**03 Archiving the treasures by Ghandy and Axel**

Ghandy, Axel, Wed 05 Mar 2008

# Archiving the treasures
**by Ghandy of Moods Plateau and Axel of Brainstorm**

Download PDF version:

Emulators can pretty much run anything from the old times. And as we all know, every rule has an exception. Heikki Orsila from Finland loves exceptions. Under his nickname SHD of the former group Mystic, he fixed old productions so they would run in current emulators, so people could continue to enjoy the classics of the Commodore Amiga era.

Orsila lives in the Finnish town Tampere, which is an industrial town as well as being known for its theater affinity. The region is host to approximately 5000 companies which focus on the production of paper, textile and clothing products. Orsila doesn't work in any of those areas. He's working in parallel computing projects at the Tampere University of Technology.

"For some odd reason, I was fascinated by computers at a very young age," he starts to explain, before adding with a smirk on his face: "...probably due to games." At age of 12 he got very interested in programming and in learning 68k Assembler. "That was my main hobby until nineteen or so," he jokingly admits.

The fact that he isn't married gives him some more spare time. Working on program code to fix old productions for the emulators of today, takes quite some time after all. But when he started out, others were also working on similar projects. "To begin with, I think WHDLoad people did a much better job," he explains. "They had a solid infrastructure for loaders, which increases quality; avoiding bugs and helping to maintain old releases. All my fixes were custom jobs, not using any specific infrastructure, so it meant doing extra work. Stupid, but hey, I hadn't studied computer science when I was eighteen or so. Unfortunately, I don't have a list of successful fixes anywhere. I did maybe twenty projects, so that's not much compared to the WHDLoad guys.

# "I was fascinated by computers at a very young age; probably due to games."

Orsila just fixed demos he liked. He hated rebooting, floppies, and external compatibility programs (turning off caches, selecting screen modes, etc). "I don't recall doing anything with 'Voyage', and 'Hardwired' was never finished. I did the 'State of the Fix' production, though. It was not much work. I just found all instances of track loaders and put jumps to my load routines, turned off caches, etc."

# The big one

The hardest fix was Mental Hangover by Scoopex. "It had many odd bugs that took a long time to fix, maybe fifty hours of work or so. I encountered issues such as 24-bit addressing where bits 25 to 31 were crap, and caused memory errors on 32-bit architectures (68020+). A copper list was not terminated, causing copper to do crazy things. Self-modifying code was used. I added blitter waits so that blitter registers weren't rewritten while the last operation was being executed. It was fun; it gave me some satisfaction."


shd (irc-galleria.net)

Most of his projects don't take that long to complete; between four and eight hours. Most of them went pretty smoothly, with the exception of Mental Hangover.

What did take time was learning everything relevant. Coding demos and fixing or HD-loading games was where he learned something. Demos didn't have code to obfuscate the reader, as games did. Also, demos were structurally much simpler than games.

His biggest project by far was making an operating-system-friendly version of the game Warhead. "I had to go through all forty thousand lines of code in the game, inspect data structures for pointers, and relocate memory addresses to make it a standard executable file. This project took probably a hundred hours or so."


# "There wasn't any testing or customer support. That implies crap."

"In the age of non-internet communication, it was rare to get feedback from users. "I rarely got any feedback at all, but I recall two cases: First, the person who wrote WHDLoader for Mental Hangover did look into my loader to fix some of the harder bugs, and thanked me in his info file. Second, a local TV series, depicting actually committed crimes, needed "hackerish" material (laughs). They displayed a short segment from Traffic, the demo I coded for Mystic, in one of their shows. Somehow they had gotten into contact with Brainlock who forwarded them to me."

Orsila did it all for the fun. "Let's put elitism aside. I didn't know that much about the whole system, and obviously, not many did. We were young and did it for fun. There wasn't any testing or customer support for that process. That implies crap."

"I did something like five-plus-one-third demos. Two for Fanatic, three for Mystic, and one-third of one for RNO. Plus, a few intros. Of these demos, I liked Traffic the most."

With the decline of the Amiga and the increasing power of the PC, Orsila lost his motivation too. "My motivation was always of technical origin, how computers operated internally. Occasionally I wanted to compete against other groups with nice demo effects, but it became pretty clear that although I could write the same level of code as other youngsters could, I didn't have an eye or thought for design. When the Internet, GNU/Linux and Open Source came, it shook me completely. I was sold when I realized there is an operating system for which I have the source code for every part, and those parts could be modified and redistributed as modified versions. So I dropped Amiga projects."

# Embracing the PC

"In 2000 I missed good old songs from the Amiga, and I could only find crappy MOD players for GNU/Linux, so I needed something. Fortunately, UAE, the Amiga emulator, was around, and it occured to me that it would not be a huge project to customize UAE just to play some Amiga formats. I wrote a small 68k assembler minikernel that was emulated with UAE. The problem was then to acquire playroutines for music formats that were interesting.  Fortunately, I recalled such a thing had already been done with Delitracker.  They had the infrastructure for that. So I reimplemented necessary parts of their infrastructure in 300 lines of asm or so, and got maybe twenty formats to play somehow. This was done in a month or so. After two years of hacking, I got 3000 lines of asm with 150 formats supported. Note, I did not copy their code; I just used deliplayer plugins written by various people. Each deliplayer plugin handles one or more music formats."

"The initial release of UADE played only a handful of formats, and was released after an intense weekend of programming. However, the project became huge when Orsila needed to make UADE really usable for many people.  "I must have rewritten each part separately at least two times. Fortunately, others could help as it was an Open Source project. Michael Doering was very helpful, he wrote GUI code, for example. He did pretty much everything at least once."

As an end result, UADE is a pretty flexible program that plays almost all Amiga music formats. This took about 4 years; after that it was just usability, maintenance and cleanups. The last finishing touch came in 2006 when Antti S. Lankila modeled the Amiga's sound filter system, and put that into UADE and some UAEs. UADE supports both A500 and A1200 sound models. The result is clearly

recognizable even when the filter is turned off.

# "I wish we could have done more cooperation projects."

In retrospect, Orsila only regrets one thing: "I wish we could have done more cooperation projects. The results would have been better."