

Ad-hoc and Explicit Recording

Fripp is always recording your MIDI input. Sometimes, similar to a multitrack sequencer, Fripp records your input into one or more staves of the score you're working on. We call this recording mode Explicit Recording, and you enter into that mode in the way you would expect, by toggling on the Record button and then clicking the Play button.

All the rest of the time, Fripp performs Ad-hoc Recording. This feature allows you to improvise freely, and never have to interrupt yourself to commit some passing idea to memory. That idea will be preserved, and available for you to review and work with when you're ready to do so. Tomorrow, next week, or next month, you can browse your old recorded history, play the sequences from it, and import the sequences into scores.

So, while Fripp is running it is either in some flavor of Ad-hoc Recording state, or some flavor of Explicit Recording state. The diagram below shows each of the possible states, and the transitions between them.

Working from left to right:

- **Ad-hoc Recording:** You're playing, but Fripp is not playing a sequence back (you're not playing along with Fripp).
- **Coordinating Recording:** A merely transitional state that handles the signals that end one kind of recording and begin another.
- **Ad-hoc Overdubbing:** Fripp is playing a sequence back and you're playing along with it.
- **Ad-hoc Looped Overdubbing:** Fripp is playing a sequence back in loop mode, and you're playing along with it.
- **Explicit Recording:** You explicitly start and stop recording, and can play along with previously recorded material
- **Explicit Multitake Recording:** You explicitly start and stop recording, playing along with a repeated loop of previously recorded material. Fripp treats each loop as a mini-session, and preserves your inputs during each session. When you stop playback, you can choose which session's input (which "take") to keep.
- **Explicit Overdubbed Recording:** You explicitly start and stop recording, playing along with a repeated loop of previously recorded material. During each successive loop iteration, Fripp merges your input with that from previous iterations.

Your ad-hoc recordings aren't necessarily one gigantic piece. More likely, they're discreet chunks with a lot of silence in between. So Fripp saves each chunk as a separate item, with a timestamp. Fripp also time-stamps and saves your explicit recordings.

There's no trick to determining the start and end of an explicit recording, but Fripp does need to guess when you've come to the end of an ad-hoc recording. In some cases, Fripp employs a timeout condition (some significant length of time passes without more MIDI input from you); in other cases, no timeout is necessary.

If you look at the diagram above, you'll see that pressing the Stop button obviously signals the end of any recording (ad-hoc or explicit) that begins when you press Play. There are more ways to transition out of the Ad-hoc Recording state (on the far left of the diagram). Having examined these recording states at a high level, lets look at each one in more detail.

[Next: Fripp States During Ad-hoc Recording](#)