombination, check whether the "other" stem is computed as the remainder after the other stems are removed.

When processing stems that will be recombined, be careful with phase. A minimum-phase equaliser on one stem changes its phase relative to the others and can cause cancellation where the separator split a single sound between stems. Linear-phase processing avoids that, at the cost of latency.

Choosing and combining models

No single model is best at every stem on every recording. Professionals often run several models and pick the best output for each stem or even each section. Some tools ensemble models automatically, averaging their outputs to reduce the distinctive artefacts of each.

A few practical rules help. Choose a vocal-specialist model for vocals and a drum-specialist model for drums. Process at the highest quality setting and native sample rate available; many models were trained at 44.1 kHz and should be run at that rate. Feed the separator the best source you

have, preferably uncompressed; separation from a low-bitrate stream will separate codec artefacts too. Run separation before other processing, not after mastering-style limiting, which smears sources together. And when a single pass leaves residue, consider a second pass on the separated stem with a different model.

What separation is used for

Separation now supports a wide range of work.

Remixing and remastering. Engineers can rebalance old mixes, for example raising a buried vocal or reducing an over-bright cymbal, when original multitracks are lost. Some mastering tools expose this directly: iZotope's Ozone 12, released in 2025, includes a Stem EQ module that separates a stereo mix into vocals, bass, drums and other inside the plug-in and lets the engineer equalise each separately. High-profile restorations have used separation too. The Beatles' 2023 single "Now and Then" was completed using a vocal from a 1970s home demo by John Lennon that machine-learning tools developed by Peter Jackson's team had separated from the piano on the same cassette, a technique first developed for the 2021 documentary *Get Back*.

Sampling and production. Producers isolate a drum break or a vocal phrase from a record to sample it. The technical ease of that has not changed the legal position: sampling a recording still requires permission from its owners, as Chapter 20 explains.

Karaoke, practice and education. Removing a vocal or an instrument lets musicians practise along, and lets teachers isolate parts for students.

Transcription and analysis. Separating a bass line or a melody makes it easier for people and for automatic transcription tools to follow.

Post-production. Dialogue extraction, music-and-effects creation, and the removal of unlicensed background music from location recordings, a frequent problem in documentary and reality television, all rely on separation.

Accessibility. Dialogue enhancement at playback helps viewers with hearing loss.

Forensics. Separation can isolate speech from background, though forensic use raises the same evidential cautions as generative restoration in Chapter 8.

Separation as a creative tool

Separation is not only a repair tool. Producers use it to create: pulling the drums out of a loop to process them separately, isolating the reverb tail of a vocal to use as a pad, splitting a field recording into its components and rearranging them, or running a separation model deliberately on material it was not designed for and using the strange artefacts as texture. Harmonic-percussive separation, one of the oldest techniques, remains a favourite for this, splitting a sound into its sustained and transient parts so that each can be treated differently. Used creatively, the artefacts that are failures in restoration can become a sound in their own right; the rights questions, of course, remain the same.

Rights and separation

Separation creates stems, but it does not create rights. A separated vocal from a commercial recording is still part of that recording, owned by whoever owns the master, and still contains a performance by the artist and a composition owned by its writers and publishers. Releasing a remix made from separated stems needs the same permissions as releasing one made from the original multitracks. Training a model on separated stems from commercial music is caught by the same legal disputes about training data as any other use. If you separate a client's own recordings for their own project, the position is simple; if you separate anyone else's, assume you need permission.

The phone vocal, revisited

Back at Lantern Street, the singer-songwriter's 2014 phone recording goes through two vocal-specialist separators and an ensemble. The best result has a clean voice in the verses and audible guitar residue in two sustained notes in the chorus, along with the boxy tone of the phone's microphone and the bedroom it was recorded in. Enhancement removes some of the room; equalisation helps the tone. The vocal is good enough

for a lo-fi intro, layered under the new recording, where its character is the point. It is not good enough to carry the whole song, which is why the singer re-recorded it. That is a typical outcome: separation can rescue material and create new options, but it rarely makes a compromised recording equal to a good one.

In practice

- Separate from the best available source, uncompressed and before any heavy limiting.
- Try more than one model; choose per stem or per section, or ensemble.
- Check stems solo, in context and by null-testing the recombination.
- Use linear-phase processing on stems that will be recombined.
- Expect bleed in dense passages, phasiness on sustained sounds and softened transients; listen for them specifically.
- Remember that separated stems carry the same rights as the original recording and composition.

CHAPTER 10

Voices: Synthesis, Cloning and Conversion

No area of AI audio has moved faster or raised harder questions than synthetic voice. In 2016 a natural-sounding synthetic voice was a research result that took minutes of computation per second of speech. By 2023, research systems could imitate a speaker from a three-second sample. By 2026, commercial and open models offered expressive speech in dozens of languages, real-time voice conversion on a laptop and conversational agents that listen and reply in a fraction of a second. A voice is also part of a person's identity, and the technology that lets a narrator license a digital double of their voice also lets a fraudster impersonate a relative on the phone.

This chapter explains the main technologies, how to direct and check synthetic voice, and a consent workflow that should be routine for anyone who records, clones or converts voices. Chapter 20 covers the law in more detail.

Three different technologies

People say "AI voice" to mean three different things, and they behave differently.

Text-to-speech turns written text into spoken audio in a chosen voice. The voice may be a stock voice designed by the vendor, a voice cloned from recordings of a specific person or a voice designed from a description.

Voice cloning is the process of capturing a specific person's voice so that a text-to-speech system or a conversion system can reproduce it. Cloning can be "zero-shot", from a short sample provided at the moment of

generation, or "fine-tuned", from a larger set of recordings used to train or adapt a model.

Voice conversion, sometimes called speech-to-speech, takes a recording of one person speaking or singing and changes the voice so that it sounds like another, while keeping the words, timing and much of the performance. It is the technology behind singing-voice conversion and real-time voice changers.

A fourth category, conversational speech models, combines recognition, language understanding and synthesis in one system that can hold a spoken conversation. It matters to sound engineers mainly as a source of material and as a recording challenge.

How synthetic speech got here

Early speech synthesis joined together small recorded units of speech, a method called concatenative synthesis, or generated speech from hand-built models of the vocal tract. Both sounded mechanical. Statistical parametric synthesis smoothed the joins but sounded muffled.

The neural turn began in 2016 with DeepMind's WaveNet, which generated speech one sample at a time and was rated far more natural than earlier systems. Google's Tacotron 2, published in 2017, paired a network that predicted mel spectrograms from text with a WaveNet vocoder, and that two-stage design, text to spectrogram to waveform, dominated for several years, with faster neural vocoders replacing WaveNet.

The next turn came from treating speech as tokens. Microsoft's VALL-E, published in January 2023, used neural codec tokens and a language-model approach to synthesise speech in the voice of a speaker heard in a prompt of about three seconds. From then on, zero-shot cloning from very short samples was a standard capability. Flow-matching systems such as F5-TTS in 2024 offered similar zero-shot cloning with fast, non-autoregressive generation. Commercial services advanced in parallel: by 2026, ElevenLabs had made its expressive Eleven v3 model generally available, announcing in February 2026 that the general-availability version made 68 percent fewer errors on numbers, symbols and specialised notation than its preview, a vendor's own figure. Open-weight

models proliferated, with permissive licences and multilingual coverage; as an example of the pace, a compact open model released by Zyphra in November 2026 claimed real-time zero-shot cloning in 43 languages on device. Treat specific claims of this kind as dated vendor statements to be tested, not as settled facts.

Voice conversion followed a similar arc. Open-source projects such as so-vits-svc and Retrieval-based Voice Conversion, widely shared from 2023, let hobbyists train a singing-voice model from minutes of a singer's recordings and apply it to another singer's performance. In April 2023 a song called "Heart on My Sleeve", which used AI-generated vocals imitating Drake and The Weeknd, spread widely online before being removed from major platforms after a complaint from their label. That episode made the music industry's concerns about voice imitation concrete. Some artists went the other way: Holly Herndon released a licensed vocal model of her own voice, Holly+, in 2021, and Grimes invited fans in 2023 to release songs using an AI model of her voice in exchange for a share of royalties.

What modern voice systems can do

As of late 2026, the leading systems, commercial and open, typically offer the following, with large variation in quality between them.

Zero-shot cloning from a few seconds to a minute of reference audio, with higher fidelity from longer, cleaner references or from fine-tuning.

Multilingual synthesis, including speaking a language the reference speaker never spoke, usually with an accent shaped by the training data.

Expressive control through natural-language instructions such as "sad, quiet, as if confiding a secret", through bracketed performance tags for things like whispering or laughter in some systems, through reference performances whose prosody the model imitates and through classical controls for speed and pitch.

Pronunciation control through phonetic spellings, custom dictionaries or markup such as SSML, the Speech Synthesis Markup Language.

Streaming output with low latency for interactive use.

Voice design from a text description, producing a new voice that does not belong to any real person.

Speech-to-speech conversion that keeps a performer's timing and emotion while changing the voice, increasingly in real time.

Directing synthetic voice

Getting a good performance from a synthetic voice is closer to directing an actor than to typing text.

Write for the ear. Scripts written for reading often contain long sentences, parentheses and abbreviations that synthetic voices stumble over. Break sentences, spell out numbers and abbreviations the way they should be spoken and mark emphasis where the system allows.

Choose the reference carefully. For cloning, the reference recording sets the voice's tone, the recording quality and often the emotional baseline. A noisy, reverberant or compressed reference will produce a noisy, reverberant or compressed voice. Record references in a dead room, with a good microphone at a consistent distance, in the emotional register you want.

Generate in short sections. Most systems are more reliable on sentences or short paragraphs than on long passages, and short sections make retakes cheap. Generate several takes of important lines and choose, as you would with a human performer.

Edit like dialogue. Synthetic speech still needs editing: tightening pauses, fixing a stressed syllable, matching levels and tone across takes. Treat synthetic takes as raw recordings.

Match the space. Synthetic voices usually sound like a close microphone in a dead room. If the voice must sit in a scene, apply room tone, reverberation and equalisation to place it, as you would with ADR, the studio-recorded dialogue that replaces location sound in film.

How synthetic voices fail

Common failures include mispronunciations, especially of names, technical terms and words borrowed from other languages; wrong stress

on words with different meanings depending on stress; unnatural rhythm in long sentences; flat or inappropriate emotion; artefacts at the start and end of generations, such as clicks or swallowed first syllables; skipped or repeated words in autoregressive systems; drift in timbre over long passages; and accent leakage, where a voice speaking a second language carries an accent from the training data rather than from the speaker.

Check every synthetic line against the script. A practical method is to run speech recognition on the output and compare the transcript with the script automatically, which catches skipped, repeated and substituted words, then listen to every line for pronunciation and emotion. Recognition-based checks miss errors of stress and tone, so they do not replace listening.

Voice conversion in production

Speech-to-speech conversion has a particular strength: it keeps the timing, breath and emotional shape of a real performance. That makes it useful in several professional settings.

In dubbing and localisation, a voice actor performs in the target language and conversion makes the result sound closer to the original actor, with that original actor's consent. In games, a small number of performers can produce many character voices. In music, a songwriter can demo a song in the voice of the singer it is intended for, with their agreement, or a singer can test a harmony arrangement. In accessibility, people who are losing their voice to illness can bank recordings so that a synthetic version of their own voice can speak for them later. Voice banking for people with motor neurone disease is one of the most humane uses of this technology.

Conversion quality depends on the source performance being clean and close in range to the target voice. Converting a deep male voice into a high female voice, or the reverse, is harder than converting between similar voices. Singing conversion is more demanding than speech, because pitch, vibrato and vowel shaping all have to be right.

Singing voices

Singing synthesis and conversion deserve separate mention, because singing makes demands that speech does not. Pitch must be accurate to a few cents and follow the melody with the singer's characteristic scoops, vibrato and ornaments. Vowels are held for long durations, exposing any instability in timbre. Breath and dynamics carry much of the emotion. And a sung line sits in a mix, where small artefacts are either masked or, in a sparse arrangement, cruelly exposed.

Dedicated singing-synthesis software, which sings from notes and lyrics entered in an editor, has existed for many years and improved greatly with neural models; it suits producers who want precise control of each note. Singing-voice conversion, which re-voices a sung performance, suits producers who can sing the part themselves but want a different voice. Music generators produce sung vocals as part of a whole track, with the least control of all.

In every case the consent rules apply with full force, because a singer's voice is the core of their professional identity and income. The episodes of 2023 described above made the industry acutely sensitive to imitation of famous singers, and the licensed platforms announced by music generators in 2025 and 2026 are built around opt-in participation by artists. For a working engineer, the practical rule is simple: a singing voice model of a real person is used only with that person's written agreement, and a designed voice that is not modelled on anyone is the safe default when no such agreement exists.

Consent as a workflow

The technical ease of cloning has made consent the central professional question in voice work. In many places law now requires it, as Chapter 20 explains: Tennessee's ELVIS Act, effective in July 2024, protects a person's voice, including a simulated voice, against unauthorised commercial use; performer union agreements negotiated in the United States from 2023 require informed consent and compensation for digital replicas; the European Union's AI Act requires that deepfake audio be disclosed from August 2026; and other jurisdictions have bills in

progress. Even where the law is unclear, using someone's voice without permission damages trust, invites legal action and is wrong.

Lantern Street treats consent as a workflow with five parts, applied before any cloning, conversion or fine-tuning that involves a real person's voice.

Identify. Confirm that the person giving consent is the person whose voice will be used, or their legally authorised representative. A recording sent by a third party is not consent.

Inform. Explain in plain language what will be done: which system will be used, whether a persistent voice model will be created, where it will be stored, who can use it and for what.

Agree in writing. Record consent in a signed licence that states the scope: the projects or uses covered, the languages, the media, the territory, the duration, the compensation, whether the voice may be used to say things the person has not approved line by line, and how approval of outputs will work. Appendix A includes a template.

Control. Store reference recordings and voice models securely, restrict access to named people, watermark outputs where the tool allows and keep a log of every generation used in a deliverable.

End. Specify how the person can withdraw consent for future uses and what happens to the voice model, including deletion, when the licence ends.

For stock voices and designed voices that belong to no real person, consent questions shift to the vendor's licence terms: check that commercial use is permitted, whether attribution is required and whether the vendor claims any rights in the output.

Security and misuse

Cloned voices are used in fraud, typically in calls impersonating a family member, an executive or a public official. Studios that handle voice data are targets. Keep reference audio and models off shared drives, delete them when projects end, and never upload a client's voice to a service

whose terms you have not read. If a client asks you to clone a voice they "have permission" for, ask to see it.

Detection and watermarking, discussed in Chapter 19, help but do not solve the problem. A detector can say that a recording is probably synthetic; it cannot prove that a recording is genuine.

Synthetic voice in the mix

From a mixing point of view, synthetic voices are dialogue tracks with unusual properties. They are often extremely consistent in level, which can sound unnatural next to recorded voices; a little automation of level and tone restores life. They may have unusual high-frequency content from their vocoders that codecs handle poorly, so check them after encoding. They usually lack room tone, so add matching room tone under edits. And they often lack the small noises of real speech, such as breaths and lip sounds, whose absence listeners notice even when they cannot say why. Some systems add breaths; if not, consider adding them sparingly from the performer's real recordings.

A Lantern Street narration job

A client wants an audiobook narrated in their own voice but cannot spare forty hours in a booth. Lantern Street records three hours of the author reading varied material in the booth, with a signed licence limiting the voice model to this book and its marketing, in English and Spanish, for five years, with approval of every chapter by the author and deletion of the model at the end. The engineers fine-tune a voice model on the recordings, generate chapters in short sections, check every line with automated transcript comparison and listening, and hand-edit pronunciation and pacing. The author records pickups for the passages where the model never sounds right, mostly the most emotional ones. The final audiobook mixes human and synthetic narration, and the distributor's disclosure requirements are met in the metadata and the product description.

The result costs less than a fully recorded audiobook, more than a pure synthetic one, and keeps the author in control of their own voice.

In practice

- Distinguish text-to-speech, cloning and conversion; each fails and is regulated differently.
- Record clean, dead, consistent reference audio in the emotional register you need.
- Generate in short sections, take several versions of important lines and edit synthetic speech like dialogue.
- Check every line with transcript comparison and listening for pronunciation, stress and emotion.
- Never clone or convert a real person's voice without identified, informed, written consent with a defined scope and an end.
- Store voice data and models securely, log generations used and delete models when licences end.

CHAPTER 11

Generative Music and Sound Effects

Generative music is the part of AI audio that most people have heard of and the part that provokes the strongest reactions. To some it is a creative instrument that lets anyone sketch a song. To others it is a machine trained on musicians' work without permission that now floods streaming services with imitation. Both descriptions contain truth. For a sound engineer the practical questions are narrower: what these systems can produce, how to direct them, how to make their output usable in a real production and what rights and disclosure obligations come with it.

Sound effects generation raises fewer cultural arguments and is quietly changing post-production and game audio. It is covered in the second half of this chapter.

From notes to audio

Computers have composed music for as long as there have been computers, but for decades they composed symbols, such as scores or MIDI, which then had to be performed or rendered with instruments. Google's Magenta project, launched in 2016, popularised neural models for melody and performance generation in symbolic form. The leap to generating audio directly came later. OpenAI's Jukebox, published in 2020, generated raw audio in the styles of named artists, including singing, at enormous computational cost and with heavy artefacts. It showed what was possible and what was hard.

From 2023 the pace became extraordinary. Google's MusicLM research, published in January 2023, generated coherent music from text descriptions. Meta's MusicGen, released openly in June 2023 as part of its AudioCraft library, generated music from text and could follow a

supplied melody. Stability AI released Stable Audio in September 2023, a latent diffusion model that generated clips of specified length, and an open-weight Stable Audio Open in 2024. Google DeepMind introduced its Lyria model in late 2023. Consumer services, notably Suno and Udio, made full songs with vocals and lyrics available to anyone with a browser in 2023 and 2024.

By late 2026 the leading systems had reached the following points, according to their developers' own announcements.

Suno released its v6 family on 9 September 2026: a flagship v6 model, an exploratory "v6-wild" model and a faster "v6-mini" available to all users. Suno described the new models as able to edit part of an existing song from a plain-language instruction while keeping the rest, build mashups from several sources, sample and isolate a part and build a new track around it, change a single lyric without regenerating the whole song, and start from text, audio, images or video. Reuters reported that the models were built in partnership with Warner Music Group and BMG, following Warner's settlement and licensing agreement with Suno in November 2025.

Google launched Lyria 3.5 on 29 July 2026 in its Google Flow Music product, describing improvements in musicality, lyrics, vocals and control of tempo and duration, with tracks of up to three minutes. Google's Lyria family also includes Lyria RealTime, designed for interactive, continuous generation, and an open model, Magenta RealTime. Google states that all Lyria output is watermarked with its SynthID technology.

Stability AI's Stable Audio 2.5, announced as a model for enterprise sound production, was described as trained on licensed data, generating tracks of up to three minutes and supporting audio inpainting, the regeneration of a selected section. Universal Music Group announced a strategic alliance with Stability AI in October 2025 to develop professional music-creation tools.

ElevenLabs, known for speech, launched Eleven Music in August 2025 and Music v2 on 26 May 2026, describing better vocals and instrumentation, more languages and composition by sections.

Udio settled with Universal Music Group in October 2025 and with Warner in November 2025, and announced plans for a licensed platform on which users could create with the voices and styles of participating artists. During its transition, existing models remained available under new terms.

Open-weight music models also matured in 2025 and 2026, giving researchers and studios systems they can run and fine-tune locally. Their quality generally trailed the best commercial services but improved quickly.

What generators are good at

Current music generators are good at producing complete, plausible, genre-typical pieces quickly. They are good at style, texture and arrangement conventions, at mood, at production sounds associated with a genre and increasingly at intelligible vocals with coherent lyrics. They are useful for sketches, demos, temporary music for edits, background music where the budget is small, references for a human composer and creative exploration. Real-time systems open up interactive uses, such as music that responds to a player in a game or to a performer on stage.

They are weaker at precise control. Telling a generator to make the snare quieter in bar 17, move the key change two bars later, or have the bass follow the kick pattern exactly is still harder than doing it in a DAW, although editing features in the latest models are aimed squarely at this problem. They can be weak at long-form structure beyond a few minutes, at genres poorly represented in their training data, and at the clarity and separation that a skilled mix engineer would achieve. Their output is usually delivered as a finished stereo mix, sometimes with stems, often at loudness and quality that need further work.

Directing a generator

Treat a generator as a session player who has heard a great deal of music and reads instructions literally.

Describe the music, not the artist. Most services block or discourage prompts naming real artists, and imitation of identifiable artists carries legal and ethical risk. Describe instead the genre, era, tempo, key or mode, instrumentation, production style, mood, structure and vocal character. "A slow 6/8 soul ballad around 70 BPM, warm electric piano, brushed drums, upright bass, a female lead with a husky lower register, intimate and dry, building to a gospel-style choir in the last chorus" will usually get closer than any single adjective.

Specify structure explicitly where the tool allows: intro, verse, chorus, bridge, outro, and their lengths. Write your own lyrics rather than letting the model write them if the words matter to you; it also strengthens your claim to authorship of that part of the work.

Use references you own. Several services accept an audio reference, such as a melody you sang, a chord progression you played or a demo you made. Starting from your own material gives you more control and a clearer rights position than starting from text alone.

Generate many, curate few. Generation is cheap; listening is expensive. Generate a batch, shortlist quickly, then use extension, inpainting and section editing to refine the strongest candidates.

Making generated music production-ready

A generated track is rarely a finished master. Lantern Street treats it as raw material.

Separate it. If the service does not provide stems, use the separation tools in Chapter 9 to split the track. Artefacts from generation and separation combine, so listen carefully, but stems let you mix properly.

Replace weak parts. Generated drums can be replaced or layered with programmed or recorded drums; a generated bass can be doubled by a real bass; a generated lead vocal can be replaced by a human singer while keeping the arrangement. Hybrid productions of this kind are common and, depending on how much human contribution they contain, have a stronger claim to copyright protection, as discussed below.

Mix and master as you would any other material, following Chapters 5 and 6. Watch for problems typical of generated audio: a narrow or unstable stereo image, smeared transients, harsh or fizzy high frequencies from the codec, vocals with slurred consonants and occasional drifts in tempo or tuning.

Check through a codec. Generated audio sometimes has unusual high-frequency content that becomes more obvious after lossy encoding.

Check for similarity. Generators can occasionally reproduce recognisable fragments of training material, particularly melodies or lyrics from well-known songs. Before release, listen critically for familiar melodies and hooks, ask others to listen, and consider running the track through a content identification service if available. A court in Munich found in July 2026 that a music generator had produced recognisable outputs of protected songs, as Chapter 20 describes; the risk is not hypothetical.

Interactive and real-time generation

Most generators produce a fixed clip from a prompt. A different family generates continuously, adjusting as controls change. Google describes Lyria RealTime as designed for real-time, interactive music generation, and its open Magenta RealTime model makes the approach available for experimentation. Systems like these produce a stream of music whose style, instrumentation or intensity can be steered live, for example by a performer turning controls, a dancer's movement or a game's state.

For performers, real-time generators become instruments with unusual properties. They respond to direction rather than to notes, they have a latency measured in fractions of a second to seconds rather than milliseconds, and they can surprise. Those properties suit ambient, improvised and installation work better than tightly timed music. Chapter 16 explains why latency is hard to remove from neural audio systems and what research is doing about it.

For sound engineers working with them, the practical concerns are the familiar ones: capture the generated stream to disk so that a performance can be mixed afterwards, gain-stage it like any live source,

check the terms of use for recorded output and log the model and settings used.

Rights, ownership and disclosure

Three separate questions apply to every generated track.

What does the service allow? Terms of service decide whether you may use output commercially, whether you own it or receive a licence, whether the service keeps rights and what you may not do, such as imitate named artists. Terms differ between free and paid tiers and change over time. Save a copy of the terms in force when you generated a track.

Can anyone own it? In the United States, the Copyright Office's January 2025 report on copyrightability concluded that material generated by AI without sufficient human control over its expressive elements is not protected by copyright, though human selection, arrangement and modification can be, and prompts alone are generally not enough. Other countries take different positions. In practice, the more you write, perform, arrange and edit yourself, the stronger your claim.

What must you disclose? Distributors and streaming services increasingly ask. Spotify launched AI Credits in April 2026, letting artists declare which roles and instruments involved AI, and reported in October 2026 that artists were submitting tens of thousands of such credits a day. The DDEX standard used for music distribution metadata includes fields for declaring AI involvement, described in Chapter 13. Deezer has detected and tagged fully AI-generated albums since June 2025 and excludes them from algorithmic and editorial recommendations. In the European Union, the AI Act's transparency obligations, applying from August 2026, require providers to mark synthetic audio in a machine-readable way and deployers to disclose deepfakes, including audio that falsely appears authentic, as Chapter 19 explains.

The scale of generated music on streaming services makes these questions practical, not abstract. Deezer reported in July 2026 that about 90,000 fully AI-generated tracks were being delivered to it each day, more than half of all new deliveries at the peak in June 2026, while such tracks

accounted for only 1 to 3 percent of streams, and that up to 85 percent of streams on fully AI-generated tracks in 2025 were fraudulent. Spotify reported in October 2026 that it had removed more than 50 million "slop and spam" tracks from recommendations that year. A release that is honestly credited and of good quality is less likely to be swept up in measures aimed at mass-produced spam.

Sound effects generation

Sound effects have always been made by recording, by performing Foley, by synthesis and by editing library sounds. Generative models add a new source: describe a sound and get it, or show a video and get its soundtrack.

Text to sound

Text-to-audio models such as Meta's AudioGen in 2022 and the AudioLDM models in 2023 showed that general sounds could be generated from descriptions. Commercial sound-effects generators followed, from speech companies, music generator companies and established sound-library vendors. Quality varies by category. Ambiences, textures, impacts, whooshes and stylised effects tend to work well. Very specific, physically detailed events, such as a particular car's door closing or a specific mechanical sequence, are harder, and the results can sound generic or physically wrong.

Video to sound

Video-to-audio models generate a soundtrack that follows what happens in a video clip, synchronised to the action. MMAudio, presented at the CVPR conference in 2025, became a widely used open baseline, trained jointly on video, audio and text. Research in 2026, including the EchoFoley and FoleyDirector papers at CVPR 2026, focused on control: letting a sound designer specify which events should make which sounds, when and how, rather than accepting whatever the model decides. In parallel, video generation models began producing synchronised audio natively; Google's Veo 3, announced in May 2025, generated dialogue, effects and ambience alongside video.

For post-production these tools are useful for temporary tracks, for previsualisation, for quickly filling background ambience and for generating starting points that a sound designer then edits and layers. They are not yet a substitute for a Foley artist on a scene where every footstep must match an actor's weight, shoes and mood, and their synchronisation, while impressive, needs checking frame by frame for anything sharp.

Working with generated effects

Generate variations and layer them, as sound designers have always done with recorded sounds. A single generated explosion may sound thin; three layered and processed explosions plus a recorded debris tail can sound huge. Edit generated effects to picture by hand where timing matters. Check sample rate and bandwidth; some generators output at lower rates than professional effects libraries. Keep a record of the tool, prompt and terms for each effect used, so that your effects library can be audited later. The licensing position for generated effects is usually governed entirely by the service's terms, which vary widely.

The footsteps job, revisited

For the game studio's three hundred footsteps from Chapter 1, Lantern Street compared three approaches. Generated footsteps were quick to produce and acceptable on gravel, but on wood and metal they sounded similar to one another and lacked the variation of weight and shoe that the game designers wanted. Recorded Foley sounded best but would have taken days. The final approach recorded about forty footsteps per surface with a Foley artist, then used the game engine's procedural variation, covered in Chapter 15, to vary pitch, timing and layering at runtime, and used generation only to create a handful of unusual surfaces, such as alien metal grating, that nobody could record. That hybrid delivered more variety than three hundred fixed files, with a clear rights position for every source.

In practice

- Describe music by genre, tempo, instrumentation, production and mood, not by naming artists.

- Start from your own melodies, lyrics and demos where you can; it improves control and strengthens authorship.
- Treat generated tracks as raw material: separate, replace weak parts, mix and master properly.
- Check generated music for accidental similarity to existing songs before release.
- Save the terms of service in force for every generation used, and disclose AI involvement in metadata as distributors require.
- Use generated sound effects as layers and starting points; edit to picture by hand where timing matters.

CHAPTER 12

Machines at the Mix Bus

Mixing and mastering are where an engineer's taste is most visible, and they are also where AI assistance has been on sale longest. Automated mastering services have existed since the mid-2010s; assistants that listen to a track and propose an equaliser and compressor setting have been in mainstream plug-in suites for almost as long. What changed between 2023 and 2026 was depth. Assistants learned to work on separated parts of a finished mix, to explain more of their choices and to fit into larger automated pipelines. Research systems learned to produce complete multitrack mixes. Language-model agents began operating audio software directly.

This chapter explains what these systems actually do, where they help and how to keep the decisions that matter in human hands.

A short history of automatic mixing

The idea is older than machine learning. In the 1970s Dan Dugan designed an automatic microphone mixer for speech that shared gain among open microphones so that the total stayed constant, and automixers based on that idea are still used in conferences, broadcast panels and theatre. They solve a well-defined problem with a simple rule: when one person speaks, their microphone opens and the others close down, so the room noise does not build up.

Academic research on automatic music mixing grew from the late 2000s, notably in Joshua Reiss's group at Queen Mary University of London, where researchers built rule-based systems that set levels, panning and equalisation from measured features and from principles drawn from interviews with engineers and analysis of professional mixes. The 2019 book *Intelligent Music Production* by Brecht De Man, Reiss and Ryan Stables summarises that work. Its central lesson is that many mixing

decisions follow widely shared conventions, such as a centred lead vocal, bass and kick, a broadly predictable spectral balance for each genre and a vocal sitting a certain amount above the band, and that conventions can be measured and imitated. What remains hard is the part that is not convention.

Commercial tools brought these ideas to market. Automated online mastering services launched around 2014 and let musicians upload a mix and receive a master in minutes. Plug-in assistants appeared in iZotope's Neutron and Ozone suites from 2016 and 2017 and in other vendors' products. Most early assistants matched a mix to genre-based target curves and loudness values and then set processor parameters to move toward them.

Deep learning added two things. First, models that learn mixing from examples: Christian Steinmetz and colleagues showed in 2021 a "differentiable mixing console" built from neural audio effects that could be trained end to end to produce mixes, and Marco Martínez Ramírez and colleagues reported in 2022 a system, trained on data processed to stand in for unmixed multitracks, whose mixes listeners in their tests rated comparably to human mixes on some material. Second, models that infer processing from audio: style-transfer systems that analyse a reference and estimate the equaliser and compressor settings that would make your track sound more like it. Both are active research areas; the techniques behind them are discussed further in Chapter 16.

What current assistants do

As of late 2026 a typical mixing or mastering assistant does four things.

It analyses. It listens to a section of audio, measures loudness, dynamics, spectral balance and stereo image and often classifies the genre or the instrument on a track. iZotope documents that the Master Assistant in Ozone 12, released in 2025, analyses at least eight seconds of audio before making suggestions.

It compares with a target. The target may be a built-in genre profile, a reference track supplied by the user or a loudness value. Ozone 12's Master Assistant lets the user choose genre targets or custom references,

enable or disable modules, set a target loudness and control the intensity of processing.

It proposes a chain. The assistant chooses processors and settings: equaliser curves, compressor and limiter settings, stereo width and often saturation or dynamic equalisation. Good assistants show the resulting chain in full and let you edit everything.

It may operate on separated parts. Ozone 12's Stem EQ separates a stereo mix into vocals, bass, drums and other inside the plug-in, so that a mastering engineer can, for example, lift the vocal slightly without touching the rest. Neutron 5, the mixing counterpart, includes a Mix Assistant and an Unmask module that identifies where one track masks another and suggests cuts, alongside tools for phase alignment, clipping and density.

Other vendors offer comparable tools, including assistants in DAWs, standalone mastering applications and online services. Their details differ and change with each release; the four functions above are a useful way to compare them.

Agents at the console

A newer development is the language-model agent that operates software rather than processing audio itself. In September 2026, Audiokinetic announced that version 2026.1.3 of Wwise, its game-audio middleware, includes a built-in server for the Model Context Protocol, an open standard that lets AI agents call software tools. Through it, an agent can build and edit a Wwise project's structure, such as buses, containers, events and sound banks, from instructions. Community projects had connected agents to Wwise and to DAWs through their scripting interfaces before that. Chapter 15 returns to agents in game audio.

For mixing, agents are most useful for repetitive, rule-based work: preparing sessions, naming and colouring tracks, routing to buses, inserting standard chains, printing stems, running loudness checks and generating delivery reports. They are least useful, and most risky, when asked to make subjective decisions with no clear target, because they can make many changes quickly, and a confident wrong change repeated across fifty tracks is harder to notice than one wrong fader. Treat an

agent like a new assistant engineer: give it clear conventions, let it do the preparation, check its work and keep the creative decisions yourself.

Where assistants help

Assistants are genuinely useful in several situations.

They give beginners a starting point. A beginner who does not yet know whether a mix is too bright or too muddy can learn by seeing what an assistant changes and listening to why. Turning each suggested module on and off, at matched loudness, is an excellent training exercise.

They speed up routine work for professionals. Many mastering and mixing sessions involve similar first moves. An assistant that sets up a sensible starting chain saves minutes per track, and those minutes add up across an album or a series.

They catch problems. Masking detection, phase checks, resonance detection, loudness and true-peak warnings and mono-compatibility checks help an engineer find problems that tired ears miss.

They make volume work affordable. Podcast networks, music libraries and content producers who need thousands of files to meet consistent technical standards can rely on automated processing with human spot checks.

They provide a second opinion. An engineer who has worked on a mix for hours can run an assistant to see whether it suggests something surprising, then decide whether the surprise is insight or error.

Where assistants mislead

Assistants also mislead in predictable ways.

They pull toward the average. A target derived from many tracks in a genre represents what is typical, not what is best for a particular song. A deliberately dark, lo-fi or unusually dynamic mix will be pushed toward the norm. If the character of a piece depends on departing from convention, an assistant will fight it.

They confuse loud with good. An assistant that applies more limiting or brightness than you would can make its version sound better in a quick

comparison simply because it is louder or brighter. Always compare at matched loudness. Several assistants include a loudness-matched bypass for exactly this reason; use it.

They inherit your room's errors. If your reference is chosen in a room that misleads you, the assistant aims at the wrong target. If you accept a suggestion because it sounds right on your monitors, you accept your monitors' errors.

They misread the material. Genre classifiers can be wrong, particularly for hybrid or unusual music. An assistant that thinks a folk song is pop will treat it as pop.

They hide decisions in complexity. A chain of many modules with dozens of parameters can be hard to evaluate as a whole. A human engineer might achieve a better result with two processors.

They separate imperfectly. Stem-based processing inside a mastering tool can introduce separation artefacts, especially when the stem is changed a lot. Small moves are safe; large ones need careful listening.

A method for working with assistants

Lantern Street uses assistants on most projects, under a simple method.

Decide intent first. Before running any assistant, write a sentence or two about what the track should feel like and what is wrong with it now. "Warmer and closer, vocal slightly more forward, keep the dynamics of the bridge" gives you a standard against which to judge any suggestion.

Choose references deliberately. Use two or three commercial tracks you know well on several systems, close to the intended style and released in a similar format. Level-match them to your mix.

Run the assistant and read its chain. Look at every module and setting before listening. Ask what problem each is solving.

Audit module by module. Bypass each module in turn at matched loudness. Keep only those you can hear improving the track against your intent. Reduce the intensity of the rest.

Check translation. Listen on headphones, small speakers, a phone and in mono, and through a lossy codec, as in Chapter 6.

Record and keep. Save the assistant's suggested settings, your final settings and a note of what you changed and why. It helps with recalls and with explaining your work to clients.

Reference matching in detail

Reference matching deserves a closer look, because it is the feature most assistants share and the one most often misused. A matching tool measures the long-term average spectrum of a reference track and of your mix, then designs an equaliser curve that moves your spectrum toward the reference. More advanced tools also compare dynamics, stereo width and loudness, and some estimate the settings of compressors and other processors that would produce a similar result, an approach researchers call audio effects style transfer.

Matching works best when the reference and your mix share arrangement and instrumentation. Match a sparse acoustic song to a dense rock record and the equaliser will try to add energy where your song has no instruments, raising noise and room sound instead of music. It also works best on long-term averages: a match computed over a whole song is more stable than one computed over a chorus. Apply only part of the suggested curve, typically a fraction of the full correction, and listen.

Matching is also a diagnostic. Seeing that your mix has several decibels more energy around 250 Hz than every reference you trust tells you something about your room or your habits, whether or not you apply the curve. Many engineers use matching tools mainly to see such differences and then fix them by hand at the source, on the tracks that cause them, which usually sounds better than a corrective curve on the whole mix.

Assistants and the client relationship

Clients increasingly ask whether a mix or master was made "by AI". The honest answer is usually that AI-assisted tools were used, under the engineer's direction, in the same way that an engineer uses any other processor. Some clients care deeply about this; others do not. Some distributors and platforms ask about AI in creative roles, such as composition, performance and vocals, rather than in technical

processing. A practical approach is to be open when asked, to note significant AI involvement in the project notes, and to agree in advance with clients who have strong views.

There is also a pricing question. If an assistant does in minutes what took an hour, should a client pay less? Many engineers have answered that clients pay for judgement and accountability as much as for time, and that the time saved goes into more careful listening, more revisions and faster turnaround. That answer only holds if the judgement is real. An engineer who merely accepts an assistant's suggestions is offering the client a service they could buy themselves.

A mastering session

A folk artist brings an album of ten acoustic songs to Lantern Street for mastering. The mastering engineer first listens to the whole album on the main monitors at a moderate level and writes notes on each song. She runs a mastering assistant on the first track with a reference chosen from a well-regarded acoustic record. The assistant proposes a bright equaliser curve, a multiband compressor, a stereo widener and limiting to about −10 LUFS integrated. At matched loudness, the brightness helps slightly, the multiband compressor flattens the guitar's natural swell and the widener makes the voice less focused. She keeps a gentler version of the equaliser, removes the multiband compressor and widener, replaces them with a gentle broadband compressor and sets the limiter for a master around −14 LUFS with peaks at −1 dBTP, which suits the dynamics of the songs and plays at full level on services normalising to that value. She uses the assistant's stem equaliser on one song to lift a slightly buried vocal by a decibel, checking carefully for artefacts. The rest of the album is mastered by hand, using the first track's chain as a template and adjusting each song by ear for continuity across the record.

The assistant saved perhaps half an hour and caught one useful problem. The judgement that made the album sound like itself came from the engineer.

In practice

- Write down the intent of a mix or master before running any assistant.
- Use deliberate, level-matched references, and always compare an assistant's result at matched loudness.
- Audit suggested chains one module at a time and keep only what clearly helps.
- Keep stem-based moves inside mastering tools small, and listen for separation artefacts.
- Let agents do preparation, routing, printing and reporting; keep subjective decisions human.
- Be open with clients about AI-assisted processing and record significant AI involvement in project notes.

PART III

Pipelines in Practice

CHAPTER 13

From Bedroom to First Release

This chapter follows a single song from an idea to a release on streaming services, made mostly in a home studio with help from Lantern Street at two points. It is the path most readers starting out will take, and it shows where AI tools fit naturally, where they are a temptation and what a professional release needs beyond good audio. The artist is a composite: a songwriter-producer called Ana, working in a spare bedroom with a laptop, an audio interface, one condenser microphone, a pair of headphones, small monitors and a MIDI keyboard.

The stages of a release

Every recorded release passes through the same stages, whether made in a bedroom or a major studio: writing and pre-production, arrangement and programming, recording, editing, mixing, mastering, and delivery with metadata. AI tools can help at every stage. The table summarises where Ana uses them and how she checks them.

Stage	Possible AI use	Check
Writing	Lyric brainstorming, chord ideas, a generated sketch to test a groove	Keep the melody and lyrics her own; save prompts and terms
Arrangement	Drum pattern suggestions, generated textures	Replace or rework anything that will be prominent
Recording	None on the monitor path; noise suppression only for the guide vocal	Record clean takes with no processing printed
Editing	Comping aids, pitch and timing correction	Listen to every corrected phrase in context
Mixing	Assistant for a starting chain, masking detection	Audit module by module at matched loudness
Mastering	Mastering assistant; a professional master as comparison	Compare on several systems and through a codec

Stage	Possible AI use	Check
Delivery	Metadata and AI disclosure forms	Disclose accurately; keep records

Writing and pre-production

Ana starts with a chorus melody and a lyric idea recorded as a voice memo. Before opening the DAW, she picks two reference tracks, a contemporary indie-pop song and an older soul record, and writes down what she likes about each: the dry, close vocal of the first and the warm bass of the second.

She uses a general-purpose chatbot to brainstorm rhyme options for one verse line and discards most of them. She also uses a music generator to test whether her chord progression works at a slower tempo with a half-time groove, prompting it with the chords and a description of the feel rather than with artists' names. The result helps her decide on the tempo. She does not use any part of the generated audio in the song. She keeps the prompt, the date and a copy of the service's terms in her project folder anyway, because it costs nothing and might matter if anyone later asks how the song was made.

Her reasons for keeping the melody and lyrics her own are practical as well as artistic. The song is her work; it is what she will register with a songwriters' collecting society, and in many jurisdictions, including the United States as of 2025 guidance, material generated without sufficient human control is not protected by copyright. Human authorship is easiest to demonstrate when it is obvious.

Arrangement and programming

Ana programmes drums with a sample-based drum instrument, using its pattern suggestions as a starting point and then editing every bar so that fills and dynamics follow the song. She plays the bass and keyboard parts on her MIDI keyboard. For an atmospheric texture under the bridge she generates a few seconds of sound from a text-to-audio tool, then pitches, filters and reverses it until it is no longer recognisable as its source. She logs that too.

At this point the song exists as a rough arrangement with a guide vocal. She exports a rough mix and listens on her phone, in the car and on headphones, noting what she wants to change.

Recording at home

Following Chapters 3 and 4, Ana records vocals in the corner of her bedroom with a duvet hung behind her and a portable shield behind the microphone, about 15 centimetres from her mouth with a pop filter. She records at 48 kHz and 24 bits with peaks around −10 dBFS, monitoring through the interface's direct monitoring to avoid latency. She records several full takes and then the difficult lines separately.

She is tempted to record with a real-time noise suppressor on the input, because a neighbour's air conditioner is audible in quiet passages. She resists. Printing noise suppression into the takes would make it permanent. Instead she records clean and deals with the noise later, if it is still audible in the mix, with a light pass of the repair techniques from Chapter 8.

For a short guitar part she visits a friend who plays better than she does and records him through a direct box and a microphone at the same time, giving her the option of a dry and an amplified tone.

Editing

Editing a vocal means choosing the best phrases from several takes, a process called comping, then tidying timing, breaths and pitch. Pitch-correction software has been part of popular music since the late 1990s, when a hard setting became a recognisable effect on Cher's 1998 single "Believe". Current tools detect pitch with learned models and can correct notes transparently or audibly, by choice.

Ana corrects only a few notes that are distractingly off and leaves the small pitch movements that make her voice expressive. She checks every corrected phrase in the context of the full arrangement, not solo, because tuning that sounds natural in solo can sound robotic against the band, and the reverse. She removes distracting breaths but keeps the ones that make phrases feel human.

Mixing

Ana mixes the song herself first. She follows a simple order: balance with faders and panning only, then high-pass filters, corrective equalisation, compression, shared reverb and delay, then automation. She runs a mixing assistant on her vocal to see what it suggests. It proposes a brighter top end and a heavier de-esser than she would choose. At matched loudness she keeps a little of the brightness and less of the de-essing. She uses a masking-detection tool between the bass and the kick to find where they overlap and makes a small cut in the bass around the kick's fundamental.

She then books Lantern Street for a mix session. The mix engineer listens to her rough mix, keeps most of her balance, re-records nothing and spends most of the time on the vocal sound, the low end and automation. He removes the air conditioner noise with a light pass of a discriminative denoiser on the quietest vocal phrases only. The final mix has more depth than hers mostly because of detailed automation and better low-end decisions, which is typical.

Mastering

Ana tries an AI mastering service and a mastering assistant on the final mix, then books a human mastering engineer for comparison. She listens to all three at matched loudness on her headphones, her monitors, her phone and in the car. The automated masters are louder by default; at matched level, one is slightly harsh and the other close to the human master. The human master has a better low end and a more natural vocal.

She chooses the human master for the release. That choice is not universal. Many artists release automated masters successfully, particularly for demos, frequent releases and genres where the automated targets fit. The important thing is that Ana compared them fairly, which most people never do.

The final master measures about −11 LUFS integrated with true peaks at −1 dBTP. On Spotify, with default normalisation to −14 LUFS, it will be

turned down by about 3 dB, which is fine; it was mastered loud because the genre suits it, not to win a loudness contest.

Delivery and metadata

Getting a song onto streaming services usually means using a distributor, a company that delivers audio and metadata to many services at once. Distributors ask for the audio, normally a 16-bit or 24-bit WAV or FLAC at 44.1 kHz or higher, artwork and a substantial amount of information.

Every recording needs an ISRC, the International Standard Recording Code, which identifies that recording uniquely; a release needs a UPC or EAN barcode. Distributors can usually assign both. Credits list the artist, featured artists, songwriters, producers and performers. Splits record how songwriting and master ownership are divided among contributors, which matters for royalties. Songwriters should register their compositions with a collecting society, such as APRA AMCOS in Australia and New Zealand, so that performance and mechanical royalties reach them.

Distribution metadata increasingly includes AI disclosure. The DDEX standards used to send release information from labels and distributors to services include, in recent versions of the Electronic Release Notification standard, a way to state whether a release contains AI-generated content and to describe AI contributions. Spotify launched AI Credits in April 2026, displaying in song credits which roles and instruments artists declared as made with AI. Deezer, as noted in Chapter 11, detects and tags fully AI-generated albums.

Ana's song involved AI in ways that are small: an assistant's suggestions in mixing, a denoiser on a few phrases, a generated texture that was heavily processed and a generated sketch that was not used. She answers the distributor's questions accurately. Services generally aim their AI disclosures at creative roles such as vocals, composition and instrumentation rather than at technical processing, but where a form asks, she answers it. If she had used a generated instrumental or a cloned voice, she would have disclosed it clearly, and in the case of a voice, she would have needed written consent, as Chapter 10 explained.

Avoiding the spam filters

Streaming services have tightened their defences against mass uploads, duplicates and artificial streaming, much of it driven by generated content. Spotify reported in October 2026 that it had removed more than 75 million spammy tracks from the platform altogether and taken more than 50 million slop and spam tracks out of recommendations that year, and that tracks with a high probability of being AI-generated were ten times more likely to fall below its 1,000-stream royalty threshold, in place since 2024. Deezer reported excluding fully AI-generated tracks from algorithmic and editorial recommendations.

These measures are aimed at abuse, not at independent artists, but artists can avoid looking like abuse. Release music at a sustainable pace rather than in bulk. Do not upload near-duplicates or very short tracks to inflate counts. Use accurate artist names and avoid names that imitate other artists. Never buy streams or join schemes promising streams, which services treat as fraud and which can lead to removal. Claim your artist profile on each service; Spotify, for example, opened its Artist Profile Protection feature, which lets artists review releases that list them before they appear on their profile, to all artists in November 2026.

Release and after

Ana schedules her release several weeks ahead so that her distributor can deliver it and so she can pitch it to playlist editors through the services' artist tools. She archives the project: the DAW session, all raw recordings, the final mix and master files, stems, the session notes, the AI log with prompts and terms and the signed agreement with her guitarist friend about his performance and credit. If the song ever needs a remix, a sync licence for a film or an audit of its AI use, everything is in one place.

What the example shows

Three lessons from Ana's single apply to most first releases.

AI saved her time at the edges, in brainstorming, sketching, mixing suggestions and repair, but the song's identity came from her writing,

performance and decisions. That is both artistically satisfying and legally useful.

Fair comparison, at matched loudness on several systems, was the most valuable skill she used. It let her choose between automated and human work on evidence.

The non-audio parts of a release, such as identifiers, credits, splits, registrations, disclosure and archiving, take real effort and decide whether the song earns money and stays out of trouble.

In practice

- Keep the melody, lyrics and main performances human if you want clear authorship and copyright.
- Record clean; do not print AI processing on the way in.
- Compare automated and human mixes and masters at matched loudness on several systems before choosing.
- Get ISRCs, credits and splits right, register compositions with a collecting society and answer AI disclosure questions accurately.
- Avoid patterns that look like spam: bulk uploads, duplicates, very short tracks and bought streams.
- Archive sessions, raw files, stems, agreements and an AI log for every release.

CHAPTER 14

Dialogue, Podcast and Post-Production Sound

Speech is the most demanding material in audio. Listeners forgive a slightly dull cymbal; they do not forgive a word they cannot understand. They are also exquisitely sensitive to anything unnatural in a voice. That combination makes dialogue work, from a two-person podcast to a feature film, the place where AI tools save the most time and where their mistakes are most noticed.

This chapter moves from podcasts, where one person may do everything, to film and television post-production, where large teams divide the work among specialists. The principles are the same: capture well, edit invisibly, keep speech intelligible and deliver to specification.

Podcasts and spoken word

Recording

The best podcast recordings come from a dynamic or close condenser microphone, at a consistent distance, in a quiet and dead room, as Chapters 3 and 4 described. When participants are in different places, the most reliable method is the "double-ender": each person records their own microphone locally while the conversation happens over a call, and the local recordings are combined afterwards. Several remote-recording platforms automate this, recording each participant locally in the browser and uploading the files during or after the session. Their quality is far better than recording the call itself, which suffers from codec artefacts, dropouts and the call software's own noise suppression and echo cancellation.

Ask every remote participant to wear headphones, which prevents the other voices from leaking into their microphone, to sit in the quietest,

softest room available and to stay at the same distance from the microphone. Record a few seconds of silence at the start of each session; it gives repair tools a noise profile and editors room tone.

Editing by transcript

The biggest workflow change in spoken-word production since 2019 has been editing by transcript. Software transcribes the recording, aligns each word with the audio and lets the editor cut, move and delete audio by editing the text. Accurate open speech recognition, such as OpenAI's Whisper models released in 2022, made such transcripts cheap and accurate enough for most purposes, and transcript editing is now built into many podcast and video tools.

Transcript editing is fast, and it changes how editors think: structure decisions become text decisions. It also has traps. Cuts made in text may land in the middle of a breath or sound, so listen to every edit. Removing every "um" and repeated word can make speech sound unnaturally clipped; remove the ones that distract and keep the rhythm of thought. Transcripts are not perfect; names, technical terms and accents produce errors, so do not publish a transcript without checking it.

Repair, levelling and loudness

Podcasts typically need light noise reduction, de-essing, equalisation, compression and levelling between speakers. Chapter 8's method applies: treat lightly, prefer discriminative tools and check against the original. Automated levelling, which rides each speaker's gain toward a consistent loudness, is one of the most useful tools in spoken word, as Chapter 6 noted, but it should not lift breaths, laughs and background noise to speech level.

Deliver to the loudness recommended by the platform, typically around −16 LUFS integrated for stereo podcasts, with true peaks no higher than −1 dBTP, unless a platform specifies otherwise.

Regenerating words

Many podcast tools now offer to fix a misspoken word by generating it in the speaker's voice: you type the correction and the tool synthesises it. Technically this is voice cloning, discussed in Chapter 10, applied to a

single word. It is convenient and, used for a genuine slip such as a wrong date, often harmless. But it changes what a person said, in their voice. Lantern Street's rule is that synthetic corrections in a real person's speech require that person's approval of each correction, and that any correction changing meaning must be approved in writing. In journalism and documentary, generated corrections should not be used at all; a misspoken fact is handled with a clarification or a re-recorded line.

Translation and dubbing

Automated dubbing tools can translate a podcast or video into other languages, voiced in a synthetic version of the original speaker's voice and sometimes with lip movements adjusted to match. The quality of translation and voice varies; a native speaker should review every translated episode before release. Voice cloning for dubbing requires the speaker's consent like any other cloning, and under the European Union's AI Act transparency rules from August 2026, synthetic audio of a real person that would falsely appear authentic must be disclosed. Many producers disclose all AI-dubbed versions as a matter of course.

Film and television post-production

The team and the workflow

Film and television sound is divided among specialists. A production sound mixer records dialogue on set, usually with boom and radio microphones. After the picture edit is locked, the editorial team receives the project, typically as an AAF or OMF file containing the edited production audio, along with all the original recordings.

The dialogue editor cleans and assembles the production dialogue: choosing the best microphone for each line, smoothing transitions between shots with room tone, removing noises and marking lines that cannot be saved. Those lines go to ADR, automated dialogue replacement, where actors re-record them in a studio to picture. The Foley artist performs footsteps, cloth movement and handled props to picture. Sound effects editors build backgrounds and hard effects. The composer and music editor deliver music. Finally, re-recording mixers

balance everything on a dubbing stage, first in premixes for dialogue, effects and music, then in a final mix.

Deliverables

A modern drama or documentary may require many deliverables: a theatrical mix if it plays in cinemas, a near-field mix for home viewing, in 5.1 and stereo and often in Dolby Atmos, separate stems for dialogue, music and effects, and an M&E, music-and-effects, mix with no dialogue so that the programme can be dubbed into other languages. Each comes with a specification. Netflix's published specification, as of 2026, sets dialogue-gated loudness at −27 LKFS with a tolerance of 2 LU and true peaks no higher than −2 dBTP for near-field 5.1 and stereo mixes, describes verifying Atmos mixes through a 5.1 re-render and publishes separate guidelines for preparing dubbing deliverables as ADM Broadcast Wave files. Broadcasters publish their own specifications.

The M&E is often where productions discover problems. If production dialogue was recorded with footsteps, props and ambience in it, removing the dialogue removes those sounds too, and they must be replaced with Foley and effects so that dubbed versions do not feel empty. Planning the M&E from the start saves money later.

Location sound for post

Much of what post-production can achieve depends on what was recorded on set, and AI tools have changed some of the trade-offs that production sound mixers make. A good boom microphone, positioned close to the actors just outside the frame, still gives the most natural dialogue. Radio microphones hidden on costumes capture clean dialogue in noisy places but add clothing rustle and a close, dry tone. Most productions record both on separate tracks, and the dialogue editor chooses or blends them.

Dialogue isolation in post makes noisy locations more forgiving than they used to be, which tempts some productions to accept noise they would once have fixed on set, such as a generator too close or a refrigerator left running. That is a false economy. Isolation works best on recordings that are already reasonably clean, and a scene recorded in heavy noise will either go to ADR or sound processed. The old rules still

apply: switch off what can be switched off, record room tone at every set-up, note problems in the sound report and record wild lines, extra readings of difficult lines without picture, immediately after a noisy take while the actor still remembers the performance. Those wild lines give the dialogue editor genuine alternatives that no model can invent honestly.

Where AI helps in post

Dialogue cleanup. Neural dialogue isolation has become standard in dialogue editing, often saving lines that would otherwise go to ADR. A line recorded under a generator, wind or traffic can frequently be made usable. The cautions from Chapter 8 apply: process conservatively, preserve the original, check for changed words and keep a little natural room tone so that the cleaned line matches its neighbours.

Separation for M&E and archives. For older programmes whose stems are lost, dialogue-music-effects separation, discussed in Chapter 9, can create a usable M&E or let a restorer remix a soundtrack. For new productions, dialogue isolation can help separate production effects from dialogue so that they survive into the M&E. Separation artefacts are less forgiving in an M&E, where the dialogue's absence exposes them, so check carefully.

Transcription and search. Transcripts of every take let editors find alternative readings of a line in seconds, which used to take hours of listening. Search by sound, using the machine-listening models in Chapter 17, lets editors find a specific effect or a take with no background noise.

Temporary tracks. Generated music and effects provide temp tracks for editors and directors before the composer and sound team deliver. Make sure everyone understands that temp material will be replaced, and that generated temp music does not end up in the final mix by accident without its rights being checked.

ADR matching. ADR often sounds different from production sound because it was recorded in a different room with a different microphone. Matching tools, including learned equalisation-matching and convolution with impulse responses captured or estimated from the

location, help place ADR in the scene. Some tools go further and use voice conversion to make ADR recorded by a different performer, or by the same performer on a different day, match the original. That raises consent questions covered below.

Dubbing. AI dubbing pipelines, combining translation, synthetic voices and in some cases visual lip-sync adjustment, are being used for some content, particularly documentaries, corporate video and online content. Premium drama continues to use human dubbing actors, often with AI-assisted tools for timing and matching.

Accessibility. Audio description, the narrated description of on-screen action for blind and visually impaired viewers, is increasingly produced with synthetic voices, which lowers its cost and broadens availability. Dialogue intelligibility, a common viewer complaint, benefits from dialogue-enhancement features at playback that rely on separation.

Performers and consent in post

Film and television dialogue is performance, and performers' rights are strongly protected in collective agreements. In the United States, the SAG-AFTRA agreements negotiated with studios after the 2023 strike require informed consent and compensation before a performer's digital replica is created or used, and a 2025 agreement for video-game performers included consent and disclosure terms for digital replicas. Similar protections are being negotiated in other countries and are part of many individual contracts.

For post-production sound this means that using voice conversion or synthesis to alter or create an actor's lines requires checking the actor's contract and the applicable agreement, obtaining specific consent and often paying. Even an apparently minor change, such as using a synthetic version of an actor's voice to fix a single mispronounced word in a mix, may require consent. Production companies are increasingly specific about this in their delivery requirements; ask before you do it.

A documentary dialogue pass

The documentary producer from Chapter 1 brings Lantern Street the full film: eighty minutes, with interviews, archival footage, location scenes

and a narration recorded by the director. The dialogue editor builds a session from the AAF, then works through the film scene by scene. Most interviews need only light cleanup. Two location scenes recorded in wind are rescued with dialogue isolation at moderate settings, leaving some wind for realism. One line from an interview is unrecoverable; the producer agrees to subtitle it rather than reconstruct it. The archival footage contains music that the production cannot license; separation removes most of it, and a new ambience bed covers the remaining residue. The director's narration, recorded in a spare room at home, is cleaned with a discriminative denoiser and equalised to match a short reference section recorded at Lantern Street.

The re-recording mixer then mixes the film in 5.1 and stereo, checking dialogue-gated loudness against the broadcaster's specification and listening on a small television speaker as well as the main system. The delivery package includes the mixes, stems, an M&E with replacement Foley for scenes where production effects were tied to dialogue, a loudness report and a processing note listing which scenes used AI dialogue isolation and confirming that no generative reconstruction or synthetic voice was used. The broadcaster's compliance team asks for exactly that note.

Intelligibility

The single most common complaint from viewers about film and television sound is that they cannot understand the dialogue. The causes are many: performance styles that favour naturalistic mumbling, wide dynamic range mixes designed for cinemas played on thin television speakers, loud music and effects, and playback devices that process sound in unpredictable ways. Mixers can help by checking mixes on small speakers, keeping dialogue consistent in level, limiting the loudness range of home mixes and avoiding masking of dialogue by music in the presence range. AI tools help at both ends: in production, by rescuing marginal recordings; at playback, by enhancing dialogue. Neither substitutes for clear speech in the mix.

In practice

- Record podcasts as double-enders with headphones on every participant; record room tone at the start.
- Edit by transcript, but listen to every cut and check every published transcript.
- Never generate words in a real person's voice without their approval, and never in journalism or documentary.
- Plan the M&E from the first day of a film or series, and check separation-based M&Es carefully.
- Check performer contracts and collective agreements before using any voice synthesis or conversion on actors' dialogue.
- Deliver a processing note with every post-production job stating where AI tools were used and confirming the absence of generative reconstruction where that matters.

CHAPTER 15

Game and XR Audio

In linear media, a sound happens at a fixed time and the engineer decides exactly how it will sound. In a game or an extended-reality experience, the player decides when things happen, where they stand, which way they face and what they do next. The sound designer cannot mix a moment; they design a system that will mix itself, thousands of times, in situations nobody has heard in advance. That makes interactive audio a natural home for procedural and generative techniques, and it is one of the places where AI is changing workflows quickly, from asset creation to agents that build audio projects.

This chapter explains how interactive audio works, where AI fits and what constraints keep it honest.

How game audio works

Events, not files

A game does not play files directly. Game code raises events, such as "footstep", "door open", "enemy alert" or "enter cave", and an audio system decides what to play in response. A footstep event might choose among several recordings at random, vary pitch and volume slightly, pick a different set depending on the surface and the character's speed, and apply a reverb depending on the room. Parameters passed from the game, such as speed, health, distance, time of day or intensity of combat, continuously change the sound.

Two middleware systems dominate commercial game audio: Audiokinetic's Wwise and Firelight Technologies' FMOD. Both let sound designers build these behaviours in a dedicated authoring tool, then package the result into banks that the game loads. Game engines also offer native audio systems; Unreal Engine 5 includes MetaSounds, a

node-based system for real-time procedural sound design, and Unity has its own audio tools.

Variation and repetition

The enemy of game audio is repetition. A player who hears the same footstep a thousand times will notice. Designers fight it with variation: several recordings per sound, random selection that avoids immediate repeats, small random changes in pitch, volume and filtering, layering of components such as heel, toe and surface debris, and parameters that change the sound with context. Good variation from a small set of well-recorded assets usually beats a large set of mediocre ones, and it uses less memory.

Budgets

Games run on fixed hardware with limited memory, processing and storage. The audio system has a budget for how many voices can play at once, how much memory sound banks can use and how much processor time effects can take. Sounds are streamed from storage or loaded into memory depending on how often they are needed. Mobile platforms have tighter budgets than consoles or PCs. Every creative decision has a cost in this budget, which is one reason heavy neural processing at runtime is still rare in games.

The interactive mix

Games mix themselves through rules. Priorities decide which sounds survive when too many play at once. Ducking lowers music and ambience when dialogue plays. Mix states change the balance for menus, cut scenes, combat or exploration. High dynamic range systems choose which sounds to emphasise based on their loudness relative to everything else playing. The sound designer's job is to set rules that produce a clear, exciting mix in situations they cannot predict, then test them in play.

Space in interactive audio

Spatial audio matters more in games than in almost any other medium, because players use sound to locate threats and navigate.

Basic spatialisation pans sounds according to their position relative to the listener and attenuates them with distance using designed curves. Occlusion and obstruction filter and reduce sounds blocked by walls or objects. Reverb zones apply different reverberation in different spaces. More advanced systems simulate sound propagation through the level geometry, calculating how sound travels around corners and through openings, either in real time or precomputed. Valve's Steam Audio, Meta's XR Audio SDK and Microsoft's Project Acoustics are examples of tools that provide physically based propagation and binaural rendering.

For headphone listeners, binaural rendering with head-related transfer functions, discussed in Chapter 18, lets sounds appear above, below and behind the player. In virtual and augmented reality, head tracking makes spatial audio essential: when the player turns their head, sounds must stay fixed in the world, with low enough latency between head movement and sound update that the illusion holds. Six-degrees-of-freedom experiences, in which the player can move as well as turn, require sounds to change correctly with position, which is harder than rendering a fixed 360-degree recording.

Procedural audio

Procedural audio generates sound at runtime from models rather than playing recordings. Andy Farnell's 2010 book *Designing Sound* made the case that many sounds, such as wind, rain, fire, engines and impacts, can be synthesised from simple physical or signal models with parameters that follow the game. Procedural sound can vary endlessly, respond continuously to gameplay and use little memory. Its weakness has been realism: synthesised sounds often sound synthetic.

MetaSounds and similar tools made procedural design more accessible. Neural synthesis, discussed in Chapter 16, is beginning to close the realism gap by learning models of real sounds that can be controlled with parameters at runtime. For now, most games combine recorded assets with procedural variation and some purely procedural sounds where they suit.

Where AI fits in game audio

Asset creation

Generative tools, discussed in Chapter 11, help create sound effects, ambiences and music during development. They are most useful for prototyping, for unusual or fantastical sounds and for large volumes of background variation. As in film, they are less reliable for sounds that must match specific, detailed actions, and every generated asset needs a recorded rights position: which tool, which terms, which prompt.

Dialogue and voice

Games can contain enormous amounts of dialogue, often recorded in many languages. Synthetic voice is used for placeholder dialogue during development, so that writers and designers can hear scenes long before actors record them, and in some games for final dialogue, particularly in smaller productions and for lines generated at runtime. Some games experiment with characters whose lines are written at runtime by language models and voiced by synthetic voices, which raises new problems: content cannot be fully reviewed in advance, and a synthetic voice based on a real actor might be made to say anything.

Performer agreements are central here. Video-game performers in the United States went on strike from 2024 into 2025 over protections against AI, and the agreement that ended the strike included consent and disclosure requirements for creating and using digital replicas of performers. Studios elsewhere are negotiating similar terms. A game audio team should know exactly what each performer agreed to, keep a record and never extend a voice beyond that.

Adaptive and generative music

Game music has long been adaptive: composers write layered stems and transitions that the game mixes according to the action. Real-time generative music systems, such as Google's Lyria RealTime, extend this to music generated continuously in response to parameters. For most commercial games, composed adaptive music remains standard, because it gives composers control and because generated music must

still be reviewed for quality and rights. Generative systems are more common in experimental and creative-tool contexts.

Mixing and analysis

Machine-listening models, discussed in Chapter 17, can analyse recordings of gameplay to find mix problems automatically, such as dialogue masked by music, sudden loudness jumps or missing sounds, across many hours of automated play. That kind of analysis scales quality assurance in a way human listening cannot, though human playtesting remains the final check.

Agents in the project

In September 2026 Audiokinetic announced that Wwise 2026.1.3 includes a built-in server for the Model Context Protocol, the open standard through which AI agents call software tools. Through it, an agent connected to Wwise can, with the user's permission, create and edit project structure such as buses, containers, sounds, events and sound banks, query project data and operate through Wwise's existing authoring API. Audiokinetic's announcement described connecting an agent to Wwise and to a game engine at the same time so that it could assemble a complete sound design setup from instructions, and recommended that teams write an instructions file for agents describing naming conventions, terminology, project context and engine integration. Community-built connectors for Wwise had appeared earlier.

This is a significant change in how game audio work can be organised. Much of the labour in a large audio project is structural: importing hundreds of assets, building containers with correct randomisation and naming, wiring events and setting up banks. Agents can do that labour quickly. The risks are the familiar ones from Chapter 12. An agent can make many wrong changes quickly, so work under version control, review the agent's changes as you would a colleague's, keep the conventions file current and keep creative decisions, such as how a sound should feel, with people.

Recording and localising game dialogue

Game dialogue is recorded in large sessions, often thousands of lines per character, in many languages. Each line is a separate file that the game triggers, so consistency matters more than in linear media: a line recorded on Tuesday must sit beside a line recorded six months later without the player hearing a change in microphone, distance or room. Studios fix microphone, position, preamp settings and processing for each character across sessions and keep reference recordings to compare against.

Localisation multiplies the work. A game released in a dozen languages may need a dozen casts, each recording lines that must fit the timing of animations and cut scenes, sometimes without seeing them. File naming and metadata systems keep the lines organised, and middleware manages which language's files are loaded. AI tools help here in clearly bounded ways: speech recognition checks recorded lines against scripts automatically, alignment tools check timing against the original language, loudness tools normalise thousands of files consistently and machine translation helps writers draft, though professional translators and native-speaking directors remain essential for quality. Synthetic voices are used for placeholder lines in all languages during development, which lets designers test timing long before recording, provided everyone tracks which files are placeholders so that none ship by accident.

Capturing for XR

Extended-reality projects often need spatial recordings: ambisonic microphone arrays for environments, binaural dummy-head recordings for fixed-perspective experiences and close mono recordings of individual sources that the engine places as objects. Ambisonic ambiences recorded on location, decoded and rotated by the engine as the user turns, give a sense of place that mono or stereo beds cannot. Individual sources are best recorded dry and close so that the engine's spatialisation and propagation can place them convincingly; a source recorded with its own room reverberation will sound wrong when the engine adds a different room. As in all spatial work, test on the headsets

and headphones the audience will use, and with the engine's actual rendering, because spatial impressions change with the renderer.

Testing and accessibility

Interactive audio can only be judged in play. Test on every target platform, with headphones and speakers, at different volume settings and with the game's own audio options. Listen for repetition, for sounds that cut out when voice limits are reached, for mix problems in crowded moments and for spatial errors.

Accessibility matters in games more than in most media because sound carries gameplay information. Subtitles with speaker names, visual indicators for important sounds, separate volume controls for dialogue, music and effects and options to reduce dynamic range all help players with hearing loss or who play without sound. Machine-listening tools can help identify which sounds need visual equivalents.

The footsteps, in the engine

Back to the game studio's footsteps. Lantern Street delivers about forty recorded footsteps per surface, edited into heel and toe components, plus a few generated surfaces for alien materials. In Wwise, the studio's audio designer builds switch containers by surface, random containers for each surface with avoid-repeat settings, small random pitch and volume variation, and a parameter linking the character's speed to the choice of walk or run layers and to a gentle change in brightness. A layer of debris sounds is added on gravel and metal. An agent connected through the new interface handles the import, naming and container building from a written specification in an hour, and the designer then reviews every container, tunes the randomisation by ear in the game and adjusts the attenuation curves for the camera distance. In play, three hundred variations would have used more memory and sounded less alive than this system does.

Voice pipelines and localisation

Dialogue in a large game is a production in its own right. A role-playing game may contain tens of thousands of lines, recorded by dozens of

performers and localised into many languages. Each line has an identifier that links the script, the recording, its translations, its subtitles and the event that plays it. The audio team's job is to keep all of those in step as the script changes, which it does constantly during development.

AI tools fit into several points of that pipeline. Synthetic placeholder voices let designers hear new or changed lines immediately, long before a recording session, and they can be regenerated automatically whenever the script changes. Speech recognition checks recorded lines against the script and flags missing words or wrong takes. Loudness and quality checks run across every file in every language. Lip-sync tools generate facial animation from recorded or synthetic speech. Some studios use voice conversion to keep a character's voice consistent across languages, with the consent of both the original and the localisation performers.

Three rules keep such a pipeline sound. First, mark every synthetic placeholder clearly in file names and metadata, so that none ships by accident. Second, store each performer's consent terms alongside their recordings in the dialogue database, so that anyone generating or converting a line can see what is permitted. Third, treat every line generated at runtime as unreviewed content: constrain what the language model can say, filter its output and design the synthetic voice so that it is not a replica of any performer unless that performer has explicitly agreed to real-time generation, which the 2025 United States video-game agreement treats as a distinct use requiring specific consent.

Notes for XR

Extended reality adds constraints of its own. Head-tracked audio must update quickly when the head moves; delays between movement and sound break the illusion and can contribute to discomfort. Sounds that should be part of the world must stay fixed in the world, while interface sounds and some narration may be "head-locked", moving with the listener. Augmented-reality experiences mix virtual sound with the real environment, so their sounds must be convincing against real acoustics, and machine-listening models that estimate the real room's reverberation are being used to match virtual sounds to it. Battery-

powered headsets have tight processing and power budgets, which limits heavy neural processing on the device.

In practice

- Design systems, not sounds: events, variation, parameters and mix rules.
- Prefer a small set of excellent recordings with procedural variation over large sets of generic assets.
- Respect platform budgets; heavy neural processing at runtime is rarely affordable yet.
- Use spatial and propagation tools so that players can locate sounds, and keep head-tracking latency low in XR.
- Know exactly what each voice performer agreed to, and never extend a voice beyond it.
- Let agents build project structure under version control and review; keep creative and mix decisions human and test in play.

PART IV

The Frontier

CHAPTER 16

Neural DSP and Real-Time AI

The tools in Part II mostly run offline: you give them a file, wait and get a file back. The frontier this chapter covers is AI that runs inside the signal path in real time, as an instrument, an effect or a processor a performer plays through. Real-time use imposes the hardest constraints in audio. A guitarist will reject an amp model that adds 20 milliseconds of delay however good it sounds. A singer will reject a voice effect that occasionally glitches. A mastering engineer will reject a processor that sounds different every time the session is reopened.

Meeting those constraints has pushed research toward hybrids of neural networks and classical digital signal processing, or DSP. The resulting field, sometimes called neural DSP or differentiable DSP, is among the most interesting in audio, because it combines what each approach does best: the learned richness of neural networks and the efficiency, stability and interpretability of filters, oscillators and delay lines.

Why hybrids

A pure neural network that generates or processes audio sample by sample is flexible but expensive, hard to interpret and prone to strange failures outside its training data. A classical DSP processor is cheap, predictable and well understood, but it only does what its designer built it to do. A hybrid uses a neural network to control or adjust classical components, and trains the whole system end to end. If the classical components are differentiable, meaning that the effect of changing each parameter on the output can be calculated, then standard machine-learning training can tune the network and the DSP parameters together to match a target.

The advantages are practical. Hybrids can be small enough to run on an ordinary processor in real time. Their DSP components give them a

built-in bias toward physically plausible output, such as harmonic tones, stable filters and smooth reverberation, so they need less training data. Their parameters can often be read and edited by people. And they fail more gracefully, because the classical structure limits how wrong they can sound.

DDSP

The idea reached a wide audience in 2020 when Jesse Engel and colleagues at Google's Magenta team published "DDSP: Differentiable Digital Signal Processing" at the ICLR conference. Their model represented a monophonic instrument as a bank of harmonic oscillators plus filtered noise, with a reverb. A neural network looked at features of the input, chiefly its pitch and loudness over time, and predicted the amplitudes of the harmonics and the shape of the noise filter frame by frame. Because the oscillators, filters and reverb were all differentiable, the network could be trained on recordings of a single instrument, such as a violin, from a few minutes of audio.

The striking application was timbre transfer. Feed the trained violin model the pitch and loudness of someone singing, and it plays the melody as a violin, following the singer's phrasing. Google released a public demonstration of the idea, and DDSP became a reference point for the field. Its limitations were equally instructive: it modelled monophonic, harmonic sounds well and polyphonic or noisy sounds less well, and its quality depended on accurate pitch tracking.

DDSP opened a research programme in differentiable versions of every classical audio component: filters, including time-varying all-pole filters; compressors; equalisers; reverberation networks; and synthesiser modules. Libraries collecting differentiable effects appeared, and in 2025 Marco Comunità, Christian Steinmetz and Joshua Reiss at Queen Mary University of London released NablAFx, an open framework for building, training and comparing "black-box" neural models of audio effects and "grey-box" models assembled from differentiable DSP blocks with neural controllers. Its authors describe the trade-off directly: black-box models have achieved the best accuracy, while grey-box models attract interest for interpretability and potential efficiency.

Neural synthesis: RAVE and its descendants

A different line of work trains neural networks to synthesise general audio in real time. RAVE, the Realtime Audio Variational autoEncoder, published by Antoine Caillon and Philippe Esling at IRCAM in 2021, compresses audio into a compact latent representation and reconstructs it with a decoder that runs faster than real time on a laptop processor, using a multiband decomposition to keep the computation manageable. Musicians trained RAVE models on their own recordings, such as a drum kit, a voice or a field recording, and then played them by feeding in other audio or by moving through the latent space with controllers. RAVE became a staple of experimental music, used through plug-ins and through objects for visual programming environments such as Max and Pure Data.

RAVE's weakness for performance was latency. Neural audio synthesis models often introduce latencies of several hundred milliseconds, as a 2025 paper by Franco Caspe, Jordie Shier, Mark Sandler, Charalampos Saitis and Andrew McPherson pointed out. Their study, accepted for the Journal of the Audio Engineering Society, analysed where latency comes from in audio-to-audio neural synthesis: buffering delay from the size of audio blocks, cumulative delay through the network's layers, representation delay from how the model encodes time, data-dependent latency and jitter in when events appear. They then redesigned RAVE step by step to reduce each source, producing a model they called BRAVE that is low in latency, follows pitch and loudness better and keeps similar timbre-transfer abilities, and they demonstrated it in a proof-of-concept plug-in for instrument signals. The work matters less for the specific model than for its method: latency is an architectural property, and it has to be designed in from the start.

Getting models into plug-ins

A trained model in a research framework is not a plug-in a musician can load in a DAW. Bridging that gap has become a field of its own.

Neutone, a company and open-source project, provides a software development kit that wraps PyTorch models into a host plug-in for standard formats such as VST3 and Audio Units. Its documentation

describes handling the plumbing that real-time hosts demand, such as different buffer sizes, sample-rate conversion, parameter controls and latency reporting so that the DAW can compensate for the delay, and notes that RAVE models typically add about one buffer of 2,048 samples of delay. That is about 43 milliseconds at 48 kHz, acceptable for sound design and mixing but too much for playing live.

RTNeural, an open-source inference library published by Jatin Chowdhury in 2021, provides efficient implementations of common neural network layers designed for real-time audio constraints: no memory allocation in the audio thread, predictable timing and processor-friendly arithmetic. Many neural plug-ins use it or similar libraries.

A 2026 demonstration by Facundo Franchino and Sebastian Schlecht took another route. Their compiler, ADAC, takes a trained differentiable audio processor, lowers it to an intermediate representation and emits code in FAUST, a long-established language for real-time DSP whose toolchain can compile to many plug-in and hardware targets. They reported that the generated processor's impulse response matched the trained model to within floating-point rounding, that training could be made audible by swapping the model into a running plug-in after each training step and that the exported plug-in carried a stability certificate checked before it was built. Their demonstration used feedback delay networks, a classic reverberation structure. Approaches like this point toward a future in which learned processors are designed in machine-learning tools and shipped as ordinary, efficient, verifiable DSP.

Neural amp and effect modelling

The most commercially successful real-time neural audio so far is the modelling of guitar amplifiers and effects. Amplifiers and pedals are nonlinear systems with memory, hard to model accurately with hand-designed circuits. From around 2019, researchers including Alec Wright, Eero-Pekka Damskägg and Vesa Välimäki at Aalto University showed that recurrent and convolutional networks trained on recordings of an amplifier's input and output could imitate it convincingly in real time. Commercial products followed, from hardware modelling units to plug-

ins that let players capture their own amplifiers by playing test signals through them. The open-source Neural Amp Modeler, created by Steven Atkinson, became popular from 2023 and spawned a large community library of shared captures.

These systems are black-box models in the NablAFx sense: they learn the input-output behaviour without modelling the circuit. They are typically trained at one setting of the amplifier's controls, or a few, and imitating the effect of turning a knob requires conditioning the model on control values, which needs more data. Grey-box approaches, which combine learned nonlinearities with classical filters for the tone stack, are one answer.

For engineers, neural amp models are now ordinary tools: they track well at low latency on standard processors. The same techniques are spreading to compressors, tape machines, analogue equalisers and other studio hardware.

Real-time AI elsewhere in the chain

Real-time neural processing is already common in places where it is not called an effect.

Calls and streaming. Noise suppression and echo cancellation in video-call and streaming software rely on small neural networks running in real time on phones and laptops. Many of these trace their lineage to designs like RNNoise and the Deep Noise Suppression challenges discussed in Chapter 8.

Hearing technology. Hearing aids and earbuds increasingly use on-device neural networks to separate speech from noise. Manufacturers announced hearing aids with dedicated neural-network processors from 2024, aimed at speech in noisy places. These systems must run within strict power budgets and with latencies of a few milliseconds, because wearers hear the processed sound mixed with sound arriving directly.

Live voice. Real-time voice conversion lets streamers and performers change their voice live, with latencies low enough for speech. The consent rules from Chapter 10 apply in full: converting your voice into a recognisable real person's in real time is impersonation.

Live sound. Neural feedback suppression, automatic microphone mixing descended from Dugan's design and real-time dialogue enhancement for broadcast are in use, mostly under conservative settings because failures happen in front of an audience.

Latency budgets

Working with real-time AI means managing a latency budget. A useful way to think about it is to list everything that adds delay between a sound entering the system and a listener hearing the result.

Converters add a fraction of a millisecond to a millisecond or two. The audio buffer adds its length, and round-trip monitoring involves at least two buffers. Each plug-in may add latency for lookahead or block processing, which the DAW compensates for in playback but not in live monitoring. A neural model adds its own buffering and any lookahead it needs, which can be the largest item. A graphics processor, if used, adds transfer time. Network round trips for cloud processing add tens to hundreds of milliseconds and are unsuitable for performance.

Different uses tolerate different totals. Musicians monitoring their own instrument or voice are sensitive to a few milliseconds and generally uncomfortable above about 10. Speech in a call becomes awkward when one-way delay reaches a few hundred milliseconds. Broadcast and streaming can absorb more if audio and video stay in sync. Offline mixing tolerates anything, as long as the DAW compensates correctly.

Jitter, variation in delay from block to block, matters as much as average latency for performers, because it makes timing unpredictable. Models whose processing time varies with input content can produce jitter even when average latency is low. When you test a neural effect for live use, measure both.

Stability and repeatability

Professional tools must behave the same way every time. Neural processors can be surprisingly hard to make repeatable. Some generate randomness internally. Some produce slightly different results on different processors or with different library versions. Some recurrent

models change their output depending on what they processed before, so playing a session from a different start point changes the sound. Some become unstable on input unlike anything in their training data, producing bursts of noise or silence.

Ask of any neural processor whether it is deterministic, whether it has internal state that depends on playback position and how it behaves on silence, on full-scale signals and on unusual material. Render critical tracks to audio once you are happy with them, so that a future plug-in update cannot change a finished mix. The stability certificates in work like ADAC are a sign of the direction research is taking: proving properties of learned processors rather than hoping for them.

Neural reverberation and acoustic modelling

Reverberation is a natural target for hybrid methods because its classical structures, such as feedback delay networks and convolution, are well understood and efficient, while real rooms are complex. Differentiable versions of feedback delay networks can be trained to match the measured impulse response of a real room, producing an efficient algorithmic reverb that sounds like that room and can be adjusted with familiar controls. The ADAC work described above used exactly this case to demonstrate compiling a trained processor into a plug-in. Related research estimates a room's impulse response from a reverberant recording, which would let a post-production engineer match studio-recorded ADR to a location from the production sound alone, and generates reverberation consistent with the space seen in a video. These techniques are moving from research into tools, and they illustrate the general pattern of this chapter: learning is used to fit classical, controllable structures rather than to replace them.

A Lantern Street experiment

A sound artist books Lantern Street to build a live set in which her voice controls textures modelled on recordings of a harbour: rope creaks, hull groans and water. The engineers train a RAVE-family model on an hour of the harbour recordings and load it into the DAW through a wrapper plug-in. Offline, it is beautiful. Live, the latency of roughly a buffer of

2,048 samples is audible as a lag between her voice and the response. Because her performance is slow and textural, she decides the lag is acceptable and even musical, and the set uses the model on a separate output from her direct voice, which stays latency-free. For a percussive piece in the same set, the lag is not acceptable, so the engineers use a classical granular processor instead. The decision is made by listening in the actual performance setup, not by reading specifications.

In practice

- Prefer hybrid, grey-box neural processors where stability, efficiency and interpretability matter.
- Measure latency and jitter of any neural effect in the actual live setup before using it for performance.
- Keep high-latency neural processing off performers' monitor paths, or use it where lag suits the music.
- Check whether neural processors are deterministic and stateful; render finished tracks to audio.
- Test neural processors on silence, full-scale signals and unusual material before trusting them.
- Watch research on compiling learned processors into ordinary DSP; it is likely to make neural effects cheaper and more reliable.

CHAPTER 17

Machine Listening

Most of this book is about machines that change or make sound. This chapter is about machines that listen: systems that detect, classify, transcribe, measure, describe and answer questions about audio. Machine listening is less visible than generation, but it underpins much of what the rest of the toolkit does. Loudness meters detect dialogue. Transcript editors recognise speech. Mastering assistants classify genre. Streaming services identify songs and detect generated tracks. And since 2023, audio-language models have begun to describe what they hear in ordinary language and reason about it.

For a sound engineer, machine listening is most useful as a tireless assistant that can find things in hours of audio, flag problems for checking and attach searchable descriptions to archives. It is least useful when treated as a judge of quality.

A short history

Automatic speech recognition has a long research history, from small-vocabulary recognisers in the mid-twentieth century to statistical systems based on hidden Markov models from the 1980s and neural systems from the 2010s. Music information retrieval, the field that develops beat tracking, key and chord recognition, melody extraction and music recommendation, formed its own research community around the turn of the century, with an annual conference, ISMIR, from 2000. Audio fingerprinting, which identifies a recording from a short sample even in noise, became widely known through Shazam, whose approach Avery Wang described in a 2003 paper; fingerprinting now underlies content identification systems on video and music platforms.

General sound recognition, the task of recognising sounds of any kind, gained momentum with large datasets. Google's AudioSet, released in

2017, provided around two million ten-second clips from online videos labelled with hundreds of sound categories arranged in an ontology, from “dog” and “siren” to “electric guitar” and “applause”. The annual DCASE challenges, on detection and classification of acoustic scenes and events, which began in 2013, gave researchers shared tasks and evaluations.

The neural era produced a series of influential models. PANNs, pretrained audio neural networks published in 2020, provided strong general-purpose sound classifiers trained on AudioSet. The Audio Spectrogram Transformer of 2021 applied vision-transformer techniques to spectrograms. Self-supervised models, such as BEATs from Microsoft in 2022, learned general audio representations from unlabelled audio. Contrastive language-audio pretraining, known as CLAP, introduced by Microsoft researchers in 2022 and developed in open form by the LAION community, trained audio and text encoders together so that a sound and its description map to nearby points in a shared space. CLAP made it possible to search audio with free text and to classify sounds into categories that were never part of training, simply by describing them.

In speech, OpenAI’s Whisper, released in 2022 and trained on a very large amount of weakly supervised multilingual audio, made accurate transcription and translation widely available as open weights. In music, Spotify’s Basic Pitch, released in 2022, offered lightweight open transcription of audio into notes.

Audio-language models

From 2023, researchers connected audio encoders to large language models, producing systems that accept audio and text and respond in text. Early examples included Pengi, LTU (“Listen, Think and Understand”), SALMONN and Alibaba’s Qwen-Audio family. General-purpose multimodal models from major AI developers added audio input during 2024 and 2025. NVIDIA’s Audio Flamingo series continued the line in open form; NVIDIA describes Audio Flamingo 3 as a fully open model for understanding and reasoning over speech, sound and music, with support for long audio inputs and multi-turn conversation.

These systems can do things that earlier classifiers could not. Asked "what is happening in this recording?", they can describe a scene: rain on a tin roof, a car passing, a man speaking in the distance. Asked "is there distortion on the vocal in the second chorus?" or "what instrument plays the melody after the bridge?", they can sometimes answer correctly. They can summarise a podcast, identify a speaker's emotional tone, describe a mix in words and combine listening with reasoning, for example estimating whether a sound effect fits a described scene.

How well do they listen?

Benchmarks have tried to measure this. MMAU, presented at the ICLR conference in 2025, collects about ten thousand audio clips with human-written questions covering speech, environmental sounds and music, testing 27 distinct skills of perception and reasoning. A harder successor, MMAU-Pro, published in 2026, extends the test to long recordings, spatial audio, multiple audio inputs and open-ended questions across 49 skills; its authors reported that a leading commercial model of the time scored around 59 percent. Leaderboards for the original MMAU have reported much higher scores for the best systems as of late 2026. The gap between the two benchmarks is the important information: models do well on short, single-clip, multiple-choice questions and much less well on long, complex, open-ended listening, which is closer to what engineers need.

Audio-language models also share the familiar failure of language models: they can answer confidently and wrongly. They may describe a sound that is not present because it usually appears in similar scenes, misjudge timing, miss quiet events or invent details of a mix. Treat their descriptions as hypotheses to check.

The research frontier

The DCASE 2026 challenge gives a useful snapshot of where the research community's attention was as of 2026. Its tasks included classifying heterogeneous audio, detecting anomalous sounds in machine monitoring in noisy conditions, acoustic imaging and localisation of sound events, spatial semantic segmentation of sound scenes, answering questions that depend on audio, retrieving moments from long

recordings and incremental learning of new sound classes without forgetting old ones. Two themes stand out: understanding long recordings, rather than ten-second clips, and understanding where sounds are in space, not only what they are.

What machine listening is good for in a studio

Search

The simplest and most valuable use is search. A sound-effects library with tens of thousands of files is only as useful as its metadata, and metadata is often incomplete or inconsistent. Embedding every file with a CLAP-style model makes the library searchable by description, such as "heavy wooden door slam in a large hall", and by similarity, such as "more sounds like this one". Many professional sound-library tools now offer this. The same applies to archives of interviews, field recordings, music stems and session takes. Transcripts make spoken archives searchable by word and speaker.

Tagging and metadata

Classifiers can tag genre, mood, instrumentation, tempo, key, vocal presence, language and explicit content. Music libraries and broadcasters use such tags at scale. They are useful starting points and often wrong in detail, especially for hybrid genres, so check the tags that matter for licensing, such as explicit-content and vocal flags.

Quality control

Machine listening can flag technical problems across hours of material: clicks, dropouts, digital silence, clipping, phase cancellation in mono, hum, sudden loudness jumps, missing channels, dialogue masked by music and, with speech recognition, mismatches between a script and a recorded or synthetic performance, as Chapter 10 described. For delivery work, an automated pass that produces a list of time-stamped candidates for a human to review can save hours. The human still decides what counts as a problem: a click may be a deliberate percussive sound and a silence may be dramatic.

Transcription and alignment

Speech recognition supports transcript editing, captions, subtitles, translation, search and script comparison. Forced alignment, matching a known script to a recording word by word, supports dialogue editing, lip-sync and automated dialogue replacement. Speaker diarisation, determining who spoke when, helps in multi-person recordings. All have error rates that rise with noise, overlapping speech, accents and specialised vocabulary.

Music analysis

Beat and downbeat tracking, tempo estimation, key and chord recognition, melody extraction and note transcription support editing, remixing, education and the conditioning of generative models. Basic transcription of a separated bass line or melody into notes, as in Chapter 9, is now routine.

Detection of synthetic audio

Machine listening is also used to detect generated audio, as Chapter 19 explains. Deezer reported in 2026 that its detector identifies artefacts left by generative models, often inaudible to people, and can even indicate which generator was used. Such detectors are useful in aggregate and imperfect for any single file.

Monitoring and accessibility

Outside the studio, machine listening watches machines for faults, cities for gunshots or traffic, wildlife for species counts and homes for alarms and breaking glass. It also helps accessibility: live captioning, sound awareness notifications on phones for deaf and hard-of-hearing users and descriptions of sounds in video.

Building a local listening toolkit

A small studio can assemble a useful machine-listening toolkit from open models and a little scripting, without sending client audio anywhere. A typical kit contains a speech recogniser with word timestamps for transcription and alignment; a diarisation model for multi-speaker recordings; a general audio embedding model of the CLAP

type for search by text and similarity; a sound-event classifier trained on AudioSet categories for rough tagging; beat, tempo and key estimators for music; and a set of simple signal checks, such as clipping, digital silence, DC offset, phase correlation and loudness, which need no machine learning at all.

The important design choices are about storage and review rather than models. Store results alongside files in a consistent format, such as a database or one sidecar file per audio file, with the model name and version that produced each result, so that results can be regenerated when models improve. Build a simple review interface, even a spreadsheet with time codes and links, so that people can confirm or correct results; corrections can later be used to measure each model's error rate on your material. And run the kit on new material as it arrives, as part of the ingest room in Chapter 21, rather than as an occasional clean-up project.

Limits and cautions

Training-data bias. Systems trained mostly on online video, English speech and Western popular music perform worse elsewhere. A genre classifier may be confident and wrong about music from traditions under-represented in its data. A speech recogniser may perform worse on some accents and dialects, with consequences when transcripts are used for captions or legal records.

Benchmark limits. Short-clip, multiple-choice benchmarks overstate how well systems handle long, messy, real recordings. Results on public benchmarks can also be inflated if test material leaks into training data.

Confidence is not accuracy. A classifier's confidence score and an audio-language model's fluent answer are not guarantees. Calibrate tools on your own material: run them on files whose answers you know and count the errors.

Privacy. Listening systems process speech, which identifies people and may contain private information. Check where processing happens. Cloud services may retain audio; local models do not need to. Recordings of members of the public, clients' unreleased material and

confidential interviews deserve local processing or services with clear retention terms.

Judgement. Machine listening can measure and describe; it cannot decide whether a mix is good, whether a performance moves people or whether a sound fits a story. Use it to find and organise, not to decide.

A Lantern Street archive project

Lantern Street has fifteen years of sessions: tens of thousands of takes, stems and sound-effects recordings, inconsistently named. Over a quiet month the engineers run a local pipeline across the archive. A speech recogniser transcribes every spoken file and a diarisation model separates speakers. A CLAP-style model embeds every file for text and similarity search. A set of classifiers tags instrument, tempo and key for music files. A quality-control pass flags clipped, silent and corrupted files. Everything runs on the studio's own computers, because the archive contains clients' unreleased work.

The results are uneven but transformative. Searching "rain on car roof" finds a dozen forgotten recordings in seconds. The instrument tags are right for common instruments and often wrong for unusual ones. The transcripts make old interviews findable by phrase. The quality-control pass finds a corrupted batch of files from 2017 that nobody had noticed, early enough for a backup to be restored. Nobody trusts any single tag, but the archive has become usable, and that is the point.

Building a listening QC pass

A practical automated quality-control pass need not be elaborate. Lantern Street's pass for delivery files runs in stages. A conventional analysis measures loudness, true peak, loudness range, digital silence, DC offset and clipped samples, and checks the file's format against the specification. A click and dropout detector scans for sudden discontinuities. A mono-compatibility check sums the channels and measures how much level is lost in each section. For speech content, a recogniser transcribes the file and compares it with the script or the approved transcript, flagging missing, repeated or substituted words. For mixed content, a classifier marks where speech, music and effects occur,

and a simple rule flags passages where music is louder than speech by more than a set margin.

The output is a list of time-stamped flags with a short description of each. An engineer then listens to every flag and marks it as a real problem, an intended feature or a false alarm. Over time, those marks show which checks are useful and which produce mostly noise, and the thresholds are adjusted. The pass never approves a file by itself. It decides where a human should listen first.

In practice

- Use machine listening first for search, tagging and quality-control triage, where errors are cheap and coverage matters.
- Treat audio-language model answers as hypotheses; check against the audio before acting.
- Calibrate classifiers and recognisers on your own material and count their errors.
- Process confidential and client audio locally or with services whose retention terms you have read.
- Expect worse performance on under-represented languages, accents and musical traditions.
- Keep decisions about quality, meaning and taste with people.

CHAPTER 18

Spatial, Immersive and Object-Based Audio

Stereo has been the default way to hear recorded music for about seventy years. Its principles were patented by Alan Blumlein in 1931, and stereo records reached the public in the late 1950s. Cinema moved to multichannel sound much earlier and kept adding channels: surround, then 5.1 in the 1990s, then 7.1. Since 2012 a different idea has spread from cinemas into homes, headphones and phones: instead of mixing to a fixed set of loudspeakers, describe where each sound should be in space and let the playback system render that description for whatever speakers or headphones the listener has. That idea, object-based audio, together with scene-based formats such as ambisonics and personalised binaural rendering on headphones, is the subject of this chapter.

Spatial audio is a frontier for AI in two ways. AI is used to create spatial mixes from material that was never spatial, by separating stereo recordings into parts that can be placed around the listener. And AI is being used to make headphone rendering personal, by estimating how a particular listener's head and ears filter sound.

Three ways to describe space

Channel-based audio

In channel-based audio, the mix is made for a defined set of loudspeakers, and each channel feeds one speaker. Stereo has two; 5.1 has five full-range channels plus a low-frequency effects channel; 7.1.4 adds four height speakers to 7.1. The advantage is simplicity: what the mixer hears is what the listener gets if their speakers match. The disadvantage is rigidity: a 5.1 mix played on stereo, headphones or a

different layout must be downmixed, and the result depends on the downmix rules.

Scene-based audio: ambisonics

Ambisonics, developed in the 1970s with Michael Gerzon's work on "periphony", sound reproduction with height, as a key reference, describes a whole sound field around a point rather than a set of speaker feeds. First-order ambisonics uses four channels, which encode the overall pressure and the front-back, left-right and up-down components of the sound field. Higher orders use more channels and give sharper spatial resolution. An ambisonic recording, made with a special microphone array or created by encoding mono sources at chosen directions, can be decoded to any speaker layout or rendered binaurally to headphones, and it can be rotated easily, which makes it ideal for 360-degree video and virtual reality, where the listener turns their head. Major video platforms have accepted first-order ambisonic soundtracks for 360-degree video since the mid-2010s.

Object-based audio

Object-based audio represents a mix as a set of audio objects, each a signal with metadata describing its position, size and other properties over time, usually alongside channel-based beds for ambience and music. A renderer in the playback device turns the objects and beds into signals for the actual speakers or headphones. Dolby Atmos, introduced in cinemas with Pixar's *Brave* in 2012, is the best-known object-based system; it later moved into home theatre, televisions, soundbars, phones, cars and music streaming. Other object-based and next-generation systems include MPEG-H 3D Audio, standardised by ISO and used in some broadcast systems and in Sony's 360 Reality Audio music format, and the Immersive Audio Model and Formats specification, IAMF, published by the Alliance for Open Media in 2023 as an open, royalty-free format.

Google and Samsung introduced Eclipsa Audio, a brand for IAMF-based immersive audio, in January 2025. YouTube began accepting Eclipsa Audio uploads, and Samsung adopted it in its televisions and soundbars from its 2025 range, with further profile support and distribution

through its own streaming service reported in 2026. As of late 2026, Dolby Atmos remained by far the most widely used immersive format in music and film, and the open alternatives were growing from a small base.

The interchange format that ties much of this together is the Audio Definition Model, standardised by the ITU as Recommendation BS.2076, which describes channel-based, object-based and scene-based audio in a common metadata language. ADM metadata is usually carried in Broadcast Wave 64 files, so that a single file can hold all the beds, objects and metadata of an immersive master. Many delivery specifications for immersive content, including Netflix's guidelines for dubbing deliverables, call for ADM Broadcast Wave files.

Headphones and binaural rendering

Most people who hear immersive audio hear it on headphones. Binaural rendering makes that possible by filtering each sound with head-related transfer functions, the filters that describe how sound from a given direction is changed by the listener's head, torso and outer ears on its way to each eardrum, as Chapter 2 introduced. A sound filtered with the right HRTF for a direction above and behind you will seem to come from above and behind you, even though it is played through headphones.

The catch is that HRTFs are personal. The shape of each person's outer ears differs, and the fine spectral cues that let us distinguish front from back and up from down depend on that shape. Rendering with a generic HRTF, measured on a dummy head or averaged across people, works reasonably well for left-right placement and less well for elevation and front-back distinction; some listeners hear sounds inside their head rather than around it. Head tracking, which keeps sounds fixed in space when the listener turns, helps a great deal, because the brain uses changes in the signal as the head moves to resolve ambiguities.

Personalised HRTFs

Measuring a person's HRTF accurately requires an anechoic chamber, small microphones in the ear canals and a loudspeaker moved to hundreds of positions, which is impractical for consumers. The frontier

is estimating a personal HRTF from easier data, such as photographs or depth scans of the ears and head, using acoustic simulation or machine learning trained on databases of measured HRTFs and head shapes.

Consumer products have adopted this. Sony offered HRTF personalisation from photographs of the listener's ears for its 360 Reality Audio format from 2019. Apple introduced Personalized Spatial Audio in iOS 16 in 2022: an iPhone's depth-sensing front camera captures the shape of the listener's head and ears, and the device creates a personalised profile used to render spatial audio on compatible headphones. Apple states that the processing happens on the device and that the scan images are not stored. Research continues on how accurately such methods capture the cues that matter, and on evaluating them with listening tests rather than only with spectral error measures.

HRTF data is exchanged in the SOFA format, the Spatially Oriented Format for Acoustics, standardised by the Audio Engineering Society as AES69, which lets measured or estimated HRTFs be used in different renderers.

Mixing for immersive formats

Immersive mixing follows the principles of good stereo mixing, with more space to use and more ways to go wrong.

Start from the music or the story. Immersive mixing works best when space serves the material: a choir that surrounds the listener, a film scene where rain falls overhead, an electronic track where elements move around in time with the music. It works worst when every element is placed somewhere different to show off the format.

Anchor the centre. Lead vocals and dialogue usually belong at the front, stable and clear. Moving them, or spreading them too widely, makes them hard to follow, especially on headphones.

Use height sparingly. Height channels and elevated objects add air and envelopment. Too much content overhead can feel unnatural and collapse oddly in re-renders to formats without height.

Mind the low end. Low frequencies carry little directional information, and the low-frequency effects channel in cinema-derived formats is an

extra, not a replacement for bass in the main channels. Music mixes in Atmos generally keep bass in the bed and use the effects channel sparingly or not at all.

Check every re-render. An immersive master is played on everything from a cinema to a phone. Listen to the binaural render on headphones, including the renderer's distance settings, which change how far each object seems from the head, and to downmixes and re-renders to 7.1, 5.1, stereo and, ideally, mono. Netflix's specification asks that Atmos loudness and peaks be verified through a 5.1 re-render. For Atmos music, Dolby's and streaming services' guidance has commonly cited an integrated loudness no louder than about −18 LKFS with true peaks at or below −1 dBTP; check the current specification for each service.

Measure appropriately. Loudness for multichannel and immersive programmes is measured with BS.1770 using its channel weightings, and for object-based formats often through a defined re-render. Use a meter that supports the format.

Monitoring immersive mixes on a budget

A full immersive monitoring room, with seven or more speakers at ear level, a subwoofer and four or more height speakers, calibrated and treated, is expensive and out of reach for many engineers. Two compromises are common. The first is a smaller speaker layout, such as 5.1.2 or 5.1.4, in a treated room, with the renderer folding the mix into the available speakers; it is adequate for many decisions but can mislead about the exact placement of objects intended for missing speakers. The second is headphone monitoring with binaural rendering, ideally with head tracking and a personalised HRTF. Headphone monitoring is how most listeners will hear the result, which is an argument for using it, but it is a poor guide to how the mix will sound on speakers in a cinema or home theatre.

Many engineers mix immersive music mainly on headphones and check periodically in a properly equipped room, booked by the hour. That approach works if the checks are scheduled early enough to change decisions, not only at the end. Whatever the monitoring, keep the stereo mix as a reference: an immersive mix that loses the balance and energy

that made the stereo mix work is a step backwards, however much space it adds.

AI and spatial audio

Upmixing with separation

Most of the world's music exists only in stereo. Record companies wanting immersive versions of back catalogue face a problem when multitrack masters are lost or damaged. Source separation, discussed in Chapter 9, provides a route: separate the stereo mix into stems, then place the stems as objects or bed elements. Several commercial tools and services offer AI-assisted upmixing on this principle, and engineers use separation inside conventional immersive mixing sessions to extract elements they need.

The cautions are the separation cautions, amplified. In an immersive mix, separated elements are physically placed apart, so bleed between them becomes audible as an echo of one element arriving from another element's direction. A guitar residue inside a separated vocal object placed at the front, while the main guitar object sits at the side, produces a phantom guitar in front. Careful engineers place separated elements conservatively, keep closely related elements near each other and use separation to create space rather than to scatter.

Automatic placement and rendering

Research systems propose object positions automatically from the content of each stem, and machine-listening models are being trained to understand spatial scenes, as the DCASE 2026 spatial tasks and the spatial questions in the MMAU-Pro benchmark show. Generative models are also being developed to produce spatial audio directly, for example binaural or ambisonic soundtracks from video. As of 2026 these were research and early commercial tools rather than standard practice.

Room and listener modelling

Machine learning is being used to estimate a room's acoustics from a few measurements or a photograph, to render virtual rooms convincingly on headphones and to adapt playback to the room a device is in. Combined

with personalised HRTFs and head tracking, these techniques point toward headphone listening that convincingly reproduces the experience of a particular room and speaker setup, which matters both for consumers and for engineers who mix on headphones.

Spatial audio beyond entertainment

Spatial audio has serious uses beyond film and music: in hearing aids that preserve spatial cues so that wearers can locate sounds; in teleconferencing that places voices around a virtual table so that participants can tell who is speaking; in simulators and training systems; and in navigation aids for people with visual impairments. Each has its own design constraints and, in the case of medical devices, regulation.

A Lantern Street Atmos project

A jazz trio asks Lantern Street for a Dolby Atmos version of an album recorded live in a studio room five years earlier. The multitracks exist: close microphones on each instrument, a stereo pair over the band and a pair of room microphones. The engineers build the mix from those, not from separation. The stereo pair goes in the front bed, the room microphones are placed as objects behind and above the listening position to recreate the room's space and the close microphones reinforce each instrument's position at the front. The bass stays in the bed, centred. One tune has a guest saxophone recorded separately; it is placed slightly to the left and back of the trio to match where the player stood. The engineers check the binaural render with several headphone models, adjust the binaural distance settings so that the trio sits at a natural distance, and check the 7.1.4, 5.1 and stereo re-renders against the original stereo mix. Separation is used once, to reduce piano spill in the drum overheads on a single track.

The result is restrained: the listener feels inside the room with the band rather than surrounded by flying instruments. That is usually the right result for acoustic music.

In practice

- Choose channel-, scene- or object-based approaches according to the destination: ambisonics for 360 and VR, objects for Atmos-style music and film, channels where layouts are fixed.
- Anchor lead vocals and dialogue at the front; use height and movement where they serve the material.
- Check every immersive master on headphones and in every required re-render, and measure loudness through the specified method.
- Use separation for upmixing conservatively, keeping related elements close to avoid audible bleed from the wrong direction.
- Encourage listeners and clients to use personalised HRTFs and head tracking where available, and test mixes with generic rendering too.
- Deliver immersive masters as ADM files where specifications ask for them, with documented loudness and re-render checks.

CHAPTER 19

Provenance, Watermarks and Detection

Until recently a recording was, by default, evidence that something had happened: someone had spoken, sung or played, and a microphone had captured it. Editing could mislead, but fabricating a convincing recording of a person saying something they never said required skill, time and usually access to their recorded voice. Generative audio removed most of that friction. As a result, audio now needs ways to answer questions that used to be answered by the medium itself: where did this recording come from, who made it, was AI involved and has it been altered?

Four families of technique address those questions: metadata and signed provenance records, watermarks embedded in the audio, fingerprints derived from the audio and detectors that infer synthetic origin from the audio alone. None is sufficient alone. This chapter explains each, how platforms and regulators were using them as of late 2026 and what a studio should do in practice.

Metadata and signed provenance

The simplest provenance is metadata: information stored with a file describing what it is and how it was made. Broadcast Wave files carry descriptive fields and timecode. Music distribution metadata, as Chapter 13 described, now includes fields for declaring AI involvement, and Spotify displays artists' AI credits. Metadata is easy to create and easy to read. It is also easy to remove or falsify, and most of it is stripped when a file is converted, uploaded to social media or played through a different system.

Signed provenance adds cryptography. The Coalition for Content Provenance and Authenticity, C2PA, publishes an open technical specification for "content credentials": a manifest attached to a file that records claims about its origin and history, such as which device or software created it, what edits were made and whether generative AI was used, signed so that tampering can be detected. Each editing step can add a new signed claim, building a chain of custody. C2PA manifests can be embedded in common audio and video container formats or stored separately and linked to the file.

Signed provenance proves that the claims were made by the holder of a particular signing key and have not been altered since. It does not prove that the claims are true, and its absence proves nothing, because most content in circulation carries no credentials. Its weakness, like ordinary metadata, is that it can be stripped. That is why C2PA defines "soft bindings": ways to reconnect a file to its manifest after metadata is lost, using a watermark or fingerprint in the audio itself.

Watermarks

An audio watermark is a signal embedded in the audio that is designed to be imperceptible to listeners but detectable by software, and to survive common processing such as compression, resampling and mixing with other sounds. Watermarks have long been used to track broadcast content and to identify leaked pre-release music. Generative AI has given them a new role: marking content as machine-made at the point of generation.

Two neural watermarking systems illustrate the state of the art. Google DeepMind's SynthID marks generated content, including audio; Google states that all music generated by its Lyria models is watermarked with SynthID. Meta's AudioSeal, presented at the ICML conference in 2024 with open code, was designed for detecting voice cloning and localises its watermark at the level of short segments, so that a detector can identify which parts of a recording were generated even if they have been spliced into genuine speech. Both are trained jointly with a detector, so that the watermark can be found reliably after distortions applied during training.

What watermarks can and cannot do

Watermarks are strongest where the generator cooperates. A provider that watermarks all output lets anyone with the detector check whether a file came from that provider. A watermark is weak against an adversary who uses a generator that does not watermark, which includes many open-weight models, and against deliberate removal. Benchmarks such as AudioMarkBench, published in 2024, test watermarks against a range of distortions and attacks, and have found that some kinds of processing can weaken or defeat some watermarking schemes. Heavy processing, such as time stretching, pitch shifting, re-recording through a speaker and microphone, or regeneration through another model, can remove marks.

Watermarks are also, by design, a provider's tool. A studio cannot easily add a SynthID watermark to its own output, and the detectors for proprietary schemes are not always public. For voice cloning in particular, a studio that runs its own models can use open watermarking such as AudioSeal to mark every output it generates, which is a sensible precaution even when not required.

Fingerprints

An audio fingerprint is a compact summary of a recording's perceptual content, computed from the audio itself, that can be matched against a database. Fingerprints underlie song identification apps and the content identification systems that platforms use to detect copyrighted music in uploads. They do not need anything to be embedded, so they work on any content that is in the database, including content created before the system existed.

For provenance, fingerprints can link a file back to a registered original even when metadata is stripped, acting as a soft binding for C2PA. For generated music, fingerprinting can detect when a generator's output closely reproduces an existing recording, which is relevant to the similarity checks in Chapter 11. Fingerprints identify recordings, not methods: they cannot tell whether an unknown recording is synthetic.

Detectors

A detector infers from the audio alone whether it was generated or manipulated, without any watermark or fingerprint. Research on detecting synthetic speech has a long history in speaker-verification security, notably the ASVspoof challenge series, which began in 2015 and has tracked the arms race between spoofing attacks and countermeasures. Detection for generated music is newer.

Detectors work by learning artefacts that generators leave, many inaudible: patterns from neural codecs and vocoders, unusual spectral statistics, phase behaviour or traces specific to a particular model. Deezer, for example, reported in July 2026 that its detector identifies such artefacts and can tell which generator was used, that it detects output from prolific generators such as Suno and Udio and can be extended to others given examples, and that it had made progress toward detecting generated music without training data for a specific generator. Deezer reported an accuracy of 99.8 percent, that the system may miss about 2 in every 1,000 AI-generated tracks and may wrongly flag fewer than 1 in 10,000 human-made tracks, and that it applies AI tags at the album level to reduce errors. These are the company's own figures for its own system. Deezer has licensed the technology to other organisations since January 2026.

Spotify took a different approach. In October 2026 it said that no single signal catches slop or spam, so it combines listener engagement and feedback, upload patterns, metadata and information from several third-party AI detection tools, and that AI detection technology has limitations that will grow as generative AI evolves.

That caution is important. Detectors trained on today's generators may miss tomorrow's. They can be fooled by processing. And their false positives fall on human artists, who may find their work flagged as synthetic. A detector's verdict on a single file is evidence, not proof, and it should never be the only basis for an accusation.

Regulation: the EU transparency rules

The European Union's AI Act made provenance a legal matter. Its transparency obligations in Article 50 apply from 2 August 2026, with a grace period until 2 December 2026 for the machine-readable marking obligation in the case of systems placed on the market before August 2026. According to the European Commission's guidance, providers of AI systems that generate synthetic audio, image, video or text must ensure that outputs are marked in a machine-readable format and detectable as artificially generated or manipulated. Deployers who use AI to create deepfakes, defined as AI-generated or manipulated content that resembles existing persons, objects, places, entities or events and would falsely appear to be authentic or truthful, must disclose that the content is artificial, clearly and at the latest at the time of first exposure. For evidently artistic, creative, satirical or fictional works, the disclosure must be made in an appropriate way that does not hamper enjoyment of the work. The Commission's guidance notes that standard post-production uses, such as AI-generated background scenes or effects in films, are not likely to make content falsely appear authentic.

A Code of Practice on transparency of AI-generated content, published in 2026 and confirmed by the Commission and the AI Board as an adequate voluntary tool for demonstrating compliance, sets out specific measures for audio. For deepfake audio-only content shorter than 30 seconds, such as advertisements, signatories include a short audible disclaimer at the latest at the beginning; for longer formats, including AI-generated phone calls, podcasts and radio, they repeat audible disclaimers at the beginning, at appropriate intermediate points and at the end. Where audio is delivered through a screen, they also display a visual label, and the Commission has published icons for the purpose. The Code calls for disclaimers in plain language in the language of the content, and for accessible alternatives such as tactile cues where devices allow.

For sound engineers working on content distributed in the EU, the practical consequences are concrete. Synthetic voices of real people in contexts where listeners could take them as genuine need audible disclosure. AI-generated podcasts and radio segments need repeated

disclaimers under the Code's approach. Machine-readable marks from generation tools should be preserved rather than stripped by processing where possible. And records of what was generated, by what tool, become evidence of compliance.

Other jurisdictions are moving in similar directions with different details, and platforms impose their own labelling rules. Chapter 20 discusses the wider legal picture.

A practical provenance policy

Lantern Street adopted a provenance policy in 2026 with six elements.

Log every generation. For every AI-generated or AI-transformed element used in a deliverable, record the tool, model version, date, prompt or input, settings, operator and the terms in force. Appendix A has a template.

Keep originals. Preserve unprocessed source recordings for every deliverable, so that the studio can show what was recorded as opposed to generated.

Preserve marks. Avoid processing that strips watermarks from generated content unnecessarily, and do not deliberately remove them. Where tools support C2PA, keep content credentials attached through the workflow and add the studio's own claims.

Watermark synthetic voices. Mark every synthetic voice output generated by the studio's own models with an open watermark, and note the watermark in the delivery record.

Disclose appropriately. Agree with each client at the start of a project how AI involvement will be disclosed: in metadata, credits, programme notes, audible disclaimers or labels, according to platform rules and law.

Do not over-trust detectors. Use detection tools to screen incoming material, for example to check whether a "found" recording submitted for a documentary might be synthetic, but treat results as one piece of evidence alongside source verification, metadata and journalistic checks.

Checking incoming material

Provenance is not only about what a studio sends out; it is also about what comes in. Clients send stems, samples, voice references, archival recordings and sound effects of uncertain origin. A brief intake check reduces risk. Ask where the material came from and who owns it. Look at the metadata and, where present, the content credentials. Run a content identification check on music that might contain uncleared samples. For voice references, confirm that the person whose voice it is has consented to the intended use. For material presented as a genuine recording of a real event or person, especially for journalism or documentary, treat a detector's result as one input and ask for the original file and its chain of custody.

Record the answers in the project file. If questions arise later, the record shows that the studio asked.

Authenticity for journalism and documentary

For news and documentary, provenance is about trust. When a producer receives a recording of a public figure, the question is whether it is genuine. Detectors can help raise or lower suspicion. Stronger evidence comes from the chain of custody: who recorded it, on what device, when and where, and whether the original file and its metadata are available. Capture devices and apps that sign recordings at the moment of capture, using C2PA or similar schemes, can make genuine recordings easier to verify. Journalists increasingly ask for original files rather than re-recordings or social-media copies, and a sound engineer asked to "clean up" a recording of uncertain origin should first ask where it came from.

In practice

- Use layers: metadata and signed credentials, watermarks, fingerprints and detection, knowing that each has gaps.
- Log every AI generation used in a deliverable, keep original recordings and preserve watermarks and content credentials.
- Watermark synthetic voice outputs from your own models with an open scheme.

- For EU-distributed work, plan audible and visual disclosures of deepfake audio from the start, following the transparency Code's approach for audio.
- Treat detector results as evidence, not proof, and never accuse a human artist on a detector's verdict alone.
- Verify the origin of any recording of a real person before processing it for publication.

PART V

Rights, Operations and What Comes Next

CHAPTER 20

Rights, Consent and Copyright

This chapter is not legal advice, and it cannot be: the law on AI and audio differs between countries, was changing month by month in 2025 and 2026 and depends heavily on facts. What it can do is give a working sound engineer a map of the questions, the main legal developments as of November 2026 and the professional practices that keep work defensible wherever the law settles. When a project raises real legal risk, such as a commercial release using a cloned voice, a generated track with a resemblance to a known song or a contract that is silent on AI, get advice from a lawyer who practises in the relevant country.

The rights in a recording

Every piece of recorded music involves at least two layers of copyright: the composition, meaning the music and lyrics, usually owned by songwriters and their publishers; and the sound recording, the master, usually owned by whoever paid for or made the recording, often a record label. Performers have rights in their performances, which in many countries include rights to consent to recording and to share in some income. Producers and engineers may have contractual rights, such as royalties or credits. In film and television, rights are layered further among producers, writers, composers, performers and the owners of any licensed material.

AI touches each layer. Training a model may involve copying recordings and compositions. A generated output may reproduce parts of them. A cloned voice imitates a performer. A separated stem is still part of the original master and composition. A generated track may or may not be anyone's property at all.

Training on copyrighted audio

The central legal question for generative audio has been whether training models on copyrighted recordings without permission is lawful. Developers have argued that training is a transformative use, permitted in the United States as fair use and in some other countries under exceptions for text and data mining. Rights holders have argued that training copies their works on an enormous scale to build products that compete with them.

The United States

In June 2024, Universal Music Group, Sony Music Entertainment and Warner Music Group sued the two leading music-generation companies, Suno and Udio, in federal courts in Massachusetts and New York respectively, alleging mass copying of recordings for training. Over the following two years the landscape shifted from litigation toward licensing, without a court deciding the core question. Universal settled with Udio in October 2025. Warner settled with Udio in November 2025 and, days later, became the first major label to settle with Suno, announcing a partnership under which Suno would launch licensed models in 2026. Suno released its v6 models in September 2026, describing them as built with licensed music from participating artists of Warner, BMG and, according to press reports, the distributor Believe. Udio announced a licensed platform built on its agreements with Universal and Warner.

Sony settled with neither company as of November 2026. Universal and Sony continued their original case against Suno, narrowed after Warner's settlement, and on 18 September 2026 filed a second lawsuit alleging that Suno had copied more than 60,000 of their recordings to train its newer models. Suno's principal defence in the first case is fair use; summary judgment motions in that case were due by June 2027, according to court filings reported in November 2026. A musicians' union, the American Federation of Musicians, separately sued over the labels' licensing deals, arguing that musicians' interests had not been protected; the labels sought dismissal.

The US Copyright Office has published a series of reports on copyright and AI. Part 3, on generative AI training, released in pre-publication form in May 2025, analysed how fair use applies to training and concluded that some training uses are likely fair and others are not, depending on factors such as the purpose of the model and whether its outputs compete with the works it was trained on. The report is guidance, not law; courts decide.

Germany and the European Union

German courts produced the first major rulings against AI developers on music. On 11 November 2025, the Munich Regional Court I held, in a case brought by GEMA, the German collecting society for composers and lyricists, that OpenAI's language models had memorised the lyrics of nine German songs, that this memorisation was a reproduction relevant to copyright, that the text and data mining exception did not cover such reproductions retained in a trained model and that chatbot outputs reproducing recognisable lyrics infringed copyright. On 31 July 2026, the same court largely ruled for GEMA in a case against Suno, finding infringement relating to six works through training, through retaining the works in models stored in Germany and through generating recognisable outputs, and holding that the text and data mining exception did not apply. Both judgments were reported as not final and subject to appeal.

The EU's copyright framework includes a text and data mining exception, with an opt-out for rights holders who reserve their rights. The AI Act adds obligations for providers of general-purpose AI models, including publishing a summary of training content and having a policy to respect rights reservations. How these interact with music generation was one of the most contested questions in European law in 2026.

The United Kingdom and elsewhere

The UK government consulted in 2024 and 2025 on a proposal to allow text and data mining of copyrighted works for AI training with an opt-out for rights holders. After strong opposition from the creative industries, including the music industry, the government reported in March 2026 that it was no longer pursuing that preferred option and would gather

further evidence and consider other approaches. Other countries, including Japan and Singapore, have broad data-mining exceptions; others have none. A model may be trained in one country, hosted in another and used in a third, which is part of why the Munich court's focus on models stored in Germany mattered.

What this means in the studio

For an engineer choosing tools, the training question translates into a risk question: how confident are you that the tool's outputs can be used commercially without legal exposure? As of late 2026, a growing number of vendors advertise models trained on licensed or owned data, or offer indemnities against claims arising from their outputs. Read the terms. Note whether the vendor claims licensed training, whether it offers indemnity, what it excludes and in which territories the terms apply.

Copyright in outputs

The second question is whether anyone owns what a generator produces. In the United States, the Copyright Office's Part 2 report, on copyrightability, published in January 2025, concluded that copyright protects only human authorship; that material generated by AI without sufficient human control over its expressive elements is not protected; that prompts alone are generally insufficient; and that human selection, arrangement, modification and the human-authored parts of a mixed work remain protectable. Courts in the United States have upheld refusals to register works claimed to be authored by an AI system. Other countries take different positions, and the United Kingdom's law contains a provision for computer-generated works whose scope in the age of generative AI is uncertain.

The practical consequence is that a purely generated track may be difficult to protect against copying, while a track in which a person wrote the melody and lyrics, performed the vocal or substantially arranged and edited the generated material is on much firmer ground. Keep records of human contributions: drafts, recordings, session files and notes. They are evidence of authorship.

Outputs that resemble existing works

A generated output that reproduces a recognisable part of an existing song, such as a melody, a lyric or a distinctive recorded sound, can infringe that song's rights regardless of how it was made. The Munich rulings found infringement in outputs reproducing protected works. Infringement through similarity is assessed in the same way as for human-made music, which in many countries means asking whether a substantial part of the protected work was copied. Chapter 11 described checks to run before release.

Sampling, interpolation and separated stems

Sampling, the use of part of an existing recording in a new one, has its own long legal history, and source separation has made it technically easier without changing its rules. Using a sample normally requires two permissions: from the owner of the sound recording, for the actual audio, and from the owners of the composition, for the music and words it contains. Re-recording the musical material instead of using the original audio, sometimes called interpolation, avoids the need for the master owner's permission but still requires the composition owners'. A separated vocal or drum break from a commercial record is a sample of that record, however much it has been processed.

Generative tools blur these lines in a new way. A generator asked to produce "a drum break in the style of" a famous record may produce something close enough to the original composition or recording to raise the same questions as a sample. The safest course for commercial releases is to treat any recognisable resemblance to a specific existing recording or composition as a clearance question, whether it came from a sample, a separation or a generator.

Voice, likeness and personality rights

Copyright protects recordings and compositions, not voices as such. A voice that sounds like a famous singer is not, in itself, a copy of any recording. Protection for voice and likeness comes instead from rights of publicity, personality rights, passing off, consumer protection and, increasingly, specific laws.

In the United States, protection is mostly at state level. Tennessee's ELVIS Act, the Ensuring Likeness, Voice, and Image Security Act, effective 1 July 2024, expanded the state's right of publicity to cover a person's voice, including a simulated voice, and created liability for distributing tools whose primary purpose is producing an identifiable person's voice or likeness without authorisation. California and other states passed laws on digital replicas in contracts and of deceased performers. At federal level, the NO FAKES Act, which would create a national right against unauthorised digital replicas, was introduced in 2025 and in a revised version in 2026; as of November 2026 it had not become law, after a Senate attempt to pass it by unanimous consent in September 2026 was blocked.

In Denmark, the government introduced Bill L 49 in October 2026 to amend the Copyright Act so that sharing realistic digitally generated imitations of a person's appearance, voice or other personal characteristics would require that person's consent, with limited exceptions such as parody and satire; as of November 2026 it was before parliament. Other European countries rely on personality rights in civil law and on data protection law, because a voice is personal data under the EU's General Data Protection Regulation when it identifies a person, and a voiceprint used to identify someone is biometric data that receives special protection.

The EU AI Act's transparency rules, described in Chapter 19, require disclosure of deepfakes, including synthetic voices that would falsely appear authentic, from August 2026.

Performers and collective agreements

Collective agreements have moved faster than legislation. In the United States, SAG-AFTRA's 2023 television and theatrical agreement, which ended that year's actors' strike, required informed consent and compensation for the creation and use of digital replicas. Its 2025 Interactive Media Agreement for video games, ratified on 9 July 2025 after an eleven-month strike, requires written consent, separately signed, with a reasonably specific description of the intended use of any digital replica, including whether it will be used to generate dialogue in

real time, allows performers to suspend consent for generating new material during a strike and requires employers to report on replica use after a game's release. Unions in other countries, including Equity in the United Kingdom, have campaigned for and negotiated similar protections, and many individual contracts now include AI clauses.

For anyone engaging performers, the lesson is to address AI explicitly in every contract: whether recordings may be used to train or create a replica, for what purposes, with what approval, payment, duration and deletion.

Contracts with clients and collaborators

Many disputes will arise not from statutes but from contracts that did not anticipate AI. Lantern Street now includes AI terms in its standard engagement letter, covering five points.

Whether the studio may use AI tools on the client's material, and which categories, such as repair, separation, mixing assistance, generation or voice synthesis, require specific approval.

That client material will not be uploaded to services that retain it or use it for training without the client's consent.

How AI involvement will be disclosed in credits and metadata.

That the client warrants it has the rights to all material it supplies, including consent from any person whose voice will be cloned or converted.

That the studio will keep an AI log for the project and provide it on request.

Similar terms belong in agreements with session musicians, voice actors and co-writers: whether their contributions may be used to train models, whether their voices may be cloned and how they will be credited and paid.

Ethics beyond the law

Law sets a floor. Professional ethics in audio have always gone further, around consent to record, honesty in editing, crediting contributors and

respecting the people whose voices and performances we handle. AI makes some of those principles more pressing.

Consent is not a box. Real consent requires that people understand what they are agreeing to. A voice actor who signs a broad licence without understanding that it permits indefinite synthetic reuse has not meaningfully consented. Explain plainly.

Honesty in documentary and news. Generating or altering the words of real people in factual work deceives the audience, whatever the law requires. Chapters 8 and 14 drew this line; hold it.

Credit the humans. Generative tools learn from human work. When you use them, credit the people who made your own production, and be honest about what the machine did.

Think about displacement. AI tools change the market for session musicians, voice actors, Foley artists and library composers. Engineers make choices every day about when to hire a person and when to generate. Making those choices consciously, and paying fairly when hiring, is part of being a professional.

Accessibility is a benefit. Not every ethical question argues against AI. Voice banking for people losing their speech, audio description at scale, captioning and dialogue enhancement extend access to sound for people who have been excluded. Those uses deserve support.

In practice

- Assume composition, master and performer rights all apply to anything you separate, sample or train on.
- Read and save the terms of every generative tool you use commercially; note licensed-training claims and indemnities.
- Keep evidence of human authorship in every project that uses generated material.
- Check generated outputs for resemblance to existing works before release.

- Obtain identified, informed, written, scoped consent for any voice clone or conversion, and address AI explicitly in every performer, collaborator and client contract.
- Track the law in the countries where you work and distribute, and get legal advice for high-risk releases.

CHAPTER 21

Designing and Costing the Pipeline

By this point the book has covered dozens of tools. A professional operation needs more than tools: it needs a pipeline, a defined sequence of stages that turns raw material into deliverables reliably, with known costs, known checks and known owners. Pipelines are what let a studio take on more work without losing quality, bring a new engineer up to speed, explain a result to a client months later and adopt a better model without breaking everything that depends on the old one.

This chapter describes how to design an AI-assisted audio pipeline, how to estimate its costs and how to change it safely.

Rooms in a pipeline

Think of a pipeline as a series of rooms, each with one job. For each room, define five things: its input, its output, the tools used, the check that must pass before material leaves and who owns that check. A generic post-production and podcast pipeline at Lantern Street has these rooms.

Capture. Input: performances. Output: raw recordings with session sheets. Tools: microphones, interfaces, recorders. Check: levels, noise and completeness reviewed at the session. Owner: recording engineer.

Ingest and log. Input: raw recordings. Output: files named and stored, with metadata, backed up in two places. Tools: file management scripts, checksums. Check: file counts and checksums match. Owner: assistant engineer.

Transcribe and index. Input: raw recordings. Output: transcripts, speaker labels and searchable embeddings. Tools: speech recognition, diarisation, audio embedding models, run locally for client material.

Check: spot check of transcript accuracy on names and key terms. Owner: assistant engineer.

Edit. Input: recordings and transcripts. Output: edited session. Tools: DAW, transcript editor. Check: full listen of edits. Owner: editor.

Repair. Input: edited session. Output: repaired tracks alongside originals. Tools: classical and neural repair, as in Chapter 8. Check: comparison against originals and against the source transcript; changed words flagged. Owner: editor or dialogue editor.

Separate. Input: material needing stems. Output: stems with recombination check. Tools: separation models, as in Chapter 9. Check: solo, context and null tests. Owner: mix engineer.

Generate. Input: briefs for music, effects or voice. Output: generated assets with logs. Tools: generators, as in Chapters 10 and 11. Check: quality, similarity, rights and consent confirmed. Owner: producer or sound designer.

Mix. Input: all tracks and assets. Output: mixes and stems. Tools: DAW, processors, assistants. Check: translation on multiple systems. Owner: mix engineer.

Master and conform. Input: mixes. Output: masters to specification. Tools: mastering processors, assistants, loudness meters. Check: loudness, true peak, format and encoded preview. Owner: mastering engineer.

Quality control. Input: deliverables. Output: approved deliverables and QC report. Tools: automated QC and a full human listen. Check: specification compliance and listen-through. Owner: someone other than the mix engineer.

Deliver and archive. Input: approved deliverables. Output: delivered files, delivery note, archived project with AI log. Tools: delivery platforms, archive storage. Check: receipt confirmed; archive verified. Owner: producer.

Not every job uses every room. A music single skips transcription; a podcast rarely needs separation. Writing the rooms down makes it obvious which ones a job needs and who is responsible for each check.

The run record

Every AI process in the pipeline should leave a record. Lantern Street calls it a run record, and it has the same fields for every tool: project, file or files processed, room, tool and model name, model version or date, whether it ran locally or in the cloud, settings, operator, date and time, output file names and the result of the check. For generation, the record also includes the prompt or input, the terms of service in force and the rights and consent status. Appendix A contains a template.

Run records serve three purposes. They make results reproducible: if a client asks for a revision six months later, the engineer can rerun the same process or knows exactly what would differ. They make results explainable: when a broadcaster asks whether generative reconstruction was used, the answer is in the record. And they support compliance, because disclosure obligations under platform rules and laws such as the EU's transparency rules depend on knowing what was generated.

Run records do not need special software. A spreadsheet per project works. Many tools can export their settings, and some write processing history into file metadata; capture that where possible.

Local or cloud

Many AI audio tools are available both as software that runs on your own computers and as cloud services. The choice affects cost, privacy, speed and control.

Local processing keeps material on your own hardware, which matters for unreleased music, confidential interviews and anything covered by a non-disclosure agreement. Its costs are mostly up front: capable computers, often with a powerful graphics processor, and the time to install and maintain software. Once set up, the marginal cost of processing is electricity and time. Local models can be pinned to a version and will not change unless you change them.

Cloud processing needs no special hardware, often offers the most advanced models and scales easily. Its costs are ongoing, typically charged per minute of audio, per character of text, per generation or by subscription with usage limits. Material leaves your control: read the

provider's terms on retention, use for training and the location of processing. Cloud models can change without notice, which can make results non-reproducible, and services can change terms or close. Udio's move to transitional terms in late 2025, as it rebuilt its service around licensing agreements, is an example of how quickly a cloud tool's conditions can change.

A sensible default for a professional studio is local processing for client material wherever quality allows, cloud processing for tasks where cloud models are clearly better and the client has agreed, and generation in the cloud under terms that have been checked and saved.

Estimating cost

AI changes the cost structure of audio work in two directions: it removes hours of manual work and it adds subscription, usage and hardware costs, plus new checking work. A cost estimate should include both. The following illustrative estimate shows the method; the figures are invented round numbers for the example and are not current prices.

A podcast network produces 20 episodes a month, each 60 minutes, from double-ender recordings with two to four participants. Before adopting AI tools, each episode took about 6 hours of editing and repair and 1 hour of mixing and QC. With transcript editing, neural repair and automated levelling, the same work takes about 2.5 hours of editing and repair and 1 hour of mixing and QC, plus about 30 minutes of additional checking of AI output, mainly transcript accuracy and repaired passages. Labour per episode falls from about 7 hours to about 4 hours.

On the cost side, suppose transcription and repair run locally on a workstation that costs the equivalent of a modest monthly amount over its useful life, and a cloud levelling and enhancement service charges by the hour of audio. Add the time spent maintaining the tools and updating the pipeline, perhaps a few hours a month. In most such cases the labour saving far exceeds the tool costs. The interesting decisions are about where the saved time goes: more episodes, faster turnaround, more careful editing or lower prices.

The method generalises. For each room, estimate hours before and after, the new checking time, tool costs per unit of audio and fixed costs, and

compare. Be honest about checking time; pipelines that skip it look cheaper until a mistake reaches a client.

A music-library pipeline

A second example shows how the same method applies to music. A production-music library commissions and publishes around two hundred tracks a month for film, television and online creators. Its pipeline has rooms for brief, composition and production by contracted composers, technical intake, stem preparation, mastering, tagging, rights clearance and delivery to its catalogue.

AI appears in several rooms. At intake, machine listening checks every file for format, clipping, silence and loudness, and flags tracks whose stems do not sum to the full mix. In stem preparation, separation is used only to recover stems for a few older catalogue tracks whose stems were lost, each checked by an engineer. In mastering, an assistant proposes chains that engineers audit, and loudness and true peak are verified automatically against the library's delivery specification. In tagging, classifiers propose genre, mood, instrumentation and tempo, and editors confirm or correct them before publication. In rights clearance, every composer warrants that tracks contain no generated material unless declared, generated elements are logged with their tool and terms, and a content identification check looks for unintended similarity to existing recordings before release.

The library's rule is that machines propose and people approve in every room where an error would reach a customer. Its run records let it answer customers' questions about AI involvement precisely, which has become a selling point as broadcasters and platforms ask for such assurances.

Throughput and scheduling

AI processing changes where the bottlenecks are. Separation of a feature-length soundtrack or transcription of a hundred hours of archive may take hours of machine time, but it can run overnight. Human review cannot. Design schedules around human attention: batch machine processing so that results are ready when the engineer arrives,

and prioritise review by risk, checking flagged passages and high-stakes material first.

Real-time factor, introduced in Chapter 7, helps estimate machine time. A model with a real-time factor of 0.2 on your hardware will process an hour of audio in twelve minutes. Measure it on your own machines rather than trusting published figures, which may come from different hardware.

A golden set

Before adopting a new model or a new version of an existing one, test it on a golden set: a fixed collection of difficult files from your own work, with known problems and known good results. Lantern Street's golden set includes a reverberant interview, a phone recording, a dense rock mix, a solo piano piece, a whispered voice-over, a recording with overlapping speakers, a field recording with wind and a few files in languages other than English. For each file the set records what a good result sounds like and what failures to listen for.

When a new separation model appears, the engineers run it on the golden set's music files and compare with the current model at matched loudness, blind where possible. When a new dialogue isolator appears, they run it on the speech files and check for changed words. A tool is adopted only when it is better on the material the studio actually handles, not on someone else's benchmark. The golden set also catches regressions: a model update that improves average quality but fails badly on whispered speech is visible immediately.

Versioning and change

Models change, and their outputs change with them. Three practices keep projects stable.

Pin versions for the life of a project. If a project starts with a particular model version, finish it with that version, unless a change is deliberate and logged.

Render results. Once an AI-processed track is approved, render it to audio and keep the rendered file, so that the project no longer depends on the model being available or behaving the same way.

Change deliberately. Introduce new tools or versions through the golden set, update the pipeline documentation and tell the team what changed.

Security and data handling

AI pipelines concentrate valuable and sensitive data: unreleased music, private interviews, voice references and voice models. Protect them as you would client masters. Restrict access to people who need it. Keep voice data and models encrypted and separate. Prefer local processing for sensitive material. Read cloud providers' terms on retention and training, and turn off any option that allows your uploads to be used to train their models unless the client agrees. Delete material according to project agreements, and keep a record of deletion.

People and roles

AI does not remove the need for people in a pipeline; it changes what they do. Editors spend less time on mechanical cleanup and more on judgement and checking. Assistants spend less time logging by hand and more time running and checking automated processes. New responsibilities appear: someone must maintain the pipeline, manage tool licences and terms, keep the golden set current and evaluate new models. In a small studio that may be one person's part-time job; in a large facility it becomes a role in its own right, sometimes called an audio pipeline lead or technical director.

Training matters. Engineers who learned their craft before AI need time to learn the new tools' failure modes. Engineers who start with AI tools need to learn the fundamentals in Part I, so that they can judge what the tools produce. Pairing them helps both.

Energy

AI processing uses electricity, sometimes a great deal, particularly for training and for generation at scale. For most studios, the energy used by

inference on local hardware is modest compared with heating, cooling and lighting, but it is not zero, and cloud services have their own footprint. Avoid needless processing, such as regenerating dozens of variations when a few would do, and prefer efficient models where quality is equal.

In practice

- Write your pipeline down as rooms, each with input, output, tools, check and owner.
- Keep a run record for every AI process: tool, version, settings, operator, inputs, outputs and check result, plus terms and consent for generation.
- Default to local processing for client material; use cloud services under checked and saved terms with client agreement.
- Estimate costs including checking time, tool fees, hardware and maintenance, and decide deliberately where saved time goes.
- Test every new model on a golden set of your own difficult material before adoption.
- Pin model versions per project, render approved results and change tools deliberately.

CHAPTER 22

Research Directions to Watch

Writing about the future of a field that changes monthly is a good way to be wrong. This chapter does not predict products. It describes research directions that were clearly active as of late 2026, explains why each matters to working engineers and suggests what to watch for. Some will reach studios within a year or two; some will take longer; a few may stall. The aim is to help you recognise important developments when they arrive and judge them with the four questions from Chapter 1.

Fewer, smarter tokens

Neural audio codecs, introduced in Chapter 7, are the foundation of most generative audio, and codec research was among the busiest areas of audio machine learning in 2025 and 2026. The direction is consistent: fewer tokens per second, a single codebook rather than stacks of codebooks, tokens that carry meaning as well as sound and streaming operation with low latency. Papers through that period reported speech codecs operating below one kilobit per second, some at a few hundred bits per second, by separating linguistic content from acoustic detail and aligning tokens with text or phonemes.

Why it matters: the codec sets the ceiling on what a generator can produce and how fast. Better low-rate codecs mean faster generation, longer coherent outputs and cheaper real-time systems. They also matter for communication: neural speech codecs at very low bitrates could make remote recording and calls usable on poor connections, with the caveat that codecs which reconstruct speech from meaning-bearing tokens can, like generative restoration, produce words that are clear but not quite what was said.

What to watch: codecs that handle music and general sound as well as speech at low rates; evaluation of intelligibility and fidelity on real,

noisy, multilingual speech; and whether low-rate neural codecs appear in mainstream communication and recording tools.

Conversation in real time

Kyutai's Moshi, released in 2024, demonstrated a full-duplex spoken dialogue model: it listens and speaks at the same time, handles interruptions and overlaps and responds with latency comparable to human conversation, built on its Mimi codec. Since then, major AI developers and voice companies have released speech-to-speech conversational systems that skip the traditional chain of recognition, text reasoning and synthesis.

Why it matters: synthetic voices will increasingly be live participants rather than rendered files, in customer service, games, broadcasting and podcasts. Engineers will record, mix and process conversations in which one party is a model, with challenges of timing, overlap, consistent tone and, under the EU transparency rules, audible disclosure for longer synthetic audio formats.

What to watch: production tools for mixing and editing live human-AI conversations; how disclosure is handled in live audio; and the security implications of real-time voice impersonation.

One model, many tasks

Early AI audio tools did one thing each: separate, denoise, generate music, transcribe. The research direction is toward unified models that understand, generate and edit audio from instructions. NVIDIA's Fugatto, announced in 2024, was described as a generative model that follows free-form instructions to generate or transform speech, music and sound. Meta's SAM Audio, released in December 2025, separates any sound described by text, shown in video or marked in time. Audio-language models such as NVIDIA's Audio Flamingo 3 combine listening with reasoning. Commercial music generators in 2026 added natural-language editing of existing songs, such as changing one section or one lyric while keeping the rest.

Why it matters: for engineers, unified models promise interfaces in which you describe what you want done to a recording, such as "remove the air conditioning, keep the room, make the second speaker a little brighter", and the model does it. That is convenient and, for the reasons discussed throughout this book, demands careful checking: a single model doing many things can change many things.

What to watch: whether instruction-driven editing becomes precise and repeatable enough for professional work; whether such models expose what they changed; and how they integrate with DAWs rather than replacing them.

Control

The most consistent complaint from professionals about generative audio is lack of control. Research is answering on several fronts. Video-to-audio work presented at CVPR 2026, such as EchoFoley and FoleyDirector, focused on letting a sound designer specify which events produce which sounds and when. Music research explores generation conditioned on chords, melodies, stems and structure, and generation of separate stems rather than finished mixes. Differentiable signal processing, covered in Chapter 16, offers models whose controls map to familiar parameters.

Why it matters: professional work is revision. A tool that produces impressive first drafts but cannot make a specific change on request is a toy in a professional context. Tools that generate stems, follow timing precisely and accept edits at the level of a note, a hit or a word fit into real workflows.

What to watch: stem-level generation with reliable separation between parts; generation that follows a DAW's timeline, tempo map and markers; and symbolic and audio hybrids that let you edit the notes of a generated part and regenerate its sound.

Real time, on device, verified

Chapter 16 described the work on low-latency neural synthesis, such as BRAVE, and on compiling learned processors into ordinary DSP code,

such as ADAC, with stability certificates. Meanwhile, phones, laptops and hearing devices gained dedicated neural processors. The direction is toward neural audio that runs locally with low latency and predictable behaviour.

Why it matters: real-time AI on ordinary hardware removes cloud dependence, protects privacy and makes neural processing usable in performance and live sound. Verified behaviour, such as guaranteed stability, is what professional tools need.

What to watch: neural plug-ins with latency under a few milliseconds and no jitter; neural processors whose learned parameters can be inspected and edited like classical ones; and on-device enhancement in recorders, microphones and interfaces.

Spatial and personal

Personalised HRTFs, spatial machine listening and generation of spatial audio are active areas, as Chapter 18 described. DCASE 2026 included tasks on acoustic imaging and spatial semantic segmentation of sound scenes, and the MMAU-Pro benchmark included spatial questions. Research on estimating room acoustics from images and sparse measurements, and on rendering virtual rooms convincingly, continued.

Why it matters: headphone listening dominates, and its quality depends on how well rendering fits the listener. For engineers, convincing virtual monitoring on headphones could change how and where mixing happens. For listeners, personalised rendering could make immersive audio work reliably rather than for some people only.

What to watch: validation of personalised HRTFs with listening tests; standard ways to share personal profiles across devices; and virtual-room monitoring good enough for professional mixing decisions.

Measuring what matters

Audio research still struggles to measure quality. Objective metrics such as SDR, Fréchet Audio Distance and PESQ correlate imperfectly with listener judgements. Listening tests are expensive and hard to compare. Benchmarks such as MMAU-Pro showed large gaps between short-clip

and long-form performance. Work on learned quality predictors, on better listening-test methods and on benchmarks closer to real tasks is ongoing.

Why it matters: when vendors claim improvements, the claims rest on metrics. Better metrics mean claims you can trust more. Until then, your golden set from Chapter 21 is the metric that matters most.

What to watch: benchmarks built from professional tasks, such as repairing real dialogue or separating real multitracks, with expert listeners; and vendors who publish listening-test methods rather than just scores.

Provenance that survives

Chapter 19 described watermarks, fingerprints, signed credentials and detectors, and their weaknesses. Research continues on watermarks that survive heavier processing, on combining watermarks and fingerprints as soft bindings for content credentials, with open benchmarks published in 2026 comparing such methods across images, audio and video, and on detectors that generalise to generators they were not trained on. Regulation, particularly the EU's Article 50 obligations from August 2026, has created demand for all of these.

Why it matters: the information value of recorded audio depends on being able to tell real from synthetic and to trace origin. Engineers will be asked to preserve and add provenance as routinely as they add metadata today.

What to watch: C2PA support in DAWs, recorders and delivery platforms; open, robust watermarks adopted by providers; and how detection error rates hold up as generators improve.

Data, consent and payment

The legal and commercial shift toward licensing, described in Chapter 20, is also a research agenda. Licensed models need licensed datasets with clear provenance. Opt-in artist models need ways to control and audit how a voice or style is used. Payment schemes need some measure of how much a training work contributed to an output, a problem known

as training data attribution, which remains technically hard for generative models.

Why it matters: the economics of music and voice work in the coming decade may depend on whether attribution and opt-in systems work well enough to pay people fairly and transparently.

What to watch: technical methods for attribution, independent audits of licensed models and how artist opt-in programmes report usage and payment, a gap noted by critics of the 2026 label deals.

Under-served languages and traditions

Most AI audio systems are best at English speech and Western popular music because that is what their data contains. Research on low-resource languages, regional accents and musical traditions outside the Western mainstream is growing, often led by communities themselves. It matters for fairness, for the accuracy of captions and transcripts and for the preservation of cultural heritage. It also raises questions of data sovereignty: communities may reasonably want control over how recordings of their languages and music are used to train models.

Accessibility

Some of the most valuable research applies AI audio to access: hearing aids that separate speech from noise, voice banking and synthetic voices for people losing speech, automatic audio description, sound awareness for deaf users, dialogue enhancement at playback and real-time captioning. These uses rarely make headlines, but they show the technology at its best.

Back to the cylinder

Edison's phonograph, on the cover of this book, recorded a voice by cutting a groove whose shape followed the sound. Everything that came after, from tape to neural codecs, has been a more powerful way of doing the same thing: turning sound into something that can be stored, changed and returned. AI adds the ability to make sounds that never happened and to change recordings in ways no one can see. That power

makes the oldest parts of the craft more important: knowing how sound behaves, listening carefully, keeping the original, respecting the people whose voices and performances pass through your hands and being honest about what you did.

The signal rooms will keep changing. The person listening at the end, deciding whether it worked, is still you.

Reading research as an engineer

Much of the material in this chapter first appears as research papers, often as preprints that have not yet been peer reviewed. Engineers do not need to follow the mathematics to get value from them. A few questions extract most of what matters.

What task does the paper address, and is it your task? A speech codec evaluated on clean English read speech tells you little about noisy multilingual interviews. What data was the system trained and tested on, and is that data like yours? What was it compared with, and were the comparisons fair, using the same data and comparable model sizes? Which metrics are reported, and is there a listening test? If there is, how many listeners, on what material and under what conditions? Are audio examples available? Listen to them, preferably on headphones, and listen especially to the examples the authors did not choose to feature, if they publish a full test set. Is code or a model released, and under what licence? And what do the authors say about limitations? Good papers state them plainly.

Finally, note the date. A result that was state of the art in a preprint may be surpassed within months. The most durable things in a paper are usually its analysis of why a problem is hard and its evaluation method, which can inform your own testing long after the model itself is obsolete.

In practice

- Follow codec, real-time and controllability research; these will shape tools soonest.
- Expect unified, instruction-driven models, and insist that they show what they changed.

- Judge new tools on your golden set, not on vendor metrics.
- Build provenance into your workflow now; regulation and platforms will require more of it.
- Watch licensing, attribution and opt-in systems, and support fair payment for the people whose work trains models.
- Pay attention to accessibility uses; they are among the best reasons to adopt AI audio.

Appendix A. Templates and Checklists

These templates collect the working documents used throughout the book. Adapt them to your own studio, keep them with each project and replace angle-bracketed values. The legal templates are starting points for discussion with a lawyer, not finished contracts.

A.1 Session sheet

```
PROJECT:        <project name>          DATE: <yyyy-mm-dd>
SESSION TYPE:   <tracking / overdub / podcast / ADR / Foley>
LOCATION/ROOM:  <room>                  ENGINEER: <name>
PEOPLE PRESENT: <names and roles>

SYSTEM
  Sample rate / bit depth:  <48 kHz / 24-bit>
  Interface / preamps:      <model, gain settings>
  Buffer (tracking):        <64 samples>   Monitoring: <direct / DAW>

SOURCES
  Track | Source      | Mic / DI        | Position / distance | Notes
  01    | Lead vox    | <LDC cardioid>  | 15 cm, pop filter   |
  02    | Ac. guitar  | <SDC>           | 12th fret, 20 cm    |

TAKES
  Take | Time  | Notes (best lines, problems, noises)
  T01  | 10:42 |
  T02  | 10:51 |

PROBLEMS NOTED:   <noise, clipping, off-mic moments>
ROOM TONE:        <recorded? duration, track>
BACKUP:           <drive 1 / drive 2 / off-site, time, verified by>
```

A.2 Gain-staging and capture check

- Sample rate and bit depth set to project specification before recording.
- Phantom power on only for microphones that need it; ribbons checked.

- Loudest passage rehearsed; peaks between about −18 and −10 dBFS; loudest moments below −6 dBFS.
- No processing printed on the way in unless deliberately chosen and noted.
- Latency acceptable to the performer; neural or high-latency processors off the monitor path.
- Headphones on all remote podcast participants; local recording running on every end.
- At least ten seconds of room tone recorded per location or setup.
- Files named before or immediately after recording.
- Two backups made and verified before the session closes.

A.3 AI run record

Keep one row per AI process applied to material that reaches a deliverable.

```
RUN ID:         <project>-<nnn>
DATE/TIME:      <yyyy-mm-dd hh:mm, time zone>
OPERATOR:       <name>
ROOM/STAGE:     <repair / separate / generate / mix / master / QC>
INPUT FILES:    <names, checksums if used>
TOOL:           <product or model name>
VERSION:        <version number or model date>
WHERE RUN:      <local machine name / cloud service and region>
SETTINGS:       <exported preset or key parameters>
PROMPT/INPUT:   <for generation: full prompt, reference files>
OUTPUT FILES:   <names>
CHECK:          <what was checked, by whom, result>
TERMS:          <for cloud or generative tools: link and saved copy of
                 terms in force on this date>
RIGHTS:         <ownership of inputs; consent reference for any voice>
WATERMARK:      <present / added (scheme) / none / unknown>
DISCLOSURE:     <required? how and where>
```

A.4 Voice consent and licence: key terms

A voice licence should be a signed document, separate from general paperwork, that a person can read and understand. At minimum it should cover the following points.

1. **Parties and identity.** The person whose voice is licensed (the performer), identified and verified, and the licensee. If the performer is represented by an agent or estate, the authority to sign.
2. **What is created.** Whether recordings will be used to create a persistent voice model, a fine-tuned model or zero-shot generations only; the systems or vendors involved.
3. **Permitted uses.** The projects or categories of use, media, languages, territories and whether the voice may be used for real-time or interactive generation.
4. **Content limits.** Whether the voice may say lines the performer has not approved; any prohibited content, such as political, sexual or endorsement content.
5. **Approval.** How outputs will be approved: line by line, by chapter or by episode, and within what time.
6. **Duration.** Start date and end date, and what happens at the end.
7. **Compensation.** Fees for creation of the model and for use, including any per-use, per-project or royalty payments.
8. **Credit and disclosure.** How the performer will be credited and how synthetic use will be disclosed to audiences.
9. **Security.** Where reference recordings and models are stored, who has access and how outputs are watermarked and logged.
10. **Withdrawal and deletion.** How the performer can withdraw consent for future uses, the notice period and the obligation to delete models and reference recordings, with written confirmation.
11. **Union and contract terms.** Reference to any applicable collective agreement or prior contract, and confirmation that this licence complies.
12. **Law and disputes.** Governing law and how disputes will be resolved.

A.5 Client engagement: AI clauses

- The studio may use AI-assisted tools for: . The following require the client's specific written approval: <generation of music, effects or voice; voice cloning or conversion; generative restoration>.
- Client material will not be uploaded to any service that retains it beyond processing or uses it for training, without the client's written consent.
- AI involvement will be disclosed in credits and metadata as follows: , and as required by distributors, platforms and applicable law.
- The client warrants that it holds the rights to all material supplied and has obtained written consent from any person whose voice is to be cloned or converted.
- The studio will keep an AI run record for the project and provide it to the client on request.
- Rendered deliverables will be archived for ; voice models and reference material will be deleted at .

A.6 Dialogue repair check

- Original preserved and aligned under processed track.
- Problems diagnosed and listed with time codes before processing.
- Filters and spectral repair applied before neural processing.
- Neural processing type noted: discriminative / generative.
- Generative reconstruction used only with brief and speaker approval; never for journalism, documentary, legal or academic records.
- Processed passages compared with originals, level-matched.
- Transcript of the original read against the processed audio; changed or doubtful words flagged.
- Natural room tone retained where appropriate.
- Listened on headphones, a laptop speaker and in mono, and for fatigue over a long stretch.
- Processing note written for the producer.

A.7 Separation check

- Separated from best available source, uncompressed.
- At least two models compared per critical stem.
- Each stem soloed and checked for bleed, phasiness, transient smearing and timbre loss.
- Stems rebalanced in context and compared with original at matched loudness.
- Recombination null-tested; residue reviewed.
- Linear-phase processing used where stems are recombined.
- Rights in the original recording and composition confirmed for the intended use.

A.8 Generated asset check

- Prompt, tool, version, date and terms saved.
- Output reviewed for quality on several systems.
- Output checked for resemblance to existing works: melodies, lyrics, recognisable sounds.
- Content identification check run where available.
- Human contributions documented.
- Disclosure requirements identified for each destination.
- Watermarks preserved; content credentials kept where supported.

A.9 Delivery quality control

```
DELIVERABLE:      <name>              SPEC: <document, version, date>
FORMAT:           <WAV/BWF/ADM BW64>  RATE/DEPTH: <48 kHz / 24-bit>
CHANNELS/LAYOUT:  <2.0 / 5.1 / 7.1.4 / Atmos ADM>
NAMING:           <checked against spec>

MEASUREMENTS (BS.1770-5 meter: <name, version>)
  Integrated loudness:      <value> LUFS  Target: <value> ± <tol>
  Dialogue-gated loudness:  <value> LKFS  (if specified)
  True peak:                <value> dBTP  Max: <value>
  Loudness range:           <value> LU    (if specified)

LISTENING
  Full listen-through:      <yes, by whom, date>
  Start and end:            <clicks, tails, noise checked>
  Mono / downmix / re-renders: <checked>
  Encoded preview:          <codec, bitrate; true peak after encoding>
```

```
AI PROCESSING NOTE:       <summary from run records; generative used? Y/N>
ISSUES AND RESOLUTIONS:   <list>
APPROVED BY:              <name, date>
```

A.10 Disclosure wording examples

Adapt wording to platform rules and law in each territory. Examples:

- Music credits: “Vocals: . Additional textures generated with and edited by .” Or, in AI credit fields: “AI used in: instrumentation (partial).”
- Audiobook: “Narrated by , with portions generated using a licensed synthetic version of the author’s voice, approved by the author.”
- Podcast with a synthetic voice segment (audible, at the start of the segment and again at the end): “The following segment uses an AI-generated voice.”
- Documentary processing note to producer: “Dialogue cleaned with AI noise reduction (discriminative). No words were generated or reconstructed.”
- Dubbed content: “This version was dubbed using AI voice technology, with the consent of the original performers.”

Appendix B. Ninety-Day Plan

This plan has two tracks. The beginner track takes someone with a laptop, an interface, a microphone and headphones from first setup to a first release. The professional track takes an established engineer or studio from occasional use of AI tools to a documented, tested, AI-assisted pipeline. Both are thirteen weeks long, and both put listening before tools. Adjust the pace to your circumstances; the order matters more than the dates.

Beginner track

Weeks 1 to 2: ears and room

Set up your space following Chapter 4. Place speakers or choose headphones, and if you can, measure your listening position with a measurement microphone and free software. Choose three reference tracks in the style you want to make and listen to them daily at a moderate, consistent level. Do the listening exercises in Chapter 2: equal loudness, level matching and the null test. Learn to say what you hear in words: boomy, boxy, harsh, thin, distant, wide.

Weeks 3 to 4: signal and capture

Learn your DAW's routing: tracks, inserts, sends, buses and the master. Record your voice and an instrument following Chapter 3, with peaks around −18 to −10 dBFS, at several distances and positions, and compare. Record in your best and worst corners and listen to the difference. Start a session sheet habit and a naming and backup routine from the first session.

Weeks 5 to 6: core processing

Work through Chapter 5 on your own recordings. Practise equalisation with narrow sweeps and broad shelves, compression with different

attack and release settings, a shared reverb on a send and automation of a vocal. Compare every change at matched loudness. Mix a simple song or podcast segment without any AI tools, so that you know what you can do alone.

Week 7: repair and separation

Try neural noise reduction and dialogue isolation on a deliberately noisy recording, following the method in Chapter 8, and null-test the result against the original to hear what was removed. Try separating a song you own the rights to, and check the stems solo, in context and by recombination, following Chapter 9.

Week 8: assistants and generators

Run a mixing assistant and a mastering assistant on your week 6 mix. Audit their suggestions module by module, following Chapter 12. Try a music or sound generator for a sketch or texture, following Chapter 11, and start an AI log for everything you generate, with prompts, dates and saved terms.

Weeks 9 to 10: your release

Write, record and arrange a song or a short podcast episode, following Chapter 13. Keep the melody, lyrics and main performances your own. Record clean. Edit, mix and compare your mix with your references at matched loudness on headphones, speakers, a phone and in a car.

Week 11: master and check

Master your track or have it mastered, and compare automated and human masters fairly if you can. Measure integrated loudness and true peak, listen through a lossy codec and run the delivery check in Appendix A.

Week 12: metadata and delivery

Choose a distributor or podcast host. Prepare credits, splits, identifiers and AI disclosure answers. Register your composition with a collecting society. Claim your artist profiles on streaming services.

Week 13: release and archive

Release. Archive the project: session, raw files, stems, mixes, masters, session sheets, agreements and the AI log. Write a page of notes on what you learned and what you would change, and pick the skill to improve next.

Professional track

Weeks 1 to 2: inventory and policy

List every AI tool currently used in your studio or team, formally or informally, with version, where it runs and its terms. Identify where client material goes when processed. Write a short interim policy: which tool categories are allowed, which need approval, which are prohibited for client material and what must be logged.

Weeks 3 to 4: pipeline map

Write your main pipelines as rooms, following Chapter 21: input, output, tools, check and owner for each. Identify the rooms where AI is already used and those where it could save the most time. Mark the checks that are currently informal or missing.

Weeks 5 to 6: golden set

Build a golden set from your own work: ten to twenty difficult files that represent your typical problems, with notes on good results and expected failures. Run your current tools on the golden set and record baseline results. Make sure the set includes speech in the languages you work in, dense music, sparse acoustic material and at least one recording with overlapping speakers.

Weeks 7 to 8: evaluate and choose

Evaluate candidate tools for your highest-value rooms on the golden set, at matched loudness and blind where possible. Measure real-time factor and, for live use, latency and jitter on your own hardware. Compare local and cloud options for privacy, cost and quality. Choose tools per room and pin versions.

Week 9: records and contracts

Introduce run records for every AI process that reaches a deliverable. Add AI clauses to client engagement letters and to agreements with performers and collaborators, using Appendix A as a starting point and taking legal advice. Set up secure storage and deletion procedures for voice data and models.

Week 10: consent and provenance

Adopt the consent workflow from Chapter 10 for all voice work. Choose a watermarking approach for synthetic voice outputs from your own models. Decide how you will preserve content credentials and watermarks through your pipeline. Prepare disclosure wording for your main distribution territories and platforms, including audible disclosures for EU-distributed synthetic audio where needed.

Week 11: pilot

Run one real project through the new pipeline from start to finish, with full run records and checks. Hold a short review afterwards with everyone involved: what worked, what took longer than expected, which checks caught problems and which were unnecessary.

Week 12: train and document

Update the pipeline documentation from the pilot. Train the team: fundamentals for those who started with AI tools, failure modes and checking methods for those who started before them. Pair engineers across that divide.

Week 13: operate and review

Adopt the pipeline for new projects. Schedule a quarterly review: rerun the golden set on current tools and on new candidates, update terms and policies, check run records for gaps and review any incidents. Assign someone to own the pipeline.

Measures of progress

For the beginner track, progress is a finished release that translates across playback systems, an archive you can reopen and an honest AI

log. For the professional track, progress is a written pipeline, a golden set, run records on every project, contracts that address AI and a team that can explain, for any deliverable, exactly what was recorded, what was processed and what was generated.

Glossary

Definitions are condensed for working use. Where a term has a formal definition in a standard or law, the standard or law governs.

ADM (Audio Definition Model). An ITU metadata model, Recommendation BS.2076, for describing channel-based, object-based and scene-based audio in a common way. Usually carried in Broadcast Wave 64 files for immersive deliverables.

ADR (automated dialogue replacement). Re-recording of dialogue by actors in a studio, in sync with picture, to replace unusable production sound.

Aliasing. Distortion that occurs when frequencies above the Nyquist frequency fold back into the audible range during sampling or digital processing.

Ambisonics. A scene-based spatial audio format that describes a sound field around a point and can be decoded to any loudspeaker layout or rendered binaurally.

Audio-language model. A model that accepts audio, and usually text, and responds in text, able to describe, answer questions about and reason over audio.

Autoregressive model. A generative model that produces a sequence one step at a time, each step predicted from the steps before.

AudioSeal. An open neural audio watermarking method from Meta, published in 2024, designed to localise generated segments within a recording.

Binaural rendering. Processing that filters sounds with head-related transfer functions so that, on headphones, they appear to come from positions around the listener.

BS.1770. ITU-R Recommendation specifying algorithms for measuring programme loudness and true peak; revision 5 dates from November 2023.

Bus. A path in a mixer that combines several signals for shared processing or routing.

C2PA. The Coalition for Content Provenance and Authenticity, and its open specification for signed content credentials that record a file's origin and edit history.

CLAP. Contrastive language-audio pretraining: training audio and text encoders together so that sounds and their descriptions map to nearby points in a shared space, enabling text search and zero-shot classification of audio.

Codec. A system that encodes and decodes audio, usually to reduce data. Lossless codecs preserve every sample; lossy codecs discard information predicted to be inaudible.

Comping. Compiling the best parts of several takes into one performance.

Compression (dynamic range). Reduction of the level of a signal above a threshold, narrowing the range between loud and quiet.

Conditioning. The information that steers a generative model, such as a text prompt, reference audio, melody, timing or video.

Consent workflow. The process of identifying a person, informing them, obtaining written scoped agreement, controlling use and ending use, applied before cloning or converting their voice.

Critical distance. The distance from a source in a room at which direct and reflected sound are equal in level.

DAW (digital audio workstation). Software for recording, editing, mixing and mastering audio.

dBFS. Decibels relative to digital full scale. Zero dBFS is the maximum representable level.

dBTP. Decibels true peak: peak level estimated including peaks between samples.

DCASE. Detection and Classification of Acoustic Scenes and Events, an annual research challenge and workshop in machine listening.

DDEX. Digital Data Exchange, the standards body and family of standards for exchanging music release and rights information between labels, distributors and services.

DDSP (differentiable digital signal processing). Classical signal-processing components, such as oscillators, filters and reverbs, implemented so that they can be trained with gradient-based machine learning, often controlled by a neural network.

Deepfake. Under the EU AI Act, AI-generated or manipulated image, audio or video content that resembles existing persons, objects, places, entities or events and would falsely appear to a person to be authentic or truthful.

De-esser. A processor that reduces harsh sibilant frequencies when they exceed a threshold.

Dialogue-gated loudness. Loudness measured only over passages where speech is detected.

Diarisation. Determining who spoke when in a recording.

Diffusion model. A generative model that learns to reverse a gradual noising process, generating output by starting from noise and removing it step by step.

Discriminative enhancement. Enhancement that filters or masks the input to keep what it judges to be the wanted signal, as opposed to generating a new signal.

Dither. Low-level noise added before reducing bit depth to decorrelate quantisation error from the signal.

DME. Dialogue, music and effects: the three main stem groups of a film or television mix.

Double-ender. A remote recording method in which each participant records their own microphone locally while talking over a call.

Dynamic equaliser. An equaliser band that acts only when the signal in that band crosses a threshold.

EBU R 128. European Broadcasting Union recommendation for loudness normalisation, with a target of −23.0 LUFS and a maximum true peak of −1 dBTP.

Equal-loudness contours. Curves showing the sound pressure levels at different frequencies that listeners judge equally loud, standardised in ISO 226.

Fingerprint (audio). A compact summary of a recording's perceptual content used to identify it by matching against a database.

Flow matching. A generative modelling technique that learns a direct transformation from noise to data, often requiring fewer steps than diffusion.

Foley. Sound effects performed to picture by a Foley artist, such as footsteps, cloth movement and handled props.

Gain staging. Managing signal level at each point in a chain so that each stage operates with appropriate headroom and noise.

Generative restoration. Restoration that generates new audio conditioned on a degraded input, rather than filtering the input; can change words or identity.

Golden set. A fixed collection of difficult files from a studio's own work, used to evaluate new tools and catch regressions.

Grey-box model. A model combining known signal-processing structure with learned components; contrasted with a black-box neural model.

HRTF (head-related transfer function). The filtering that a listener's head, torso and outer ears apply to sound from a given direction; personal to each listener.

IAMF. Immersive Audio Model and Formats, an open immersive audio specification from the Alliance for Open Media, used by Eclipsa Audio.

Impulse response. The response of a system or space to a very short sound, used in convolution reverb and acoustic measurement.

ISRC. International Standard Recording Code, a unique identifier for a sound recording.

K-weighting. The frequency weighting used in BS.1770 loudness measurement.

Latency. Delay between input and output in a system.

Linear-phase equaliser. An equaliser that changes frequency response without phase shift, at the cost of latency and pre-ringing.

LRA (loudness range). A measure of the variation in loudness across a programme, in LU.

LUFS / LKFS. Loudness units relative to full scale; the two terms are equivalent.

M&E (music and effects). A mix without dialogue, used to create dubbed versions.

Masking. Reduction in audibility of one sound caused by another, strongest at nearby frequencies and times.

MCP (Model Context Protocol). An open protocol through which AI agents call tools exposed by other software.

Mel spectrogram. A spectrogram with its frequency axis warped to approximate human frequency resolution.

Mid/side. Representation of stereo as the sum (mid) and difference (side) of left and right channels.

Mimi. The neural audio codec released by Kyutai with its Moshi dialogue model in 2024, running at 12.5 frames per second.

MUSDB18. A standard benchmark dataset for music source separation with four stems: vocals, drums, bass and other.

Neural audio codec. A codec using a learned encoder and decoder to compress audio into discrete tokens, widely used as a basis for generative models.

Null test. Inverting one signal and summing it with another to reveal the difference between them.

Nyquist frequency. Half the sample rate; the highest frequency a sampled system can represent.

Object-based audio. A representation of a mix as audio objects with positional metadata, rendered for the listener's playback system.

Precedence effect. The perception of a single sound from the direction of the first arrival when similar sounds arrive within a short interval.

Procedural audio. Sound generated at runtime from models and parameters rather than played from recordings.

Proximity effect. Increase in bass response of directional microphones as the source gets closer.

Real-time factor. Processing time divided by audio duration.

Residual vector quantisation. A quantisation method in which successive codebooks encode the remaining error of earlier ones, producing stacks of tokens per frame.

Room mode. A resonance of a room at frequencies related to its dimensions, causing peaks and nulls at low frequencies.

RT60. The time for sound in a room to decay by 60 dB after the source stops.

Run record. A log entry describing an AI process applied in a pipeline: tool, version, settings, inputs, outputs, operator, check and, for generation, terms and consent.

Schroeder frequency. The approximate frequency above which a room's modes are dense enough for statistical behaviour to dominate.

SDR (signal-to-distortion ratio). A measure, in decibels, of how close a separated source is to the true source.

Sidechain. Using one signal to control a processor acting on another.

SOFA (AES69). The Spatially Oriented Format for Acoustics, a standard file format for HRTFs and related data.

Source separation. Recovering individual sources, such as vocals or dialogue, from a mixture.

Stem. A submix of related tracks, or a separated source, delivered as a single file.

STFT (short-time Fourier transform). Analysis of a signal into frequency components over short overlapping frames, producing a spectrogram.

SynthID. Google DeepMind's watermarking technology for AI-generated content, applied to Lyria music output.

Text-to-speech. Generation of speech audio from text.

Timbre transfer. Rendering the pitch and dynamics of one performance with the timbre of another instrument or voice.

Token. A discrete unit, in audio usually a codebook index from a neural codec, that a model predicts or processes.

True peak. The estimated maximum level of the continuous waveform, including peaks between samples.

Vocoder (neural). A model that converts a compact representation, such as a mel spectrogram, into a waveform.

Voice cloning. Capturing a specific person's voice so that a system can reproduce it.

Voice conversion. Changing the voice in a recording to sound like another voice while keeping words, timing and performance.

Watermark (audio). An imperceptible signal embedded in audio, detectable by software, used to identify origin or mark synthetic content.

Zero-shot. Performing a task, such as cloning a voice or classifying a sound, without training specifically on that example, typically from a short prompt.

Notes and References

Sources are listed by chapter in the order they are first relied on. Statements about products, models, platforms, court cases and legislation reflect public sources available up to November 2026; product features and figures attributed to companies are those companies' own statements. Lantern Street Sound and Ana are composite, illustrative examples; nothing described about them refers to a real studio or person. Where a standard or paper is cited by title and year without a web address, consult the publisher's current edition.

Preface and Chapter 1

Edison, T. A. "Improvement in Phonograph or Speaking Machines." US Patent 200,521, granted 19 February 1878.

Chapter 2. Sound, Hearing and Numbers

Fletcher, H. and Munson, W. A. "Loudness, its definition, measurement and calculation." *Journal of the Acoustical Society of America*, 1933.

ISO 226, *Acoustics: Normal equal-loudness-level contours* (2003 and later editions).

Moore, B. C. J. *An Introduction to the Psychology of Hearing*. Brill / Emerald, multiple editions.

Nyquist, H. "Certain topics in telegraph transmission theory," 1928; Shannon, C. E. "Communication in the presence of noise," 1949.

Lipshitz, S. P., Wannamaker, R. A. and Vanderkooy, J. "Quantization and dither: a theoretical survey." *Journal of the Audio Engineering Society*, 1992.

Chapter 3. Signal Flow and Gain Staging

Izhaki, R. *Mixing Audio: Concepts, Practices and Tools.* Focal Press, multiple editions.

EBU Tech 3285, Broadcast Wave Format specification.

Chapter 4. Rooms, Monitoring and Acoustics

Sabine, W. C. *Collected Papers on Acoustics.* Harvard University Press, 1922.

Everest, F. A. and Pohlmann, K. C. *Master Handbook of Acoustics.* McGraw-Hill, multiple editions.

Toole, F. E. *Sound Reproduction: The Acoustics and Psychoacoustics of Loudspeakers and Rooms.* Focal Press / Routledge, multiple editions.

Room EQ Wizard (roomeqwizard.com), free acoustic measurement software.

Chapter 5. The Core Processors

Katz, B. *Mastering Audio: The Art and the Science.* Focal Press, multiple editions.

Izhaki, R., as above.

Chapter 6. Loudness and Delivery

ITU-R Recommendation BS.1770-5 (11/2023), *Algorithms to measure audio programme loudness and true-peak audio level.* itu.int.

EBU R 128, *Loudness normalisation and permitted maximum level of audio signals* (v5.0), and EBU Tech 3343, *Guidelines for production of programmes in accordance with EBU R 128*. tech.ebu.ch.

ATSC A/85, *Techniques for Establishing and Maintaining Audio Loudness for Digital Television.* Commercial Advertisement Loudness Mitigation (CALM) Act, 2010.

Free TV Australia, Operational Practice OP-59, *Measurement and management of loudness for TV broadcasting.*

Spotify for Artists, “Loudness normalization” (support.spotify.com), consulted November 2026.

Third-party platform loudness tables (for example riseupmastering.com, 2026) noting that Apple Music and YouTube do not publish official LUFS targets.

Netflix Partner Help Center and Studio Knowledge Hub, sound mix specification and loudness guidance (−27 LKFS dialogue-gated ±2 LU; −2 dBTP; Atmos 5.1 re-render), consulted November 2026.

Audio Engineering Society, TD1004 (2015) and TD1008 (2021), recommendations for loudness of internet audio streaming and on-demand distribution.

Chapter 7. How Audio Models Hear and Speak

van den Oord, A. et al. “WaveNet: A generative model for raw audio,” 2016.

Kong, J., Kim, J. and Bae, J. “HiFi-GAN,” 2020.

Baevski, A. et al. “wav2vec 2.0,” 2020; Hsu, W.-N. et al. “HuBERT,” 2021; Chen, S. et al. “WavLM,” 2021.

Zeghidour, N. et al. “SoundStream: An end-to-end neural audio codec,” 2021. Défossez, A. et al. “High fidelity neural audio compression” (EnCodec), 2022. Kumar, R. et al. “High-fidelity audio compression with improved RVQGAN” (DAC), 2023.

Défossez, A. et al. “Moshi: a speech-text foundation model for real-time dialogue,” Kyutai, 2024 (Mimi codec).

Della Libera, L. et al. “FocalCodec-Stream: Streaming low-bitrate speech coding via causal distillation,” arXiv:2509.16195, September 2025.

“HH-Codec: High compression high-fidelity discrete neural codec for spoken language modeling,” arXiv:2507.18897, July 2025.

“PhoenixCodec: Taming neural speech coding for extreme low-resource scenarios,” arXiv:2510.21196, October 2025 (LRAC 2025 Challenge).

“ContextCodec: Content-focused context guidance for ultra-low bitrate speech coding,” arXiv:2606.10591, June 2026.

Borsos, Z. et al. “AudioLM,” 2022. Copet, J. et al. “Simple and controllable music generation” (MusicGen), 2023. Wang, C. et al. “Neural codec language models are zero-shot text to speech synthesizers” (VALL-E), 2023. Liu, H. et al. “AudioLDM,” 2023. Le, M. et al. “Voicebox,” 2023. Chen, Y. et al. “F5-TTS,” 2024.

Kilgour, K. et al. “Fréchet Audio Distance,” 2019. ITU-R BS.1534 (MUSHRA).

Chapter 8. Repair, Restoration and Dialogue Cleanup

Valin, J.-M. “A hybrid DSP/deep learning approach to real-time full-band speech enhancement” (RNNoise), 2018.

Reddy, C. K. A. et al. Deep Noise Suppression Challenge papers, Microsoft, 2020 onward.

Schröter, H. et al. “DeepFilterNet,” 2022.

Chapter 9. Taking Mixes Apart

Rafii, Z. et al. MUSDB18 and MUSDB18-HQ datasets, 2017 and 2019.

Hennequin, R. et al. “Spleeter: a fast and efficient music source separation tool,” Deezer, 2019/2020. Stöter, F.-R. et al. “Open-Unmix,” 2019.

Défossez, A. “Hybrid spectrogram and waveform source separation,” 2021. Rouard, S., Massa, F. and Défossez, A. “Hybrid transformers for music source separation,” 2022/2023.

Luo, Y. and Yu, J. “Music source separation with band-split RNN,” 2022.

Lu, W.-T., Wang, J.-C., Kong, Q. and Hung, Y.-N. “Music source separation with band-split RoPE transformer,” arXiv:2309.02612; and the SAMI-ByteDance system description for the Sound Demixing Challenge 2023. Reported median SDR on MUSDB18-HQ: 9.80 dB without extra data; 11.99 dB with 500 extra songs; listening-test observations on preference.

SCNet and SCNet-large results as tabulated on MUSDB18-HQ leaderboards (snapshot July 2025).

Mitsufuji, Y. et al. “Music Demixing Challenge 2021”; Fabbro, G. et al. “The Sound Demixing Challenge 2023.” Petermann, D. et al. “The cocktail fork problem” (Divide and Remaster dataset), 2022.

Meta AI. “Introducing SAM Audio: the first unified multimodal model for audio separation,” 16 December 2025; github.com/facebookresearch/sam-audio.

iZotope. Ozone 12 documentation: Master Assistant and Stem EQ (docs.izotope.com), consulted November 2026.

The Beatles, “Now and Then,” November 2023; public statements on the use of machine-learning audio separation developed for *The Beatles: Get Back* (2021).

Chapter 10. Voices

Shen, J. et al. “Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions” (Tacotron 2), 2017. Wang et al. (VALL-E), 2023; Chen et al. (F5-TTS), 2024, as above.

ElevenLabs. “Eleven v3 is now generally available,” February 2026 (elevenlabs.io/blog).

Zyphra. ZONOS3-Mini technical report, arXiv:2611.16525, and Hugging Face blog post, November 2026.

Reporting on “Heart on My Sleeve,” April 2023; Holly Herndon, Holly+ (2021); Grimes, Elf.Tech (2023).

Tennessee ELVIS Act, effective 1 July 2024 (see Chapter 20).

Chapter 11. Generative Music and Sound Effects

Dhariwal, P. et al. “Jukebox: A generative model for music,” OpenAI, 2020. Agostinelli, A. et al. “MusicLM,” 2023. Copet et al. (MusicGen), 2023. Evans, Z. et al. “Stable Audio Open,” 2024.

Suno. “Introducing v6,” release notes, 9 September 2026. Reuters, “Suno releases new AI music models in partnership with Warner Music, BMG,” 9 September 2026. *Los Angeles Times*, 21 September 2026, on licensing deals.

Google. “Introducing Lyria 3.5 in Google Flow Music,” 29 July 2026; Google DeepMind Lyria model page (Lyria RealTime, Magenta RealTime, SynthID watermarking).

Stability AI. “Stability AI introduces Stable Audio 2.5”; Universal Music Group and Stability AI strategic alliance announcement, October 2025.

ElevenLabs. Eleven Music (August 2025); “Introducing Music v2,” 26 May 2026.

Udio. “Udio with Warner Music Group,” 19 November 2025, and help-centre articles on transitional terms.

US Copyright Office. *Copyright and Artificial Intelligence, Part 2: Copyrightability*, January 2025.

Spotify. “One year in: putting human artistry first on Spotify,” 20 October 2026 (AI Credits launched April 2026).

Deezer. “AI music has surpassed 50% of new music uploads for the first time,” 21 July 2026.

Kreuk, F. et al. “AudioGen,” 2022. Cheng, H. K. et al. “MMAudio: Taming multimodal joint training for high-quality video-to-audio synthesis,” CVPR 2025. EchoFoley and FoleyDirector papers, CVPR 2026 open access. Google, Veo 3 announcement, May 2025.

Chapter 12. Machines at the Mix Bus

Dugan, D. Automatic microphone mixing, 1975 (*Journal of the Audio Engineering Society*).

De Man, B., Reiss, J. D. and Stables, R. *Intelligent Music Production.* Focal Press, 2019.

Steinmetz, C. J., Pons, J., Pascual, S. and Serrà, J. “Automatic multitrack mixing with a differentiable mixing console of neural audio effects,” ICASSP 2021.

Martínez Ramírez, M. A. et al. “Automatic music mixing with deep learning and out-of-domain data,” ISMIR 2022.

Steinmetz, C. J., Bryan, N. J. and Reiss, J. D. “Style transfer of audio effects with differentiable signal processing,” 2022.

iZotope. Ozone 12 Master Assistant documentation; Neutron 5 product information, consulted November 2026.

Audiokinetic. “Introducing Audiokinetic’s Wwise MCP,” 24 September 2026.

Chapter 13. From Bedroom to First Release

DDEX. Electronic Release Notification (ERN) 4.3.2 release notes and “ContainsAI” allowed-value set (service.ddex.net).

Spotify. “Spotify strengthens AI protections for artists, songwriters, and producers,” 25 September 2025; “One year in,” 20 October 2026.

Chapter 14. Dialogue, Podcast and Post-Production Sound

Radford, A. et al. “Robust speech recognition via large-scale weak supervision” (Whisper), 2022.

Netflix Partner Help Center, sound mix specifications and “BWAV ADM Creation Guidelines for Dubbing,” consulted November 2026.

SAG-AFTRA, 2023 TV/Theatrical Agreement digital replica provisions; 2025 Interactive Media (Video Game) Agreement (see Chapter 20).

Chapter 15. Game and XR Audio

Farnell, A. *Designing Sound.* MIT Press, 2010.

Audiokinetic Wwise and Firelight FMOD documentation; Epic Games, MetaSounds documentation; Valve, Steam Audio; Meta XR Audio SDK; Microsoft Project Acoustics.

Audiokinetic, “Introducing Audiokinetic’s Wwise MCP,” 24 September 2026.

Reuters, “Industry video game actors pass agreement with studios for AI security,” 10 July 2025.

Chapter 16. Neural DSP and Real-Time AI

Engel, J., Hantrakul, L., Gu, C. and Roberts, A. “DDSP: Differentiable digital signal processing,” ICLR 2020.

Comunità, M., Steinmetz, C. J. and Reiss, J. D. “NablAFx: A framework for differentiable black-box and gray-box modeling of audio effects,” arXiv:2502.11668, 2025.

Caillon, A. and Esling, P. “RAVE: A variational autoencoder for fast and high-quality neural audio synthesis,” 2021.

Caspe, F., Shier, J., Sandler, M., Saitis, C. and McPherson, A. “Designing neural synthesizers for low-latency interaction,” arXiv:2503.11562, accepted for the *Journal of the Audio Engineering Society*, 2025.

Neutone SDK documentation (github.com/neutone/neutone_sdk).

Chowdhury, J. “RTNeural: Fast neural inferencing for real-time systems,” 2021.

Franchino, F. and Schlecht, S. J. “Compiling differentiable audio graphs to real-time DSP” (ADAC), arXiv:2606.21277, 2026.

Wright, A., Damskägg, E.-P. and Välimäki, V. “Real-time black-box modelling with recurrent neural networks,” DAFx 2019, and related Aalto University papers. Neural Amp Modeler (S. Atkinson), open source.

Chapter 17. Machine Listening

Wang, A. “An industrial-strength audio search algorithm,” ISMIR 2003.

Gemmeke, J. F. et al. “Audio Set: An ontology and human-labeled dataset for audio events,” ICASSP 2017.

Kong, Q. et al. “PANNs,” 2020. Gong, Y., Chung, Y.-A. and Glass, J. “AST: Audio spectrogram transformer,” 2021. Chen, S. et al. “BEATs,” 2022. Elizalde, B. et al. “CLAP,” 2022/2023; Wu, Y. et al. LAION-CLAP, 2023. Bittner, R. et al. “A lightweight instrument-agnostic model for polyphonic note transcription” (Basic Pitch), 2022.

Deshmukh, S. et al. “Pengi,” 2023; Gong, Y. et al. “LTU: Listen, think and understand,” 2023; Tang, C. et al. “SALMONN,” 2023; Chu, Y. et al. “Qwen-Audio,” 2023.

NVIDIA ADLR. Audio Flamingo 3 project page.

Sakshi, S. et al. “MMAU: A massive multi-task audio understanding and reasoning benchmark,” ICLR 2025. “MMAU-Pro,” AAAI 2026.

DCASE 2026 Challenge task descriptions (dcase.community/challenge2026).

Chapter 18. Spatial, Immersive and Object-Based Audio

Blumlein, A. D. British Patent 394,325 (filed 1931).

Gerzon, M. A. “Periphony: With-height sound reproduction.” *Journal of the Audio Engineering Society*, 1973.

ITU-R BS.2076, Audio Definition Model; ITU-R BS.2088, BW64.

ISO/IEC 23008-3, MPEG-H 3D Audio. Alliance for Open Media, IAMF specification, 2023.

Google Open Source Blog. “Introducing Eclipsa Audio: immersive audio for everyone,” January 2025; Samsung newsroom items on Eclipsa Audio, 2025–2026.

Apple Support. “Listen with Personalized Spatial Audio for AirPods and Beats”; Apple privacy notice on Personalized Spatial Audio.

AES69, SOFA file format for spatial acoustics.

Chapter 19. Provenance, Watermarks and Detection

C2PA Technical Specification (c2pa.org).

San Roman, R. et al. “Proactive detection of voice cloning with localized watermarking” (AudioSeal), ICML 2024; github.com/facebookresearch/audioseal.

Liu, H. et al. “AudioMarkBench: Benchmarking robustness of audio watermarking,” arXiv:2406.06979, 2024.

ASVspoof challenge series, 2015 onward.

Deezer, 21 July 2026, as above. Spotify, 20 October 2026, as above.

European Commission. “Transparency obligations under Article 50 of the AI Act” (FAQ); Code of Practice on Transparency of AI-generated Content; EU icons for labelling AI-generated content (digital-strategy.ec.europa.eu), consulted November 2026.

Chapter 20. Rights, Consent and Copyright

Music Business Worldwide. “Universal and Warner press court to end US musicians’ lawsuit over Udio, Suno deals,” 17 September 2026; “Universal and Sony fight Suno’s bid to freeze second lawsuit,” 17 November 2026.

Warner Music Group. “Warner Music Group and Suno forge groundbreaking partnership,” 25 November 2025.

US Copyright Office. *Copyright and Artificial Intelligence, Part 3: Generative AI Training* (pre-publication version), 9 May 2025; Part 2, January 2025.

Munich Regional Court I. GEMA v OpenAI, judgment of 11 November 2025 (press release 11/2025); GEMA v Suno, Case 42 O 763/25, judgment of 31 July 2026 (Reuters, 31 July 2026; EU IP Helpdesk, August 2026).

UK Government. *Report on Copyright and Artificial Intelligence* (gov.uk, 2026); Music Week and UK Music on the government’s decision, March 2026.

Tennessee Code Annotated §47-25-1101 et seq., as amended by the ELVIS Act (2024).

Congress.gov. NO FAKES Act of 2025 (S.1367; H.R.2794) and NO FAKES Act of 2026 (S.4591; H.R.8915). *Forbes,* 30 September 2026, on the blocked unanimous-consent request.

Folketingstidende. L 49 (2026–27), bill amending the Danish Copyright Act on protection of personal characteristics, introduced 8 October 2026.

SAG-AFTRA. “SAG-AFTRA members approve 2025 video game agreement,” July 2025, and agreement summary.

Thaler v. Perlmutter, US Court of Appeals for the D.C. Circuit, 2025.

Chapters 21 and 22

Kyutai, Moshi (2024), as above. NVIDIA, Fugatto announcement, November 2024 (Reuters, 25 November 2024). Meta, SAM Audio, December 2025, as above.

“Watermarks and fingerprints as soft bindings for content provenance: An open-licence benchmark for images, audio and video,” arXiv:2610.04151, 2026.

Index

ADM, 48, 122, 157, 211, 225
ADR, 84, 121, 145, 211
aliasing, 15, 39, 211
ambisonics, 155, 211
Atmos, 46, 122, 156, 221
ATSC A/85, 43, 220
Audio Flamingo, 148, 192, 226
AudioSeal, 164, 211, 227
AudioSet, 147
automixer, 101
BEATs, 148, 226
binaural, 14, 129, 155, 211
BRAVE, 141, 193
BS-RoFormer, 75
BS.1770, 44, 159, 212, 220
C2PA, 164, 195, 212, 227
CLAP, 58, 148, 212, 226
codec, 5, 14, 47, 53, 64, 78, 82, 95
comping, 111, 212
compression, 11, 23, 36, 49, 55, 114, 120, 164
consent, 6, 81, 115, 121, 130, 143, 173, 184
critical distance, 12, 212
DAC, 55, 221
DAW, 4, 16, 22, 49, 60, 73, 93, 103
DCASE, 148, 160, 194, 213, 227
DDEX, 48, 96, 115, 213, 225
DDSP, 140, 213, 225
de-esser, 38, 114, 213
Deezer, 74, 96, 115, 151, 166, 222
Demucs, 74
dialogue-gated, 45, 122, 213, 221
diarisation, 151, 183, 213
diffusion, 28, 57, 92, 213
dither, 16, 48, 213, 219
double-ender, 119, 186, 213
dynamic equaliser, 36, 213
EBU R 128, 214, 220
Eclipsa Audio, 156, 214, 227
ELVIS Act, 86, 178, 223
EnCodec, 55, 221
equal-loudness, 13, 214, 219
Fingerprint, 165, 214
flow matching, 58, 214
Foley, 3, 97, 121, 180, 214
gain staging, 4, 19, 214
GEMA, 175, 228
golden set, 188, 195, 207, 214
grey-box, 140, 214
HRTF, 14, 157, 194, 214
IAMF, 156, 214, 227
impulse response, 39, 123, 142, 214
ISRC, 48, 115, 214
K-weighting, 44, 215
latency, 7, 22, 36, 53, 65, 77, 83, 95
linear-phase, 36, 77, 203, 215
loudness range, 44, 125, 153, 215
LUFS, 12, 44, 107, 114, 120, 214, 221
Lyria, 92, 130, 164, 217, 224
M&E, 122, 215
masking, 9, 13, 37, 55, 76, 104, 111, 125
MCP, 215, 225
mel spectrogram, 14, 54, 82, 215, 223
mid/side, 215
Mimi, 55, 192, 215, 221
MMAU, 149, 160, 194, 226
Moshi, 55, 192, 215, 221
MUSDB18, 73, 215, 222
MUSHRA, 59, 222
NablAFx, 140, 226
Netflix, 46, 122, 157, 221
Neutone, 141, 226
NO FAKES, 178, 228
null test, 13, 184, 205, 215
Nyquist, 15, 211, 219

Ozone, 78, 102, 223
precedence effect, 14, 40, 216
procedural audio, 129, 216
proximity effect, 20, 216
RAVE, 141, 226
real-time factor, 59, 188, 207, 216
RNNoise, 65, 143, 222
room mode, 28, 216
RT60, 27, 216
run record, 185, 202, 208, 216
SAG-AFTRA, 124, 178, 225
SAM Audio, 75, 192, 223
Schroeder frequency, 28, 216
SDR, 59, 75, 194, 216, 222
SOFA, 158, 216, 227
Spleeter, 74, 222
Spotify, 45, 96, 114, 148, 163, 221
Stable Audio, 57, 92, 223
stem, 5, 24, 36, 47, 58, 73, 93, 103
STFT, 54, 217
Suno, 92, 166, 174, 223
SynthID, 92, 164, 217, 224
timbre transfer, 140, 217
true peak, 44, 114, 120, 153, 159, 184, 206, 212
Udio, 92, 166, 174, 186, 224
VALL-E, 57, 82, 222
vocoder, 54, 82, 166, 217
voice cloning, 81, 120, 164, 202, 217, 227
voice conversion, 81, 124, 134, 143, 217
watermark, 8, 87, 163, 195, 203, 208, 217, 228
WaveNet, 53, 82, 221
Whisper, 120, 148, 225
Wwise, 103, 127, 225
zero-shot, 81, 201, 212, 222