

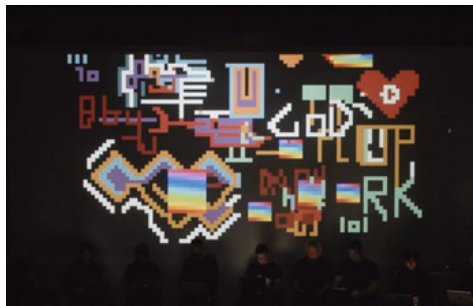
[View all by mattmora](#)[Follow mattmora](#)[Add To Collection](#)

Set Up, Play, Tear Down

A downloadable game for Windows and macOS

[Download](#)

Set Up, Play, Tear Down, by Matt Wang



Performed by the Princeton Laptop Orchestra on March 28, 2024.

This is intended to serve more as an instrument/performance system than a game; the controls and systems are consequently somewhat esoteric.

Controls

- Move with Arrow Keys.
 - +Shift to push pixels.
 - +Alt/Option to change size (+Shift to flip change axes).
- Press/hold Space to paint.
- 123456789 for solid colors.
- Ø for eraser.
- Q for rainbow.
- R to reset position.
- M to mute (local sound only).
- O to toggle player outlines (local display only).
- (Host) I to show/hide IP address in top left.
- (Host) C to clear the canvas.

Set Up, Play, Tear Down

These are rough instructions for performance. Incorporate any rules/procedures/gestures/sub-sections you want that support player intention and coordination or produce interesting sounds.

One player must host and the rest must join by entering the host's IP address in the according field while on the same network. The intended number of players is around 8-10, but any number of players, up to where technical performance becomes an issue, can go through the steps.

I recommend the following three-section structure for a performance. There is no built-in cue system, so it's up to a given group to decide how to cue each section.

I. Set Up

Collaboratively paint something (an image, text, or pattern - anything the group agrees on).

II. Play

Perform non-destructively (do not paint, erase, or push pixels) around what you've painted.

III. Tear Down

Perform destructively, freely painting, erasing, and pushing, toward a blank screen.

When no non-player pixels remain colored, players can move offscreen to end.

Sound System

Each player has 2 streams of sound, read from the canvas as raw audio data, where HSV color converts to a sample value as $\sin(H * 2 * \pi) * S$, with H and S between 0 and 1.

Both streams start at the bottom-left pixel of the player's block then read into the contiguous blob of colored pixels via queue of 4-connectivity (a not-quite spiral pattern).

One stream looks at all visible pixels while the other only looks at pixels that are part of player blocks. The player block stream runs a 1/4 the sample rate of the other and so sounds 2 octaves lower.

Mac permissions workarounds

In terminal:

- **chmod -R 777 [path/to/app]**
- **sudo spctl --master-disable** to disable Gatekeeper
- **(sudo spctl --master-enable** if you want to re-enable Gatekeeper when done)

[More information](#) ^

Status [In development](#)

Platforms [Windows](#), [macOS](#)

Author [mattmora](#)

Download

[Download](#)**SPT_Windows.zip** 27 MB [Download](#)**SPT_OSX.zip** 36 MB 

Leave a comment

[Log in with itch.io](#) to leave a comment.

[itch.io](#) · [View all by mattmora](#) · [Report](#) · [Embed](#)
[Games](#) › [Free](#)

