Pink Cloud Assembly Language

By Sam Von Ehren

Pink Cloud Assembly Language is a single player game I developed as an experiment in form. I wanted to recreate a music album in video game form, a “Video Game Cover Album”. I was partially inspired to do this by *Clair De Lune*, one of my favorite pieces of classical music. *Clair de Lune* attempts to create the same feelings and experience as the eponymous poem in music form. I wanted to apply this idea to games and decided to recreate a Japanese rock album I listened to frequently as a teenager, *Ja, Zoo*.

The game features 10 mini-games loosely based on each of the tracks from the original album. I tried to reconcile the feel and tempo of the songs with the lyrical content (at least to the point that I could translate) when designing the games. The most important thing I hoped to capture was the experience of track finding on a CD. Rapidly switching through unique experiences is not something games currently offer and I wanted to see if it could work.

I think the game delivers on its promised experience and I want to make an original video game album in the future. I also think the basic CD track switching mechanic has a lot of applications. Next time I make a game with a traditional level structure, I hope to reuse the CD style set up. Rather than each level being a more difficult challenge unlocked for completing the last, levels could vary in content and tone and players could switch between them at will. This plays into my favorite aspect of *Pink Cloud Assembly Language.* You don't *beat it* in the traditional sense. The tracks loop around back to the beginning, each experience is come and go. I think that's a very powerful model for games.

One interesting property of the game that came about organically is the amount of variance between the “tracks”. *Ja, Zoo* is a relatively homogenous punk rock album with most songs focused on the same subjects: escape, consumption, and dissatisfaction. Without making it an intentional choice, most of my game “tracks” became relatively homogenous in their conflicts and objectives. Whether that says more about my design skills or was caused by recreating *Ja, Zoo* I can't say for sure.

Creating *Pink Cloud Assembly Language* taught me a lot of things about myself as a designer, programmer, and artist. As a designer, I discovered many of my weaknesses, particularly level design. Reflecting on games I've made, I usually focus on making interesting systems that deliver an experience rather than an experience created from an interesting system. Forcing myself to actually do level design was a great exercise.

As a programmer, I created a much bigger system than I've ever had to deal with. *Pink Cloud Assembly Language* clocks in at over 2300 lines of code. While writing it, I had to take some advice I heard from a Jonathan Blow lecture: worry about efficiency later. Writing really good efficient code is usually a waste of time since most code isn't a problem anyway. This experience emphasized that philosophy and will make future games easier to write.

Finally as an artist, I learned that I could work as an artist. This is the first time I've really drawn anything for my own games. The art is far from spectacular, but I no longer feel confined by lack of art.

I think my biggest failure in the project is the transitions between tracks, because they don't exist. *Ja, Zoo*'s tracks flow from one to the other in a clearly intentional way, often with unique transitional passages, but partially due to the differences in gameplay and partially because of my more iterative development process, real transitions became very difficult to implement by the time I realized they were missing. If I were to do this project again, I think I would include transitions as part of my top level goals, so that I would factor them in when trying to conceive every track.

Pink Cloud Assembly Language can be downloaded at
https://github.com/gamesketches/PinkCloudAssemblyLanguage/archive/master.zip