

Interview with Scott Cronce re: Disclone

John Aycock
Department of Computer Science
University of Calgary
2500 University Drive N.W.
Calgary, AB, Canada T2N 1N4
aycock@ucalgary.ca

TR 2020-1117-03, July 2020

Preamble

This is an interview with Scott Cronce [krans], conducted via email between July 14–July 20, 2020, with follow-up questions via Zoom on July 21, 2020. He worked for Disclone, a service company that provided software duplication and copy protection for, among many others, the game company Electronic Arts.

This work received ethics approval from the University of Calgary’s Conjoint Faculties Research Ethics Board, file REB16-0770. Both interviewer and interviewee have agreed to release this interview under a Creative Commons Attribution-ShareAlike 3.0 Unported License.¹

Interview

(Interview questions appear in *italics*. Scott had extensive personal archives which he used to answer the initial set of interview questions, and that was followed by an interview on Zoom. The latter has been transcribed, and relevant excerpts have been inserted into the appropriate places in the interview below, shown with a gray background.)

You’ve had an extensive career in technology generally, and games specifically. In particular, you spent nearly 28 years(!) at Electronic Arts, eventually leaving EA as the Vice President Technology. But I want to talk with you about the pre-EA time, when you were the manager of technology at Disclone, a company providing disk duplication and software copy protection services, from 1985 to 1987.

¹<https://creativecommons.org/licenses/by-sa/3.0/>

Let's begin with how you got started with computers. I was intrigued to see that you have training in the culinary arts, which is an unusual way to launch a career in computers and games. There must be a backstory here.

My older brother, myself, and our group of friends were all computer hobbyists in the '70s. We lived in the Atlanta area. Our friend David Durrance's father was an engineer for Southern Bell and built an Altair 8080 in his basement. Around that time, we joined the Explorer Scouts Post 984 which was sponsored by Southern Bell. Our meetings were at a Southern Bell office where we learned how to put together phone networks. We never wore uniforms even when we went to events. We used to set up phone networks for things like the National Canoe & Kayak races.

Our dad taught clinical dermatology at Emory University on his days off from private practice. As a professor, he was given an account on Emory's computer system. He did not know what to do with it so he gave us the credentials. We would go onto campus after hours or on weekends to a terminal room and play Adventure and Star Trek. These were paper-based teletype terminals. Eventually we pieced together our own teletype terminal, a 43 Teleprinter, that we used from home. It used an acoustic coupled modem to dial into the mainframe systems at Emory, DeKalb County School District in Georgia, and I think Fernbank Science Center (which is connected with the school district).

In 1977, the Apple II and TRS-80 home computers came out. Our friends had those so we played around with them. In culinary college at Johnson and Wales, I played around with a friend's Osborne 1, a portable CP/M-based computer. In 1982 I bought my first computer, a Commodore 64, and began learning it inside and out. My brother [Allen], who is two years older, went to Georgia Tech to study computer programming but dropped out since they were so far behind what we had already learned. They were teaching people Fortran programming for banks. I remember one class where Allen was supposed to turn in punch cards of his program for his final. He wrote and tested it on a mainframe, then programmed a punch card writer to create the cards once it was perfected. Allen soon found his way to Silicon Valley where writing in machine code was common. So in short, I learned about computers as a hobby but never thought it was a career path for me.

It seems to me that there were probably some takeaways from being able to organize and manage a kitchen that you probably used as a skill set throughout your career.

Oh very much so. In fact one of the training things that we used to do [at EA] for the advanced leaders, we'd bring them to a restaurant and train them to essentially do a performance in a kitchen and to work with each other because you think about it every day in a restaurant. You've got everything prepped and you're hoping it's going to go right and it's just a production run that entire time. And so the skills you learn about how to work with other people and then how to improvise when things go wrong I think is useful for pretty much any business.

After that, how did you wind up at Disclone? Tell me about being hired there.

By 1985 my mom, stepfather, and older brother were all in California working in Silicon Valley. I had visited many times in the early '80s so I knew something about what was happening. I was working as a kitchen manager for The Country Place Restaurant in Colony Square, Midtown Atlanta. It was one of six restaurants owned by Peasant Restaurants LLC. I'd work crazy long hours then, after work, relax either playing games or programming on my C64. At work we had a Remanco Systems setup for order taking, cook station order printouts, and customer checks. It was written in BASIC, so with a little extra code, I was able to use the orders to do an automated first round of inventory that we did weekly. I still have a printout of the main program that read and wrote to the two floppy drives; it's from one of the printers used at a cook station, 3.5" wide.

One day on the way to work, my 1982 Alfa Romeo Spider convertible broke down. I used a pay phone to call a tow truck then called my GM to say I can't make it in. He said "if you don't make it in, you're fired." I left with the tow truck to get my car fixed then drove it home. I called my brother for advice and he said to just leave and come out to California. Later I got a call from one of the Peasant executives. He said "I hear you're having problems with your GM" to which I said, yeah. . . he fired me or something like that. He then said they were opening up a new restaurant across town and asked if I wanted to be the GM of that restaurant. Fired to promoted in the same day? That's when I knew I had to get out of the restaurant business. I packed up my C64 and some belongings into my Alfa and drove to California that week. My brother said if I made it to San Jose in three days not driving more than ten hours each day, that he would pay for my drinks that night. I got a speeding ticket with a gun drawn by the officer in Arizona but I made the challenge and Allen paid for my drinks. I think it was a week, maybe less that I landed the job at Disclone. I don't remember who made the introduction but it was someone in my family, or a friend of a friend connection.

What were your responsibilities at Disclone?

I was responsible for all copy protection and any IT-related tasks (I created a token ring network for the office). At first that meant contracting to get the copy protection schemes (Formaster program disk, client detection code, and instructions). I then learned how to make my own.

Just to confirm, I believe Disclone was responsible for duplication and copy protection for Electronic Arts games. Is that correct?

Yes, that's correct. EA had used XEMAG for copy protection in the '84/'85 timeframe when my brother worked there. They moved to Disclone because of the PACE copy protection (my brother's company) that we had. I made a spreadsheet of the few EA disk records I have. With dates and part numbers, I can estimate how many EA disks came through. The Disclone part number for a disk started with 184220 followed by a dash and a three-digit disk number. Seven Cities of Gold for Macintosh is part number 184220-009 on October 23, 1986. That means it was the 9th disk we made for them. I started in May of '85 so we probably won their business in August/September of

that year. My last record is for Financial Cookbook on July 22, 1987 as disk number 80. I left in October of 1987 so the total EA disks while I was there was around 90. Take 30% off for two-disk sets and for new betas before replication and I estimated we protected around 60 EA games. It started with Macintosh and Commodore 64 but we also did Apple II, Apple IIGS, Amiga, Atari ST, and PC.

What other notable companies (both game companies and otherwise) do you recall using Disclone's services?

We had over 200 companies in our database. For games the big ones were EA and The Learning Company. There were lots of small game publishers and other big ones in the valley that only used us for some titles.² We also had a lot of business software on the PC that used our protection as well. For non-games and no protection runs, the big ones were ABC Compute Publishing (Gazette and other Compute! magazines), Xerox Desktop Publishing, and AT&T Unix distributors. Large runs were in 10s to 100s of thousands. Small runs were in the 100s to 1000s range.

Where were you located when you were at Disclone? Was this where the production and distribution were done, or was that separately located?

Everything was done at the same location. Our address was 1585 N. Fourth Street, Suite E, San Jose CA, 95112. Note that they now call it 4th street and not written out as "Fourth." You can see the building on Google Maps. When I first started working, we had one of the smaller single warehouse door units toward the back. Inside we had a long room for the duplicators, a small room for the office, and the rest was open. We moved to the front of the building taking up two or three of those large rollup docks. You entered from the other side in what is now marked "A." That was the front desk and waiting area. To the right were the accounting offices. Straight down was first Jay Ebling's office (president) then John Roberts (GM), then my office. Shipping and receiving was behind me taking up the rollup dock bays. In the middle was a large room for sorting, labeling, and packaging. Behind that was the duplication room. We had racks of bins that fit 5.25" and 3.5" floppy disks.

Did you do any networking with people from other duplication and copy protection companies, either normally or perhaps at trade shows?

Yes, constantly. It was a very small group of people and we all knew each other. While we tended to keep the current protection secret, we freely talked about technique and hacks. We also helped each other through bugs and issues. It would not be uncommon for one of the engineers to get together at each other's work place to solve a difficult problem. As for trade shows... I remember one in Santa Clara where across from the Disclone booth was someone I recognized since I had seen him in magazines. His suit was too big and he was introducing Windows, that was Bill Gates. We also did the

²Scott's records indicate that Disclone did duplication for ABC Compute Publishing Inc., Antic Software, AT&T, Brown Bag Software, Creative Software, Datacraft Inc., Electronic Arts, Enlightenment Inc., Human Edge Software, Kyocera Unison Inc., Radarsoft, Synectics Software & Systems, The Learning Company, Wendin Inc., Wyse Technology, and Xerox.

winter CES shows in Las Vegas.

What other services did Disclone provide for clients, beyond disk duplication and copy protection? For example, did they also handle packaging and other production services?

From just disk duplication (blank, no sleeve) to full-service ship-out of final shrink-wrapped product. Whatever the customer wanted.

What kind of disk duplicators were used?

We were a Formaster duplicator shop, one of their biggest customers. These were stacks of boxes that sat on top of tables. The very top one was the controller that ran everything. The ones underneath were the drive boxes, two per box. You slide the disk in and when it was done it would partially pop out. That setup was for small runs where it was OK to use manual labor. For large runs we had autoloaders that we would stack the disks into. The Formaster ones jammed so we ended up using autoloaders from Mountain Computing. That would eventually become the standalone Trace duplicators. You could set the Formaster to do dup and verify in the same run. It would lock and show an error for any disk that was bad. That took twice as long so typically we would do a separate verification run and the customer could say how they wanted it tested (sampling or full 100%). EA typically required a host boot test of some percentage of disks as well.

In a tangential discussion on Zoom, Scott added: Mountain Computer made the better versions of the autoloaders. So Formaster made an autoloader that would jam all the time, a total pain. It was a mechanical nightmare.

You actually have a family connection to Formaster. Tell me about that.

Quite a bit, actually. My stepfather, Dennis Ward, was one of the Formaster engineers that created the Formaster machine. My mom, Nancy Dorrien, ran the duplication department. They were longtime friends with Gerry Pressman, the [Formaster] co-founder, so started early on in the company's formation. On my trips out to California I had attended parties and wine tastings with people from Formaster so knew many of them well. My older brother, Allen Cronce, also worked at Formaster in the format department. That's where he learned everything about how to create program disks that led him to do copy protection. Allen left Formaster to work at XEMAG for Greg Galloway, the largest copy protection duplicator at that time. XEMAG was purchased by Dysan, a disk manufacturer. In 1984, Dysan merged with Xidex Magnetics, another disk manufacturer. Just before I got to California, Allen had left to form his own company, PACE Anti-Piracy, which he still runs today. His first two copy protections were for the Commodore 64 and Macintosh. We used those at Disclone which is one of the reasons we got the [EA] account. The head of EA Technology was Greg Riker who lived across the street from Joe Fontana, another copy protection programmer, who is Allen's CTO today. It was a small circle of people and everyone was connected

in some way. So for Jay Ebling, founder of Disclone, hiring me was less about what I knew but who I knew so I could get the job done.

It sounds like networking was really vital.

Yeah. Yeah it was pretty much all in a valley and everybody knew each other. There was a lot of a movement between companies as well because there was this... this [was] kind of at the beginning of the end of the floppy days and so the consolidation was happening like crazy. So that whole move that, so my brother went from Formaster to, Allen went from Formaster to XEMAG, XEMAG got bought by Dysan, Dysan then merged and became Magnetic Designs or whatever it was. That whole thing happened within about a year and half, maybe two years. And so it was clear that there wasn't a lot of new money coming so people were consolidating their businesses. And that process, that small community of copy protection people were just getting, you know, traded around and didn't particularly care for their new corporate overlords.

[Laughs] You had also said that your mother ran the duplication department at Formaster.

Yeah I didn't know it existed so when I was doing some of these [written answers] I was texting back and forth. And I said, "So did you work at Formaster because I kind of remember you did?" and she goes, "Oh yeah, I ran the duplication department." And I said, "I didn't even know there was a duplication department," so apparently they did duplication as well. I don't know whether it was just a side business or that's where it started and then they started selling machines.

Your relationship that you had between Disclone and Formaster actually sounds fairly tightly-knit; you had the family connections but also you were Formaster's biggest client. Do you think especially given, just in general how tightly-knit this community was, do you think it was fairly common between Formaster and other companies or do you Disclone was unique in this respect?

I think it was the same with XEMAG early on as well until XEMAG became just this merger of three different companies. And mainly because, you know, that was filled with people who came from Formaster as well. So Formaster, the people who could write PDG disks or data disks were spread from Formaster in some fashion or taught by somebody from there into all the other companies that were out there; so that meant they already had those connections. For me it was family connections and I knew them through wine-tasting and things like that. But it was, for others it was their work connections that they had.

So that's why I said it was, you know, you get stuck on a new format you can't figure out or new signatures you're trying to replicate and it's causing problems and you just call up somebody you know and say, "Can you take a look at my code? I don't know what's wrong." And then – so I remember several late nights, usually over drinks in the office trying to figure out why it wasn't working, and was it either a bug or is it a bug in the program or was it just something was

written wrong.

How were the duplicators programmed or otherwise configured for a production run? What sort of computers were used for that task?

The Formaster head cabinet had a PDP-11 inside. It was laid down on its side and connected by a hinge. When you unscrewed the top, you pulled the Q-Bus upright and could access the cards. Besides the PDP-11 card there was also an 8" floppy drive controller connected to the 8" drive cabinet. This made the basis of the PDP-11 system we used to make PDG (Program Disk Generator) data disks. For my setup I had a Falcon card, a faster single board PDP than the default. In my office I had a Digital VT100 terminal connected to the Formaster machine in the duplication area.

You booted up RT-11 on an 8" system disk which included Macro-11, the Pascal compiler, and the Meta macro assembler. You then mounted and ran the Program Disk Generator System from Formaster. To write a copy protection signature in a different format, you needed to write a controller program. The custom disk controller that Formaster invented was based on four 2900 bit-slice chips from AMD (4 bits each, 16 bits total). They were connected in parallel so were very hard to program. These .MAC source files used the Microtech Meta Macro Assembler to generate the 2900 binary files. These source files are very well documented code to explain things in detail since they are very hard to follow from assembly alone. The end result was to write out a PDG data disk. On boot of a Formaster machine, you loaded the PDG data disk to tell it what type of master it would be reading and what type of protection it would be placing on the disks. You then loaded and read in the master disk into memory and started duplicating.

In part of the discussion on Zoom, Scott added: There was one entire program on the PDG disk to compress the data that was going on the disk, so you read it and then compress it because they didn't have enough memory to essentially have a raw disk image sitting around in memory.

With the Formaster controllers, did they actually only give you well-documented assembly source files, or did they supply you manuals as well?

Not so much manuals, pretty much everything was a README file on the disk.

I had seen the references to the 2900 [in the source code] and I presumed it was the AMD2900 but I wasn't 100% sure.

Yeah. And that's the one I'm pretty damn sure that it was for in parallel but I don't have anything that kind of shows that out. Like I remember that – so it was a custom controller that went into the Q-Bus and then that ribbon cable went down to each of the cabinets; and the source code kind of shows that as well. It's really difficult to read because you'll see that, you'll start a compare operation before you actually do any of the math for the compare.

It looks very much like microcode when I read through it.

Yeah, yeah it definitely was. So that was the Meta macro assembler that made it somewhat readable but you'll see tons of text comments on it to try to do that.

I asked Allen whether or not he remembered how many 2900s because I thought it was four, and I said to him, "was it two?" and he said, "I think it was four but you need to ask Kate," so Kate Manbert's another one of our friends that we still know and see. She wrote a lot of the micro-controllers, so when I asked her she said probably four to make it 16 bit but it's so long ago she doesn't remember [laughs]. But she went to, I know that she went to Formaster from some sort of secret satellite project that used 2900s. So they picked her up because they wanted, you know, they wanted her to help with their controller programs. So she wrote a lot of the EPROM stuff for the 2900 controller.

You said you had the VT100 in your office.

Yeah.

It looked like a lot of the Formaster programs were actually fairly menu-driven.

Yeah, yup.

And so the configuration was just going through and punching in the values into the menu options, of course possibly with the custom controller code and—

The way that I think about it, a lot of it was data-driven except for the 2900 custom code. So in the Magnetic Designs one, that source for that one that I did and right before I went to EA, that kind of shows you what a proper meta-language would have looked like if they started from that and worked backwards. But at the time they didn't, you know, nobody really knew how many different types of crazy magnetic disk formats would exist. So when you're at the end of it then it's easy to go back and look at it and say, well it's the encoding type, number of sectors, the error correction method that was used, and then those then all become table-driven and you can have a very short two or three page, you know, piece of data that would tell you exactly what that format looks like.

Because the FreeForm example you sent me was a very elegant description of what you were doing.

Yeah, versus then you look at Formaster and there's all this like precomputation tables trying to figure out exactly how they're going to write each of the data bits and the clock bits and it's just, you know. But you could also skip that because the output of that typically was either a hex file or text file that you could just go modify yourself if you just want to do it quickly. Let's say you wanted just to move a signature to a different sector. All you really need to do is just make a copy of that [program] disk and go hack in the new sector versus the old one in the hex file, then you're done.

Got you.

And that was a lot easier than going through the whole process of creating a brand new program disk from scratch.

And entering in all those values all over again.

Yeah. So it's kind of like peeking and poking in a program to change it, we just do it on the disk because when somebody would figure out "oh the signature is on this track" and you just wanted to move it for a different customer or just make it slightly different it was easy just to make that manual change and have a new disk.

In terms of copy protection and duplication, which computer platforms, operating systems, and disk formats did Disclone support while you were there?

From a physical size point of view we mainly duplicated 5.25" and 3.5". 5.25" disks came in single sided, double sided, single density, double density, and high density media. The three formatting schemes were FM, MFM, and GCR. Each computer manufacturer decided on the number of tracks per side and the number of sectors per track. The Formaster could read and write any of this so it was just a matter of getting the size, density, encoding scheme, tracks, and sectors set on the PDG data disk. The most common disks we replicated for 5.25" disks were Commodore 64, Apple II, Apple IIGS, Atari 800, IBM PC, and Unix. The most common 3.5" disks were Apple Macintosh, Atari ST, Commodore Amiga, and IBM PC. There were dozens of lesser known computer manufactures that we also duplicated, just on lower volume. Given a disk and the disk analyzer program, we could identify and duplicate any type of disk.

How did you stay abreast of software crackers? Did you have people who would study others' copy protection, or did you examine copy programs and hardware that was on the market, or perhaps even have (unofficial) contacts with crackers?

Sophisticated customers like EA would do their own tracking of hacks and provide us with their analysis. Less sophisticated customers would bring a hacked disk to us for analysis. The vast majority of the hacks were software based where the detection code was removed or bypassed. The early copy protection schemes (before I started) were simple signatures like adding an extra sector on a track or making sectors that failed CRC (cyclic redundancy check). Those could be copied by some of the software-based copy programs – we had all of them for testing. The best of them all was Copy II Plus from Central Point Software. Later, Central Point came out with their own floppy controller called the Copy II PC Deluxe Option Board. It went into an IBM PC and you connected it to a 3.5" disk drive. We all had one when we developed new copy protection signatures to make sure it failed.

By this time, we were all using some form of "flaky bits" or "marching bits" signature schemes. They relied on the quirks of the manufacturers' disk controllers to lose sync during reads, but in a predictable way. A typical signature would be a known bit format but written with illegal clock bits for that disk format (like spaced too far apart). The clock bits were invisible to the system, just used by the controller to keep in phase lock

to validate data. This would cause a CRC failure on the sector and a reliable bit pattern once you shifted it left or right to match the known pattern. The Copy II PC Deluxe Option Board could not copy those signatures.

From that point on, hacks moved into attacking the detection code entirely. This escalated into sophisticated detection schemes using techniques like self-modifying code so the detection code was never written out on the disk and was only in memory long enough to do the check. Another technique was to delay the effect of what you did when you found a bad copy. That was so the hacker did not know when the signature detection took place. The next level was to do something non-obvious. Sometimes they would make it so you can't finish a level, sometimes they would make the game eventually crash. This worked on the principle that most hackers did not play the games they hacked, they were more like collectors and distributors of hacked games. Their hacking handle was important so you could damage that if they published a hacked game that didn't work. Could take weeks until people realized it was a bad hack. Those are some of the games we played with the hackers.

I was reminded that you probably had access to these where you were located: there's a magazine which was variously called Hardcore Computist or Computist and I was wondering if that rang any bells in terms of figuring out what crackers were up to, because they would publish parameters that you could use to copy. . . it was focused on the Apple II but it would give you parameters to crack Apple II software. I was wondering if that was one of your forms of intel.

Back then the distribution method typically was either friends of friends, small stuff, or the larger scale or BBS, so bulletin board, and you need to get a credential for somebody to be able to get into the bulletin board and start to download the images that you're going to write onto your own disks. So that was one way that we could go through and find out what people were hacking. That kind of was a never-ending rabbit hole of stuff that worked and didn't work. It was—

You would actually monitor BBSs to find out what had been cracked?

It was more curiosity because it wasn't necessary, we didn't have a real business reason. So it's up to the customer to say, "hey this got cracked I want something new" [laughs]. But early on it was a combination of them figuring it out, so like if you had an early scheme, like you wrote a sector within a sector, that's completely writable if you knew how to do low-level drive code; so like, you know, the Commodore 64, I got my old 64s behind me over here. The 1541 drive was actually more powerful than the 6502 sitting inside the Commodore 64. And it was completely software driven so if you had some knowledge about how that worked you could pretty much write all of the early schemes that were being done on a Commodore 64.

There were some more advanced hackers but they always went after the easiest way which typically was attacking the detection code; so in the beginning it was check the code and if it's bad don't run it. So then you go in and you find out where the check is and you just skip over it and you just jump to, you know,

put in a register that says everything's fine and then jump to starting the game. And that was kind of that first level. Then after that we started saying, "OK you can't just have this sitting out on the disk so somebody can go and change the value on the disk and it's good to go." And we give them bizarre schemes where we are constructing the code to do the detection only in memory, that they can then execute at the right address to go and check and then it clears the stuff out afterwards, after it passes in a register, the pass or fail. That way they never actually – unless they could, unless they had something to stop the machine and examine memory they couldn't tell what we were doing.

Did you develop or oversee the development of any new copy protection techniques while at Disclone? Tell me about those.

Yes, lots of different ones. Flaky bits were probably the most sophisticated ones. That said, we all talked to each other and all knew at a high level the types of signatures possible. While we all developed our own unique signatures, the type of signatures were very similar. Just a lot of trial and error with PDG and the target computer platform.

Handling unprotected master disks containing as-yet-unreleased software must have been like guarding the Crown Jewels. What security procedures were in place to manage them?

This is before the days were a hacker would try to break into a facility to steal secrets (like what would later happen to Sony and Microsoft console development kits). All we used was a locking file cabinet to secure the master disks and PDG data disks. Sometimes a customer would want the master disk back, other times they wanted us to keep it on file for repeat orders. In the duplication room it was either a trusted technician or manager that got the disks, and set up the Formaster machine for the duplication run. That meant the only theft that could happen is someone on the assembly line taking a finished disk home. We had inventory control and a loyal employee base so never had any gray market leaks from our company.

Walk me through how one of Disclone's clients would have gone through the copy protection and duplication process from start to finish.

First they would sign a customer agreement which had confidentiality, ownership, and non-infringement clauses. I would prepare and send them instructions on how to insert the detection code, a disk with the code example, and signature disks so they could test that the protection worked. I would provide support, sometimes a site visit, to help with detection getting the code into their program. For publishers who did not have source code access or whose program was written in BASIC, they would send me their disk(s) and I would put a wrapper around their executable that would check the signature before running their program. After the customer sent us their master disk, we would run sample disks and have them verify they worked before doing a duplication run. The majority of our clients were in the Bay Area and their timelines were very short so

they would drop off their masters and pick up the sample disks. Once approved, they would make an order and we would duplicate the disks. The QC process they wanted was in the order (verification percentage, host boot check, etc.).

Do you recall any clients having “stop-the-presses!” moments where they had to urgently supply an update to their software at the last second? Tell me about some of those.

That was common. We would be in the middle of a duplication run when the customer would call and said they found a bug and would have another master to us soon. They were paying for what we replicated so not an issue for Disclone. Those mistakes became costlier for them if we had already packaged and shrink-wrapped their software. That happened occasionally as well. I can remember the entire assembly line tearing open and pulling out the disks one late afternoon after we got the call.

When you had a client say, “hey wait I need to have all these redone,” was this an all-hands-on-deck moment or was this limited to the assembly line staff when that happened?

Well they needed to catch us during business hours otherwise they’re screwed, and if we were doing a big run and we had a swing shift or a night shift then it would be all done and they’d just have to pay to redo it [laughs]. Typically what happened because they were on a very tight schedule and they were willing to take the risk that they would find a bug while they’re finishing their testing process but start the replication anyway because if it would, you know, if it’s a three or four day job and we’re doing, you know, several hundred thousand copies of something they get it started and then it’s like “oh crap this is a fatal bug.”

Then having gone to EA I could see kind of what it was. There’s a negotiation that goes on between the group that has made the game and the QA organization that is saying, “no, no this is a fatal bug we can’t let it go” and they’re going, “no, no it’s actually not fatal for these reasons.” And there’s an argument that goes back and forth until you decide “all right are there just enough bugs now that I have to do a new master.” Now that all faded away as things became online because now, you know, you see a game come out and there’s a day one patch.

Sure, yeah.

That’s the same thing. The reason you have a day one patch is because they’ve got to go through with it and get this disk out and there’s a timeline for pressing of the disk and now we’ve got X number of weeks before we could have a day one patch that will fix whatever critical issues are found in the meantime. In this case that’s, you know, when you put out a disk that was it, there was no way of patching it.

When these were being redone were the disks actually erased before reimaging or did you just reblast them?

No, you just write over them because it does an erase, so there's a format that happens first before it actually writes the data.

Oh, OK.

So it's a format, write, and then optional 100% verify.

I've heard rumors that technology doesn't always work perfectly. What things tended to break or otherwise go awry during duplication? What preventative measures did you take to head them off?

We had the copy protection pretty much covered with the customer signing off that the disks worked for them, and us doing a boot test during the replication to verify we used the correct PDG disk. The work orders had all the part numbers but it was still possible for human error in programming the Formaster machines. The most common problem was mechanical problems like disks jamming in the autoloaders. They were pretty finicky machines and needed constant maintenance and adjustments. The 5.25" jams would fold the disks in half, the 3.5" jams were normally tolerance issues with the thickness of the disks. The other problem we had were a bad batch of disks from a manufacturer. If we had too many disk errors on write, we would send the full batch back. If it happened too often, we would stop using that disk supplier.

Do you recall any improvised hardware, any jury-rigging of things that went on?

All these mechanical things had problems, so we would go in and, you know, instead of going back to a manufacturer trying to get something working we would go and make our own mods to it; and so sometimes it's like, well the bands they were using weren't strong enough and they would only last for a few weeks and they'd have to change it, so you find another supplier that could keep it tighter; or that the rolls for the printing were too hard so you could find other ones that were softer that would have a better effect and not crush the disk as they go through; so it's just all those kinds of things. Yeah, the mechanical parts of all of it were just a constant fiddling to find out what worked.

I had a good laugh when I read that when the 5.25" ones jammed they'd just fold the disks in half, that was [laughs]

[Laughs] Yeah, and we bought everything in bulk so that a case was maybe like, let's call that three feet by two feet, so it's usually two rows of 5.25" disks that come in, and they would just be stacked as is and wrapped in plastic, and those are the bulk disks that you get and you load them into the loader and, you know, hope you didn't fold too many of them.

Tell me a story of something out of the ordinary that happened at Disclone.

Disclone and Formaster got into a legal dispute. We had bought some parts from a 3rd-party technician and Formaster said they were not authorized to do that. I think

they even alleged he stole the parts. That was my first introduction into the legal world. I remember being very nervous on my way over to a law office for my deposition. When I got there, I was warmly greeted by Formaster co-founder Gerry Pressman. Gerry and his wife Stephanie were old friends of mine from our monthly wine tasting group. Seeing his smile made me relax. Disclone and Formaster were reliant on each other so the case was settled. I think we gave back the parts and bought them from Formaster.

What were you most proud of accomplishing during your time at Disclone?

I'm just proud to have played a small part in the copy protection business. That was my first job as a programmer. I had traded my culinary arts profession for my self-taught programming hobby. That was a life-changing event for me. I found myself making more money than I had ever made before and that first year, my brother and I bought a house together. That helped us trade rent for equity and pave our way for future homes in California. At the time I did not know how lucky I was.

How did your time at Disclone come to an end?

I was watching the consolidation of the disk market and the beginning of CD-ROMs. The equipment was too expensive for home PCs but that was just a matter of time. I knew that CD writers were years off so no copy protection would be needed. Disclone did not have the capital to become a CD-ROM manufacturer (clean rooms and 10s of millions in equipment investment). Disclone was still at their height of customers and sales so I was in no rush to leave but one day an opportunity came up that I had to take.

We had a meeting with a new client who was forming a software gaming company. Tom Frisina, former co-founder of Accolade, and Jim Rushing, famous programmer from Ozark Software (MULE, Seven Cities of Gold, etc.). They took my card and soon after I got a call asking if I wanted to join their company. I found a replacement for my job at Disclone and attached his resume to my resignation letter. I joined Three-Sixty Pacific as an Associate Producer. That was in October of 1987 and about six weeks later we shipped our first games, ports of the famous Mac game, Dark Castle. We did the Amiga, Apple IIGS, Atari ST, Commodore 64, DOS, and Spectrum versions. We then did Thud Ridge (Vietnam flight sim) for C64 and DOS. We also did two non-games. Fire Rescue Plus was a PC file recovery tool and we did a Myers-Briggs personality utility. We then did Harpoon with author Larry Bond. The developer chosen for this was the son of one of our investors who was not able to finish the game. Jim and I had to live in Bryan, Texas to get the game done. That process burned me out and I left Three-Sixty in October 1988.

I was not sure I wanted to be in the game business after that but I thought I would interview with EA to see if they were different. Greg Riker had been after me since I left Disclone. I interviewed for a Technical Director position with three different producers. Two of them I had no interest in working with, but I got along with Richard Hilleman right away. I told Greg that but he said he did not have a position open under Rich. I said that's fine. . . call me when there is one. Two weeks later I was hired and

have been working with Rich ever since. Jim Rushing soon after left Three-Sixty and came to EA. We are still friends today as well.

What haven't I asked you about that time at Disclone that I should have?

You didn't ask me about Disclone's revenue. I remember that we made around \$5m in fiscal 1986. We were a private company. I think that was more than Formaster was making and they were a traded public company. Shows how the business was better in the per-disk fees for duplication than making the duplication equipment.

Formaster and these other companies must have had a fairly small, easy to saturate market. I mean there can't have been that many people buying them.

Yeah, and we're getting paid per disk, and in some cases multiple times when you've got a rush job and you have to redo the entire thing. But also we had automatic label machines, and occasionally they [the clients] would come and supply their own labels and not on the rolls and in the right format for our machines so then those would have to be hand-applied which then adds cost to it. Usually that would only happen once and they'd go, "OK we'll give you the design and you go print them" so that we could have these rolls of the 5.25" or the 3.5" that would automatically get applied; you'd put a stack in and it would just go through the entire thing. You'd still have to go through and make sure there wasn't any misalignment. If there was misalignment you'd take it off and put another one on but for the most part that was automated as well.

That actually pretty much exhausts all the follow-up questions I had since you did such a good job at writing up answers.

Cool, cool. I think there's still more for me to remember and I'm wondering what I'm going to forget now that I've remembered all this stuff [laughs].