SYNPHONOS

HOW TO USE IT

VERSION 1.0.0 · WINDOWS

A TIMECODE GENERATOR THAT PLAYS THE SHOW

CONTENTS

Every control in the window, in the order you meet it. If you are standing in front of a show that starts in ten minutes, read page 3, then page 32.

STARTING

What Synphonos does 3
Installing it, and the first run 4
The start-up animation 5
The window at a glance 6

PLAYING

The readout and the file line 7
The transport row 8
Moving the playhead 9
The waveform 10
Zoom, height and follow 11
Speed 12
Rate, offset, pause and jump 13

THE SHOW

The set list 14
Show files and the backup rig 15
Tracks, or stems 16
Marks 17

GETTING TIMECODE OUT

Device and channels 18
LTC, the audio timecode 19
MTC, and a MIDI port on Windows 20

RUNNING IT FROM THE DESK

Keys 21
Keys that work from other apps 22
The lock 23

THE LOOK

Appearance 24
The six looks 25

KNOWING WHAT IT IS DOING

The status bar 26
About, diagnostics, crash log 27
Where things are saved 28
What comes out when 29

WHEN IT GOES WRONG

If something goes wrong 30
What it deliberately does not do 31

REFERENCE

Every key, on one page 32
Licence and credits 33

ABOUT THE SHOW IN THESE PICTURES

Every screenshot was taken from a real, running copy. The show in it — Winter Gala 2026, with Haze Machine Blues and It Worked In Rehearsal on the set list — is not. The audio files are placeholders, named after things that have happened to all of us.

WHAT IT DOES

Synphonos plays an audio file and sends SMPTE timecode locked to where that file has got to.
Press Play, and the console follows.

A lighting or video desk that runs from timecode needs something to tell it what time it is. Synphonos is that something, and it is also the thing playing the music — which is the point. The timecode is not synchronised to the audio; it is counted from it, one sample at a time, in the same audio callback that fills the output buffer. There is nothing to line up and nothing that can drift.

It sends timecode two ways, and you can use either or both at once:

- LTC — longitudinal timecode, as audio, on a channel of the same output device. Plug that channel into the desk's timecode input.
- MTC — MIDI timecode, as quarter-frame messages, out of a MIDI port. On Windows that usually means a virtual port such as loopMIDI, to another program on the same machine.

It generates timecode. It never reads or chases it.

WHAT IT IS NOT

Synphonos is a generator only. It will not chase timecode coming in from somewhere else, it will not free-run without a file, and it will not loop. If you need a machine that follows another machine, this is the wrong tool — and page 31 is an honest list of everything else it does not do.

WHAT YOU NEED

- A Windows 10 or 11 machine, 64-bit.
- An audio output with a spare channel for LTC — or a MIDI port, for MTC.
- A desk that reads timecode, set to the same frame rate.
- Your show's audio, in any ordinary format.

No installer, no runtime, no licence key, no internet connection.

THE FIRST RUN

There is no installer. Put the two files somewhere sensible and double-click the program.

INSTALLING

The download holds `synphonos.exe` and `LICENCE.txt`. Keep them together — the licence asks anyone passing the program on to include it — and put them wherever you keep show software. A folder on the show machine's desktop is fine.

The program is one self-contained file. Nothing is installed, no runtime is needed, no registry keys are written, and removing the folder removes it.

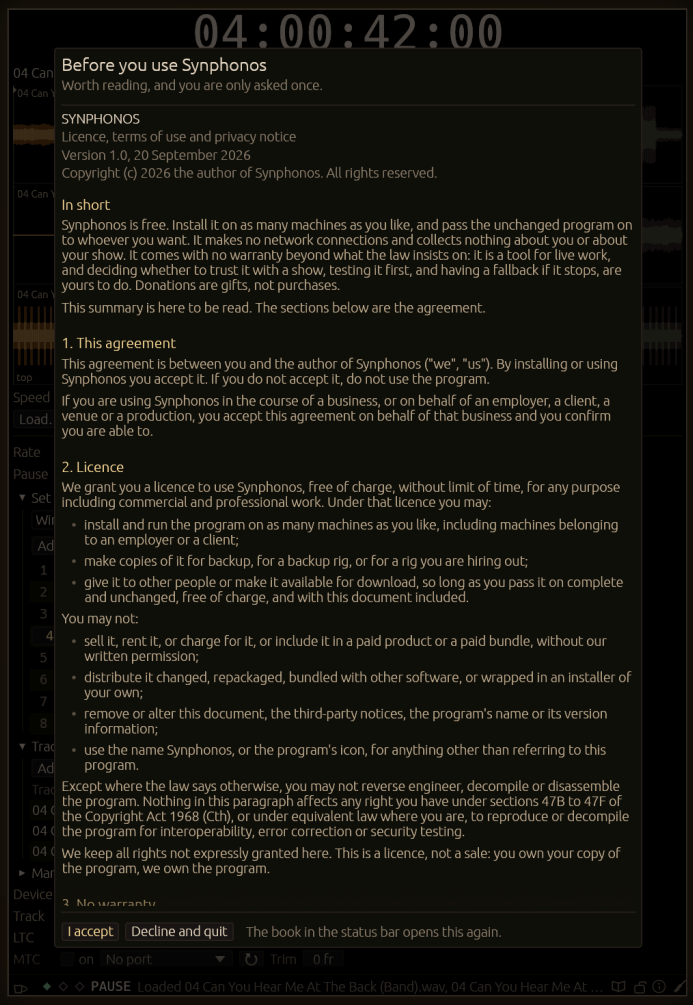
Windows has not seen this program before, so SmartScreen may warn you the first time. More info, then Run anyway.

THE AGREEMENT

The first time you run it, after the animation hands over, the licence, terms and privacy notice appear as a gate. It has to be accepted before the program can be used, and you are only asked once.

Worth knowing what you are agreeing to: the program is free, you may use it commercially, it comes with no warranty, and deciding whether to trust it with a show is your call. It makes no network connections and collects nothing.

Afterwards the book in the status bar opens the same document whenever you want it.



The first run. The gate cannot be dismissed or moved past — I accept or Decline and quit. Acceptance is remembered; deleting the settings file asks again.

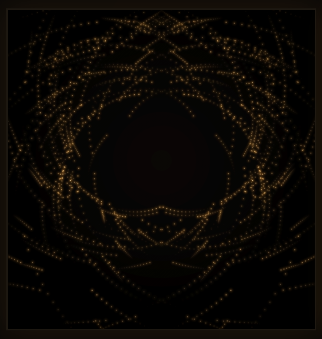
WHAT IT WRITES ON YOUR MACHINE

Its own settings and your show files, in `%APPDATA%\tcgen\config`, and nothing else. The folder keeps the program's old name on purpose, so a machine that ran an earlier version keeps its settings (page 28).

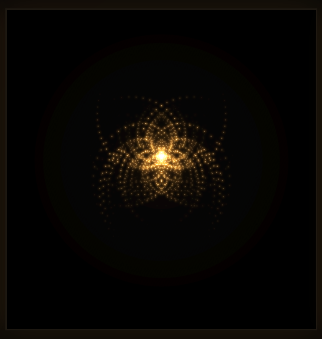
To remove it: delete the folder you put it in, and that config folder if you want the settings gone too. There is nothing else to clean up.

THE START-UP

Every launch opens on a short animation, and waits on click to begin. It is not a loading screen: the show is already open behind it.



Motes spiral inward



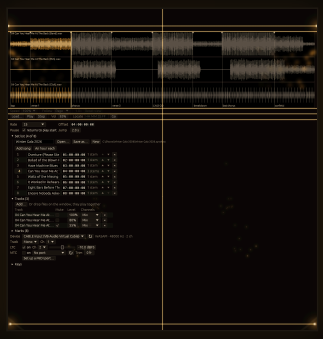
They gather, and pinch

The diamond strikes

Lines and arcs draw

The hands come up

And it waits for you



A click: the square becomes the track

The lines become the window

GETTING PAST IT

A click anywhere continues. So does a key that works from other apps, and so does dropping a file on the window — if the desk operator hits Next song before anyone has touched the mouse, the program simply carries on.

Keys pressed in its own window do nothing until it has handed over, so a stray Space cannot start playback into a dark room.

TURNING IT DOWN, OR OFF

All of it is optional, under the paintbrush (page 24):

- Play when the window opens — off, and the window opens straight into the controls.
- Play the 5 s intro — off, and it opens at the waiting loop.
- Show the name, Follow the accent colour.

Behind all of it, the last session, the audio device and the MIDI port are restored from the moment the program starts. Nothing waits for the click.

WHY THERE IS AN ANIMATION AT ALL

A show machine spends most of its life sitting on a table doing nothing, and the moment someone needs it they need to know, from the other side of a dark room, that it is awake and it is the right program. A still splash would do that. This one also gives you a deliberate click to begin before any key does anything — which is the useful part, because the most dangerous moment for a show machine is the one where somebody leans on the space bar to see if it is on.

AT A GLANCE

The whole window, with its four foldable sections closed. Everything below the transport row scrolls; the triangles open and close.

1 The readout — the timecode going out, page 7

2 What is loaded, and how long it is, page 7

3 The waveform, one strip per stem, page 10

4 Speed, follow mode and zoom, pages 11–12

5 Load, Play, Stop, Vol, Locate, page 8

6 Frame rate, start offset, pause, jump, page 13

7 The set list — the whole evening, page 14

8 The stems that play together, page 16

9 Marks in this song, page 17

10 Outputs: device, channels, LTC, MTC, page 18

11 Keys, page 21

12 The status bar, page 26

THE READOUT

The big number is the timecode leaving the program right now. Its colour tells you the state of the transport from across a dark room.

Playing. Pale gold, and lit: the glow fades in over a quarter of a second and then breathes, slowly, on a five-second cycle. Timecode is going out.

Paused. Ivory, unlit. The position is held; nothing is sent.

Stopped. Bronze, and back at the song's start offset.

READING IT

The readout is `HH:MM:SS:FF` — hours, minutes, seconds, frames. At a drop-frame rate the last colon becomes a semicolon, `10:00:00;12`, which is the standard way of saying so and the quickest way to catch a rate set wrong.

THE FILE LINE

The first stem's name, how many more play with it, the position and length in plain minutes and seconds, and the file's own sample rate.

The line under the readout names what is loaded. With several stems it names the first and counts the rest, because they are one performance and share one transport. The time is track time, not timecode: useful when someone asks how long the song is.

With nothing loaded it reads No file: drop audio here or use Load….

A different song, loaded from the set list: its own start offset is on, so the readout jumps to where that song sits in the evening (page 14).

THE READOUT IS THE TIMECODE, NOT THE FILE

It shows the position in the show — the song's start offset plus how far into the song you are. A song that starts at `04:00:00:00` reads `04:00:42:00` forty-two seconds in. The plain minutes and seconds on the line below are the position in the file, which is a different question and usually a less useful one.

THE TRANSPORT

Everything you touch during a show, on one row.

Load…	Replaces this song's stems through a file picker. Whatever was loaded goes. To add a stem instead, drop it on the window.
Play	Starts from wherever the playhead is. Becomes Pause while playing. Default key `Space`.
Pause	Stops sending, holds the position, and sends receivers one full-frame message so they park in the right place.
Stop	Returns to the song's start offset. Default key `Esc`. While the show is locked it asks twice (page 23).
Vol	Output level of the music, 0 to 100%, saved with the machine. Safe to change mid-show: the callback ramps across one buffer, so it never clicks. Timecode is not affected — LTC keeps its own level, which is what receivers lock to.
Locate	Type a timecode, press `Enter` or Go.

LOADING AUDIO

Drop files on the window and they are added as stems that play together — the window lights up while a file hovers, and several dropped at once all go in. Load… replaces instead.

WAV, AIFF, FLAC, Ogg Vorbis, MP3, AAC (`.m4a`, `.mp4`, `.aac`) and MKV/WebM audio. A file that will not open is refused with the reason in the status line, and what was loaded stays loaded.

LOCATING

A timecode typed in, ready for Go.

Locating lands inside the file and never changes the transport state: stopped stays stopped, paused stays paused. While playing it jumps and carries on.

Playing: the button has become Pause, and Stop is live.

TWO LEVELS, AND ONLY ONE OF THEM IS YOURS TO RIDE

Vol is the music. The LTC level on the Outputs row (page 19) is the timecode, and nothing on this row touches it. Pull the music to nothing for a quiet moment and the desk stays locked.

NOTHING HERE NEEDS THE MOUSE

Play/Pause is `Space`, Stop is `Esc`, and both — along with everything else on this page — can be given a key that works from the lighting console (page 22).

MOVING ABOUT

Six ways to get somewhere, all of them safe while playing.

MOVE	KEY	WHAT IT DOES
Jump back / forward	← →	Moves by the Jump setting, 0.1 to 600 seconds of track time. Hold to repeat.
Previous / next frame	, .	One timecode frame, landing exactly on the boundary. Hold to repeat.
Previous / next mark	PgUp PgDn	Between the named points in this song (page 17).
Back to play start	Backspace	To where Play was last pressed, without pausing.
Go to start	Home	To the start of the file, without stopping.
Click the waveform	—	Straight there. Drag and release to be fussy about it.

JUMP IS TRACK TIME, NOT REAL TIME

A 5-second jump moves 5 seconds of timecode. At 50% speed that takes ten seconds of wall clock to play back, which is what you want: cues live in timecode.

PAUSE RETURNS TO PLAY START

Tick returns to play start beside Pause and pausing takes the playhead back to where Play was last pressed. Press Play, run a section, jump about inside it, pause — and you are back at the top of that section, ready to go again.

Jumps and waveform clicks made while playing do not move that point. It shows on the waveform as a gold diamond, and `Backspace` goes there without pausing.

MTC receivers get their pause full-frame at the play start, not at the place you pressed pause, so the desk lands where you do.

WHILE PLAYING

Every one of these is a locate. LTC goes silent for a few audio buffers while the decoder refills, then runs from the new position; MTC receivers get a full-frame at the new place and then quarter-frames again. Held keys repeat, and each repeat is another locate — so a desk sees a series of positions, not a smear.

WHILE STOPPED OR PAUSED

Wherever you leave the playhead is where Play will start, and receivers are told about it straight away.

THE WAVEFORM

Three stems of one song, sharing a playhead and a view. Gold behind the playhead is what has been played; bronze ahead of it is what has not. The diamonds along the top are marks, and the row underneath shows the zoom.

CLICKING IT

Click anywhere on the strip to put the playhead there. Hovering shows the timecode a click would land on, which is the quickest way to answer "what time is the second chorus".

Dragging moves the playhead when you let go, not continuously, so receivers see one locate rather than a stream of them.

While stopped or paused, where you click is where Play will start. While playing, playback jumps there and carries on.

FILLING IN

The strip is drawn from its own low-priority scan of the file, so it fills in from the left over a second or two after loading. Playing does not wait for it.

SEVERAL STEMS

Each stem gets its own strip, stacked in the order they were added and named. They cannot be moved against each other — they are one performance. Stems of different lengths are fine: short ones simply stop, and the transport runs to the end of the longest.

WHAT IS DRAWN ON IT

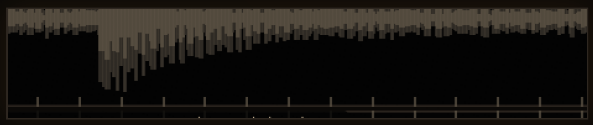
The playhead	A lit gold line, glowing while playing.
A gold diamond	The play start — where Play was last pressed, and where Pause returns to.
Ivory diamonds	Marks, with their names along the bottom.
Faint ticks	Timecode frame boundaries, once you are zoomed in far enough for them to mean anything.

ZOOM AND FOLLOW

Three modifier-scroll gestures over the waveform, and one drop-down that decides what the view does while the show runs.

GESTURE	OVER THE WAVEFORM
`Ctrl` + scroll	Zoom in and out around the pointer, down to 24 points per timecode frame.
`Shift` + scroll	Move along the track. This also stops the view following until the next Play.
`Alt` + scroll	Make the strip taller or shorter, 40 to 400 points. Saved with the look.

16×. Individual hits, and a mark coming up. The row under the strip always shows the zoom, and Reset view puts the whole track back.



Frame boundaries. Once frames are at least 6 points apart, ticks appear top and bottom of each strip — longer ones on whole seconds.

FOLLOW

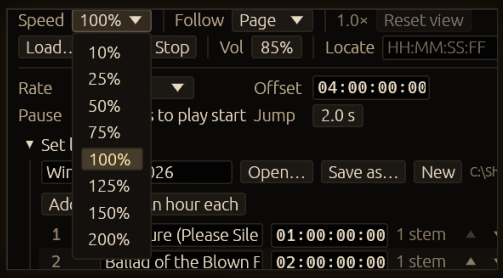
What a zoomed-in view does while playing:

- Page — jumps a screen ahead when the playhead runs off the edge. The default, and the calmest to watch.
- Smooth — keeps the playhead in the middle and scrolls the track past it.
- Stay put — never moves by itself.

Zoom, position and follow mode are not saved. They reset when the program starts, and zoom resets for each new song, because a view that made sense last night rarely makes sense tonight.

SPEED

Eight presets, from a tenth of speed to double. It works like a tape machine: slower is lower, faster is higher.



10, 25, 50, 75, 100, 125, 150, 200%.

This is for tech rehearsals. Run a song at half speed and the desk operator gets twice as long to see each cue land, with the timecode following the track exactly: one second of timecode takes two seconds, LTC comes out at half speed, and MTC quarter-frames arrive half as often.

Speed can be changed while playing. Playback carries on from the same place after a short gap to re-buffer — the same gap as a jump — and receivers get a full-frame at that place.

A speed that would need a fractional sample rate on the current device is greyed out rather than offered.

Whenever it is not 100% the figure shows in rose, in the drop-down and again beside the transport state in the status bar, so a rehearsal speed cannot be left on by accident.

IT IS NOT SAVED

Playback always starts at 100%. Closing the program is enough to undo a rehearsal speed; so is pressing the preset back.

WHY THE PITCH MOVES

Keeping the pitch would mean time-stretching, and a time-stretcher puts a variable delay between the audio and the timecode counted from it. Resampling the track instead keeps LTC and audio locked sample for sample at every speed, with exact whole-number maths throughout. That is the trade, and for finding cues it is the right way round.

Receivers vary in how far they will follow. At 10%, LTC bits are ten times longer than normal and MTC quarter-frames land 100 ms apart; some readers will not lock to that. If the desk drops out at a crawl, try 25%.

RATE & OFFSET

Two settings that decide what the numbers mean, and two that decide how the keys behave.

The four settings under the transport row.

One rate for the whole show.

RATE

24, 25, 29.97 NDF, 29.97 DF or 30 frames per second. Set it to whatever the receiving desk is set to and leave it alone: a rate mismatch shows up as cues drifting slowly through the night, which is a miserable thing to debug at half past eleven.

The rate belongs to the show, not the machine, so it travels with the show file. It can only be changed while not playing.

Drop-frame shows a semicolon before the frames — `01:00:00;00`.

OFFSET

The timecode at the first sample of the song: type it as `HH:MM:SS:FF`. Each song has its own, which is how one show fills a whole day of timecode without two songs ever sharing a cue time. An hour each on the set list does the arithmetic for you (page 14).

SONG	OFFSET	A CUE AT 1:12
1 · Overture	01:00:00:00	01:01:12:00
2 · Ballad	02:00:00:00	02:01:12:00
3 · Haze	03:00:00:00	03:01:12:00

An hour each: no two songs can ever share a cue time, and the hour digit tells you which song the desk thinks it is in.

PAUSE

Returns to play start, explained on page 9. It is a per-machine setting, and it can be given a key of its own.

JUMP

How far ← and → move: 0.1 to 600 seconds. Two seconds suits finding a cue; thirty suits skipping verses in a tech.

THE SET LIST

A show is the whole evening: an ordered list of songs, each with its own stems, its own start offset and its own marks.

Eight songs, the fourth loaded. The number is the button: click it and that song comes in.

The number	Loads that song — its stems come in, its start offset goes on, and the readout jumps to where it sits in timecode. The loaded one is gold and framed.
The name	Type over it. A new song takes the name of the first file dropped into it.
The offset	Type a timecode and press `Enter`.
Add song	Puts an empty song after the loaded one; drop files on the window to fill it.
An hour each	Song 1 at `01:00:00:00`, song 2 at `02:00:00:00`, and so on. Needs 23 songs or fewer.
▲ ▼ ×	Reorder or remove. The song that is loaded stays loaded wherever it moves to.

RUNNING A SET FROM THE DESK

Previous song and Next song do the same job from a key, and both can be ticked to work while another program has focus — which is the whole point. Give them two keys on the lighting console and nobody needs to touch this machine all night. Neither has a key by default, because which keys are free depends on your desk; page 21 sets them.

CHANGING SONG STOPS PLAYBACK

The new song's stems have to be decoded, and carrying on through the gap would send the desk somewhere else mid-cue. It is a load, not a segue — plan the changeover into the running order.

At most 200 songs in a show, and at most 8 stems in a song.

SHOW FILES

A `.synshow` file is the show. It is deliberately not the machine, and that is what makes a backup rig possible.

THE LINE AT THE TOP

Save as… writes the set list, the stems, the offsets and the marks to a file of your choosing. Open… opens one. New starts an empty show. The name beside them is the show's own name; the grey path is where it lives.

Until you save one anywhere, the show lives beside the settings — so there is always a set list, even if you never save one.

Save as… keeps working even while the show is locked, so a copy can always be taken.

WHAT IS IN THE SHOW FILE

The set list · each song's stems · start offsets · marks · the show's name and frame rate.

WHAT IS NOT

The audio device · the MIDI port and trim · LTC settings · volume · keys · the look · the lock. Those stay on the machine.

TAKING IT TO THE BACKUP RIG

Copy the show file and its audio to the other machine, open it there, and choose that machine's device and port once. Nothing else needs touching — which matters, because the backup rig's audio interface is never called the same thing as the main one's, and the moment you need the backup rig is the moment you least want to be picking device names out of a drop-down.

WHY A COPIED FOLDER FINDS ITS OWN AUDIO

Every stem is stored twice: as a full path, and as a path relative to the show file where one can be worked out — its own folder, a sibling, or one level further out. The relative one is tried first. Copy the whole show folder to another drive, another machine, a USB stick, and the audio comes with it.

A relative path is only written when the audio is at most two levels above the show file. Beyond that the two are only incidentally on the same drive, and a long chain of `..` could resolve onto a real but wrong file — which is worse than failing. Files that have gone missing are named in the status line when the show opens.

COMING FROM A VERSION BEFORE 1.0

Old settings are read once and carried into a show automatically: the tracks that were loaded become the first song, and every file that had marks becomes a song of its own. Nothing is lost, and the set list starts populated.

TRACKS

Up to eight files play together as one song — the band, the BVs and the click of the same performance.

Three stems. The click track is muted here, and kept at 55% for when it is not.

Stems share a transport, a position and one set of timecode. There is nothing to line up and nothing that can drift: a frame is only handed to the audio output once every stem can supply it, so they are locked together by construction. Drop more files in at any time and each appears as another strip under the last.

Add…	Adds stems through the file picker — or drop them on the window, which is quicker.
Mute	Silences that stem without removing it. The click track lives here.
Level	That stem's own level. Saved with the show, so a quiet BV track stays quiet next time.
Channels	Mix by default: it goes wherever the Track row in Outputs says, and everything sums to one pair. Give a stem a pair of its own — 3+4 — to send it to the desk separately.
×	Removes that stem. The others keep their level and channels.

STEMS OF DIFFERENT LENGTHS

Perfectly fine. The short ones fall silent where they end and the transport runs to the end of the longest. In the picture the BVs stop before the outro, which is what BVs do.

WHICH ONE OWNS THE MARKS

The first. Marks belong to the song rather than to any one file, and the window takes its name from the first stem too, so put the full mix at the top.

WHY THEY CANNOT DRIFT APART

Each stem is decoded by the same thread and written into one buffer, and a frame is only handed to the audio output once every stem can supply it. They are not kept in step by anything that could fall out of step; they are one stream by the time they reach the sound card. It also means a stem that stalls stalls the rest, which is exactly what stems want.

Stems may share output channels with each other. LTC may not: it always keeps a channel to itself (page 19).

MARKS

Named points in the song. Add them in the tech, use them all week.

Eight marks in this song. LX42 GO has been given `F5`, so one key takes the playhead straight to the cue everyone keeps wanting to see again.

Add at playhead	Adds Mark 1, Mark 2 and so on. The key is `M`, and it works while playing — walk the song once and drop a mark on every cue.
The name	Type over it. verse 1, key change, confetti — whatever you will shout across the room.
Go	Takes the playhead there.
Set	Moves the mark to where the playhead is now. For when the band changed the arrangement.
The key	An optional key that goes straight to this mark. Click it, then press the key.
Delete	Removes the mark.

STEPPING THROUGH THEM

`PgUp` and `PgDn` go to the previous and next mark. While playing, previous mark skips a mark less than a second behind the playhead — so pressing it twice really does take you two marks back, rather than to the top of the one you are in.

Marks show on the waveform as ivory diamonds with their names along the bottom.

MARKS BELONG TO THE SONG

Not to the audio file. The same stem used in two songs keeps a separate set of marks in each, because it is almost certainly being used for two different things.

THEY SURVIVE ALMOST EVERYTHING

Stored in seconds of track time, so a change of device, sample rate, speed, frame rate or offset leaves them where they were. Renaming or moving the audio file does not.

A mark's key only works while its song is loaded, and if an action has the same key, the action wins.

OUTPUTS

One device carries the music and the timecode, on channels you choose.

Device, where the music goes, where LTC goes, and MTC. The grey line beside the device names the host API, the sample rate it opened at, and how many channels it has.

DEVICE

Lists the output devices of the host API shown beside it — WASAPI on Windows, in shared mode, at the device's own default sample rate. ↻ rescans and reopens, which is what to press after plugging an interface in. Until one is chosen it reads System default and follows whatever Windows is set to.

TRACK

Where the music goes: a stereo pair, or Mono summed to one channel. Stems set to Mix all arrive here; stems given a pair of their own bypass it (page 16).

LTC

Any other channel. Track on 1+2 with LTC on 8 works as readily as track on 1 with LTC on 2. Assignments to channels the device does not have, or to a channel already taken, are refused with the reason in the status line.

A TWO-CHANNEL INTERFACE

Track · Mono	ch 1
LTC	ch 2

The music is mono; the desk gets timecode on the right.

EIGHT CHANNELS, OR MORE

Track · Stereo	ch 1+2
A stem of its own · the click	ch 3+4
LTC	ch 8

Stems may share channels. LTC never shares one.

Until you choose, a stereo device gets track on 1 and LTC on 2, and a device with more channels gets track on 1+2 and LTC on 3. A one-channel device has no room for both, so LTC is switched off with a note in the status line.

LTC

Timecode as audio: the oldest, dumbest, most reliable way to tell a desk what time it is.

On, on channel 2, at –10 dBFS.

Tick on beside LTC, give it a channel, and run that channel into the desk's timecode input. It is a line-level audio signal, so it goes down the same kind of cable as everything else — which is exactly why it survives in a world of adapters.

LEVEL

The slider sets the sample peak, from –20 to 0 dBFS, and defaults to **–10 dBFS**. That suits nearly every receiver. If a desk will not lock, raising it a few dB is the first thing to try; if it is distorting through a long line, lower it.

THE VOLUME CONTROL DOES NOT TOUCH IT

LTC keeps its own level, independent of Vol. Pulling the music down for a quiet moment can never pull the timecode down with it — receivers depend on that level, and a sound operator riding a fader should not be able to drop a desk out of lock.

WHEN IT IS SILENT

LTC has no stationary state: a stopped reel of tape makes no sound, and neither does this. Whenever the transport is not playing the channel is silent, and a repeating frame is never emitted. If your desk needs to be told a position while parked, that is what MTC is for — see the table on page 29.

IT IS SAMPLE-LOCKED, BY CONSTRUCTION

LTC is generated one sample per output sample, by exact whole-number arithmetic, into the same buffer and in the same callback as the music. The track is resampled to match first. There is no clock reading another clock, no drift to correct, and nothing to adjust — at any speed, at any frame rate.

PROVING IT WITHOUT A DESK

From a terminal:

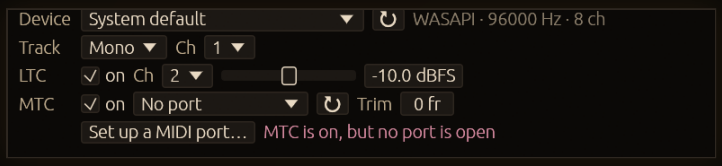
```
synphonos --render-ltc out.wav 01:00:00:00 25 30
```

writes thirty seconds of LTC starting at `01:00:00:00` at 25 fps, as a 48 kHz WAV. Open it in anything that reads timecode — or in an editor, where the biphase pattern is visible by eye — and you have proved the generator without an interface, a desk or even a window.

Both LTC and MTC can be switched on or off during playback, and they are independent: run either, or both.

MTC

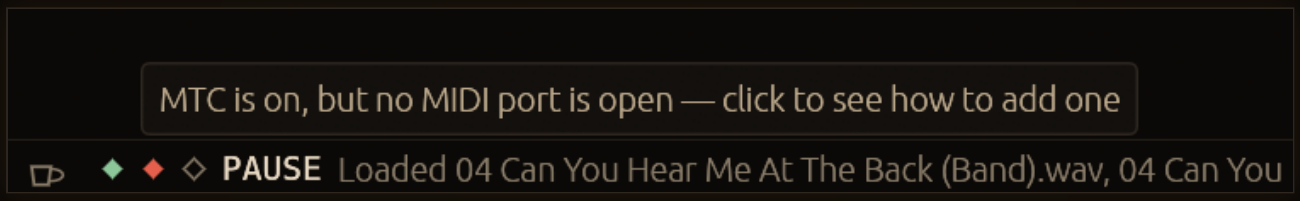
MIDI timecode, for another program on the same machine — or a real MIDI interface, if the desk is across the room.



MTC on with no port chosen: the row says No port and the MIDI diamond in the status bar turns red.

Tick on, choose a port, and set the trim if you need to. ↻ rescans for ports that appeared after the program started.

Trim shifts MTC by −8 to +8 frames against the audio, for pulling a receiver into line when something downstream adds a delay of its own. Leave it at 0 until you have a reason.



Hovering the diamond says what is wrong; clicking it opens the help window.

WINDOWS HAS NO VIRTUAL MIDI PORTS

To send MTC to another program on the same machine you need a virtual MIDI driver. loopMIDI, free, from Tobias Erichsen, is the one everybody uses. Hardware MIDI interfaces need none of this — their ports appear in the list directly.

Set up a MIDI port… under the MTC row walks through it and opens the download page. The red diamond in the status bar opens the same window.

loopMIDI is not packaged inside Synphonos: its licence does not allow that, so it is installed separately. On macOS and Linux, Create virtual port publishes a port named `tcgen MTC` without any of this.

KEYS

Every action can have two keys. Click a key, press the new one, done.

Nineteen actions, two slots each. Previous song and Next song have been given `F9` and `F10` here and ticked for use from the desk.

SETTING ONE

Click the key button — it says Press a key — then press what you want. Any key alone (an arrow, a letter, Escape) or with any of Ctrl, Alt and Shift. Clicking anywhere else cancels. × removes a key, and Reset keys puts every default back.

A key does one thing. Giving it to another action, or to a mark, takes it from the old one and says so in the status line.

THE ONES WITH NO DEFAULT

Play and Pause separately, Speed down and Speed up, Return on pause on/off, Previous song, Next song and Lock the show on/off start empty. Every sensible key is already doing something, and which ones matter depends on your show.

Keys only act when no text field has focus and no drop-down is open — so typing chorus into a mark name cannot stop the show at the letter `S`. The full default list is on page 32.

FROM OTHER APPS

Tick a key, switch the box on, and it works while the lighting console has focus. This is how a set gets run without anyone touching this machine.

The master switch. Ticked keys show in gold while it is on.

HOW TO SET IT UP

1. Give the action a key (page 21).
2. Tick the box beside that key.
3. Tick Ticked keys work from other apps.

The keys indicator in the status bar fills in, and the keys turn gold.

WHAT IT COSTS

While it is on, those keys are taken from every other program, exactly as any Windows hotkey is. A ticked `←` does nothing in other programs until it is unticked. Choose keys nothing else wants — `F9`, `F10`, a keypad digit.

IT HANDS THEM BACK

Whenever the Synphonos window has focus the keys are released, so its own text fields type normally. They are registered again the moment focus goes elsewhere.

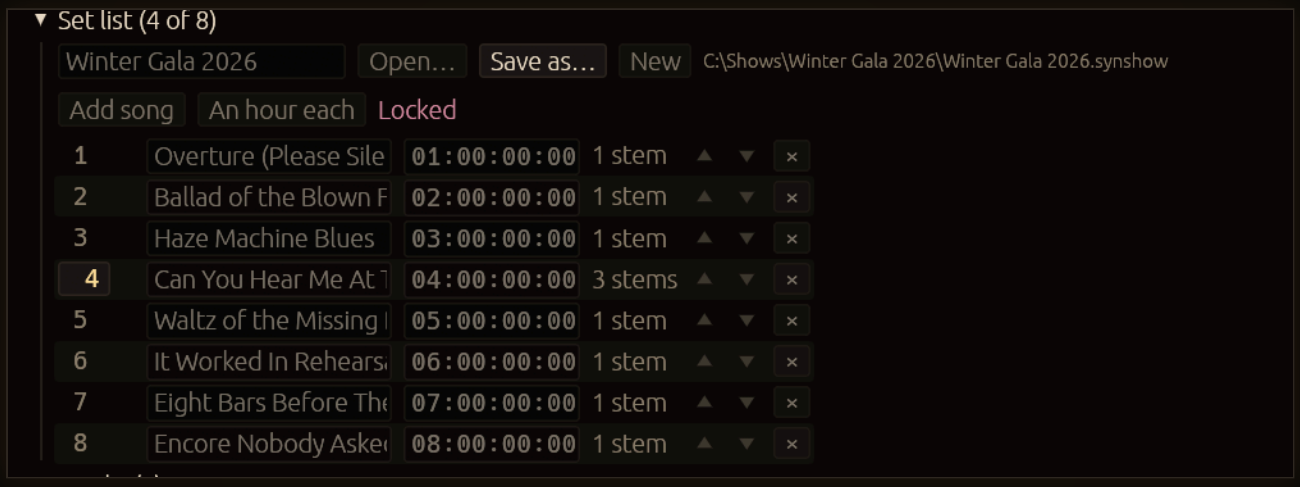
WORTH KNOWING

- It works over programs running as administrator, which is the reason it uses Windows hotkeys rather than a keyboard hook.
- If another program already holds a key, the status line says so and the keys indicator turns red. Pick a different key.
- Keys that can be ticked: letters, digits (main row and keypad), F1–F24, Space, Enter, Tab, Escape, Backspace, the arrow and navigation keys, and punctuation.
- Punctuation is looked up on the current keyboard layout: a character typed with Shift needs Shift in the key, and one typed with AltGr cannot be ticked.
- Windows only. On macOS and Linux the section says so.

It is off until you switch it on, and it is your job not to take a key the console needs.

THE LOCK

The padlock at the right-hand end of the status bar freezes the show without freezing the operator. Doors open, lock it.



Locked. Open…, New, Add song, An hour each, the names, the offsets and the reorder arrows are all greyed; the running order is exactly what it was. Save as… still works.

STILL LIVE

- Play, Pause, jumps and frame steps
- Marks — going to them, and their keys
- Previous song and Next song
- Volume and the stem levels
- Save as…, so a copy can be taken

FROZEN

- Stems: loading, adding, removing
- Marks: adding, renaming, moving, deleting
- Offsets, the frame rate, the channels
- The running order, and the show's name
- The keys themselves

STOP ASKS TWICE

The first press arms it and says so in rose; a second within two seconds stops. Let it lapse and nothing happens.

Esc is the key a hand reaches for by reflex, and a stopped cue in front of an audience is the expensive mistake. So while the show is locked, Stop is the one thing that wants asking twice.

The lock can be given a key of its own, and it survives a restart: a show machine left locked comes back locked. Anything that tries to change the show anyway — from a key, the window or a dropped file — is refused through one guard in the engine and told why in the status line.

APPEARANCE

The paintbrush at the far right of the status bar. The window changes as you choose, so you can see what you are doing.

Everything about the look, in one panel.

COLOURS

Three picks. Accent is whatever is lit and live — the readout while playing, the playhead, focus, ticked keys. Ground is the window's background. Text carries values and labels.

Everything else is derived from those three and keeps the relationships the design has: the accent from hot to dim, the ground from the deepest wells up to raised surfaces, the text from ivory down to bronze.

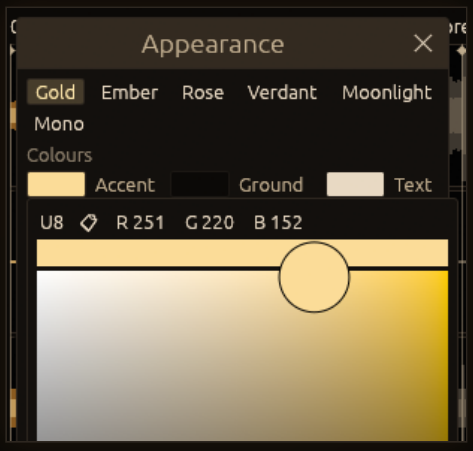
Good, Warning and Error stay separate, so status never depends on a colour you picked.

Colours are clamped so they stay readable: text and the accent cannot go dark, the ground cannot go light.

THE REST OF IT

Glow	0–200%. How strongly lit things halo. At 0 the colours stay and every halo goes.
Scale	80–150%, for a machine across the room.
Waveform	The strip's height, 40–400 points.
Density	Compact, normal or roomy.

Preview plays the start-up animation there and then; a click ends it as usual. Reset to defaults puts the gold look back. The look belongs to the machine, so a borrowed show cannot change how your rig looks.



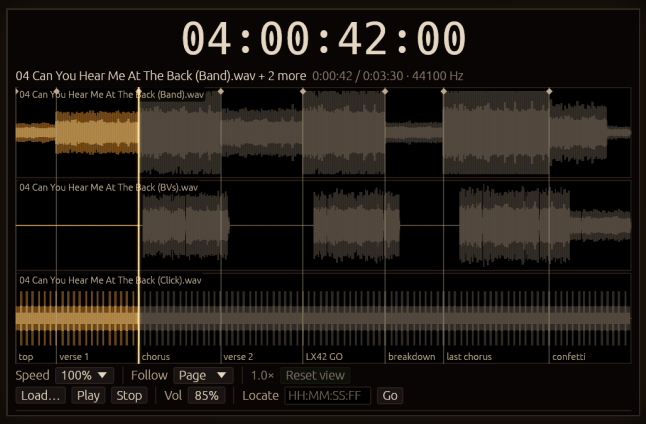
Clicking a swatch opens the picker; the window recolours under it as you drag.

START-UP ANIMATION

Play when the window opens	Off, and the window opens straight into the controls — no animation, no click to begin.
Play the 5 s intro	Off, and it opens at the waiting loop instead of building up to it.
Show the name	The word SYNPHONOS above the animation.
Follow the accent colour	The animation takes your accent instead of staying gold.

SIX LOOKS

Presets to start from. Each one sets the three colours; everything else follows.



Gold — the default, and the one the icon is drawn in.

Ember — hotter and redder. Good in a warm room.

Rose — softer, and unmistakably not a warning colour.

Verdant — the quietest of them under a work light.

Moonlight — cool, for a rig where gold means something else.

Mono — no colour at all. The readout still lights.

The look belongs to the machine, not the show, so it stays put when a show is carried to another rig. The start-up animation can be made to follow the accent colour too, or left gold.

THE STATUS BAR

The one line to glance at. Hover anything on it for the detail.

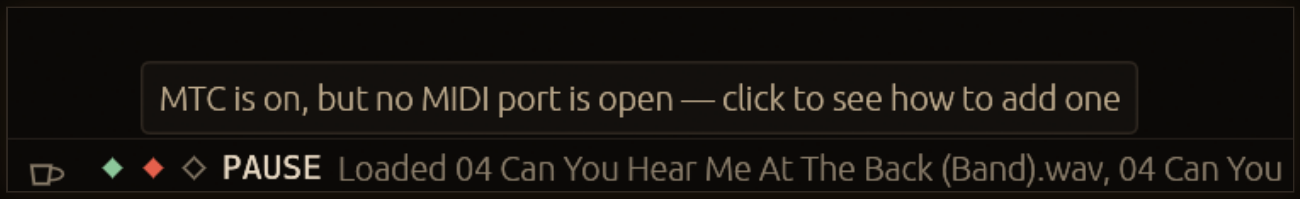
Cup, three indicators, the transport state, the latest message — and at the far right the book, the padlock, About and the paintbrush.

LEFT TO RIGHT

The cup	If you find this useful, it opens the page where you can buy the author a coffee. It steams when you hover it, and does nothing else.
Audio device	A filled green diamond when the device is open. Red if it has gone.
MIDI port	An outline when MTC is off, filled green when a port is open, and red when MTC is on with no working port — click it for help. It pulses while timecode is going out.
Keys from other apps	An outline when the master switch is off or nothing is ticked, green when the ticked keys are registered, red if one could not be, and rose while it is waiting for you to press a new key.
The transport state	PLAY pulses slowly while playing. A speed other than 100% shows beside it in rose.
The message	What just happened, or what was refused and why. Warnings in rose, errors in red.
The book	The licence, terms and privacy notice.
The padlock	The show lock (page 23). It turns rose while locked.
ⓘ	About (page 27).
The paintbrush	Appearance (page 24).

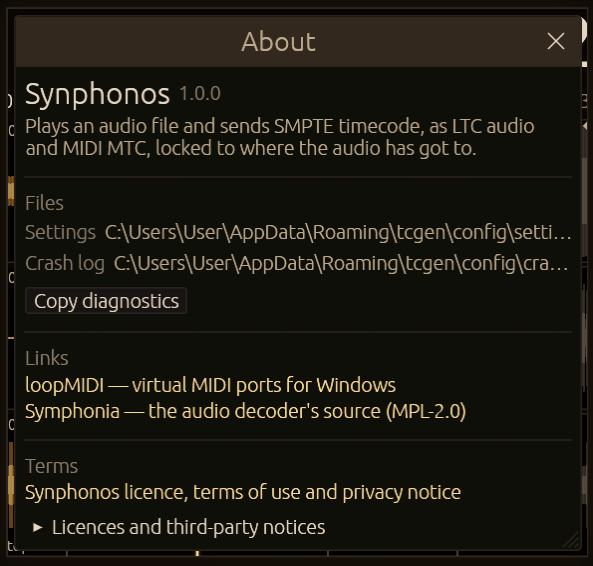
The cup, hovered. It steams, slowly, and only while the pointer is on it.

Rose and red are both reddish, so nothing here is shown by colour alone: the indicators change shape as well — filled against outlined — and every state has words behind it, on the line or in a tooltip.



Hovering an indicator says what it is doing, and names the device, the port or the keys behind it. The MIDI one also opens the help window when there is no port.

ABOUT



The ⓘ beside the paintbrush.

IF IT EVER FALLS OVER

What happened is appended to `crash.log`, beside the settings, and a dialog says where. Send that file with a report.

The log keeps the most recent entries and trims itself, so it cannot quietly fill a show machine's disk.

THE LICENCES

Synphonos is built from open-source components. Every one of them, the licence it is used under, and that licence in full, are under Licences and third-party notices — and in `THIRD-PARTY-NOTICES.txt` if you would rather read it in Notepad.

The version, where its files are kept, the links out, and the licences of everything it is built from.

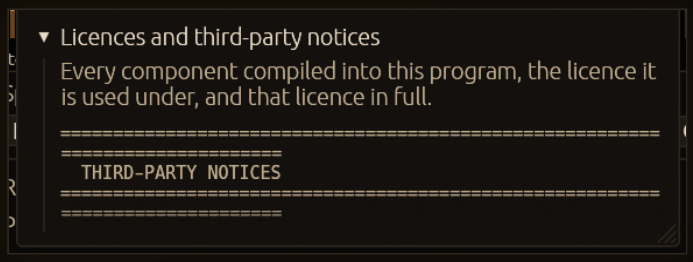
COPY DIAGNOSTICS

Puts the version, the audio device, the MIDI port and the last message on the clipboard. Paste that into a bug report and most of the first round of questions is already answered.

LINKS

- loopMIDI — the virtual MIDI port driver for Windows (page 20).
- Symphonia — the source of the audio decoder, at the exact version used.

Those are the only two addresses in the program, and they open in your browser. Synphonos itself makes no network connections at all.



Licences and third-party notices, opened: every component compiled into the program, the licence it is used under, and that licence in full. Generated from the actual build, not written by hand.

WHAT IS SAVED

Two files, deliberately kept apart: one is the show, one is the machine.

FILE	HOLDS
`<show>.synshow`	The set list: songs, their stems, start offsets and marks; the show's name and frame rate. It is the show, so it travels — to a backup rig, or to next year's revival.
`settings.toml`	Audio device, MIDI port and trim, LTC on/off and level, volume, keys, the look, the lock, and the last show opened. It is the machine, and the backup rig is a different machine.

Both sit in `%APPDATA%\tcgen\config` until a show is saved elsewhere. `crash.log` is there too. The About window prints the exact paths, and the folder is still called `tcgen` — the program's old name — on purpose, so settings written before it was renamed keep working.

WHEN IT WRITES

A second after any change, every five seconds during playback, and on exit. You should never have to save anything by hand; Save as… exists to put a show somewhere of your choosing, not to avoid losing it.

COMING BACK

On launch everything is restored and the transport lands paused where it left off. If the audio file, the device or the port has gone, the rest is restored anyway, the gap is named in the status line, and output falls back to the default device.

THE MACHINE

```
%APPDATA%\tcgen\config\
├── settings.toml        device · port · keys · look · lock
├── crash.log            if it ever falls over
└── default.synshow      the show, until you save one elsewhere
```

THE SHOW

```
D:\Shows\Winter Gala\
├── Winter Gala.synshow
└── Audio\
    └── …the stems
```

Copy this folder whole and it works anywhere.

HAND-EDITING

Both files are plain text. Values out of range are clamped on load; a file that does not parse is ignored with a note in the status line, rather than taking the program down with it. Deleting `settings.toml` resets the machine — including asking you to accept the terms again — and leaves every saved show untouched.

WHAT COMES OUT

Exactly what each output does in each state. Worth a look before blaming the desk.

STATE	LTC	MTC
Playing	Runs.	Quarter-frames.
Paused	Silent.	One full-frame at the paused position, then nothing.
Paused, returning to play start	Silent.	One full-frame at the play start.
Stopped, or located	Silent.	One full-frame at the start offset or the located position.
A jump while playing	Silent for a few audio buffers, then runs from the new position.	A full-frame at the new position, then quarter-frames from the start of a sequence once audio runs.
A speed change while playing	Silent for a few buffers, then runs at the new speed from the same place.	A full-frame at the place, then quarter-frames at the new pace.
End of file	Silent from the last sample. The transport stops there.	Quarter-frames stop with the audio; one final full-frame names the last frame played.
Device disconnected	Stops with the device.	Stops within one audio buffer; one full-frame at the last position. The status line says what happened.

LTC IS SILENT WHENEVER IT IS NOT PLAYING

It has no stationary state, and a repeating frame is never emitted. A desk that needs to be told a position while parked needs MTC.

END OF FILE LEAVES THE PLAYHEAD AT THE END

Press Play from there and it restarts at the song's start offset.

Device loss keeps the playhead where playback stopped, rather than returning to the start offset, so reselecting the device carries on from there. A watchdog treats 1.5 seconds without a callback as a lost device.

WHY A JUMP GOES QUIET FOR A MOMENT

Landing somewhere new means the decoder has to get there and fill a buffer again. Rather than play whatever happened to be in memory, the output is held silent until a quarter of a second of the new position is ready — a few audio buffers, normally under a tenth of a second. The position never runs ahead of the audio, so the desk is never told about a frame that has not been heard.

IF IT GOES WRONG

WHAT YOU SEE	WHAT TO DO
The desk will not lock to LTC	Check the LTC channel is going where you think (page 18), and that it is on. Raise the level a few dB from −10. Check the frame rate matches the desk. Remember LTC is silent unless it is playing.
The MIDI diamond is red	MTC is on with no working port. Click the diamond: the help window walks through installing loopMIDI and making a port.
The port vanished	loopMIDI was closed, or the interface unplugged. Restore it and press ↻ beside the port.
The keys indicator is red	Another program already holds one of the ticked keys. The status line names it; pick a different key.
A ticked key does nothing in another program	That is what ticking does — it takes the key. Untick it to give it back.
Keys do nothing at all	A text field has focus, or a drop-down is open. Click the window background. Or the start-up animation has not been clicked through yet.
The device diamond is red	The interface has gone. Plug it back in and press ↻ beside the device; the playhead is where it was.
Audio stutters, timecode with it	The machine is not keeping up. Close what else is running; this is one of the few programs that genuinely wants the machine to itself.
A song opens with files missing	The status line names them. The audio has been moved or renamed since the show was saved — keep the audio in the show's own folder and copy the folder whole (page 15).
An edit is refused	The show is locked. The status line says so; the padlock is at the right-hand end of the status bar.
Cues drift through the night	The frame rate does not match the desk's. It is nearly always this.
MP3 or AAC lands a few ms off	Seeking in those formats is the codec's business, not ours. Use WAV or FLAC where it matters.
It fell over	A dialog says where `crash.log` is. Send it, with Copy diagnostics from the About window.

BEFORE THE DOORS OPEN

Play a song. Watch the desk take the timecode. Lock the show. That three-step check catches the rate, the routing and the port, which between them are almost every problem anyone has.

WHAT IT WON'T DO

Better to read this now than to find out at the production meeting.

NOT IN THIS BUILD

- Chase or slave mode — it generates timecode, it never follows it.
- Free-run without a file, and looping.
- Audible scrubbing, and pitch-preserving speed change.
- 23.976 fps. ASIO. Split output devices.
- Keys from other apps on macOS and Linux.

LIMITS

- At most 8 stems in a song; they cannot be moved in time against each other or reordered.
- At most 200 songs in a show. An hour each needs 23 or fewer.
- Changing song stops playback: it is a load, not a segue.
- A song's start offset is applied when that song is loaded — typing another song's offset does not move the playhead until you go there.
- Marks belong to the song, not the audio file: the same stem used in two songs keeps a separate set in each.
- WASAPI shared mode only, at the device's default sample rate.
- `Pause`, `Scroll Lock`, `PrtSc`, media keys and keypad `*` cannot be assigned. Keypad `.` works in the window but not from other apps.

THINGS WORTH KNOWING ABOUT FILES

- AAC priming is played. The decoder does not expose the encoder's priming for `.m4a` and ADTS AAC, so the music starts about 23 ms after `00:00:00:00` with typical encoders. Use WAV, FLAC or MP3 where that matters.
- MP3 encoder delay and padding are removed from the LAME header, so `00:00:00:00` is the first real sample.
- Lengths are measured, not trusted. MP3 without a Xing header and ADTS AAC are measured by reading every packet on load — about 25 ms per hour of audio — because their declared lengths can be out by minutes.
- Multichannel files play their first two channels; mono is duplicated.
- Seeking in MP3 and AAC may land a few milliseconds off. WAV and FLAC are exact.

While the window is minimised the interface updates ten times a second, so the end-of-file and pause full-frames, and keys from other apps, can act up to 100 ms late. Quarter-frames and LTC are unaffected. Covering the window with another one costs nothing.

EVERY KEY

The defaults. All of them can be changed, and any of them can be made to work from other programs.

ACTION	KEY
Play / Pause	`Space`
Stop	`Escape`
Jump back	`Left`
Jump forward	`Right`
Previous frame	`Comma`
Next frame	`Period`
Back to play start	`Backspace`
Go to start	`Home`
Add mark	`M`
Previous mark	`PageUp`
Next mark	`PageDown`

NO KEY UNTIL YOU GIVE IT ONE	
Play	–
Pause	–
Speed down	–
Speed up	–
Return on pause on/off	–
Previous song	–
Next song	–
Lock the show on/off	–

The song and lock keys are the ones to set if the show is run from a lighting console — and to tick, so they work from it.

Keys are named the way the Keys section names them: `Comma` and `Period` are , and ., and `Left` and `Right` are the arrows.

THE MOUSE, OVER THE WAVEFORM

Click, or drag and release	Move the playhead there.
Hover	Show the timecode a click would land on.
`Ctrl` + scroll	Zoom, around the pointer.
`Shift` + scroll	Move along the track, and stop following.
`Alt` + scroll	Make the strip taller or shorter.

COMMAND LINE

`synphonos`	Open the window.
`synphonos --selftest`	Run the timecode core checks; exit 0 or 1.
`synphonos --render-ltc out.wav 01:00:00:00 25 30`	Write 30 seconds of LTC starting at `01:00:00:00` at 25 fps, as a 48 kHz WAV.

Both of those run headlessly: no window, no audio device, no MIDI port. Handy for proving a machine before it goes out.

THE SMALL PRINT

THE LICENCE

Synphonos is free. Install it on as many machines as you like, use it commercially, and pass the unchanged program on to whoever you want — with `LICENCE.txt` included, which its own section 2 asks for. You may not sell it, charge for it, or put it in a paid bundle.

It comes with no warranty of any kind. Whether to trust it with a show is your decision — testing it first, and having a fallback if it stops, are yours to do. The full agreement is in `LICENCE.txt`, and in the book at the right-hand end of the status bar.

PRIVACY

It makes no network connections at all and collects nothing about you or about your show. The only thing it writes is its own settings and your show files, on your own machine. The two links in the About window open your browser; that is the whole of it.

DONATIONS

The cup in the corner of the status bar is a way to say thank you if this saved your night. Donations are gifts, not purchases: they buy no warranty, no support and no priority, and nothing about the program is held back without one.

BUILT FROM

Rust, egui and wgpu for the window; Symphonia for decoding, under the Mozilla Public License 2.0, compiled in unmodified, with the offer of its source in the About window; cpal for audio; midir for MIDI; rubato for resampling. Every component, its licence and that licence in full are listed under Licences and third-party notices.

This document is set in Ubuntu Light and Hack — the program's own two typefaces — and the pictures in it were taken from a running copy of version 1.0.0.

SYNPHONOS · VERSION 1.0.0

Break a leg.